



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

## TERMO DE REFERÊNCIA TI

09963660996366

Pregão Eletrônico n.º XX/2019

Processo Administrativo SEI n.º 23038.002787/2019-31

\*Este Termo de Referência segue as orientações contidas na Instrução Normativa n.º 1, de 4 de abril de 2019.

O presente documento tem por objetivo descrever os elementos necessários e suficientes para subsidiar o processo licitatório para contratação do objeto a seguir exposto, demonstrando sua viabilidade e conveniência, tudo com sustentação no que se extraiu do Estudo Técnico Preliminar - ETP, devidamente aprovado pelos integrantes da equipe de planejamento da contratação competentes para tanto.

Utilizou-se o modelo de minuta padrão para **Pregão - Serviços continuados com dedicação de mão de obra exclusiva** [https://www.agu.gov.br/page/content/detail/id\\_conteudo/714621](https://www.agu.gov.br/page/content/detail/id_conteudo/714621).

Em atendimento às orientações da COTA n. 00217/2018/NLC/ETRLIC/PGF/AGU, de 20 de dezembro de 2018, foram identificadas, por meio da utilização de letra de cor vermelha, as inserções divergentes do modelo da AGU, as quais se justificam pelas características específicas da CONTRATANTE e do objeto a ser contratado, as quais, se não constantes do TR, levariam a CAPES a uma licitação parcialmente inconsistente, considerando o que a motiva.

## 1. DO OBJETO

1.1. Contratação de empresas especializadas para prestação de serviços técnicos de Tecnologia da Informação, compreendendo desenvolvimento de software com práticas ágeis, na forma de serviços continuados e presenciais com dedicação exclusiva de mão de obra, e mensuração de sistemas de informação, na forma de serviços não presenciais mensurados por Pontos de Função, conforme condições, quantidades e exigências estabelecidas neste instrumento:

Grupo	Itens	Serviço	Quantidade
1	1	Analista Programador Sênior	28 Postos de trabalho
	2	Analista Programador Pleno	18 Postos de trabalho
	3	Analista Programador Júnior	6 Postos de trabalho
2	1	Mensuração de Sistemas	24.000 Pontos de Função Anuais

1.2. O objeto da licitação tem a natureza de serviço comum de tecnologia da informação e comunicação, uma vez que suas qualidades e seus atributos são predeterminados, com características invariáveis ou sujeitas a diferenças mínimas e irrelevantes. Configura-se como **única solução de TIC**, na forma do art. 3º, inciso I, da IN n.º 1, de 04 de abril de 2019.

1.3. Considerando a interdependência dos serviços do objeto a ser licitado, fica estabelecido que os serviços devam ser prestados por diferentes empresas, nos termos do Art. 4º da IN nº 01/2019, não sendo possível à contratação da mesma empresa para execução dos serviços do GRUPO 1 e do GRUPO 2.

1.4. Os quantitativos e respectivos códigos dos itens são os discriminados na tabela acima.

1.5. A presente contratação adotará como regime de execução a empreitada por preço unitário, de acordo com as especificações e os padrões de desempenho e qualidade estabelecidos pela CAPES, mediante Ordens de Serviço (OS), limitadas ao quantitativo máximo estimado, sem garantia de consumo mínimo.

1.6. O contrato terá vigência pelo período de 12 meses, podendo ser prorrogado por interesse das partes até o limite de 60 (sessenta) meses, conforme inciso II do art. 57 da Lei nº 8.666, de 1993, respeitadas as características técnicas e condições deste TR.

## 2. JUSTIFICATIVA E OBJETIVO DA CONTRATAÇÃO

2.1. A Justificativa e objetivo da contratação encontram-se pormenorizados em Tópico específico dos Estudos Preliminares, apêndice desse Termo de Referência.

2.2. Todavia, entende-se por bem inserir, também no TR, extrato da justificativa constante do ETP.

2.3. Considerando o papel desempenhado pela CAPES nas políticas educacionais determinadas pelo PNE, a utilização de Tecnologia de Informação e Comunicação (TIC) tem importantíssima participação, pois busca sistematizar ações e programas, maximizando os resultados pretendidos e melhorando a divulgação das informações de domínio público ou dos próprios programas institucionais.

2.4. As aplicações e os sistemas desenvolvidos e mantidos atualmente pela área de TIC da CAPES apoiam a missão crítica da Instituição e são fundamentais para o adequado funcionamento da Fundação, de forma que a interrupção no funcionamento ou indisponibilidade deles, implica na paralisação das atividades, causando potenciais prejuízos a bolsistas, pesquisadores e usuários desses sistemas, além de afetar as organizações públicas e privadas que dependem de alguns desses serviços para a realização de atividades da CAPES.

2.5. Nesse sentido, a fim de colaborar para o cumprimento das missões institucionais, e em pleno atendimento às suas competências, a Coordenação-Geral de Sistemas (CGS) da DTI desenvolve e mantém 118 aplicações de sistemas de informação, que permitem a execução e a gestão dos processos de concessão de bolsas, gestão e avaliação de cursos de pós-graduação e acesso a publicações nacionais e internacionais, dentre outros.

2.6. As referidas aplicações e os sistemas, além das Plataformas e do Portal da CAPES, são desenvolvidos e mantidos pelas equipes da CGS em concordância com a Metodologia de Desenvolvimento de Sistemas (MDS), ENCARTE C - METODOLOGIA DE DESENVOLVIMENTO DE SISTEMAS (Sei nº 0983589). A utilização de MDS para o desenvolvimento de sistemas, além de otimizar recursos e potencializar resultados, tanto em rapidez quanto em qualidade, tem por intenção nortear o processo de desenvolvimento e manutenção de software, contemplando todo o ciclo de vida do projeto, desde sua concepção até a sua entrega em ambiente de produção. Assim, a implantação de metodologias que tragam resultados com qualidade, no menor tempo possível e tragam a ideia de inovação adequando a CAPES às melhores práticas do mercado, auxiliam, além dos técnicos da CGS, a todas as entidades e fornecedores envolvidos no processo de criação e manutenção de sistemas, principalmente as áreas demandantes da Fundação.

2.7. Convém anotar que o ciclo completo de desenvolvimento de software da CAPES, incluindo o levantamento de cenários, atual e futuro, está documentado na nota técnica nº 11/2018, contido no processo SEI nº 23038.003397/2018-05. O referido documento contém informações sobre as

metodologias de desenvolvimento, os perfis profissionais, os fluxos, as atividades compreendidas no modelo, abordando a atual MDS versão 3.1, sua transição e cenário desejado com adoção da MDS 4.0, os resultados mais importantes obtidos com a utilização da MDS 3.1, as lições aprendidas em razão das dificuldades encontradas, e a forma como o modelo de desenvolvimento de sistemas da Fundação se reflete, abordando metodologias ágeis, nos contratos de prestação de serviços.

2.8. Como se vê, a iniciativa de se revisar a MDS, visando a definição de um novo modelo a partir de métodos ágeis, está integralmente alinhada às mais eficientes tendências mundiais e nacionais nessa área da TIC. Nesse sentido, a nota técnica nº 11/2018 faz referência à MDS 4.0, que foi desenvolvida para adoção de técnicas e práticas ágeis para entregas com maior qualidade em alinhamento com as necessidades da CAPES.

2.9. Durante os meses de julho e agosto de 2018, foi realizada uma consultoria pelo Instituto CESAR, por solicitação da CAPES. O trabalho visava atingir alguns objetivos, os quais se destacam o fornecimento à Diretoria Executiva uma visão comprensiva dos problemas e relações que impactam a qualidade dos serviços de TIC entregues às áreas, bem como apresentar um conjunto de recomendações organizacionais e técnicas para superar os problemas identificados, bem como reforçar as boas práticas existentes e ações planejadas. Como resultado desta consultoria, foi construído um documento de Análise Crítica de Fatores de Sucesso para Serviços de TIC, no qual são feitas recomendações à DTI nos seguintes aspectos:

2.9.1. *Qualidade dos Serviços de TIC da CAPES*

2.9.1.1. *Melhorar visibilidade e comunicação pela criação de processo unificado de submissão, acompanhamento e avaliação de demandas de alto nível;*

- *O processo atual de solicitação e acompanhamento de demandas de alto nível, que exigem o desenvolvimento de novos módulos de software com múltiplas funcionalidades, torna complexo o rastreamento de requisitos, o acompanhamento do status de ações e a comparação do progresso versus o cronograma planejado. De forma geral, pode ser argumentado que um dos componentes do desafio de comunicação entre diretorias finalísticas e DTI reside na ausência de uma ferramenta que efetivamente promova esta visibilidade.*
- *O processo SCRUM, adotado dentro da MDS CAPES, prevê o acompanhamento frequentes das atividades de desenvolvimento e a validação de resultados ao final de curtos períodos de trabalho, chamados sprints. A revisão dos resultados de cada sprint gera correções de curso, se necessário, e planejamento de ações subsequentes.*
- *As equipes da DTI já utilizam ferramentas de suporte ao acompanhamento de atividades compatíveis com o processo SCRUM, mas não de forma unificada. A criação de um processo de submissão, acompanhamento e avaliação de demandas de alto nível apoiado ne stas ferramentas, onde comitês formados por representantes das diretorias demandantes e DTI pudessem submeter conjuntos de demandas (backlogs, sprint planning), acompanhar a evolução do trabalho em um quadro de atividades (SCRUM board) e avaliar os resultados parciais (sprint review) ajudaria na resolução de vários dos desafios identificados neste estudo.*

2.9.1.2. *Implementação de processo iterativo e incremental de construção de soluções;*

- *Parte dos comentários das diretorias finalísticas, como desafios nos aspectos de usabilidade, poderiam ser endereçados pela implementação efetiva de um processo iterativo de construção de soluções, onde o ciclo de validação pudesse acontecer de forma frequente. Desvios entre a ideia de produto dos demandantes e o sistema construído seriam identificados precocemente, evitando retrabalho e frustração de expectativas.*
- *A disciplina de desenvolvimento ágil de software menciona iterações, colaboração com demandantes e adaptação à mudança como princípios norteadores. Em sua versão 4.0, a Metodologia de Desenvolvimento de Software da CAPES estabelece o processo SCRUM e, entre outros, o papel do Product Owner (PO) como "profissional que representa os interesses do negócio" durante o processo de desenvolvimento.*
- *Apesar da existência da MDS revisada em 2018 prevendo métodos ágeis e de um cronograma de implantação previsto para ser concluído em julho de 2019, ressaltamos a urgência da implantação dos novos aspectos da metodologia, tendo em vista os benefícios esperados, diretamente endereçando parte dos desafios identificados.*

2.9.1.3. *Implementação de abordagens de integração contínua integradas aos processos de Mudança, Desenvolvimento e Testes da MDS;*

- *Relatos colhidos durante a imersão indicam a ocorrência de situações onde problemas funcionais nos sistemas desenvolvidos chegam aos ambientes de homologação e produção. Mesmo considerando a influência de condições externas, como prazos exígues para demandas estratégicas que implicam em redução de esforço de testes, o impacto negativo de falhas detectadas por usuários externos é avaliado como muito alto.*
- *A abordagem de integração contínua prevê que o desenvolvimento de funcionalidades de software seja feito de forma paralela a testes automatizados nos níveis unitário, de carga e de integração. Um servidor de integração executa de forma automática o conjunto de testes para cada iteração de desenvolvimento, que deve idealmente acontecer pelo menos uma vez ao dia.*
- *Ao criar um ciclo curto entre uma submissão de novo código e a identificação de falhas, é esperado uma redução de falhas que cheguem até etapas de homologação ou produção, além de evitar que a equipe de testes precise fazer a execução manual de validações, liberando capacidade produtiva para análises mais refinadas de qualidade.*

2.9.1.4. *Implementação abordagens de entrega contínua de software integradas aos processos de Mudança, Desenvolvimento e Testes da MDS.*

- *Durante o processo atual de desenvolvimento de soluções na CAPES, a transição de ambientes para cada estágio (codificação, testes, homologação e produção) acontece de forma não automatizada, exigindo esforço manual e abrindo possibilidade para problemas de configuração e consequentes falhas de execução.*
- *A abordagem de entrega contínua prevê a automação de todo o processo de entrega, incluindo construção do executável com base no código mais recente; configuração do ambiente de execução, preferencialmente usando suporte de máquinas virtuais ou containers; e testes de sistema, garantindo a sanidade do sistema entregue.*
- *Ao adotar as abordagens de entrega contínua as equipes de desenvolvimento e infraestrutura são liberadas de esforço manual, tendo em vista a automatização dos passos para entrega. Ao mesmo tempo, as janelas de tempo de possíveis entregas são ampliadas, flexibilizando a disponibilização de novas versões dos sistemas para os usuários finais.*

2.9.1.5. *Alinhamento de responsabilidades no processo de agregação, disponibilização e homologação de dados da CAPES;*

*Previsibilidade*

*A grande quantidade de sistemas não integrados, com projetos heterogêneos de bases de dados, herdados e mantidos pela DTI, são responsáveis também pela baixa produtividade das equipes de manutenção desta Diretoria. Para esta realidade objetiva, recomendamos um sobre esforço visando:*

- *Normalização das bases de dados de forma a tornar eficiente os processos de migração de dados;*
- *Redução do número dos sistemas mantidos, unificando aqueles de funcionalidades semelhantes, como exemplo, convergir para o SCBA a gestão dos pagamentos de todos os Programas.*

*Recomendamos que a DTI publique na intranet da CAPES, em tempo real, o andamento das atividades dos projetos em desenvolvimento, notadamente aqueles demandados pelas Diretorias usuárias. Este aplicativo de publicação deverá estar integrado ao aplicativo de controle das atividades de fábrica da DTI. Desta maneira as áreas usuárias (stakeholders) acompanharão o desenvolvimento e rastreabilidade de suas demandas.*

*Para todos os projetos, devem ser publicados seus indicadores de desempenho, bem como análises de desvios das linhas de base (caso existam), causas e consequências.*

#### 2.9.1.6. Gestão de Demandas de Serviços de TIC

*Entendemos ser prioritário modelar e transformar o processo de Gestão de Demandas de Serviços de TI por todo seu ciclo de vida, devendo ser apoiado por um aplicativo robusto. A DTI detém toda competência para este projeto. É fundamental a visão outside in que privilegie o cliente.*

*Neste contexto os usuários precisam entender perfeitamente que serviços podem demandar à DTI. Para isto a DTI deve comunicar aos usuários que serviços pode atender dentro de suas atribuições normatizadas. Cabe então à DTI estruturar e publicar o Portfólio de Serviços.*

*O Portfólio da DTI/CAPES deve ter o objetivo de comunicar, aos seus usuários, os serviços, sistemas e softwares disponíveis ou utilizados no ambiente da TI, bem como os projetos executados ou em curso, de modo a agilizar o atendimento de suas necessidades e demandas por serviços e recursos de tecnologia da informação.*

*O Portfólio deve ser composto por:*

*Catálogo de Serviços de TI – apresenta uma visão geral de cada serviço prestado, sua descrição, algumas informações importantes e os canais de prestação do serviço.*

- *Os Serviços devem ser agrupados por natureza, como: Serviços de Rede e Internet, Cadastros e Acessos, Serviços de Impressão, entre outros.*
- *Cada Serviço deve ter minimamente: sua descrição, forma de solicitação, prazo de atendimento (SLA).*

*Catálogo de Sistemas e Aplicativos – apresenta uma visão sucinta e os links para acesso aos principais sistemas e aplicativos desenvolvidos, ou sob responsabilidade da DTI, que suportam ou apoiam as áreas finalísticas da CAPES.*

*Projetos – apresenta uma visão geral do portfólio de projetos em suas diferentes fases de planejamento ou execução pela DTI.*

#### 2.10. Da justificativa da contratação

2.10.1. A Coordenação Geral de Sistemas (CGS) resultou da alteração na estrutura organizacional da CAPES pelo Decreto nº 7.692, de 2 de março de 2012, que aprovou o novo estatuto e o quadro Demonstrativo dos Cargos em Comissão da CAPES. A CGS está inserida na estrutura interna da Diretoria de Tecnologia da Informação (DTI), e a ela compete:

Planejar, desenvolver, implantar e manter atualizados os sistemas de informação da CAPES;

Especificar e supervisionar os serviços de desenvolvimento, operação e manutenção de sistemas de informação da CAPES prestados por terceiros, dentro de normas e padrões estabelecidos pela DTI, bem como garantir o desempenho e correto funcionamento dos sistemas implantados;

Realizar a modelagem, automação e otimização dos processos, procedimentos e rotinas de trabalho através da utilização de tecnologias da informação;

Estabelecer os padrões, modelos e metodologias de desenvolvimento de sistemas para a manutenção e construção de sistemas de informação na CAPES;

Estabelecer os padrões e modelos de arquitetura de sistemas de informação utilizados pelos sistemas de informação da CAPES;

Estabelecer os padrões e tecnologias para a integração entre sistemas de informação da CAPES e de órgãos externos, seguindo as recomendações e diretrizes da Diretoria de Tecnologia da Informação e em concordância com os Padrões de Interoperabilidade de Governo Eletrônico (e-PING);

Estabelecer critérios tecnológicos, de design e de acessibilidade utilizados nos portais corporativos da CAPES, seguindo as recomendações e diretrizes da Diretoria de Tecnologia da Informação e em concordância com o Modelo de Acessibilidade de Governo Eletrônico (e-MAG);

Planejar, promover e executar treinamentos de capacitação referentes aos sistemas de informação e serviços disponibilizados aos usuários da CAPES;

Auxiliar a Diretoria de Tecnologia da Informação na proposição de capacitação e atualização técnica dos recursos humanos envolvidos nos projetos de desenvolvimento de sistemas de informação e modernização tecnológica;

Auxiliar a Diretoria de Tecnologia da Informação no planejamento, elaboração e execução de planos estratégicos de tecnologia da informação;

Apoiar a Diretoria de Tecnologia da Informação na definição e implantação de normas e procedimentos para a administração dos recursos de tecnologia da informação disponíveis na CAPES;

Acompanhar e fiscalizar contratos e convênios relativos à área de tecnologia da informação, verificando a correta utilização dos recursos computacionais e tecnológicos;

Apoiar as demais áreas da CAPES quanto à utilização de ferramentas e tecnologias para elaboração de relatórios gerenciais e de apoio a decisão solicitados pelos dirigentes da instituição; e

Propor a adaptação dos instrumentos, procedimentos e rotinas de trabalho à evolução da tecnologia e do respectivo modelo de informações.

2.10.2. Considerando a extinção do contrato atual pelo término de vigência, e considerando a inexistência na CAPES de equipe própria para atuar na prestação dos serviços, surge a necessidade de se contratar os serviços de desenvolvimento e de manutenção de sistemas de informação prestados por uma empresa de mercado, além da necessidade natural de atualizar os dispositivos de controle que haviam sido estabelecidos anteriormente. Diante disso, faz-se necessária a realização de processo de contratação para prestação de serviços de desenvolvimento e manutenção de sistemas, com vistas à continuidade do negócio da CAPES.

### 3. ALINHAMENTO ESTRATÉGICO DA CONTRATAÇÃO

3.1. Em sintonia com o PDTIC 2017-2019 da CAPES, essa contratação apresenta a seguinte vinculação:

ID	OBJETIVO ESTRATÉGICO	ID	NECESSIDADE PREVISTA NO PDTIC	ID	AÇÃO NO PDTIC
OB3				AE10	Aprimorar o sistema único de pagamento de bolsas e auxílio
				AE15	Aprimorar o sistema único de prestação de contas de beneficiários dos programas da CAPES
				AE16	Ampliar o sistema de consulta aos documentos históricos da pós-graduação

			AE17	Desenvolver sistema único de gestão de programas da CAPES
			AE19	Desenvolver sistema para lançamento de edital e seleção de bolsistas para os programas da CAPES
			AE20	Desenvolver sistema para consulta aos documentos históricos de bolsas
			AE21	Desenvolver interface única para os sistemas da CAPES
			AE23	Desenvolver Portal de Egressos da CAPES
			AE11	Integrar os sistemas de pagamento com a base do SIPREC
			AE12	Integrar a Plataforma Sucupira aos demais sistemas da CAPES
			AE13	Integrar os sistemas de pagamento ao SEI e ao SIAFI
			AE14	Mapear e implantar o processo de acompanhamento dos programas de pós-graduação na Plataforma Sucupira
			AE18	Desenvolver central integrada de gestão de fluxos (SEI-SCDP-SAE)
		N7	AE22	Aprimorar a integração do SIGRH com o SIAPE
		N12	AE28	Implantar solução para criação da base de conhecimento da CAPES
			AE37	Desenvolver sistema que integre as bases de dados dos programas de formação docente da CAPES
		N13	AE42	Disponibilizar serviços digitais aos usuários externos por meio de dispositivos móveis e inteligentes
			AE29	Mapear e melhorar o processo APCN na Plataforma Sucupira
			AE31	Migrar bolsistas do SAC País para o SCBA
			AE32	Implementar melhorias no Cadastro de Consultores (CadCons)
		N14	AE38	Adequar os sistemas de inscrição, análise, concessão e pagamento de bolsas e auxílios para o programa Mais Ciência Mais Desenvolvimento
			AE39	Mapear e implantar melhorias no processo de liberação para pagamentos no SCBA
			AE40	Adaptar os sistemas ao vocabulário comum construído pelas áreas finalísticas da CAPES
			AE43	Sistematizar notificação de solicitação de pagamento ao interessado
			AE44	Migrar inscrição, análise e pagamento do PIBID para o SICAPES 3, SAP e SCBA
OB6	Desenvolver e sustentar os sistemas com qualidade e inovação	N15	AE40	Desenvolver e adaptar os conteúdos digitais da CAPES, em conformidade com a legislação em vigor, em especial ao modelo de acessibilidade de Governo Eletrônico
		N16	AE41	Realizar manutenção contínua dos sistemas e aplicações da CAPES

#### 4. DESCRIÇÃO DA SOLUÇÃO DE TIC

4.1. A descrição da solução como um todo, conforme minudenciado nos Estudos Preliminares, abrange a prestação de serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação, conforme especificações contidas neste TR.

4.2. Em síntese, o modelo de contratação prevê que a empresa contratada para o GRUPO 1 proverá equipe de desenvolvimento para prestação do serviço sob demanda, que será formalizada por meio de ordem de serviço – OS durante período a ser definido, de acordo com o projeto a ser executado, com a composição e qualificação mínimas exigidas e será remunerada pelos postos de trabalho mediante atendimento de níveis mínimos de serviço previstos no instrumento convocatório. Ou seja, caso haja descumprimento de nível mínimo de serviço, haverá desconto no valor ofertado pela contratada, de acordo com as regras estabelecidas no edital.

4.3. A empresa contratada para mensuração de sistemas para o GRUPO 2 irá validar a quantidade de Pontos de Função desenvolvidas pela empresa do GRUPO 1, e será remunerada pelos Pontos de Função contados. Os pontos de função mensurados irão compor o nível mínimo de serviço da empresa do GRUPO 1.

## 5. DA CLASSIFICAÇÃO DOS SERVIÇOS E FORMA DE SELEÇÃO DO FORNECEDOR

5.1. Trata-se de serviço comum de caráter continuado, com fornecimento de mão de obra em regime de dedicação exclusiva, a ser contratado mediante licitação, na modalidade pregão, em sua forma eletrônica.

5.2. Os serviços a serem contratados enquadram-se nos pressupostos do Decreto nº 9.507, de 21 de setembro de 2018, não se constituindo em quaisquer das atividades previstas no art. 3º do aludido decreto, cuja execução indireta é vedada.

5.3. A prestação dos serviços não gera vínculo empregatício entre os empregados da CONTRATADA e a Administração CONTRATANTE, vedando-se qualquer relação entre estes que caracterize pessoalidade e subordinação direta.

## 6 REQUISITOS DA CONTRATAÇÃO

6.1. Conforme Estudos Preliminares, os requisitos da contratação abrangem o seguinte:

6.1.1. Contratação de empresa especializada para prestação de serviços de desenvolvimento de software com práticas ágeis e mensuração de sistemas de informação, na modalidade fábrica de Software Ágil, na forma de serviços continuados presenciais e não presenciais, com dedicação exclusiva de mão de obra, conforme condições, quantidades e exigências estabelecidas neste instrumento.

6.1.2. Quanto aos requisitos financeiros necessários para o atendimento da necessidade, a planilha abaixo indica a previsão orçamentária elaborada com base em cotações de mercado, consideradas as especificações produzidas e seguindo as orientações da IN SLTI/MP 05/2014 alterada pela IN 03/2017. As despesas para atender a esta licitação estão programadas em dotação orçamentária própria, e correrão por conta dos recursos orçamentários constantes do Orçamento Geral da União, aprovado pela LOA - Lei Orçamentária Anual para o exercício de 2019.

Custo Total da Solução						
Grupo	Item	Fonte do Recurso	Natureza da Despesa	2019 (X meses)	2020	2021
1	1	8100	44.90.40.01	R\$	R\$	R\$
	2	8100	44.90.40.01	R\$	R\$	R\$
	3	8100	44.90.40.01	R\$	R\$	R\$
2	1	8100	33.90.40.21	R\$	R\$	R\$

## 6.2. Critérios e práticas de sustentabilidade

6.2.1. Os critérios práticas de sustentabilidade adotadas nessa contratação estão descritos no item 30 deste TR.

## 6.3. Necessidade de transição gradual com transferência de conhecimento, tecnologia e técnicas empregadas

6.3.1. Estão previstas neste TR no item 8 Modelo de Gestão e Execução do objeto.

6.4. Declaração do licitante de que tem pleno conhecimento das condições necessárias para a prestação do serviço.

6.5. As obrigações da Contratada e Contratante estão previstas nos itens 13 e 14 deste termo de Referência.

## 7. VISTORIA PARA A LICITAÇÃO

7.1. Para o correto dimensionamento e elaboração de sua proposta, o licitante *poderá* realizar vistoria nas instalações do local de execução dos serviços, acompanhado por servidor designado para esse fim, de segunda à sexta-feira, das 9:00 às 11:00 e 15:00 às 17:00.

7.2. O representante legal da CONTRATADA deverá apresentar-se no ato da visita munido de procuração que comprove estar autorizado e habilitado a representar a CONTRATADA.

7.3. A CONTRATANTE, por intermédio de Servidor Público especificamente designado para este fim, irá fornecer Atestado de Visita Técnica conforme modelo apresentado no Encarte Técnico Contratação - TI CGS (1006100), constante nos encartes comprovando que a CONTRATADA licitante visitou as instalações.

7.4. A visita técnica deverá ser realizada até o último dia útil anterior à data do pregão, no endereço e horários a seguir: Local: Setor Bancário Norte, quadra 2, bloco L, lote 6, sobreloja DTI. Brasília – DF.

7.5. A visita de vistoria técnica é uma faculdade dos interessados e sua ausência não restringe a participação dos licitantes.

7.6. A não realização da vistoria, quando facultativa, não poderá embasar posteriores alegações de desconhecimento das instalações, dúvidas ou esquecimentos de quaisquer detalhes dos locais da prestação dos serviços, devendo a licitante vencedora assumir os ônus dos serviços decorrentes.

7.7. A licitante deverá declarar que tomou conhecimento de todas as informações e das condições locais para o cumprimento das obrigações objeto da licitação.

## 8. MODELO DE GESTÃO DO OBJETO

8.1. A execução do objeto seguirá a seguinte dinâmica:

8.2. A execução dos serviços será iniciada por ordem de serviço, a partir da data de sua assinatura contratual.

### GRUPO 1 - Desenvolvimento de Sistemas:

8.2.1. A empresa CONTRATADA deverá prestar serviço de desenvolvimento seguindo a Metodologia de Desenvolvimento de Software vigente na CAPES para execução de projetos de desenvolvimento, sustentação e manutenção de sistemas, que prevê a adoção de práticas ágeis e frameworks como Scrum e Kanban, respeitando os padrões técnicos adotados na CONTRATANTE.

8.2.2. Os serviços serão prestados por meio de equipe técnica da empresa CONTRATADA disponibilizada exclusivamente para atender à CONTRATANTE. Esta equipe técnica irá compor o time de desenvolvimento, previsto na MDS, ENCARTE C - METODOLOGIA DE DESENVOLVIMENTO DE SISTEMAS (Sei nº 0983589).

8.2.3. A equipe técnica deverá seguir a MDS e a Arquitetura de Referência vigentes na CAPES.

8.2.4. A equipe técnica da empresa CONTRATADA poderá ser composta por profissionais com perfil Analista Programador Sênior, Pleno e Júnior, estando a qualificação mínima para estes perfis especificada no ENCARTE B - PERFIS E QUALIFICAÇÃO PROFISSIONAL (Sei nº 0920366).

8.2.5. A quantidade de postos de trabalho demandados à empresa deverá constar na abertura da OS.

8.2.6. O somatório de postos de trabalho previstos em todas as Ordens de Serviço não poderá exceder o quantitativo máximo de postos de trabalho previstos no edital.

8.2.7. O pagamento do serviço estará vinculado ao atendimento dos níveis mínimos de serviço exigidos, conforme regras constantes no ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364)

8.2.8. O serviço deverá ser prestado, em regra, nos dias úteis e dentro do horário normal de funcionamento da CAPES, que é de 8h as 20h.

8.2.9. Excepcionalmente, sem custo adicional, a CAPES poderá autorizar a execução do serviço em dia não útil ou fora do horário normal de funcionamento da CAPES.

8.2.10. A seu critério, a CAPES poderá especificar intervalo (“janela”) de horário para prestação do serviço mais restrito que o horário normal de funcionamento da fundação para determinados projetos ou sistemas, desde que mantenha, ao menos, 8h30min (oito horas e trinta minutos) contínuos no novo horário.

8.2.11. Cada posto de trabalho previsto nesta contratação deverá cumprir 8 horas de expediente diário dentro do horário de funcionamento normal da CAPES, observada a legislação trabalhista vigente.

8.2.12. Não haverá remuneração adicional da CAPES à CONTRATADA no caso de execução de jornada de trabalho diária superior a 8 (oito) horas.

8.2.13. A CAPES definirá, por Ordem de Serviço, os dias e horários em que ocorrerão as reuniões e demais atividades previstas no processo de desenvolvimento de software que requerem a presença de todos os profissionais da CONTRATADA. Estas definições poderão ser alteradas durante a execução da ordem de serviço, mediante comunicação prévia à CONTRATADA.

### 8.2.14. Atividades a serem executadas pela CONTRATADA

8.2.14.1. Codificação de software;

8.2.14.2. Testes de software (ex: teste de unidade, integração, sistema/funcional, aceitação/história, carga, desempenho, vulnerabilidade, usabilidade, acessibilidade, segurança);

- 8.2.14.3. Análise e projeto de software orientado a objetos;
- 8.2.14.4. Levantamento e análise de requisitos funcionais e não-funcionais, incluindo requisitos de acessibilidade;
- 8.2.14.5. Modelagem de dados (modelo lógico e físico);
- 8.2.14.6. Controle de versões de código-fonte de software e geração de builds;
- 8.2.14.7. Instalação e configuração básica de servidor de aplicação em ambiente de não produção (ex: desenvolvimento, homologação, etc) e publicação (deploy) dos sistemas desenvolvidos nestes ambientes;
- 8.2.14.8. Apoio ao Product Owner (PO) na definição e especificação de requisitos (refinamento dos itens do backlog do produto);
- 8.2.14.9. Participação ativa nas reuniões e demais práticas inerentes ao desenvolvimento ágil;
- 8.2.14.10. Medição do software produzido conforme Roteiro de Métricas de Software do SISP, última versão, e requisitos definidos no Edital;
- 8.2.14.11. Transferência de conhecimento acerca do projeto para a equipe do CAPES;
- 8.2.14.12. Manutenção de software codificado em tecnologias Java e PHP consideradas legadas.

#### 8.2.15. Acompanhamento dos serviços

- 8.2.15.1. A prestação do serviço será acompanhada por equipe técnica da CAPES.
- 8.2.15.2. Além da fiscalização da execução do contrato, a equipe da CAPES e de demais contratos, realizará também atividades inerentes aos demais papéis previstos no processo de desenvolvimento (ex: Scrum Master, Product Owner, etc) e ainda outras atividades que forem necessárias ao sucesso do projeto e à sustentabilidade do software produzido após o término do contrato.
- 8.2.15.3. Após a alocação de novo profissional da CONTRATADA para ocupação de posto de trabalho, caso a CAPES identifique que o profissional não atenda aos requisitos de qualificação deste Termo de Referência, a CAPES notificará a não-conformidade à CONTRATADA, que terá prazo de 10 (dez) dias úteis para adoção das providências cabíveis.
- 8.2.15.4. Feita a solicitação de aumento ou redução dos postos de trabalho da Ordem de Serviço, o prazo que a CONTRATADA terá para efetuar a alteração demandada será o mesmo previsto para ocupação inicial dos postos de trabalho quando da abertura de Ordem de Serviço.
- 8.2.15.5. A CAPES poderá, a seu critério, prorrogar este prazo, mediante solicitação da CONTRATADA devidamente justificada.
- 8.2.15.6. Caso não seja especificado prazo distinto, dependendo da situação particular, a CONTRATADA deverá resolver impropriedade identificada na execução do serviço contratado no prazo de 10 (dez) dias úteis.
- 8.2.15.7. A CAPES, a seu critério, em virtude de fatores como prazo do projeto, volume e ritmo de demandas, poderá solicitar a redução ou o acréscimo de postos de trabalho na OS, por meio de aditivo à Ordem de Serviço, respeitando os limites máximos permitidos.

#### 8.2.16. Ausências e substituições de profissional da CONTRATADA

- 8.2.16.1. Todo profissional da CONTRATADA deve possuir a qualificação mínima exigida, conforme este Termo de Referência, inclusive aqueles alocados temporariamente pela empresa para cobrir ausências eventuais de outros postos de trabalho.
- 8.2.16.2. Em regra, a CAPES não remunerará a CONTRATADA por postos de trabalho não ocupados, observadas as regras do ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364).
- 8.2.16.3. A taxa mínima exigida de ocupação dos postos de trabalho e sua relação com os níveis mínimos de serviço estão definidas no ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364)

#### 8.2.17. Término da prestação do serviço

- 8.2.17.1. No interesse da Administração, a CAPES poderá antecipar ou prorrogar o término da Ordem de Serviço, devendo comunicar a CONTRATADA com antecedência mínima de 22 (vinte e dois) dias úteis.
- 8.2.17.2. No caso de antecipação de término, durante o período entre a comunicação da CAPES e o término efetivo do Ordem de Serviço, a CONTRATADA deverá continuar prestando o serviço regularmente, atendendo as demandas restantes e efetuando a transferência de conhecimento acerca do software produzido para a equipe da CAPES.

#### 8.2.18. Garantia dos serviços

- 8.2.18.1. O prazo de garantia dos serviços entregues no âmbito deste contrato será de 120 (cento e vinte) dias, contados após o seu recebimento definitivo.

### GRUPO 2 - Mensuração de Sistemas

- 8.2.19. O regime de Fábrica de Métrica implica no uso de uma forma sistematizada de mensuração de software, apoiada em metodologia consolidada, utilizando processo controlado, repetitivo e padronizado de mensuração. Os serviços incluem atividades de identificação de funcionalidades, atribuição de grau de complexidade e realização de contagem estimada e/ou detalhada.
- 8.2.20. Qualquer contagem de Pontos de Função, quando realizada pelas CONTRATADAS (GRUPO 1 e/ou GRUPO 2), deverá ser assinada por profissional com certificação Certified Function Points Specialist – CFPS em vigor, emitida por instituição oficial.
- 8.2.21. Durante o processo de mensuração dos serviços passíveis de apuração por Pontos de Função deverá ser aplicado deflatores, conforme o Guia de Métricas do SISP, segundo o tipo de ação executada sobre uma função, diminuindo a quantidade de PF identificados para as funcionalidades alteradas e excluídas para fins de pagamento e estimativa de esforço.
- 8.2.22. Atividades a serem executadas pela CONTRATADA

  - 8.2.22.1. Identificar e analisar as funcionalidades implementadas nos sistemas, funções de dados e funções transacionais, previstas na técnica de Análise de Pontos de Função;
  - 8.2.22.2. Atribuir o grau de complexidade para cada função de dados ou função Transacional identificada;
  - 8.2.22.3. Atribuir à quantidade de Pontos de Função prevista na técnica de Análise de Pontos de Função;
  - 8.2.22.4. Identificar e considerar os itens não mensuráveis segundo o Guia de Métricas do SISP em sua versão mais atual;
  - 8.2.22.5. Realizar a contagem de Pontos de Função por decorrência de alterações de escopo, quando for o caso;
  - 8.2.22.6. Concluir a contagem de Pontos de Função do serviço, identificando o tamanho funcional do serviço e a quantidade de Pontos de Função para base de cálculo da remuneração;
  - 8.2.22.7. Realizar e Manter atualizada a contagem de Pontos de Função das aplicações (Base Instalada) do CONTRATANTE, conforme regras a serem definidas pela CONTRATANTE;
  - 8.2.22.8. Validar e/ou auditar contagens de Pontos de Função realizadas pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas);
  - 8.2.22.9. Validar e/ou auditar contagens de Pontos de Função pré-existentes;

8.2.22.10. Validar e/ou auditar o método de contagem de Pontos de Função utilizado pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas);

8.2.22.11. Validar e/ou auditar os métodos de contagens de Pontos de Função pré-existentes;

8.2.22.12. Elaborar relatórios das contagens de Pontos de Função realizadas que auxiliem a CONTRATANTE no processo de Gestão dos serviços e projetos de Desenvolvimento e Manutenção de Sistemas.

#### 8.2.23. Tipos de Serviços

8.2.23.1. As contagens em Pontos de Função devem ser do tipo: Indicativa e/ou Estimada e/ou Detalhada dependendo da necessidade e da solicitação da CONTRATANTE e compreendem os serviços de Desenvolvimento e Manutenção de Sistemas executados pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas).

8.2.23.2. Para demandas de Desenvolvimento de novos Sistemas ou Manutenções Evolutivas, o processo de medição do CONTRATANTE foi estruturado da seguinte forma:

8.2.23.3. Deve ser realizada uma contagem estimada de Pontos de Função. Essa contagem deve ser feita pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas) baseando-se no levantamento de requisitos ou outros insumos apontados;

8.2.23.4. Validar e/ou auditar os métodos de contagens de Pontos de Função pré-existentes;

8.2.23.5. Elaborar relatórios das contagens de Pontos de Função realizadas que auxiliem a CONTRATANTE no processo de Gestão dos serviços e projetos de Desenvolvimento e Manutenção de Sistemas.

8.2.23.6. Para demandas de Desenvolvimento de novos Sistemas ou Manutenções Evolutivas, o processo de medição do CONTRATANTE foi estruturado da seguinte forma:

8.2.23.7. Deve ser realizada uma contagem estimada de Pontos de Função. Essa contagem deve ser feita pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas) baseando-se no levantamento de requisitos ou outros insumos apontados pela CONTRATANTE, a fim de estimar o prazo e o custo necessários para o desenvolvimento da demanda. Esta estimativa será utilizada como insumo para emissão da Ordem de Serviço à CONTRATADA (GRUPO 1 - Desenvolvimento de Software);

8.2.23.8. Após a execução dos serviços pela CONTRATADA (GRUPO 1 - Fábrica de Software) e homologação por parte da CONTRATANTE será realizada uma contagem detalhada em Pontos de Função, a fim de realizar o pagamento dos serviços executados pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas);

8.2.23.9. Durante a execução da demanda, dependendo do seu tamanho funcional e dos riscos para o negócio, pode ser necessário realizar contagens intermediárias, a fim de acompanhar a evolução de escopo da demanda.

8.2.23.10. Para demandas de Manutenção Corretiva, Perfectiva, Adaptativa e/ou demandas de caráter Emergencial o processo de medição do CONTRATANTE foi estruturado da seguinte forma:

8.2.23.11. Após a execução dos serviços pela CONTRATADA (GRUPO 1 - Desenvolvimento de Software) e homologação por parte da CONTRATANTE será realizada uma contagem detalhada em Pontos de Função do serviço executado, a fim de realizar o pagamento dos serviços executados pela CONTRATADA (GRUPO 1 - Desenvolvimento de Sistemas);

8.2.23.12. As demandas de contagem de Pontos de Função das aplicações (Base Instalada) do CONTRATANTE serão realizadas com base em documentos elaborados pelo CONTRATANTE e/ou outras fontes de informações apontadas por ele. Para fins de pagamento será necessária a homologação dos serviços executados pela CONTRATADA (GRUPO 2 - Mensuração de Sistemas) por parte da CONTRATANTE.

8.2.23.13. Caso necessário, a pedido da CONTRATANTE a CONTRATADA (GRUPO 2 - Mensuração de Sistemas) deverá realizar a contagem estimada dos projetos de desenvolvimento ou apontar e/ou identificar o tipo de contagem mais apropriada a ser realizada.

8.2.23.14. Caso seja identificado algum item não mensurável não contemplado no Guia de Métricas do SISP, a(s) CONTRATADA(S) deverá encaminhar solicitação fundamentada à CONTRATANTE, que analisará a viabilidade da inclusão de um novo.

8.2.23.15. A CONTRATANTE também poderá identificar itens não mensuráveis, caso em que acordará a inclusão junto à CONTRATADA.

8.2.23.16. A CONTRATADA (GRUPO 2 - Mensuração de Sistemas) deve fornecer suporte à CONTRATANTE sobre as técnicas de mensuração de Sistemas em Pontos de Função, da seguinte forma:

8.2.23.17. Esclarecendo dúvidas e orientação na aplicação das técnicas de Análise de Pontos de Função;

8.2.23.18. Refinando o processo de métricas adotado pela CONTRATANTE;

8.2.23.19. Esclarecendo dúvidas e orientação do processo de geração e utilização de indicadores, inclusive quanto à criação de base de dados para esta finalidade;

8.2.23.20. Propondo melhorias no processo de avaliação/revisão das estimativas de esforço para projetos e serviços da CONTRATANTE, sempre utilizando melhores práticas de mercado.

8.2.23.21. Na apuração do tamanho em Pontos de Função para fins de remuneração dos serviços contratados, serão considerados os seguintes percentuais:

Demandas de Contagem	Percentual do valor do Ponto de Função cobrado por tipo de contagem
Validação e/ou Auditagem	50%
Estimada	70%
Detalhada	100%

Tabela 30 - Percentuais para remuneração da Mensuração de Sistemas

8.2.23.22. A conversão de dados dos sistemas legados deverá estar prevista nas contagens de pontos de função realizada pela CONTRATADA para os casos de desenvolvimento de novos sistemas, redesenvolvimento e projetos de melhoria, quando for o caso;

8.2.23.23. Considera-se como parte integrante da execução dos serviços a entrega de artefatos que contenham as seguintes informações:

Registro dos pontos de função contados pelos especialistas;

Quantidade/Porcentagem de chamados por sistema;

Quantidade/Porcentagem de PF por sistema;

Quantidade/Porcentagem de PF por tipo de ação (corretiva, evolutiva, verificação de erro, carga de dados, etc.);

Quantidade/Porcentagem de chamados por tipo de ação/sistema;

Histórico de Demandas por Ordens de Serviço indicando a quantidade de demandas encaminhadas na Ordem de serviço, quantidade efetivamente contada e a quantidade com parecer.

Classificação da complexidade dos elementos identificados na contagem;

Totalização dos pontos de função, com todas as informações possíveis dos itens:

- Arquivo Lógico Interno – ALI;
  - Arquivo de Interface Externa – AIE;
  - Consulta Externa – CE;
  - Tipo de Dados – TD;
  - Tipo de Registro – TR;
  - Arquivo Referenciado – AR;
  - Armazenamento do histórico dos pontos de função contados em banco de dados nativo;
  - Emissão de relatórios das operações efetuadas;
  - Referências cruzadas de arquivos x transações;
  - Referências cruzadas de funções x documentações;
  - Referências às informações e documentações utilizadas nas contagens;
  - Registro de glossário.
- 8.2.23.24. Os serviços de Suporte à Aplicação de Técnicas de Medição, descritos, serão prestados após a entrega do serviço de cada OS.
- 8.2.23.25. Para efeito deste contrato tal suporte faz parte do serviço de contagem de Ponto de Função não cabendo remuneração específica, devendo ser considerado no valor do ponto de função ofertado;
- 8.2.23.26. Os serviços relativos à coleta de dados, geração e análise de indicadores deste anexo, serão realizados pela CONTRATADA e considerados como inerentes ao processo de contagem e implicitamente inclusos na medição de Pontos de Função e não serão remunerados separadamente;
- 8.2.23.27. A CONTRATADA obriga-se a criar e manter Base de Dados Históricos, que possibilite a geração de indicadores solicitados pela CAPES, e armazenar todos os resultados dos serviços prestados, com versionamento de contagens realizadas;
- 8.2.23.28. A elaboração de documentação referente aos serviços realizados pela CONTRATADA é obrigatória e sem custo adicional;
- 8.2.23.29. No caso de “Multiple Media”, uma consulta (Consulta Externa ou Saída Externa) que for disponibilizada em diversos formatos (doc, .txt, pdf, xls, xml, tela ou papel) será contada apenas uma vez;

#### 8.2.24. Garantia e Suporte

- 8.2.24.1. A CONTRATADA deve garantir o serviço prestado pelo período de 06 (seis) meses, contados da data do aceite definitivo do Serviço, cabendo-lhe toda a recontagem, caso ocorra problemas na contagem do projeto, atualização das bases históricas, emissão de novos documentos, bem como os demais serviços, sem qualquer ônus adicional para a CONTRATANTE.
- 8.2.24.2. O término do CONTRATO não cessará a garantia do produto.
- 8.2.24.3. Durante o prazo estabelecido a CONTRATADA deverá manter canal de comunicação por telefone, e-mail ou sistema disponibilizado e deverá atender os seguintes critérios:
- 8.2.24.4. As correções deverão ser documentadas e encaminhadas ao CONTRATANTE.
- 8.2.24.5. O prazo máximo para correção de defeitos, contado a partir do acionamento da garantia, é de 5 (cinco) dias úteis durante a vigência do CONTRATO.
- 8.2.24.6. Extinta a vigência do CONTRATO, a CONTRATADA terá 10 (dez) dias úteis para atendimento.
- 8.2.24.7. Caso a CONTRATADA entenda necessária, em um serviço específico, a dilatação dos prazos definidos, deverá justificar-se tecnicamente por meio de relatório formal.
- 8.2.24.8. Caso o CONTRATANTE não aceite as argumentações, não haverá interrupção na contagem do prazo definido previamente.
- 8.2.24.9. Durante o prazo de garantia, toda a recontagem, atualização das bases históricas, emissão de novos documentos e os demais serviços em que tenham sido identificados erros ou falhas, deverão ser corrigidos pela CONTRATADA, sem ônus para o CONTRATANTE, mesmo em se tratando de contagens e outros artefatos já considerados aceitos pela CONTRATANTE.
- 8.2.24.10. A não observância ao prazo para correção de defeito implica na aplicação das penalidades cabíveis previstas no ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364).
- 8.2.24.11. Entregas parciais entram em garantia após o Termo de Aceite Definitivo de tais entregas emitido pelo CONTRATANTE.
- 8.2.24.12. O produto não perderá a garantia se o CONTRATANTE, ou FORNECEDOR por ele designado, criar uma nova versão a partir da versão entregue.
- 8.2.24.13. A garantia cessará apenas se a alteração for realizada na versão entregue.
- 8.2.24.14. Durante todo o período de execução dos serviços, a CONTRATADA é obrigada a manter, em base histórica, os dados sobre a execução de serviços em garantia.

#### 8.2.25. Prazos para Execução dos Serviços

- 8.2.25.1. A tabela a seguir será utilizada pelo CONTRATANTE como referência para o estabelecimento dos prazos para atendimento das Ordens de Serviços (OS) de mensuração dos projetos, conforme o tipo de demanda contadas a partir de sua emissão.
- 8.2.25.2. Caso uma Ordem de Serviço (OS) seja classificada como Emergencial a CONTRATANTE poderá priorizar sua execução e estabelecer prazos de acordo com sua necessidade sem a referência dos prazos da tabela a seguir.
- 8.2.25.3. O prazo para a execução da Ordem de Serviço (OS) se iniciará imediatamente após o recebimento da OS por parte da CONTRATADA.
- 8.2.25.4. Para demandas inferiores a 100 Pontos de Função:

Prazo	Descrição	Contagem Estimada	Contagem Detalhada
Prazo execução do Serviço	Prazo que a CONTRATADA dispõe para análise da demanda e sua execução.	3 dias úteis	4 dias úteis

Tabela 31 - Prazos para Execução Inferiores a 100 PF

- 8.2.25.5. Para demandas superiores a 100 Pontos de Função:

Prazo	Descrição	Contagem Estimada	Contagem Detalhada
Prazo execução do Serviço	Prazo que a CONTRATADA dispõe para análise da demanda e sua execução.	6 dias úteis	9 dias úteis

Tabela 32 - Prazos para Execução Superiores a 100 PF

- 8.2.25.6. Durante o prazo de execução, a CONTRATADA poderá solicitar um prazo adicional para execução do serviço, quando justificada e comprovada à necessidade, em função de complexidade do serviço a ser executado, ficando a critério do CONTRATANTE, aceitar ou não as

justificativas e o novo prazo apresentado pela CONTRATADA.

8.2.25.7. O prazo adicional deverá ser solicitado em até 1 (um) dia útil após o recebimento da OS, e no caso de aceito pela CONTRATANTE, será adicionado ao prazo total do serviço ou projeto contratado.

8.2.25.8. A solicitação de prazo adicional para atendimento não justifica a suspensão do atendimento pela CONTRATADA e, durante o julgamento da solicitação pela CONTRATANTE, ficam mantidas as condições estipuladas para o serviço;

8.2.25.9. Caso a justificativa não atenda ao CONTRATANTE prevalecerá o prazo inicialmente estipulado na OS.

8.2.25.10. No caso de acatadas pelo CONTRATANTE as revisões solicitadas, a qual dar-se-á no prazo máximo de 2 (dois) dias úteis, reinicia-se a contagem a partir da data em que o CONTRATANTE represtar a OS.

8.2.25.11. Caso a solicitação de revisão seja rejeitada pelo CONTRATANTE, a qual dar-se-á no prazo máximo de 2 (dois) dias úteis, o prazo de atendimento estabelecido inicialmente não será alterado.

#### 8.2.26. Formas de Execução

8.2.26.1. O fluxo de execução da Ordem de Serviço está definido conforme tabela abaixo:

Fluxo da Execução de Serviço de Mensuração de PF	
Responsável	Ação
Contratante	1. Abertura de <b>Ordem de Serviço (OS)</b> . 2. Envia a OS para a contratada
Contratada	1. Recebe a <b>Ordem de Serviço (OS)</b> . 2. Executa a O.S. de acordo com as especificações. 3. Entrega os produtos da OS.
Contratante	1. Emite Termo de Recebimento Provisório. 2. Valida os produtos entregues. Caso os produtos não sejam validados, retorna ao passo 4. 3. Emite Termo de Recebimento Definitivo.

Tabela 33 - Fluxo da execução de serviços de mensuração de PF

8.2.26.2. Sempre que necessário poderão ser realizadas reuniões entre o CONTRATANTE e empresa para esclarecimentos adicionais acerca do serviço a ser executado ou dos artefatos a serem gerados ou mantidos.

8.2.26.3. O resultado do serviço de contagem deverá ser registrado na Planilha de Contagem de Pontos de Função, de acordo os padrões definidos pelo CONTRATANTE.

8.2.26.4. Todas as funções (de dados e de transação) identificadas na planilha devem estar em conformidade com o Manual de Práticas de Contagem do IFPUG e com o Guia de Métricas do SISP.

8.2.26.5. Os itens não mensuráveis deverão ser identificados de acordo com as regras definidas pelo CONTRATANTE.

8.2.26.6. Em projetos de melhoria, poderá ser solicitada à CONTRATADA a atualização da base instalada da aplicação a cada serviço de mensuração, sem ônus para o CONTRATANTE, ficando a critério do CONTRATANTE a solicitação deste serviço. Nesse caso, a CONTRATADA deverá fornecer duas planilhas de contagem de PF: uma para o projeto de melhoria e outra planilha com a contagem atualizada das funções da aplicação.

8.2.26.7. Todas as informações de contagem devem ser fornecidas ao CONTRATANTE após a execução de cada serviço, ao final do contrato e sempre que solicitadas.

8.2.26.8. Todos os documentos e planilhas necessários para registro e coleta de métricas elaborados pela CONTRATADA deverão ser arquivados no Subversion do ambiente do CONTRATANTE conforme os padrões estabelecidos.

8.2.26.9. As planilhas de contagens referentes às estimativas, validações/auditagens e medições serão armazenadas separadamente para demandas de Projetos e de Manutenção. Além disso, os artefatos serão segregados por Aplicação e por Número de demanda.

8.2.26.10. As planilhas de contagens da Base Instalada das aplicações serão armazenadas separadamente, e devem ser atualizadas sempre que for realizada uma contagem detalhada para pagamento de uma Ordem de Serviço à CONTRATADA (GRUPO 1 - Desenvolvimento de Software), visando à manutenção atualizada do inventário de funções desta aplicação. A CONTRATADA (GRUPO 2 – Mensuração de Sistemas) se compromete a atualizar a planilha de Base Instalada sempre que necessário, gerando nova versão da planilha, sem ônus adicional ao CONTRATANTE.

8.2.26.11. A CONTRATADA deve armazenar os artefatos de acordo com a estrutura de diretórios definida pelo CONTRATANTE e conforme as práticas e Políticas de Configuração do CONTRATANTE.

#### 8.3. Fases de execução do contrato

##### Da transição contratual inicial:

8.3.1. Trata-se da etapa de repasse dos serviços, referente ao processo de absorção pela CONTRATADA dos conhecimentos e competências necessárias para prover a continuidade dos serviços licitados. A execução da etapa de repasse inicial dos serviços deverá ser finalizada em no máximo 30 (trinta) dias corridos, a partir da data acordada como data de início das atividades.

8.3.2. O início do contrato deverá seguir o modelo especificado no art. 31 da Instrução Normativa nº 01/2019.

"Art. 31. As atividades de início do contrato compreendem:

I - a realização de reunião inicial, a ser registrada em ata, convocada pelo Gestor do Contrato com a participação dos Fiscais Técnico, Requisitante e Administrativo do Contrato, da contratada e dos demais interessados por ele identificados, cuja pauta observará, pelo menos:

- a) presença do representante legal da contratada, que apresentará o preposto da mesma;
- b) entrega, por parte da contratada, do Termo de Compromisso e dos Termos de Ciência, conforme art. 18, inciso V; e
- c) esclarecimentos relativos a questões operacionais, administrativas e de gestão do contrato;

II - o repasse à contratada de conhecimentos necessários à execução dos serviços ou ao fornecimento de bens;

III - a disponibilização de infraestrutura à contratada, quando couber.

Parágrafo único. O disposto neste artigo é dispensável para soluções compostas exclusivamente por fornecimento de bens de TIC."

8.3.3. No prazo de 10 (dez) dias úteis após a assinatura do contrato deverá ser realizada reunião de alinhamento de expectativas entre representantes da CONTRATADA e da CONTRATANTE.

8.3.4. A CONTRATANTE poderá, a seu critério, prorrogar este prazo

8.3.5. Serão tratados os seguintes assuntos:

- a) Assinatura do Termo de Confidencialidade e Sigilo (ENCARTE P - MODELO DO TERMO DE CONFIDENCIALIDADE E SIGILO (Sei nº 0996366) ;
- b) Esclarecimentos sobre a forma de comunicação a ser adotada entre o Órgão e a CONTRATADA;
- c) Entrega dos documentos que compõem as metodologias e os padrões em uso na CONTRATANTE;
- d) Apresentação dos softwares utilizados pela CONTRATANTE para gestão de demandas e para o armazenamento e o controle de versões de produtos de software (documentos e fontes);
- e) Esclarecimentos acerca da forma de validação e aceite dos serviços realizados pela CONTRATADA;
- f) Esclarecimentos acerca dos níveis de serviço previstos no contrato, bem como sobre o período de adaptação e ajustes da CONTRATADA ao contrato;
- g) Esclarecimentos relacionados ao funcionamento do Órgão, tais como: horário de trabalho, local disponível para a equipe da CONTRATADA, regimento interno do Órgão, forma de acesso dos colaboradores da CONTRATADA às dependências da CONTRATANTE e demais informações pertinentes;
- h) Data de início das atividades de repasse dos serviços, sendo no máximo até 10 dias úteis após a assinatura do contrato;
- i) Demais assuntos relevantes para o início do contrato pela CONTRATADA

8.3.6. A CONTRATADA deverá apresentar, juntamente com o início do repasse dos serviços para aprovação pela CONTRATANTE, o Plano de Absorção do Repasse, contendo o detalhamento das atividades a serem executadas por ela e pela CONTRATANTE.

8.3.7. O Plano de Absorção do Repasse dos serviços deverá registrar e detalhar o método de trabalho e a estratégia da CONTRATADA para o processo de absorção e execução dos serviços.

8.3.8. O Plano de Absorção do Repasse deverá tratar, no mínimo, dos seguintes tópicos:

- 8.3.8.1. Cronograma geral do repasse, identificando para cada serviço ou grupo de serviços: as etapas e as atividades com suas respectivas datas de início e término, os produtos gerados, os recursos envolvidos, os marcos intermediários, quando aplicável. O cronograma será revisado e poderá ser complementado nas primeiras reuniões para aprovação do cronograma definitivo pela CONTRATANTE;
- 8.3.8.2. Identificação dos profissionais da CONTRATADA que irão compor a equipe de recepção, bem como os seus papéis e as suas responsabilidades;
- 8.3.8.3. Regras de relacionamento propostas pela CONTRATADA à CONTRATANTE;
- 8.3.8.4. O plano de gerenciamento de riscos e o plano de contingência;
- 8.3.8.5. Estratégia de acompanhamento do repasse.

8.3.9. A CONTRATANTE analisará o Plano de Absorção do Repasse apresentado, aprovando-o ou recusando-o de forma fundamentada.

8.3.10. Ao final do repasse de cada serviço ou tipo de serviço, a CONTRATADA deverá comunicar à CONTRATANTE acerca da conclusão do referido repasse. Dessa forma, quando da comunicação da conclusão do último repasse, a CONTRATADA estará plenamente apta para executar os serviços contratados, a partir do primeiro dia útil subsequente ao da comunicação, sendo que deve ser observado o prazo máximo de 30 (trinta) dias corridos para o repasse.

8.3.11. O repasse dos serviços estará finalizado quando todos os serviços tiverem o repasse concluído ou ter decorrido todo o prazo definido para essa etapa. A CONTRATADA não poderá alegar falta de condições para a execução dos serviços contratados, caso o prazo para o repasse tenha se esgotado sem que tenha havido o repasse total dos serviços.

8.3.12. **GRUPO 1 - Desenvolvimento de Sistemas:**

- 8.3.12.1. Antes do início da efetiva prestação dos serviços, a CONTRATADA deverá apresentar relação dos seus profissionais, contendo nome completo, cargo ou função, valor do salário, horário do posto de trabalho, número do registro geral (RG) e do Cadastro de Pessoas Físicas (CPF) e a Carteira de Trabalho e Previdência Social (CTPS).
- 8.3.12.2. Esta relação de profissionais deverá ser mantida atualizada pela CONTRATADA no caso de entrada e saída de profissional no contrato.
- 8.3.12.3. A CONTRATANTE realizará reunião com os profissionais terceirizados para informá-los de seus direitos previstos em contrato, esclarecendo que estão autorizados a noticiar à Administração da CAPES o descumprimento de quaisquer desses direitos.
- 8.3.12.4. Para cada profissional apresentado pela CONTRATADA, a CONTRATANTE conferirá o atendimento das certificações profissionais exigidas e analisará a adequação do currículo apresentado em relação aos requisitos de qualificação técnica exigidos, sem prejuízo da avaliação destes requisitos durante a efetiva prestação do serviço.

**Operação continuada, iniciada ao término da transição contratual inicial:**

8.3.13. Período em que a CONTRATADA executará, integralmente, todos os itens de serviço previstos.

8.3.14. Sempre que houver necessidade, a CONTRATANTE abrirá Ordem de Serviço (OS) para a CONTRATADA para prestação do serviço contratado.

8.3.15. **GRUPO 1 - Desenvolvimento de Sistemas:**

- 8.3.15.1. Nesta Ordem de Serviço, a CONTRATANTE informará o tipo de OS (PROJETO ou MANUTENÇÃO), a quantidade e o perfil dos postos de trabalho e as datas de início e término previstas para a OS.
- 8.3.15.2. No caso de OS de PROJETO, será informado ainda o nome do projeto. No caso de OS de MANUTENÇÃO, serão informados também os sistemas a serem manutenidos.
- 8.3.15.3. Após a abertura da Ordem de Serviço, a CONTRATADA terá o prazo de 22 (vinte e dois) dias úteis para disponibilizar os profissionais para ocupar os postos de trabalho para prestação do serviço.
- 8.3.15.4. A CONTRATANTE poderá, a seu critério, prorrogar este prazo, mediante solicitação da CONTRATADA devidamente justificada. A ocupação dos postos de trabalho será feita sob demanda da CONTRATANTE e formalizada por meio de Ordem de Serviço (OS). Ou seja, enquanto não for aberta OS pela CONTRATANTE, não deverá haver disponibilização de profissional da CONTRATADA e, consequentemente, não haverá prestação de serviço a ser remunerado.
- 8.3.15.5. A CONTRATANTE abrirá Ordem de Serviço (OS) quando houver projeto de desenvolvimento de software ou demanda por manutenção de sistema a ser executada pela CONTRATADA. Observada sempre a vigência do contrato, a OS permanecerá aberta até o término do Projeto, no caso de OS do tipo PROJETO, ou até que a CAPES formalize o encerramento da demanda por manutenção, no caso de OS do tipo MANUTENÇÃO.
- 8.3.15.6. A CONTRATANTE deverá comunicar com antecedência mínima de 22 (vinte e dois) dias úteis o encerramento de OS do tipo MANUTENÇÃO.
- 8.3.15.7. O serviço será prestado por meio dos postos de trabalho previstos na Ordem de Serviço até que a OS seja finalizada. A CONTRATADA será remunerada pelo serviço prestado no âmbito desta Ordem de Serviço de acordo com os postos de trabalho efetivamente ocupados no período, observando os níveis mínimos de serviço definidos (Anexo VI).

8.3.15.8. Eventual não-ocupação de posto de trabalho durante a prestação do serviço ensejará desconto conforme Cláusula Décima Sexta deste Contrato, sem prejuízo do disposto no Anexo VI – Níveis Mínimos de Serviço e Cálculo de Pagamento do Edital do Pregão Eletrônico n.º 046/2018.

8.3.15.9. O mesmo profissional não poderá ocupar posto de trabalho de Ordens de Serviço distintas simultaneamente.

8.3.15.10. Tendo cumprido todos os requisitos contratuais, será devido o pagamento mensal para a CONTRATADA conforme a quantidade de postos de trabalho efetivamente ocupados e os níveis de serviço alcançados.

#### Transição ao final do contrato:

8.3.16. Transferência de conhecimento e tecnologia: consiste no fornecimento de subsídios para que as equipes técnicas da CONTRATANTE ou empresa por ela designada obtenham todos os conhecimentos necessários para dar continuidade ao atendimento das solicitações de serviços da CONTRATANTE, quando da rescisão do contrato firmado com a CONTRATADA;

8.3.17. A CONTRATADA deverá promover o repasse de todo o conhecimento adquirido ou produzido na execução dos serviços para os técnicos da CONTRATANTE;

8.3.18. Toda e qualquer informação produzida no âmbito da execução do objeto do contrato pelas empresas prestadoras dos serviços será de propriedade da CONTRATANTE e fica a CONTRATADA obrigada a documentar e registrar os produtos, serviços e eventos observando as metodologias e ferramentas utilizadas na CONTRATANTE;

8.3.19. A CONTRATADA deverá entregar à CONTRATANTE toda e qualquer documentação gerada em meio magnético e/ou físico em função da prestação de serviços;

8.3.20. Será de inteira responsabilidade da CONTRATADA garantir o repasse bem-sucedido de todas as informações necessárias para a continuidade dos serviços pelo CONTRATANTE ou empresa por ele designada;

8.3.21. A CONTRATADA deverá descrever a metodologia que será utilizada para transferir conhecimento aos técnicos da CONTRATANTE, os quais poderão ser multiplicadores do conhecimento transferido a outros técnicos ou a usuários finais;

8.3.22. A transferência de conhecimento, direcionada para os técnicos indicados pela CONTRATANTE, deverá ser focada na solução adotada, de forma que haja transferência do conhecimento utilizada em todo o processo. Ao final da transferência, técnicos da CONTRATANTE deverão estar capacitados para realizarem os serviços;

8.3.23. Caberá à CONTRATADA promover o repasse de conhecimento aos seus novos profissionais em caso de substituição dos responsáveis pela execução de serviços em andamento, minimizando problemas relacionados à continuidade e qualidade dos serviços prestados;

8.3.24. Ocorrendo nova licitação, com mudança de fornecedor dos serviços, a CONTRATADA signatária do contrato em fase de expiração, assim considerado o período dos últimos três meses de vigência, deverá repassar para a vencedora do novo certame, por intermédio de evento formal, os documentos necessários a continuidade da prestação dos serviços, bem como esclarecer dúvidas a respeito de procedimentos no relacionamento entre a CONTRATANTE e a CONTRATADA;

8.3.25. A transição ao final do contrato não é atividade remunerada pelo CONTRATANTE.

#### Local de execução

##### 8.3.26. GRUPO 1 - Desenvolvimento de Sistemas

8.3.26.1. O serviço deverá ser prestado presencialmente na Sede da CAPES em Brasília/DF.

8.3.26.2. Em casos excepcionais, a interesse da CONTRATANTE, os profissionais deverão ser alocados fora da CAPES, porém a alocação deverá ser em Brasília e eles deverão estar disponíveis na CAPES sempre que requisitados. Com a adoção de metodologia ágil e a prática de reuniões diárias, estes profissionais deverão utilizar videoconferência para participação das reuniões.

##### 8.3.27. GRUPO 2 – Mensuração de Sistemas

8.3.27.1. O serviço deverá ser prestado em Brasília/DF e os profissionais deverão estar disponíveis na CAPES sempre que requisitados.

#### Horário de prestação do serviço

8.3.28. O horário de expediente da CAPES é das 7hrs30min às 19hrs30min.

##### 8.3.29. GRUPO 1 - Desenvolvimento de Sistemas

8.3.29.1. Devido a necessidades pontuais da CAPES de prestação de serviço fora do horário de expediente, foi realizado estudo em demais editais e esta equipe técnica de planejamento da contratação entendeu que o definido pelo TCE-PR (Sei nº 0925688) atenderia as necessidades. Portanto acrescentou os itens que seguem.

8.3.29.2. Cabe à CONTRATADA e à CAPES definir as escalas de horário das equipes para o cumprimento da jornada de trabalho, que poderá, inclusive, ser estabelecida fora do horário de expediente informado, a depender da necessidade do serviço, admitida a compensação de horas desde que atendidas as exigências legais.

8.3.29.3. A jornada de trabalho será de 8 (oito) horas diárias, admitida a compensação de horas desde que atendida a legislação vigente.

8.3.29.4. O cumprimento da jornada de trabalho deverá obedecer aos dias de expediente previstos no calendário da CAPES.

#### Perfis Técnicos, Profissionais e Qualificação dos colaboradores das CONTRATADAS

8.3.30. A qualificação do profissional será avaliada por meio de análise curricular e pela capacidade de execução de tarefas concretas no âmbito da prestação do serviço.

8.3.31. A alocação de profissional pela CONTRATADA que não atenda aos requisitos de qualificação técnica exigidos por 3 (três) vezes consecutivas no âmbito da mesma Ordem de Serviço ou por 6 (seis) vezes alternadas no período de 8 (oito) meses no âmbito do contrato poderá caracterizar inexecução parcial do contrato.

8.3.32. descrição dos perfis técnicos e profissionais tal qual a qualificação exigida para os profissionais que prestarão o serviço está descrita no ENCARTE B - PERFIS E QUALIFICAÇÃO PROFISSIONAL (Sei nº 0920366).

#### Medição do tamanho funcional do software desenvolvido

8.3.33. A medição do tamanho funcional do software desenvolvido para fins de aferição do cumprimento dos níveis de serviço (ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO Sei nº 0920364) será de responsabilidade da empresa CONTRATADA em desenvolvimento de software (GRUPO 1).

8.3.34. A medição deverá ser realizada por especialista da CONTRATADA para desenvolvimento de software (GRUPO 1) certificado em Ponto de Função (Certified Function Point Specialist – CPFS) pelo International Function Point Users' Group (IFPUG) e será posteriormente validada pela CAPES, com apoio do contrato de Mensuração de Sistemas (GRUPO 2). O especialista que realizará a medição não deverá ocupar um dos postos de trabalho previstos no contrato (GRUPO 1).

8.3.35. A realização desta atividade não reduzirá a produtividade esperada nem os demais níveis mínimos de serviço exigidos (ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO Sei nº 0920364), devendo o seu custo ser incluído pela CONTRATADA (GRUPO 1) para desenvolvimento de software no item IV – “Lucros e despesas indiretas” de sua proposta de preços (ENCARTE E - MODELO DE PROPOSTA DE PREÇOS Sei nº 0981395).

8.3.36. O profissional que estiver desenvolvendo o software deverá repassar as informações necessárias ao especialista que realizará a medição do seu tamanho funcional (GRUPO 1) e também à equipe da CAPES, ou da empresa CONTRATADA para Mensuração de Sistemas (GRUPO 2), que fará a sua validação.

8.3.37. A seu critério, a CAPES poderá validar e considerar para fins de aferição de níveis de serviço somente parte da medição em tamanho funcional apresentada pela empresa de desenvolvimento de software (GRUPO 1) desde que a não conferência do restante não ocasione prejuízo no pagamento.

#### Preposto

8.3.38. No prazo de 5 (cinco) dias úteis após a assinatura do contrato, a CONTRATADA deverá indicar, mediante declaração, um preposto, aceito pelo fiscal do contrato e que seja distinto dos membros das equipes técnicas que irão prestar o serviço, para representá-la administrativamente durante o período de vigência do contrato, sempre que for necessário. Na declaração deverá constar o nome completo, nº do CPF e do documento de identidade, além dos dados relacionados à sua qualificação profissional, especificada no ENCARTE B - PERFIS E QUALIFICAÇÃO PROFISSIONAL (Sei nº 0920366).

8.3.39. Qualquer alteração de preposto deverá ser comunicada imediatamente pela CONTRATADA, mediante declaração, e deverá ter a ciência do fiscal do contrato. A CONTRATANTE poderá solicitar a alteração do preposto, mediante justificativa.

8.3.40. O preposto deverá ter experiência em gestão de serviços terceirizados mediante comprovação por currículo e conhecimentos acerca de processo de desenvolvimento de software baseado em metodologia ágil.

8.3.41. O preposto deverá estar apto a esclarecer as questões relacionadas às faturas dos serviços prestados e demais assuntos relativos à execução contratual.

8.3.42. A CONTRATADA orientará o seu preposto quanto à necessidade de acatar as orientações da CAPES, inclusive quanto ao cumprimento dos normativos internos.

8.3.43. O preposto deverá participar de reuniões sempre que solicitado pela CAPES para discutir o andamento da execução contratual.

#### De Segurança

8.3.44. Os procedimentos básicos mínimos de segurança exigidos da CONTRATADA são:

8.3.45. Credenciar junto à CONTRATANTE, seus profissionais autorizados a retirar e a entregar documentos, bem como aqueles que venham a ser designados para prestar serviços nas dependências da CONTRATANTE. A Contratante deverá utilizar o modelo para credenciamento e descredenciamento de profissionais ENCARTE G - MODELO DE TERMOS DE CREDENCIAMENTO E DESCREDENCIAMENTO (Sei nº 0987886).

8.3.46. Identificar qualquer equipamento da CONTRATADA que venha a ser instalado nas dependências da CONTRATANTE, utilizando placas de controle patrimonial, selos de segurança, etc.:

8.3.47. Fazer com que seus funcionários mantenham sigilo absoluto sobre informações, dados e documentos integrantes dos serviços a serem executados, inclusive com a assinatura de Termo de Responsabilidade e manutenção de sigilo próprio, semelhante ao disponível na CONTRATANTE;

8.3.48. A empresa CONTRATADA deverá seguir todas as determinações e orientações contidas nas Diretrizes e Normas de Segurança da CONTRATANTE;

8.3.49. Adotar critérios adequados para o processo seletivo dos profissionais, com o propósito de evitar a incorporação de pessoas com características e/ou antecedentes que possam comprometer a segurança ou credibilidade da CONTRATANTE;

8.3.50. Comunicar com antecedência mínima de 3 (três) dias à CONTRATANTE qualquer ocorrência de transferência, remanejamento ou demissão, para que seja providenciada a revogação de todos os privilégios de acesso aos sistemas, informações e recursos da CONTRATANTE, porventura colocados à disposição para realização dos serviços contratados;

8.3.51. Manter sigilo sobre todos os ativos de informações e de processos da CONTRATANTE e da CONTRATADA que se refiram à CONTRATANTE;

8.3.52. A CONTRATADA deverá manter sigilo absoluto sobre quaisquer dados, informações, códigos-fonte, artefatos, contidos em quaisquer documentos e em quaisquer mídias, de que venha a ter conhecimento durante a execução dos trabalhos, não podendo, sob qualquer pretexto divulgar, reproduzir ou utilizar, sob pena de lei, independentemente da classificação de sigilo conferida pela CONTRATANTE a tais documentos;

8.3.53. A CONTRATADA não poderá divulgar quaisquer informações a que tenha acesso em virtude dos trabalhos a serem executados ou de que tenha tomado conhecimento em decorrência da execução do objeto, sem autorização, por escrito, da CONTRATANTE, sob pena de aplicação das sanções cabíveis, além do pagamento de indenização por perdas e danos;

8.3.54. Cada profissional deverá assinar Termo de Ciência (Sei nº 0981161), comprometendo-se a não divulgar nenhum assunto tratado nas dependências da CONTRATANTE ou a serviço desta, salvo se expressamente autorizado.

#### Acordos de Nível de Serviço

8.3.55. Na definição de Acordos de Nível de Serviço (ANS) são estabelecidos critérios objetivos e mensuráveis com a finalidade de aferir e avaliar o desempenho relacionado aos serviços contratados, conforme apresentado mais adiante. Nos Acordos encontram-se definidos, de forma objetiva, a maneira pela qual estes fatores serão avaliados, o nível mínimo aceitável e os descontos a serem aplicados na fatura mensal, quando o serviço prestado não alcançar o nível esperado.

8.3.56. A instituição e o estabelecimento dos parâmetros de níveis de serviços encontram respaldo e assento no Tribunal de Contas da União, no documento “Levantamento acerca da Governança de Tecnologia da Informação na Administração Pública Federal”, encontrado em <http://portal.tcu.gov.br/biblioteca-digital/levantamento-acerca-da-governanca-de-tecnologia-da-informacao-na-administracao-publica-federal.htm>, como transrito a seguir:

8.3.57. A prestação de um bom serviço para os cidadãos é, em última instância, o negócio de toda Instituição pública. A definição do que é um “bom serviço”, sintonizando as expectativas dos clientes com a oferta, é exatamente o que constitui um acordo de nível de desempenho (SLA, sigla do inglês Service Level Agreement).

8.3.58. No caso de um ANS é definida a qualidade dos serviços de TI em função das necessidades da organização, quantificadas e especificadas para cada serviço. Assim, a disponibilidade da infraestrutura de rede, o desempenho dos sistemas, o tempo de solução de problemas e outros dados semelhantes costumam constituir indicadores dos documentos de acordos de nível de serviço, e devem ser adequadamente verificados e tratados quando detectadas falhas, de modo a atender às necessidades do negócio. Sem a definição de tais indicadores, fica difícil responder à questão: “os serviços de TI da minha organização estão adequados às necessidades do negócio?”. Igualmente, fica difícil priorizar investimentos e ações na área de TI sem saber onde o desempenho está mais próximo ao limite do esperado ou é mais crítico para o negócio.

8.3.59. Um aspecto particularmente importante é a gestão de níveis de serviço também para serviços contratados. A especificação formal de tais indicadores pode ser o principal Instrumento dos gestores para garantir o cumprimento dos contratos de TI e possibilitar a aplicação de penalidades em casos de não-atendimento. A necessidade de acordo prévio e de mensuração da qualidade de serviços de TI é citada, inclusive, em trechos de Acórdãos do TCU, como o Acórdão nº 2.172/2005-TCU-Plenário e o Acórdão nº 786/2006-TCU-Plenário.

8.3.60. Corrobora com tal posicionamento da Corte de Contas o julgado proferido para o Ministério da Educação, nos termos do Acórdão nº. 669/2008-Plenário, reafirmado pelo Acórdão nº. 1724/2010-Plenário, como transcrito:

ACORDAM os Ministros do Tribunal de Contas da União, reunidos em Sessão do Plenário, ante as razões expostas pelo Relator em:

9.1. recomendar ao Ministério da Educação, com fulcro nos arts. 43, I, da Lei nº 8.443/1992 e 250, III, do Regimento Interno do TCU, que: [...] 9.1.15. Estabeleça e utilize um processo de gestão de contratos de bens e serviços de TI que aborde aspectos legais, financeiros, organizacionais, documentais, de níveis de serviço, de segurança, de propriedade intelectual, de penalidades e sanções e relativos às responsabilidades das partes envolvidas, com o intuito de promover a conformidade, celeridade, economicidade, eficiência e qualidade na elaboração do produto contratado. Referido processo deverá conter procedimentos específicos para mudanças e encerramento do contrato, à semelhança das orientações contidas nos itens A15.1 e A15.2 do Cobit 4.1 (Controle das aquisições e Gerência de contratos com fornecedores); 9.1.16. Estabeleça controles internos para a área de TI referentes à proteção, integridade e disponibilidade dos recursos e defina as responsabilidades de forma a garantir a efetividade e eficiência dos processos de TI e a aderência aos normativos relacionados. Deverão ser elaborados procedimentos de monitoração e avaliação dos controles internos estabelecidos, à semelhança das orientações contidas no item ME2.1 do Cobit 4.1 (Monitoração dos Controles Internos); [...] 9.4. Determinar ao Ministério da Educação, com fulcro nos Arts. 43, I, da Lei nº 8.443/1992 e 250, II, do Regimento Interno do TCU, que: [...] 9.4.9. Exija, nas próximas contratações de TI, níveis de serviço a serem cumpridos pelas empresas, especialmente para atividades como suporte de infraestrutura de TI e atendimento e suporte aos usuários, em atenção ao § 1º do art. 3º do Decreto nº 2.271/1997; (destaque) [...] 9.4.14. Em atenção aos princípios da economicidade e da eficiência, nas contratações de serviços relativos à tecnologia da informação, estabeleça remuneração vinculada a resultados ou ao atendimento de níveis de serviço, de acordo com a natureza dos serviços.

8.3.61. Assim sendo, com o respaldo jurisprudencial anterior, fica definido que os primeiros 30 (trinta) dias após o início da execução dos serviços serão considerados como período de estabilização e de ajustes específicos, durante o qual as metas definidas serão validadas.

8.3.62. Os ANS's devem ser considerados e entendidos pela CONTRATADA como um compromisso e comprometimento de qualidade que está assumindo para a prestação dos serviços.

8.3.63. Eventualmente poderão existir inconsistências técnicas (divergência entre ambientes, por exemplo) que impactem na detecção de desconformidades nos produtos entregues, refletindo diretamente na apuração do indicador de nível de serviço. Nestes casos, a CONTRATADA deverá notificar formalmente a CONTRATANTE, identificando a desconformidade apurada e detalhando a inconsistência que a originou indevidamente e, caso aceita, a desconformidade não será considerada na apuração do nível de serviço.

8.3.64. As apurações dos ANS's deverão constar do Relatório de Acompanhamento, apresentado mensalmente à CONTRATANTE, onde será possível verificar a efetividade da prestação do serviço e permitir a depuração do processo;

8.3.65. O não atendimento das metas poderá ser objeto apenas de notificação nas duas primeiras ocorrências, de modo a não comprometer a continuidade da contratação.

8.3.66. A CONTRATADA será avaliada pelo critério de resultados alcançados, incluindo definições de ANS, de forma que, caso a CONTRATADA não apresente os resultados desejados, serão aplicados redutores ao pagamento, quando cabíveis. Para tal, deverão ser observadas as características descritas a seguir:

Tabela de grau de infração	
Grau	Correspondência
01	Glosa de 0,5% (meio por cento), sobre o pagamento da Ordem de Serviço
02	Glosa de 5% (cinco por cento), sobre o pagamento da Ordem de Serviço
03	Glosa de 10% (dez por cento), sobre o pagamento da Ordem de Serviço

Tabela 34 - Grau de infração

8.3.67. As penalidades a que se sujeita a CONTRATADA terão como referência as infrações relacionadas na tabela a seguir:

Infrações			
Nº	Descrição	Referência	Grau
01	Não devolver cartão de acesso às dependências da CONTRATANTE quando o profissional não mais atender o contrato na instituição	Por ocorrência	01
02	Quando o profissional não assinar o termo de responsabilidade	Por ocorrência	01
03	Suspender ou interromper, salvo motivo de força maior ou caso fortuito, os serviços solicitados.	Por ocorrência	02
04	Manter empregado sem qualificação para executar os serviços contratados.	Por dia e por profissional	02
05	Causar qualquer indisponibilidade dos serviços do CONTRATANTE por motivo de imperícia na execução das atividades contratuais.	Por ocorrência	03
06	Utilizar indevidamente os recursos de TI (acessos indevidos, utilização para fins particulares, etc.) ou utilizar equipamento particular, salvo em situação excepcional e devidamente autorizado pelo CONTRATANTE.	Por ocorrência	02
07	Não prover a substituição, no prazo de 48 horas corridas, após notificação do CONTRATANTE, de empregado que se conduza de modo inconveniente ou não corresponda aos requisitos definidos.	Por dia e por profissional	02
08	Não registrar, registrar erroneamente, fraudar, manipular ou descaracterizar indicadores e/ou metas de níveis de serviço por quaisquer subterfúgios.	Por registro alterado	03
09	Ocorrência de erro em produção em aplicação testada, desde que a versão seja a mesma testada e não haja nenhuma alteração na build/arquivos de configuração, massa de dados utilizada para testes e esteja em conformidade com os guias de testes da CONTRATANTE.	Por ocorrência	01

Para os itens a seguir, deixar de:

10	Zelar pelas instalações do CONTRATANTE.	Por ocorrência	01
11	Substituir empregado que se conduza de modo inconveniente ou não atenda às necessidades.	Por dia e por profissional	02
12	Executar serviço relacionado ao objeto do contrato, determinado pela fiscalização, por serviço.	Por ocorrência	02
13	Cumprir quaisquer obrigações estabelecidas no Edital, seus anexos e no contrato, não previstas nesta tabela, após reincidência formalmente notificada pelo CONTRATANTE.	Por ocorrência	03
14	Executar procedimentos em conformidade com a Política de Segurança.	Por ocorrência	02
15	Atingir índices ou indicadores de níveis de serviços, estabelecidos no Catálogo de Serviços.	Por chamado	01
16	Manter, nas dependências da CONTRATANTE o Preposto para gerenciamento de serviços, sem ônus para o CONTRATANTE. Na ausência ou desligamento do Preposto, a CONTRATANTE terá o prazo de 3 dias úteis para substitui-lo.	Por ocorrência	02
17	De realizar transição plena do CONTRATO, com total transferência de conhecimento	Por ocorrência	03

Tabela 35 - Infrações

8.3.68. No caso de ocorrência das infrações aqui descritas ou descumprimento de ANS por três vezes, durante um ano, sejam relativas a um mesmo item descumprido ou itens distintos, serão aplicadas as sanções administrativas, conforme previsto na Lei 8.666/93, podendo ensejar advertência, multa e,

em reincidência, rescisão de contrato e demais penalidades cabíveis, todas registradas no SICAF. As duas primeiras ocorrências serão objeto apenas de notificação.

8.3.69. Caso a CONTRATADA venha a solicitar atestado de capacidade técnica, os Acordos de Nível de Serviço apurados durante o contrato, assim como glosas por infrações serão parte integrante deste.

8.3.70. A aplicação de glosa devido a infrações, por ordem de serviço, não deverá ultrapassar 20% do valor estipulado da OS.

8.3.71. A aplicação das sanções previstas não exclui a possibilidade de responsabilidade da CONTRATADA por eventuais perdas e danos causados à Administração.

8.3.72. As multas deverão ser recolhidas no prazo máximo de dez dias úteis, a contar da data do recebimento da comunicação enviada pelo MP, quando for o caso.

8.3.73. As sanções previstas são independentes entre si, podendo ser aplicadas de forma isolada ou, no caso das multas, cumulativamente com outras, sem prejuízo de outras medidas cabíveis.

#### **Da Qualificação Técnica/ Para a fase de habilitação**

8.3.74. A empresa licitante deverá apresentar, na fase de habilitação, atestados de capacidade técnica da lavra de pessoa jurídica de direito público (órgãos da Administração Pública direta e indireta) ou por pessoa jurídica de direito privado (empresas privadas), estabelecidas no Brasil

#### **GRUPO 1 - Desenvolvimento de Sistemas**

8.3.75. Comprovação de qualificação econômico-financeira:

Balanço Patrimonial do último exercício social exigível, apresentado na forma da lei e regulamentos na data de realização deste Pregão, vedada sua substituição por balancetes ou balanços provisórios, podendo ser atualizado por índices oficiais quando encerrados há mais de 3 (três) meses da data da sessão pública de abertura deste processo licitatório;

Demonstração do Resultado do Exercício (DRE) relativa ao último exercício social exigível, apresentado na forma da lei;

Declaração de contratos firmados com a iniciativa privada e com a Administração Pública cujo objeto seja prestação de serviços mediante a disponibilização de empregado de maneira exclusiva, na forma de posto de trabalho, vigentes na data da sessão pública de abertura deste Pregão, conforme modelo constante no ENCARTE N - MODELO DE DECLARAÇÃO DE CONTRATOS FIRMADOS COM A INICIATIVA PRIVADA E COM A ADMINISTRAÇÃO PÚBLICA (Sei nº 0992017);

Prova de inexistência de débitos inadimplidos perante a Justiça do Trabalho, mediante a apresentação de certidão negativa, nos termos do Título VII-A da Consolidação das Leis do Trabalho, aprovada pelo Decreto-Lei nº 5.452, de 1º de maio de 1943, tendo em vista o disposto no art. 3º da Lei nº 12.440, de 7 de julho de 2011;

Certidão negativa de feitos sobre falência, recuperação judicial ou recuperação extrajudicial, expedida pelo distribuidor da sede da sociedade empresária licitante.

8.3.76. Os documentos exigidos para fins de qualificação econômico-financeira deverão comprovar o seguinte:

Índices de Liquidez Geral (LG), Liquidez Corrente (LC) e Solvência Geral (SG) superiores a 1;

Capital Circulante Líquido (CCL) ou Capital de Giro (Ativo Circulante – Passivo Circulante) de, no mínimo, 16,66% (dezesseis inteiros e sessenta e seis centésimos por cento) do valor estimado para a contratação;

Patrimônio Líquido (PL) igual ou superior a 10% (dez por cento) do valor estimado para a contratação; e

Patrimônio Líquido igual ou superior a 1/12 (um doze avos) do valor total dos contratos firmados com a Administração Pública e com a iniciativa privada cujo objeto seja prestação de serviços mediante a disponibilização de empregado de maneira exclusiva, na forma de posto de trabalho, vigentes na data da sessão pública de abertura deste Pregão.

8.3.77. Caso o valor total constante na declaração de que trata a Subcondição 14.12.2.1 item 'c' apresente divergência percentual superior a 10% (dez por cento), para mais ou para menos, em relação à receita bruta discriminada na Demonstração de Resultado do Exercício (DRE), a licitante deverá apresentar as devidas justificativas no ENCARTE N - MODELO DE DECLARAÇÃO DE CONTRATOS FIRMADOS COM A INICIATIVA PRIVADA E COM A ADMINISTRAÇÃO PÚBLICA (Sei nº 0992017).

Se as justificativas não forem entregues concomitantemente à documentação, o Pregoeiro fixará prazo para a sua apresentação.

8.3.78. Para fins de qualificação técnico-operacional:

8.3.79. Comprovação de, no mínimo, 3 (três) anos de experiência na execução de serviços de desenvolvimento de software, comprovados por meio de atestados ou declarações de capacidade técnica, cópias de contratos, registros em órgãos oficiais, ou outros documentos idôneos.

#### **GRUPO 2 - Mensuração de Sistemas**

8.3.80. Experiência da licitante em contagem detalhada baseada na técnica de Análise de Ponto de Função do International Function Point Users' Group (IFPUG), realizada por especialista certificado em Ponto de Função (Certified Function Point Specialist – CPFS) pelo IFPUG em projetos de desenvolvimento de software que totalizem, no mínimo, 10.000 (dez mil) Pontos de Função no período de um ano ininterrupto os quais demonstrarão a capacidade do futuro fornecedor em prestar a integralidade dos serviços, nos termos do inciso II, art. 30 da Lei nº 8.666/93.

8.3.81. Experiência da licitante em contagem baseada na técnica da NESMA (Netherlands Software Metrics Users Association) em projetos de desenvolvimento de software que totalizem, no mínimo, 10.000 (dez mil) Pontos de Função no período de um ano ininterrupto.

8.3.82. Comprovação de participação na execução de pelo menos 01 (um) projeto, no qual foram utilizadas as melhores práticas do mercado, tais como: Adoção de Indicadores de Nível de Serviço, PMBOK, ITIL v.3, CMMI, MPSBR, COBIT 4.1, ISO/IEC 27002, ISO/IEC 27001, ISO/IEC 20000, ISO/IEC 17799, ISO/IEC 15504, ISO/IEC 12207, ISO/IEC 9196 ou equivalentes.

8.3.83. O total dos pontos de função e de unidades de serviço técnico executados, para efeito de comprovação dos requisitos de habilitação, será obtido pelo somatório das experiências devidamente comprovadas via descrição de projetos, respeitado o intervalo máximo de um ano entre o início do primeiro e o término do último projeto;

8.3.84. A exigência do período de um ano ininterrupto para a comprovação da compatibilidade da quantidade visa evitar que o somatório de atestados acumulados durante um longo período de tempo atinja o quantitativo exigido sem, no entanto, comprovar a capacidade logística e operacional da licitante em executar o volume de serviço previsto. Trata-se de limitação de prazo relacionada à comprovação da capacidade de execução do objeto, aceita como legítima pelo Tribunal de Contas da União (Acórdão nº 2.048/2006 – Plenário e Acórdão nº 1.287/2008 – Plenário).

8.3.85. Para os itens do Grupo 2, somente serão aceitos atestados em Pontos de Função.

8.3.86. A CAPES se reserva ao direito de realizar diligências os atestados de capacidade técnica apresentados pelas licitantes para esclarecer eventuais dúvidas ou atestar a veracidade da documentação apresentada.

8.3.87. Os atestados deverão ser emitidos por pessoa jurídica de direito público (órgãos da Administração Pública direta e indireta) ou por pessoa jurídica de direito privado (empresas privadas), estabelecidas no Brasil, que comprove(m) a aptidão da licitante para desempenho de atividade pertinente e compatível com o objeto da licitação, mencionando explicitamente que forneceu objetos com características compatíveis e/ou similares.

- 8.3.88. Obrigatoriamente, todos os atestados deverão atender as seguintes condições:
- Que seu conteúdo deixe inequívoco o órgão/entidade/empresa emissor;
  - Conter nome, cargo/função, CPF, Identidade, telefone e endereço de correio eletrônico do representante do atestante (ou qualquer outra forma que o atestante possa se valer para firmar contratos);
  - Redação clara, sucinta e objetiva que demonstre o atendimento ao fim a que se presta;
  - Expressa concordância de que o atestante possa ser diligenciado com o objetivo de averiguar a veracidade, esclarecer ou completar as informações atestadas;
  - Assinatura (aceita também a digital) do representante legal do órgão público ou empresa emissor do atestado.

- 8.3.89. Para atestados expedidos por pessoa jurídica de direito privado, estes deverão atender, ainda, às seguintes condições complementares:

Ser anexada cópia do contrato social, no caso de sócio-proprietário;

No caso de atestados emitidos por empresa da iniciativa privada, não serão considerados aqueles emitidos por empresas pertencentes ao mesmo grupo empresarial do licitante, isto é, empresas controladas ou controladoras da empresa proponente, ou que tenha pelo menos uma mesma pessoa física ou jurídica que seja sócio da empresa emitente do atestado e da empresa proponente.

- 8.3.90. Conforme previsto na Lei nº 8.666, no art. 43, § 3º, e em consonância com as orientações e determinações do Tribunal de Contas da União, os Atestados de Capacidade Técnica apresentados, caso necessário, serão objeto de diligência para verificação de autenticidade de seu conteúdo, momento em que serão solicitados ao emitente dos atestados documentos e evidências que descrevam e comprovem a execução dos serviços ali declarados.

- 8.3.91. No processo de diligência poderão ser colhidas evidências que comprovem a capacidade técnica, tais como: relatórios, registros de reunião, impressão das telas das soluções, documentação de projetos (planejamento de projeto, planos de gestão, documentos de requisitos, diagramas, especificações técnicas, padrões, dentre outros) para a devida comprovação dos serviços atestados, afim de satisfazer plenamente o interesse público envolvido na fase de habilitação.

- 8.3.92. Encontrada divergência entre o especificado nos atestados e o apurado em eventual diligência, inclusive validação do Contrato de prestação de serviços entre o emissor do atestado e a licitante, além da desclassificação no processo licitatório, fica sujeita a licitante às penalidades legais cabíveis;

- 8.3.93. A recusa do emitente do atestado em prestar esclarecimentos, informações, em fornecer documentos comprobatórios etc., desconstituirá o atestado de capacidade técnica e poderá configurar prática de falsidade ideológica, ensejando comunicação ao Ministério Público Federal e abertura de Processo Administrativo Disciplinar, para fins de apuração de responsabilidade, em atendimento aos termos do Acórdão nº. 1724/2010-Plenário:

*Recomendar ao Ministério da Educação que preveja expressamente, em seus futuros Instrumentos convocatórios para aquisição de bens e serviços de TI, possibilidades de aplicação de sanções no que tange à apresentação de atestados de capacidade técnica incompatíveis com o objeto do certame, buscando, de antemão, inibir a participação de empresas que não satisfaçam as condições editalícias e/ou interfiram negativamente no normal andamento de qualquer ato da licitação;*

- 8.3.94. Tal prática de diligência, além de absolutamente regular e legalmente prevista, vem recebendo do TCU reiteradas recomendações no sentido de que seja aplicada, a exemplo dos julgados que transcrevemos com os nossos destaques:

ACÓRDÃO Nº 0747-10/11-P

9.1. Conhecer da presente representação, com amparo no art. 113, § 1º, da Lei nº 8.666, de 21 de junho de 1993, e art. 237, inciso VII, do Regimento Interno do TCU, para, no mérito, considerá-la improcedente;

[PROPOSTA DE DELIBERAÇÃO]

3. Por meio do expediente de fls. 1/47 (peça 1), aponta a representante, em suma, as seguintes irregularidades:

[...]

3.3. A diligência realizada pela pregoeira, no intuito de aclarar informações referentes ao atestado de capacidade técnica apresentado pela [licitante omissis], teria permitido a obtenção de dados que deveriam constar originariamente da proposta;

3.4. Os requisitos de qualificação técnica não teriam sido comprovados, mesmo após as diligências realizadas;

[...]

22. Com respeito à diligência realizada pela pregoeira (itens 3.3 e 3.4), sabe-se que, à luz do disposto no art. 43, § 3º, da Lei nº 8.666, de 21 de junho de 1993, é facultada à Comissão ou autoridade superior, em qualquer fase da licitação, a realização de diligência com vistas a esclarecer ou complementar a instrução do processo, vedada a inclusão posterior de documento ou informação que deveria constar originariamente da proposta.

23. E, no caso em apreço, vejo que a medida saneadora efetuada pelo MDA, por meio de contato telefônico, em 15/10/2010, não infringiu a lei de licitações, já que teve por objetivo o detalhamento dos pontos de função por sistema, no tocante ao atestado de capacidade técnica fornecido pela Secretaria de Educação da Bahia e apresentado pela [licitante omissis].

24. Assim, ante os esclarecimentos prestados, anuo ao encaminhamento sugerido pela unidade técnica, e pugno pela improcedência da presente representação, considerando prejudicada a cautelar pleiteada pela [empresa representante].

ACÓRDÃO Nº 4827-32/09-2

9.6. Determinar:

9.6.1. À Coordenação-Geral de Logística e Administração do MDS - CGLA que:

9.6.1.9. Atente à possibilidade de promoção de diligência pela comissão ou autoridade superior, em qualquer fase da licitação, para esclarecer ou complementar a instrução do processo licitatório, vedada a inclusão posterior de documento ou informação que deveria constar originariamente da proposta, em conformidade com o § 3º do art. 43 da Lei 8.666/1993;

ACÓRDÃO Nº. 5857-37/09-1

9.5. Determinar, com fulcro no art. 18 da Lei nº 8.443/1992, à CORE/FUNASA/MS que:

[...]

9.5.3 nas licitações que executar, promova, sempre que necessário, diligência destinada a esclarecer ou a complementar a instrução do processo, nos termos do art. 43 §3º, da lei 8.666/93, de preferência, previamente à execução dos atos de homologação e adjudicação do objeto da licitação;

## 9. MODELO DE GESTÃO DO CONTRATO E CRITÉRIOS DE MEDIÇÃO

- 9.1. Durante a vigência do contrato, existirão os seguintes papéis relacionados à sua gestão:

### 9.1.1. Gestor do Contrato

- 9.1.1.1. Servidor da CAPES com atribuições gerenciais, técnicas e operacionais, relacionadas ao processo de gestão do contrato.
- 9.1.1.2. Promove a reunião de abertura do contrato com os fiscais, preposto e representante da CONTRATADA.
- 9.1.1.3. Confere a validade contratual das glosas aplicadas e encaminhar soluções juntamente com os representantes da CONTRATADA, fiscais de contrato para eventuais divergências.
- 9.1.1.4. Confecção e assinatura do Termo de Recebimento Definitivo para fins de encaminhamento para pagamento.

9.1.1.5. Receber definitivamente os serviços prestados, e enviar a Nota Fiscal, juntamente, com as Ordens de Serviços e documentos de manutenção contratual, para providências e pagamento.

#### 9.2. Grupo de Fiscalização Técnica do Contrato

9.2.1. Composto por servidores da CAPES.

9.2.2. Receber as Ordens de Serviços dos requisitantes, avaliar a compatibilidade contratual, registrar, autorizar e encaminhar à CONTRATADA para execução;

9.2.3. Verificar se os níveis de serviços contratados foram alcançados e aplicar as glosas estipuladas para cada caso;

9.2.4. Assinatura do Termo de Recebimento Definitivo e Termo de Recebimento Provisório para fins de encaminhamento para pagamento;

9.2.5. Receber e avaliar os Relatórios de Atividades entregues pela CONTRATADA.

#### 9.3. Fiscal Requisitante do Contrato

9.3.1. Servidor da CAPES.

9.3.2. Identificação de não conformidade com os termos contratuais.

9.3.3. Assinatura do Termo de Recebimento Definitivo para fins de encaminhamento para pagamento.

9.3.4. Verificação da manutenção da necessidade, economicidade e oportunidade da contratação.

#### 9.4. Fiscal Administrativo do Contrato

9.4.1. Servidor da CAPES.

9.4.2. Verificação da aderência da CONTRATADA com os termos contratuais;

9.4.3. Verificar a regularidade administrativa, fiscal, trabalhistas e previdenciárias da CONTRATADA, para fins de pagamento.

#### 9.5. Equipe Técnica da CONTRATADA

9.5.1. Profissional(is) da CONTRATADA devidamente certificado(s) e/ou capacitado(s) na solução.

9.5.2. Executar fielmente o contrato.

9.5.3. Efetuar, caso necessário, correções durante a execução contratual.

### 10. MATERIAIS A SEREM DISPONIBILIZADOS

10.1. A CONTRATANTE deverá fornecer as condições necessárias à execução regular do objeto do contrato, permitindo o acesso dos colaboradores às dependências da CONTRATANTE, respeitadas as normas de segurança vigentes.

10.2. Para a perfeita execução dos serviços, os materiais, equipamentos, ferramentas e utensílios necessários, nas quantidades estimadas e qualidades estabelecidas, promovendo sua substituição quando necessário, deverão ser disponibilizados pela CONTRATANTE quando os serviços forem prestados na CONTRATANTE e pela CONTRATADA quando os serviços forem prestados na CONTRATADA.

### 11. INFORMAÇÕES RELEVANTES PARA O DIMENSIONAMENTO DA PROPOSTA

11.1. Além de todas as informações já constantes do TR, as quais possibilitam à licitante o pleno conhecimento das características objetivas do serviço a ser contratado e sua complexidade geral, a serem levadas em consideração no dimensionamento da proposta, a demanda do órgão tem como base as seguintes características:

11.1.1. A possibilidade de a CONTRATADA prestar a quantidade de serviços estimadas no item 1.1 deste Termo de Referência, resguardada, ainda a prerrogativa de acréscimo de até 25% (vinte e cinco por cento), nas expressas situações legais autorizativas; e

11.1.2. Possuir, na assinatura do contrato, em quantidade suficiente para garantir o pleno atendimento dos serviços, os perfis profissionais definidos neste TR.

#### 12. UNIFORMES \*( NÃO SE APLICA)

#### 13. OBRIGAÇÕES DA CONTRATANTE

13.1. Exigir o cumprimento de todas as obrigações assumidas pela CONTRATADA, de acordo com as cláusulas contratuais, os termos de sua proposta e as condições deste TR;

13.2. Exercer o acompanhamento e a fiscalização dos serviços, por servidor especialmente designado, anotando em registro próprio as falhas detectadas, indicando dia, mês e ano, bem como o nome dos empregados eventualmente envolvidos, e encaminhando os apontamentos à autoridade competente para as providências cabíveis;

13.3. Notificar a CONTRATADA por escrito da ocorrência de eventuais imperfeições, falhas ou irregularidades constatadas no curso da execução dos serviços, fixando prazo para a sua correção, certificando-se que as soluções por ela propostas sejam as mais adequadas.

13.4. Não permitir que os empregados da Contratada realizem horas extras, exceto em caso de comprovada necessidade de serviço, formalmente justificada pela autoridade do órgão para o qual o trabalho seja prestado e desde que observado o limite da legislação trabalhista;

13.5. Pagar à CONTRATADA o valor resultante da prestação do serviço, no prazo e condições estabelecidas neste Termo de Referência;

13.6. Efetuar as retenções tributárias devidas sobre o valor da Nota Fiscal/Fatura da CONTRATADA, no que couber, em conformidade com o item 6 do Anexo XI da IN SEGES/MP n. 5/2017.

13.7. Não praticar atos de ingerência na administração da CONTRATADA, tais como:

13.7.1. exercer o poder de mando sobre os empregados da CONTRATADA, devendo reportar-se somente aos prepostos ou responsáveis por ela indicados, exceto quando o objeto da contratação previr o atendimento direto, tais como nos serviços de recepção e apoio ao usuário;

13.7.2. direcionar a contratação de pessoas para trabalhar nas empresas CONTRATADA;

13.7.3. considerar os trabalhadores da CONTRATADA como colaboradores eventuais do próprio órgão ou entidade responsável pela contratação, especialmente para efeito de concessão de diárias e passagens.

13.8. fiscalizar mensalmente, por amostragem, o cumprimento das obrigações trabalhistas, previdenciárias e para com o FGTS, especialmente:

13.8.1. A concessão de férias remuneradas e o pagamento do respectivo adicional, bem como de auxílio-transporte, auxílio-alimentação e auxílio-saúde, quando for devido;

13.8.2. O recolhimento das contribuições previdenciárias e do FGTS dos empregados que efetivamente participem da execução dos serviços contratados, a fim de verificar qualquer irregularidade;

13.8.3. O pagamento de obrigações trabalhistas e previdenciárias dos empregados dispensados até a data da extinção do contrato.

13.9. Analisar os termos de rescisão dos contratos de trabalho do pessoal empregado na prestação dos serviços no prazo de 30 (trinta) dias, prorrogável por igual período, após a extinção ou rescisão do contrato.

- 13.10. Fornecer por escrito as informações necessárias para o desenvolvimento dos serviços objeto do contrato;
- 13.11. Realizar avaliações periódicas da qualidade dos serviços, após seu recebimento;
- 13.12. Cientificar o órgão de representação judicial da Advocacia-Geral da União para adoção das medidas cabíveis quando do descumprimento das obrigações pela CONTRATADA;
- 13.13. Arquivar, entre outros documentos, projetos, "as builds", especificações técnicas, orçamentos, termos de recebimento, contratos e aditamentos, relatórios de inspeções técnicas após o recebimento do serviço e notificações expedidas;
- 13.14. Fiscalizar o cumprimento dos requisitos legais, quando a CONTRATADA houver se beneficiado da preferência estabelecida pelo art. 3º, § 5º, da Lei nº 8.666, de 1993;
- 13.15. Assegurar que o ambiente de trabalho, inclusive seus equipamentos e instalações, apresentem condições adequadas ao cumprimento, pela contratada, das normas de segurança e saúde no trabalho, quando o serviço for executado em suas dependências, ou em local por ela designado.
- 13.16. Credenciar junto à CONTRATANTE, seus profissionais autorizados a retirar e a entregar documentos, bem como aqueles que venham a ser designados para prestar serviços nas dependências da CONTRATANTE;
- 13.17. Permitir acesso aos profissionais da empresa CONTRATADA às suas dependências, sempre que necessário à execução contratual;
- 13.18. Prestar as informações e os esclarecimentos que venham a ser solicitados pelos técnicos da CONTRATADA;
- 13.19. Assistir e homologar serviços prestados, conforme definido em Contrato;
- 13.20. Notificar a empresa CONTRATADA, por escrito, acerca de todas as ocorrências atípicas registradas durante a execução contratual;
- 13.21. Rejeitar, no todo ou em parte, serviços prestados em desacordo com o escopo e especificações técnicas estabelecidas em Edital/Contrato;
- 13.22. Aplicar à empresa CONTRATADA, quando necessário, as sanções legais cabíveis, garantindo-lhe o direito à ampla defesa;
- 13.23. Emitir Ordens de Serviços, quando necessárias;
- 13.24. Designar os Fiscais e o Gestor do Contrato, nomeando-os quando da assinatura do Contrato;
- 13.25. Prover infraestrutura mínima necessária ao desenvolvimento das atividades previstas, ou seja, locais e materiais adequados, tais como espaço físico, mesas, cadeiras, energia elétrica, software e hardware, salvo os recursos materiais previstos neste TR e quando o serviço for prestado nas instalações da CONTRATADA, de acordo com a conveniência da CONTRATANTE; e
- 13.26. Emitir Termo de Recebimento Definitivo, quando os serviços prestados estiverem em acordo com o Termo de Referência.

#### 14. DAS OBRIGAÇÕES DA CONTRATADA

14.1. Alocar profissionais devidamente capacitados e habilitados para os serviços contratados, devendo também contratar como mão de obra pessoas presas ou egressos do sistema prisional, desde que tenham as condições técnicas exigidas no contrato, em aplicação ao disposto no Decreto nº 9.450/2018.

14.1.1. Aplicam-se as cautelas a seguir em relação à pessoa presa em regime fechado, em atendimento ao disposto nos art. 35 e art. 36 da Lei nº 7.210, de 1984:

- I - apresentação de prévia autorização do Juízo da Execução;
- II - comprovação de aptidão, disciplina e responsabilidade da pessoa presa;
- III - comprovação do cumprimento mínimo de um sexto da pena; e
- IV - observância do limite máximo de dez por cento do número de presos na prestação do serviço.

14.2. Executar os serviços conforme especificações deste Termo de Referência, de sua proposta e do contrato, com a alocação dos empregados necessários ao perfeito cumprimento das cláusulas contratuais, além de fornecer e utilizar os materiais e equipamentos, ferramentas e utensílios necessários, na qualidade e quantidade mínimas especificadas neste Termo de Referência e em sua proposta;

14.3. Reparar, corrigir, remover ou substituir, às suas expensas, no total ou em parte, no prazo fixado pelo fiscal do contrato, os serviços efetuados em que se verificarem vícios, defeitos ou incorreções resultantes da execução ou dos materiais empregados, sem que isso importe em custos à CONTRATANTE;

14.4. Manter a execução do serviço nos horários fixados pela Administração;

14.5. Responsabilizar-se pelos vícios e danos decorrentes da execução do objeto, bem como por todo e qualquer dano causado à União ou à entidade federal, devendo ressarcir imediatamente a Administração em sua integralidade, ficando a CONTRATANTE autorizada a descontar da garantia, caso exigida no edital, ou dos pagamentos devidos à CONTRATADA, o valor correspondente aos danos sofridos;

14.6. Utilizar empregados habilitados e com conhecimento pleno dos serviços a serem executados, em conformidade com as normas e determinações em vigor, atendendo à notificações da contratada quanto à atualização técnica dos profissionais alocados.

14.7. Vedar a utilização, na execução dos serviços, de empregado que seja familiar de agente público ocupante de cargo em comissão ou função de confiança no órgão CONTRATANTE, nos termos do artigo do Decreto nº 7.203, de 2010;

14.8. Disponibilizar à Contratante os empregados devidamente identificados por meio de crachá. Não se aplica a essa contratação a necessidade de colaboradores uniformizados.

14.9. As empresas contratadas que sejam regidas pela Consolidação das Leis do Trabalho (CLT) deverão apresentar a seguinte documentação no primeiro mês de prestação dos serviços, conforme alínea "g" do item 10.1 do Anexo VIII-B da IN SEGES/MP n. 5/2017:

14.9.1. relação dos empregados, contendo nome completo, cargo ou função, salário, horário do posto de trabalho, números da carteira de identidade (RG) e da inscrição no Cadastro de Pessoas Físicas (CPF), com indicação dos responsáveis técnicos pela execução dos serviços, quando for o caso;

14.9.2. Carteira de Trabalho e Previdência Social (CTPS) dos empregados admitidos e dos responsáveis técnicos pela execução dos serviços, quando for o caso, devidamente assinada pela contratada; e

14.9.3. exames médicos admissionais dos empregados da contratada que prestarão os serviços;

14.9.4. declaração de responsabilidade exclusiva da contratada sobre a quitação dos encargos trabalhistas e sociais decorrentes do contrato;

14.9.5. Os documentos acima mencionados deverão ser apresentados para cada novo empregado que se vincule à prestação do contrato administrativo. De igual modo, o desligamento de empregados no curso do contrato de prestação de serviços deve ser devidamente comunicado, com toda a documentação pertinente ao empregado dispensado, à semelhança do que se exige quando do encerramento do contrato administrativo.

14.10. Apresentar relação mensal dos empregados que expressamente optarem por não receber o vale transporte.

14.11. Quando não for possível a verificação da regularidade no Sistema de Cadastro de Fornecedores – SICAF, a empresa CONTRATADA deverá entregar ao setor responsável pela fiscalização do contrato, até o dia trinta do mês seguinte ao da prestação dos serviços, os seguintes documentos: 1) prova de regularidade relativa à Seguridade Social; 2) certidão conjunta relativa aos tributos federais e à Dívida Ativa da União; 3) certidões que comprovem a regularidade perante a Fazenda Municipal ou Distrital do domicílio ou sede do contratado; 4) Certidão de Regularidade do FGTS – CRF; e 5) Certidão Negativa de Débitos Trabalhistas – CNDT, conforme alínea "c" do item 10.2 do Anexo VIII-B da IN SEGES/MP n. 5/2017;

14.12. Substituir, no prazo de dez dias úteis, sem prejuízo do disposto no item 1.3 do ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364), em caso de eventual ausência, tais como faltas e licenças, o empregado posto a serviço da Contratante, devendo identificar previamente o respectivo substituto ao Fiscal do Contrato

14.13. Responsabilizar-se pelo cumprimento das obrigações previstas em Acordo, Convenção, Dissídio Coletivo de Trabalho ou equivalentes das categorias abrangidas pelo contrato, por todas as obrigações trabalhistas, sociais, previdenciárias, tributárias e as demais previstas em legislação específica, cuja inadimplência não transfere a responsabilidade à CONTRATANTE;

14.13.1. Não serão incluídas nas planilhas de custos e formação de preços as disposições contidas em Acordos, Dissídios ou Convenções Coletivas que tratem de pagamento de participação dos trabalhadores nos lucros ou resultados da empresa contratada, de matéria não trabalhista, de obrigações e direitos que somente se aplicam aos contratos com a Administração Pública, ou que estabeleçam direitos não previstos em lei, tais como valores ou índices obrigatórios de encargos sociais ou previdenciários, bem como de preços para os insumos relacionados ao exercício da atividade.

14.14. Efetuar o pagamento dos salários dos empregados alocados na execução contratual mediante depósito na conta bancária de titularidade do trabalhador, em agência situada na localidade ou região metropolitana em que ocorre a prestação dos serviços, de modo a possibilitar a conferência do pagamento por parte da Contratante. Em caso de impossibilidade de cumprimento desta disposição, a contratada deverá apresentar justificativa, a fim de que a Administração analise sua plausibilidade e possa verificar a realização do pagamento.

14.15. Autorizar a Administração contratante, no momento da assinatura do contrato, a fazer o desconto nas faturas e realizar os pagamentos dos salários e demais verbas trabalhistas diretamente aos trabalhadores, bem como das contribuições previdenciárias e do FGTS, quando não demonstrado o cumprimento tempestivo e regular dessas obrigações, até o momento da regularização, sem prejuízo das sanções cabíveis.

14.15.1. Quando não for possível a realização desses pagamentos pela própria Administração (ex.: por falta da documentação pertinente, tais como folha de pagamento, rescisões dos contratos e guias de recolhimento), os valores retidos cautelarmente serão depositados junto à Justiça do Trabalho, com o objetivo de serem utilizados exclusivamente no pagamento de salários e das demais verbas trabalhistas, bem como das contribuições sociais e FGTS decorrentes.

14.16. Não permitir que o empregado designado para trabalhar em um turno preste seus serviços no turno imediatamente subsequente;

14.17. Atender às solicitações da Contratante quanto à substituição dos empregados alocados, no prazo fixado pelo fiscal do contrato, nos casos em que ficar constatado descumprimento das obrigações relativas à execução do serviço, conforme descrito neste Termo de Referência;

14.18. Instruir seus empregados quanto à necessidade de acatar as Normas Internas da Administração;

14.19. Instruir seus empregados a respeito das atividades a serem desempenhadas, alertando-os a não executar atividades não abrangidas pelo contrato, devendo a Contratada relatar à Contratante toda e qualquer ocorrência neste sentido, a fim de evitar desvio de função;

14.20. Instruir seus empregados, no início da execução contratual, quanto à obtenção das informações de seus interesses junto aos órgãos públicos, relativas ao contrato de trabalho e obrigações a ele inerentes, adotando, entre outras, as seguintes medidas:

14.20.1. viabilizar o acesso de seus empregados, via internet, por meio de senha própria, aos sistemas da Previdência Social e da Receita do Brasil, com o objetivo de verificar se as suas contribuições previdenciárias foram recolhidas, no prazo máximo de 60 (sessenta) dias, contados do início da prestação dos serviços ou da admissão do empregado;

14.20.2. viabilizar a emissão do cartão cidadão pela Caixa Econômica Federal para todos os empregados, no prazo máximo de 60 (sessenta) dias, contados do início da prestação dos serviços ou da admissão do empregado;

14.20.3. oferecer todos os meios necessários aos seus empregados para a obtenção de extratos de recolhimentos de seus direitos sociais, preferencialmente por meio eletrônico, quando disponível.

14.21. Manter preposto nos locais de prestação de serviço, aceito pela Administração, para representá-la na execução do contrato;

14.22. Relatar à Contratante toda e qualquer irregularidade verificada no decorrer da prestação dos serviços;

14.23. Fornecer, sempre que solicitados pela Contratante, os comprovantes do cumprimento das obrigações previdenciárias, do Fundo de Garantia do Tempo de Serviço - FGTS, e do pagamento dos salários e demais benefícios trabalhistas dos empregados colocados à disposição da Contratante;

14.23.1. A ausência da documentação pertinente ou da comprovação do cumprimento das obrigações trabalhistas, previdenciárias e relativas ao FGTS implicará a retenção do pagamento da fatura mensal, em valor proporcional ao inadimplemento, mediante prévia comunicação, até que a situação seja regularizada, sem prejuízo das demais sanções cabíveis.

14.23.2. Ultrapassado o prazo de 15 (quinze) dias, contados na comunicação mencionada no subitem anterior, sem a regularização da falta, a Administração poderá efetuar o pagamento das obrigações diretamente aos empregados da contratada que tenham participado da execução dos serviços objeto do contrato, sem prejuízo das demais sanções cabíveis.

14.23.2.1. O sindicato representante da categoria do trabalhador deverá ser notificado pela contratante para acompanhar o pagamento das respectivas verbas.

14.24. Não permitir a utilização de qualquer trabalho do menor de dezesseis anos, exceto na condição de aprendiz para os maiores de quatorze anos; nem permitir a utilização do trabalho do menor de dezoito anos em trabalho noturno, perigoso ou insalubre;

14.25. Manter durante toda a vigência do contrato, em compatibilidade com as obrigações assumidas, todas as condições de habilitação e qualificação exigidas na licitação;

14.26. Guardar sigilo sobre todas as informações obtidas em decorrência do cumprimento do contrato;

14.27. Não beneficiar-se da condição de optante pelo Simples Nacional, salvo as exceções previstas no § 5º-C do art. 18 da Lei Complementar no 123, de 14 de dezembro de 2006;

14.28. Comunicar formalmente à Receita Federal a assinatura do contrato de prestação de serviços mediante cessão de mão de obra, salvo as exceções previstas no § 5º-C do art. 18 da Lei Complementar no 123, de 14 de dezembro de 2006, para fins de exclusão obrigatória do Simples Nacional a contar do mês seguinte ao da contratação, conforme previsão do art.17, XII, art.30, §1º, II e do art. 31, II, todos da LC 123, de 2006.

14.28.1. Para efeito de comprovação da comunicação, a contratada deverá apresentar cópia do ofício enviado à Receita Federal do Brasil, com comprovante de entrega e recebimento, comunicando a assinatura do contrato de prestação de serviços mediante cessão de mão de obra, até o último dia útil do mês subsequente ao da ocorrência da situação de vedação.

14.29. Arcar com o ônus decorrente de eventual equívoco no dimensionamento dos quantitativos de sua proposta, inclusive quanto aos custos variáveis decorrentes de fatores futuros e incertos, tais como os valores providos com o quantitativo de vale transporte, devendo complementá-los, caso o previsto inicialmente em sua proposta não seja satisfatório para o atendimento do objeto da licitação, exceto quando ocorrer algum dos eventos arrolados nos incisos do § 1º do art. 57 da Lei nº 8.666, de 1993.

14.30. Comunicar ao Fiscal do contrato, no prazo de 24 (vinte e quatro) horas, qualquer ocorrência anormal ou acidente que se verifique no local dos serviços

14.31. Prestar todo esclarecimento ou informação solicitada pela CONTRATANTE ou por seus prepostos, garantindo-lhes o acesso, a qualquer tempo, ao local dos trabalhos, bem como aos documentos relativos à execução do serviço.

- 14.32. Paralisar, por determinação da CONTRATANTE, qualquer atividade que não esteja sendo executada de acordo com a boa técnica ou que ponha em risco a segurança de pessoas ou bens de terceiros.
- 14.33. Promover a guarda, manutenção e vigilância de materiais, ferramentas, e tudo o que for necessário à execução dos serviços, durante a vigência do contrato.
- 14.34. Promover a organização técnica e administrativa dos serviços, de modo a conduzi-los eficaz e eficientemente, de acordo com os documentos e especificações que integram este Termo de Referência, no prazo determinado.
- 14.35. Conduzir os trabalhos com estrita observância às normas da legislação pertinente, cumprindo as determinações dos Poderes Públicos, mantendo sempre limpo o local dos serviços e nas melhores condições de segurança, higiene e disciplina.
- 14.36. Submeter previamente, por escrito, à CONTRATANTE, para análise e aprovação, quaisquer mudanças nos métodos executivos que fujam às especificações do memorial descritivo.
- 14.37. Não permitir a utilização de qualquer trabalho do menor de dezesseis anos, exceto na condição de aprendiz para os maiores de quatorze anos; nem permitir a utilização do trabalho do menor de dezoito anos em trabalho noturno, perigoso ou insalubre;
- 14.38. Manter durante toda a vigência do contrato, em compatibilidade com as obrigações assumidas, todas as condições de habilitação e qualificação exigidas na licitação;
- 14.39. Cumprir, durante todo o período de execução do contrato, a reserva de cargos prevista em lei para pessoa com deficiência ou para reabilitado da Previdência Social, bem como as regras de acessibilidade previstas na legislação, quando a CONTRATADA houver se beneficiado da preferência estabelecida pela Lei nº 13.146, de 2015.
- 14.40. Guardar sigilo sobre todas as informações obtidas em decorrência do cumprimento do contrato;
- 14.41. Arcar com o ônus decorrente de eventual equívoco no dimensionamento dos quantitativos de sua proposta, inclusive quanto aos custos variáveis decorrentes de fatores futuros e incertos, tais como os valores providos com o quantitativo de vale transporte, devendo complementá-los, caso o previsto inicialmente em sua proposta não seja satisfatório para o atendimento do objeto da licitação, exceto quando ocorrer algum dos eventos arrolados nos incisos do § 1º do art. 57 da Lei nº 8.666, de 1993.
- 14.42. Cumprir, além dos postulados legais vigentes de âmbito federal, estadual ou municipal, as normas de segurança da CONTRATANTE;
- 14.43. Prestar os serviços dentro dos parâmetros e rotinas estabelecidos, fornecendo todos os materiais, equipamentos e utensílios em quantidade, qualidade e tecnologia adequadas, com a observância às recomendações aceitas pela boa técnica, normas e legislação;
- 14.44. Assegurar à CONTRATANTE, em conformidade com o previsto no subitem 6.1, "a"e "b", do Anexo VII – F da Instrução Normativa SEGES/MP nº 5, de 25/05/2017:
- 14.44.1. O direito de propriedade intelectual dos produtos desenvolvidos, inclusive sobre as eventuais adequações e atualizações que vierem a ser realizadas, logo após o recebimento de cada parcela, de forma permanente, permitindo à Contratante distribuir, alterar e utilizar os mesmos sem limitações;
- 14.44.2. Os direitos autorais da solução, do projeto, de suas especificações técnicas, da documentação produzida e congêneres, e de todos os demais produtos gerados na execução do contrato, inclusive aqueles produzidos por terceiros subcontratados, ficando proibida a sua utilização sem que exista autorização expressa da CONTRATANTE, sob pena de multa, sem prejuízo das sanções civis e penais cabíveis.
- 14.45. Realizar a transição contratual com transferência de conhecimento, tecnologia e técnicas empregadas, sem perda de informações, podendo exigir, inclusive, a capacitação dos técnicos da CONTRATANTE ou da nova empresa que continuará a execução dos serviços;
- 14.46. Manter confidencialidade e, em nenhum momento, divulgar a terceiros, sem a ciência e o consentimento da CONTRATANTE, documentos, imagens/fotos, dados ou outra informação que tiver sido direta ou indiretamente proporcionada pela CONTRATANTE, antes, durante ou depois da execução do contrato;
- 14.47. Manter, durante toda a execução dos serviços e vigência do Contrato, as condições de habilitação e qualificação exigidas no processo licitatório e posteriores necessidades de mudanças tecnológicas, devidamente informadas;
- 14.48. Recrutar e contratar mão-de-obra qualificada em seu nome e sob sua responsabilidade, sem qualquer solidariedade da CONTRATANTE, cabendo-lhe efetuar todos os pagamentos, inclusive os relativos aos encargos previstos na legislação trabalhista, previdenciária e fiscal, bem como de seguros e quaisquer outros decorrentes da sua condição de empregadora, assumindo, ainda, com relação ao contingente alocado, total responsabilidade pela coordenação e supervisão dos encargos administrativos, tais como: controle, fiscalização e orientação técnica, controle de frequência, ausências permitidas, licenças autorizadas, férias, punições, admissões, demissões, transferências e promoções;
- 14.49. Manter em dia todos os direitos pecuniários de seus colaboradores, tais como horas extras, adicionais noturnos, indenizações e outras vantagens, de forma que não sejam os trabalhos da CONTRATANTE prejudicados em função de reivindicações por parte dos seus colaboradores;
- 14.50. Assumir todas as despesas e ônus relativos à pessoal e a quaisquer outras derivadas ou conexas com o Contrato, ficando ainda, para todos os efeitos legais, expressada pela CONTRATADA, a inexistência de qualquer vínculo empregatício entre seus colaboradores e/ou preposto e a CONTRATANTE;
- 14.51. Indenizar todos os custos e despesas financeiras que porventura venham a ser suportados pela CONTRATANTE por força de sentença judicial que aponte a existência de vínculo empregatício entre seus colaboradores e/ou preposto da CONTRATADA;
- 14.52. Assumir todas as providências e obrigações estabelecidas na legislação específica de acidentes de trabalho quando, em ocorrência da espécie, forem vítimas os seus colaboradores no desempenho dos serviços ou em conexão com eles, ainda que verificados nas dependências da CONTRATANTE;
- 14.53. Informar à CONTRATANTE, para efeito de controle de acesso às suas dependências, os dados que constam no formulário de acesso às dependências da CONTRATANTE, como: nome, CPF e o respectivo número da carteira de identidade dos colaboradores disponibilizados para a prestação de serviços, bem como ocorrências de afastamento definitivo e as substituições em casos de falta, ausência legal ou férias;
- 14.54. Para controle de acesso dos colaboradores, a CONTRATADA deverá preencher o Encarte Técnico Contratação - TI CGS (0987886) ENCARTE G - MODELO DE TERMOS DE CREDENCIAMENTO E DESCREDENCIAMENTO.
- 14.55. Substituir, sempre que solicitado pela CONTRATANTE, qualquer colaborador cuja atuação, permanência e/ou comportamento sejam julgados prejudiciais, inconvenientes insatisfatórios às normas de disciplina, ou ainda, incompatíveis com o exercício das funções que lhe foram atribuídas, quando os serviços forem executados nas dependências da CONTRATANTE;
- 14.56. Assegurar, nos casos de greve ou paralisação de seus colaboradores, a continuação da prestação dos serviços, por meio da execução de plano de contingência, inclusive nos casos de greve ou paralisação dos transportes públicos, hipótese em que deverá promover, às suas expensas, os meios necessários para que seus colaboradores cheguem aos seus locais de trabalho;
- 14.57. Assegurar, nos casos de desastres naturais, acidentes, falhas de equipamentos, falhas de segurança, perda de serviços e ações intencionais, que por ventura possam ocorrer em seu ambiente, a continuidade da prestação dos serviços, por meio da execução de planos de contingência, visando à recuperação das operações a tempo de não causar paralisação dos serviços prestados à CONTRATANTE;
- 14.58. Certificar-se de que o Preposto mantenha contato com os servidores da CONTRATANTE, responsáveis pela fiscalização do contrato, adotando as providências requeridas à execução dos serviços pelos empregados da CONTRATADA, comandando, coordenando e observando a execução dos serviços por estes prestados. Ressalta-se que o Preposto deverá ser alocado no ambiente da CONTRATANTE, devendo a CONTRATADA indicar também os eventuais substitutos, que deverão responder com grau de autonomia decisória equivalente ao primeiro.

14.59. Obedecer às prerrogativas de Segurança Interna estabelecida nas Normas Internas de Segurança dos Órgãos participantes e respeitar o Código de Ética Profissional do Servidor Público Civil do Poder Executivo Federal;

14.60. Administrar todo e qualquer assunto relativo aos empregados alocados para execução dos serviços, operacionalizando o Contrato;

14.61. Zelar para que todos os privilégios de acesso a sistema, informação e qualquer outro recurso da CONTRATANTE sejam utilizados exclusivamente na execução dos serviços e pelo tempo estritamente essencial à realização dos mesmos;

14.62. Encaminhar à unidade fiscalizadora todas as faturas dos serviços prestados, detalhando todos os serviços/eventos e suas referidas pontuações para conferência e ateste pelo Gestor/Fiscal do contrato;

14.63. Reportar imediatamente à CONTRATANTE qualquer anormalidade, erro ou irregularidade que possa comprometer a execução dos serviços e o bom andamento das atividades previstas em contrato;

14.64. Responsabilizar-se pelo fornecimento do objeto do Termo de Referência, respondendo civil e criminalmente por todos os danos, perdas e prejuízos que, por dolo ou culpa sua, de seus empregados, preposto, ou terceiros no exercício de suas atividades, vier a, direta ou indiretamente, causar ou provocar à CONTRATANTE;

14.65. Obter todas as licenças, aprovações e franquias necessárias à execução dos fornecimentos e dos serviços, pagando os emolumentos prescritos por lei e observando as leis, regulamentos e posturas aplicáveis. É obrigatório o cumprimento de quaisquer formalidades e o pagamento, à sua custa, das multas porventura impostas pelas autoridades, mesmo daquelas que, por força dos dispositivos legais, sejam atribuídas à Administração Pública;

14.66. Não ceder ou transferir, total ou parcialmente, parte alguma do contrato. A fusão, cisão ou incorporação só será admitida com o consentimento prévio e por escrito da CONTRATANTE;

14.67. Abster-se, qualquer que seja a hipótese, de veicular publicidade ou qualquer outra informação acerca das atividades objeto do Contrato, sem prévia autorização da CONTRATANTE;

14.68. Dar ciência, imediatamente e por escrito, de qualquer anormalidade que verificar na execução do objeto, bem como, prestar esclarecimentos que forem solicitados pela CONTRATANTE;

14.69. Manter sigilo, sob pena de responsabilidade, sobre todo e qualquer assunto de interesse da CONTRATANTE ou de terceiros de que tomar conhecimento em razão da execução do objeto do Contrato, respeitando todos os critérios estabelecidos, aplicáveis aos dados, informações e às regras de negócio;

14.70. Executar os serviços solicitados formalmente pela CONTRATANTE por meio de Ordens de Serviço, de acordo com as suas características e com as cláusulas do Contrato firmado;

14.71. Somente desativar o software e qualquer outro recurso computacional relacionado à execução do objeto, mediante prévia autorização da CONTRATANTE;

14.72. Prestar qualquer tipo de informação solicitada pela CONTRATANTE sobre os fornecimentos e sobre os serviços contratados, bem como fornecer qualquer documentação julgada necessária ao perfeito entendimento do objeto do Termo de Referência;

14.73. Elaborar e apresentar documentação técnica dos fornecimentos e serviços executados, nas datas aprazadas, visando homologação da mesma pela CONTRATANTE;

14.74. Não deixar de executar qualquer atividade necessária ao perfeito fornecimento do objeto, sob qualquer alegação, mesmo sob pretexto de não ter sido executada anteriormente qualquer tipo de procedimento;

14.75. Implementar rigorosa gerência de contrato com observância a todas as disposições constantes do Termo de Referência;

14.76. Atender aos prazos estabelecidos e acordados nas ordens de Serviço abertas pela CONTRATANTE, observando o disposto no Termo de Referência;

14.77. Reparar, corrigir, remover, reconstruir ou substituir às suas expensas, no todo ou em parte, serviços efetuados em que se verificarem vícios, defeitos e incorreções;

14.78. A CONTRATADA não poderá de forma alguma comercializar, no todo ou em parte, qualquer produto ou sistema gerado a partir do Contrato resultante do Termo de Referência, sob pena de rescisão contratual, podendo inclusive responder por ações judiciais decorrentes do não cumprimento deste item;

14.79. Sujectar-se às normas internas da CONTRATANTE em suas dependências, principalmente as de segurança, incluindo aqueles referentes à identificação, trajes, trânsito e permanência;

14.80. Planejar, desenvolver, executar e manter os serviços objeto desse Contrato, dentro das exigências de níveis de serviços aqui estabelecidos;

14.81. Assumir, ainda, a responsabilidade pelos encargos fiscais e comerciais;

14.82. A inadimplência da CONTRATADA, com referência aos encargos estabelecidos nas condições anteriores, não transfere a responsabilidade por seu pagamento a CONTRATANTE, nem poderá onerar o objeto deste contrato, razão pela qual a CONTRATADA, renuncia expressamente a qualquer vínculo de solidariedade, ativa ou passiva, para com a CONTRATANTE;

14.83. Adotar as providências necessárias que viabilizem a realização dos serviços objeto deste contrato.

## 15. DA SUBCONTRATAÇÃO E A PARTICIPAÇÃO DE EMPRESAS EM CONSÓRCIOS

15.1. Não será admitida a subcontratação do objeto licitatório.

15.2. É vedada a participação de consórcios de empresas, quaisquer que sejam suas formas de constituição, dadas as características específicas da contratação de serviços a serem fornecidos, que não pressupõem complexidade e multiplicidade de atividades empresariais distintas.

## 16. ALTERAÇÃO SUBJETIVA

16.1. É admissível a fusão, cisão ou incorporação da CONTRATADA com/ou outra pessoa jurídica, desde que sejam observados pela nova pessoa jurídica todos os requisitos de habilitação exigidos na licitação original; sejam mantidas as demais cláusulas e condições do contrato; não haja prejuízo à execução do objeto pactuado e haja a anuência expressa da Administração à continuidade do contrato.

## 17. CONTROLE E FISCALIZAÇÃO DA EXECUÇÃO

17.1. O acompanhamento e a fiscalização da execução do contrato consistem na verificação da conformidade da prestação dos serviços, dos materiais, técnicas e equipamentos empregados, de forma a assegurar o perfeito cumprimento do ajuste, que serão exercidos por um ou mais representantes da CONTRATANTE, especialmente designados, na forma dos arts. 67 e 73 da Lei nº 8.666, de 1993.

17.2. As atividades de gestão e fiscalização da execução contratual são o conjunto de ações que tem por objetivo aferir o cumprimento dos resultados previstos pela Administração para o serviço contratado, verificar a regularidade das obrigações previdenciárias, fiscais e trabalhistas, bem como prestar apoio à instrução processual e o encaminhamento da documentação pertinente ao setor de contratos para a formalização dos procedimentos relativos a repactuação, alteração, reequilíbrio, prorrogação, pagamento, eventual aplicação de sanções, extinção do contrato, dentre outras, com vista a assegurar o cumprimento das cláusulas avançadas e a solução de problemas relativos ao objeto.

17.3. O conjunto de atividades de gestão e fiscalização compete ao gestor da execução do contrato, podendo ser auxiliado pela fiscalização técnica, administrativa, setorial e pelo público usuário, de acordo com as seguintes disposições:

I – Gestão da Execução do Contrato: é a coordenação das atividades relacionadas à fiscalização técnica, administrativa, setorial e pelo público usuário, bem como dos atos preparatórios à instrução processual e ao encaminhamento da documentação pertinente ao setor de contratos para formalização dos procedimentos quanto aos aspectos que envolvam a prorrogação, alteração, reequilíbrio, pagamento, eventual aplicação de sanções, extinção do contrato, dentre outros;

II – Fiscalização Técnica: é o acompanhamento com o objetivo de avaliar a execução do objeto nos moldes contratados e, se for o caso, aferir se a quantidade, qualidade, tempo e modo da prestação dos serviços estão compatíveis com os indicadores de níveis mínimos de desempenho estipulados no ato convocatório, para efeito de pagamento conforme o resultado, podendo ser auxiliado pela fiscalização pelo público usuário;

III – Fiscalização Administrativa: é o acompanhamento dos aspectos administrativos da execução dos serviços, quanto às obrigações previdenciárias, fiscais e trabalhistas, bem como quanto às providências tempestivas nos casos de inadimplemento;

17.4. Quando a contratação exigir fiscalização setorial, o órgão ou entidade deverá designar representantes nesses locais para atuarem como fiscais setoriais.

17.5. As atividades de gestão e fiscalização da execução contratual devem ser realizadas de forma preventiva, rotineira e sistemática, podendo ser exercidas por servidores, equipe de fiscalização ou único servidor, desde que, no exercício dessas atribuições, fique assegurada a distinção dessas atividades e, em razão do volume de trabalho, não comprometa o desempenho de todas as ações relacionadas à Gestão do Contrato.

17.6. A fiscalização administrativa poderá ser efetivada com base em critérios estatísticos, levando-se em consideração falhas que impactem o contrato como um todo e não apenas erros e falhas eventuais no pagamento de alguma vantagem a um determinado empregado.

17.7. Na fiscalização do cumprimento das obrigações trabalhistas e sociais exigir-se-á, dentre outras, as seguintes comprovações (os documentos poderão ser originais ou cópias autenticadas por cartório competente ou por servidor da Administração), no caso de empresas regidas pela Consolidação das Leis do Trabalho (CLT):

a) no primeiro mês da prestação dos serviços, a CONTRATADA deverá apresentar a seguinte documentação:

a.1. relação dos empregados, contendo nome completo, cargo ou função, horário do posto de trabalho, números da carteira de identidade (RG) e da inscrição no Cadastro de Pessoas Físicas (CPF), com indicação dos responsáveis técnicos pela execução dos serviços, quando for o caso;

a.2. Carteira de Trabalho e Previdência Social (CTPS) dos empregados admitidos e dos responsáveis técnicos pela execução dos serviços, quando for o caso, devidamente assinada pela CONTRATADA; e

a.3. exames médicos admissionais dos empregados da CONTRATADA que prestarão os serviços.

b) entrega até o dia trinta do mês seguinte ao da prestação dos serviços ao setor responsável pela fiscalização do contrato dos seguintes documentos, quando não for possível a verificação da regularidade destes no Sistema de Cadastro de Fornecedores (SICAF):

b.1. Certidão Negativa de Débitos relativos a Créditos Tributários Federais e à Dívida Ativa da União (CND);

b.2. certidões que comprovem a regularidade perante as Fazendas Estadual, Distrital e Municipal do domicílio ou sede do contratado;

b.3. Certidão de Regularidade do FGTS (CRF); e

b.4. Certidão Negativa de Débitos Trabalhistas (CNDT).

c) entrega, quando solicitado pela CONTRATANTE, de quaisquer dos seguintes documentos:

c.1. extrato da conta do INSS e do FGTS de qualquer empregado, a critério da CONTRATANTE;

c.2. cópia da folha de pagamento analítica de qualquer mês da prestação dos serviços, em que conste como tomador CONTRATANTE;

c.3. cópia dos contracheques dos empregados relativos a qualquer mês da prestação dos serviços ou, ainda, quando necessário, cópia de recibos de depósitos bancários;

c.4. comprovantes de entrega de benefícios suplementares (vale-transporte, vale-alimentação, entre outros), a que estiver obrigada por força de lei ou de Convenção ou Acordo Coletivo de Trabalho, relativos a qualquer mês da prestação dos serviços e de qualquer empregado; e

c.5. comprovantes de realização de eventuais cursos de treinamento e reciclagem que forem exigidos por lei ou pelo contrato.

d) entrega de cópia da documentação abaixo relacionada, quando da extinção ou rescisão do contrato, após o último mês de prestação dos serviços, no prazo definido no contrato:

d.1. termos de rescisão dos contratos de trabalho dos empregados prestadores de serviço, devidamente homologados, quando exigível pelo sindicato da categoria;

d.2. guias de recolhimento da contribuição previdenciária e do FGTS, referentes às rescisões contratuais;

d.3. extratos dos depósitos efetuados nas contas vinculadas individuais do FGTS de cada empregado dispensado;

d.4. exames médicos demissionais dos empregados dispensados.

17.8. A CONTRATANTE deverá analisar a documentação solicitada na alínea “d” acima no prazo de 30 (trinta) dias após o recebimento dos documentos, prorrogáveis por mais 30 (trinta) dias, justificadamente.

17.9. No caso de sociedades diversas, tais como as Organizações Sociais, será exigida a comprovação de atendimento a eventuais obrigações decorrentes da legislação que rege as respectivas organizações.

17.10. Sempre que houver admissão de novos empregados pela contratada, os documentos elencados no subitem 16.7 acima deverão ser apresentados.

17.11. Em caso de indício de irregularidade no recolhimento das contribuições previdenciárias, os fiscais ou gestores do contrato deverão oficiar à Receita Federal do Brasil (RFB).

17.12. Em caso de indício de irregularidade no recolhimento da contribuição para o FGTS, os fiscais ou gestores do contrato deverão oficiar ao Ministério do Trabalho.

17.13. O descumprimento das obrigações trabalhistas ou a não manutenção das condições de habilitação pela CONTRATADA poderá dar ensejo à rescisão contratual, sem prejuízo das demais sanções.

17.14. A CONTRATANTE poderá conceder prazo para que a CONTRATADA regularize suas obrigações trabalhistas ou suas condições de habilitação, sob pena de rescisão contratual, quando não identificar má-fé ou a incapacidade de correção.

17.15. Além das disposições acima citadas, a fiscalização administrativa observará, ainda, as seguintes diretrizes:

17.15.1. Fiscalização inicial (no momento em que a prestação de serviços é iniciada):

a) Será elaborada planilha-resumo de todo o contrato administrativo, com informações sobre todos os empregados terceirizados que prestam serviços, com os seguintes dados: nome completo, número de inscrição no CPF, função exercida, salário, adicionais, gratificações, benefícios recebidos, sua especificação e quantidade (vale-transporte, auxílio-alimentação), horário de trabalho, férias, licenças, faltas, ocorrências e horas extras trabalhadas;

- b) Todas as anotações contidas na CTPS dos empregados serão conferidas, a fim de que se possa verificar se as informações nelas inseridas coincidem com as informações fornecidas pela CONTRATADA e pelo empregado;
- c) O número de terceirizados por função deve coincidir com o previsto no contrato administrativo;
- d) O salário não pode ser inferior ao previsto no contrato administrativo e na Convenção Coletiva de Trabalho da Categoria (CCT);
- e) Serão consultadas eventuais obrigações adicionais constantes na CCT para a CONTRATADA;
- f) Será verificada a existência de condições insalubres ou de periculosidade no local de trabalho que obriguem a empresa a fornecer determinados Equipamentos de Proteção Individual (EPI).
- g) No primeiro mês da prestação dos serviços, a contratada deverá apresentar a seguinte documentação:
  - g.1. relação dos empregados, com nome completo, cargo ou função, horário do posto de trabalho, números da carteira de identidade (RG) e inscrição no Cadastro de Pessoas Físicas (CPF), e indicação dos responsáveis técnicos pela execução dos serviços, quando for o caso;
  - g.2. CTPS dos empregados admitidos e dos responsáveis técnicos pela execução dos serviços, quando for o caso, devidamente assinadas pela contratada;
  - g.3. exames médicos admissionais dos empregados da contratada que prestarão os serviços; e
  - g.4. declaração de responsabilidade exclusiva da contratada sobre a quitação dos encargos trabalhistas e sociais decorrentes do contrato.

17.15.2. **Fiscalização mensal (a ser feita antes do pagamento da fatura):**

- a) Deve ser feita a retenção da contribuição previdenciária no valor de 11% (onze por cento) sobre o valor da fatura e dos impostos incidentes sobre a prestação do serviço;
- b) Deve ser consultada a situação da empresa junto ao SICAF;
- c) Serão exigidos a Certidão Negativa de Débito (CND) relativa a Créditos Tributários Federais e à Dívida Ativa da União, o Certificado de Regularidade do FGTS (CRF) e a Certidão Negativa de Débitos Trabalhistas (CNDT), caso esses documentos não estejam regularizados no SICAF;
- d) Deverá ser exigida, quando couber, comprovação de que a empresa mantém reserva de cargos para pessoa com deficiência ou para reabilitado da Previdência Social, conforme disposto no art. 66-A da Lei nº 8.666, de 1993.

17.15.3. **Fiscalização diária:**

- a) Devem ser evitadas ordens diretas da CONTRATANTE dirigidas aos terceirizados. As solicitações de serviços devem ser dirigidas ao preposto da empresa. Da mesma forma, eventuais reclamações ou cobranças relacionadas aos empregados terceirizados devem ser dirigidas ao preposto.
- b) Toda e qualquer alteração na forma de prestação do serviço, como a negociação de folgas ou a compensação de jornada, deve ser evitada, uma vez que essa conduta é exclusiva da CONTRATADA.
- c) Devem ser conferidos, por amostragem, diariamente, os empregados terceirizados que estão prestando serviços e em quais funções, e se estão cumprindo a jornada de trabalho.

17.16. Cabe, ainda, à fiscalização do contrato, verificar se a CONTRATADA observa a legislação relativa à concessão de férias e licenças aos empregados, respeita a estabilidade provisória de seus empregados e observa a data-base da categoria prevista na CCT, concedendo os reajustes dos empregados no dia e percentual previstos.

17.16.1. O gestor deverá verificar a necessidade de se proceder a repactuação do contrato, inclusive quanto à necessidade de solicitação da contratada.

17.17. A CONTRATANTE deverá solicitar, por amostragem, aos empregados, seus extratos da conta do FGTS e que verifiquem se as contribuições previdenciárias e do FGTS estão sendo recolhidas em seus nomes.

17.17.1. Ao final de um ano, todos os empregados devem ter seus extratos avaliados.

17.18. A CONTRATADA deverá entregar, no prazo de 15 (quinze) dias, quando solicitado pela CONTRATANTE quaisquer dos seguintes documentos:

- a) extrato da conta do INSS e do FGTS de qualquer empregado, a critério da CONTRATANTE;
- b) cópia da folha de pagamento analítica de qualquer mês da prestação dos serviços, em que conste como tomador a CONTRATANTE;
- c) cópia dos contracheques assinados dos empregados relativos a qualquer mês da prestação dos serviços ou, ainda, quando necessário, cópia de recibos de depósitos bancários; e
- d) comprovantes de entrega de benefícios suplementares (vale-transporte, vale-alimentação, entre outros), a que estiver obrigada por força de lei, Acordo, Convenção ou Dissídio Coletivo de Trabalho, relativos a qualquer mês da prestação dos serviços e de qualquer empregado.

17.19. Durante a execução do objeto, o fiscal técnico deverá monitorar constantemente o nível de qualidade dos serviços para evitar a sua degeneração, devendo intervir para requerer à CONTRATADA a correção das faltas, falhas e irregularidades constatadas.

17.20. O fiscal técnico deverá apresentar ao preposto da CONTRATADA a avaliação da execução do objeto ou, se for o caso, a avaliação de desempenho e qualidade da prestação dos serviços realizada.

17.21. Em hipótese alguma, será admitido que a própria CONTRATADA materialize a avaliação de desempenho e qualidade da prestação dos serviços realizada.

17.22. A CONTRATADA poderá apresentar justificativa para a prestação do serviço com menor nível de conformidade, que poderá ser aceita pelo fiscal técnico, desde que comprovada a excepcionalidade da ocorrência, resultante exclusivamente de fatores imprevisíveis e alheios ao controle do prestador.

17.23. Na hipótese de comportamento contínuo de desconformidade da prestação do serviço em relação à qualidade exigida, bem como quando esta ultrapassar os níveis mínimos toleráveis previstos nos indicadores, além dos fatores redutores, devem ser aplicadas as sanções à CONTRATADA de acordo com as regras previstas no ato convocatório.

17.24. O fiscal técnico poderá realizar avaliação diária, semanal ou mensal, desde que o período escolhido seja suficiente para avaliar ou, se for o caso, aferir o desempenho e qualidade da prestação dos serviços.

17.25. O representante da Contratante deverá ter a qualificação necessária para o acompanhamento e controle da execução dos serviços e do contrato.

17.26. A verificação da adequação da prestação do serviço deverá ser realizada com base nos critérios previstos neste Termo de Referência.

17.27. A fiscalização do contrato, ao verificar que houve subdimensionamento da produtividade pactuada, sem perda da qualidade na execução do serviço, deverá comunicar à autoridade responsável para que esta promova a adequação contratual à produtividade efetivamente realizada, respeitando-se os limites de alteração dos valores contratuais previstos no § 1º do art. 65 da Lei nº 8.666, de 1993.

17.28. A conformidade do material a ser utilizado na execução dos serviços deverá ser verificada juntamente com o documento da CONTRATADA que contenha sua relação detalhada, de acordo com o estabelecido neste Termo de Referência e na proposta, informando as respectivas quantidades e especificações técnicas, tais como: marca, qualidade e forma de uso.

17.29. O representante da CONTRATANTE deverá promover o registro das ocorrências verificadas, adotando as providências necessárias ao fiel cumprimento das cláusulas contratuais, conforme o disposto nos §§ 1º e 2º do art. 67 da Lei nº 8.666, de 1993.

17.30. O descumprimento total ou parcial das obrigações e responsabilidades assumidas pela CONTRATADA, incluindo o descumprimento das obrigações trabalhistas, não recolhimento das contribuições sociais, previdenciárias ou para com o FGTS ou a não manutenção das condições de habilitação, ensejará a aplicação de sanções administrativas, previstas no instrumento convocatório e na legislação vigente, podendo culminar em rescisão contratual, por ato unilateral e escrito da CONTRATANTE, conforme disposto nos arts. 77 e 80 da Lei nº 8.666, de 1993.

17.31. Caso não seja apresentada a documentação comprobatória do cumprimento das obrigações trabalhistas, previdenciárias e para com o FGTS, a CONTRATANTE comunicará o fato à CONTRATADA e reterá o pagamento da fatura mensal, em valor proporcional ao inadimplemento, até que a situação seja regularizada.

17.31.1. Não havendo quitação das obrigações por parte da CONTRATADA no prazo de quinze dias, a CONTRATANTE poderá efetuar o pagamento das obrigações diretamente aos empregados da contratada que tenham participado da execução dos serviços objeto do contrato.

17.31.2. O sindicato representante da categoria do trabalhador deverá ser notificado pela CONTRATANTE para acompanhar o pagamento das verbas mencionadas.

17.31.3. Tais pagamentos não configuram vínculo empregatício ou implicam a assunção de responsabilidade por quaisquer obrigações dele decorrentes entre a contratante e os empregados da contratada.

17.32. O contrato só será considerado integralmente cumprido após a comprovação, pela CONTRATADA, do pagamento de todas as obrigações trabalhistas, sociais e previdenciárias e para com o FGTS referentes à mão de obra alocada em sua execução, inclusive quanto às verbas rescisórias.

17.33. A fiscalização de que trata este tópico não exclui nem reduz a responsabilidade da CONTRATADA, inclusive perante terceiros, por qualquer irregularidade, ainda que resultante de imperfeições técnicas, vícios redibitórios, ou emprego de material inadequado ou de qualidade inferior e, na ocorrência desta, não implica corresponsabilidade da CONTRATANTE ou de seus agentes, gestores e fiscais, de conformidade com o art. 70 da Lei nº 8.666, de 1993.

17.34. As disposições previstas neste Termo de Referência não excluem o disposto no Anexo VIII da Instrução Normativa SLTI/MP nº 05, de 2017, aplicável no que for pertinente à contratação.

17.35. A fiscalização de que trata este Termo de Referência não exclui nem reduz a responsabilidade da CONTRATADA, inclusive perante terceiros, por qualquer irregularidade, ainda que resultante de imperfeições técnicas, vícios redibitórios, ou emprego de material inadequado ou de qualidade inferior e, na ocorrência desta, não implica corresponsabilidade da CONTRATANTE ou de seus agentes, gestores e fiscais, de conformidade com o art. 70 da Lei nº 8.666, de 1993.

## 18. DO RECEBIMENTO E ACEITAÇÃO DO OBJETO

18.1. O recebimento do objeto do fornecimento será dado de acordo com os Art. 73 a 76 da Lei 8.666/93, conforme:

18.1.1. Provisoriamente, na apresentação dos produtos definidos na Ordem de Serviço, acompanhados da devida Nota Fiscal, devendo neste momento ser realizada conferência inicial pelos Fiscais Técnicos e Gestores do Contrato ou qualquer responsável citado pela CONTRATANTE e, se identificada a conformidade com a OS, o canhoto da Nota Fiscal é assinado e será considerado como Termo de Recebimento Provisório;

18.1.2. Definitivamente, após nova conferência e verificação por Gestor competente da CONTRATANTE e, se confirmada a conformidade com as exigências da Ordem de Serviço, a Nota Fiscal será atestada e será considerada como Termo de Recebimento Definitivo.

18.2. A emissão da Nota Fiscal/Fatura deve ser precedida do recebimento definitivo dos serviços, nos termos abaixo.

18.3. No prazo de até 5 dias corridos do adimplemento da parcela, a CONTRATADA deverá entregar toda a documentação comprobatória do cumprimento da obrigação contratual;

18.4. O recebimento provisório será realizado pelo fiscal técnico e setorial ou pela equipe de fiscalização após a entrega da documentação acima, da seguinte forma:

18.4.1. A CONTRATANTE realizará inspeção minuciosa de todos os serviços executados, por meio de profissionais técnicos competentes, acompanhados dos profissionais encarregados pelo serviço, com a finalidade de verificar a adequação dos serviços e constatar e relacionar os arremates, retoques e revisões finais que se fizerem necessários.

18.4.1.1. Para efeito de recebimento provisório, ao final de cada período de faturamento, o fiscal técnico do contrato irá apurar o resultado das avaliações da execução do objeto e, se for o caso, a análise do desempenho e qualidade da prestação dos serviços realizados em consonância com os indicadores previstos, que poderão resultar no redimensionamento de valores a serem pagos à CONTRATADA, registrando em relatório a ser encaminhado ao gestor do contrato.

18.4.1.2. A CONTRATADA fica obrigada a reparar, corrigir, remover, reconstruir ou substituir, às suas expensas, no todo ou em parte, o objeto em que se verificarem vícios, defeitos ou incorreções resultantes da execução ou materiais empregados, cabendo à fiscalização não atestar a última e/ou única medição de serviços até que sejam sanadas todas as eventuais pendências que possam vir a ser apontadas no Recebimento Provisório.

18.4.1.3. O recebimento provisório também ficará sujeito, quando cabível, à conclusão de todos os testes de campo e à entrega dos Manuais e Instruções exigíveis.

18.4.1.4. Da mesma forma, ao final de cada período de faturamento mensal, o fiscal administrativo deverá verificar as rotinas previstas no Anexo VIII-B da IN SEGES/MP nº 5/2017, no que forem aplicáveis à presente contratação, emitindo relatório que será encaminhado ao gestor do contrato;

18.4.2. No prazo de até 10 dias corridos a partir do recebimento dos documentos da CONTRATADA, cada fiscal ou a equipe de fiscalização deverá elaborar Relatório Circunstanciado em consonância com suas atribuições, e encaminhá-lo ao gestor do contrato.

18.4.2.1. quando a fiscalização for exercida por um único servidor, o relatório circunstanciado deverá conter o registro, a análise e a conclusão acerca das ocorrências na execução do contrato, em relação à fiscalização técnica e administrativa e demais documentos que julgar necessários, devendo encaminhá-los ao gestor do contrato para recebimento definitivo.

18.4.2.2. Será considerado como ocorrido o recebimento provisório com a entrega do relatório circunstanciado ou, em havendo mais de um a ser feito, com a entrega do último.

19.4.2.2.1. Na hipótese de a verificação a que se refere o parágrafo anterior não ser procedida tempestivamente, reputar-se-á como realizada, consumando-se o recebimento provisório no dia do esgotamento do prazo.

18.5. No prazo de até 10 (dez) dias corridos a partir do recebimento provisório dos serviços, o Gestor do Contrato deverá providenciar o recebimento definitivo, ato que concretiza o ateste da execução dos serviços, obedecendo as seguintes diretrizes:

18.6. Realizar a análise dos relatórios e de toda a documentação apresentada pela fiscalização e, caso haja irregularidades que impeçam a liquidação e o pagamento da despesa, indicar as cláusulas contratuais pertinentes, solicitando à CONTRATADA, por escrito, as respectivas correções.

18.7. Emitir Termo Circunstanciado para efeito de recebimento definitivo dos serviços prestados, com base nos relatórios e documentações apresentadas;

18.8. O recebimento provisório ou definitivo do objeto não exclui a responsabilidade da CONTRATADA pelos prejuízos resultantes da incorreta execução do contrato, ou, em qualquer época, das garantias concedidas e das responsabilidades assumidas em contrato e por força das disposições legais em vigor.

18.9. Os serviços poderão ser rejeitados, no todo ou em parte, quando em desacordo com as especificações constantes neste Termo de Referência e na proposta, devendo ser corrigidos/refeitos/substituídos no prazo fixado pelo fiscal do contrato, às custas da CONTRATADA, sem prejuízo da aplicação de penalidades.

**19. DO PAGAMENTO**

19.1. O pagamento será efetuado pela CONTRATANTE no prazo de 14 dias, contados do recebimento da Nota Fiscal/Fatura.

19.1.1. Os pagamentos decorrentes de despesas cujos valores não ultrapassem o limite de que trata o inciso II do art. 24 da Lei 8.666, de 1993, deverão ser efetuados no prazo de até 5 (cinco) dias úteis, contados da data da apresentação da Nota Fiscal/Fatura, nos termos do art. 5º, § 3º, da Lei nº 8.666, de 1993.

19.2. A emissão da Nota Fiscal/Fatura será precedida do recebimento definitivo do serviço, conforme este Termo de Referência.

19.3. A Nota Fiscal ou Fatura deverá ser obrigatoriamente acompanhada da comprovação da regularidade fiscal, constatada por meio de consulta online ao SICAF ou, na impossibilidade de acesso ao referido Sistema, mediante consulta aos sítios eletrônicos oficiais ou à documentação mencionada no art. 29 da Lei nº 8.666, de 1993.

19.3.1. Constatando-se, junto ao SICAF, a situação de irregularidade do fornecedor contratado, deverão ser tomadas as providências previstas no do art. 31 da Instrução Normativa nº 3, de 26 de abril de 2018.

19.4. O setor competente para proceder o pagamento deve verificar se a Nota Fiscal ou Fatura apresentada expressa os elementos necessários e essenciais do documento, tais como:

19.4.1. o prazo de validade;

19.4.2. a data da emissão;

19.4.3. os dados do contrato e do órgão CONTRATANTE;

19.4.4. o período de prestação dos serviços;

19.4.5. o valor a pagar; e

19.4.6. eventual destaque do valor de retenções tributárias cabíveis.

19.5. Havendo erro na apresentação da Nota Fiscal/Fatura, ou circunstância que impeça a liquidação da despesa, o pagamento ficará sobreposto até que a CONTRATADA providencie as medidas saneadoras. Nesta hipótese, o prazo para pagamento iniciar-se-á após a comprovação da regularização da situação, não acarretando qualquer ônus para a CONTRATANTE;

19.6. Nos termos do item 1, do Anexo VIII-A da Instrução Normativa SEGES/MP nº 05, de 2017, será efetuada a retenção ou glosa no pagamento, proporcional à irregularidade verificada, sem prejuízo das sanções cabíveis, caso se constate que a CONTRATADA:

19.6.1. não produziu os resultados acordados;

19.6.2. deixou de executar as atividades contratadas, ou não as executou com a qualidade mínima exigida;

19.6.3. deixou de utilizar os materiais e recursos humanos exigidos para a execução do serviço, ou utilizou-os com qualidade ou quantidade inferior à demandada.

19.7. Será considerada data do pagamento o dia em que constar como emitida a ordem bancária para pagamento.

19.8. Antes de cada pagamento à CONTRATADA, será realizada consulta ao SICAF para verificar a manutenção das condições de habilitação exigidas no edital.

19.9. Constatando-se, junto ao SICAF, a situação de irregularidade da CONTRATADA, será providenciada sua notificação, por escrito, para que, no prazo de 5 (cinco) dias úteis, regularize sua situação ou, no mesmo prazo, apresente sua defesa. O prazo poderá ser prorrogado uma vez, por igual período, a critério da CONTRATANTE.

19.10. Previamente à emissão de nota de empenho e a cada pagamento, a Administração deverá realizar consulta ao SICAF para identificar possível suspensão temporária de participação em licitação, no âmbito do órgão ou entidade, proibição de contratar com o Poder Público, bem como ocorrências impeditivas indiretas, observado o disposto no art. 29, da Instrução Normativa nº 3, de 26 de abril de 2018.

19.11. Não havendo regularização ou sendo a defesa considerada improcedente, a CONTRATANTE deverá comunicar aos órgãos responsáveis pela fiscalização da regularidade fiscal quanto à inadimplência da CONTRATADA, bem como quanto à existência de pagamento a ser efetuado, para que sejam adionados os meios pertinentes e necessários para garantir o recebimento de seus créditos.

19.12. Persistindo a irregularidade, a CONTRATANTE deverá adotar as medidas necessárias à rescisão contratual nos autos do processo administrativo correspondente, assegurada à CONTRATADA a ampla defesa.

19.13. Havendo a efetiva execução do objeto, os pagamentos serão realizados normalmente, até que se decida pela rescisão do contrato, caso a CONTRATADA não regularize sua situação junto ao SICAF.

19.13.1. Será rescindido o contrato em execução com a CONTRATADA inadimplente no SICAF, salvo por motivo de economicidade, segurança nacional ou outro de interesse público de alta relevância, devidamente justificado, em qualquer caso, pela máxima autoridade da CONTRATANTE.

19.14. Quando do pagamento, será efetuada a retenção tributária prevista na legislação aplicável, em especial a prevista no artigo 31 da Lei 8.212, de 1993, nos termos do item 6 do Anexo XI da IN SEGES/MP n. 5/2017, quando couber.

19.15. É vedado o pagamento, a qualquer título, por serviços prestados, à empresa privada que tenha em seu quadro societário servidor público da ativa do órgão CONTRATANTE, com fundamento na Lei de Diretrizes Orçamentárias vigente.

19.16. A parcela mensal a ser paga a título de aviso prévio trabalhado e indenizado corresponderá, no primeiro ano de contratação, ao percentual originalmente fixado na planilha de preços.

19.16.1. Não tendo havido a incidência de custos com aviso prévio trabalhado e indenizado, a prorrogação contratual seguinte deverá prever o pagamento do percentual máximo equivalente a 03 (três) dias a mais por ano de serviço, até o limite compatível com o prazo total de vigência contratual.

19.16.2. A adequação de pagamento de que trata o subitem anterior deverá ser prevista em termo aditivo.

19.16.3. Caso tenha ocorrido a incidência parcial ou total dos custos com aviso prévio trabalhado e/ou indenizado no primeiro ano de contratação, tais rubricas deverão ser mantidas na planilha de forma complementar/proportional, devendo o órgão contratante esclarecer a metodologia de cálculo adotada.

19.17. A Contratante providenciará o desconto na fatura a ser paga do valor global pago a título de vale-transporte em relação aos empregados da Contratada que expressamente optaram por não receber o benefício previsto na Lei nº 7.418, de 16 de dezembro de 1985, regulamentado pelo Decreto nº 95.247, de 17 de novembro de 1987.

19.18. Nos casos de eventuais atrasos de pagamento, desde que a CONTRATADA não tenha concorrido, de alguma forma, para tanto, fica convencionado que a taxa de compensação financeira devida pela CONTRATANTE, entre a data do vencimento e o efetivo adimplemento da parcela é calculada mediante a aplicação da seguinte fórmula:

$$EM = I \times N \times VP, \text{ sendo:}$$

EM = Encargos moratórios;

N = Número de dias entre a data prevista para o pagamento e a do efetivo pagamento;

VP = Valor da parcela a ser paga.

I = Índice de compensação financeira = 0,00016438, assim apurado:

I = (TX)	I =	$\frac{(.6 / 100)}{365}$	I = 0,00016438
			TX = Percentual da taxa anual = 6%

19.19. Provisoriamente, na apresentação dos produtos definidos na Ordem de Serviço, acompanhados da devida Nota Fiscal, devendo neste momento ser realizada conferência inicial pelos Fiscais Técnicos e Gestores do Contrato ou qualquer responsável citado pela CONTRATANTE e, se identificada a conformidade com a OS, o canhoto da Nota Fiscal é assinado e será considerado como Termo de Recebimento Provisório;

19.20. Definitivamente, após nova conferência e verificação por Gestor competente da CONTRATANTE e, se confirmada a conformidade com as exigências da Ordem de Serviço, a Nota Fiscal será atestada e será considerada como Termo de Recebimento Definitivo.

19.21. A CONTRATANTE rejeitará, no todo ou em parte, os fornecimentos executados em desacordo com o disposto neste Termo de Referência. Se, após o recebimento provisório, constatar-se que os fornecimentos foram executados em desacordo com o especificado, com defeito ou incompleto, os responsáveis da CONTRATANTE notificarão a empresa fornecedora para que a mesma providencie a correção necessária dentro dos prazos de recebimento.

19.22. Independentemente da aceitação, a CONTRATADA deverá garantir a qualidade do serviço e produtos fornecidos pelo prazo estabelecido nas especificações e nas condições de Garantia, obrigando-se a corrigir aquele que apresentar erro ou defeito, no prazo estabelecido pela CONTRATANTE;

#### 19.23. GRUPO 1 - Desenvolvimento de Sistemas:

19.23.1. No âmbito de cada Ordem de Serviço, será aferido o Índice de Evolução do Sistema - IES no caso de OS de PROJETO, ou o Índice de Manutenção de Sistema – IMS no caso de OS de MANUTENÇÃO, em períodos de 1 (um) a 2 (dois) meses, em momento escolhido pela CONTRATANTE e ao término da OS.

19.23.2. A aferição do IES e do IMS corresponderá ao recebimento definitivo do serviço e será feita por servidor ou comissão designada pela autoridade competente e formalizada.

19.23.3. A aferição do IES e do IMS será formalizada por meio de termo que seguirá o modelo constante no Anexo X deste Edital com as adaptações que forem necessárias.

19.23.4. Após a aferição do IES ou do IMS será remunerado o valor devido pelo serviço prestado no âmbito da OS durante o período coberto por esta aferição, considerando a efetiva ocupação dos postos de trabalho e os níveis mínimos de serviço definidos no Anexo VI.

19.23.5. Durante a execução da OS, quando for necessário aplicar desconto em virtude de não-ocupação de posto de trabalho, considerar-se-á inicialmente a relação entre a quantidade de DIAS-DESENVOLVEDOR esperados e a quantidade de DIAS-DESENVOLVEDOR efetivamente realizados no período.

19.23.6. Considera-se por DIA-DESENVOLVEDOR um dia de prestação de serviço de um posto de trabalho.

19.23.7. A quantidade de DIAS-DESENVOLVEDOR executados no período por perfil será a quantidade de DIAS-DESENVOLVEDOR nos quais os postos de trabalho do perfil foram efetivamente ocupados.

19.23.8. A quantidade de DIAS-DESENVOLVEDOR esperados no período por perfil será o produto da quantidade de postos de trabalho do perfil definida na OS pela quantidade de dias úteis no período.

19.23.9. O percentual de desconto a ser aplicado por não-ocupação de posto de trabalho no período por perfil será obtido segundo a seguinte fórmula:

19.23.10. Percentual de desconto por perfil = (quantidade de DIAS-DESENVOLVEDOR esperados do perfil - quantidade de DIAS-DESENVOLVEDOR executados do perfil) / (quantidade de DIAS-DESENVOLVEDOR esperados do perfil).

19.23.11. Este percentual de desconto por não-ocupação de posto de trabalho no período será aplicado sem prejuízo de outros descontos decorrentes de eventual não cumprimento dos níveis mínimos de serviços definidos no Anexo VI.

19.23.12. A CONTRATANTE realizará o pagamento em até 10 (dez) dias úteis após a emissão do termo de recebimento definitivo do serviço, observadas as disposições do Pregão Eletrônico n.º 046/2018.

19.23.13. A CONTRATANTE, observados os princípios do contraditório e da ampla defesa, poderá deduzir, cautelar ou definitivamente, do montante a pagar à CONTRATADA, os valores correspondentes a multas, resarcimentos ou indenizações devidas pela CONTRATADA, nos termos deste Contrato.

19.23.14. No caso de atraso de pagamento, desde que a CONTRATADA não tenha concorrido de alguma forma para tanto, serão devidos pela CONTRATANTE encargos moratórios à taxa nominal de 6% a.a. (seis por cento ao ano), capitalizados diariamente em regime de juros simples.

19.23.15. O valor dos encargos será calculado pela fórmula: EM = I x N x VP, onde: EM = Encargos moratórios devidos; N = Números de dias entre a data prevista para o pagamento e a do efetivo pagamento; I = Índice de compensação financeira = 0,00016438; e VP = Valor da prestação em atraso.

#### 19.24. GRUPO 2 - Mensuração de Sistemas

19.24.1. Para os serviços dimensionados em Ponto por Função (PF), o valor a ser pago será calculado a partir da quantidade de Pontos por Função executada, multiplicado pelo valor do Ponto de Função contratado e deduzidos os descontos referentes às penalidades ocorridas. A fórmula para o cálculo do valor do serviço a ser faturado, é representada a seguir:

$$VF = (QPF * VPF) - VPA$$

Onde:

VF: Valor Faturado;

QPF: Quantidade de Pontos de Função (PF);

VPF: Valor atribuído ao Ponto de Função (PF);

VPA: Valor das Penalidades Aplicadas.

19.25. Com base nas Ordens de Serviço (OS) emitidas pela CONTRATANTE e executada pela CONTRATADA, será calculado o pagamento a partir do resultado apurado de todas as ordens de serviço homologadas pelo requisitante do serviço no mês corrente, descontado o valor referente ao percentual de desconto definido nos ANS, quando for o caso;

19.26. A apuração do resultado das Ordens de Serviço será realizada com base nos itens de serviço solicitados, bem como, na quantidade de UST demandadas, nos prazos acordados e nos ANS calculados;

19.27. As Ordens de Serviço somente serão aceitas e liberadas para pagamento quando todas as condições a seguir forem verdadeiras:

- 19.27.1. Toda a documentação obrigatória (descritos por fases na MDS) tiver sido entregue e aceita;
- 19.27.2. O repositório do software utilizado pela CONTRATANTE para o controle de versões tiver sido atualizado com a versão final da documentação obrigatória;
- 19.27.3. A qualidade do serviço tiver sido avaliada e aceita pela CONTRATANTE;
- 19.27.4. O Relatório de Serviços Executados tiver sido aceito pela CONTRATANTE;
- 19.27.5. O serviço tiver sido homologado e aceito pelo Requisitante do Serviço;
- 19.27.6. Os indicadores de ANS tiverem sido aferidos.
- 19.27.7. O pagamento deverá ser efetivado pela CONTRATANTE em até 14 (quatorze) dias corridos contados a partir da certificação do Gestor/Fiscal do contrato, em moeda corrente, mediante emissão de ordem bancária para crédito em conta da licitante vencedora, conforme disposto no artigo 40 Inciso XIV alínea "a" da Lei nº 8.666/93, após apresentação da respectiva Nota Fiscal/Fatura com a discriminação dos serviços, devidamente atestada pelo fiscal do contrato e comprovada a regularidade da empresa, mediante consulta "on-line" ao SICAF;
- 19.27.8. Caso haja divergência na fatura, a CONTRATANTE deverá devolvê-la para que a CONTRATADA possa refazê-la, contando-se novo prazo para pagamento a partir da sua reapresentação;
- 19.27.9. A Nota Fiscal ou Fatura deverá ser obrigatoriamente acompanhada das seguintes comprovações:
- 19.27.10. Do pagamento da remuneração e das contribuições sociais (Fundo de Garantia do Tempo de Serviço e Previdência Social), correspondentes ao mês da última nota fiscal ou fatura vencida, compatível com os empregados vinculados à execução contratual, nominalmente identificados, na forma do § 4º do Art. 31 da Lei nº 9.032, de 28 de abril de 1995, quando se tratar de mão-de-obra diretamente envolvida na execução dos serviços na contratação de serviços continuados;
- 19.27.11. Da regularidade fiscal, constatada através de consulta "on-line" ao Sistema de Cadastramento Unificado de Fornecedores (SICAF), ou na impossibilidade de acesso ao referido Sistema, mediante consulta aos sítios eletrônicos oficiais ou à documentação mencionada no art. 29 da Lei 8.666/93;
- 19.27.12. Do cumprimento das obrigações trabalhistas, correspondentes à última nota fiscal ou fatura que tenha sido paga pela Administração.
- 19.27.13. O descumprimento das obrigações trabalhistas, previdenciárias e as relativas ao FGTS ensejará o pagamento em juízo dos valores em débito, sem prejuízo das sanções cabíveis.
- 19.27.14. Os serviços deverão ser entregues pela CONTRATADA no endereço especificado na Solicitação de Serviço ou Ordem de Serviço, em conformidade com os prazos de entrega discriminados nesses documentos;
- 19.27.15. O pagamento será efetuado com a apresentação da(s) respectiva(s) Nota(s) Fiscal(is), tendo sido cumpridos, no que couber, todos os critérios estabelecidos neste Termo de Referência, devidamente atestada;
- 19.27.16. A CONTRATANTE efetuará o pagamento somente para a empresa CONTRATADA, vedada a negociação dos documentos de cobrança com terceiros, ou a sua colocação em cobrança bancária.

## 20. REAJUSTE

### GRUPO 1 - Desenvolvimento de Sistemas

- 20.1. É admitida a repactuação dos preços deste contrato, desde que seja observado o interregno mínimo de 12 (doze) meses, contado da data limite para apresentação da proposta de preços pela licitante.
- 20.2. Nas repactuações subsequentes à primeira, o interregno de 12 (doze) meses será contado a partir da data de início dos efeitos financeiros da última repactuação ocorrida.
- 20.3. Caso a CONTRATADA não requeira tempestivamente a repactuação e prorrogue o contrato sem pleiteá-la, ocorrerá a preclusão do direito.
- 20.4. Ocorrerá igualmente a preclusão do direito à repactuação caso o pedido seja formulado depois de extinto o contrato.
- 20.5. As repactuações serão precedidas de solicitação da CONTRATADA, acompanhada de demonstração analítica da alteração dos custos, por meio de apresentação das planilhas de composição de custos e formação de preços, e, se for o caso, dos documentos indispensáveis à comprovação da alteração dos preços de mercado de cada um dos itens da planilha a serem alterados.
- 20.6. O percentual final da repactuação não poderá ultrapassar o percentual limite de crescimento da despesa pública para o exercício, fixado nos termos do novo regime fiscal instituído pela Emenda Constitucional nº 95, de 15/12/2016.

### GRUPO 2 - Mensuração de Sistemas

- 20.7. Os preços dos serviços serão fixos e irreajustáveis para o período de 12 (doze) meses, após o qual poderão ser atualizados, anualmente, desde que haja previsão de reajuste de preços por aplicação de índice de correção monetária, é obrigatória a adoção do Índice de Custos de Tecnologia da Informação - ICTI, mantido pela Fundação Instituto de Pesquisa Econômica Aplicada - IPEA.
- 20.8. Nos reajustes subsequentes ao primeiro, o interregno mínimo de um ano será contado a partir dos efeitos financeiros do último reajuste.
- 20.9. No caso de atraso ou não divulgação do índice de Custos de Tecnologia da Informação - ICTI, a CONTRATANTE pagará à CONTRATADA a importância calculada pela última variação conhecida, liquidando a diferença correspondente tão logo seja divulgado o índice definitivo. Fica a CONTRATADA obrigada a apresentar memória de cálculo referente ao reajustamento de preços do valor remanescente, sempre que este ocorrer.
- 20.10. Nas aferições finais, o índice utilizado para reajuste será, obrigatoriamente, o definitivo.
- 20.11. Caso o índice estabelecido para reajustamento venha a ser extinto ou de qualquer forma não possa mais ser utilizado, será adotado, em substituição, o que vier a ser determinado pela legislação então em vigor.
- 20.12. Na ausência de previsão legal quanto ao índice substituto, as partes elegerão novo índice oficial, para reajustamento do preço do valor remanescente, por meio de termo aditivo.
- 20.13. O reajuste será realizado por apostilamento.

## 21. AUMENTO E SUSPENSÃO DOS QUANTITATIVOS

- 21.1. No interesse da Administração, o objeto deste TERMO DE REFERÊNCIA poderá ser suprimido ou aumentado até o limite de 25% (vinte e cinco por cento) do valor inicial atualizado do objeto, facultada a supressão além desse limite, por acordo entre as partes, conforme disposto no art. 65, § 1º e 2º, inciso II, da Lei nº 8.666/93.
22. CANCELAMENTO DOS SERVIÇOS

- 22.1. Nos casos em que a Ordem de Serviço for cancelada por solicitação da CONTRATANTE, o trabalho já executado deverá ser avaliado e pago;
- 22.2. Quando do cancelamento do serviço, a CONTRATADA deverá entregar os produtos do serviço executado, imediatamente, mesmo que inacabados;
- 22.3. O pagamento dos serviços cancelados está vinculado à entrega dos produtos parciais elaborados pela CONTRATADA até o momento do cancelamento;
- 22.4. É vedado à CONTRATADA complementar os produtos do serviço interrompido após o cancelamento da OS.

**23. GARANTIA DA EXECUÇÃO**

23.1. O adjudicatário prestará garantia de execução do contrato, nos moldes do art. 56 da Lei nº 8.666, de 1993, com validade durante a execução do contrato e por 90 (noventa) dias após o término da vigência contratual, em valor correspondente a 5% (cinco por cento) do valor total do contrato.

23.2. No prazo máximo de 10 (dez) dias úteis, prorrogáveis por igual período, a critério do CONTRATANTE, contados da assinatura do contrato, a CONTRATADA deverá apresentar comprovante de prestação de garantia, podendo optar por caução em dinheiro ou títulos da dívida pública, seguro-garantia ou fiança bancária.

23.2.1. A inobservância do prazo fixado para apresentação da garantia acarretará a aplicação de multa de 0,07% (sete centésimos por cento) do valor total do contrato por dia de atraso, até o máximo de 2% (dois por cento).

23.2.2. O atraso superior a 25 (vinte e cinco) dias autoriza a Administração a promover a rescisão do contrato por descumprimento ou cumprimento irregular de suas cláusulas, conforme dispõem os incisos I e II do art. 78 da Lei n. 8.666 de 1993.

23.3. A validade da garantia, qualquer que seja a modalidade escolhida, deverá abranger um período de 90 dias após o término da vigência contratual, conforme item 3.1 do Anexo VII-F da IN SEGES/MP nº 5/2017.

23.4. A garantia assegurará, qualquer que seja a modalidade escolhida, o pagamento de:

23.4.1. prejuízos advindos do não cumprimento do objeto do contrato e do não adimplemento das demais obrigações nele previstas;

23.4.2. prejuízos diretos causados à Administração decorrentes de culpa ou dolo durante a execução do contrato;

23.4.3. multas moratórias e punitivas aplicadas pela Administração à CONTRATADA; e

23.4.4. obrigações trabalhistas e previdenciárias de qualquer natureza e para com o FGTS, não adimplidas pela CONTRATADA, quando couber.

23.5. A modalidade seguro-garantia somente será aceita se contemplar todos os eventos indicados no item anterior, observada a legislação que rege a matéria.

23.6. A garantia em dinheiro deverá ser efetuada em favor da CONTRATANTE, em conta específica na Caixa Econômica Federal, com correção monetária. Caso a opção seja por utilizar títulos da dívida pública, estes devem ter sido emitidos sob a forma escritural, mediante registro em sistema centralizado de liquidação e de custódia autorizado pelo Banco Central do Brasil, e avaliados pelos seus valores econômicos, conforme definido pelo Ministério da Fazenda.

23.7. No caso de garantia na modalidade de fiança bancária, deverá constar expressa renúncia do fiador aos benefícios do artigo 827 do Código Civil.

23.8. No caso de alteração do valor do contrato, ou prorrogação de sua vigência, a garantia deverá ser ajustada à nova situação ou renovada, seguindo os mesmos parâmetros utilizados quando da contratação.

23.9. Se o valor da garantia for utilizado total ou parcialmente em pagamento de qualquer obrigação, a CONTRATADA obriga-se a fazer a respectiva reposição no prazo máximo de 30 (trinta) dias úteis, contados da data em que for notificada.

23.10. A CONTRATANTE executará a garantia na forma prevista na legislação que rege a matéria.

23.11. Será considerada extinta a garantia:

23.11.1. com a devolução da apólice, carta fiança ou autorização para o levantamento de importâncias depositadas em dinheiro a título de garantia, acompanhada de declaração da CONTRATANTE, mediante termo circunstanciado, de que a CONTRATADA cumpriu todas as cláusulas do contrato;

23.11.2. no prazo de 90 (noventa) dias após o término da vigência do contrato, caso a Administração não comunique a ocorrência de sinistros, quando o prazo será ampliado, nos termos da comunicação, conforme estabelecido na alínea "h2" do item 3.1 do Anexo VII-F da IN SEGES/MP n. 05/2017.

23.12. O garantidor não é parte para figurar em processo administrativo instaurado pela CONTRATANTE com o objetivo de apurar prejuízos e/ou aplicar sanções à CONTRATADA.

23.13. A CONTRATADA autoriza a CONTRATANTE a reter, a qualquer tempo, a garantia, na forma prevista no neste TR.

23.14. A garantia da contratação somente será liberada ante a comprovação de que a empresa pagou todas as verbas rescisórias decorrentes da contratação, e que, caso esse pagamento não ocorra até o fim do segundo mês após o encerramento da vigência contratual, a garantia será utilizada para o pagamento dessas verbas trabalhistas, incluindo suas repercussões previdenciárias e relativas ao FGTS, conforme estabelecido no art. 8º, VI do Decreto nº 9.507, de 2018, observada a legislação que rege a matéria.

23.14.1. Também poderá haver liberação da garantia se a empresa comprovar que os empregados serão realocados em outra atividade de prestação de serviços, sem que ocorra a interrupção do contrato de trabalho.

23.15. Por ocasião do encerramento da prestação dos serviços contratados, a Administração Contratante poderá utilizar o valor da garantia prestada para o pagamento direto aos trabalhadores vinculados ao contrato no caso da não comprovação: (1) do pagamento das respectivas verbas rescisórias ou (2) da realocação dos trabalhadores em outra atividade de prestação de serviços, nos termos da alínea "j" do item 3.1 do Anexo VII-F da IN SEGES/MP n. 5/2017.

**24. GARANTIA DOS SERVIÇOS**

24.1. A empresa fornecedora deverá prestar garantia dos serviços prestados pelo período mínimo de 180 (cento e oitenta) dias, assegurando o pleno funcionamento dentro do ambiente operacional da CONTRATANTE, a serem prestados nas instalações do órgão, sem qualquer ônus para a CONTRATANTE. O prazo de garantia terá início após o atesto e emissão do Termo de Recebimento Definitivo;

24.2. Dentro do período de garantia, a correção de erros nos serviços entregues pela CONTRATADA deverá ser efetuada sem qualquer ônus para o CONTRATANTE, seja financeiro ou de atraso na prestação de outro(s) serviço(s), desde que, comprovadamente, não tenham se dado em razão das especificações feitas pelo CONTRATANTE ou por terceiros contratados para esta finalidade.

**25. DAS SANÇÕES ADMINISTRATIVAS**

25.1. Comete infração administrativa nos termos da Lei nº 10.520, de 2002, a CONTRATADA que:

25.1.1. inexecutar total ou parcialmente qualquer das obrigações assumidas em decorrência da contratação;

25.1.2. ensejar o retardamento da execução do objeto;

25.1.3. falhar ou fraudar na execução do contrato;

25.1.4. comportar-se de modo inidôneo; ou

25.1.5. cometer fraude fiscal.

25.2. Pela inexecução total ou parcial do objeto deste contrato, a Administração pode aplicar à CONTRATADA as seguintes sanções:

25.2.1. **Advertência por escrito**, quando do não cumprimento de quaisquer das obrigações contratuais consideradas faltas leves, assim entendidas aquelas que não acarretam prejuízos significativos para o serviço contratado;

25.2.2. **Multas**:

25.2.2.1. As sanções em função de descumprimento de prazos de execução serão definidas no instrumento de Acordos de Nível de Serviço - ANS - observados os níveis de qualidade e prazo de entrega e produtos a serem entregues.

- 25.2.2.2. Decorridos 30 (trinta) dias de atraso o CONTRATANTE poderá decidir pela continuidade da multa ou pela rescisão, em razão da inexecução total;
- 25.2.2.3. 0,06% (seis centésimos por cento) por dia sobre o valor global deste contrato, para ocorrências de atrasos em qualquer outro prazo previsto neste instrumento, não abrangido pelas demais alíneas;
- 25.2.2.4. 1% (um por cento) por dia sobre o valor da garantia contratual, pela não apresentação/actualização, no prazo estabelecido neste instrumento, da garantia de execução contratual;
- 25.2.2.5. 5% (cinco por cento) sobre o valor global atualizado do contrato, pela não manutenção das condições de habilitação e qualificação exigidas no instrumento convocatório;
- 25.2.2.6. 20% (vinte por cento) sobre o valor do contrato, nas hipóteses de rescisão contratual por inexecução do contrato, caracterizando-se quando houver reiterado descumprimento de obrigações contratuais, se a entrega for inferior a 50% (cinquenta por cento) do contratado, caso o atraso ultrapassar o prazo limite de trinta dias, estabelecido na alínea "a" ou os serviços forem prestados fora das especificações constantes do Termo de Referência e da proposta da CONTRATADA;
- 25.2.2.7. as penalidades de multa decorrentes de fatos diversos serão consideradas independentes entre si.
- 25.2.3. Suspensão de licitar e impedimento de contratar com o órgão, entidade ou unidade administrativa pela qual a Administração Pública opera e atua concretamente, pelo prazo de até dois anos;
- 25.2.4. Sanção de impedimento de licitar e contratar com órgãos e entidades da União, com o consequente descredenciamento no SICAF pelo prazo de até cinco anos
- 25.2.4.1. A Sanção de impedimento de licitar e contratar também é aplicável em quaisquer das hipóteses previstas como infração administrativa neste Termo de Referência.
- 25.2.4.2. Declaração de inidoneidade para licitar ou contratar com a Administração Pública, enquanto perdurarem os motivos determinantes da punição ou até que seja promovida a reabilitação perante a própria autoridade que aplicou a penalidade, que será concedida sempre que a CONTRATADA resarcir a CONTRATANTE pelos prejuízos causados;
- 25.3. As sanções previstas poderão ser aplicadas à CONTRATADA juntamente com as de multa, descontando-a dos pagamentos a serem efetuados.
- 25.4. Para efeito de aplicação de multas, às infrações são atribuídos graus, de acordo com as tabelas 1 e 2:

GRAU	CORRESPONDÊNCIA
1	0,2% ao dia sobre o valor mensal do contrato
2	0,4% ao dia sobre o valor mensal do contrato
3	0,8% ao dia sobre o valor mensal do contrato
4	1,6% ao dia sobre o valor mensal do contrato
5	3,2% ao dia sobre o valor mensal do contrato

Tabela

INFRAÇÃO		
ITEM	DESCRIÇÃO	GRAU
1	Permitir situação que crie a possibilidade de causar dano físico, lesão corporal ou consequências letais, por ocorrência;	05
2	Suspender ou interromper, salvo motivo de força maior ou caso fortuito, os serviços contratuais por dia e por unidade de atendimento;	04
3	Manter funcionário sem qualificação para executar os serviços contratados, por empregado e por dia;	03
4	Recusar-se a executar serviço determinado pela fiscalização, por serviço e por dia;	02
5	Retirar funcionários ou encarregados do serviço durante o expediente, sem a anuência prévia do CONTRATANTE, por empregado e por dia;	03
Para os itens a seguir, deixar de:		
6	Registrar e controlar, diariamente, a assiduidade e a pontualidade de seu pessoal, por funcionário e por dia;	01
7	Cumprir determinação formal ou instrução complementar do órgão fiscalizador, por ocorrência;	02
8	Substituir empregado que se conduza de modo inconveniente ou não atenda às necessidades do serviço, por funcionário e por dia;	01
9	Cumprir quaisquer dos itens do Edital e seus Anexos não previstos nesta tabela de multas, após reincidência formalmente notificada pelo órgão fiscalizador, por item e por ocorrência;	03
10	Indicar e manter durante a execução do contrato os prepostos previstos no edital/contrato;	01
11	Providenciar treinamento para seus funcionários conforme previsto na relação de obrigações da CONTRATADA.	01

Tabela

- 25.5. Também ficam sujeitas às penalidades do art. 87, III e IV da Lei nº 8.666, de 1993, as empresas ou profissionais que:
- 25.5.1. tenham sofrido condenação definitiva por praticar, por meio dolosos, fraude fiscal no recolhimento de quaisquer tributos;
- 25.5.2. tenham praticado atos ilícitos visando a frustrar os objetivos da licitação;
- 25.5.3. demonstrem não possuir idoneidade para contratar com a Administração em virtude de atos ilícitos praticados.
- 25.6. A aplicação de qualquer das penalidades previstas realizar-se-á em processo administrativo que assegurará o contraditório e a ampla defesa à CONTRATADA, observando-se o procedimento previsto na Lei nº 8.666, de 1993, e subsidiariamente a Lei nº 9.784, de 1999.
- 25.7. As multas devidas e/ou prejuízos causados à CONTRATANTE serão deduzidos dos valores a serem pagos, ou recolhidos em favor da União, ou deduzidos da garantia, ou ainda, quando for o caso, serão inscritos na Dívida Ativa da União e cobrados judicialmente.
- 25.7.1. Caso a CONTRATANTE determine, a multa deverá ser recolhida no prazo máximo de 15 (quinze) dias, a contar da data do recebimento da comunicação enviada pela autoridade competente.
- 25.8. A autoridade competente, na aplicação das sanções, levará em consideração a gravidade da conduta do infrator, o caráter educativo da pena, bem como o dano causado à Administração, observado o princípio da proporcionalidade.
- 25.9. Se, durante o processo de aplicação de penalidade, se houver indícios de prática de infração administrativa tipificada pela Lei nº 12.846, de 1º de agosto de 2013, como ato lesivo à administração pública nacional ou estrangeira, cópias do processo administrativo necessárias à apuração da responsabilidade

da empresa deverão ser remetidas à autoridade competente, com despacho fundamentado, para ciência e decisão sobre a eventual instauração de investigação preliminar ou Processo Administrativo de Responsabilização - PAR.

25.10. A apuração e o julgamento das demais infrações administrativas não consideradas como ato lesivo à Administração Pública nacional ou estrangeira nos termos da Lei nº 12.846, de 1º de agosto de 2013, seguirão seu rito normal na unidade administrativa.

25.11. O processamento do PAR não interfere no seguimento regular dos processos administrativos específicos para apuração da ocorrência de danos e prejuízos à Administração Pública Federal resultantes de ato lesivo cometido por pessoa jurídica, com ou sem a participação de agente público.

25.12. O cálculo da sanção administrativa a ser aplicada em razão do não cumprimento dos níveis de conformidade e prazos, por Ordem de Serviço, limita-se a 20% (vinte por cento), por Ordem de Serviço.

25.13. As penalidades serão obrigatoriamente registradas no SICAF.

## 26. CRITÉRIOS DE SELEÇÃO DO FORNECEDOR.

## 27. ESTIMATIVA DE PREÇOS E PREÇOS REFERENCIAIS.

## 28. DOS RECURSOS ORÇAMENTÁRIOS

28.1. O recurso orçamentário para atender a despesa desta aquisição está previsto no Orçamento Geral da CAPES, no **PTRES 136549, Fonte 8100 e Elemento de Despesa 44.90.40.01** para o Grupo 1 Desenvolvimento de Sistemas e **PTRES 136549, Fonte 8100 e Elemento de Despesa 33.90.40.21** para o Grupo 2 Mensuração de Sistema.

28.2. A contratação para exercício subsequente estará submetida à dotação orçamentária própria prevista para atendimento, a ser consignada pela CAPES na Lei Orçamentária da União.

## 29. DEFINIÇÕES E PROCEDIMENTOS DE SEGURANÇA

29.1. Os procedimentos básicos mínimos de segurança exigidos da CONTRATADA são:

29.1.1. Credenciar junto à CONTRATANTE, seus profissionais autorizados a retirar e a entregar documentos, bem como aqueles que venham a ser designados para prestar serviços nas dependências da CONTRATANTE;

29.1.2. Identificar qualquer equipamento da CONTRATADA que venha a ser instalado nas dependências da CONTRATANTE, utilizando placas de controle patrimonial, selos de segurança, etc.;

29.1.3. Fazer com que seus funcionários mantenham sigilo absoluto sobre informações, dados e documentos integrantes dos serviços a serem executados, inclusive com a assinatura de Termo de Responsabilidade e manutenção de sigilo próprio, semelhante ao disponível na CONTRATANTE;

29.1.4. A empresa CONTRATADA deverá seguir todas as determinações e orientações contidas nas Diretrizes e Normas de Segurança da CONTRATANTE;

29.1.5. Adotar critérios adequados para o processo seletivo dos profissionais, com o propósito de evitar a incorporação de pessoas com características e/ou antecedentes que possam comprometer a segurança ou credibilidade da CONTRATANTE;

29.1.6. Comunicar com antecedência mínima de 3 (três) dias à CONTRATANTE qualquer ocorrência de transferência, remanejamento ou demissão, para que seja providenciada a revogação de todos os privilégios de acesso aos sistemas, informações e recursos da CONTRATANTE, porventura colocados à disposição para realização dos serviços contratados;

29.1.7. Manter sigilo sobre todos os ativos de informações e de processos da CONTRATANTE e da CONTRATADA que se refiram à CONTRATANTE;

29.1.8. A CONTRATADA deverá manter sigilo absoluto sobre quaisquer dados, informações, códigos-fonte, artefatos, contidos em quaisquer documentos e em quaisquer mídias, de que venha a ter conhecimento durante a execução dos trabalhos, não podendo, sob qualquer pretexto divulgar, reproduzir ou utilizar, sob pena de lei, independentemente da classificação de sigilo conferida pela CONTRATANTE a tais documentos;

29.1.9. A CONTRATADA não poderá divulgar quaisquer informações a que tenha acesso em virtude dos trabalhos a serem executados ou de que tenha tomado conhecimento em decorrência da execução do objeto, sem autorização, por escrito, da CONTRATANTE, sob pena de aplicação das sanções cabíveis, além do pagamento de indenização por perdas e danos;

29.1.10. Cada profissional deverá assinar Termo de Ciência - Modelo de Termo de Ciência (0981161) comprometendo-se a não divulgar nenhum assunto tratado nas dependências da CONTRATANTE ou a serviço desta, salvo se expressamente autorizado.

## 30. CRITÉRIOS DE SUSTENTABILIDADE

30.1. Considerando o Decreto nº 7.746/12, que regulamentou o art. 3º, "caput", da Lei nº 8.666/93, a Lei nº 12.305/10, a IN MP nº 01/10, a Portaria MP nº 20/16, para a presente contratação, aplicar-se-ão os seguintes critérios de sustentabilidade ambiental:

30.2. Adotar processos administrativos na sua forma eletrônica. Os documentos deverão ser gerados e mantidos em sua forma digital e, com o objetivo de garantir a integridade dos mesmos, nestes poderão ser utilizados recursos tecnológicos de segurança da informação. O objetivo da referida adoção é reduzir o número de cópias e impressões em papel.

30.3. Os serviços prestados pela CONTRATADA deverão pautar-se sempre no uso racional de recursos e equipamentos, de forma a evitar e prevenir o desperdício de insumos e material consumidos, bem como a geração excessiva de resíduos, a fim de atender às diretrizes de responsabilidade ambiental adotadas pela CAPES.

30.4. A CONTRATADA deverá instruir os seus empregados quanto à necessidade de racionalização de recursos no desempenho de suas atribuições, bem como das diretrizes de responsabilidade ambiental adotadas pela CAPES.

30.5. Considerando as orientações contidas no "Guia Nacional de Licitações Sustentáveis" da AGU disponibilizado no link [http://www.agu.gov.br/page/content/detail/id\\_conteudo/400787](http://www.agu.gov.br/page/content/detail/id_conteudo/400787), destaca-se que não há obrigações adicionais que mereçam destaque.

## 31. DISPOSIÇÃO FINAL

31.1. A Administração não responderá por quaisquer compromissos assumidos pela CONTRATADA com terceiros, ainda que vinculados à execução do presente Termo e Contrato, bem como por qualquer dano causado a terceiros em decorrência de ato da CONTRATADA, de seus empregados, prepostos ou subordinados;

31.2. A CONTRATADA não poderá se valer do contrato a ser celebrado para assumir obrigações perante terceiros, dando-o como garantia, nem utilizar os direitos de crédito a serem auferidos em função das atividades prestadas em quaisquer operações de desconto bancário, sem prévia autorização da CONTRATANTE, sob pena de incorrer em quebra de cláusula contratual ensejando inclusive sua rescisão de pleno direito.

## 32. APROVAÇÃO E ENCaminhamento

32.1. Em cumprimento ao §6º, art. 12, da Instrução Normativa nº 1 /2019, a equipe de Planejamento da contratação e autoridade máxima da Área de TIC assinam o Termo de Referência.

Integrante Requisitante
Gustavo Jardim Portella
SIAPe nº 1644951

Integrante Técnico	Integrante Técnico	Integrante Técnico	Integrante Técnico	Integrante Administrativo
Edmilson Coelho Chaves Junior	Gabriel Costa Resende	Roosevelt Aguilar Silva	Gilene do Espírito S. Borges Gomes	Lucas Josijuan Abreu E
SIAPE nº 2692794	SIAPE nº 2013848	SIAPE nº 1666477	SIAPE nº 1670879	SIAPE nº 211895

32.2. Em acatamento ao mesmo §6º, art. 12, da Instrução Normativa nº 1 /2019, a autoridade competente aprova, no que se refere às competências exclusivas da DTI, devendo a mesma ter prosseguimento, todavia, somente após cumpridas todas as exigências legais referentes ao conteúdo administrativo e contratual da contratação, afetos à DGES/CAPES.

Sandro de Oliveira Araujo
Diretor de Tecnologia da Informação
SIAPE nº 1249926

Documentos Relacionados:	<p>Encarte Técnico Contratação - TI CGS (0920364) ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO</p> <p>Encarte Técnico Contratação - TI CGS (0920366) ENCARTE B - PERFIS E QUALIFICAÇÃO PROFISSIONAL</p> <p>Encarte Técnico Contratação - TI CGS (0983589) ENCARTE C - Metodologia de Desenvolvimento de Sistemas CGS</p> <p>Encarte Técnico Contratação - TI CGS (0952148) ENCARTE D - ORÇAMENTO ESTIMATIVO</p> <p>Encarte Técnico Contratação - TI CGS (0981395) ENCARTE E - MODELO DE PROPOSTA DE PREÇOS</p> <p>Encarte Técnico Contratação - TI CGS (0983593) ENCARTE F - MODELO DE TERMOS DE RECEBIMENTO ( TRP e TRD)</p> <p>Encarte Técnico Contratação - TI CGS (0987886) ENCARTE G - MODELO DE TERMOS DE CREDENCIAMENTO E DESCREDENCIAMENTO</p> <p>Encarte Técnico Contratação - TI CGS (0983069) ENCARTE H - Arquitetura de Referência - Guia Geral</p> <p>Encarte Técnico Contratação - TI CGS (0983071) ENCARTE I - Arquitetura de Referência - Guia Java</p> <p>Encarte Técnico Contratação - TI CGS (0983080) ENCARTE J - Arquitetura de Referência - Guia PHP</p> <p>Encarte Técnico Contratação - TI CGS (0983083) ENCARTE K - Arquitetura de Referência - Guia Python</p> <p>Encarte Técnico Contratação - TI CGS (0983089) ENCARTE L - Arquitetura de Referência - Guia Frontend</p> <p>Encarte Técnico Contratação - TI CGS (0982310) ENCARTE M - MODELO DE ORDEM DE SERVIÇO</p> <p>Encarte Técnico Contratação - TI CGS (0992017) ENCARTE N - MODELO DE DECLARAÇÃO DE CONTRATOS FIRMADOS COM A INICIATIVA PRIVADA E COM A ADMINISTRAÇÃO PÚBLICA</p> <p>Encarte Técnico Contratação - TI CGS (1006100) ENCARTE O - MODELO DE ATESTADO DE VISITA TÉCNICA</p> <p>Encarte Técnico Contratação - TI CGS (1031881) ENCARTE P - MODELO DO TERMO DE CONFIDENCIALIDADE E SIGILO</p> <p>Encarte Técnico Contratação - TI CGS (0981161) ENCARTE Q - Termo Modelo de Termo de Ciência</p>
--------------------------	---

Atenciosamente,



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO

##### 1. NÍVEIS DE SERVIÇO - GRUPO 1 - DESENVOLVIMENTO DE SISTEMAS

###### 1.1. Índice de Evolução do Sistema (IES)

1.1.1. Os níveis de serviço de OS do tipo PROJETO serão representados por indicador de desempenho denominado Índice de Evolução do Sistema (IES), que será a média ponderada de 3 componentes:

Componente do IES	Peso	Índice de Produtividade (IP)	Peso
Índice de Produtividade (IP)	0,45		
Índice de Qualidade (IQ)	0,40		
Avaliação do Product Onwer (APO)	0,15		

1.1.2. O Índice de Evolução do Sistema - IES será aferido em períodos de 1 a 2 meses, em momento escolhido pela CAPES, e obrigatoriamente ao término da Ordem de Serviço.

1.1.3. Adicionalmente, há níveis mínimos de serviço exigidos para alguns componentes do IES isoladamente, conforme item 1.1.36 – “Descontos a serem aplicados de acordo com o índice de cumprimento dos níveis de serviço”.

###### 1.1.4. Índice de Produtividade (IP)

1.1.4.1. O Índice de Produtividade (IP) é composto por dois itens: Item de Controle de Produtividade em Tamanho Funcional (ICPTF) e Item de Controle de Produtividade em Linhas de Código (ICPLC). Estes itens serão aferidos comparando a produtividade no período de aferição com a produtividade-base estabelecida neste Anexo do Edital.

1.1.4.2. A divisão do peso do IP entre seus dois componentes encontra-se no quadro a seguir:

Componente do IP	Peso
Item de Controle de Produtividade em Tamanho Funcional (ICPTF)	0,25
Item de Controle de Produtividade em Linhas de Código (ICPLC)	0,20
Peso Total do IP	0,45

1.1.4.3. A produtividade será estabelecida em termos de dias de efetiva ocupação de posto de trabalho, unidade esta que será identificada por DIA-DESENVOLVEDOR.

1.1.4.4. Por exemplo, se em um dado período de aferição tivermos 40 dias de efetiva ocupação de 6 postos de trabalho, haverá neste período 240 DIAS-DESENVOLVEDOR.

1.1.4.5. Não serão considerados aqueles dias em que não houver efetiva prestação do serviço, como finais de semana e feriados, em regra.

1.1.5. **Item de Controle de Produtividade em Tamanho Funcional (ICPTF)**

1.1.5.1. Para calcular o Item de Controle de Produtividade em Tamanho Funcional (ICPTF) deverá ser medido primeiro o Tamanho da Evolução Funcional (TEF), que é o quantitativo de pontos de função correspondente aos aprimoramentos realizados na versão atual do software em relação à versão anterior, de acordo com o Roteiro de Métricas de Software do SISP.

1.1.5.2. Por versão atual considera-se aquela versão do software que contiver os aprimoramentos homologados pelo Product Owner (PO) até a data final do período escolhido pela CAPES para aferição do Índice de Evolução do Sistema (IES).

1.1.5.3. Por versão anterior do sistema entende-se aquela analisada e medida na aferição imediatamente anterior do IES.

1.1.5.4. Eventuais aprimoramentos no software, como inclusões, alterações ou exclusões de funcionalidades, que forem realizadas em versões intermediárias do software e que não sejam identificáveis ao se comparar a versão atual com a versão anterior do sistema não serão consideradas no cálculo do TEF e, consequentemente, do ICPTF.

1.1.5.5. Para ilustrar esta regra, consideremos o seguinte cenário:

- a) A CAPES realizou a primeira aferição do IES;
- b) Na próxima sprint foi acrescentado o campo C1-A na tela T1, os campos C2-A e C2-B na tela T2 e o campo C3-A na tela T3.

Tela	Campo(s) acrescentados em relação à versão anterior da tela
T1	C1-A
T2	C2-A / C2-B
T3	C3-A

- c) O Product Owner (PO) homologou as alterações realizadas na sprint.
- d) Na próxima sprint, o campo C2-B da tela T2 foi excluído, foi acrescentado o campo C3-B na tela T3 e não houve alteração na tela T1.

Tela	Campo(s) acrescentados em relação à versão anterior da tela
T1	C1-A
T2	C2-A / C2-B
T3	C3-A / C3-B

- e) O PO novamente homologou estas alterações.
- f) CAPES decidiu aferir o IES novamente.
- g) Nesta nova aferição, serão consideradas somente a inclusão do campo C1-A na tela T1, a inclusão do campo C2-A na tela T2 e a inclusão dos campos C3-A e C3-B na tela T3 para efeito do cálculo do TEF. Ou seja, nem a inclusão do campo C2-B na tela T2 e nem a sua posterior exclusão serão consideradas para fins de apuração da produtividade em tamanho funcional.

Tela	Campo(s) acrescentados em relação à versão anterior da tela
T1	C1-A
T2	C2-A
T3	C3-A / C3-B

1.1.5.6. A produtividade no período em tamanho funcional será obtida dividindo o Tamanho da Evolução Funcional (TEF) do sistema pela quantidade de DIAS-DESENVOLVEDOR do período de aferição.

1.1.5.7. O Item de Controle de Produtividade em Tamanho Funcional (ICPTF) será calculado dividindo a produtividade no período em tamanho funcional pela produtividade-base estabelecida neste Anexo do Edital.

1.1.5.8. Caso o resultado do cálculo do ICPTF seja maior que 150%, será adotado 150% para o valor deste Item de Controle. Caso a soma do ICPTF com o ICPLC seja maior que 100% do IP, será considerado 100% do valor do IP.

#### 1.1.6. **Item de Controle de Produtividade em Linhas de Código (ICPLC)**

1.1.6.1. Para calcular o Item de Controle de Produtividade em Linhas de Código (ICPLC) primeiro será contabilizada a quantidade de linhas de código alteradas no software no período de aferição, com apoio das ferramentas GIT, SVN e SonarQube.

1.1.6.2. É considerada linha de código alterada uma nova linha inserida ou uma linha existente que foi modificada. Não serão contabilizadas linhas removidas, linhas em branco e alterações com intuito de somente aumentar artificialmente o ICPLC.

1.1.6.3. A produtividade no período em linhas de código será obtida dividindo essa quantidade de linhas de código pela quantidade de DIAS-DESENVOLVEDOR do período de aferição.

1.1.6.4. O Item de Controle de Produtividade em Linhas de Código (ICPLC) será calculado dividindo a produtividade no período em linhas de código pela produtividade-base estabelecida neste Anexo do Edital.

1.1.6.5. Caso o resultado do cálculo do ICPLC seja maior que 150%, será adotado 150% para o valor deste Item de Controle. Caso a soma do ICPTF com o ICPLC seja maior que 100% do IP, será considerado 100% do valor do IP.

#### 1.1.7. **Índice de Qualidade (IQ)**

1.1.7.1. O Índice de Qualidade (IQ) será obtido a partir de indicadores de qualidade do código-fonte do software aferidos com apoio da ferramenta SonarQube, versão 6.7.1 LTS ou superior.

1.1.7.2. Cada indicador possui meta a ser alcançada.

1.1.7.3. Os indicadores de qualidade do código-fonte com suas respectivas metas estão relacionados no quadro a seguir:

Grupo	Indicador	Unidade	Meta
Projeto	Problemas confirmados	Unidades	= 0
	Complexidade	Média total	<= 10
	Métodos	Média total	<= 3
	Índice de manutenibilidade	Nota	A
	Índice de confiabilidade	Nota	A
	Índice de segurança	Nota	A
	Taxa de dívida técnica	%	<= 2,5%
	Classes	Média total	<=10
	Arquivos	Média total	<=10
Violações	Linhas duplicadas (%)	%	<=4%
	Problemas impeditivos	Unidades	=0
Testes	Problemas Críticos	Unidades	=0
	Testes unitários ignorados	Unidades	=0
	Sucesso em testes unitários (%)	%	=100%
	Cobertura (camada de negócio)	%	>=70%

1.1.7.4. O Índice de Qualidade (IQ) será calculado dividindo a quantidade de indicadores que alcançaram as suas respectivas metas pelo quantitativo total de indicadores.

1.1.7.5. Caso algum indicador seja depreciado ou removido em futura versão do SonarQube que venha a ser adotada pela CAPES e não seja substituído por indicador equivalente nesta mesma ferramenta, ele deixará de ser considerado para fins de aferição do cumprimento de níveis mínimos de serviço no âmbito desta contratação.

1.1.7.6. Problemas de qualidade no código-fonte do software pré-existentes à abertura da OS serão desconsiderados na aferição do Índice de Qualidade (IQ).

### 1.1.8. Avaliação do Product Owner (APO)

1.1.8.1. Considerando os princípios e valores do desenvolvimento ágil, o Product Owner (PO) do projeto deverá avaliar os seguintes aspectos da prestação do serviço.

Item	Avaliação
1 - Periodicidade de entrega de software para homologação (preferência por períodos entre 1 a 4 semanas)	Entregas realizadas em período igual ou menor a 4 semanas: <input type="checkbox"/> 4 – Todas <input type="checkbox"/> 3 – Maioria <input type="checkbox"/> 2 – Metade <input type="checkbox"/> 1 – Minoria <input type="checkbox"/> 0 – Nenhuma
2 - Receptividade da equipe de desenvolvimento a mudança de requisitos, mesmo em estágio avançado de desenvolvimento	<input type="checkbox"/> 4 – Não houve resistência à nenhuma solicitação de mudança de requisitos <input type="checkbox"/> 3 – Não houve resistência à maioria das solicitações de mudança de requisitos <input type="checkbox"/> 2 – Não houve resistência à metade das solicitações de mudança de requisitos <input type="checkbox"/> 1 – Não houve resistência à minoria das solicitações de mudança de requisitos <input type="checkbox"/> 0 – Houve resistência a todas às solicitações de mudança de requisitos
3 - Manutenção pela equipe de desenvolvimento de diálogo contínuo e eficiente com o Product Owner para esclarecer dúvidas e obter feedback	Eventual falta de comunicação da equipe de desenvolvimento com o PO ocasionou: <input type="checkbox"/> 4 – nenhuma falha ou desvio <input type="checkbox"/> 3 – falha ou desvio na minoria das entregas <input type="checkbox"/> 2 – falha ou desvio na metade das entregas <input type="checkbox"/> 1 – falha ou desvio na maioria das entregas <input type="checkbox"/> 0 – falha ou desvio em todas as entregas
Comentários adicionais (optativo)	

1.1.8.2. O valor final da Avaliação do Product Owner (APO) será obtido da seguinte forma:

1.1.8.3.  $APO = (Valor\ do\ Item\ 1 + Valor\ do\ Item\ 2 + Valor\ do\ Item\ 3) / 12$

1.1.8.4. Em casos de indisponibilidade do PO, quem fará a APO é o Fiscal Requisitante da Ordem de Serviço.

1.1.9. **Descontos a serem aplicados de acordo com o índice de cumprimento dos níveis de serviço**

1.1.9.1. O quadro a seguir mostra o desconto a ser aplicado no faturamento da Ordem de Serviço (OS) de acordo com o Índice de Evolução do Sistema (IES):

Índice de Evolução do Sistema (IES)	Desconto sobre o valor de faturamento da OS
Igual ou superior a 80%	0%
Igual ou superior a 70% e inferior a 80%	10%
Igual ou superior a 60% e inferior a 70%	20%
Igual ou superior a 50% e inferior a 60%	30%
Inferior a 50%	40%

1.1.9.2. O quadro a seguir mostra o desconto a ser aplicado no faturamento da OS de acordo com níveis mínimos de serviço de componentes do IES, sem prejuízo a outros eventuais descontos previstos neste Edital:

Componente do IES	Nível de serviço	Desconto sobre o valor de faturamento da OS
Índice de Qualidade (IQ)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%
Avaliação do Product Owner (APO)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%

#### 1.1.10. Descontos a serem aplicados na primeira aferição do Índice de Evolução do Sistema (IES)

1.1.10.1. Excepcionalmente, na primeira aferição do Índice de Evolução do Sistema (IES) da Ordem de Serviço (OS), caso o período considerado não ultrapasse mais que um terço do período total da OS, os descontos a serem aplicados no faturamento da Ordem de Serviço considerarão os do quadro a seguir.

Índice de Evolução do Sistema (IES)	Desconto sobre o valor de faturamento da OS
Igual ou superior a 60%	0%
Igual ou superior a 50% e inferior a 60%	10%
Igual ou superior a 40% e inferior a 50%	20%
Inferior a 40%	30%

Componente do IES	Nível de serviço	Desconto sobre o valor de faturamento da OS
Índice de Qualidade (IQ)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%
Avaliação do Product Owner (APO)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%

#### 1.1.11. Produtividade-base

1.1.11.1. Nos quadros a seguir temos a produtividade-base em tamanho funcional e em linhas de código que serão utilizadas na aferição do cumprimento do Índice de Evolução do Sistema (IES):

Produtividade-base em Tamanho Funcional	Produtividade-base em Linhas de Código	Tecnologia
0,48 Pontos de Função / DIA-DESENVOLVEDOR	25,95 Linhas de Código / DIA-DESENVOLVEDOR	Java
	64,81 Linhas de Código / DIA-DESENVOLVEDOR	PHP
	24,00 Linhas de Código / DIA-DESENVOLVEDOR	Python

## 1.2. Índice de Manutenção de Sistema (IMS)

1.2.1. Os níveis de serviço de OS do tipo MANUTENÇÃO serão representados por indicador de desempenho denominado Índice de Manutenção de Sistema (IMS), que será a média ponderada de 3 componentes:

Componente do IMS	Peso
Índice de Produtividade de Manutenção (IP-M)	0,45
Índice de Qualidade de Manutenção (IQ-M)	0,40
Avaliação do Product Onwer - Manutenção (APO-M)	0,15

1.2.2. O Índice de Manutenção do Sistema - IMS será aferido em períodos de 1 a 2 meses, em momento escolhido pela CAPES, e obrigatoriamente ao término da Ordem de Serviço.

1.2.3. Adicionalmente, há níveis mínimos de serviço exigidos para alguns componentes do IMS isoladamente, conforme item 1.2.27 – “Descontos a serem aplicados de acordo com o índice de cumprimento dos níveis de serviço”.

### 1.2.4. Índice de Produtividade de Manutenção (IP-M)

1.2.4.1. O Índice de Produtividade de Manutenção (IP-M) é composto por dois itens: Item de Controle de Produtividade em Tamanho Funcional de Manutenção (ICPTF-M) e Item de Controle de Produtividade em Linhas de Código de Manutenção (ICPLC-M). Estes itens serão aferidos comparando a produtividade no período de aferição com a produtividade- base estabelecida neste Anexo do Edital.

1.2.4.2. A divisão do peso do IP entre seus dois componentes encontra-se no quadro a seguir:

Componente do IP-M	Peso
Item de Controle de Produtividade em Tamanho Funcional de Manutenção (ICPTF-M)	0,25
Item de Controle de Produtividade em Linhas de Código de Manutenção (ICPLC- M)	0,20
Peso Total do IP-M	0,45

1.2.4.3. A produtividade será estabelecida em DIA-DESENVOLVEDOR como no caso do Índice de Evolução do Sistema - IES.

### 1.2.5. Item de Controle de Produtividade em Tamanho Funcional de Manutenção (ICPTF-M)

1.2.5.1. Para calcular o Item de Controle de Produtividade em Tamanho Funcional de Manutenção (ICPTF-M), deve ser feita, em princípio, a medição detalhada em pontos de função de cada demanda de manutenção atendida e homologada no período, de acordo com o Roteiro de Métricas de Software do SISP.

1.2.5.2. Caso o resultado do cálculo do ICPTF-M seja maior que 150%, será adotado 150% para o valor deste Item de Controle. Caso a soma do ICPTF-M com o ICPLC-M seja maior que 100% do IP-M, será considerado 100% do valor do IP-M.

1.2.5.3. Entretanto, considerando o custo da medição detalhada em pontos de função de grande volume de demandas de manutenção, esta medição somente será realizada em caráter excepcional, a pedido de uma das partes contratuais para defesa de seus interesses.

1.2.5.4. Por exemplo, o Tribunal poderá exigir a medição detalhada caso haja indícios de que a produtividade da CONTRATADA esteja abaixo do exigido neste Edital. Por outro lado, a CONTRATADA poderá exigir a medição detalhada caso a CAPES reclame do desempenho do serviço prestado no âmbito de uma Ordem de Serviço.

1.2.5.5. Não sendo solicitada a medição detalhada, adotar-se-á 100% de cumprimento do ICPTF-M.

**1.2.6. Item de Controle de Produtividade em Linhas de Código de Manutenção (ICPLC-M)**

1.2.6.1. O cálculo do Item de Controle de Produtividade em Linhas de Código de Manutenção (ICPLC-M) considerará a quantidade de linhas de código alteradas no período de aferição para atendimento das demandas de manutenção.

1.2.6.2. Este cálculo será feito de modo similar ao do Item de Controle de Produtividade em Linhas de Código (ICPLC).

1.2.6.3. Caso o resultado do cálculo do ICPLC -M seja maior que 150%, será adotado 150% para o valor deste Item de Controle. Caso a soma do ICPTF-M com o ICPLC-M seja maior que 100% do IP-M, será considerado 100% do valor do IP-M.

**1.2.7. Índice de Qualidade de Manutenção (IQ-M)**

1.2.7.1. O Índice de Qualidade (IQ-M) será obtido a partir de indicadores de qualidade do código-fonte do software aferidos com apoio da ferramenta SonarQube, versão 6.7.1 LTS ou superior.

1.2.7.2. Cada indicador possui meta a ser alcançada.

1.2.7.3. Os indicadores de qualidade do código-fonte estão relacionados no quadro do item 1.1.7.3. Contudo, a meta de qualidade em OS de MANUTENÇÃO é não reduzir o resultado dos indicadores que o software manutenido possuía no momento da abertura da Ordem de Serviço.

1.2.7.4. O Índice de Qualidade (IQ-M) será calculado dividindo a quantidade de indicadores que alcançaram as suas respectivas metas pelo quantitativo total de indicadores.

1.2.7.5. O (IQ-M) dever obter sempre 100%

1.2.7.6. Caso algum indicador seja depreciado ou removido em futura versão do SonarQube que venha a ser adotada pela CAPES e não seja substituído por indicador equivalente nesta mesma ferramenta, ele deixará de ser considerado para fins de aferição do cumprimento de níveis mínimos de serviço no âmbito desta contratação.

**1.2.8. Avaliação do Product Owner – Manutenção (APO-M)**

1.2.8.1. Considerando os princípios e valores do desenvolvimento ágil, o Product Owner (PO) do sistema manutenido deverá avaliar os seguintes aspectos da prestação do serviço.

Item	Avaliação
1 - Estabilidade das modificações realizadas	<p>Eventual erro no atendimento das demandas ocasionou:</p> <p>[ ] 4 – nenhum erro em produção  [ ] 3 – até 1 falha em produção  [ ] 2 – até 3 falhas em produção  [ ] 1 – até 5 falhas em produção  [ ] 0 – mais que 5 falhas em produção</p>
2 - Manutenção pela equipe de desenvolvimento de diálogo contínuo e eficiente com o Product Owner para esclarecer dúvidas e obter feedback	<p>Eventual falta de comunicação da equipe de desenvolvimento com o PO ocasionou:</p> <p>[ ] 4 – nenhuma falha ou desvio  [ ] 3 – falha ou desvio na minoria das entregas  [ ] 2 – falha ou desvio na metade das entregas  [ ] 1 – falha ou desvio na maioria das entregas  [ ] 0 – falha ou desvio em todas as entregas</p>

Item	Avaliação
Comentários adicionais (optativo)	

1.2.8.2. O valor final da Avaliação do Product Owner - Manutenção (APO-M) será obtido da seguinte forma:

1.2.8.3.  $APO-M = (\text{Valor do Item 1} + \text{Valor do Item 2}) / 8$

1.2.8.4. Em casos de indisponibilidade do PO, quem fará a APO é o Fiscal Requisitante da Ordem de Serviço.

1.2.9. **Descontos a serem aplicados de acordo com o índice de cumprimento dos níveis de serviço**

1.2.9.1. O quadro a seguir mostra o desconto a ser aplicado no faturamento da Ordem de Serviço (OS) de acordo com o Índice de Manutenção do Sistema (IMS):

Índice de Manutenção do Sistema (IMS)	Desconto sobre o valor de faturamento da OS
Igual ou superior a 80%	0%
Igual ou superior a 70% e inferior a 80%	10%
Igual ou superior a 60% e inferior a 70%	20%
Igual ou superior a 50% e inferior a 60%	30%
Inferior a 50%	40%

1.2.9.2. O quadro a seguir mostra o desconto a ser aplicado no faturamento da OS de acordo com níveis mínimos de serviço de componentes do IMS, sem prejuízo a outros eventuais descontos previstos neste Edital:

Componente do IMS	Nível de serviço	Desconto sobre o valor de faturamento da OS
Avaliação do Product Owner - Manutenção (APO-M)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%

1.2.10. **Descontos a serem aplicados na primeira aferição do Índice de Manutenção do Sistema (IMS)**

1.2.10.1. Excepcionalmente, na primeira aferição do Índice de Manutenção do Sistema (IMS) da Ordem de Serviço (OS), caso o período considerado não ultrapasse mais que um terço do período total da OS, os descontos a serem aplicados no faturamento da Ordem de Serviço considerarão os do quadro a seguir.

Índice de Manutenção do Sistema (IMS)	Desconto sobre o valor de faturamento da OS
Igual ou superior a 60%	0%
Igual ou superior a 50% e inferior a 60%	10%
Igual ou superior a 40% e inferior a 50%	20%
Inferior a 40%	30%

Componente do IMS	Nível de serviço	Desconto sobre o valor de faturamento da OS
Avaliação do Product Owner - Manutenção (APO-M)	Igual ou superior a 50%	0%
	Igual ou superior a 25% e inferior a 50%	5%
	Inferior a 25%	10%

### 1.2.11. Produtividade-base

1.2.11.1. Nos quadros a seguir temos a produtividade-base em tamanho funcional e em linhas de código que serão utilizadas na aferição do cumprimento do Índice de Manutenção do Sistema (IMS):

Produtividade-base em Tamanho Funcional (Manutenção)	Produtividade-base em Linhas de Código (Manutenção)	Tecnologia
0,36 Pontos de Função / DIA-DESENVOLVEDOR	19,46 Linhas de Código / DIA-DESENVOLVEDOR	Java
	48,61 Linhas de Código / DIA-DESENVOLVEDOR	PHP
	18,00 Linhas de Código / DIA-DESENVOLVEDOR	Python

### 1.3. Taxa Efetiva de Ocupação dos Postos de Trabalho da Ordem de Serviço (TEOPT)

1.3.1. Juntamente com a aferição do Índice de Evolução do Sistema (IES) será verificada a Taxa Efetiva de Ocupação dos Postos de Trabalho - TEOPT (quantidade de membros da equipe técnica da CONTRATADA) previstos na Ordem de Serviço.

1.3.2. A TEOPT será a calculada dividindo a ocupação efetiva dos postos de trabalho (em dias- desenvolvedor) pela ocupação total prevista para este período (também em dias- desenvolvedor).

1.3.3. O quadro a seguir mostra o desconto adicional a ser aplicado no faturamento mensal da Ordem de Serviço de acordo com a TEOPT, sem prejuízo do desconto a ser aplicado por dia de não-ocupação de posto de trabalho, conforme previsto no item 19.23 do Termo de Referência (Sei nº 0996410):

Taxa Efetiva de Ocupação dos Postos de Trabalho previstos na OS (TEOPT)	Desconto sobre o valor do faturamento mensal da OS
Igual ou superior a 80%	0%
Igual ou superior a 50% e inferior a 80%	80% - TEOPT
Inferior a 50%	40%

1.3.4. A CAPES, a seu critério, poderá desconsiderar período de não-ocupação de posto de trabalho para efeito de aferição da Taxa Efetiva de Ocupação dos Postos de Trabalho (TEOPT) caso o profissional que ocupe regularmente o posto se ausente por razão que não enseje desconto salarial de acordo com a legislação trabalhista vigente.

1.3.5. Excepcionalmente, caso os resultados obtidos no período de aferição do Índice de Evolução do Sistema corresponderem a, no mínimo, 80% do IES considerando 100% de TEOPT, serão desconsideradas eventuais não-ocupações de postos de trabalho até o limite de 5% da ocupação total prevista para o período para efeito de desconto por não preenchimento do posto de trabalho e para efeito do cálculo do TEOPT.

### 1.4. Inexecução parcial do contrato por descumprimento de nível de serviço

1.4.1. Poderá caracterizar inexecução parcial do contrato:

1.4.2. Descumprimentos de níveis mínimos de serviço no âmbito da mesma Ordem de Serviço que tenham ensejado desconto total igual ou superior a 20% (vinte por cento) do valor de faturamento da OS em dois faturamentos consecutivos ou por três faturamentos em seis faturamentos consecutivos dessa OS.

- 1.4.3. Não alcance de meta do mesmo indicador de qualidade no âmbito da mesma Ordem de Serviço, em três faturamentos consecutivos ou por quatro faturamentos em seis faturamentos consecutivos dessa OS.
- 1.4.4. Descumprimentos de níveis mínimos de serviço que tenham ensejado desconto total igual ou superior a 20% (vinte por cento) do valor de faturamento da OS em três faturamentos dessa OS ou de OS distintas no período de 6 (seis) meses no âmbito do contrato.
- 1.4.5. Tentativa de burla de mecanismos de aferição dos níveis de serviço previstos neste Edital.

## 2. NÍVEIS DE SERVIÇO - GRUPO 2 - MENSURAÇÃO DE SISTEMAS

2.1. A CAPES avaliará os serviços executados em cada Ordem de Serviço por meio da utilização de Indicadores, apresentados abaixo, que estabelecem critérios de prazo e qualidade dos produtos entregues.

### 2.2. INDICADOR 1 - TEMPO DE ATRASO PARA SERVIÇOS EM PFs

2.2.1. Mede o tempo de atraso nas entregas parciais ou totais da execução das Ordens de Serviço (OS).

2.2.2. A aferição ocorre a cada entrega parcial ou total prevista de Ordem de Serviço.

2.2.3. Fórmula de cálculo:

$$T_{AE} = P_{EE} - P_{PE}$$

Onde:

$T_{AE}$  = Tempo de atraso da entrega;

$P_{EE}$  = Prazo efetivo de entrega, em dias úteis;

$P_{PE}$  = Prazo planejado de entrega, em dias úteis, acordado na O.S.

2.2.4. Nível mínimo de serviço exigido:

$T_{AE}$  Desejável = 0 (zero) dias;

$T_{AE}$  Aceitável = 1 dia útil;

$T_{AE}$  Inaceitável = acima de 1 dia útil;

2.2.5. Sanções:

$T_{AE}$  Desejável = Nenhuma;

$T_{AE}$  Aceitável = Advertência;

$T_{AE}$  Inaceitável = Glosa de 2% (dois por cento) no valor da OS + 0,66% por dia de atraso até o limite de 20%, no valor da entrega parcial ou total da OS correspondente.

### 2.3. INDICADOR 2 - QUALIDADE

2.3.1. Mede a quantidade de erros ou não conformidades nos produtos e artefatos decorrentes da entrega da Ordem de Serviço (OS).

2.3.2. Entende-se por não conformidade tudo aquilo que estiver em desacordo com o especificado numa Ordem de Serviço.

2.3.3. A aferição ocorre a cada entrega parcial, total ou chamado previsto da Ordem de Serviço.

2.3.4. Fórmula de cálculo:

$$Q_{ENE} = \text{Nº. de erros e não conformidades encontradas}$$

Onde:

$Q_{ENE}$  = Número de erros e não conformidades encontradas;

2.3.5. Nível mínimo de serviço exigido:

$Q_{ENE}$  Desejável = 0 (zero) erros;

**Q<sub>ENE</sub>** Aceitável = 5 erros;

**Q<sub>ENE</sub>** Inaceitável = acima de 5 erros;

### 2.3.6. Sanções:

**Q<sub>ENE</sub>** Desejável = Nenhuma;

**Q<sub>ENE</sub>** Aceitável = Advertência;

**Q<sub>ENE</sub>** Inaceitável = Glosa de 2% (dois por cento) no valor da OS correspondente.

### 2.4. Para análise da qualidade de uma entrega, serão considerados os seguintes requisitos:

Requisitos	Descrição
Completude:	Todos os artefatos solicitados na Ordem de Serviço (OS) deverão ter sido entregues.
Forma:	Os artefatos deverão ser construídos dentro dos padrões e metodologias adotadas no Desenvolvimento e Manutenção de Sistemas da CONTRATANTE.
Localização:	Os artefatos deverão ser entregues em locais a serem determinados pela CONTRATANTE seguindo normas de padronização e nomenclatura. Entende-se por artefatos qualquer documentação utilizada para abertura de uma Ordem de Serviço bem como resultante de sua execução
Testabilidade:	Os artefatos e produtos deverão estar em condições para serem testados pela CONTRATANTE.
Rastreabilidade:	Deve ser possível a rastreabilidade de um artefato e dos demais a ele relacionados.
Referências:	Todos os insumos utilizados devem estar corretamente referenciados.
Consistência:	Não deve haver duplicidade nem conflito de artefatos ou informações.
Coerência:	O histórico com as atualizações do artefato deve ser corretamente atualizado.
Histórico:	Os responsáveis pela elaboração ou manutenção do artefato devem estar corretamente identificados.
Responsabilidade:	Os responsáveis pela elaboração ou manutenção do artefato devem estar corretamente identificados.
Completude:	Um artefato só será aceito quando estiver completamente terminado.
Respeito às Normas:	Normas gerais da CAPES, com sua política de segurança da informação deverão ser respeitadas na construção dos artefatos.



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE B - PERFIS E QUALIFICAÇÃO PROFISSIONAL

##### 1. **PERFIS PROFISSIONAIS PARA O GRUPO 1 - DESENVOLVIMENTO DE SISTEMAS**

###### 1.1. **ANALISTA PROGRAMADOR**

1.1.1. Profissional da CONTRATADA responsável por executar os serviços necessários a codificação de aplicações, sistemas, componentes e/ou serviços (back-end), abrangendo o levantamento de requisitos, manipulação de banco de dados, elaboração de documentação técnica e manuais de usuários, além de atividades relacionadas à criação/adaptação de interface do usuário (front-end). Dessa forma o profissional deve ter um conhecimento multidisciplinar nas várias áreas exigidas. Tal perfil é comumente denominado “full-stack” e visa valorizar as habilidades e os conhecimentos de computação do desenvolvedor e da equipe, em linha com o que pregam as orientações “ágil”.

###### 1.1.2. **ANALISTA PROGRAMADOR SÊNIOR**

###### 1.1.2.1. **Responsabilidades:**

- a) Trabalhar em equipe de desenvolvimento (código/testes) com múltiplos projetos;
- b) Entender e transformar requisitos de negócio, funcionais e não funcionais, em código modular e reutilizável;
- c) Aplicar as boas práticas de engenharia de software e padrões de projeto;
- d) Adotar frameworks, linguagens de programação, arquitetura e divisão de componentes do sistema e documentação ao longo do ciclo de vida do projeto;
- e) Implementar e manter aplicações, bibliotecas, componentes e frameworks;
- f) Desenvolver e fazer manutenção de códigos/testes de sistemas;
- g) Orientar programadores Plenos e Juniores nos processos de desenvolvimento;

###### 1.1.2.2. **Formação acadêmica:**

- a) Graduação em curso de nível superior na área de Tecnologia da Informação, ou conclusão de qualquer curso de nível superior acompanhado de certificado de curso de pós-graduação (especialização, mestrado ou doutorado) na área de Tecnologia da Informação de, no mínimo, 360 horas.

###### 1.1.2.3. **Requisitos obrigatórios por tecnologia:**

###### 1.1.2.4. **JAVA**

- a) Experiência com EJB 3.x (Stateless Session Bean, Message Drive Bean, Statefull Session Bean) e Java Message Service em geral;
- b) Vasta experiência em algoritmos e estrutura de dados;

- c) Experiência com o servidor de aplicações JBOSS, contemplando configuração básica, administração, deploy e otimização;
- d) Experiência em JAX-RS, JAX-WS, JAXB, RMI, JPA, JTA, JDBC, SQL, segurança JEE, framework springboot e spring cloud;
- e) Experiência em bancos de dados relacionais e não relacionais;
- f) Experiência com ECMAScript 5 e 6;
- g) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- h) Experiências nas melhores práticas de codificação / design e padrões de projeto, otimizações, escalabilidade e clusterização;
- i) Sólidos conhecimentos em ambiente DevOps;
- j) Sólidos conhecimentos em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- k) Vasta experiência em Agile e práticas de integração contínua e análise estática de código;
- l) Experiência em ferramentas de versionamento como GIT;
- m) Experiência em gerenciamento de dependências utilizando maven;
- n) Inglês técnico (leitura e escrita).

#### 1.1.2.5.

##### PHP

- a) Experiência nos principais frameworks (Laravel, Lumen, Zend, entre outros);
- b) Vasta experiência em algoritmos e estrutura de dados;
- c) Experiência em bancos de dados relacionais e não relacionais;
- d) Experiência com ECMAScript 5 e 6;
- e) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- f) Experiência nas melhores práticas de codificação / design e padrões de projeto, otimizações, escalabilidade e clusterização;
- g) Sólidos conhecimentos em ambiente DevOps;
- h) Sólidos conhecimentos em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- i) Vasta experiência em Agile e práticas de integração contínua e análise estática de código;
- j) Experiência em ferramentas de versionamento como GIT;
- k) Experiência em gerenciamento de dependências utilizando Composer;
- l) Inglês técnico (leitura e escrita).

#### 1.1.2.6.

##### PYTHON

- a) Experiência em frameworks de mercado como Django, Flask ou CherryPy;
- b) Vasta experiência em algoritmos e estrutura de dados;
- c) Experiência nas melhores práticas de codificação / design e padrões de projeto, otimizações, escalabilidade e clusterização;
- d) Experiência em bancos de dados relacionais e não relacionais;
- e) Experiência com ECMAScript 5 e 6;
- f) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;

- g) Sólidos conhecimentos em ambiente DevOps;
- h) Sólidos conhecimentos em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- i) Vasta conhecimentos em Agile e práticas de integração contínua e análise estática de código;
- j) Experiência em ferramentas de versionamento como GIT;
- k) Experiência em gerenciamento de dependências utilizando pip;
- l) Inglês técnico (leitura e escrita).

#### 1.1.2.7. **Certificações ou período de experiência por tecnologia:**

##### 1.1.2.8. JAVA

- a) Experiência mínima de 7 anos em desenvolvimento em java; ou
- b) Experiência mínima de 5 anos em desenvolvimento java e ao menos duas **Certificações de Desenvolvedor JAVA em vigor e válidas:**

##### 1.1.2.9. PHP

- a) Experiência mínima de 7 anos em desenvolvimento em php; ou
- b) Experiência mínima de 5 anos em desenvolvimento php e ao menos duas **Certificações de Desenvolvedor PHP em vigor e válidas.**

##### 1.1.2.10. PYTHON

- a) Experiência mínima de 7 anos em desenvolvimento em python; ou
- b) Experiência mínima de 5 anos em desenvolvimento python e ao menos:
  - I - **Uma Certificação em Python; ou**
  - II - **Certificado de conclusão de treinamento em Python de no mínimo 80 horas.**

#### 1.1.2.11. **Competências comportamentais:**

- a) Proatividade, capacidade de trabalho em equipe, capacidade de autogerenciamento e tomada de decisão, capacidade de comunicação (capacidade de se expressar oralmente e por escrito com precisão e clareza e de compreender com facilidade mensagens escritas e faladas).

#### 1.1.3. **ANALISTA PROGRAMADOR PLENO**

##### **1.1.3.1. Responsabilidades:**

- a) Trabalhar em equipe de desenvolvimento (código/testes) com múltiplos projetos;
- b) Entender e transformar requisitos de negócio, funcionais e não funcionais, em código modular e reutilizável;
- c) Aplicar as boas práticas de engenharia de software e padrões de projeto;
- d) Implementar e manter aplicações, manter biblioteca e componentes;
- e) Desenvolver e fazer manutenção de códigos/testes de sistemas;

##### **1.1.3.2. Formação acadêmica:**

- a) Graduação em curso de nível superior na área de Tecnologia da Informação, ou conclusão de qualquer curso de nível superior acompanhado de certificado de curso de pós-graduação (especialização, mestrado ou doutorado) na área de Tecnologia da Informação de, no mínimo, 360 horas.

##### **1.1.3.3. Requisitos obrigatórios por tecnologia:**

##### 1.1.3.4. JAVA

- a) Experiência com EJB 3.x (Stateless Session Bean, Message Drive Bean, Statefull Session Bean) e Java Message Service em geral;
- b) Vasta experiência em algoritmos e estrutura de dados;
- c) Experiência com o servidor de aplicações JBOSS, contemplando configuração básica, administração, deploy e otimização;
- d) Experiência em JAX-RS, JAX-WS, JAXB, RMI, JPA, JTA, JDBC, SQL, segurança JEE, framework springboot e spring cloud;
- e) Experiência em bancos de dados relacionais e não relacionais;
- f) Experiência com ECMAScript 5 e 6;
- g) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- h) Conhecimento das melhores práticas de codificação / design e padrões de projeto;
- i) Conhecimentos em ambiente DevOps;
- j) Vivência em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- k) Conhecimentos em Agile e práticas de integração contínua e análise estática de código.
- l) Conhecimento em ferramentas de versionamento como GIT;
- m) Conhecimento em gerenciamento de dependências utilizando maven;
- n) Inglês técnico (leitura e escrita).

#### 1.1.3.5.

##### PHP

- a) Conhecimento dos principais frameworks (Laravel, Lumen, Zend, entre outros);
- b) Conhecimento em algoritmos e estrutura de dados;
- c) Experiência em bancos de dados relacionais e não relacionais;
- d) Experiência com ECMAScript 5 e 6;
- e) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- f) Conhecimento das melhores práticas de codificação / design e padrões de projeto;
- g) Conhecimento em ambiente DevOps;
- h) Vivência em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- i) Conhecimento em Agile e práticas de integração contínua e análise estática de código.
- j) Conhecimento em ferramentas de versionamento como GIT;
- k) Conhecimento em gerenciamento de dependências utilizando Composer;
- l) Inglês técnico (leitura e escrita).

#### 1.1.3.6.

##### PYTHON

- a) Experiência em frameworks de mercado como Django, Flask ou CherryPy;
- b) Conhecimento em algoritmos e estrutura de dados;
- c) Conhecimento das melhores práticas de codificação / design e padrões de projeto, otimizações, escalabilidade e clusterização;
- d) Experiência em bancos de dados relacionais e não relacionais;
- e) Experiência com ECMAScript 5 e 6;

- f) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- g) Conhecimento em ambiente DevOps;
- h) Conhecimento em plataformas para o provisionamento e gerenciamento de containers (Rancher, OpenShift entre outros);
- i) Conhecimento em Agile e práticas de integração contínua e análise estática de código.
- j) Conhecimento em ferramentas de versionamento como GIT;
- k) Conhecimento em gerenciamento de dependências utilizando pip;
- l) Inglês técnico (leitura e escrita).

#### 1.1.3.7. **Certificações ou período de experiência por tecnologia:**

##### 1.1.3.8. JAVA

- a) Experiência mínima de 5 anos em desenvolvimento java; ou
- b) Experiência mínima de 3 anos em desenvolvimento java e ao menos uma **Certificação de Desenvolvedor JAVA em vigor e válida.**

##### 1.1.3.9. PHP

- a) Experiência mínima de 5 anos em desenvolvimento php; ou
- b) Experiência mínima de 3 anos em desenvolvimento php e ao menos uma **Certificação de Desenvolvedor PHP:**

##### 1.1.3.10. PYTHON

- a) Experiência mínima de 5 anos em desenvolvimento python; ou
- b) Experiência mínima de 3 anos em desenvolvimento python e ao menos uma **certificação em Python ou certificado de conclusão de treinamento em Python de no mínimo 40 horas.**

#### 1.1.3.11. **Competências comportamentais:**

- a) Proatividade, capacidade de trabalho em equipe, capacidade de iniciativa e autogerenciamento, capacidade de comunicação (capacidade de se expressar oralmente e por escrito com precisão e clareza e de compreender com facilidade mensagens escritas e faladas).

#### 1.1.4. **ANALISTA PROGRAMADOR JÚNIOR**

##### 1.1.4.1. **Responsabilidades:**

- a) Trabalhar em equipe de desenvolvimento (código/testes) com múltiplos projetos;
- b) Entender e transformar requisitos de negócio, funcionais e não funcionais, em código modular e reutilizável;
- c) Implementar e manter aplicações;
- d) Desenvolver e fazer manutenção de códigos/testes de sistemas

##### 1.1.4.2. **Formação acadêmica:**

- a) Graduação em curso de nível superior na área de Tecnologia da Informação, ou conclusão de qualquer curso de nível superior acompanhado de certificado de curso de pós-graduação (especialização, mestrado ou doutorado) na área de Tecnologia da Informação de, no mínimo, 360 horas.

##### 1.1.4.3. **Requisitos obrigatórios por tecnologia:**

##### 1.1.4.4. JAVA

- a) Experiência com EJB 3.x (Stateless Session Bean, Message Drive Bean, Statefull Session Bean) e Java Message Service em geral;
- b) Experiência em algoritmos e estrutura de dados;
- c) Conhecimento com o servidor de aplicações JBOSS, contemplando configuração básica, administração, deploy e otimização;
- d) Conhecimento em JAX-RS, JAX-WS, JAXB, RMI, JPA, JTA, JDBC, SQL, segurança JEE, framework springboot e spring cloud;
- e) Conhecimento em bancos de dados relacionais e não relacionais;
- f) Experiência com ECMAScript 5 e 6;
- g) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- h) Conhecimento das melhores práticas de codificação / design e padrões de projeto;
- i) Conhecimento em ambiente DevOps;
- j) Conhecimento em Agile e práticas de integração contínua e análise estática de código;
- k) Conhecimento em ferramentas de versionamento como GIT;
- l) Conhecimento em gerenciamento de dependências utilizando maven;
- m) Inglês técnico (leitura e escrita).

#### 1.1.4.5.

##### PHP

- a) Conhecimento dos principais frameworks (Laravel, Lumen, Zend, entre outros);
- b) Experiência em algoritmos e estrutura de dados;
- c) Conhecimento em bancos de dados relacionais e não relacionais;
- d) Experiência com ECMAScript 5 e 6;
- e) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- f) Conhecimento das melhores práticas de codificação / design e padrões de projeto;
- g) Conhecimento em ambiente DevOps;
- h) Conhecimento em Agile e práticas de integração contínua e análise estática de código;
- i) Conhecimento em ferramentas de versionamento como GIT;
- j) Conhecimento em gerenciamento de dependências utilizando Composer;
- k) Inglês técnico (leitura e escrita).

#### 1.1.4.6.

##### PYTHON

- a) Experiência em frameworks de mercado como Django, Flask ou CherryPy;
- b) Conhecimentos em algoritmos e estrutura de dados;
- c) Experiência com ECMAScript 5 e 6;
- d) Experiência em tecnologias Web GUI, como Angular, HTML5 e CSS3;
- e) Conhecimento das melhores práticas de codificação / design e padrões de projeto;
- f) Conhecimento em ambiente DevOps;
- g) Conhecimento em Agile e práticas de integração contínua e análise estática de código;
- h) Conhecimento em ferramentas de versionamento como GIT;

- i) Conhecimento em gerenciamento de dependências utilizando pip;
- j) Inglês técnico (leitura e escrita).

**1.1.4.7. Certificações ou período de experiência por tecnologia:**

**1.1.4.8. JAVA**

- a) Experiência mínima de 2 anos em desenvolvimento java.

**1.1.4.9. PHP**

- a) Experiência mínima de 2 anos em desenvolvimento php.

**1.1.4.10. PYTHON**

- a) Experiência mínima de 2 anos em desenvolvimento python.

**1.1.4.11. Competências comportamentais:**

- a) Proatividade, capacidade de trabalho em equipe, capacidade de iniciativa e autogerenciamento, capacidade de comunicação (capacidade de se expressar oralmente e por escrito com precisão e clareza e de compreender com facilidade mensagens escritas e faladas).

**1.1.5. Ferramentas de apoio**

1.1.6. No caso de ferramentas de apoio ao desenvolvimento de software, frameworks, bibliotecas de software e boas práticas de desenvolvimento, entre outros, que a CAPES venha a adotar no seu processo de desenvolvimento, relativo as linguagens acima relacionadas, os profissionais de ambos os níveis terão prazo de 22 dias úteis para se adaptar ao seu uso.

1.1.7. Exemplos de ferramentas de apoio, frameworks, bibliotecas de software ou boas práticas as quais este item se refere integram o documento da arquitetura de referência vide 0983069, 0983071, 0983080, 0983083 e 0983089.

**1.2. PREPOSTO**

1.2.1. Funcionário representante da CONTRATADA, responsável “in loco” (no ambiente físico da CAPES), por acompanhar a execução do contrato e atuar como interlocutor principal junto à CAPES, incumbido de receber, diligenciar, encaminhar e responder as principais questões, legais e administrativas referentes ao andamento contratual; trabalhar a arquitetura de atendimento de acordo com os tipos de demandas, sempre aderindo à MDS; e resolver impedimentos gerais no nível tático/estratégico do contrato.

**1.2.2. Formação acadêmica:**

- a) Curso superior completo;
- b) Desejável certificações em **Scrum e/ou Kanban**.

**1.2.3. Certificações ou período de experiência:**

a) Experiência profissional de 2 (dois) anos em gestão de contratos, comprovada através de Atestado de Capacidade Técnica, nominal ao profissional, contendo a descrição das principais atividades desenvolvidas pelo profissional, fornecido por órgão da Administração Pública ou empresa privada;

b) Conhecimento em gerenciamento de projetos, gestão de serviços terceirizados comprovado através de Atestado de Capacidade Técnica, nominal ao profissional, contendo a descrição do projeto e as atividades desenvolvidas pelo profissional, fornecido por órgão da Administração Pública ou empresa privada abordando:

- I - Métodos ágeis;
- II - Análise de riscos, planejamento, liderança de equipe, estimativas;

**III - Avaliação de resultados.****2. PERFIS PROFISSIONAIS PARA O GRUPO 2 - MENSURAÇÃO DE SISTEMAS****2.1. ANALISTA DE MÉTRICAS E MENSURAÇÃO DE SOFTWARE**

2.1.1. Profissional da CONTRATADA responsável por realizar a contagem de Pontos de Função em conformidade com o Manual de Práticas de Contagem do IFPUG e com o Guia de Métricas do SISP.

**2.1.2. Formação acadêmica:**

a) Graduação em curso de nível superior na área de Tecnologia da Informação, ou conclusão de qualquer curso de nível superior acompanhado de certificado de curso de pós-graduação (especialização, mestrado ou doutorado) na área de Tecnologia da Informação de, no mínimo, 360 horas.

**2.1.3. Certificações ou período de experiência:**

a) Experiência comprovada de, no mínimo, 3 (três) anos em atividades de métrica e mensuração de projetos de software com a utilização da técnica de Análise de Pontos de Função.

b) Certificação CFPS (Certified Function Point Specialist) em vigor, concedida pelo IFPUG.

**2.2. PREPOSTO**

2.2.1. Funcionário representante da CONTRATADA, responsável por acompanhar a execução do contrato e atuar como interlocutor principal junto à CAPES, incumbido de receber, diligenciar, encaminhar e responder as principais questões, legais e administrativas referentes ao andamento contratual; trabalhar a arquitetura de atendimento de acordo com os tipos de demandas; e resolver impedimentos gerais no nível tático/estratégico do contrato.

**2.2.2. Formação acadêmica:**

a) Curso superior completo;

b) Desejável certificações em **Scrum e/ou Kanban**.

**2.2.3. Certificações ou período de experiência:**

a) Experiência profissional de 2 (dois) anos em gestão de contratos, comprovada através de Atestado de Capacidade Técnica, nominal ao profissional, contendo a descrição das principais atividades desenvolvidas pelo profissional, fornecido por órgão da Administração Pública ou empresa privada;

b) Conhecimento em gerenciamento de projetos, gestão de serviços terceirizados comprovado através de Atestado de Capacidade Técnica, nominal ao profissional, contendo a descrição do projeto e as atividades desenvolvidas pelo profissional, fornecido por órgão da Administração Pública ou empresa privada.



**MDS – METODOLOGIA DE DESENVOLVIMENTO DE SOFTWARE - CAPES**

**Versão: 4.0**

**Histórico de Revisões**

Data	Versão	Descrição	Responsável	Revisado por
21/12/2017	4.0	Criação da metodologia ágil	Grupo de Trabalho instituído pela Portaria DTI no. 1	

## Sumário

## 1. Introdução

O objetivo desse documento é definir e detalhar a proposta de metodologia ágil a ser aplicada em desenvolvimento e manutenção de softwares para a CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.

Este guia é a principal referência do assunto para Gerentes de Projetos, Analistas de Negócios, Desenvolvedores, Arquitetos, Especialistas em Gerência de Configuração, Analistas de Qualidade, Designer de UX e UI que atuam na DTI da CAPES. Pode ainda apoiar o trabalho de outros profissionais da DTI que tenham envolvimento com o assunto.

## 2. Metodologia Ágil

A metodologia ágil é uma filosofia que defende a satisfação do cliente e a entrega incremental prévia; equipes de projetos pequenas e altamente motivadas; métodos informais; artefatos de engenharia de software mínimos e, acima de tudo, simplicidade no desenvolvimento geral. Os princípios de desenvolvimento priorizam a entrega mais que a análise e projeto (embora essas atividades não sejam desencorajadas); também priorizam a comunicação ativa e contínua entre desenvolvedores e clientes. (Pressman, 2011)

Dentro dessa metodologia existem frameworks que tem por objetivo buscar atingir as metas por meio de processos contínuos e fáceis de medir, integrando os participantes e as ferramentas de forma colaborativa, lembrando-se sempre das necessidades do cliente e trocando feedbacks, para que todos possam andar sempre no mesmo ritmo.

Para isso a Capes adotou dois Frameworks para apoio ao método ágil; o Scrum, para projetos de sistemas; e o Kanban para a sustentação de sistemas.

### 2.1. Scrum

Scrum é um framework testado e consolidado pelo mercado para estruturar equipes de desenvolvimento Agile.

Suas principais características são:

- Montar times multidisciplinares;
- Trabalhar como equipe;
- Definir o “dono do produto”: o guardião dos interesses do usuário final;
- Criar uma lista de tarefas que devem ser cumpridas (backlog);
- Determinar um “Scrum Master”;
- Adotar uma rotina de feedbacks regulares e diários;

- Organizar-se por “sprints” de trabalho, com tempo determinado para acabar.

As reuniões diárias são chamadas de “Daily Stand-ups” e ao final de cada sprint, é organizada uma reunião de retrospectiva, para se analisar e discutir tudo que foi feito e guardar aprendizados para os próximos projetos.

### 2.1.1. Objetivos da metodologia Scrum

Conforme já citado o Scrum será o framework a ser adotado no projeto de sistemas da CAPES. Desta forma, será definido de forma clara “quem” faz “o que”, “quando”, “como”, e até mesmo “onde”, para todos que estejam envolvidos diretamente ou não com o desenvolvimento dos sistemas. Além disso, um conjunto de padrões já existentes dão apoio nesta disciplina o que de certo modo evita a subjetividade na abordagem garantindo o padrão de qualidade. Portanto alguns critérios importantes devem ser seguidos, dentre eles, podemos citar:

- O desenvolvimento do software deve ser alinhado às necessidades do negócio;
- Haver entrega do maior valor possível para o negócio com qualidade;
- O processo de trabalho da Equipe do Projeto ser adaptado continuamente;
- As entregas de software serem frequentes;
- Os riscos serem identificados e tratados eficientemente;
- A Equipe do Projeto trabalhar colaborativamente;
- Os envolvidos terem visibilidade do progresso do projeto;
- Atingir as ações do PDTIC
- A documentação do software ser concisa e atualizada.

## 2.2. O Kanban

O Kanban é uma forma de organizar os projetos Agile, por meio de melhorias incrementais, isto é, pequenos avanços graduais de performance ou funcionalidades.

Para a maioria dos especialistas, o Kanban é um método de gestão de mudanças, que dá ênfase nos seguintes princípios:

- Visualizar o trabalho em andamento;
- Visualizar cada passo em sua cadeia de valor, do conceito geral até o software que se possa disponibilizar;
- Limitar o Trabalho em Progresso (WIP – Work in Progress), restrinindo o total o trabalho em cada estágio definido no modelo;
- Tornar explícitas e claras as políticas que estão sendo seguidas;

- Medir e gerenciar o fluxo, para tomar decisões fundamentadas, além de visualizar as consequências das mesmas;
- Identificar oportunidades de melhorias, criando uma cultura Kainzen, na qual a melhoria contínua é responsabilidade de todos;
- Estabelece a filosofia de que começar com o que está fazendo agora; e
- Respeitar o processo atual, com seus papéis, responsabilidades e cargos.

Segundo (Jesper Boeg) o **Kanban** ( com K maiúsculo) é o método de mudança evolucionária que utiliza um sistema **kanban** ( com k minúsculo), além de visualização e outras ferramentas, para catalisar a introdução das ideias Lean nas áres de desenvolvimento e operações de TI.

Todo trabalho no Kanban se torna visível. Os limites WIP são estabelecidos para dar transparência na produtividade e nos indicadores. Com isso, as políticas se tornam explícitas e o fluxo passa a ser medido.

A razão pela qual o sistema Kanban é referido como “Sistema Puxado Kanban” (Kanban Pull System) é que, ao visualizar o fluxo e estabelecer os limites do WIP, garantimos que nunca de pode introduzir mais trabalho no sistema que a capacidade do sistema de processar esse trabalho. É importante frisar que não há regras quanto à aparência específica do quadro.

O Kanban é excelente em situações em que estruturas organizacionais inibem mudanças radicais, desta forma, e fundamental o respeito ao processo atual, seus papéis, responsabilidades e cargos.

Nenhum dos princípios do Kanban restringe o uso do Scrum. O Kanban funciona como um agente de mudanças, e os princípios do Scrum devem, portanto, ser usados apenas nos casos em que ajudam a otimizar o fluxo de trabalho. Desta forma, na impede de se começar com o Scrum e utilizar o Kanban para impulsionar futuras mudanças.

O Kanban é antes de tudo um catalisador para a condução de mudanças e precisa de um ponto de partida. Assim, apesar de a maioria dos projetos poder se beneficiar do uso do kanban, o Kanban não é um substituto do Scrum, que é , na maioria dos casos, um ponto de partida perfeito para a adoção do kanban.

O importante é entender que, independente do seu fluxo de trabalho, sempre haverá uma maneira mais adequada de empregar estas metodologias ou outras ao seu negócio. Desta forma, a maioria dos projetos, ágeis ou não, podem ser beneficiados pelo uso dos princípios do Kanban, para impulsionar as mudanças e a melhoria contínua.

### 2.3. Papéis

Aqui se elencam os papéis específicos de uma equipe ágil, de parceiros diretos em atividades essenciais e as habilidades necessárias de cada um deles para a MDS.

## 2.4. Time Scrum

No Time Scrum cada um é responsável por uma atividade não existem tarefas definidas por perfil. As pessoas trabalham juntas para completar com sucesso uma Sprint. As características importantes para um Time Scrum e que elas sejam Multidisciplinares, Auto-Organizada, tenho um tamanho certo e sejam colaborativas. Fazem parte desse time os seguintes profissionais:

### 2.4.1. Product Owner (PO)

Profissional que representa a área de negócio com conhecimento suficiente para definir e priorizar requisitos do negócio e responder aos questionamentos da equipe de desenvolvimento. É o representante do cliente e sua atuação tem por finalidade garantir a entrega do valor esperado através do produto final do projeto.

### 2.4.2. Gerente de Projetos

Profissional experiente no processo ágil que zela pela sua correta execução e ajuda a equipe na resolução de possíveis impedimentos e dissemina as práticas do processo. Este profissional assume o papel de Scrum Master.

### 2.4.3. Time de Desenvolvimento

O time de desenvolvimento é composto por todas as pessoas responsáveis por transformar o Product Backlog em incrementos de funcionalidades que possam ser entregues ao cliente e que agregam valor.

Os integrantes do time são membros interdisciplinares, o qual devem possuir conhecimentos necessários para desenvolver valor ao trabalho. Estes podem possuir conhecimentos especializados, tais como: controle de qualidade, programação, arquitetura, análise, banco de dados ou outros, mas o mais importante é a habilidade de pegar uma funcionalidade e transformá-la em um produto viável.

Dentre os principais papéis de um Time de Desenvolvimento, devemos citar:

- Fazer as estimativas necessárias;
- Definir as tarefas que serão realizadas;
- Desenvolver o produto;
- Realizar repasses de conhecimento para as equipes de suporte e sustentação;
- Garantir a qualidade do produto;
- Apresentar o produto ao cliente.

Para tanto podemos definir critérios mais específicos dentro de um modelo Híbrido de Métodos Ágeis, tais como:

#### 2.4.3.1. Analista Programador

Profissional responsável por executar os serviços necessários a codificação de aplicações, sistemas, componentes e/ou serviços (back-end), abrangendo o levantamento de requisitos, manipulação de banco de dados, elaboração de documentação técnica e manuais de usuários, além de atividades relacionadas à criação/adaptação de interface do usuário (front-end). Dessa forma o profissional deve ter um conhecimento multidisciplinar nas várias áreas exigidas. Tal perfil é comumente denominado “full-stack” e visa valorizar as habilidades e os conhecimentos de computação do desenvolvedor e da equipe, em linha com o que pregam as orientações “ágil”.

#### 2.4.3.2. Analista de Qualidade

Profissional que garante o processo de qualidade de software junto a CAPES, relacionando-se com os gestores, gerando registro de suas atividades e desenvolvendo artefatos de acordo com padrões pré-estabelecidos.

#### 2.4.3.3. Analista de Negocio

Profissional responsável pelo alinhamento entre as áreas de negócios e a área de TI, no ambiente como desenvolvedora de soluções tecnológicas. Esse especialista é fundamental para a inovação dos produtos e processos, e avanços dos resultados. Atua na elaboração de plano estratégico das áreas de negócios e nos relacionamentos, o que requer uma análise do contexto, com o objetivo de elaborar recomendações de ações e soluções.

#### 2.4.3.4. Designer UX (User Experience)

A ISO 9241-210 define a experiência do usuário como "as percepções e reações de uma pessoa que resultam do uso ou utilização prevista de um produto, sistema ou serviço." De acordo com a definição da ISO, experiência do usuário inclui todas as emoções, crenças, preferências, percepções, respostas físicas e psicológicas, comportamentos e realizações do usuário que ocorrem antes, durante e após o uso.

O(a) Designer UX é quem realiza, principalmente, a elaboração de mapas de navegação do usuário, testes de usabilidade, criação, desenvolvimento e implementação de soluções inovadoras e atraentes, prototipação, fluxo do usuário e de processos, ligando a interação comunicativa às ideias de Design.

Ele(a) também é responsável por realizar benchmarks, estudos de aprimoramento, criando padrões de naveabilidade voltados a otimização de performance, engajamento e taxas de conversão de diversas plataformas.

## 2.4.4. Outros papéis

Além dos profissionais que compõem o Time de desenvolvimento existem outros importantes que fazem parte do processo de desenvolvido, tais como:

### 2.4.4.1. Designer UI (User Interface)

O serviço de design de interface de usuário (UI) consiste, em síntese, na materialização em formato real das interfaces (ex: telas, imagens, ícones, etc) concebidas pelo serviço de aprimoramento da experiência do usuário (UX).

Portanto, em linhas gerais, cabe ao profissional de UX a concepção da forma de interação do usuário com o sistema web, forma esta capturada muitas vezes por meio da construção de protótipos, ficando a cargo do designer de UI a criação, com base nesses protótipos, dos artefatos visuais (ex: páginas HTML, arquivos de imagens e estilos, etc) a serem incorporados ao sistema real.

### 2.4.4.2. Analista de Configuração e Mudança

Profissional da DTI responsável pela configuração e controle dos ambientes de Desenvolvimento, Teste e Homologação.

### 2.4.4.3. Administrador de Dados

Equipe formada por profissionais da DTI, responsável pela geração e manutenção das estruturas de dados. É composta pelo Administrador de Dados (AD) e pelo Administrador de Banco de Dados (DBA).

### 2.4.4.4. Analista de Business Intelligence

Equipe formada por profissionais da DTI, responsável pela geração e disseminação de informação para apoiar a tomada de decisão nos níveis estratégico, tático e operacional na Capes. É composta pelo Analista de ETL, pelo Cientista de Dados e pelo Analista / Administrador de OLAP.

### 2.4.4.5. Analisa de Infraestrutura

Profissional com conhecimento sobre o ambiente de TI e responsável por garantir a disponibilidade e performance adequadas para os serviços de TI que são mantidos por recursos de infra, tais como servidores / clusters.

### 2.4.4.6. Arquiteto

Profissional experiente em programação de computadores e com visão sistêmica dos sistemas da Capes. É um serviço complementar ao de UX e que faz fronteira com aquele executado pela equipe de desenvolvimento de software em sentido estrito.

### **3. Macroprocesso de Desenvolvimento (Fases do Projeto)**

O modelo de projeto da Capes traz uma base comum para todos os projetos de desenvolvimento de produtos.

Para tanto são conduzidos a partir de um conjunto de etapas básicas independentemente dos objetivos finais.

As etapas funcionam de modo a complementar o modelo de projeto, ou seja, o resultado de cada processo contribui para garantir a qualidade do processo como um todo.

O presente modelo estabelece a metodologia de desenvolvimento de sistemas ágil da CAPES, a qual estrutura os processos do ciclo de vida de software.

Este guia está estruturado de acordo com os processos redesenhados, começando com a apresentação do macroprocesso (Figura 2), seguido por uma breve apresentação do escopo e das principais atividades de cada um dos processos. Este guia será estruturado em três partes: a primeira apresentará os processos relacionados à execução de projetos de desenvolvimento e manutenção de novos projetos ; a segunda demonstrará as atividades relacionadas à manutenção Evolutiva Pontual; a terceira exibirá as atividades relacionadas as soluções alternativas.

Os processos terão um breve descritivo, sendo apresentadas as principais atividades em uma tabela contendo os responsáveis pelo processo, os profissionais envolvidos, as ferramentas utilizadas e as entradas e saídas do processo.

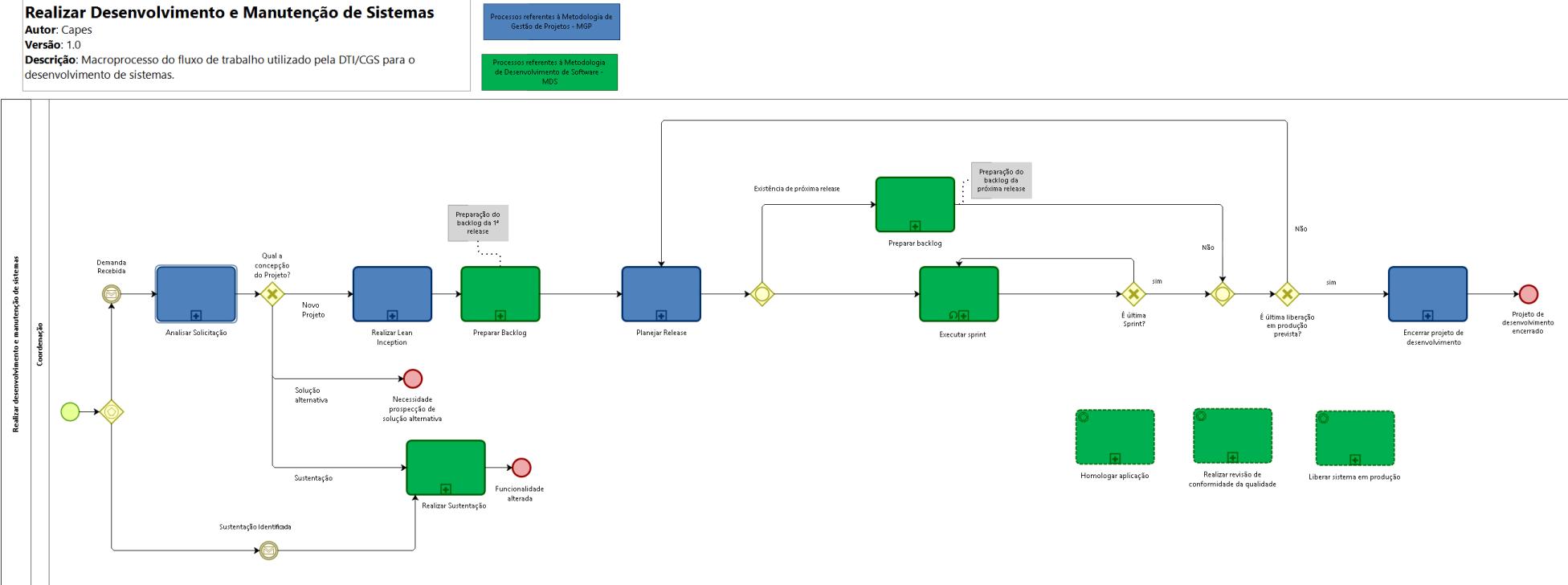


Figura 2 – Macroprocesso da MDS

## 4. Projetos de Sistemas da Capes

Os projetos de sistemas da Capes consistem esforços temporários e com finalidade clara, executados com o objetivo de criar um novo sistema ou módulo ou de realizar uma grande manutenção em um sistema já existente. Uma evolução pontual pode ser tratada de forma isolada, sem a abertura de um projeto, pois caracterizam como uma sustentação de sistemas.

### 4.1. Analisar a solicitação

Este processo visa identificar o tipo da demanda para posterior encaminhamento de acordo com suas especificidades.

#### 4.1.1. Descrição e atividades do processo

Ao receber uma solicitação o Gerente de Projetos da CAPES, junto ao PMO, deverá analisar a concepção da demanda e verificar o seu enquadramento em Sustentação, Novo Projeto ou Solução Alternativa. No caso de Novos projetos, deve realizar as seguintes atividades:

- Entender o escopo do projeto;
- Preparar o material de apoio para a Concepção do projeto;
- Preparar o material de contextualização;
- Realizar o agendamento para a concepção do projeto

<b>Processo</b>	Analisar a solicitação
<b>Responsável</b>	Gerente de Projetos
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"><li>• Gerente de projeto;</li><li>• PMO</li><li>• Product Owner.</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Demanda encaminhada para a DTI</li></ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"><li>• Redmine;</li><li>• CATI;</li><li>• E-mail;</li><li>• Memorando.</li></ul>
<b>Saídas</b>	<ul style="list-style-type: none"><li>• Declaração de Escopo Preliminar;</li><li>• Apresentação da dinâmica da Lean Inception;</li><li>• Material de Contextualização.</li></ul>

## 4.2. Realizar Lean Inception

Processo de Lean Inception, envolve a definição do produto; identificação dos objetivos do produto, identificação das personas, identificação e avaliação das macrofuncionalidades e planejamento do roadmap de release.

### 4.2.1. Descrição e atividades do processo

Caso seja identificado que a demanda se enquadra como projeto de desenvolvimento ou manutenção, deverá ser realizada a Lean Inception, a partir das seguintes atividades:

- Apresentar dinâmica;
- Apresentar a dinâmica de execução do projeto;
- Consolidar Visão do produto;
- Identificar os interessados pelo projeto;
- Identificar as macrofuncionalidades;
- Apresentar critérios de priorização;
- Elaborar Rodmap de Releases;
- Cadastrar o backlog do produto;
- Identificar requisitos não funcionais;
- Formalizar a equipe de arquitetura da CAPES.

<b>Processo</b>	Realizar Concepção do Produto
<b>Responsável</b>	Gerente de Projetos PMO Product Owner
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"><li>• Gerente de projetos;</li><li>• PMO;</li><li>• Analista de Infraestrutura;</li><li>• Administrador de Dados;</li><li>• Arquiteto</li><li>• Analista de Qualidade</li><li>• Product Owner;</li><li>• Analista de Negócio;</li><li>• Designer UX;</li><li>• Designer UI</li><li>• Analista Programador.</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Declaração de Escopo Preliminar</li></ul>

<b>Ferramentas</b>	<ul style="list-style-type: none"> <li>• Redmine;</li> </ul>
<b>Saídas</b>	<ul style="list-style-type: none"> <li>• Documento de Arquitetura de Sistemas;</li> <li>• Roadmap de releases;</li> <li>• Documento de regras de negócio;</li> <li>• Backlog do Produto;</li> <li>• Documentação online.</li> </ul>

### 4.3. Preparar backlog

Processo de preparação do backlog, que envolve a priorização dos itens do backlog do projeto e o detalhamento desses itens em Histórias de Usuário.

#### 4.3.1. Descrição e atividades do processo

Na reunião de planejamento da release, os itens do backlog do projeto devem ser priorizados e detalhados de tornar viável a execução do projeto. Para isso, deve-se executar as seguintes atividades:

- Conduzir a preparação do backlog;
- Consolidar alterações no backlog do produto;
- Dividir itens grandes do backlog do produto em Histórias de Usuário;
- Escrever critérios de aceitação;
- Conferir se os critérios de aceitação viabilizam a execução dos testes;
- Atualizar documento de regras de negócio;
- Realizar suporte na avaliação tecnológica dos itens do backlog;
- Realizar formalização com Product Owner.

<b>Processo</b>	Preparar backlog
<b>Responsável</b>	Gerente de Projetos Analista de Negócio Product Owner
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"> <li>• Gerente de projetos;</li> <li>• Product Owner;</li> <li>• Analista de Negócios;</li> <li>• Analista Programador;</li> <li>• Analista de Qualidade</li> </ul>
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• Backlog do projeto cadastrado no Redmine;</li> <li>• Rodmaps de Releases;</li> </ul>

	<ul style="list-style-type: none"> <li>• Documento de regras e negócio;</li> <li>• Documentação online;</li> </ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"> <li>• Redmine;</li> </ul>
<b>Saídas</b>	<ul style="list-style-type: none"> <li>• Memória de reunião;</li> <li>• Backlog do projeto atualizado;</li> <li>• Documento de regras de negócio atualizado;</li> <li>• Histórias de Usário criadas;</li> <li>• Critérios de aceitação criados</li> </ul>

## 4.4. Planejar Realease

Processo de planejamento de uma release que envolve a definição dos itens do backlog que serão inseridos na release.

### 4.4.1. Descrição e atividades do processo

Após a preparação do backlog e ao término do desenvolvimento de cada release o Gerente de Projetos deve levar em consideração o backlog do projeto e levantar junto ao demandante quais itens de backlog deverão compor a release. Além disso, deve avaliar os impactos da programação de entregas. O Gerente de Projetos deve conduzir reuniões com todas as equipes envolvidas de modo a garantir a disponibilidade adequada dos profissionais ao atendimento das demandas da release. Para isso, deve-se executar as seguintes atividades

- Discutir o objetivo e prazos negociais da release;
- Definir itens de backlog prioritários para a release;
- Conduzir o planejamento de release;
- Atualizar o roadmap de releases;
- Registrar necessidade de elaborar modelo de banco de dados e/ou scripts de banco de dados.

<b>Processo</b>	Planejar Release
<b>Responsável</b>	Gerente de Projetos
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"> <li>• Product Owner;</li> <li>• Gerente de Projetos;</li> <li>• Arquiteto;</li> <li>• Analista de Qualidade;</li> <li>• Analista de Negócio;</li> <li>• Desenvolvedor.</li> </ul>

<b>Entradas</b>	<ul style="list-style-type: none"> <li>• Backlog do projeto atualizado;</li> <li>• Documento de regras de negócio atualizado;</li> <li>• Histórias de usuário criadas;</li> <li>• Critérios de aceitação criadas;</li> </ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"> <li>• Redmine;</li> </ul>
<b>Saídas</b>	<ul style="list-style-type: none"> <li>• Backlog da release;</li> </ul>

## 4.5. Executar sprint

Processo dinâmico da execução de sprints aplicado ao desenvolvimento e manutenção de sistemas.

### 4.5.1. Descrição e atividades do processo

As sprints representam a etapa de execução do projeto, sendo realizadas de forma iterativa e sequencial, com um tamanho fixo e definido. Desta forma, cada release pode conter uma ou mais sprints, dependendo do seu tamanho e da realidade do projeto. A execução das sprints é acompanhada pelo Gerente do Projeto, que deve garantir a execução das seguintes atividades:

- Planejar a Sprint.
- Mapear comportamento das telas e gerar protótipo;
- Realizar o mapeamento de cenário de testes;
- Gerar HTML e CSS das telas;
- Gerar release com funcionalidades passíveis de teste a cada três dias;
- Modelar o banco de dados;
- Realizar teste funcional com base no escopo da Sprint;
- Reunião de review;
- Reunião de retrospectiva.

QA: Realiza testes automatizados com base no escopo da Sprint. As sprints são realizadas em sequência até que se conclua o desenvolvimento de uma release.

<b>Processo</b>	Executar Sprint
<b>Responsável</b>	Gerente de Projetos
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"> <li>• Product Owner;</li> <li>• Gerente de Projetos;</li> <li>• Arquiteto;</li> <li>• Analista de Qualidade;</li> <li>• Analista de Negócio;</li> </ul>

	<ul style="list-style-type: none"><li>• Desenvolvedor;</li><li>• Designer.</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Histórias de Usuário;</li><li>• Critérios de aceitação;</li><li>• Backlog da Sprint definido e autorizado para execução;</li><li>• Regras de negócio atualizadas;</li></ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"><li>• Redmine;</li><li>• Repositório – SVN;</li><li>• Axure;</li></ul>
<b>Saídas</b>	<ul style="list-style-type: none"><li>• Backlog da sprint atualizado;</li><li>• Protótipos de telas navegáveis;</li><li>• Regras de negócio atualizadas;</li><li>• Testes de aceitação elaborados;</li><li>• Modelo de Banco de Dados (conceitual e físico);</li><li>• Scripts de banco de dados;</li><li>• Código da aplicação – front-end;</li><li>• Código da aplicação – back-end;</li><li>• Componentes;</li><li>• Serviços e microsserviços;</li><li>• Relatório de qualidade (evidências de testes unitário e integração);</li><li>• Documento de arquitetura do sistema atualizado;</li><li>• Memória de reunião de Review.</li></ul>

## 4.6. Homologar aplicação

Processo dinâmico de Homologar aplicação como premissa de avaliar uma versão preliminar do sistema. Essa versão deve ser testada conforme as expectativas da CAPES para verificar a eficácia do software como solução.

### 4.6.1. Descrição e atividades do processo

A homologação é a comprovação, pelo cliente e demais partes interessadas, de que o produto resultante do projeto de software atende aos critérios de aceite previamente estabelecidos com o cliente. Inclui elementos de verificação e de validação do produto todo ou de partes do produto selecionadas em comum acordo com o cliente e tem como meta principal a obtenção do aceite do produto.

Tal atividade só deve começar depois que todos os outros testes forem executados, e os erros encontrados forem corrigidos ou aceitos. A homologação da aplicação é acompanhada pelo Gerente do Projeto, que deve garantir a execução das seguintes atividades

- Atestar se o software funciona da forma esperada pelo cliente no ambiente proposto;
- Identificar oportunidades de correção e aplicar os ajustes correlatos à aplicação;
- Obter o aceite do produto de software pelo cliente;
- Registrar sugestões para as próximas versões da aplicação.

<b>Processo</b>	Homologar Aplicação
<b>Responsável</b>	Gerente de Projetos
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"><li>• Product Owner;</li><li>• Gerente de Projetos;</li><li>• Arquiteto;</li><li>• Analista de Negócio;</li><li>• Desenvolvedor;</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Especificação de Requisitos</li><li>• Histórias de Usuário</li></ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"><li>• Redmine;</li><li>• Repositório – SVN</li></ul>
<b>Saídas</b>	<ul style="list-style-type: none"><li>• Redmine atualizado</li></ul>

## 4.7. Realizar revisão de conformidade da qualidade

Processo dinâmico de avaliar a conformidade dos produtos entregues, bem como garantir a qualidade do processo de software no contexto dos Projetos e Sustentação dos sistemas da Capes.

### 4.7.1. Descrição e atividades do processo

Ao término da execução de cada Sprint dos projetos ou sustentação dos sistemas o Gerente de projetos, de acordo com os serviços executados, encaminha para avaliação dos representantes de cada área para realizar a avaliação da qualidade dos serviços executados, garantindo a habilitação necessária dos sistemas da Capes. Após isto, as entregas são submetidos a área da qualidade para avaliar se o processo está em conformidade aos padrões e normas internas previstas.

A revisão da conformidade da qualidade deve garantir a execução das seguintes atividades:

- Identificar chamados e linha de serviço;

- Conferir a conformidade da documentação em relação aos padrões do Capes;
- Analisar se os artefatos estão dentro o que determina o catálogo de serviço;
- Validar o escopo dos artefatos;
- Conferir a conformidade da documentação em relação aos padrões do Capes;
- Emitir relatório de checklist de conformidade.

<b>Processo</b>	Realizar revisão de conformidade da qualidade
<b>Responsável</b>	Analista de Qualidade
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"><li>• Scrum Master;</li><li>• Time Scrum;</li><li>• Analista de Qualidade;</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Artefatos previstos na lista do Catálogo de Serviços;</li><li>• Catálogo de Serviços;</li><li>• Padrão de Nomenclatura da Capes;</li><li>• Padrão de Caminho da Capes;</li><li>• Templates Padronizados;</li></ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"><li>• Portal Gestão;</li><li>• Repositório – SVN;</li></ul>
<b>Saídas</b>	<ul style="list-style-type: none"><li>• Checklist de conformidade</li></ul>

## 4.8. Liberar sistema em produção

Processo dinâmico de realização da liberação de sistemas em produção no contexto de desenvolvimento e manutenção de sistemas da CAPES.

### 4.8.1. Descrição e atividades do processo

Ao final da release, o Gerente de Projeto tomará a decisão de liberar ou não o sistema em produção, tendo como principal critério a necessidade do negócio e a adequação da release, ou pode se utilizar uma versão do sistema realizada na manutenção corretiva ou manutenção evolutiva pontual, sendo assim o sistema deverá ser liberado em produção pela equipe técnica da CAPES. Para isso, deve-se executar as seguintes atividades:

- Avaliar versão para liberar em produção;
- Elaborar GMUD;
- Solicitar criação de esquemas de banco de dados;
- Solicitar persistir scripts de banco de dados;

- Solicitar configuração do ambiente;
- Disponibilizar versão para produção;
- Realizar deploy da versão em ambiente de produção;
- Acompanhar resultado do deploy.

<b>Processo</b>	Liberar sistema em produção
<b>Responsável</b>	Gerente de Projetos Product Owner
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"> <li>• Product Owner;</li> <li>• Gerente de Projetos;</li> <li>• Analista de infraestrutura;</li> <li>• Administrador de Dados;</li> <li>• Analista de infraestrutura.</li> </ul>
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• Backlog da sprint atualizado;</li> <li>• Modelo de Banco de Dados (conceitual e físico);</li> <li>• Scripts de banco de dados;</li> <li>• Código da aplicação – front-end;</li> <li>• Código da aplicação – back-end;</li> <li>• Componentes;</li> <li>• Serviços e microsserviços;</li> <li>• Relatório de qualidade (evidências de testes unitário e integração);</li> <li>• Documento de arquitetura do sistema atualizado;</li> <li>• Memória de reunião de Review;</li> </ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"> <li>• Redmine</li> </ul>
<b>Saídas</b>	<ul style="list-style-type: none"> <li>• Sistema disponível em produção.</li> </ul>

## 4.9. Encerrar projeto de desenvolvimento

Processo dinâmico que finaliza o projeto de desenvolvimento ou manutenção por meio da comunicação do encerramento aos envolvidos.

### 4.9.1. Descrição e atividades do processo

Após a liberação da última versão do sistema em produção, o Gerente do projeto deverá encerrar o projeto de desenvolvimento, em conjunto com o Product Owner da CAPES. Para isso, deve-se executar as seguintes

atividades:

- Checar status do projeto nos sistemas;
- Elaborar "Termo de encerramento do Projeto";
- Verificar termo de encerramento;
- Validar termo de encerramento;

<b>Processo</b>	Encerrar projeto de desenvolvimento
<b>Responsável</b>	Gerente de Projetos PMO
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"><li>• Product Owner;</li><li>• Gerente de Projetos;</li></ul>
<b>Entradas</b>	<ul style="list-style-type: none"><li>• Sistema liberado em produção;</li><li>• Termo de Encerramento de Projeto</li></ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"><li>• Redmine;</li></ul>
<b>Saídas</b>	<ul style="list-style-type: none"><li>• Termo de encerramento de projeto.</li></ul>

## 5. Sustentação

Evolução em sistemas são tratados como manutenções evolutivas pontuais quando não se enquadram nos critérios de definição de projetos. Representam, portanto, melhorias ou adaptações em sistemas já existentes não possuem prazos definidos. Isso significa que as demandas de evolução pontual não são gerenciadas como projetos, e por isso possuem procedimentos específicos, com reduzido esforço de gerenciamento. Dessa forma, é possível que cada coordenação atenda às demandas pontuais mais livremente, o que tende a aumentar a velocidade das entregas realizadas pela DTI aos usuários.

### 5.1. Realizar Sustentação

Processo que apresenta a dinâmica de realização de uma manutenção evolutiva pontual nos sistemas da CAPES.

#### 5.1.1. Descrição e atividades do processo

Sempre que uma demanda de sustentação for identificada, o Gerente de Projeto deve acionar o Time para que o item seja incluído no backlog do Kanban de manutenção. Cada item incluído deverá ser priorizado pelo Product owner e dividido em tarefas menores para que seja executado de acordo com o nível de prioridade atribuído e com os outros itens do Kanban. Após a realização de desenvolvimento e testes, a evolução será avaliada e liberada em produção. As seguintes atividades são executadas até que uma evolução seja liberada em produção:

- Solicitar evolução em sistema;
- Priorizar itens do backlog do Kanban;
- Definir classes dos serviços a serem executados;
- Inserir item de evolução pontual no Kanban;
- Preparar backlog;
- Quebrar item de backlog em partes pequenas;
- Mapear comportamento das telas e gerar protótipo;
- Realizar o mapeamento de cenário de testes;
- Gerar HTML e CSS das telas;
- Gerar entrega com funcionalidades passíveis de teste;
- Modelar o banco de dados;
- Realizar teste funcional com base no escopo da Sprint;
- Implementar em ambiente de homologação;
- Realizar avaliação técnica de qualidade;
- Liberar em produção;
- Atualizar Kanban.

<b>Processo</b>	Realizar manutenção evolutiva pontual
<b>Responsável</b>	Gerente de Projetos
<b>Perfis e áreas envolvidas</b>	<ul style="list-style-type: none"> <li>• Product Owner;</li> <li>• Gerente de Projetos;</li> <li>• Arquiteto;</li> <li>• Desenvolvedor;</li> <li>• Analista de qualidade;</li> <li>• Analista de negócio;</li> </ul>
<b>Entradas</b>	<ul style="list-style-type: none"> <li>• Evolução pontual identificada.</li> </ul>
<b>Ferramentas</b>	<ul style="list-style-type: none"> <li>• Redmine;</li> <li>• Kanban;</li> <li>• Axure.</li> </ul>
<b>Saídas</b>	<ul style="list-style-type: none"> <li>• Protótipos de telas navegáveis;</li> <li>• Regras de negócio atualizadas;</li> <li>• Testes de aceitação elaborados;</li> <li>• Templates de tela criados;</li> <li>• Modelo de Banco de Dados (conceitual e físico);</li> <li>• Scripts de banco de dados;</li> <li>• Código da aplicação – front-end;</li> </ul>

	<ul style="list-style-type: none"><li>• Código da aplicação – back-end;</li><li>• Componentes;</li><li>• Serviços e microsserviços;</li><li>• Relatório de qualidade (evidências de testes unitário e integração);</li><li>• Documentação técnica;</li><li>• Documento de arquitetura do sistema atualizado;</li><li>• Kanban atualizado.</li></ul>
--	---

## 6. Necessidade de prospecção de solução alternativa

A necessidade de prospecção de solução alternativa é uma chamada para a análise de eventuais produtos que necessitam de avaliação técnica da DTI e que sejam produtos do âmbito de mercado. Sendo assim, ao aderir o processo de aquisição de software às normas e padrões, diversos benefícios podem ser alcançados, os quais se destacam: a garantia que os processos deverão seguir métodos com expectativas de segurança e qualidade bem definidas e a natural redução de erros; a definição de padrões baseados em modelo de maturidade e nas melhores práticas para a avaliação da maturidade dos processos de software dos terceiros; a utilização de procedimentos focados na melhoria contínua dos processos de contratação; a definição de níveis de serviço (SLA) baseados em parâmetros da maturidade dos processos e da qualidade dos produtos; a realização de avaliações periódicas nos processos dos fornecedores; a criação de um banco de fornecedores avaliados; e a melhoria contínua dos procedimentos de contratação e dos processos de software dos terceiros (WEBER et al, 2001).

## 7. Referências

- <http://www.scrumguides.org>;
- <http://www.manifestoagil.org>;
- Livro “Programação Extrema (XP) Explicada” – Kent Beck;
- PDSBCÁgil - Processo Ágil de Desenvolvimento de SW do BACEN.
- WEBER, K.C., CAVALCANTI DA ROCHA, A.R., MALDONADO, J.C. Qualidade de software: teoria e prática, vol. III, Prentice Hall, 2001.
- PRESSMAN, R. Engenharia de Software: Uma Abordagem Profissional. 7<sup>a</sup> Ed., São Paulo: McGraw-Hill, 2011.

## **ANEXO I - Definições, Acrônimos e Abreviações**

- Backlog – Backlog refere-se a um log de acumulação de trabalho num determinado intervalo de tempo. Backlog é uma espécie de estoque de folhas de requisições/encomendas relativas a produtos ainda não produzidos. Grosso modo, backlog é uma "pilha de pedidos" em espera.. O Product Backlog pode crescer muito e com o tempo pode se tornar muito grande e desorganizado. Ele poderá apresentar histórias que já foram finalizadas e não necessitam mais serem exibidas, assim como histórias que, com a evolução do projeto, se tornam obsoletas e devem ser excluídas.
- CAPES – Coordenação de Aperfeiçoamento de Pessoal de Nível Superior.
- CATI – Central de Atendimento de TI.
- Critérios de aceitação - Lista de itens de negócio que expressam as formas de usar a funcionalidade implementada em uma história, bem como as regras e detalhamentos que validam se a história foi implementada de acordo com o que o Product Owner esperava e se está aceitável para o uso em produção.
- E-Ping - A arquitetura de Padrões de Interoperabilidade (ePing) define um conjunto mínimo de premissas, políticas e especificações técnicas que regulamentam a utilização da Tecnologia de Informação e Comunicação no Governo Federal, estabelecendo as condições de interação com os demais Poderes e esferas de governo e com a sociedade em geral.
- E-mag - Modelo de Acessibilidade em Governo Eletrônico (eMAG) consiste em um conjunto de recomendações a ser considerado para que o processo de acessibilidade dos sítios e portais do governo brasileiro seja conduzido de forma padronizada e de fácil implementação.
- Deploy – É a instalação da sua aplicação em um servidor de aplicações.
  - Deploy de Release - Corresponde a versão do build gerado na homologação, testado e aprovado pela área gestora que será atualizado em produção. A versão para produção deve estar devidamente aprovada por meio do Aceite Formal.
- Done (Pronto) – Estado esperado para a aceitação do produto criado em uma Sprint, que esteja rigorosamente de acordo com os requisitos definidos pela equipe para a aceitação do produto feito.
- Frameworks – Um framework em desenvolvimento de software, é uma abstração que une códigos comuns entre vários projetos de software provendo uma funcionalidade genérica. Um framework pode atingir uma funcionalidade específica, por configuração, durante a programação de uma aplicação
- História – Breve descrição de algo que o sistema deve fazer, com clareza suficiente para ser entendida por usuários e desenvolvedores, escrita de maneira simples, objetiva, eficaz e eficiente.

Não deve conter detalhes, apenas informar: o que se deverá realizar, o motivo para a execução e para quem está sendo escrita.

- Lista de necessidades da área solicitante – Lista de itens que a área gestora aponta como necessárias e que gera um projeto de implementação de software. No Workshop de Requisitos ela será tratada, priorizada e a partir dela será gerado o Product Backlog.
- Produto - Conjunto de programas de computador (código fonte, testes automatizados, scripts de banco de dados, testes e operação), procedimentos, documentação e dados associados, que traz o resultado esperado de uma implementação que pode ser parte ou o todo de uma aplicação e atende às necessidades do Product Owner. É o principal objetivo do Projeto.
- PMBOK - O guia Project Management Body of Knowledge é um conjunto de práticas na gestão de projetos organizado pelo instituto PMI e é considerado a base do conhecimento sobre gestão de projetos por profissionais da área.
- Release – É o intervalo entre dois objetivos primários do desenvolvimento do projeto. O objetivo primário é entender com maior clareza, o escopo e o planejamento do projeto com vistas a produzir o produto. Para cada Release são definidas metas a serem atingidas, artefatos que deverão ser concluídos e tomadas de decisões.
- RUP - O **RUP**, abreviação de **Rational Unified Process** (ou Processo Unificado da Rational), é um processo proprietário de Engenharia de software criado pela Rational Software Corporation, adquirida pela IBM, ganhando um novo nome **IRUP** que agora é uma abreviação de **IBM Rational Unified Process** e tornando-se uma *brand* na área de Software, fornecendo técnicas a serem seguidas pelos membros da equipe de desenvolvimento de software com o objetivo de aumentar a sua produtividade no processo de desenvolvimento.
- Sprint – Ciclos específicos de tempo, curtos, onde serão executadas as histórias de usuário divididas em tarefas com o objetivo de gerar incremento da aplicação, potencialmente entregável.
- Stakeholders – São os envolvidos ou interessados no software que o projeto trará, sejam pessoas, um departamento ou mesmo clientes externos. Poderão atuar junto à equipe ou apenas ter interesse e consultar informações.
- SVN – Subversion para Windows – Aplicativo para controle de versão.
- Tarefas – Divisão de uma história em atividades menores que devem ter como características: serem específicas, mensuráveis, capazes de serem executadas pela equipe com sua qualificação, relevantes para o usuário e realizável dentro de um espaço de tempo de no máximo de 8h, preferencialmente.

- Timebox – Técnica de gestão de tempo comum no planejamento de projetos, que fraciona o cronograma em períodos separados, com duração e entregáveis pré-determinados. Tempo de duração do projeto, dos Releases, das Sprints, das reuniões e da execução de tarefas. *Fatia de tempo fixada e usada para garantir o prazo de entrega e facilitar o planejamento.* [WIKIPEDIA].
- Velocidade da Equipe – medida da quantidade de trabalho que uma equipe consegue executar em um período de tempo determinado. É variável e específica para cada equipe de projeto. Só pode ser identificada após algumas semanas de trabalho, quando a equipe se torna mais integrada e inteirada do negócio do sistema e do processo de desenvolvimento.
- Versão do produto – É a versão da aplicação que deverá ser testada, homologada e colocada em produção. Ela deve ser preparada antes da promoção de ambientes ser feita
- Visão do Produto – É a visão do que o produto vai entregar de benefícios para o cliente. Ela deverá ser clara e conter as seguintes informações: nome, categoria, para quem está sendo desenvolvida, a necessidade da área gestora que o produto irá atender, que benefícios trará e, se houver concorrente ou legado, informar o diferencial do já existente.

## ANEXO II – Produtos de Trabalho

- Atas de Reuniões - São documentos utilizados para registrar as ocorrências que ocorram durante o desenvolvimento do projeto, independente da metodologia de desenvolvimento escolhida.
- Cenários de Teste – Roteiro em tópicos que descreve “como” será realizado o teste. Contempla os cenários previstos na história incluindo as validações de regras de negócio e obrigatoriedade. Os cenários devem ser registrados em ferramenta específica considerando:
  - Identificação do cenário;
  - O que deve ser feito;
  - Dados de Entrada;
  - Resultado Esperado.
- Chamado de Atendimento - Chamado registrado pelo usuário gestor para atendimento de demandas da sua área.
- Código implementado - Corresponde à codificação do sistema, ou parte deste, em linguagem de programação específica. O código para estar completo deve exercer a função que fora especificada e atender aos requisitos funcionais e não funcionais descritos na documentação do sistema.
- Cronograma - É o sequenciamento das atividades do projeto conforme acordo firmado com o gestor do sistema. No cronograma é contemplado o marco de entrega, as atividades com a quantidade de horas e início/fim, alocação de recursos e os registros de trabalho em cada uma das tarefas. Para desenvolvimento ágil, os marcos de entrega devem ser registrado ao final dos Releases e os registros das tarefas dentro das Sprints.
- Documento de Arquitetura - Descreve a arquitetura do projeto. Utilizado apenas para projetos que não utilizam integralmente a arquitetura definida pela Equipe de Arquitetura da CAPES. Pode ser utilizado independente da metodologia de desenvolvimento escolhida.
- Documento de Regras de Negócio – Tem o objetivo de documentar os requisitos funcionais e não funcionais sistematizados. Para melhor organização e entendimento, recomenda-se que as regras sejam agrupadas por assunto ou funcionalidade. As regras de negócio que são comum a mais de um assunto ou funcionalidade devem ter a mesma referência para possibilitar a rastreabilidade.
- Documento de Visão - Este artefato delimita o escopo do projeto a ser realizado. Auxilia na análise da solução, na identificação de requisitos, na expectativa de prazo, custo e recursos consumidos pelo projeto. O template proposto para o novo Documento de Visão recebe em sessão específica e opcional, o Product Backlog, para os casos de desenvolvimento ágil.

- Evidência de Teste - Artefato utilizado para descrever o resultado de testes conforme os Cenários de Teste. Contém os seguintes atributos:
  - Ambiente de Teste;
  - Técnico da realização dos testes;
  - Data e hora do teste;
  - Cenário de testes envolvidos;
  - Registro de anomalias e/erros do sistema;
  - Parecer do teste.

Este artefato é gerado na ferramenta utilizada para criação de cenários de teste e registro de evidência, que possui integração com a ferramenta de bug tracking adotada pela CAPES.

- Formulário de Solicitação de Mudança - Documento que descreve qual será a alteração, entrada ou saída de um ou mais ativos de informação do ambiente de produção, contendo um planejamento das ações a serem executadas na mudança.
- Kanban – É uma estrutura popular, usada para realizar o desenvolvimento de software ágil e comunica em tempo real e de forma transparente o estado atual do trabalho que está sendo feito. Através de colunas que identificam: o que foi programado para ser feito, o que está em execução, o que foi concluído e o que está pendente de recursos ou de terceiros, pode-se verificar em que situação está cada tarefa de uma história dentro de uma Sprint.
- Product Backlog – Lista priorizada das necessidades do cliente que precisam ser produzidas para que a Visão do Produto em construção seja atingida. É um artefato vivo e deve ser atualizado ao final dos Releases e das Sprints, bem como toda vez que houver mudança relevante que altere seu conteúdo. Deve ser registrado em seção específica no Documento de Visão
- Protótipo – Desenho ou imagem da interface a ser utilizada em uma funcionalidade da aplicação. O objetivo é tornar mais claros, os conceitos que possam ser abstratos se forem apenas registrados por escrito. Pretende também facilitar a internalização de novas ideias e a visão da comunicação entre as telas, de forma que a equipe possa identificar conflitos, caso haja, e tratá-los a tempo. Recomenda-se que o processo seja feito em ferramenta determinada pela equipe ou em prototipagem de papel (descartável e envolve a criação de desenhos simples de uma interface, feitos à mão), sem complexidade de desenho ou gasto de tempo que possa impactar no cronograma.
- Produto de Software – Conjunto de programas de computador (código fonte, testes automatizados, scripts de banco de dados, testes e operação), procedimentos, documentação e dados associados.

- Plano de Implantação - Garantir que o sistema alcance seus usuários com sucesso, fornecendo uma agenda detalhada de eventos, pessoas responsáveis e dependências necessárias para garantir a mudança bem-sucedida para o novo sistema. Este documento é gerado nas atividades da MGP.
- Plano de Migração de Dados - Este documento consiste em descrever a migração de dados que será necessária para o projeto, independente da metodologia de desenvolvimento escolhida.
- Protótipo - Consiste na simulação das telas de um determinado sistema. Sua função principal é auxiliar na elucidação dos requisitos funcionais junto ao Product Owner, uma vez que permite visualizar a aplicação ainda não produzida. A prototipação ajuda na tomada de decisões e consequentemente reduz as mudanças durante o processo de homologação do produto.
- Relatório de Teste - Este documento tem como finalidade mostrar informações resumidas determinadas pela análise de um ou mais registros de teste, que permitem uma avaliação relativamente detalhada da qualidade dos Itens de teste-alvo e do status de cada esforço de teste. Tem como finalidade apresentar os resultados obtidos com a execução das funcionalidades dos sistemas. Organiza e apresenta uma análise resumida dos resultados e as principais medidas do teste para revisão e avaliação, geralmente executadas pelos principais envolvidos na questão de qualidade. Este sumário apresenta indicadores gerais sobre a qualidade dos testes da Sprint e fornece recomendações para um futuro esforço de teste
- Release Backlog – Parte do Product Backlog destacada para ser executada em um determinado Release. É um subconjunto do Product Backlog.
- Sprint Backlog – Parte do Release Backlog destacada para ser executada em uma determinada Sprint. É um subconjunto do Release Backlog.
- Tipos de Teste – Tipos de testes disponíveis que garante uma maior confiabilidade e qualidade aos sistemas alvo dos testes.

## Ferramentas

## Referências

- Fonte: <http://tutano.trampos.co/16410-guia-de-profissoes-scrum-master/>
- BOEG, Jesper. Kanban em 10 Passos: Otimizando o fluxo de trabalho em sistemas de entrega de software. Leonardo Galvão. InfoQ Brasil, C4Media Inc., 2014.



coordenação de aperfeiçoamento de pessoal de nível superior

### ENCARTE D - ORÇAMENTO ESTIMATIVO

Grupo	Itens	Serviço	Qtd	Valor unitário mensal (R\$)	Valor unitário mensal (R\$)	Valor total anual (R\$)
1	1	Analista Programador Sênior	28 Postos de trabalho			
	2	Analista Programador Pleno	18 Postos de trabalho			
	3	Analista Programador Júnior	6 Postos de trabalho			
2	1	Mensuração de Sistemas	24.000 Pontos de Função			
						Valor total anual estimado:

Analista Programador Sênior		
Jornada		40 h
Turno		Diurno
I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)		
Salário base		
SUB-TOTAL DA REMUNERAÇÃO (R\$)		
TOTAL DA REMUNERAÇÃO (R\$)		
II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)		
GRUPO A - ENCARGOS		
A.01 SEGURIDADE SOCIAL		
A.02 FGTS		
A.03 SESI/SESC		
A.04 SENAI/SENAC		
A.05 INCRA		
A.06 SEBRAE		
A.07 Salário Educação		
A.08 Riscos Ambientais do Trabalho – RAT x FAP		

<b><i>Analista Programador Sênior</i></b>		
<b>TOTAL - GRUPO A - ENCARGOS</b>		
<b>GRUPO B</b>		
B.01 13º Salário		
B.02 Férias (sem o abono de 1/3)		
B.03 Aviso Prévio Trabalhado		
B.04 Auxílio Doença		
B.05 Acidente de Trabalho		
B.06 Faltas Legais		
B.07 Férias sobre Licença Maternidade		
B.08 Licença Paternidade		
<b>TOTAL - GRUPO B</b>		
<b>GRUPO C</b>		
C.01 Aviso Prévio Indenizado		
C.02 Indenização Adicional		
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)		
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)		
C.05 Abono de Férias - 1/3 constitucional		
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade		
<b>TOTAL - GRUPO C</b>		
<b>GRUPO D</b>		
D.01 Incidência dos encargos do grupo A sobre o grupo B		
<b>TOTAL - GRUPO D</b>		
<b>GRUPO E</b>		
E.01 Incidência do FGTS sobre o aviso prévio indenizado		
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho		
E.03 Incidência de FGTS sobre férias 1/3 constitucional		
<b>TOTAL - GRUPO E</b>		
<b>GRUPO F</b>		
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade		
<b>TOTAL - GRUPO F</b>		
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>		
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>		
<b>III - INSUMOS</b>		
Uniforme		
Auxílio alimentação		
Vale transporte		
Desconto legal sobre transporte (máximo 6% do salário-base)		
<b>TOTAL - INSUMOS (R\$)</b>		
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>		

<b><i>Analista Programador Sênior</i></b>			
<b>IV - LDI E TRIBUTAÇÃO</b>			
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>			
Despesas Administrativas/Operacionais			
Lucro			
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>			
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
INSS			
ISS			
COFINS			
PIS			
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
<b>PREÇO MENSAL PARA 1 (UM) EMPREGADO (R\$)</b>			
<b>Quantidade de profissionais deste perfil</b>			
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>			
<b>PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)</b>			

<b><i>Analista Programador Pleno</i></b>			
Jornada		40 h	
Turno		Diurno	
<b>I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)</b>			
Salário base			
<b>SUB-TOTAL DA REMUNERAÇÃO (R\$)</b>			
<b>TOTAL DA REMUNERAÇÃO (R\$)</b>			
<b>II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)</b>			
<b>GRUPO A - ENCARGOS</b>			
A.01 SEGURIDADE SOCIAL			
A.02 FGTS			
A.03 SESI/SESC			
A.04 SENAI/SENAC			
A.05 INCRA			
A.06 SEBRAE			
A.07 Salário Educação			
A.08 Riscos Ambientais do Trabalho – RAT x FAP			
<b>TOTAL - GRUPO A - ENCARGOS</b>			
<b>GRUPO B</b>			
B.01 13º Salário			
B.02 Férias (sem o abono de 1/3)			
B.03 Aviso Prévio Trabalhado			
B.04 Auxílio Doença			
B.05 Acidente de Trabalho			
B.06 Faltas Legais			

<b>Analista Programador Pleno</b>		
B.07 Férias sobre Licença Maternidade		
B.08 Licença Paternidade		
<b>TOTAL - GRUPO B</b>		
<b>GRUPO C</b>		
C.01 Aviso Prévio Indenizado		
C.02 Indenização Adicional		
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)		
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)		
C.05 Abono de Férias - 1/3 constitucional		
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade		
<b>TOTAL - GRUPO C</b>		
<b>GRUPO D</b>		
D.01 Incidência dos encargos do grupo A sobre o grupo B		
<b>TOTAL - GRUPO D</b>		
<b>GRUPO E</b>		
E.01 Incidência do FGTS sobre o aviso prévio indenizado		
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho		
E.03 Incidência de FGTS sobre férias 1/3 constitucional		
<b>TOTAL - GRUPO E</b>		
<b>GRUPO F</b>		
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade		
<b>TOTAL - GRUPO F</b>		
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>		
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>		
<b>III - INSUMOS</b>		
Uniforme		
Auxílio alimentação		
Vale transporte		
Desconto legal sobre transporte (máximo 6% do salário-base)		
<b>TOTAL - INSUMOS (R\$)</b>		
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>		
<b>IV - LDI E TRIBUTAÇÃO</b>		
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>		
Despesas Administrativas/Operacionais		
Lucro		
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>		
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>		
INSS		

<b><i>Analista Programador Pleno</i></b>			
ISS			
COFINS			
PIS			
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
<b>PREÇO MENSAL PARA 1 (UM) EMPREGADO (R\$)</b>			
<b>Quantidade de profissionais deste perfil</b>			
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>			
<b>PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)</b>			

<b><i>Analista Programador Júnior</i></b>			
Jornada		40 h	
Turno		Diurno	
<b>I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)</b>			
Salário base			
<b>SUB-TOTAL DA REMUNERAÇÃO (R\$)</b>			
<b>TOTAL DA REMUNERAÇÃO (R\$)</b>			
<b>II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)</b>			
<b>GRUPO A - ENCARGOS</b>			
A.01 SEGURIDADE SOCIAL			
A.02 FGTS			
A.03 SESI/SESC			
A.04 SENAI/SENAC			
A.05 INCRA			
A.06 SEBRAE			
A.07 Salário Educação			
A.08 Riscos Ambientais do Trabalho – RAT x FAP			
<b>TOTAL - GRUPO A - ENCARGOS</b>			
<b>GRUPO B</b>			
B.01 13º Salário			
B.02 Férias (sem o abono de 1/3)			
B.03 Aviso Prévio Trabalhado			
B.04 Auxílio Doença			
B.05 Acidente de Trabalho			
B.06 Faltas Legais			
B.07 Férias sobre Licença Maternidade			
B.08 Licença Paternidade			
<b>TOTAL - GRUPO B</b>			
<b>GRUPO C</b>			
C.01 Aviso Prévio Indenizado			
C.02 Indenização Adicional			
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)			
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)			

<b><i>Analista Programador Júnior</i></b>		
C.05 Abono de Férias - 1/3 constitucional		
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade		
<b>TOTAL - GRUPO C</b>		
<b>GRUPO D</b>		
D.01 Incidência dos encargos do grupo A sobre o grupo B		
<b>TOTAL - GRUPO D</b>		
<b>GRUPO E</b>		
E.01 Incidência do FGTS sobre o aviso prévio indenizado		
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho		
E.03 Incidência de FGTS sobre férias 1/3 constitucional		
<b>TOTAL - GRUPO E</b>		
<b>GRUPO F</b>		
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade		
<b>TOTAL - GRUPO F</b>		
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>		
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>		
<b>III - INSUMOS</b>		
Uniforme		
Auxílio alimentação		
Vale transporte		
Desconto legal sobre transporte (máximo 6% do salário-base)		
<b>TOTAL - INSUMOS (R\$)</b>		
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>		
<b>IV - LDI E TRIBUTAÇÃO</b>		
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>		
Despesas Administrativas/Operacionais		
Lucro		
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>		
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>		
INSS		
ISS		
COFINS		
PIS		
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>		
<b>PREÇO MENSAL PARA 1 (UM) EMPREGADO (R\$)</b>		
Quantidade de profissionais deste perfil		
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>		

**Analista Programador Júnior****PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)****Analista Programador Pleno**

Jornada	40 h		
Turno	Diurno		

**I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)**

Salário base			
<b>SUB-TOTAL DA REMUNERAÇÃO (R\$)</b>			
<b>TOTAL DA REMUNERAÇÃO (R\$)</b>			

**II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)****GRUPO A - ENCARGOS**

A.01 SEGURIDADE SOCIAL			
A.02 FGTS			
A.03 SESI/SESC			
A.04 SENAI/SENAC			
A.05 INCRA			
A.06 SEBRAE			
A.07 Salário Educação			
A.08 Riscos Ambientais do Trabalho – RAT x FAP			

**TOTAL - GRUPO A - ENCARGOS****GRUPO B**

B.01 13º Salário			
B.02 Férias (sem o abono de 1/3)			
B.03 Aviso Prévio Trabalhado			
B.04 Auxílio Doença			
B.05 Acidente de Trabalho			
B.06 Faltas Legais			
B.07 Férias sobre Licença Maternidade			
B.08 Licença Paternidade			

**TOTAL - GRUPO B****GRUPO C**

C.01 Aviso Prévio Indenizado			
C.02 Indenização Adicional			
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)			
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)			
C.05 Abono de Férias - 1/3 constitucional			
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade			

**TOTAL - GRUPO C****GRUPO D**

D.01 Incidência dos encargos do grupo A sobre o grupo B			
<b>TOTAL - GRUPO D</b>			
<b>GRUPO E</b>			
E.01 Incidência do FGTS sobre o aviso prévio indenizado			

<b>Analista Programador Pleno</b>			
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho			
E.03 Incidência de FGTS sobre férias 1/3 constitucional			
<b>TOTAL - GRUPO E</b>			
<b>GRUPO F</b>			
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade			
<b>TOTAL - GRUPO F</b>			
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>			
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>			
<b>III - INSUMOS</b>			
Uniforme			
Auxílio alimentação			
Vale transporte			
Desconto legal sobre transporte (máximo 6% do salário-base)			
<b>TOTAL - INSUMOS (R\$)</b>			
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>			
<b>IV - LDI E TRIBUTAÇÃO</b>			
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>			
Despesas Administrativas/Operacionais			
Lucro			
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>			
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
INSS			
ISS			
COFINS			
PIS			
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
<b>PREÇO MENSAL PARA 1 (UM) EMPREGADO (R\$)</b>			
<b>Quantidade de profissionais deste perfil</b>			
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>			
<b>PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)</b>			

1. **OBSERVAÇÕES:**

- 1.1. Remuneração mínima aceitável para profissional com perfil “Analista Programador Java nível Sênior”: **R\$ 9.913,60**.
- 1.2. Remuneração mínima aceitável para o profissional com perfil “Analista Programador Java nível Pleno”: **R\$ 7.315,56**.

1.3. Remuneração mínima aceitável para o profissional com perfil “Analista Programador Java nível Júnior”: **R\$ 4.459,42**.

1.4. Os patamares salariais mínimos previstos nos subitens 1 e 2 serão reajustados nos mesmos termos que os salários dos trabalhadores da categoria previstos na convenção coletiva de trabalho vinculada à proposta comercial da CONTRATADA.

1.5. O quantitativo de postos de trabalho é estimativo e não constitui compromisso de demanda por parte da CONTRATANTE.



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE E - MODELO DE PROPOSTA DE PREÇOS

##### 1. GRUPO 1 - DESENVOLVIMENTO DE SISTEMAS

1.1. Tendo em vista as peculiaridades desta contratação, apresentam-se os esclarecimentos referentes às planilhas estimativas, os quais deverão ser observados pelas licitantes quando da elaboração de suas propostas de preços.

1.2. As licitantes deverão apresentar a proposta de preço sintética na forma do Quadro 1 deste anexo, juntamente com uma planilha de custos e formação de preços para cada perfil profissional, ou seja, uma planilha para o Analista Programador Java nível Sênior, outra para o Analista Programador Java nível Pleno e outra para o Analista Programador Java nível Júnior, utilizando o modelo também apresentado neste encarte.

1.3. Na elaboração das planilhas as licitantes deverão observar convenção coletiva de trabalho, ou outra norma coletiva mais benéfica, aplicável à categoria envolvida na contratação e à qual a licitante esteja obrigada em convenção coletiva de trabalho, ou outra norma coletiva mais benéfica, aplicável à categoria envolvida na contratação e à qual a licitante esteja obrigada.

1.4. O item 1.10, deste documento, contém planilha de exemplo com memória de cálculo explicativa para subsidiar a elaboração da proposta de preço pela licitante. Cabe destacar que valores identificados como exemplificativos e estimativos e demais elementos dependentes de aspectos particulares da empresa (ex.: regime tributário) deverão ser ajustados pela licitante, observados os dispositivos aplicáveis deste edital e a legislação vigente.

1.5. Na hipótese de eventual repactuação do contrato, somente serão considerados os itens previstos nas respectivas planilhas.

1.6. Ainda que, em acordo ou convenção coletiva da categoria, haja previsão de reajuste escalonado de salários, a CONTRATADA aplicará aos salários dos empregados que prestam serviços à CONTRATANTE os mesmos índices concedidos na repactuação contratual, independentemente da data de admissão do empregado nos quadros da CONTRATADA.

1.7. O LDI (Lucros e Despesas Indiretas) constante das planilhas de composição de custos e formação de preços engloba o lucro e as despesas administrativas e operacionais (Acórdão 2.369/2011-TCU-Plenário).

1.8. O item B.03 - Aviso prévio trabalhado será zerado após o primeiro ano de vigência do contrato.

Perfil do Posto de Trabalho	Qdt	Turno/Carga Horária	Valor unitário mensal (R\$)	Valor total anual (R\$)	Valor total anual (R\$)	Valor total para 24 meses (R\$)

Perfil do Posto de Trabalho	Qdt	Turno/Carga Horária	Valor unitário mensal (R\$)	Valor total anual (R\$)	Valor total anual (R\$)	Valor total para 24 meses (R\$)
Analista Programador Java nível Sênior		Diurno (8h)				
Analista Programador Java nível Pleno		Diurno (8h)				
Analista Programador Java nível Júnior		Diurno (8h)				
<b>Total (R\$)</b>						

### 1.9. Modelo de planilha de custos e formação de preços

(nº do item do objeto - perfil profissional)			
Jornada		<b>40 h</b>	
Turno		<b>Diurno</b>	
<b>I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)</b>			
Salário base			<b>R\$</b>
<b>SUB-TOTAL DA REMUNERAÇÃO (R\$)</b>			<b>R\$</b>
<b>TOTAL DA REMUNERAÇÃO (R\$)</b>			<b>R\$</b>
<b>II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)</b>			
<b>GRUPO A - ENCARGOS</b>			
A.01 SEGURIDADE SOCIAL		%	<b>R\$</b>
A.02 FGTS		%	<b>R\$</b>
A.03 SESI/SESC		%	<b>R\$</b>
A.04 SENAI/SENAC		%	<b>R\$</b>
A.05 INCRA		%	<b>R\$</b>
A.06 SEBRAE		%	<b>R\$</b>
A.07 Salário Educação		%	<b>R\$</b>
A.08 Riscos Ambientais do Trabalho – RAT x FAP		%	<b>R\$</b>
<b>TOTAL - GRUPO A - ENCARGOS</b>		%	<b>R\$</b>
<b>GRUPO B</b>			
B.01 13º Salário		%	<b>R\$</b>
B.02 Férias (sem o abono de 1/3)		%	<b>R\$</b>
B.03 Aviso Prévio Trabalhado		%	<b>R\$</b>
B.04 Auxílio Doença		%	<b>R\$</b>
B.05 Acidente de Trabalho		%	<b>R\$</b>
B.06 Faltas Legais		%	<b>R\$</b>
B.07 Férias sobre Licença Maternidade		%	<b>R\$</b>
B.08 Licença Paternidade		%	<b>R\$</b>
<b>TOTAL - GRUPO B</b>		%	<b>R\$</b>

(nº do item do objeto - perfil profissional)			
<b>GRUPO C</b>			
C.01 Aviso Prévio Indenizado		%	R\$
C.02 Indenização Adicional		%	R\$
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)		%	R\$
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)		%	R\$
C.05 Abono de Férias - 1/3 constitucional		%	R\$
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade		%	R\$
<b>TOTAL - GRUPO C</b>		%	<b>R\$</b>
<b>GRUPO D</b>			
D.01 Incidência dos encargos do grupo A sobre o grupo B		%	R\$
<b>TOTAL - GRUPO D</b>		%	<b>R\$</b>
<b>GRUPO E</b>			
E.01 Incidência do FGTS sobre o aviso prévio indenizado		%	R\$
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho		%	R\$
E.03 Incidência de FGTS sobre férias 1/3 constitucional		%	R\$
<b>TOTAL - GRUPO E</b>		%	<b>R\$</b>
<b>GRUPO F</b>			
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade		%	R\$
<b>TOTAL - GRUPO F</b>		%	<b>R\$</b>
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>		%	<b>R\$</b>
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>			<b>R\$</b>
<b>III - INSUMOS</b>			
Uniforme			R\$
Auxílio alimentação			R\$
Vale transporte			R\$
Desconto legal sobre transporte (máximo 6% do salário-base)			R\$
<b>TOTAL - INSUMOS (R\$)</b>			<b>R\$</b>
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>			<b>R\$</b>
<b>IV - LDI E TRIBUTAÇÃO</b>			
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>			
Despesas Administrativas/Operacionais		%	R\$
Lucro		%	R\$
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>		%	<b>R\$</b>
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
INSS		%	R\$
ISS		%	R\$
COFINS		%	R\$

(nº do item do objeto - perfil profissional)			
PIS		%	R\$
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>		%	<b>R\$</b>
<b>PREÇO MENSAL PARA 1 (UM) PROFISSIONAL (POSTO)</b>			<b>R\$</b>
	<b>Quantidade de profissionais deste perfil</b>	<i>qtde</i>	
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>			<b>R\$</b>
<b>PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)</b>			<b>R\$</b>

**1.10. EXEMPLO DE PLANILHA DE CUSTOS E FORMAÇÃO DE PREÇOS COM MEMÓRIA DE CÁLCULO**

(nº do item do objeto - perfil profissional)			
Jornada	<b>40 h</b>		
Turno		<b>Diurno</b>	
<b>I - COMPOSIÇÃO DA REMUNERAÇÃO (R\$)</b>			
Salário base			<b>R\$ 9.836,06</b>
<b>SUB-TOTAL DA REMUNERAÇÃO (R\$)</b>			<b>R\$ 9.836,06</b>
<b>TOTAL DA REMUNERAÇÃO (R\$)</b>			<b>R\$ 9.836,06</b>
<b>II - ENCARGOS SOCIAIS INCIDENTES SOBRE A REMUNERAÇÃO (R\$)</b>			
<b>GRUPO A - ENCARGOS</b>			
A.01 SEGURIDADE SOCIAL		0,00%	R\$ 0,00
A.02 FGTS		8,00%	R\$ 786,88
A.03 SESI/SESC		1,50%	R\$ 147,54
A.04 SENAI/SENAC		1,00%	R\$ 98,36
A.05 INCRA		0,20%	R\$ 19,67
A.06 SEBRAE		0,60%	R\$ 59,02
A.07 Salário Educação		2,50%	R\$ 245,90
A.08 Riscos Ambientais do Trabalho – RAT x FAP		1,00%	R\$ 98,36
<b>TOTAL - GRUPO A - ENCARGOS</b>		<b>14,80%</b>	<b>R\$ 1.455,74</b>
<b>GRUPO B</b>			
B.01 13º Salário		8,333%	R\$ 819,64
B.02 Férias (sem o abono de 1/3)		8,333%	R\$ 819,64
B.03 Aviso Prévio Trabalhado		1,918%	R\$ 188,66
B.04 Auxílio Doença		1,370%	R\$ 134,75
B.05 Acidente de Trabalho		0,329%	R\$ 32,36
B.06 Faltas Legais		0,274%	R\$ 26,95
B.07 Férias sobre Licença Maternidade		0,056%	R\$ 5,51
B.08 Licença Paternidade		0,021%	R\$ 2,07
<b>TOTAL - GRUPO B</b>		<b>20,634%</b>	<b>R\$ 2.029,57</b>
<b>GRUPO C</b>			
C.01 Aviso Prévio Indenizado		0,417%	R\$ 41,02
C.02 Indenização Adicional		0,167%	R\$ 16,43
C.03 Indenização (rescisão sem justa causa – multa de 40% do FGTS)		3,200%	R\$ 314,75

<b>(nº do item do objeto - perfil profissional)</b>			
C.04 Indenização (rescisão sem justa causa – contribuição de 10% do FGTS)		0,800%	R\$ 78,69
C.05 Abono de Férias - 1/3 constitucional		2,778%	R\$ 273,25
C.06 Abono de Férias - 1/3 constitucional sobre licença maternidade		0,019%	R\$ 1,87
<b>TOTAL - GRUPO C</b>		<b>7,381%</b>	<b>R\$ 726,00</b>
<b>GRUPO D</b>			
D.01 Incidência dos encargos do grupo A sobre o grupo B		3,054%	R\$ 300,38
<b>TOTAL - GRUPO D</b>		<b>3,054%</b>	<b>R\$ 300,38</b>
<b>GRUPO E</b>			
E.01 Incidência do FGTS sobre o aviso prévio indenizado		0,033%	R\$ 3,25
E.02 Incidência do FGTS sobre o período médio de afastamento superior a 15 dias motivado por acidente do trabalho		0,026%	R\$ 2,56
E.03 Incidência de FGTS sobre férias 1/3 constitucional		0,222%	R\$ 21,84
<b>TOTAL - GRUPO E</b>		<b>0,281%</b>	<b>R\$ 27,64</b>
<b>GRUPO F</b>			
F.01 Incidência dos encargos do Grupo A sobre os valores constantes da base de cálculo referente ao salário maternidade		0,099%	R\$ 9,70
<b>TOTAL - GRUPO F</b>		<b>0,099%</b>	<b>R\$ 9,70</b>
<b>TOTAL - ENCARGOS SOCIAIS (R\$)</b>		<b>46,248%</b>	<b>R\$ 4.549,03</b>
<b>VALOR TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS (R\$)</b>			<b>R\$ 14.385,09</b>
<b>III - INSUMOS</b>			
Uniforme			R\$ 0,00
Auxílio alimentação			R\$ 369,60
Vale transporte			R\$ 0,00
Desconto legal sobre transporte (máximo 6% do salário-base)			-R\$ 0,00
<b>TOTAL - INSUMOS (R\$)</b>			<b>R\$ 369,60</b>
<b>TOTAL DE REMUNERAÇÃO + ENCARGOS SOCIAIS + INSUMOS (R\$)</b>			<b>R\$ 14.754,69</b>
<b>IV - LDI E TRIBUTAÇÃO</b>			
<b>LUCRO E DESPESAS INDIRETAS (LDI)</b>			
Despesas Administrativas/Operacionais		7,78%	R\$ 765,25
Lucro		10,55%	R\$ 1.037,70
<b>TOTAL - LUCRO E DESPESAS INDIRETAS</b>		<b>18,33%</b>	<b>R\$ 1.802,95</b>
<b>TRIBUTAÇÃO SOBRE FATURAMENTO</b>			
INSS		4,50%	R\$ 857,91
ISS		5,00%	R\$ 953,23
COFINS		3,00%	R\$ 571,94
PIS		0,65%	R\$ 123,92
<b>TOTAL - TRIBUTAÇÃO SOBRE FATURAMENTO</b>		<b>13,15%</b>	<b>R\$ 2.507,00</b>
<b>PREÇO MENSAL PARA 1 (UM) EMPREGADO (R\$)</b>			<b>R\$ 19.064,63</b>

(nº do item do objeto - perfil profissional)			
Quantidade de profissionais deste perfil			52
<b>PREÇO MENSAL POR PERFIL PROFISSIONAL (R\$)</b>			R\$ 991.360,76
<b>PREÇO ANUAL POR PERFIL PROFISSIONAL (R\$)</b>			R\$ 11.896.329,12

### 1.11. **Memória de Cálculo**

#### 1.11.1. Mão de Obra – Remuneração

1.11.1.1. Salário exemplificativo de R\$ 9.836,06 para o perfil profissional.

#### 1.11.2. **Encargos sociais incidentes sobre a remuneração**

##### 1.11.2.1. **Cálculos do Grupo A**

Item	%	Fundamento
A.01 – SEGURIDADE SOCIAL	0,000%	Lei 12.546/2011
A.02 – FGTS	8,000%	Art. 15, Lei nº 8.036/90 e Art. 7º, III, CF.
A.03 – SESI/SESC	1,500%	Art. 30, Lei 8.036, de 11 de maio de 1990.
A.04 – SENAI/SENAC	1,000%	Art. 1º, caput, Decreto-Lei 6.246, de 1944 (SENAI) e art. 4º, caput do Decreto-Lei 8.621, de 1946. (SENAC).
A.05 – INCRA	0,200%	Art. 1º, I, 2 c/c art. 3º, ambos do Decreto-Lei 1.146, de 31 de dezembro de 1970.
A.06 – SEBRAE	0,600%	Art. 8º, Lei 8.029, de 12 de abril de 1990.
A.07 – Salário Educação	2,500%	Art. 3º, Inciso I, Decreto 87.043, de 22 de março de 1982.

### A.08 – Riscos Ambientais do Trabalho RAT X FAP:

A.08 = RAT x FAP, em que:

RAT – 1% (Desenvolvimento de programas de computador sob encomenda - código 6201-5/00. Anexo V do Decreto nº 3.048/1999);

FAP – 1,000 (Padrão).

A.08 = RAT x FAP = 1 x 1,000 = 1,000%. Encargos do Grupo A= 14,80%

**Observação:** A licitante deve preencher o item A.08 das planilhas de composição de custos e formação de preços com o valor de seu FAP, a ser comprovado no envio de sua proposta adequada ao lance vencedor, mediante apresentação da GFIP ou outro documento apto a fazê- lo. Caso o FAP seja declarado acima de 1,000, o custo a maior deverá ser suportado pela CONTRATADA.

##### 1.11.2.2. **Cálculos do Grupo B**

Item	%	Memória de cálculo	Fundamento
B.01 – 13º Salário	8,333%	$[(1/12)] = 8,333\%$	Art. 7º, VIII, CF/88
B.02 – Férias	8,333%	$[(1/12)] = 8,333\%$	Art. 7º, XVII, CF/88
B.03 – Aviso prévio trabalhado <sup>1</sup>	1,918%	$(7/365) = 1,918 \%$	t Art. 7º, XXI, CF/88, 477, 487 e ss. da CLT
B.04 – Auxílio Doença <sup>2</sup>	1,370%	$\{[(5/365)} = 1,370\%$	Art. 59 e ss. da Lei nº 8.213/91
B.05 – Acidente de trabalho <sup>3</sup>	0,329%	$(15/365) \times 0,08 = 0,329\%$	Art. 19 a 23 da Lei nº 8.213/91
B.06 – Faltas legais <sup>4</sup>	0,274%	$(1/365) = 0,274\%$	Art. 473 da CLT

B.07 – Férias sobre licença maternidade <sup>5</sup>	0,056%	$[(1/12) \times 0,02 \times (4/12)] = 0,056\%$	Impacto do item férias sobre a licença maternidade
B.08 – Licença paternidade <sup>6</sup>	0,021%	$(5/365) \times 0,015 = 0,021\%$	Art. 7º, XIX, CF/88 e 10, §1º, da ADCT
<b>Total</b>	<b>20,634 %</b>		

<sup>1</sup> Redução de 7 dias ou de 2h por dia para 100 % dos empregados. Percentual relativo a contrato de 12 (doze) meses.

<sup>2</sup> Estimativa de 5 (cinco) dias de licença por ano.

<sup>3</sup> Estimativa de 1 (uma) licença de 15 (quinze) dias por ano para 8% (oito por cento) dos empregados.

<sup>4</sup> Estimativa de 1 (uma) ausência por ano.

<sup>5</sup> Estimativa de 2% (dois por cento) dos empregados usufruindo de 4 (quatro) meses de licença por ano.

<sup>6</sup> Estimativa de 1,5% (um inteiro e cinco décimos por cento) dos empregados usufruindo 5 (cinco) dias da licença por ano.

#### 1.11.2.3. Cálculos do Grupo C

Item	%	Memória de cálculo	Fundamento
C.01 – Aviso prévio indenizado <sup>1</sup>	0,417%	$[0,05 \times (1/12)] = 0,417\%$	Art. 7º, XXI, CF/88, 477, 487 e ss. CLT
C.02 – Indenização adicional <sup>2</sup>	0,167%	$[0,02 \times (1/12)] = 0,167\%$	Art. 9º da Lei nº 7.238, de 1984
C.03 – Indenização 40% FGTS (100%) <sup>3</sup>	3,200%	$(1 \times 0,40 \times 0,08) = 3,200\%$	Art. 18, §1º da Lei 8.036/90
C.04 – Indenização 10% FGTS (100%) <sup>4</sup>	0,800%	$(1 \times 0,10 \times 0,08) = 0,800\%$	Art. 1º da Lei Complementar nº 110/01
C.05 – Abono de Férias - 1/3 constitucional	2,778%	$[(1/3)/12] = 2,778\%$	Art. 7º, XVII, CF/88
C.06 – Abono de Férias - 1/3 constitucional sobre licença maternidade	0,019%	$[(1/3)/12] \times 0,02 \times (4/12)] = 0,019\%$	Art. 7º, XVII, CF/88
<b>Total</b>	<b>7,381%</b>		

<sup>1</sup> Estimativa de que 5% (cinco por cento) dos empregados serão substituídos durante um ano.

<sup>2</sup> Estimativa de que 2% (dois por cento) dos empregados serão demitidos em situação de recebimento de indenização adicional.

<sup>3</sup> Multa de 40% do FGTS em relação aos trabalhadores contratados.

<sup>4</sup> Contribuição de 10% do FGTS em relação aos trabalhadores contratados.

#### 1.11.2.4. Cálculos do Grupo D

##### D.01 – Encargos do Grupo A sobre os Encargos do Grupo B

$$D.01 = 0,148 \times 0,20634 = 3,054\%;$$

#### 1.11.2.5. Cálculos do Grupo E

Item	%	Memória de cálculo	Fundamento
E.01 – Inc. do FGTS exclusivamente sobre o aviso prévio indenizado.	0,033 %	$A.02 \times C.01 = (0,08 \times 0,00417) = 0,033\%$	Súmula nº 305 do TST
E.02 – FGTS sobre afastamento superior a 15 dias por acidente de trabalho. <sup>1</sup>	0,026%	$A.02 \times B.05 = (0,08 \times 0,0033) = 0,026\%$	Lei 8.036, de 1990, art. 15, §5º
E.03 – FGTS sobre abono de Férias - 1/3 constitucional.	0,222%	$A.02 \times C.05 = (0,08 \times 0,02778) = 0,222\%$	Lei 8.036, de 1990, art. 15, §5º
<b>Total</b>	<b>0,281%</b>		

<sup>1</sup> Estimativa de que 8% (oito por cento) dos empregados sofrerão acidentes durante o ano, com ausência média de 30 dias durante o ano. O percentual do FGTS (8%) será aplicado somente sobre os 15 dias restantes do afastamento, porque os 15 primeiros dias já foram calculados no item B.05.

#### 1.11.2.6. Cálculos do Grupo F

##### F F.01 – Encargos do Grupo A sobre salário maternidade

$$F.01 = (\text{Encargos Grupo A}) \times (\text{Remuneração}) \times 4/12 \times 2\%, \text{ em que:}$$

$$\text{Encargos do Grupo A} = 0,1480$$

$$4/12 = \text{período de 4 meses de licença em um ano};$$

$$2\% = \text{Estimativa de que 2\% dos empregados usufruirão da licença maternidade de 4 meses em um ano.}$$

$$F.01 = 0,34800 \times (4/12) \times (2/100) = 0,099\%;$$

$$\text{Total Encargos Sociais} = \text{Total Grupo A (14,800\%)} + \text{Total Grupo B (20,634\%)} + \text{Total Grupo C (7,381\%)} + \text{Total Grupo D (3,054\%)} + \text{Total Grupo E (0,281\%)} + \text{Total Grupo F (0,099\%)} = 46,248\%$$

#### 1.11.3. Insumos:

1.11.3.1. **Uniformes:** Não há, da parte da CONTRATANTE, exigência de uniforme para o profissional terceirizado.

1.11.3.2. **Auxílio Alimentação:** Valor exemplificativo.

1.11.3.3. **Vale transporte:** Não foi previsto vale transporte em função do valor estimado da remuneração do profissional.

a) **Dedução legal do vale transporte:** O valor da dedução do vale transporte deve obedecer o art. 4º, parágrafo único da Lei 7.418, de 16 de dezembro de 1985 (desconto máximo de 6% do salário-base). Dado este desconto máximo permitido, não foi previsto vale transporte em função do valor estimado da remuneração do profissional.

#### 1.11.4. Lucro e Despesas Indiretas e Tributação sobre Faturamento

##### Lucro e Despesas Indiretas - LDI:

1.11.4.1. Para fins de estimativa da CONTRATANTE, em Lucro e Despesas Indiretas - LDI, foram consideradas as despesas administrativas e operacionais e a margem de lucro. Os valores de 7,78% e 10,55% para estes itens, respectivamente, são exemplificativos e devem ser ajustados na proposta de preço da licitante.

- a) Obs.: Eventuais custos não previstos expressamente na memória de cálculo devem ser cobertos pelo LDI (Lucro e Despesas Indiretas).

#### Tributação sobre Faturamento

1.11.4.2. Os tributos (ISS, COFINS e PIS) foram definidos utilizando o regime de tributação de Lucro PRESUMIDO. A licitante deve elaborar sua proposta e, por conseguinte, sua planilha com base no regime de tributação ao qual estará submetida durante a execução do contrato.

## 2. GRUPO 2 - MENSURAÇÃO DE SISTEMAS

<b>Tipo de Serviço</b>	<b>Métrica</b>	<b>Volume anual</b>	<b>Valor unitário (R\$)</b>	<b>Valor anual estimado (R\$)</b>
Mensuração de Sistemas	PF	24.000	R\$	R\$
<b>Valor total anual estimado:</b>				R\$



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

## ENCARTE TÉCNICO CONTRATAÇÃO - TI

## ENCARTE F - MODELO DE TERMOS DE RECEBIMENTO

## 1. TERMO DE RECEBIMENTO PROVISÓRIO

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
<b>Termo de Recebimento Provisório – Contrato nº / _____</b> <b>Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.</b> <b>Nº. do Processo: _____</b>	<b>Nº da OS: _____</b>
<b>Relato do RECEBIMENTO PROVISÓRIO DOS SERVIÇOS</b>	
<p>Por este instrumento, atestamos, para fins de cumprimento do disposto no art. 33, inciso I, da Instrução Normativa nº 1 do Ministério do Planejamento, Orçamento e Gestão – MPOG, de 04/04/2019, que os serviços da _____ referentes ao Contrato Nº. _____ prestados pela empresa _____, foram recebidos nesta data e serão objetos de avaliação quanto à conformidade da qualidade e adequação do serviço, de acordo com os critérios de aceitação previamente definidos em Contrato pela CONTRATANTE.</p> <p>Ressaltamos que o recebimento definitivo destes serviços ocorrerá em observância aos prazos legais, após avaliação quanto aos aspectos técnicos e de qualidade, de acordo com os critérios de aceitação definidos em Contrato pela CONTRATANTE</p>	
<b>Representantes da CAPES:</b>	Nome - Matrícula (Fiscal Requisitante)  Nome - Matrícula (Gestor ou Fiscal do Contrato)
<b>Ciência do representante da empresa:</b>	Nome – Cargo na empresa

## 2. TERMO DE RECEBIMENTO DEFINITIVO - TRD

## 2.1. Grupo 1 - Desenvolvimento de Sistemas

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
<p>Termo de Recebimento Definitivo – Contrato nº / _____</p> <p>Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.</p>	<p>Nº da OS: _____</p>
<p><b>1. Dados básicos</b></p>	

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

1.1 Período considerado: \_\_\_/\_\_\_/\_\_\_ a \_\_\_/\_\_\_/\_\_\_

1.2 Período considerado em meses: \_\_\_

1.3 Quantidade de dias úteis no período: \_\_\_

1.4 Quantidade de DIAS-DESENVOLVEDOR realizados no período: \_\_\_

1.5 Quantidade de DIAS-DESENVOLVEDOR esperados no período <sup>a</sup>: \_\_\_

1.6 Quantidade de DIAS-DESENVOLVEDOR sênior JAVA realizados no período: \_\_\_

1.7 Quantidade de DIAS-DESENVOLVEDOR pleno JAVA realizados no período: \_\_\_

1.8 Quantidade de DIAS-DESENVOLVEDOR júnior JAVA realizados no período: \_\_\_

1.9 Quantidade de DIAS-DESENVOLVEDOR sênior JAVA esperados no período <sup>b</sup>: \_\_\_

1.10 Quantidade de DIAS-DESENVOLVEDOR pleno JAVA esperados no período <sup>c</sup>: \_\_\_

1.11 Quantidade de DIAS-DESENVOLVEDOR júnior JAVA esperados no período <sup>d</sup>: \_\_\_

1.12 Quantidade de DIAS-DESENVOLVEDOR sênior PHP realizados no período: \_\_\_

1.13 Quantidade de DIAS-DESENVOLVEDOR pleno PHP realizados no período: \_\_\_

1.14 Quantidade de DIAS-DESENVOLVEDOR júnior PHP realizados no período: \_\_\_

1.15 Quantidade de DIAS-DESENVOLVEDOR sênior PHP esperados no período <sup>e</sup>: \_\_\_

1.16 Quantidade de DIAS-DESENVOLVEDOR pleno PHP esperados no período <sup>f</sup>: \_\_\_

1.17 Quantidade de DIAS-DESENVOLVEDOR júnior PHP esperados no período <sup>g</sup>: \_\_\_

1.18 Quantidade de DIAS-DESENVOLVEDOR sênior PYTHON realizados no período: \_\_\_

1.19 Quantidade de DIAS-DESENVOLVEDOR pleno PYTHON realizados no período: \_\_\_

1.20 Quantidade de DIAS-DESENVOLVEDOR júnior PYTHON realizados no período: \_\_\_

1.20 Quantidade de DIAS-DESENVOLVEDOR sênior PYTHON esperados no período <sup>h</sup>: \_\_\_

1.21 Quantidade de DIAS-DESENVOLVEDOR pleno PYTHON esperados no período <sup>i</sup>: \_\_\_

1.22 Quantidade de DIAS-DESENVOLVEDOR júnior PYTHON esperados no período <sup>j</sup>: \_\_\_

a Quantidade total de membros definido na OS x quantidade de dias úteis no período (item 1.3)

b Quantidade de postos de nível sênior JAVA definido na OS x quantidade de dias úteis no período (item 1.3)

c Quantidade de postos de nível pleno JAVA definido na OS x quantidade de dias úteis no período (item 1.3)

d Quantidade de postos de nível júnior JAVA definido na OS x quantidade de dias úteis no período (item 1.3)

e Quantidade de postos de nível sênior PHP definido na OS x quantidade de dias úteis no período (item 1.3)

f Quantidade de postos de nível pleno PHP definido na OS x quantidade de dias úteis no período (item 1.3)

g Quantidade de postos de nível júnior PHP definido na OS x quantidade de dias úteis no período (item 1.3)

h Quantidade de postos de nível sênior PYTHON definido na OS x quantidade de dias úteis no período (item 1.3)

i Quantidade de postos de nível pleno PYTHON definido na OS x quantidade de dias úteis no período (item 1.3)

j Quantidade de postos de nível júnior PYTHON definido na OS x quantidade de dias úteis no período (item 1.3)

## 2 Taxa Efetiva de Ocupação dos Postos de Trabalho previstos na OS (TEOPT)

2.1 TEOPT (itens (Item 1.4 / 1.5) = \_\_\_/\_\_\_ = \_\_\_ %

2.2 TEOPT sênior [item (1.6+1.12+1.18) / (1.9+1.15+1.20)] = \_\_\_/\_\_\_ = \_\_\_ %

2.3 TEOPT pleno[item (1.7+1.13+1.19) / (1.10+1.16+1.21)]= \_\_\_/\_\_\_ = \_\_\_ %

2.4 TEOPT júnior[item (1.8+1.14+1.20) / (1.11+1.17+1.22)]= \_\_\_/\_\_\_ = \_\_\_ %

## 3. Itens homologados pelo Product Owner (PO) no período

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

3.1 Descrição	3.2 Quantidade de Pontos de Função
3.3 Tamanho da Evolução Funcional (TEF)	

### 4 – Índice de Produtividade (IP)

4.1 Produtividade no período em tamanho funcional = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>a</sup>

4.2 Item de Controle de Produtividade em Tamanho Funcional (ICPTF) = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>b</sup>

#### 4.3 Total de Linhas de Código Alteradas

4.3.1 Total de Linhas de Código Alteradas no período - JAVA = \_\_\_\_\_

4.3.2 Total de Linhas de Código Alteradas no período - PHP = \_\_\_\_\_

4.3.3 Total de Linhas de Código Alteradas no período - PYTHON = \_\_\_\_\_

#### 4.4 Produtividade no período em linhas de código

4.4.1 Produtividade no período em linhas de código - JAVA = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>c</sup>

4.4.2 Produtividade no período em linhas de código - PHP = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>d</sup>

4.4.3 Produtividade no período em linhas de código - PYTHON = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>e</sup>

#### 4.5 Item de Controle de Produtividade

4.5.1 Item de Controle de Produtividade em Linhas de Código (ICPLC) - JAVA = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>f</sup>

4.5.2 Item de Controle de Produtividade em Linhas de Código (ICPLC) - PHP = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>g</sup>

4.5.3 Item de Controle de Produtividade em Linhas de Código (ICPLC) - PYTHON = \_\_\_\_/\_\_\_\_ = \_\_\_\_<sup>h</sup>

4.5.4 Item de Controle de Produtividade em Linhas de Código (ICPLC) = (\_\_\_\_ + \_\_\_\_ + \_\_\_\_)/3 = \_\_\_\_<sup>i</sup>

4.6 Item de Índice de Produtividade = \_\_\_\_ \*0,25+\_\_\_\_ \*0,20<sup>j</sup>

a (Item 3.3) / (Item 1.4)

b (Item 4.1) / Produtividade-base em Tamanho Funcional

c (Item 4.3.1) / (Item 1.6+1.7+1.8)

d (Item 4.3.2) / (Item 1.12+1.13+1.14)

e (Item 4.3.3) / (Item 1.18+1.19+1.20)

f (Item 4.3.1) / Produtividade-base em Linhas de Código JAVA

g (Item 4.3.2) / Produtividade-base em Linhas de Código PHP

h (Item 4.3.3) / Produtividade-base em Linhas de Código PYTHON

i (Item 4.5.1+4.5.2+4.5.3)/3

j (Item 4.2) \* 0,25 + (Item 4.5.4) \* 0,20

### 5 – Índice de Qualidade (IQ)

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

Grupo	Indicador	Unidade	Meta	Valor Obtido	Alcançou a meta? (S / N)
Projeto	Problemas confirmados	Unidades	= 0		
	Complexidade	Média total	<= 10		
	Métodos	Média total	<= 3		
	Índice de manutenibilidade	Nota	A		
	Índice de confiabilidade	Nota	A		
	Índice de segurança	Nota	A		
	Taxa de dívida técnica	%	<= 2,5%		
	Classes	Média total	<=10		
	Arquivos	Média total	<=10		
Violações	Linhas duplicadas (%)	%	<=4%		
	Problemas impeditivos	Unidades	=0		
	Problemas Críticos	Unidades	=0		
	Testes	Testes unitários ignorados	Unidades	=0	
	Sucesso em testes unitários (%)	%	=100%		
	Cobertura (camada de negócio)	%	>=70%		

5.1 Total de Indicadores que alcançaram a meta = \_\_\_\_\_

5.2 Total de Indicadores = 15

5.3 Índice de Qualidade (IQ) = \_\_\_\_\_ / 15 = \_\_\_\_\_ % <sup>a</sup>

a (Item 5.1) / 12

## 6 - Avaliação do Product Owner (APO)

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

Item	Avaliação
1 - Periodicidade de entrega de software para homologação (preferência por períodos entre 1 a 4 semanas)	Entregas realizadas em período igual ou menor a 4 semanas: <input type="checkbox"/> 4 – Todas <input type="checkbox"/> 3 – Maioria <input type="checkbox"/> 2 – Metade <input type="checkbox"/> 1 – Minoria <input type="checkbox"/> 0 – Nenhuma
2 - Receptividade da equipe de desenvolvimento a mudança de requisitos, mesmo em estágio avançado de desenvolvimento	[ ] 4 – Não houve resistência à nenhuma solicitação de mudança de requisitos <input type="checkbox"/> 3 – Não houve resistência à maioria das solicitações de mudança de requisitos <input type="checkbox"/> 2 – Não houve resistência à metade das solicitações de mudança de requisitos <input type="checkbox"/> 1 – Não houve resistência à minoria das solicitações de mudança de requisitos <input type="checkbox"/> 0 – Houve resistência a todas às solicitações de mudança de requisitos
3 - Manutenção pela equipe de desenvolvimento de diálogo contínuo e eficiente com o Product Owner para esclarecer dúvidas e obter feedback	Eventual falta de comunicação da equipe de desenvolvimento com o PO ocasionou: <input type="checkbox"/> 4 – nenhuma falha ou desvio <input type="checkbox"/> 3 – falha ou desvio na minoria das entregas <input type="checkbox"/> 2 – falha ou desvio na metade das entregas <input type="checkbox"/> 1 – falha ou desvio na maioria das entregas <input type="checkbox"/> 0 – falha ou desvio em todas as entregas

Comentários adicionais (optativo)

6.1 APO = (Valor do Item 1 + Valor do Item 2 + Valor do Item 3) / 12 = \_\_\_\_\_ %

**7 - Índice de Evolução do Sistema (IES)**

7.1 IES = (0,25 x ICPTF) + (0,20 x ICPLC) + (0,40 x IQ) + (0,15 x APO)

IES = (0,25 x \_\_\_\_\_) + (0,20 x \_\_\_\_\_) + (0,40 x \_\_\_\_\_) + (0,15 x \_\_\_\_\_) = \_\_\_\_\_

**8 – Valor a pagar**

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

8.1 Valor mensal realizado para a OS: R\$ \_\_\_\_\_

8.2 Valor mensal correspondente aos postos de nível sênior <sup>a</sup>: R\$ \_\_\_\_\_

8.3 Valor mensal correspondente aos postos de nível pleno <sup>a</sup>: R\$ \_\_\_\_\_

8.4 Valor mensal correspondente aos postos de nível júnior <sup>a</sup>: R\$ \_\_\_\_\_

8.4 Desconto por não-ocupação de posto de trabalho - sênior: \_\_\_\_\_ <sup>b</sup>

8.5 Desconto por não-ocupação de posto de trabalho - pleno: \_\_\_\_\_ <sup>c</sup>

8.6 Desconto por não-ocupação de posto de trabalho - júnior: \_\_\_\_\_ <sup>d</sup>

8.7 Desconto adicional por infração ao nível mínimo de serviço exigido para TEOPT = \_\_\_\_\_ % <sup>e</sup>

( ) Foi aplicada exceção prevista no ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364) ao calcular os itens 8.4, 8.5, 8.6 e 8.7

8.8 Desconto a ser aplicado em função do ICPTF = \_\_\_\_\_ % <sup>f</sup>

8.9 Desconto a ser aplicado em função do IQ = \_\_\_\_\_ % <sup>g</sup>

8.10 Desconto a ser aplicado em função do APO = \_\_\_\_\_ % <sup>h</sup>

8.11 Desconto a ser aplicado em função do IES = \_\_\_\_\_ % <sup>i</sup>

8.12 Subtotal de descontos a serem aplicados por não alcance de NMS = \_\_\_\_\_ % <sup>j</sup>

8.13 Valor a pagar = [ (Item 8.1 Valor mensal previsto para a OS) x

(1.2 Período considerado em meses) -

(Desconto por não-ocupação de posto de trabalho - sênior) -

(Desconto por não-ocupação de posto de trabalho - pleno) -

(Desconto por não-ocupação de posto de trabalho - júnior)] x

(100% - Item 8.12).

Valor a pagar = R\$ \_\_\_\_\_

a Ver Cláusula 15<sup>a</sup> do Anexo XVI – Minuta do Contrato do Pregão n.º /20 .

b Item 8.2 x (100% - (item 2.2)), observando exceções previstas no Anexo VI – Níveis mínimos de serviço e cálculo de pagamento.

c Item 8.3 x (100% - (item 2.3)), observando exceções previstas no Anexo VI – Níveis mínimos de serviço e cálculo de pagamento.

d Item 8.3 x (100% - (item 2.4)), observando exceções previstas no Anexo VI – Níveis mínimos de serviço e cálculo de pagamento.

e Ver ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364).

f, g, h, i Ver regra para primeira aferição no âmbito da OS e para segunda aferição em seguida do ENCARTE A - NÍVEIS MÍNIMOS DE SERVIÇO E CÁLCULO DE PAGAMENTO (Sei nº 0920364).

j item 8.7 + item 8.8 + item 8.9 + item 8.10 + item 8.11

### 9 - Fiscalização Administrativa

### COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

Verificar o que o fiscal administrativo faz neste momento

9.1 ( ) Regularidade Fiscal, Trabalhista e Previdenciária – 1<sup>a</sup> parte <sup>a</sup>

9.2 ( ) Comprovação de pagamento de salários (respeitando o limite salarial mínimo estabelecido no Projeto Básico), inclusive férias e 13º salário, quando cabível, de vale- transporte e de vale-alimentação <sup>b</sup>

9.3 ( ) Extratos comprobatórios do recolhimento do FGTS e da contribuição social previdenciária (INSS) <sup>c</sup>

9.4 ( ) Guias da Previdência Social e Guias de Recolhimento do Fundo de Garantia do Tempo de Serviço e Informações à Previdência social com comprovante de entrega <sup>d</sup>

Os documentos referentes aos itens \_\_\_\_\_ encontram-se juntados a este processo.

a Portaria-TCU nº 297/2012, art. 7º, incisos I a V, observado o § 2º.

b Portaria-TCU nº 297/2012, art. 7º, inc. VI e art. 13.

c Portaria-TCU nº 297/2012, art. 7º, inc. VII e artigos 10 e 11.

d Portaria-TCU nº 297/2012, art. 7º, inc. VIII

Representantes da CAPES:	Nome - Matrícula (Fiscal Requisitante)
	Nome - Matrícula (Gestor ou Fiscal do Contrato)
Ciência do representante da empresa:	Nome – Cargo na empresa

## 2.2. Grupo 2 - Mensuração de Sistema

### COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

**Termo de Recebimento Definitivo – Contrato nº / \_\_\_\_\_**

**Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.**

**Nº. do Processo: \_\_\_\_\_**

**Nº da OS: \_\_\_\_\_**

#### Faturamento Apurado

Serviço	Métrica	Qtde	Valor-PF (R\$)	Deflator	Valor (R\$)
<b>Total:</b>					

#### Relato do Recebimento Definitivo dos Serviços

Por este instrumento, atestamos, para fins de cumprimento do disposto no art. 33, inciso XXII, da Instrução Normativa nº 1 do Ministério do Planejamento, Orçamento e Gestão – MPOG, de 04/04/2019, que os serviços da OS- \_\_\_\_\_ referentes ao Contrato Nº. \_\_\_\_\_ prestados pela empresa \_\_\_\_\_, atendem às exigências especificadas no Termo de Referência do Contrato acima referenciado.

Deste modo, está autorizado o encaminhamento da Nota Fiscal no valor de R\$ \_\_\_\_\_

( \_\_\_\_\_ ) relacionado ao pagamento da OS em questão juntamente com a documentação comprobatória da regularidade jurídica, trabalhista e fiscal da empresa.

#### Avaliação da Prestação dos Serviços

A aferição do serviço prestado foi feita mediante a análise da planilha de contagem de Pontos de Função entregue. A execução dos serviços ocorreu dentro da expectativa do Contratante.

Não ocorreu nenhum fato extraordinário que desabone a prestação do serviço da Contratada.

#### De acordo do representantes da CAPES:

	<hr/> <p>Nome - Matrícula (Fiscal Requisitante)</p> <hr/>
	<p>Nome - Matrícula (Gestor ou Fiscal do Contrato)</p> <hr/>
<b>Ciência do representante da empresa:</b>	<hr/> <p>Nome – Cargo na empresa</p> <hr/>



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE G - MODELO DE TERMOS DE CREDENCIAMENTO E DESCREDENCIAMENTO

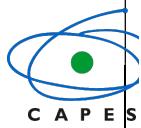
##### 1. MODELO DE TERMO DE CREDENCIAMENTO

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
TERMO DE CREDENCIAMENTO	
A empresa <nome da empresa> CNPJ , Contrato , Endereço: , vem por meio deste Termo solicitar o <u>credenciamento</u> e a liberação de acesso às dependências da CONTRATANTE dos seguintes funcionários abaixo identificados:	
CIÊNCIA FUNCIONÁRIOS DA CONTRATADA	
<Nome> RG	<Nome> RG
De acordo do representante da CAPES:	Gestor do contrato <Nome> RG
Solicitação de acesso do representante da empresa:	Preposto da Contratada <Nome, CPF>

##### 2. MODELO DE TERMO DE DESCREDENCIAMENTO

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
TERMO DE DESCREDENCIAMENTO	
1. A empresa <nome da empresa> CNPJ , Contrato , Endereço: , vem por meio deste Termo solicitar o <u>descredenciamento</u> e o bloqueio de acesso às dependências da CONTRATANTE dos seguintes funcionários abaixo identificados:	

<p>2. Informamos ainda que estamos devolvendo os seguintes materiais que estavam de posse do funcionário acima relacionado:</p> <ul style="list-style-type: none"><li><input type="checkbox"/> Crachá</li><li><input type="checkbox"/> Cartões certificadores</li><li><input type="checkbox"/> Pen drive</li><li><input type="checkbox"/> Outros _____</li></ul>	
<b>De acordo do representante da CAPES:</b>	Gestor do contrato <i>&lt;Nome&gt;</i> RG
<b>Solicitação de bloqueio de acesso do representante da empresa:</b>	Preposto da Contratada <i>&lt;Nome, CPF&gt;</i>



## ARQUITETURA DE REFERÊNCIA - GUIA GERAL

**Versão: 1.1**

Data	Versão	Descrição	Responsável
13/03/2019	1.0	Criação do Guia Geral da Arquitetura de Referência – Alinhada a MDS ágil	Carlos Alberto Rodrigues de Oliveira Santana Francisco Ernesto Diaz Teixeira Neto José Augusto de Jesus
19/03/2019	1.0	Homologação do Guia Geral	Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Leonardo Morais Borges Théo Alves Monteiro Valdecy Lourenço de Araújo Júnior
13/05/2019	1.1	Alteração do item 5 - Anexo A para referenciar o Guia de Frontend de forma geral.	Nirian Martins Silveira dos Santos

## Sumário

<b>1</b>	<b>INTRODUÇÃO.....</b>	<b>5</b>
<b>2</b>	<b>PROPÓSITO.....</b>	<b>5</b>
<b>3</b>	<b>REPRESENTAÇÃO ARQUITETURAL.....</b>	<b>5</b>
3.1	METAS E RESTRIÇÕES ARQUITETURAIS.....	6
3.1.1	<i>Camada</i> .....	9
3.1.2	<i>Frontend</i> .....	9
3.1.3	<i>Backend</i> .....	9
3.1.4	<i>Namespace</i> .....	9
3.1.5	<i>Padrões</i> .....	10
3.1.5.1	Tecnologias.....	10
3.1.5.2	Componentes .....	10
3.1.5.3	Nomenclatura.....	10
3.1.6	<i>Padrões para construção de estória de usuários</i> .....	11
3.2	VISÕES.....	12
3.2.1	<i>Visão lógica (ou de projeto)</i> .....	12
3.2.1.1	Pacotes .....	13
3.2.1.1.1	Pacote extensões .....	13
3.2.1.1.2	Pacote utilitários.....	13
3.2.1.1.3	Pacote modelos .....	13
3.2.1.1.4	Pacote persistência .....	14
3.2.1.1.5	Pacote regras .....	14
3.2.1.1.6	Pacote serviços - camada apresentação.....	14
3.2.1.1.7	Pacote serviços - camada de agregação.....	14
3.2.1.1.8	Pacote diretivas .....	15
3.2.1.1.9	Pacote controladores .....	15
3.2.1.1.10	Pacote configurações .....	15
3.2.1.2	Camadas .....	16
3.2.1.2.1	Camada de acesso de dados .....	17
3.2.1.2.2	Camada de apresentação.....	18
3.2.1.2.3	Camada de serviços .....	19
3.2.2	<i>Visão de implementação (ou de componente)</i> .....	21
3.2.3	<i>Visão de processos (ou concorrência)</i> .....	22
3.2.4	<i>Visão de dados</i> .....	22
3.2.5	<i>Visão física (ou implantação)</i> .....	22
3.3	DEVOPS.....	23
3.3.1	<i>Comunicação</i> .....	23
3.3.2	<i>Colaboração</i> .....	23

<b>3.3.3</b>	<i>Automação</i>	23
<b>3.3.4</b>	<i>Monitoramento</i>	23
<b>3.3.5</b>	<i>Padrão para execução da automação</i>	23
<b>3.3.5.1</b>	<i>Kubernetes</i>	23
<b>3.3.5.2</b>	<i>GitOps</i>	24
<b>3.3.5.3</b>	<i>Ambientes de execução</i>	24
<b>3.3.5.4</b>	<i>Pipeline</i>	24
<b>3.3.5.4.1</b>	<i>Feature branch</i>	25
<b>3.3.5.4.2</b>	<i>hotfix branch, release branch</i>	26
<b>3.3.5.4.3</b>	<i>Tags</i>	26
<b>3.3.5.5</b>	<i>Scripts</i>	27
<b>3.3.5.5.1</b>	<i>Scripts de banco</i>	27
<b>3.3.5.6</b>	<i>Operators</i>	27
<b>3.4</b>	<i>CONTROLE DE VERSÃO</i>	28
<b>3.4.1</b>	<i>Fluxo de trabalho</i>	29
<b>3.5</b>	<i>REQUISITOS DE QA</i>	31
<b>3.5.1</b>	<i>Pré-requisitos de qualidade</i>	31
<b>3.5.2</b>	<i>Automação</i>	32
<b>3.5.3</b>	<i>Cobertura de código por teste</i>	32
<b>4</b>	<i>REFERÊNCIAS</i>	33
<b>5</b>	<i>O ANEXO A – GUIAS DE ARQUITETURAS</i>	34

## 1 Introdução

Este documento descreve uma arquitetura padrão para a construção de aplicativos orientados a recursos REST e interação rica com o usuário. Essa arquitetura será aplicada as seguintes plataformas: Java, PHP, Python e Javascript. Por tratar-se de uma discussão com o enfoque no *design* da aplicação, foram consideradas as diferenças entre as plataformas a serem tratadas em documentos específicos sobre a arquitetura de cada plataforma e o seu respectivo guia operacional para a construção do código.

Aqui, são apresentadas as motivações e diretrizes que orientam o *design* da arquitetura, orientações de como o desenvolvedor pode empregar diretrizes para resolver problemas de sua aplicação, e, por fim, a descrição dos elementos que compõem a arquitetura, tanto na sua estrutura de pacotes quanto na divisão de camadas.

## 2 Propósito

O objetivo principal deste documento é demonstrar as características que orientam a arquitetura, em especial, orientar o desenvolvedor no que concerne na divisão da aplicação em camadas, em pacotes e em estereótipos de classe. Assim, não tem a intenção de ser um documento definitivo de arquitetura para cada sistema individual. Ao invés disso, tenta traçar os objetivos de alto nível que levem a uma boa arquitetura de software. Cada sistema em particular deve ser analisado junto com a equipe de desenvolvimento e um dos arquitetos de sistemas da CAPES.

## 3 Representação Arquitetural

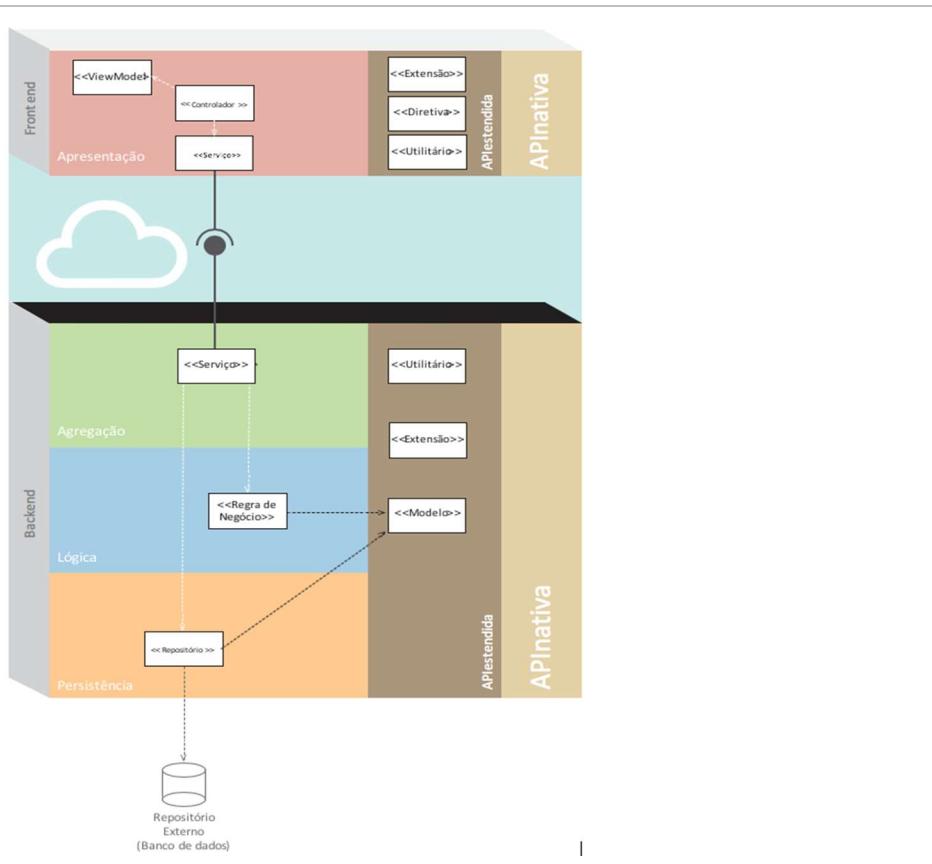


Figura 3.0 - Visão geral da arquitetura

Agrupamento lógico de fragmentos de códigos que implementa um componente, funcionalidade ou aplicação. Conforme figura 3.0, na vertical encontram-se dispostas quatro camadas lógicas: camada de apresentação (vermelho), de agregação (verde), de lógica (azul) de persistência (amarelo). Cada camada está disposta em um ambiente de execução: *Frontend* e *Backend*. Esses ambientes de execução são interligados por serviços.

Alguns elementos, como os que pertencem aos pacotes extensões e utilitários, não pertencem a nenhuma camada específica, por terem a finalidade de estender as capacidades das plataformas que hospedam as aplicações. Nesses casos, para fins da arquitetura, esses elementos serão considerados parte da API básica de cada plataforma.

Na horizontal, são representadas as camadas físicas de cada ambiente. As seções mais à direita (nos tons de marrom e dourado) representam dois níveis de API das plataformas hospedeiras (nativa e estendida), que reúnem os elementos que não fazem parte da aplicação em si, ou seja, linguagem de programação e componentes de terceiros utilizados na codificação.

As caixas brancas representam os estereótipos das classes que serão construídas para a aplicação. As linhas pontilhadas, o relacionamento de dependência entre as classes na sua implementação: *viewmodels*, controladores, serviços, regras de negócio, repositórios, utilitários, extensões, diretivas e modelos.

O espaço azul entre os dois ambientes de execução representa o meio utilizado para o tráfego de informações.

A comunicação entre os elementos da arquitetura hospedadas nos diferentes ambientes dar-se-á em único ponto representada pela conexão entre os estereótipos Serviço da camada de apresentação e Serviço de agregação por meio do protocolo aberto de rede HTTP e a notação JSON.

### 3.1 Metas e Restrições arquiteturais

A arquitetura de software apresentada nesse documento objetiva descrever aplicações modularizadas hospedadas em plataformas web, orientadas a serviços REST, com forte distinção entre seus elementos operacionais. Dentre suas características principais, podemos citar:

- *Alta coesão*: caracteriza-se pela composição de elementos de baixa complexidade e com funções bem definidas, permitindo alto grau de reuso e baixo esforço de manutenção;
- *Baixo acoplamento*: o código produzido será composto por elementos independentes, passíveis de substituição a baixo custo a qualquer momento do ciclo de vida da aplicação. Permite, por exemplo, a troca do gerenciador de banco de dados por um competidor com ajustes pontuais em um único componente da aplicação (o repositório), sem a necessidade de modificação dos demais;
- *Interoperabilidade*: baseada em serviços que funcionam sobre padrões abertos, esse modelo de aplicação permite, ao longo do tempo, o reaproveitamento de funcionalidades completas entre sistemas executados em plataformas distintas. Por exemplo: uma funcionalidade

implementada a partir de um serviço na plataforma Java é passível de utilização imediata em um sistema da plataforma PHP ou Python;

- *Interface intercambiável*: a interação com o usuário é executada em plataforma independente dos demais elementos da aplicação. Essa característica é evidente ao observar-se no diagrama de representação arquitetural, que a camada de apresentação tem um ambiente de aplicação (frontend) para si, isolada das demais camadas. Dessa forma, quando conveniente, é possível acrescentar-se uma nova interface com o usuário a um custo mínimo.
- *Design responsivo*: permite a disponibilização de aplicações construídas seguindo as orientações dessa arquitetura a partir de dispositivos com fatores de interface distintas. Isso significa que, sem esforço adicional, essas aplicações são imediatamente utilizáveis a partir de smartphones e tablets;

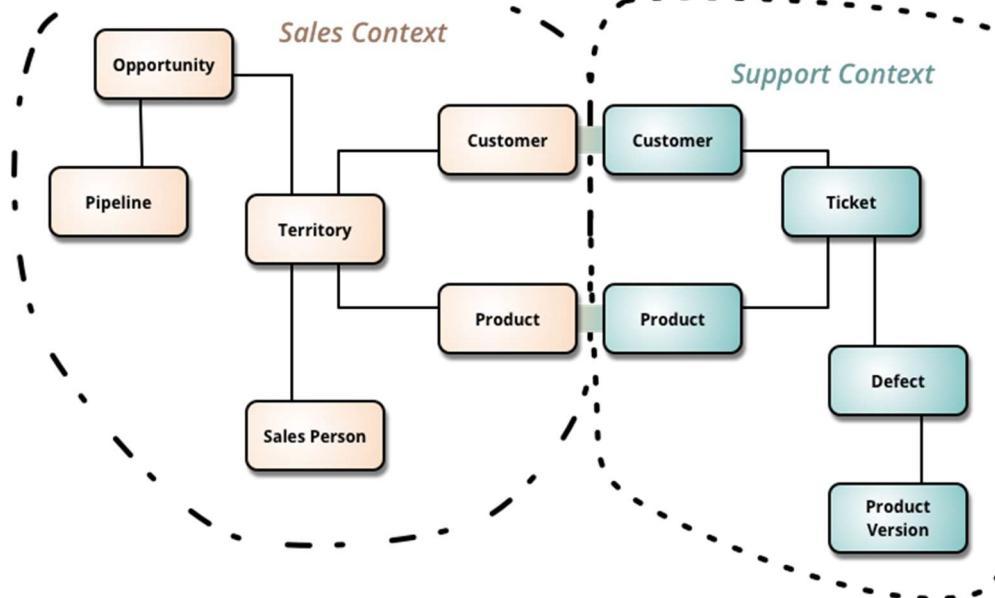
*Testabilidade*: fronteiras claras e distintas entre os componentes da aplicação lhe concede a capacidade inerente de testar esses elementos de forma automática e independente (testes unitários) ou em conjunto (teste de integração e teste funcional).

Além dos princípios acima, a arquitetura das aplicações preza:

- *Alinhamento do código com o negócio*: o contato dos desenvolvedores com especialistas do domínio.
- *Favorecer a reutilização*: códigos que possibilitem o real reaproveitamento.
- *Independência da tecnologia*: soluções que não foquem em tecnologia e sim, no atendimento do negócio.

Dado a separação de responsabilidades apresentada pela *figura 02*, técnicas e padrões do DDD (*Domain-Driven Design*) serão adotadas para o agrupamento de contexto de pacotes, que, segundo Martin Fowler, é o padrão central do DDD.

Um *Agrupamento de Contexto* permite criar seções com responsabilidades específicas dentro de um sistema de contexto mais amplo. Agrupamentos de Contexto se traduz em pacotes da aplicação com responsabilidades e fronteiras bem definidas.



Fonte: <https://martinfowler.com/>

Figura 3.1 – Agrupamento de Contexto

Um *grupamento de Contexto* encapsula todas as funcionalidades necessárias ao atendimento de uma funcionalidade.

A exemplo, um módulo *Foo* terá seu namespace próprio e todas as suas camadas agrupadas sob este. Exemplo de agrupamento de camadas em um pacote:

br.gov.capes.<**NomeSistema**>.foo.camadaApresentação

br.gov.capes.<**NomeSistema**>.foo.camadaServiço

br.gov.capes.<**NomeSistema**>.foo.camadaAcessoADados

Tabela 3.0 – Tabela de agrupamento de camadas em pacotes

O módulo **Foo** conterá todos os recursos para atender ao que se propõe e deverá ser possível seu reuso por outros módulos, o que se traduz em um isolamento de seu modelo de negócio.

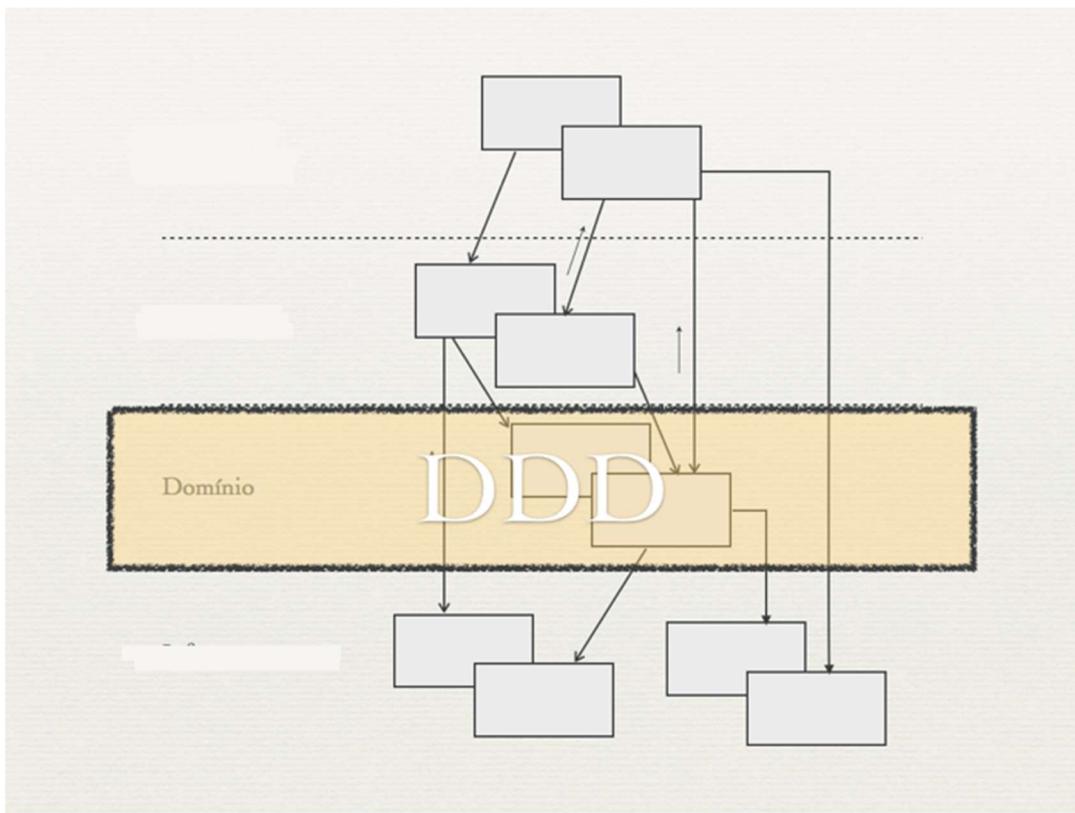


Figura 3.2 - Arquitetura em camadas, utilizada para separar o domínio do resto da aplicação.

### 3.1.1 Camada

Agrupamento lógico de códigos que visa a identificar inequivocamente determinada implementação do sistema. Seu agrupamento consistirá em domínios de negócio ou pacotes.

### 3.1.2 Frontend

O *Frontend view* consiste da camada de apresentação da aplicação ao usuário final. Em se tratando de uma aplicação com interação com usuário via tela gráfica, essa camada será o agrupamento de todos os componentes gráficos necessários para interação entre o usuário e a aplicação. Em sendo uma aplicação que demande interação em modo linha de comando, essa camada assume o papel de detalhar o processo de integração entre o que está sendo configurado e aplicação.

### 3.1.3 Backend

O *Backend* consiste no agrupamento de todas as tecnologias, separadas por camadas, necessárias ao funcionamento da aplicação e que consome o *Frontend view* como entrada ou saída dos dados processados.

### 3.1.4 Namespace

Por sua vez, camadas são logicamente agrupadas em pacotes designados por namespaces. A Capes adotará o namespace padrão:

br/gov/capes/&lt;nomeSistema&gt;

Tabela 3.1 – Namespace padrão Capes

O namespace se reflete na estrutura de diretórios da aplicação, que será iniciada por:

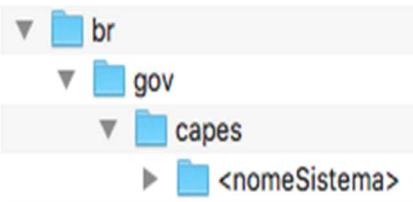


Tabela 3.2 – Estrutura de diretórios base de aplicações Capes

### 3.1.5 Padrões

#### 3.1.5.1 Tecnologias

As tecnologias utilizadas no processo de desenvolvimento de aplicações Capes serão definidas em documentos inerentes a cada pilha de desenvolvimento (Java, PHP, Python).

#### 3.1.5.2 Componentes

Haverá um repositório de componentes homologado pela Capes e referenciado nos respectivos guias de arquitetura. O Anexo A – *Guias de arquiteturas*, deste documento, elenca os Guias de arquitetura e a que tecnologia se destina.

#### 3.1.5.3 Nomenclatura

Para os projetos das plataformas PHP e Python, quando não houver regras que associem os nomes de arquivos e diretórios aos elementos implementados ali, a estrutura de pastas apresentada a seguir será aplicada. Para os projetos da plataforma Java, será necessário seguir a estrutura inerente, em que os diretórios e nomes de arquivos correspondem, respectivamente, à estrutura de pacotes da aplicação e ao nome de cada classe. Os elementos da camada de apresentação, em todos os casos, residem sob o diretório correspondente ao conteúdo estático, conforme citado adiante.

/projeto		Diretório raiz do projeto.
	/api	Contém o código necessário à disponibilização da api REST da aplicação.
	/utils	Contém os componentes da API estendida da aplicação, tais quais interfaces com componentes de logging, geração de documentos PDF ou acessos a sistemas externos.

	/modelos	Contém os modelos da aplicação, tanto os utilizados para persistência quanto aqueles necessários para implementar as APIs estendidas.
	/repositorios	Reúne as classes necessárias para as funcionalidades de persistência, em especial os repositórios e as anotações de mapeamento objeto-relacional.
	/servicos	Conjunto de classes de manipulação pura das informações da aplicação, como a geração de dados agregados ou transformação de dados para a utilização nas demais camadas (e.g. conversão de unidade ou agregação de dados para geração de relatórios ou visualizações alternativas).

Tabela 3.3 – Estrutura de pastas de um projeto CAPES

### 3.1.6 Padrões para construção de estória de usuários

Existem três perguntas primordiais na construção da estória do usuário:

- Quem? - indica o autor da ação ou para quem ela é direcionada ou vai se beneficiar da ação;
- O quê? - indica o que se deseja realizar com a funcionalidade;
- Por quê? - indica o valor e propósito, a justificativa, da funcionalidade.

Um exemplo de como escrever a estória do usuário seria: "Como [Quem?], quero [O quê?] para [Por quê?]."

Também é importante, na estória escrita, definir seus critérios de aceite. Os quais irão validar e orientar o resultado almejado na implementação da funcionalidade entregue.

Em um exemplo mais completo:

"Como ADMINISTRADOR, preciso de acesso a um relatório de vendas para saber quanto recebi em determinado período."

Critérios de aceite:

- Vendedores não poderão ter acesso ao relatório;
- Os relatórios devem ter a opção de imprimir;
- Os relatórios devem ter a opção de exportar para planilha eletrônica;
- Deverá ter filtro por data, para que o administrador possa definir o período desejado.

Pode-se também apontar nos critérios de aceite o detalhamento das informações que desejam ser vistas, ordenações, etc.

### 3.2 Visões

Existem várias formas de se observar um sistema em construção, cada pessoa envolvida ressalta propriedades que lhe interessa e omitem as não relevantes. A exemplo:

- O modo como as pessoas que desempenham papéis diferentes dentro do processo de desenvolvimento de software ou solução veem o problema;
- O modo como cada entidade (componente) da arquitetura de software pode ser observado (perspectivas diferentes).

Assim, conseguimos delimitar suas abstrações nas seguintes visões:

- Visão lógica (ou de projeto);
- Visão de implementação (ou de componente);
- Visão de processos (ou concorrência);
- Visão de dados;
- Visão física (ou implantação).

#### 3.2.1 Visão lógica (ou de projeto)

A visão lógica é a área que detém as funcionalidades do sistema ou solução a ser elaborada.

Essa visão contempla os diagramas de caso de uso e classes (UML).

Existe a possibilidade de haver uma visão em modelo de processos (BPMN) para melhor entendimento e complementação do mesmo.

Deve-se levar em consideração que em um modelo ágil tais documentos e diagramas não são obrigatórios em sua totalidade. Podendo haver a eleição de quais poderão ser entregues mediante acordo e nível de complexidade da operação.

A partir do momento em que esta documentação faça parte de um projeto, é imprescindível seu alinhamento com a visão de implementação e processos para um correto acompanhamento e histórico.

Em resumo:

- Os atores dessa visão são os clientes do produto, os analistas e os desenvolvedores;
- Existe uma forte ligação ao problema do negócio;
- Independente de decisões de projeto;
- Descreve e especifica a estrutura estática do sistema e as colaborações dinâmicas entre objetos via mensagens para realizarem as funções do sistema;
- Contém a coleção de pacotes, classes e relacionamentos.

### 3.2.1.1 Pacotes

Os elementos da aplicação hospedadas em *servidor de aplicação* (i.e. *camadas de persistência, lógica e agregação*), serão distribuídos em pacotes principais, tais como: *extensões, utilitários, modelos, persistência, regras, serviços*.

A camada de apresentação tem uma divisão de pacotes particulares, a saber: *extensões, utilitários, diretivas, serviços, controladores e configurações*. No âmbito dessa camada, pacotes são denominados *módulos*. A distribuição e denominação distintas se dá a fim de adequar os elementos dessa camada aos mecanismos inerentes da plataforma que será utilizada na sua implementação (i.e. Java).

Ambas as lista de pacotes não são extensivas e, quando conveniente, é prerrogativa do desenvolvedor subdividir os pacotes sugeridos ou acrescentar novos, desde que a nova organização não viole a que se encontra descrita neste documento e se mostre benéfica à compreensão dos mecanismos implementados para aquela aplicação específica.

#### 3.2.1.1.1 Pacote extensões

Em todas as plataformas escolhidas, é-se possível estender as classes das APIS nativas, acrescentando métodos personalizados nas classes nativas (Texto, Vetores, entre outros). O material desenvolvido deverá ser reunido neste pacote, o qual, quase sempre, se resume a código.

Métodos de extensão têm, por natureza, alto potencial de reaproveitamento em qualquer aplicação da mesma plataforma, na forma de *bibliotecas de extensão*.

As classes desse pacote podem ser referenciadas pelo código de qualquer pacote da aplicação.

#### 3.2.1.1.2 Pacote utilitários

Reúne as classes com funções das mais diversas. Escritas para realizar tarefas comuns referentes à aplicação que não se categorizam como regras de negócio, eliminando assim a redundância de código.

Conversão de tipos e formatação de dados específicas de uma aplicação são exemplos de funções comumente realizadas em classes de pacote *utilitários*.

As classes desse pacote podem ser referenciadas pelo código de qualquer pacote da aplicação.

#### 3.2.1.1.3 Pacote modelos

As classes de estereótipo *Model* (conforme descrito mais adiante) e classes utilizadas para o transporte de informações na forma de parâmetros ou resultados de operações são reunidas neste pacote.

Possuem, em essência, apenas atributos e sua lógica interna e simplificada. Na maior parte do tempo, contém apenas código com o objetivo de facilitar a conversão ou agregar os atributos ali armazenados como, por exemplo, propriedades calculáveis a partir das demais (exemplo: o valor agregado de um produto: preço unitário versus quantidade).

As classes desse pacote podem ser referenciadas pelo código de qualquer pacote da aplicação. Não é permitido que faça referência a classes pertencentes aos pacotes *regras, persistência ou controladores*.

### 3.2.1.1.4 Pacote persistência

As classes do pacote *persistência* realizam as funções de atualização e recuperação de informações a partir de repositórios destinados ao seu armazenamento.

No caso mais comum esses repositórios correspondem a bancos de dados relacionais, mas é possível a utilização de qualquer dispositivo que permita as operações necessárias para esse fim. Alguns exemplos são arquivos de texto, arquivos binários ou bancos de dados não estruturados (NoSQL, sistema de arquivos, serviços externos, entre outros).

A função primordial das classes desse pacote é tornar os mecanismos de persistência transparentes da perspectiva das demais classes da aplicação, concentrando todos os detalhes necessários para essas operações.

As classes pertencentes a esse pacote são do estereótipo repositório e devem ser referenciados somente a partir de classes pertencentes ao pacote *serviços*.

### 3.2.1.1.5 Pacote regras

Concentra as classes que implementam as regras de negócio de uma aplicação. São classes de um único estereótipo (*regra de negócio*), que realizam seus objetivos por meio da alteração do estado interno de classes individuais ou conjuntos de classes da aplicação. As funcionalidades necessárias para a efetivação dessas regras (exemplo; recuperação e persistência das informações) não fazem parte das implementadas pelas classes desse pacote. Desse modo, é comum a ocorrência de código composto de classes desse pacote e do pacote *persistência* lado a lado.

As classes pertencentes a esse pacote serão referenciadas somente a partir de classes pertencentes ao pacote *serviços*.

### 3.2.1.1.6 Pacote serviços - camada apresentação

Contém classes análogas àquelas encontradas no pacote *serviços*, agora na função de cliente. Em essência, fazem a comunicação (via rede) com os serviços hospedados no *backend* (da camada de agregação) e conversões de dados necessárias para realizar os *ViewModels* (explicados mais adiante).

O objetivo principal dessas classes é realizar o isolamento da comunicação com o *backend* das classes dos demais pacotes da camada de apresentação.

### 3.2.1.1.7 Pacote serviços - camada de agregação

Concentra as classes as quais se encontram as implementações da fronteira de serviços em uma aplicação. Nessas classes, encontra-se o código necessário para coordenar as funcionalidades encontradas nas demais. Rotinas que recuperam modelos a partir de repositórios é comum em uma classe de serviço. Processar informações utilizando *regras de negócio*, validar

parâmetros recebidos na sua chamada e atualizar informações da aplicação são tarefas comuns deste pacote.

Deste pacote, permiti-se fazer referência a classes de todos os demais pacotes. Mas o contrário não é permitido.

### 3.2.1.1.8 Pacote diretivas

Esse módulo pertence ao código que implementa a interação com o usuário, e contém as construções necessárias para a implementação de diretivas de comportamento. Em geral, nele, encontram-se os componentes (visuais ou não) necessários para a interação os quais não estão disponíveis naturalmente na plataforma ou composições de componentes que evitam redundância de código ao longo de toda aplicação.

Por exemplo, um componente visual para o preenchimento e validação de números de CEP pertenceria a essa categoria.

Classes do módulo diretivas são acessadas apenas pela infraestrutura da interface ou camada de apresentação.

### 3.2.1.1.9 Pacote controladores

As classes desse módulo realizam as funções de orquestração dos componentes necessários para implementar a interação com o usuário. Em essência, ela utiliza as classes do módulo de serviços, implementam os comportamentos esperados dos componentes visuais (exemplo: respostas a eventos capturados junto ao usuário) e ajustes dos elementos gráficos necessários.

Uma classe típica desse módulo é aquela que implementa o comportamento de formulários: ela inicializa componentes visuais, recupera informações utilizando classes do módulo serviços, implementa o comportamento da interação com o usuário (e.g. validação de informações entradas pelo usuário no lado cliente) e utiliza novamente classes do módulo serviços para atualizar informações ao final da operação.

### 3.2.1.1.10 Pacote configurações

Em alguns casos, em especial na implementação de comportamentos não nativo da interface do usuário, a execução de código específico para efetivar as configurações necessárias ao ambiente cliente é requerida.

Para manter o código organizado, as configurações necessárias para a interação com o usuário serão centralizadas no módulo configurações.

### 3.2.1.2 Camadas

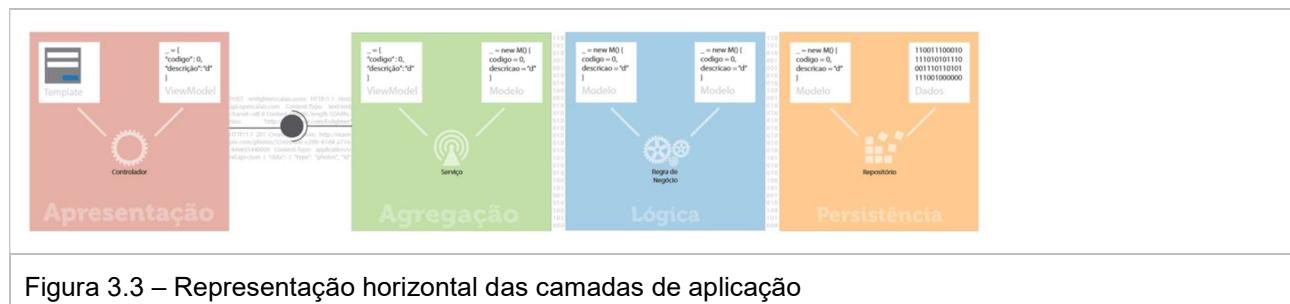


Figura 3.3 – Representação horizontal das camadas de aplicação

Os elementos que implementam uma aplicação são segregados em camadas lógicas. Essa separação permite a determinação precisa das funções realizadas por cada conjunto de classes. Alguns elementos, como os que pertencem aos pacotes extensões e utilitários, não pertencem a nenhuma camada específica, por se tratarem de elementos que têm, por fim, entender as capacidades das plataformas que hospedam as aplicações. Nesses casos, para fins da arquitetura, esses elementos serão considerados parte da API básica de cada plataforma.

Portanto, uma aplicação é conceitualmente dividida em quatro camadas:

- A *camada de persistência* reúne as classes, as quais recuperam e atualizam informações em repositórios de dados. Esses elementos abstraem as complexidades inerentes do acesso a componentes externos especializados em armazenamento e recuperação de informações.
- A *camada de lógica* contém classes chamadas "regras de negócio", que implementam a lógica particular ao sistema, elementos que manipula as informações do sistema para refletir mudanças de estado.
- A *camada de agregação* reúne o conjunto de elementos, os quais formam a fronteira de serviços expostos pela aplicação. Sua principal função é isolar a camada de apresentação das camadas de lógica e de persistência. Com esse objetivo, a camada de agregação opera, principalmente, pela orquestração dos elementos dessa camada e pela conversão de dados entre os tipos binários (classes utilizadas na plataforma de execução) e os tipos serializados (representação JSON utilizada pela camada de apresentação, por exemplo).
- A *camada de apresentação* captura estímulos dos usuários, interpreta suas instruções, envia e recebe dados contra os elementos da camada de agregação e apresenta as informações de forma humanamente compreensível. Camada mais externa do sistema e única que realiza interações diretas com o usuário. Composta

pelos elementos dos pacotes extensões, utilitários, diretivas, controladores e configurações.

Essa divisão aplica-se tanto à camada de visão lógica quanto à implementação do sistema: os elementos pertencentes a uma camada restringem-se aos de pacotes pré-determinados, os quais contém apenas elementos de determinados estereótipos. Essa correspondência entre elementos de implementação e elementos lógicos permite a verificação da consistência de uma aplicação frente a arquitetura pela utilização de regras de validação.

Há, por exemplo, regras para a referência em código de elementos de uma determinada camada para as demais, conforme citado ao longo deste texto. Considerando-se a camada de persistência como a mais interna e a de apresentação, a mais externa, em especial não é permitido a elementos de uma determinada camada fazer referência a elementos de outras mais exteriores. Há também regras para a implementação de funcionalidades em determinados pacotes, os quais permitirão o rastreio da divisão de responsabilidades entre os elementos de uma aplicação.

### 3.2.1.2.1 Camada de acesso de dados

Na camada de acesso a dados, estão codificadas todas as operações pertinentes a persistência de informações. Nenhuma manipulação de dados será realizada em componentes que pertençam a quaisquer outras camadas, em exceção às necessárias e convenientes durante a interação com o usuário.

Nesta camada, encontram-se classes de três naturezas: aquelas que representam as informações manipuladas, de estereótipo *Model*; aquelas que cuidam da persistência e recuperação dessas informações, de estereótipo *repositório*; e aquelas que realizam operações sobre essas informações, de forma individual ou em conjuntos (ou coleções), de estereótipo regras de negócio.

*Classes de estereótipo Model* representam informações com correlação semântica, desde que essa semântica não seja a de coleção de itens ou de coleções aninhadas. Uma classe *Model* contém, por exemplo, informações pertinentes a uma pessoa, ou um lugar, ou uma operação transacional. Classes desse estereótipo podem conter anotações e metadados necessários para o funcionamento dos mecanismos de persistência, mas não deverão ter nenhum tipo de dependência das classes dessas camadas que pertençam aos demais estereótipos, direta ou indiretamente. É vetado, também, que represente um conjunto de objetos resultados de uma consulta ou operação (semântica de coleção).

Não há exigências quanto aos critérios de normalização dos dados de uma classe *Model*. Ela poderá conter atributos *multivvalorados* ou de tipo *coleção*, desde que essa coleção tenha relacionamento semântico com o restante das informações e que o propósito da classe não seja

servir exclusivamente como resultado de operações (como consultas ou processamentos). Para essa finalidade, devem ser utilizadas as classes disponíveis na plataforma de implementação.

*Classes de estereótipo repositório* têm a finalidade primária de concentrar as operações de persistência de dados em repositórios diversos.

As *classes de estereótipo regra de negócio* concentram as operações, as quais não têm por consequência implícita a persistência de dados. Entre elas, encontram-se operações que alteram o estado interno das classes de estereótipo *Model* e processamento de coleções de objetos dessa classe. Por exemplo, uma operação que processe um conjunto de itens referentes a uma nota fiscal e realize as operações necessárias no sistema para sua efetivação (ajuste de estoque, financeiro, etc).

É vedado que operações das classes de estereótipo *regra de negócio* realizem ou dependam diretamente da persistência das informações que opera. A orquestração entre persistência e operações de regra de negócio deve ser realizadas no ponto em que as essas classes são originalmente instanciadas, em geral em classes da *camada de agregação*.

Quando conveniente, classes de estereótipo repositório poderão ser, também, do estereótipo *regras de negócio*, desde que os métodos de cada tipo sejam claramente segregados no código fonte e que sigam as diretrivas aqui indicadas. Essa mescla é indicada em situações de aplicações com poucas regras de negócio ou quando estas forem simples ou, ainda, quando julgar que diminui beneficamente a complexidade do código fonte.

### 3.2.1.2.2 Camada de apresentação

A camada de apresentação contém código e recursos necessários para capturar ações realizadas pelo usuário e representar as informações da aplicação de forma acessível a ele. Entre os recursos, encontram-se gráficos, textos, marcações e elementos de interação (e.g. botões, caixas de texto, etc).

Aqui, as figuras de classes e objetos dão lugar a um conjunto de aplicações especializadas que interagirão contra a camada de agregação para realizar as ações do sistema.

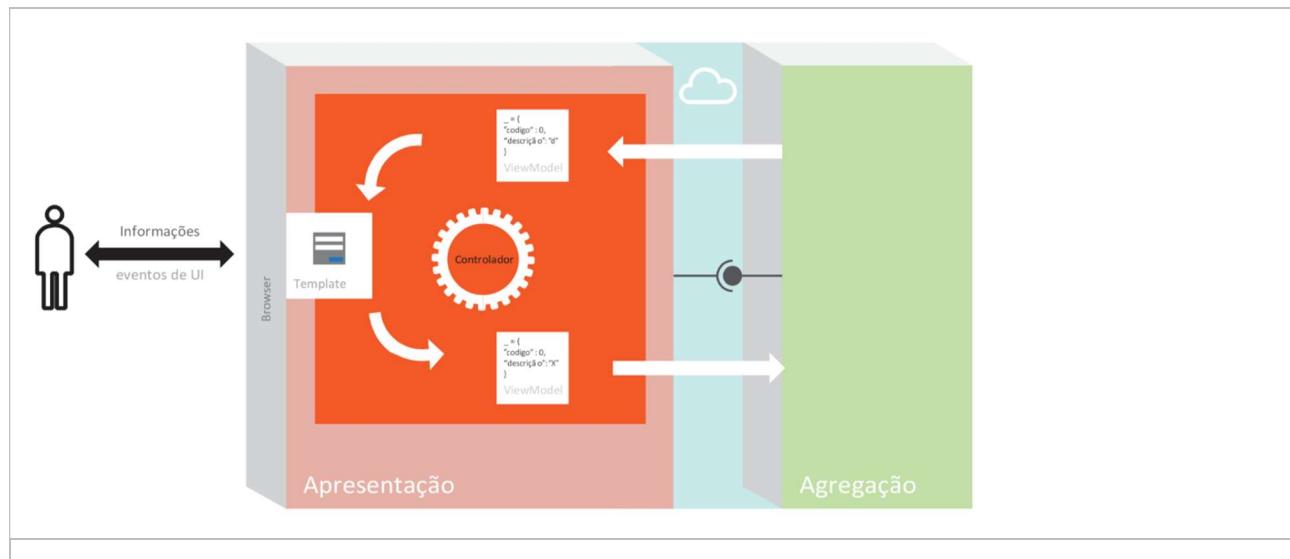
A camada de apresentação pode conter os seguintes elementos em sua composição: *controladores*, *viewmodels* e *templates*.

*Controladores* são os elementos centrais que orquestram o funcionamento dos demais. Suas funções principais são recuperar modelos através de chamadas à camada de agregação, associar modelos a *templates*, reagir aos eventos capturados e informações recebidas, e disparar processos solicitados pelos usuários, novamente a partir de chamadas à camada de agregação. Em suma, controladores representam o corpo do código da camada de apresentação.

*ViewModels* (DTO) são containers para os dados que trafegam entre serviços, controladores e templates. A grosso modo, têm as mesmas características de objetos *Model*: contém apenas atributos e nenhuma lógica interna, a não ser a necessária para manipular seus valores internos (setters e getters) ou gerar atributos calculados. A priori, *viewmodels* são criados e utilizados nas chamadas e respostas dos serviços, mas a criação de *ViewModels* internos é conveniente para simplificar a interação de controladores com *templates*. Por serem gerados a partir das informações recuperadas de serviços, *ViewModels* coincidem regularmente com classes de persistência (quando o recurso exposto por um serviço de consulta e atualização de dados refletir fielmente uma classe de persistência, por exemplo). Porém, não existe relação estrita entre esses dois elementos: em chamadas serviços compostos, por exemplo, *ViewModels* refletirão, de forma geral, os parâmetros necessários à execução daquele serviço.

*Templates*, por sua vez, são os elementos da interface do usuário: construções que exibem e capturam informações. Em geral, *templates* são associados a *ViewModels* e geram a marcação do código visual necessário para exibir e/ou manipular as informações contidas ali. Por exemplo, um *ViewModels* associado a um *template* de formulário gerará os rótulos e caixas de texto necessárias para capturar e editar aquelas informações.

Quando necessário construir uma interação mais complexa ou um componente visual que não esteja disponível ou possa ser composto pelos disponíveis na biblioteca padrão, novos *templates* podem ser construídos por código.



### 3.2.1.2.3 Camada de serviços

A camada de serviços concentra classes de dois estereótipos (serviços e *ViewModels*).

Os serviços implementarão os canais de serviços REST da aplicação e servirão como fronteira mais externa da aplicação em relação à infraestrutura de execução, tanto pública (internet) quanto privada (intranet). O conjunto de serviços de uma aplicação formam sua API pública. Esta serve como ponto de integração para todas funcionalidades da aplicação, tanto aquelas que necessitam de uma interface para interação com o usuário, quanto aquelas disponíveis para integração com outros sistemas.

Cada serviço é implementado como uma rota dentro de um controlador MVC (*classes de estereótipo controlador*). Controladores, portanto, podem ser vistos como grupos de serviços. O critério para o agrupamento poderá ser funcional (ou horizontal), quando um controlador reúne serviços que realizam operações semelhantes em classes diferentes, ou semântico (ou vertical), quando um controlador reúne serviços referentes a um único recurso (ou entidade, ou classe de negócio) do sistema. A escolha por um ou outro tipo de agrupamento deverá levar em conta a conveniência e a quantidade de itens que serão agrupados em cada controlador.

A priori, cada serviço pertence a uma das seguintes categorias: serviços de consulta e atualização de recursos, que costumam ser agrupados pelo critério funcional; ou serviços compostos (ou de operações transacionais), cuja implementação realizará operações sobre recursos (ou entidades) diferentes, que costumam ser agrupados pelo critério semântico e que, quando realizam atualização de informações, envolvem uma transação ACID implícita.

Como exemplos de serviços de consulta e atualização de recursos, cita-se a exposição de serviços REST tradicionais sobre as entidades da aplicação: por exemplo, um serviço de consulta e atualização do catálogo de produtos em um sistema de controle de estoque. Não é mandatório que um recurso exposto corresponda exatamente a uma entidade representada em uma classe *Model* persistida em um repositório, o recurso pode ser composto internamente a partir de repositórios e classes diferentes, criando funcionalidades similares aos serviços implementados sobre uma única classe *Model*. Nesse caso, o serviço deverá realizar as composição e decomposição necessárias para prover a funcionalidade descrita. Cada serviços de consulta e atualização de recursos costuma ter por denominação um substantivo (e.g. / produto) e implementar diversos verbos REST sobre um mesmo recurso.

Serviços compostos utilizam elementos da camada de acesso a dados para compor operações complexas e realizar a persistência das informações atualizadas ao seu final. Também representa um serviço composto válido aquele que é formado de diversas consultas, sobre uma mesma entidade ou sobre entidades diferentes, com a finalidade de montar representações de dados que exijam estruturas de informação complexas como, por exemplo, grafos e árvores. Esse caso distingue-se dos serviços de consulta e atualização de recursos com composição, conforme descrito anteriormente, porque não permitirá atualização de dados e por não ser sua finalidade

causar uma ilusão de recurso unificado para o exterior da aplicação. Um exemplo típico de serviço composto é aquele que provê uma API para operações transacionais típicas de um sistema de controle de estoque: de saída, entrada ou transferência de produtos nesse sistema. Serviços compostos costumam ter verbos por denominação (e.g. /vender) e implementar apenas um verbo REST.

Os *ViewModels* são representações simplificadas das classes de estereótipo *Model*, pertencentes à camada de acesso a dados. A priori, não é vedada a reutilização das classes de estereótipo *Model* na função de *ViewModels*. Porém, como as classes de estereótipo *Model*, em geral, contém mais informações que o necessário ou desejável para sua utilização, além da fronteira de serviços da aplicação. A criação de uma representação simplificada (*ViewModel*) é um artifício utilizado de forma regular. Apesar de representar, de uma certa forma, redundância de código, essas classes têm a virtude de simplificar conversões de dados na fronteira de serviços (de ambos os lados, cliente e servidor), e obfuscar as estruturas internas da aplicação para o mundo exterior, aumentando tanto a performance da comunicação em rede quanto o grau de segurança da aplicação.

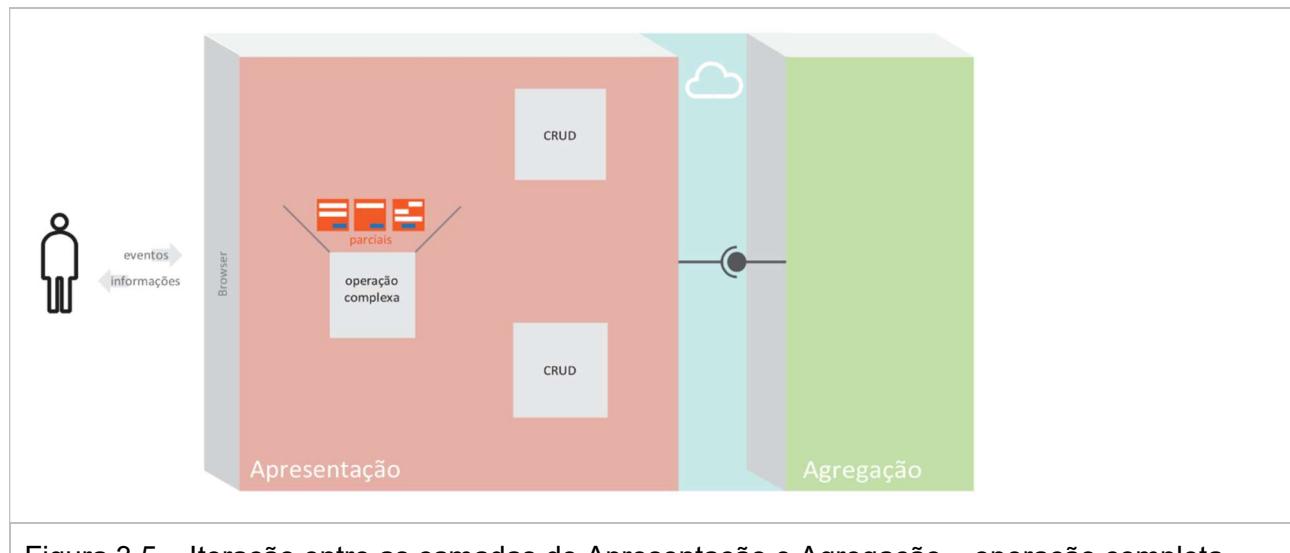


Figura 3.5 – Iteração entre as camadas de Apresentação e Agregação – operação completa

### 3.2.2 Visão de implementação (ou de componente)

Nessa visão, existe o desenvolvimento do que foi especificado na visão lógica. Entretanto, em um modelo ágil, no qual é possível haver mudanças ou ajustes, existe uma interação periódica do que está sendo desenvolvido para melhor aderência ao plano da visão lógica e minimizar os problemas que podem surgir durante o desenvolvimento.

Em resumo:

- Envolve diretamente os desenvolvedores, ainda que exista também um direcionamento pela equipe de análise e gerência do projeto;
- Implementação dos módulos e suas dependências são descritos;
- É utilizada para saber como distribuir o trabalho de implementação e manutenção entre os membros da equipe considerando aspectos de reuso, subcontratação e aquisição de softwares que auxiliem o projeto.

A CAPES utiliza a arquitetura em camadas já descritas e os pacotes, as nomenclaturas, entre outros aspectos estruturais.

### 3.2.3 Visão de processos (ou concorrência)

Na visão de processos é explicitado pontos tais como desempenho, escalabilidade, vazão das informações e tolerância a falhas. Além disso, os processos do sistema e como eles se comunicam e as atividades do seus funcionamentos deverão ser descritos

Em resumo:

- Trata a divisão do sistema em processos e processadores (propriedade não funcional);
- O sistema é dividido em linhas de execução de processos concorrentes;
- Esta visão de concorrência deverá mostrar como se dá a comunicação e as concorrências;
- Considera questões de desempenho, confiabilidade, tolerância a falhas.

### 3.2.4 Visão de dados

Fornece uma perspectiva de armazenamento de dados persistentes do sistema. A visão de dados é opcional se os dados persistentes forem poucos ou inexistentes ou se a conversão entre o Modelo de Design e o Modelo de Dados for trivial.

### 3.2.5 Visão física (ou implantação)

A visão de implementação mostra a distribuição física do processamento no sistema. Nessa visão, são especificados os maquinários e meios envolvidos para o funcionamento do sistema implementado.

O advento da virtualização e conteinerização (ver DevOps) não exime a existência deste modelo, muito pelo contrário, se torna imprescindível pois existe o aumento da quantidade de pontos de falhas e exige uma análise mais rigorosa. Felizmente existem formas de mitigar esse problema (ver DevOps).

Em resumo:

- Conter a parte física do sistema e a conexão entre suas sub-partes, interação hardware e software, com objetivo de colocar o sistema em operação;
- Mostrar a organização física do sistema, os computadores, os periféricos e como eles se conectam entre si;
- Esta visão será executada pelos desenvolvedores, integradores e testadores, e será representada pelo diagrama de implantação, pois considera o ambiente de desenvolvimento, teste e produção.

### 3.3 DevOps

DevOps tem em seu cerne a evolução e melhoria contínua do processo de entrega de software e é fundamentado em 4 pilares: Comunicação, Colaboração, Automação e monitoria.

#### 3.3.1 Comunicação

Ao criar o software, a equipe precisa de constante comunicação para o planejamento e liberação do código, bem como resolução de impedimentos. Na CAPES, foi adotado as ferramentas Redmine e Mattermost. Redmine é utilizado para gerenciamento da metodologia ágil; Mattermost, para comunicação.

Para maiores informações sobre ambas as ferramentas, analisar a documentação de uso do Mattermost e Redmine em seus respectivos sites oficiais.

#### 3.3.2 Colaboração

O processo de colaboração é executado pelo próprio time. Devendo o gerente ser o responsável por facilitar o processo em casos de impedimentos.

#### 3.3.3 Automação

A automação é o coração de todo o processo DevOps e deve tornar os procedimentos determinísticos e estáveis. Os procedimentos serão detalhados na seção padrões para execução da automação.

#### 3.3.4 Monitoramento

O monitoramento pelas equipes DevOps tem como papel fornecer informações, as quais garantam o tempo de atividade do serviço e o desempenho ideal em tempo de execução (*runtime*).

#### 3.3.5 Padrão para execução da automação

Para o correto entendimento de onde a automação se encaixa dentro do processo, exigi-se o necessário entendimento dos elementos, abaixo elencamos, que compõem a automação :

##### 3.3.5.1 Kubernetes

A CAPES utiliza o Kubernetes como orquestrador do ambiente de execução das aplicações. Mais especificamente a distribuição OpenShift. As aplicações são disponibilizadas em ambientes conteinerizados e colocados em execução pelo orquestrador.

Alguns dos benefícios da adoção da ferramenta são:

- Igualdade dos ambientes de desenvolvimento das aplicações até a produção
- Criação de ambientes dinâmicos
- Auto scale das aplicações em caso de picos de utilização

### 3.3.5.2 GitOps

GitOps é a utilização do Git como fonte única verdadeira para implantações de ambientes. Geralmente as modificações são aplicadas ao git por meio de um commit, gerando a execução de um pipeline para a execução de uma atividade qualquer relacionada a informação que foi colocada no repositório.

Na CAPES, o padrão gitops é utilizado para o desenvolvimento de aplicações, de forma que o commit do usuário acione o *pipeline* que efetuará a validação, building, testes e *deployment* do código alterado para o ambiente de desenvolvimento até a produção.

### 3.3.5.3 Ambientes de execução

A ferramenta OpenShift permite que sejam criados quantos ambiente forem necessários para a implantação de aplicações. Entretanto definimos 4 ambientes distintos para o ambiente CAPES:

*Revisão* - são ambientes dinâmicos criados sob demanda e temporários para a apresentação de uma funcionalidade, eles trabalham ligados a uma feature branch do gitflow.

*Homologação* - ambiente esse que é utilizado pelo cliente para a homologação do sistema.

*Integração contínua* - ambiente dinâmico usado para a execução automatizada dos testes de integração, carga e performance.

*Produção* - ambiente onde a aplicação é executada para atender a demanda externa dos usuários.

Embora esses sejam os ambientes padrões para os sistemas, nada impede que possam se criados novos ambientes para atendimento das especificidades das aplicações.

### 3.3.5.4 Pipeline

Conjunto de passos executados em uma sequência lógica para que um objetivo final seja atendido. O *pipeline* é integrado ao Git de forma que *commits* do usuário açãoem sua execução.

Abaixo, vemos um exemplo de *pipeline* com 4 fases: preparação, building, teste, análise de segurança.

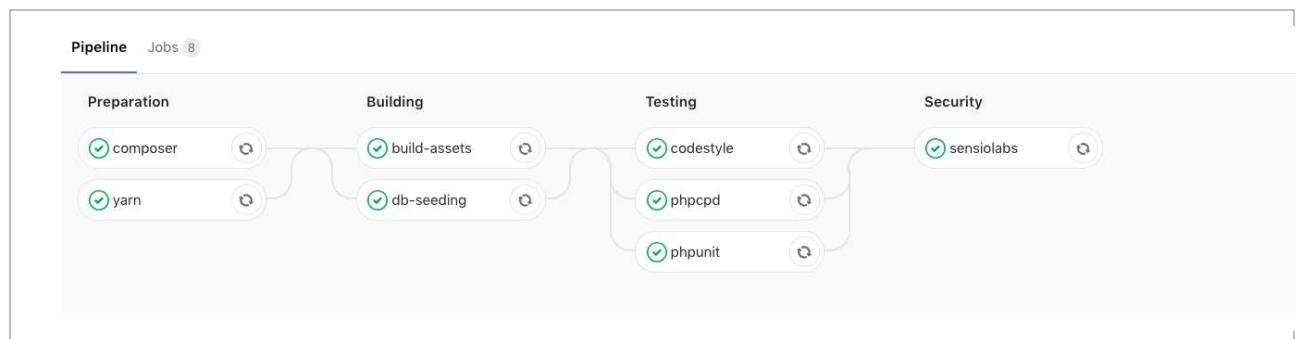


Figura 3.6 – Representação de *pipeline* executado com sucesso

Em caso de falha de uma das etapas, o commit é marcado como defeituoso pois não atendeu as exigências do pipeline e isso bloqueia a continuidade da execução do *pipeline*.

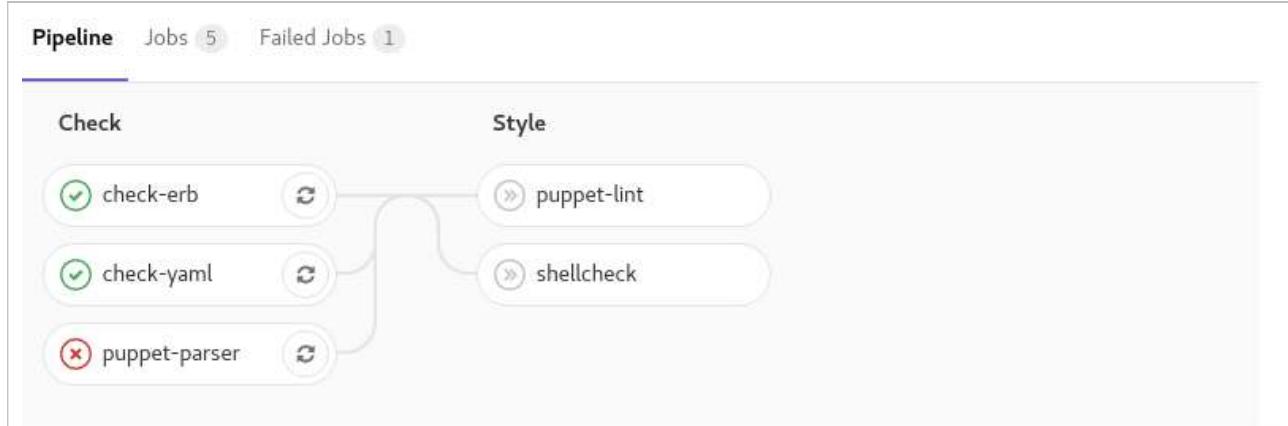


Figura 3.7 – Representação de *pipeline* executado com falha

Um pipeline pode ser customizado para se adequar as necessidades do projeto. É no pipeline que são definidos quando os testes rodam, quando os procedimentos de deployment são executados ou quando uma análise de código é necessária.

Na CAPES, um pipeline básico deverá conter as seguintes fases para cada tipo de branch segundo o gitflow:

### 3.3.5.4.1 Feature branch

São executados os seguintes passos a depender do suporte da linguagem:

- *Lint do código* – analisar se o código foi bem construído segundo os padrões estabelecidos para a linguagem ou pela CAPES.
- *Testes unitários* – executar testes unitários codificados na aplicação.
- *Analise de vulnerabilidades (SAST)* – executar analise de dependências e se não possui vulnerabilidades reportadas no CVE.
- *Compilação* – executar resolução de dependências, building da aplicação, para os casos de aplicações compiladas, e empacotamento em container para execução no OpenShift.
- *Review* – acionamento manual e sob demanda. Criar ambiente que executa a aplicação para uma visualização da funcionalidade implementada.
- *Limpeza* – acionamento manual e sob demanda. Remover ambiente de revisão.

### 3.3.5.4.2 hotfix branch, release branch

- *Lint do código* – analisar se o código foi bem construído segundo os padrões estabelecidos para a linguagem ou pela CAPES.
- *Testes unitários* – executar testes unitários codificados na aplicação.
- *Compilação* – execuatar resolução de dependências, building da aplicação, para os casos de aplicações compiladas, e empacotamento em container para execução no OpenShift.
- *Testes de integração / aceitação / pentest (DAST)* – executar testes de integração ou aceitação (via Selenium) em conjunto com *pentest* (via OWASP zed Attack proxy) para analise DAST.
- *Testes de carga / performance* – executar testes de carga e de analise de tempo de resposta da aplicação
- *Review* – açãoamento manual e sob demanda. Permitir criação de um ambiente que executa a aplicação para uma visualização da funcionalidade implementada.
- *Limpeza* – açãoamento manual e sob demanda. açãoamento manual e sob demanda. Remover ambiente de revisão.

### 3.3.5.4.3 Tags

- *Compilação* – execuatar resolução de dependências, building da aplicação, para os casos de aplicações compiladas, e empacotamento em container para execução no OpenShift.
- *Deployment* – açãoamento manual e sob demanda. Publicar aplicação em ambiente de *staging*.
- *Liberação* – açãoamento manual e sob demanda. executar testes de liberação da aplicação para acesso externo.
- *Limpeza* – açãoamento manual e sob demanda. açãoamento manual e sob demanda. Remover ambiente de *Liberação*.

O arquivo de definição do pipeline é o `.gitlab-ci.yml`. Esse arquivo define a estrutura de execução do pipeline.

Abaixo vemos um exemplo de um pipeline do Gitlab:

```
gitlab-ci.yml 371 Bytes 
```

```
1 stages:
2   - Testes
3   - Deploy
4
5 Validação código PHP:
6   stage: Testes
7   script:
8     - composer install
9     - ./vendor/phpunit/phpunit/phpunit --testdox
10  except:
11    - tags
12
13 Deploy em homolog:
14   stage: Deploy
15   script:
16     - sh ./deploy-hom
17   only:
18     - homolog
19
20 Deploy em produção:
21   stage: Deploy
22   when: manual
23   script:
24     - sh ./deploy-prod
25   only:
26     - tags
```

Figura 3.8 – Exemplo de configuração de .gitlab-ci.yml.

Neste arquivo, ficam apeas as definições do *pipeline*. Devendo os scripts de execução das operações serem colocados na pasta *devops/pipeline-scripts*.

### 3.3.5.5 Scripts

Todos os procedimentos executados devem ser dispostos em *scripts*. Essa diretriz visa garantir a replicabilidade da execução dos procedimentos. Por isso, os scripts devem ser armazenados na DevOps.

#### 3.3.5.5.1 Scripts de banco

Os scripts de banco de dados têm como finalidade o controle das modificações necessárias para a correta execução da aplicação, quer seja para desenvolvimento, quer seja para execução de testes automatizados ou aplicação em produção.

Eles devem estar no formato SQL e estarem sob a pasta *devops/scripts-banco*. Deve ser utilizado o *Liquibase* para gerenciamento das modificações de banco de dados. Os arquivos contendo a instrução sql de modificações de estruturas (DDL) ou migração / ajustes de dados (DML) devem usar o padrão YYYY-MM-DD-HH-mm\_<descrição simples da modificacao>.sql, conforme recomendação do liquibase.

### 3.3.5.6 Operators

Operators é o novo padrão para gerenciamento do ciclo de vida de aplicações expostas no cluster Kubernetes.

A função principal de um *Operador* é encapsular o conhecimento de infraestrutura de uma aplicação. De forma a tornar mais simples os procedimentos necessários a manutenção da aplicação.

Existe o operator SDK que é a ferramenta a ser utilizada para o desenvolvimento de operators para a plataforma OpenShift.

Os operators são registrados no repositório do núcleo da arquitetura no Git da CAPES. (<https://git.capes.gov.br/cgs/narq/openshift/operators>)

O funcionamento de um operator permite estender as funcionalidades do OpenShift de forma a gerenciar objetos personalizados. Abaixo vemos um exemplo de um objeto personalizado:

```
1  apiVersion: "apps.capes.gov.br/v1"
2  kind: "Sistema"
3  metadata:
4    name: "financeiro"
5  spec:
6    servicos:
7      - imagem: "financeiro-web:1.1.8"
8      | replicas: 1
9      | recursos:
10        |   cpu: 400m
11        |   memoria: 2Gi
12        firewall:
13          portas-liberadas:
14            - nome: financeiro-ws
15            | porta: 8080
16            | protocolo: TCP
17            - nome: financeiro-web
18            | porta: 8080
19            | protocolo: TCP
20            saida:
21              - nome: acesso site serpro
22              | dns: serpro.gov.br
23            acesso-externo:
24              - rota: financeiro.capes.gov.br
25              | porta-liberada: financeiro-web
26
```

Figura 3.10 – Exemplo de configuração de Operator

Ao aplicar esse objeto customizado o sistema é capaz de implantar a versão 1.1.8 do sistema financeiro. De forma que as regras descritas nesse documento sejam aplicadas ao cluster OpenShift.

Para uma descrição completa da sintaxe desse arquivo verifique a versão mais recente da documentação do operador sistemas-capes, que encontra-se no Git da CAPES.

### 3.4 Controle de versão

Para o gerenciamento de versão de código, a CAPES adotará o Git e Gitflow Workflow (*Gitflow*) como fluxo de versionamento. No Gitflow, há dois branches essenciais: *develop* e *master*. O *develop* serve de ponto de integração das *Features* (funcionalidades) finalizadas e estáveis, as quais precisam ser integradas ao

projeto. O *develop* deverá ser composto apenas por código estável. O *master* tem dois propósitos: i) versionar o código a ser publicado em produção, e; ii) servir de ponto de partida para o código que será usado para correção em produção (*hotfix*).

Além dos *branches* *develop* e *master*, há, no Gitflow, mais 3 prefixos: *Feature*, *Release* e *Hotfix*.

*Feature*<sup>1</sup> representa uma marcação no tempo de versionamento que aponta para um conjunto de código. Uma *feature* é criada para a codificação de novas funcionalidades no projeto e tem origem a partir do *develop*.

*Release* representa um conjunto de *features* que serão disponibilizadas em produção. Uma *Release* é criada sempre a partir do *develop*.

*Hotfix* representa uma marcação no versionamento que surge sempre a partir do *master* para corrigir um código em produção.

### 3.4.1 Fluxo de trabalho

O fluxo de trabalho com o Gitflow descreve o passo a passo no gerenciamento de *branches*, sua origem, transição e fechamento <sup>2</sup> <sup>3</sup>.

---

<sup>1</sup> Uma *Feature* jamais deverá interagir com o *Master*

<sup>2</sup> A Capes fornecerá o endereço do repositório e credenciais em momento oportuno durante a execução do projeto.

<sup>3</sup> Sem prejuízo das etapas de QA e demais processos especificados neste documento.

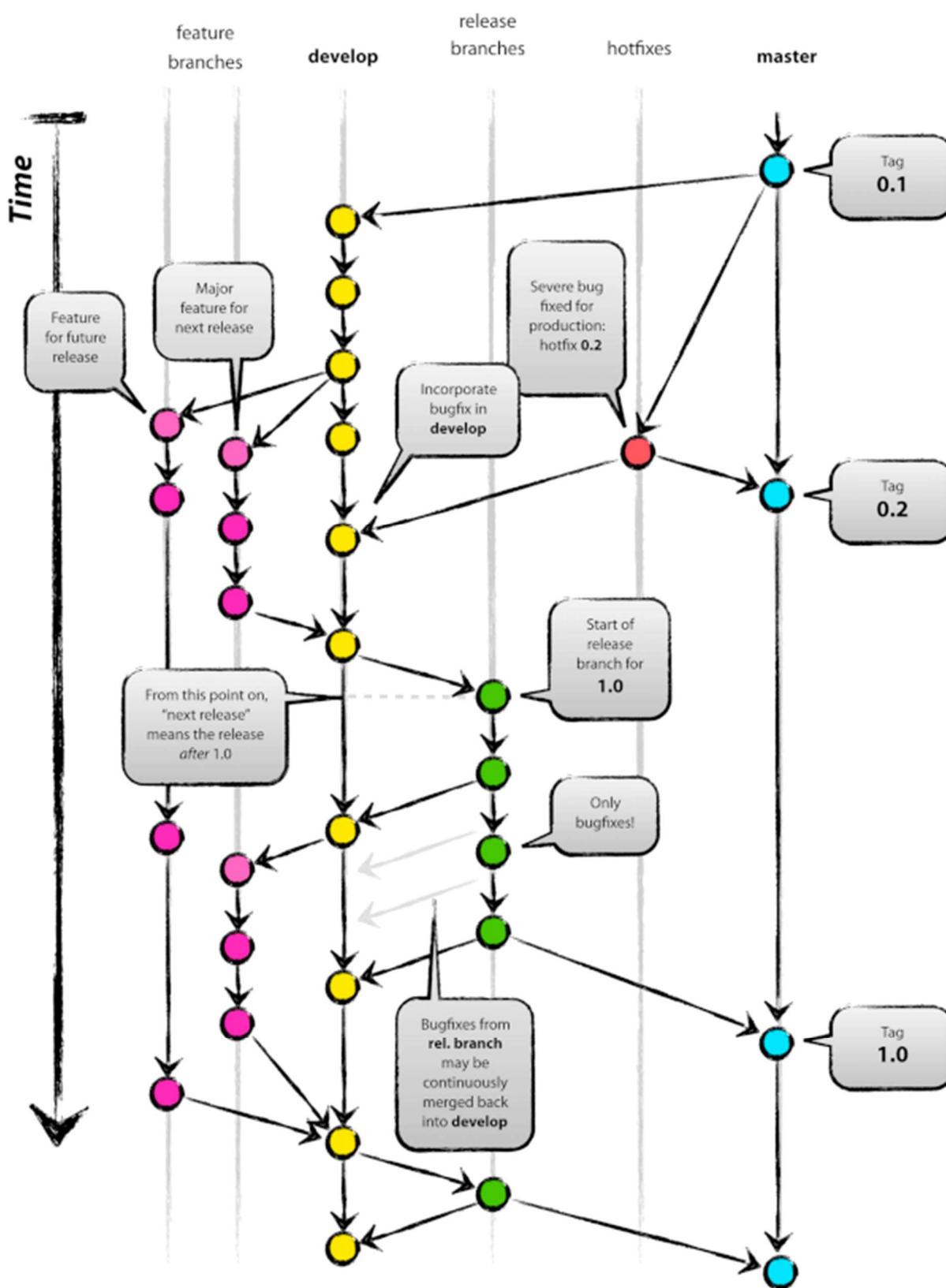


Figura 3.11 - Representação gitflow

Para cada funcionalidade, criará-se uma *feature branch*. A esta *feature*, poderá-se efetuar *commits* parciais até que seu código esteja pronto para se juntar ao *develop*.

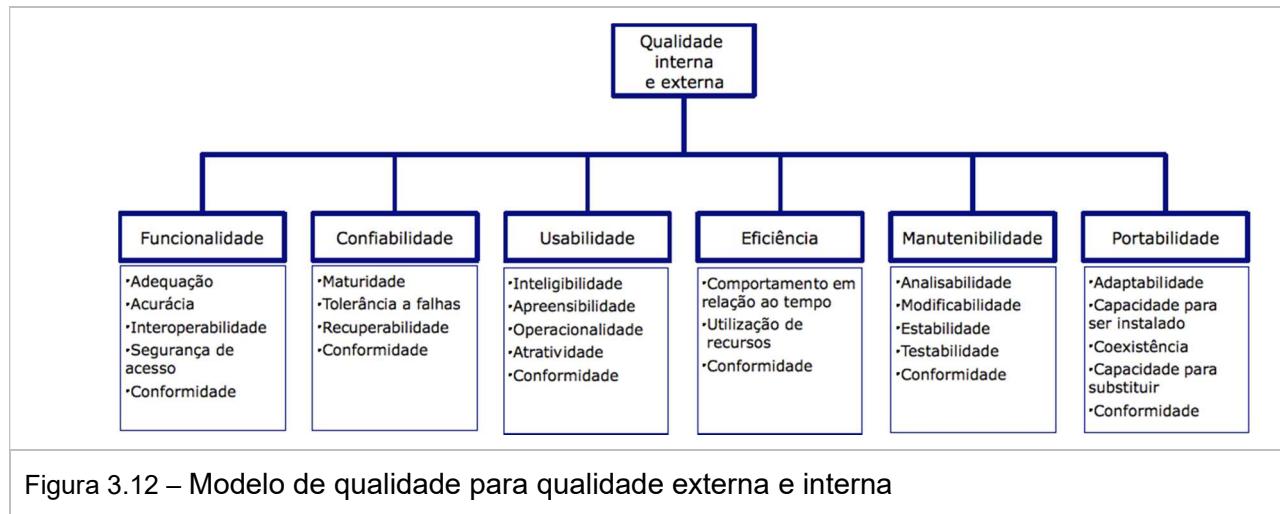
A entrega do *feature* se dá com o merge da *feature branch* no *develop*.

*Release branch* é usado para juntar um conjunto de funcionalidades a serem disponibilizadas em produção. Todas as possíveis correções necessárias a estabilização dessas funcionalidades, dar-se-ão diretamente na própria *release branch*. Tendo a *release branch* sido estabilizada, seu fechamento implicará em *merge* tanto no *master* quanto no *develop*.

Correções em código de produção, *master*, dar-se-á com o uso de *hotfix*. Ao término, fechamento do *hotfix*, implicará em *merge* tanto no *master* quanto no *develop*.

### 3.5 Requisitos de QA

Sem comprometimento dos itens relacionados à qualidade de software descritos noutros tópicos deste documento, a Capes, com o objeto de garantir a qualidade interna e externa de seus sistemas, adotará como requisitos de qualidade de software os elencados no item *Pré-requisitos de qualidades*. Tais requisitos visam a garantia dos tópicos do Modelo de qualidade para qualidade externa e interna definida na ISO/IEC 9126:2003 - Tópico 6 da norma, resumido na Figura 3.12.



#### 3.5.1 Pré-requisitos de qualidade

Cada um dos itens elencados na *tabela 00* será detalhado nos guias de arquitetura de tecnologia específica, bem como sua exigibilidade será avaliada pela CAPES conforme o projeto.

Requisito	Descrição	Entregável
Teste de Unidade	Teste de componente ou classe.	Sim
Teste de Integração	Teste que garante que componentes funcionem de forma combinada	Sim
Teste de Regressão	Teste que garante que uma alteração seja compatível com os demais componentes	Sim

Requisito	Descrição	Entregável
<b>Teste de Caixa Preta</b>	Teste que entrada e saída esteja em conformidade com o especificado	Sim
<b>Teste Funcional</b>	Teste de funcionalidades, requerimentos, regras de negócio	Sim
<b>Teste de Interface</b>	Teste da naveabilidade e os objetos de tela	Sim
<b>Teste de Performance</b>	Teste de tempo de resposta	Sim
<b>Teste de Carga</b>	Teste do sistema com um número elevado de usuários simultâneo	Sim
<b>Teste de Stress</b>	Teste de avaliação do tempo de resposta de uma ou mais funcionalidade em seu caminho feliz	Sim
<b>Teste de Configuração</b>	Teste de avaliação do funcionamento em diferentes ambientes de hardware/software	Sim
<b>Teste de Instalação</b>	Teste de instalação da aplicação se foi bem sucedida segundo o roteiro de instalação	Sim
<b>Teste de Segurança</b>	Teste de segurança da aplicação contra falhas conhecidas (CVE)	Sim

Tabela 3.4 – Pré-requisitos de qualidade

### 3.5.2 Automação

A automação de testes, quando aplicável, será detalhada no Guia de Arquitetura específico de cada linguagem de programação.

### 3.5.3 Cobertura de código por teste

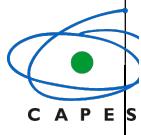
O percentual de cobertura de código será definido nos Guias de Arquitetura específicos de cada linguagem de programação (*Guia Arquitetura Java*, *Guia Arquitetura Javascript*, *Guia Arquitetura PHP*, *Guia Arquitetura Python*) e quanto cabível, elencado por camada da aplicação.

## 4 Referências

- ISO/IEC 9126 - <https://www.iso.org/standard/22749.html>
- The Art of Computer Programming: D. E. Knuth, Addison-Wesley (volumes 1--3, 4A), 1998.
- Spring Guides: <https://spring.io/guides>
- Hibernate Docs: <http://hibernate.org/orm/documentation/5.0/>
- Docker para desenvolvedores: Rafael Gomes, Leanpub, 2017

## 5 O Anexo A – Guias de arquiteturas

Nome do guia de referência	Tecnologia
Arquitetura de referência - Java	Java
Arquitetura de referência – PHP	PHP
Arquitetura de referência – Pyhton	Python
Arquitetura de referência - Frontend	-



## ARQUITETURA DE REFERÊNCIA - GUIA JAVA

**Versão: 1.1**

<b>Histórico de Revisões</b>			
<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Responsável</b>
15/04/2019	1.0	Criação do Guia Específico de JAVA	Harrysson Gilgamesh de Medeiros Nóbrega Thiago Adelino de Melo Valdecy Lourenço de Araújo Júnior
23/04/2019	1.0	Homologação do Guia Específico de JAVA	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Leonardo Morais Borges Théo Alves Monteiro Valdecy Lourenço de Araújo Júnior
13/05/2019	1.1	Alteração do quadro de métrica de qualidade no item 4 – Metas e restrições arquiteturais.	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Théo Alves Monteiro Davi Souza Rafael Jesus Nirian Martins S. dos Santos

## Sumário

1	INTRODUÇÃO.....	5
2	PROpósito.....	5
3	ESCOPO.....	5
4	METAS E RESTRIÇÕES ARQUITETURAIS .....	5
4.1	CAMADA.....	7
4.1.1	CAMADA DE RECURSOS .....	8
4.1.2	CAMADA DE NEGÓCIOS .....	8
4.1.3	CAMADA DE ACESSO A DADOS .....	8
4.2	FRONTEND.....	9
4.3	BACKEND .....	9
4.4	PADRÕES .....	10
4.4.1	TECNOLOGIA.....	10
4.4.2	COMPONENTE .....	10
4.4.3	NOMENCLATURA.....	10
5	VISÃO LÓGICA.....	10
6	VISÃO DE IMPLEMENTAÇÃO .....	10
6.1	MICROSSERVIÇOS .....	10
6.2	DECOMPOndo APLICAÇÕES EM SERVIÇOS.....	11
6.3	IMPLANTAR MICROSSERVIÇOS DEPENDE DE UM BOM PARTICIONAMENTO .....	12
6.4	AS VANTAGENS DA ARQUITETURA DE MICROSSERVIÇOS .....	13
7	VISÃO DE PROCESSOS .....	13
8	VISÃO DE DADOS .....	13
9	VISÃO DE IMPLANTAÇÃO.....	13
10	TAMANHO E DESEMPENHO .....	14
11	REQUISITOS DE QUALIDADE .....	14
12	DEFINIÇÕES, ACRÔNIMOS E ABREVIACÕES .....	16
13	REFERÊNCIAS .....	17

14	ANEXO A – TECNOLOGIAS.....	18
15	ANEXO B – COMPONENTES.....	18

## 1 Introdução

Este documento é uma especialização do **Guia Geral de Arquitetura** e tem natureza subsidiária ao Guia Geral.

## 2 Propósito

Esse documento tem por propósito fornecer uma visão arquitetural abrangente para ser usada no desenvolvimento de sistemas. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da instituição. Assim, não tem a intenção de ser um documento definitivo de arquitetura para cada sistema individual. Ao invés disso, tenta traçar os objetivos de alto nível que levem a uma boa arquitetura de software. Cada sistema em particular deve ser analisado junto com a equipe de desenvolvimento e um dos arquitetos de sistemas da CAPES. Se os tópicos aqui apresentados forem seguidos, as chances de sucesso nos projetos tendem ser maiores.

## 3 Escopo

Fornecer uma visão arquitetural abrangente para ser usada no desenvolvimento de sistemas. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da instituição.

## 4 Metas e Restrições Arquiteturais

Na década de 70 os padrões de projetos era um conceito da arquitetura, descrito no livro *A Pattern Language* escrito pelo Christopher Alexander. Nos anos 80 Kent Beck (o criador do Extreme Programming, Test Driven Development e um dos signatários originais do Agile Manifesto) e Ward Cunningham começaram a aplicar esse conceito na área da computação.

A arquitetura em camadas é um estilo arquitetural que propõe dividir um sistema em várias camadas de acordo com a responsabilidade de cada parte do software. Este estilo propõe que:

- As camadas tenham um propósito bem definido;
- Cada camada conheça apenas camadas abaixo dela;
- As camadas possam ser reaproveitáveis.

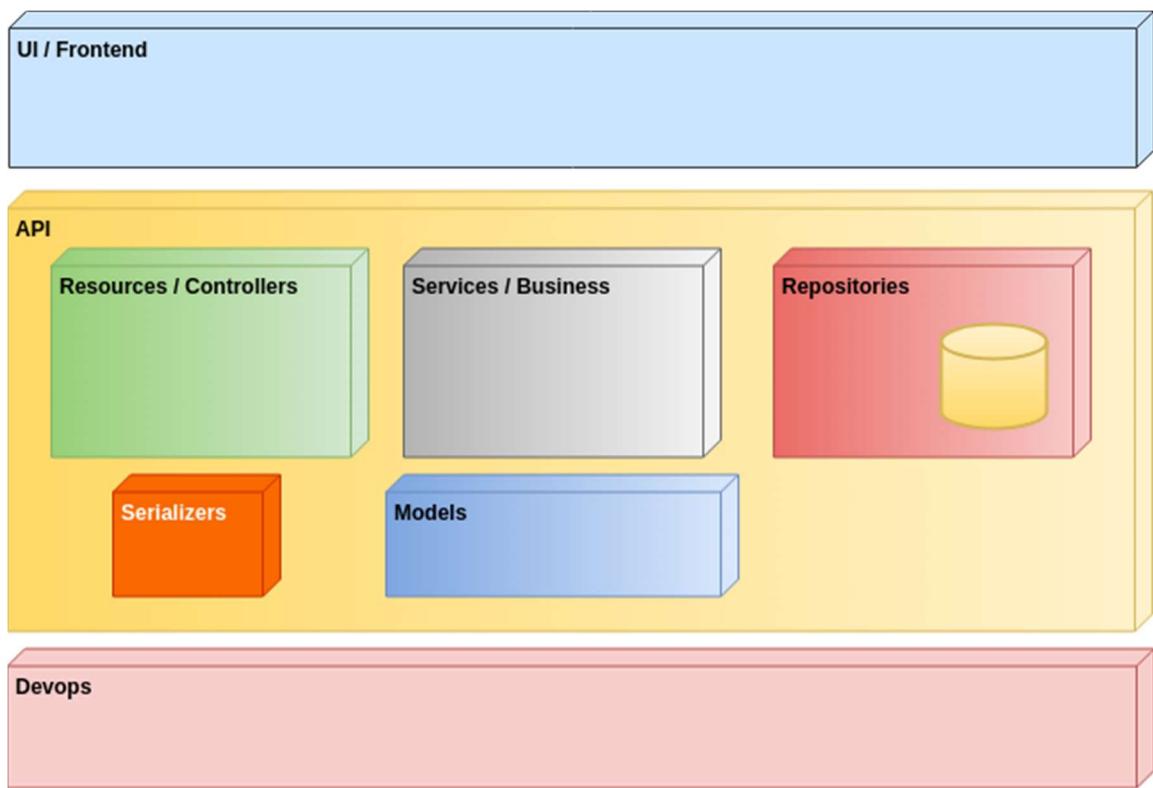


Figura 1 Arquitetura de aplicação Java

A separação de dados e comportamento leva a programação procedural e como consequência a perda de vantagens básicas da orientação a objetos como reutilização de código, encapsulamento, herança e polimorfismo. Esse modelo de programação é conhecido como modelo anêmico e devemos evitá-lo.

É preciso destacar também a importância da cobertura de testes, que traz várias vantagens para o projeto, como a maior qualidade dos sistemas entregues, facilidade de manutenção, diminuição de tempo para encontrar e corrigir problemas entre vários outros benefícios. Todos os novos sistemas devem ter suas principais funcionalidades testadas através de testes unitários ou de integração.

É importante lembrar que cada camada deve ser desenvolvida de forma independente. A camada de negócio não deve mudar seu comportamento, nem deve ser dependente de uma tecnologia que a camada de usuário utiliza. O mesmo não deve ocorrer na comunicação entre cada uma dessas camadas. Isso facilita a implementação dos testes automatizados, visando cada camada em específico.

Os sistemas desenvolvidos deverão utilizar HTTPS nas comunicações com o usuário externo com a finalidade de garantir maior segurança no trâmite das informações.

Para promover a integração entre os sistemas e facilitar a usabilidade os sistemas deverão utilizar o Segurança SSO como mecanismo de login único para as aplicações.

Os indicadores de qualidade que cada sistema deverá apresentar ao ser submetido à avaliação por meio de ferramenta Sonarqube. Cada indicador representa uma meta a ser alcançada. No que tange ao código-fonte, o quadro a seguir relaciona os indicadores e sua respectiva meta.

Grupo	Indicador	Tipo Meta	Meta
PROJETO	Problemas confirmados	Unidades	= 0
	Complexidade	Média total	<= 10
	Métodos	Média total	<= 3
	Índice de manutenibilidade	Nota	A
	Índice de confiabilidade	Nota	A
	Índice de segurança	Nota	A
	Taxa de dívida técnica	%	<= 2,5%
	Classes	Média total	<= 10
	Arquivos	Média total	<= 10
VIOLAÇÕES	Linhas duplicadas (%)	%	<= 4%
	Problemas impeditivos	Unidades	= 0
	Problemas críticos	Unidades	= 0
TESTE	Testes unitários ignorados	Unidades	= 0
	Sucesso em testes unitários (%)	%	= 100%
	Cobertura (camada de negócio)	%	>= 70%

Em alinhamento com o Guia Geral, a arquitetura de aplicações Java é projetada para o desenvolvimento em microsserviços e é agrupada em dois segmentos: API e DevOps.

- API é o agrupamento dos pontos de consumo da aplicação que podem ser consumidos por outros sistemas ou interfaces de usuários especializadas. Constitui a fronteira entre o *backend* e *frontend*.
- DevOps é o conjunto de scripts e artefatos necessários à automação do processo *build/deploy* da aplicação usando, no contexto da CAPES, o OpenShift.

#### 4.1 Camada

Conforme mostrado na *Figura 1 Arquitetura de aplicação Java*, o primeiro seguimento da arquitetura, API, agrupa as camadas que compõe às aplicações nessa linguagem. Nesse contexto, cada camada exerce papel específico e bem definido no plano de codificação. Suas responsabilidades são agrupadas em 3 grupos:

- **Resources Layer / Controllers / Recursos:** Essa camada é responsável por separar a estrutura de acesso dos dados, da camada de apresentação do sistema. Ela centraliza a complexidade requisições e retornos da API Rest. Esta camada adere ao item 3.2.1.1.9 do guia geral.
- **Business Layer / Services / Serviços:** Responsável por centralizar e validar as regras negociais utilizadas para atingir o objetivo da aplicação. Esta camada também tem como objetivo centralizar o acesso a serviços externos, sendo estes em sua maioria serviços web Rest e SOAP. Como exemplo de serviço externo consumido, está o Cadastro de Pessoas, o

Serviço de Documentos e o Serviço de Pessoas da Receita. Esta camada adere ao item 3.2.1.2.3 do guia geral.

- **Repositories Layer / Camada de banco de dados:** Representa o SGBD, ou local de persistência dos dados. Podendo ser um banco de dados relacional (Oracle, Postgresql) ou não relacional (CouchDB, MongoDB). Esta camada adere ao item 3.2.1.1.4 do guia geral.

#### 4.1.1 Camada de recursos

Nessa camada devem ser implementadas o conjunto de rotinas e padrões estabelecidos e documentados da API, ela é o ponto de partida das funcionalidades do microserviço.

Deve ser utilizado o Java EE, ou equivalente no Spring Boot, para descrever a API e expor sua documentação corretamente utilizando o Swagger.

Deve-se criar um ResourceS, que descreve e implementa o acesso à camada de negócios, processa os dados, serializa e retorna um Response contendo o conteúdo do modelo serializado solicitado ou gera uma exceção como Http404.

#### 4.1.2 Camada de negócios

Todo acesso à camada de Banco de dados deve ser feito a partir da camada de negócio, é nela também que são verificadas e validadas as regras negociais, podendo emitir exceções se for o caso. Devem ser consumidos todos os tipos de serviços externos, independente se são serviços web usando Rest, ou SOAP. Essa camada também tem como objetivo abstrair o consumo desses recursos, deixando visível para as outras camadas apenas os parâmetros de requisição e o retorno esperado.

Para as operações que envolvem manipulação de dados é nesta camada que devem ser controladas as transações.

#### 4.1.3 Camada de acesso a dados

Implementação do modelo de dados no Java é definido em conjunto com a equipe de banco de dados e suas boas práticas. Na CAPES os sistemas são totalmente isolados, cada um tem um esquema de banco de dados diferente, um sistema não deve ter acesso à dados de outro sistema via banco de dados, a maneira que estes dados devem ser compartilhados estão descritos no item "Camada de negócio" deste documento.

Todo acesso ao banco de dados no esquema do sistema é feito através da camada de acesso a dados. As consultas DML de recuperação, inclusão, remoção e modificação de informações em bancos de dados devem feitas através de uma técnica de mapeamento objeto relacional que permite fazer uma relação dos objetos com os dados que os mesmos representam (Object Relational Mapper) utilizando a especificação JPA2 implementada pela framework Hibernate.

O objetivo da especificação JPA é facilitar o mapeamento dos atributos entre uma base tradicional de dados relacionais e o modelo objeto de uma aplicação, utilizando-se de anotações.

JPA 2 permite também ao desenvolvedor realizar a criação das queries utilizando-se de sua própria API de Criteria orientada a objetos, facilitando a abstração da lógica de consulta, além de implementar a JPA

Criteria API, o Hibernate também possui sua própria API de Criteria. JPA 2 também especifica uma DSL mais próxima do SQL chamada de JPQL. JPQL lida acessando as propriedades do objeto e não as colunas na tabela. O Hibernate além de implementar JPQL tem o HQL: sua DSL que estende SQL onde assemelha-se ao próprio SQL. E por fim o SQL, consultando-se diretamente usando a estrutura presente no banco de dados.

A biblioteca assume o trabalho de automatizar tarefas redundantes enquanto o desenvolvedor permanece no controle de como o banco de dados é organizado e como o SQL é construído.

#### 4.2 Frontend

Não se aplica.

#### 4.3 Backend

Sem prejuízo à definição do Guia Geral 3.1.3. A seguir, é definido como uma aplicação Java será executada em ambiente CAPES.

Com o advento da automatização e devida inteligência na manutenção das aplicações, hoje, é esperado que a aplicação possa atender a picos de demandas com inicialização automática de novos processos, sem afetar seu comportamento.

A boa prática indica que processos de aplicações são stateless (não armazenam estado) e *share-nothing*. Quaisquer dados que precisem persistir devem ser armazenados em serviço de apoio *stateful* (armazena o estado), normalmente é usado uma base de dados. O objetivo final dessa prática não faz distinção se a aplicação é executada na máquina do desenvolvedor ou em produção, pois, nesse caso, o que muda é a quantidade de processos iniciados para atender as respectivas demandas.

É importante salientar: ao seguir a prática, uma aplicação não assume que, qualquer item armazenado em cache de memória ou no disco, estará disponível em futura solicitação ou *job* – com muitos processos de cada tipo rodando, são altas as chances de futura solicitação ser servida por processo diferente, até mesmo em servidor diferente. Mesmo quando, rodando em apenas um processo, um *restart* (desencadeado pelo *deploy* de um código, mudança de configuração, ou o ambiente de execução realocando o processo para localização física diferente) geralmente vai acabar com o estado local (memória e sistema de arquivos, por exemplo).

Durante o processo de desenvolvimento de uma aplicação é difícil imaginar o volume de requisição que ela terá no momento que for colocada em produção. Por outro lado, um serviço que suporte grandes volumes de uso é esperado nas soluções modernas. É frustrante solicitar acesso a uma aplicação e ela não estar disponível. Sugere falta de cuidado e profissionalismo, na maioria dos casos. Quando a aplicação é colocada em produção, normalmente é dimensionada para determinada carga esperada, porém é importante que o serviço esteja pronto para escalar. A solução deve ser capaz de iniciar novos processos da mesma aplicação, caso necessário, sem afetar o produto.

A CAPES adotará as seguintes opções de *Deployment* para aplicações Java utilizando-se das soluções de servidor embutido do Spring Boot, como é o caso do Tomcat, Jetty e Undertow. Essa abordagem

é recomendada na utilização de *containers*, visto que o servidor de aplicação passa a ser uma parte da aplicação.

#### 4.4 Padrões

Definições adotadas pela CAPES com base em suas necessidades para criação, manutenção e evolução de seus sistemas.

##### 4.4.1 Tecnologia

O Anexo A - Tecnologias elenca as tecnologias adotadas pela CAPES de forma não exaustiva para a pilha de desenvolvimento Java. A inclusão ou remoção de itens nele, deverá ser definida pela CAPES.

##### 4.4.2 Componente

O Anexo B- Componentes elenca os componentes adotados pela CAPES para a pilha de desenvolvimento Java. A inclusão ou remoção de itens nele, deverá ser definido pela CAPES.

##### 4.4.3 Nomenclatura

Nas referências deste documento existe um link para o guia de estilo do Java, definido e utilizado atualmente pela Google. Existe a padronização da codificação do código (UTF-8) e regras de estilo afim de manter o código organizado e legível.

### 5 Visão lógica

Vide correlação no item 4.1.

### 6 Visão de implementação

Java é não só uma linguagem de programação, mas uma plataforma de desenvolvimento. Ela possui estruturas de dados de alto nível eficientes, bem como adota uma abordagem simples e efetiva para a programação orientada a objetos. Os grupos de usuário Java são muito utilizados, além de ser possível obter materiais de estudo de forma fácil. Esses grupos são fortes e bem distribuídos, tanto em língua inglesa, quanto na língua portuguesa. Além disso a comunidade de desenvolvimento Java possui uma vasta gama de frameworks e bibliotecas. Acelerando o processo de desenvolvimento, utilizando-se do que os outros já implementaram e testaram, para a resolução de problemas.

O Spring Boot é um *framework* vasto que possui várias soluções já implementadas, com a opção de utilização parcial. Engloba desde a parte de autenticação, operações de banco de dados, até a parte de automação de testes.

O Spring Boot também possui solução de servidor de aplicação embutido, permitindo que a aplicação seja executada como uma aplicação Java convencional, sem a necessidade de se realizar o *deploy* através do servidor de aplicação externo.

#### 6.1 Microsserviços

Tradicionalmente, softwares é construído como uma estrutura fechada, com começo, meio e fim. Ou melhor, *back-end* e *front-end*.

Essa filosofia acaba gerando peças muito grandes (Arquitetura Monolítica), normalmente focadas em resolver grandes necessidades e problemas organizacionais. Se você quiser usar somente as funções que fazem gráficos do Excel para alguma outra aplicação, terá que abrir o Excel e carregar outras funcionalidades desnecessárias naquele momento.

A Arquitetura de Microserviços, ao contrário das anteriores de Arquitetura Monolítica ou até SOA, destaca-se por explorar a ideia de granularidade, o que facilita a execução do próprio serviço e a adaptação às mudanças.

A ideia é dividir um determinado sistema em serviços acionáveis e modulares, de modo que a união de pequenas partes realize um trabalho maior. Assim, os microserviços permitem a integração entre vários serviços e a inserção de vários componentes no sistema.

## 6.2 Decompondo aplicações em serviços

A forma mais usual para escalar uma aplicação é executando várias cópias idênticas de aplicação (por meio de um平衡ador de cargas), realizando um processo chamado Decomposição.

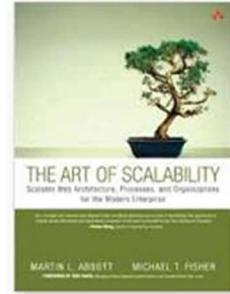
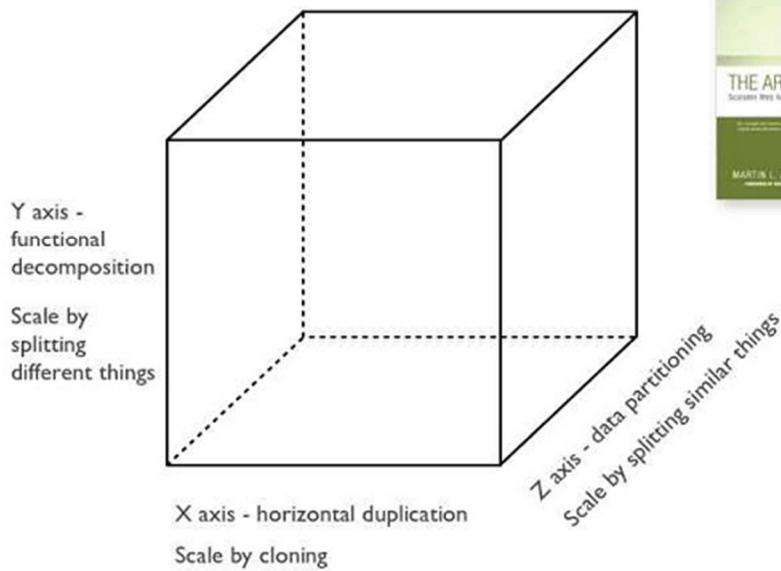
A finalidade da decomposição é resolver problemas comuns na arquitetura monolítica, implicando em que alguns serviços poderão ser muito pequenos enquanto outros serão significativamente maiores.

No conceito de três dimensões, é possível entender como os Microserviços são úteis no particionamento e escalabilidade de uma aplicação monolítica. Temos três eixos de escalabilidade:

- X: referente à escalabilidade horizontal, para ampliar a capacidade e disponibilidade da aplicação (cada servidor executa uma cópia idêntica do código);
- Z: semelhante à do eixo X, mas requer a presença de um componente que se responsabilize pelo roteamento das requisições ao servidor adequado;
- Y: é a terceira dimensão da escalabilidade, a horizontal, denominada decomposição funcional e é responsável por dividir a aplicação em uma série de serviços. A cada serviço corresponde um conjunto de funções (gerenciamento de pedidos, gerenciamento de clientes e assim por diante).

Lembrando que, enquanto o eixo Z divide elementos semelhantes, o eixo Y divide elementos distintos.

## 3 dimensions to scaling



### 6.3 Implantar Microsserviços depende de um bom particionamento

A divisão, ou particionamento, do sistema deve ser bem elaborada, segue as abordagens que adotamos para implantar a Arquitetura de microsserviços:

- **Por verbos/casos de uso:** o caso *checkout*, por exemplo, em que um serviço de conclusão de pedidos, ou *Checkout UI service*, implementa a interface com as pessoas que usam o sistema.
- **Por sinônimos/recursos:** como exemplo, considere que, para gerenciamento do catálogo de mercadorias, a empresa pode ter o *Catalog Service*. Nesse caso, o serviço se responsabiliza por todas as atividades que envolvem os recursos/entidades relacionados.

O importante é que cada serviço possua poucas responsabilidades, de acordo com o Princípio de Responsabilidade Única (SRP, Single Responsibility Principle). Ou seja, um serviço exposto deve ter um único e claro papel dentro da Arquitetura.

Se for o caso em que um serviço possui mais de uma responsabilidade, deve-se aplicar algum particionamento, como citado acima.

As funcionalidades Unix constituem outro exemplo de modelagem de serviços, em que cada funcionalidade realiza somente uma operação definida (podendo, no entanto, ser combinada, por meio de *shell script*, com outras funcionalidades a fim de realizar atividades mais complexas). Ao longo dos anos, esse baixo acoplamento facilitou que diversas variações do sistema operacional fossem lançadas, como Ubuntu, Fedora, Solaris e tantas outras.

## 6.4 As vantagens da Arquitetura de microsserviços

- Os desenvolvedores usufruem de liberdade maior para o desenvolvimento de serviços de modo independente;
- Implantação automática através de ferramentas de integração contínua e código aberto, como Hudson, Jenkins e outras;
- O contêiner web tem inicialização mais rápida;
- Possibilidade de utilizar códigos escritos em linguagens diferentes para diferentes serviços, usando uma “língua franca” para comunicação entre eles (como Json ou XML);
- Oportunidade para os desenvolvedores usarem as tecnologias mais atuais;
- Arquitetura de fácil compreensão e bastante adaptável às mudanças, o que favorece o aprendizado dos profissionais novatos, contribuindo para maior produtividade da equipe;
- Fácil ampliação e integração dos microsserviços com serviços terceirizados, através de APIs, por exemplo;
- Código organizado em função de capacidades de negócio, dando mais visão das ofertas e necessidades dos clientes;
- Mudanças necessárias poderão ser aplicadas somente sobre o serviço específico, sem necessidade de modificar todo o aplicativo. Atualizações de funcionalidades também passam a ser menos complexas;
- Gerenciamento otimizado das falhas (por exemplo, caso um serviço venha a falhar, os outros continuarão trabalhando).

## 7 Visão de processos

Não se aplica.

## 8 Visão de dados

As aplicações desenvolvidas na CAPES utilizam como mecanismo de persistência os SGBs relacionais. As aplicações em Java, fazem uso do *framework* *Hibernate* para acessar os dados dos bancos. São necessários alguns cuidados a fim de garantir que a aplicação seja estável e não venha a causar problemas no acesso aos dados:

- Preferir utilizar JPA Criteria API, ou JPQL, ao SQL puro (evitando dependência de banco). A API de Criteria ou HQL do Hibernate não são encorajados, mas podem ser notados como disponíveis.
- Usar corretamente o Lazy Loading, para trazer a maior quantidade de dados possíveis que realmente serão usados, com a finalidade de evitar 1+n consultas ao banco.
- Sempre que possível usar consultas paginadas para melhorar o desempenho.

## 9 Visão de implantação

Atualmente, os ambientes não produtivos e produtivos da Capes estão disponibilizados via Openshift, de modo que é possível baixar uma imagem de um contêiner Docker, homologada pela equipe de infraestrutura e iniciar o desenvolvimento da mesma, sem a necessidade que seja provido um ambiente específico para desenvolvimento ou servidor pré-instalado.

Deste modo, o desenvolvedor não necessita configurar ambiente, instalar servidor, ou configura-lo. Tudo está disponível direto na imagem Docker nos repositórios da Capes

Para a geração do pacote de deploy da aplicação, a Pipeline do GitLab se integra com o Jenkins. No próprio GitLab é feita a criação de uma TAG que represente o código usado nessa versão do sistema. A cada execução dessa tarefa é feita uma análise estática de código através do Sonar. E finalmente, o pacote gerado é disponibilizado no contêiner que seguirá todos os passos de aceitação para ser disponibilizado em ambiente produtivo.

Na *wiki*, existe uma página que detalha as configurações que devem ser feitas pelo desenvolvedor na sua máquina e no projeto. A página está disponível em [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy).

Cada projeto deverá ter um pacote chamado devops em seu repositório, mantido pela equipe de arquitetura e pelo líder técnico da equipe. Neste pacote deve conter os arquivos de CI e CD da aplicação.

## 10 Tamanho e Desempenho

Caso a aplicação tenha processamento em lote ou execuções agendadas, essas operações devem, preferencialmente, ser realizadas em servidores separados dos servidores destinados a atender a requisições dos usuários. Devido aos custos de processamento destes recursos a resposta do usuário pode ser onerada caso estejam no mesmo servidor.

As requisições devem ser atendidas o mais rápido possível. Processamentos de pedidos de usuários que gerem demora perceptível na request deve ter o seu processamento efetuado em *background* e o usuário deve ser notificado da conclusão da tarefa.

Preferir a separação do *front-end* e do *back-end*, visando facilitar a escalabilidade de recursos e o desacoplamento entre tecnologias com fins diferentes.

## 11 Requisitos de Qualidade

Todos os sistemas passarão por análise estática de código através do Sonar. Alguns pontos que serão mais cuidadosamente analisados com base nos relatórios do sonar são: cobertura de testes, complexidade ciclomática, acoplamento e coesão.

O acesso ao Sonar é livre para todos os desenvolvedores e deve ser incentivado dentro das equipes como meio de adequação e qualidade de código.

Com a finalidade de garantir uma boa qualidade do código desenvolvido, os desenvolvedores devem usar o BDD (Design/Desenvolvimento guiado por comportamento), que em resumo é uma técnica voltada para o comportamento da aplicação e é comumente usada para testes.

Como forma de auxiliar no desenvolvimento dos códigos, devem ser utilizadas, bibliotecas que facilitem a atividade de teste, entre elas:

- **JUnit**- usado para escrever os testes de forma que permita a sua execução automática.

- **Selenium** - ferramenta usada para mapear a navegação do usuário, usada para escrever os testes de aceitação.
- **Arquillian** - ferramenta usada para executar os testes de integração, auxiliando em testes que precisem de um banco ou recursos de container.
- **Mockito**- usado para realizar mocks e stubs dos objetos, auxiliando o isolamento do código, para testes.
- **Hamcrest**- biblioteca utilizada para auxiliar na escrita de asserções dos testes escritos, melhorando assim a legibilidade do teste.

## 12 Definições, Acrônimos e Abreviações

- Jenkins - <http://jenkins.capes.gov.br>
- Nexus - <http://nexus.capes.gov.br>
- Sonar - <http://sonar.capes.gov.br>
- GITLab- <http://git.capes.gov.br>
- Wiki - <http://wiki.capes.gov.br>
- Openshift - <http://openshift.capes.gov.br>
- Spring Boot- <http://spring.io>
- JUnit- <http://junit.org>
- Mockito - <https://site.mockito.org/>
- Swagger - <https://swagger.io/>
- Angular - <https://angular.io/>
- React - <https://reactjs.org/>
- Vuejs - <https://vuejs.org/>

## 13 Referências

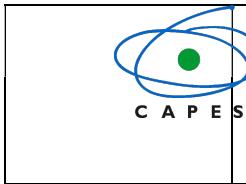
- [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy)
- [https://wiki.capes.gov.br/index.php/DTI:Visao\\_geral\\_tecnologias\\_integradas](https://wiki.capes.gov.br/index.php/DTI:Visao_geral_tecnologias_integradas)
- Google Java Style Guide: <https://google.github.io/styleguide/javaguide.html>
- The Art of Computer Programming: D. E. Knuth, Addison-Wesley (volumes 1--3, 4A), 1998.
- DEITEL, Paul J.; DEITEL, Harvey M. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2017.
- Spring Guides: <https://spring.io/guides>
- Hibernate Docs: <http://hibernate.org/orm/documentation/5.0/>
- Docker para desenvolvedores: Rafael Gomes, Leanpub, 2017

**14 Anexo A – Tecnologias**

Tecnologia	Versão
Java	>=1.8
Spring-boot	>=2.0
JUnit	>=5
Mockito	>=2
Arquillian	>=1.4.0
Hibernate	>=5

**15 Anexo B – Componentes**

NOME	VERSÃO	DESCRIÇÃO
-	-	-



## ARQUITETURA DE REFERÊNCIA - GUIA PHP

**Versão: 1.1**



<b>Histórico de Revisões</b>			
<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Responsável</b>
07/03/2019	1.0	Criação do Guia Específico de PHP	Harrysson Gilgamesh de Medeiros Nóbrega José Augusto de Jesus Thiago Adelino de Melo Valdecy Lourenço de Araújo Júnior
28/03/2019	1.0	Homologação do Guia Específico de PHP	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Leonardo Morais Borges Théo Alves Monteiro Valdecy Lourenço de Araújo Júnior
13/05/2019	1.1	Alteração do quadro de métrica de qualidade no item 4 – Metas e restrições arquiteturais.	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Théo Alves Monteiro Davi Souza Rafael Jesus Nirian Martins S. dos Santos

## SUMÁRIO

1. INTRODUÇÃO.....	5
2. PROPÓSITO .....	5
3. ESCOPO .....	5
4. METAS E RESTRIÇÕES ARQUITETURAIS .....	5
4.1 CAMADA .....	6
4.1.1 CAMADA DE RECURSOS .....	7
4.1.2 CAMADA DE NEGÓCIO .....	7
4.1.3 CAMADA DE ACESSO A DADOS.....	7
4.2 FRONTEND .....	8
4.3 BACKEND .....	8
4.4 PADRÕES .....	10
4.4.1 TECNOLOGIA .....	10
4.4.2 COMPONENTE.....	11
4.4.3 NOMENCLATURA .....	11
5. VISÃO LÓGICA .....	11
6. VISÃO DE PROCESSOS .....	11
7. VISÃO DE IMPLANTAÇÃO .....	11
8. VISÃO DE IMPLEMENTAÇÃO .....	11
8.1 MICROSSERVIÇOS .....	12
8.2 DECOMPONDÔ APLICAÇÕES EM SERVIÇOS.....	12
8.3 IMPLANTAR MICROSSERVIÇOS DEPENDE DE UM BOM PARTICIONAMENTO.....	14
9. TAMANHO E DESEMPENHO .....	14
10. REQUISITOS DE QUALIDADE.....	15
11. DEFINIÇÕES, ACRÔNIMOS E ABREVIACÕES .....	17

 C A P E S	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

<b>12. REFERÊNCIAS .....</b>	<b>18</b>
<b>ANEXO A – TECNOLOGIAS .....</b>	<b>20</b>
<b>ANEXO B – COMPONENTES .....</b>	<b>21</b>

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1  Data Emissão: 13/05/2019
---	---	--

## 1. Introdução

Este documento é uma especialização do **Guia Geral de Arquitetura** e tem natureza subsidiária ao Guia Geral.

## 2. Propósito

Apresentar arquitetura para o desenvolvimento e sustentação de sistemas PHP na CAPES. Detalhar os aspectos técnicos relativos à adoção de tecnologias, ao desenvolvimento e à implantação de sistemas. Capturar e formalizar as principais decisões tomadas com relação à arquitetura de sistemas que tenham o PHP como base. Assim, não tem a intenção de ser um documento definitivo de arquitetura para cada sistema individual. Ao invés disso, tenta traçar os objetivos de alto nível que levem a uma boa arquitetura de *software*. Cada sistema em particular deve ser analisado junto com a equipe de desenvolvimento e um dos arquitetos de sistemas da CAPES.

## 3. Escopo

Fornecer uma visão arquitetural usada no desenvolvimento de sistemas PHP. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da CAPES.

## 4. Metas e Restrições Arquiteturais

Indicadores de qualidade que cada sistema deverá apresentar ao ser submetido à avaliação. Cada indicador representa uma meta a ser alcançada. No que tange ao código-fonte, o quadro a seguir relaciona os indicadores e sua respectiva meta.

Grupo	Indicador	Tipo Meta	Meta
PROJETO	Problemas confirmados	Unidades	= 0
	Complexidade	Média total	<= 10
	Métodos	Média total	<= 3
	Índice de manutenibilidade	Nota	A
	Índice de confiabilidade	Nota	A
	Índice de segurança	Nota	A
	Taxa de dívida técnica	%	<= 2,5%
	Classes	Média total	<= 10
	Arquivos	Média total	<= 10
VIOLAÇÕES	Linhas duplicadas (%)	%	<= 4%
	Problemas impeditivos	Unidades	= 0
TESTE	Problemas críticos	Unidades	= 0
	Testes unitários ignorados	Unidades	= 0

Grupo	Indicador	Tipo Meta	Meta
	Sucesso em testes unitários (%)	%	= 100%
	Cobertura (camada de negócio)	%	>= 70%

Em alinhamento com o Guia Geral, a arquitetura de aplicações PHP é projetada para o desenvolvimento em microsserviços e é agrupada em dois segmentos: API e DevOps.

- API é o agrupamento dos pontos de consumo da aplicação que podem ser invocados por outros sistemas ou interfaces de usuários especializadas. Constitui a fronteira entre o *backend* e *frontend*.
- DevOps é o conjunto de scripts e artefatos necessários à automação do processo build/deploy da aplicação usando pelo OpenShift.

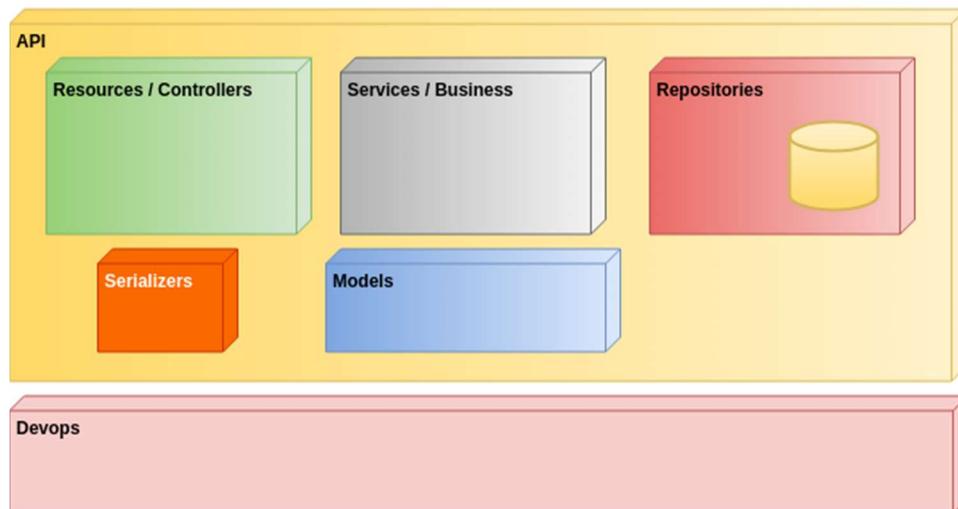


Figura 1 Arquitetura de aplicação PHP

#### 4.1 Camada

Conforme mostrado na *Figura 1 Arquitetura de aplicação PHP*, o primeiro seguimento da arquitetura, API, agrupa as camadas que compõem às aplicações. Nesse contexto, cada camada exerce papel específico e bem definido no plano de codificação. Suas responsabilidades são agrupadas em 3 grupos:

- **Resources Layer / Controllers / Recursos:** Essa camada é responsável por separar a estrutura de acesso dos dados da camada de apresentação do sistema. Ela centraliza a complexidade requisições e retornos da API Rest. Esta camada adere ao item 3.2.1.1.9 do guia geral.

- **Business Layer / Services / Serviços:** Responsável por centralizar e validar as regras negociais utilizadas para atingir o objetivo da aplicação. Esta camada também tem como objetivo centralizar o acesso a serviços externos, sendo estes em sua maioria serviços web Rest e SOAP. Como exemplo de serviço externo consumido está o Cadastro de Pessoas, o Serviço de Documentos e o Serviço de Pessoas da Receita. Esta camada adere aos itens 3.2.1.1.5 e 3.2.1.2.3 do guia geral.
- **Repositories Layer / Camada de banco de dados:** Representa o SGBD ou local de persistência dos dados. Podendo ser um banco de dados relacional (Oracle, Postgresql) ou não relacional (MEMCACHED, ElasticSearch). Esta camada adere aos itens 3.2.1.1.4 e 3.2.1.2.1 do guia geral.

#### 4.1.1 Camada de recursos

O conjunto de rotinas e o de padrões estabelecidos e documentados da API devem ser implementados nessa camada. Ela é o ponto de partida das funcionalidades do microsserviço.

Deve ser utilizado um dos frameworks homologados pela CAPES, vide Anexo A – Tecnologias, para descrever a API e expor sua documentação utilizando o Swagger.

Para cada *namespace* da API, dever-se-á criar um *Resource*, que descreve e implementa o acesso à camada de negócios, processa e serializa dados de retorno de um objeto *HttpServletResponse* contendo o conteúdo do modelo serializado solicitado ou gera uma exceção como *Http404*.

#### 4.1.2 Camada de negócio

Todo acesso à camada de Banco de dados deve ser feito a partir da camada de negócio, nela também são verificadas e validadas as regras negociais, podendo emitir exceções se for o caso. Devem ser consumidos todos os tipos de serviços externos, independente se são serviços web usando REST ou SOAP. Essa camada tem como objetivo abstrair o consumo desses recursos, deixando visível para as outras camadas apenas os parâmetros de requisição e o retorno esperado. Por fim, controla as transações quando as operações envolverem manipulação de dados.

#### 4.1.3 Camada de acesso a dados

Implementação do modelo de dados no PHP definido em conjunto com a equipe de banco de dados e suas boas práticas. Os dados presentes nessa camada devem ser utilizados pela camada de acesso a dados.

Na CAPES, os sistemas são isolados, cada um tem um esquema de banco de dados diferente, um sistema não deve ter acesso a dados de outro sistema via banco de dados. A maneira que estes dados devem ser compartilhados estão descritos no item "Camada de negócio" deste documento.

Todo acesso ao banco de dados no esquema do sistema é feito através da camada de acesso a dados. As consultas DML de recuperação, inclusão, remoção e modificação de informações em bancos de dados devem ser feitas através de técnica de mapeamento objeto relacional que permite fazer uma relação dos objetos com os dados os quais representam (*Object Relational Mapper*).

As aplicações desenvolvidas na CAPES utilizam como mecanismo de persistência SGDBs relacionais. As aplicações PHP fazem uso do framework Doctrine para acessar os dados dos bancos.

#### 4.2 Frontend

Não se aplica.

#### 4.3 Backend

Sem prejuízo à definição do Guia Geral 3.1.3. A seguir, é definido como uma aplicação PHP será executada em ambiente CAPES.

No modelo tradicional de execução, para cada requisição ao servidor web, por exemplo Apache, é criado uma *thread* e, nela, uma instância do PHP. Esse modelo se mostra demasiadamente custoso para o processo.



Figura 2 Modelo de execução tradicional do PHP

Com o modelo da execução tradicional do PHP, Figura 2 Modelo de execução tradicional do PHP, o atendimento a grande volume de requisição tende a exaurir os recursos do servidor rapidamente ou provocar lentidão nesse processo.

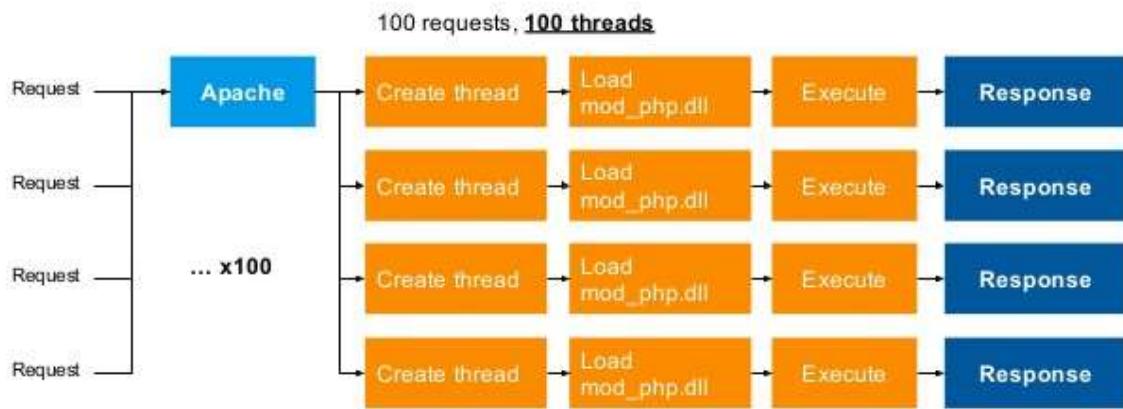


Figura 3 Alto número de threads

A Figura 3 Alto número de threads mostra o alto custo de recursos empregados para suprir a demanda de requisições conforme o número delas cresce.

Porém, desde sua versão 5.3.3, o PHP oferece suporte ao *FastCGI Process Manager* (FPM ou PHP-FPM). Neste modelo, o servidor web tem papel menos protagonista e atuar mais como um serviço de Proxy.

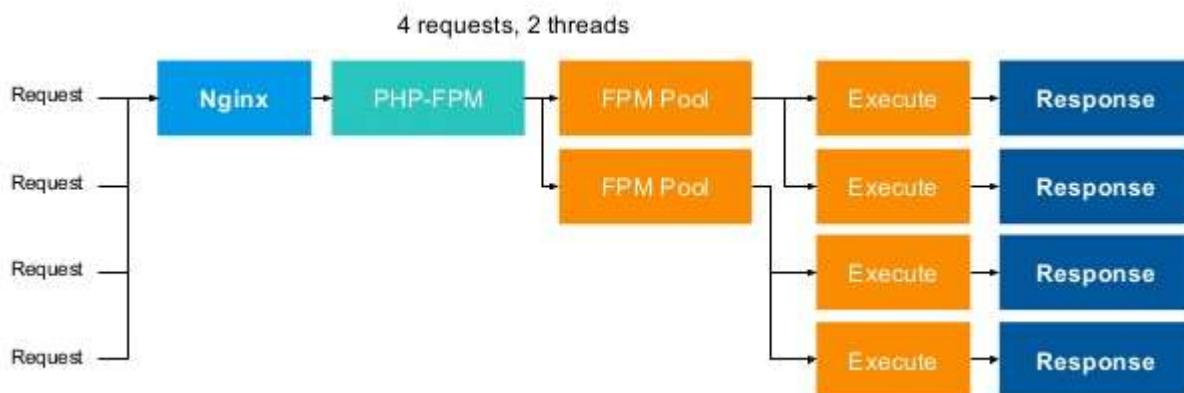


Figura 4 Modelo de execução PHP-FPM

Com o PHP-FPM, o número de *threads* do servidor web se descola do número de requisições e isso torna o processo mais célere e consome menos recursos do servidor. A Figura 4 Modelo de execução PHP-FPM evidencia que há uma instância de servidor web ligada a um serviço PHP-FPM, que, por sua vez, possui um *pool* de *threads* prontas para executar as requisições.

A CAPES adotará o modelo PHP-FPM para execução de aplicações codificadas em PHP.

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

#### 4.4 Padrões

Definições adotadas pela CAPES com base em suas necessidades para criação, manutenção e evolução de seus sistemas. A CAPES adere aos padrões especificados pela comunidade PHP: *PSR* (*PHP Standards Recommendations*). Assim, como regra, todo código PHP deverá atender aos requisitos de seu ou seus *PSRs*. A listagem completa e atualizada dos PSR está disponível em: [www.php-fig.org](http://www.php-fig.org).

Para o especificado acima, dever-se-á considerar apenas as PSR que tenha *status* 'ACCEPTED'.

ACCEPTED				
NUM	TITLE	EDITOR	COORDINATOR	SPONSOR
1	<a href="#">Basic Coding Standard</a>	Paul M. Jones	N/A	N/A
2	<a href="#">Coding Style Guide</a>	Paul M. Jones	N/A	N/A
3	<a href="#">Logger Interface</a>	Jordi Boggiano	N/A	N/A
4	<a href="#">Autoloading Standard</a>	Paul M. Jones	Phil Sturgeon	Larry Garfield
6	<a href="#">Caching Interface</a>	Larry Garfield	Paul Dragoonis	Robert Hafner
7	<a href="#">HTTP Message Interface</a>	Matthew Weier O'Phinney	Beau Simensen	Paul M. Jones
11	<a href="#">Container Interface</a>	Matthieu Napoli, David Negríer	Matthew Weier O'Phinney	Korvin Szanto
13	<a href="#">Hypermedia Links</a>	Larry Garfield	Matthew Weier O'Phinney	Marc Alexander
14	<a href="#">Event Dispatcher</a>	Larry Garfield	N/A	Cees-Jan Kiewiet
15	<a href="#">HTTP Handlers</a>	Woody Gilk	N/A	Matthew Weier O'Phinney
16	<a href="#">Simple Cache</a>	Paul Dragoonis	Jordi Boggiano	Fabien Potencier
17	<a href="#">HTTP Factories</a>	Woody Gilk	N/A	Matthew Weier O'Phinney
18	<a href="#">HTTP Client</a>	Tobias Nyholm	N/A	Sara Golemon

Figura 5 Quadro de PSRs com status ACCEPED

##### 4.4.1 Tecnologia

O Anexo A - Tecnologias elenca as tecnologias adotadas pela CAPES, de forma não exaustiva, para a pilha de desenvolvimento PHP. A inclusão ou remoção de itens nele, deverá ser definido pela CAPES.

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

#### 4.4.2 Componente

O Anexo B - Componentes elenca os componentes adotados pela CAPES para a pilha de desenvolvimento PHP. A inclusão ou remoção de itens nele, deverá ser definido pela CAPES.

#### 4.4.3 Nomenclatura

A CAPES adotará o padrão PSR-2 para o desenvolvimento em PHP.

### 5. Visão lógica

Vide correlação no item 4.1.

### 6. Visão de processos

Não se aplica.

### 7. Visão de implantação

Atualmente, os ambientes não produtivos e produtivos da Capes estão disponibilizados via Openshift. De modo que é possível baixar uma imagem de um contêiner Docker homologada pela equipe de infraestrutura e iniciar o desenvolvimento diretamente no container sem a necessidade de que seja provido um ambiente específico para desenvolvimento ou servidor pré-instalado.

Para a geração do pacote de *deploy* da aplicação, a Pipeline do GitLab se integra com o Jenkins. No GitLab, é feita a criação de uma TAG que represente o código usado na versão do sistema que será manipulado. A cada execução dessa tarefa, faz-se uma análise estática de código através do SonarQube. Por fim, o pacote é gerado e disponibilizado no contêiner que seguirá todos os passos de aceitação para ser disponibilizado em ambiente produtivo.

Na Wiki, existe uma página que detalha as configurações que devem ser feitas pelo desenvolvedor na sua máquina e no projeto. A página está disponível em [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy)

Cada projeto deverá ter um pacote chamado **devops** em seu repositório, que é mantido pela equipe de arquitetura e pelo líder técnico da equipe. Neste pacote deve conter os arquivos de CI e CD da aplicação.

### 8. Visão de implementação

PHP é uma linguagem de programação poderosa e de fácil aprendizado. Ela possui estruturas de dados de alto nível e adota uma abordagem simples e efetiva para a programação orientada a objetos. Sua sintaxe elegante e tipagem dinâmica, em adição à sua natureza interpretada, as tornam

ideal para *scripting* e para o desenvolvimento rápido de aplicações em diversas áreas e em várias plataformas.

A extensa biblioteca padrão do PHP está disponível na forma de código fonte aberto a partir do PECL (*PHP Extension Community Library*) ou por componentes prontos e disponibilizados por seu gerenciador de pacotes mais popular – Composer.

O interpretador de PHP é facilmente extensível incorporando novas funções e tipos de dados implementados em C ou C++ seguindo o modelo de extensão disponível no site oficial do próprio PHP.

Na comunidade PHP, existem uma grande quantidade de frameworks disponíveis para serem utilizadas, porém na CAPES, foram adotados os frameworks que constam no **Anexo – Tecnologias**.

### 8.1 Microsserviços

Tradicionalmente, softwares são construídos como uma estrutura fechada, com começo, meio e fim. Ou melhor, *backend* e *frontend*.

Essa filosofia acaba gerando peças muito grandes (Arquitetura Monolítica). Normalmente, focadas em resolver grandes necessidades e problemas organizacionais. Como exemplo, sendo necessário usar somente as funções que fazem gráficos do Excel para alguma outra aplicação, terá que abrir o Excel e carregar outras funcionalidades desnecessárias naquele momento.

A Arquitetura de Microsserviços, ao contrário das anteriores de Arquitetura Monolítica ou até SOA, destaca-se por explorar a ideia de granularidade, o que facilita a execução do próprio serviço e a adaptação às mudanças.

A ideia é dividir um determinado sistema em serviços acionáveis e modulares de modo que a união de pequenas partes realize um trabalho maior. Assim, os microsserviços permitem a integração entre vários serviços e a inserção de vários componentes no sistema.

### 8.2 Decompondo aplicações em serviços

A forma mais usual para escalar uma aplicação é executando várias cópias idênticas de aplicação (por meio de um平衡ador de cargas) realizando um processo chamado Decomposição.

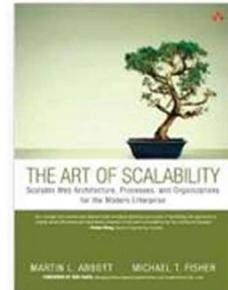
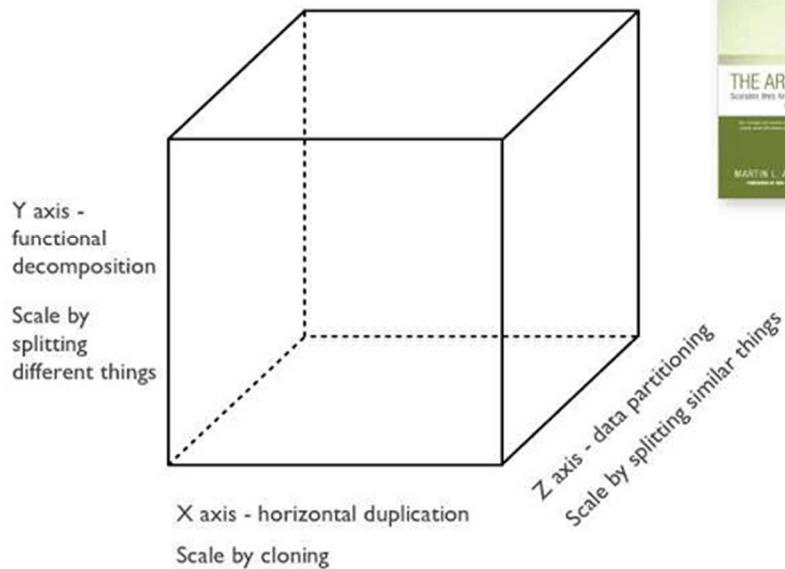
A finalidade da decomposição é resolver problemas comuns na arquitetura monolítica, implicando em que alguns serviços poderão ser muito pequenos enquanto outros serão significativamente maiores.

No conceito de três dimensões, é possível entender como os Microsserviços são úteis no particionamento e escalabilidade de uma aplicação monolítica. Temos três eixos de escalabilidade:

- X: referente à escalabilidade horizontal, para ampliar a capacidade e disponibilidade da aplicação (cada servidor executa uma cópia idêntica do código);
- Z: semelhante à do eixo X, mas requer a presença de um componente que se responsabilize pelo roteamento das requisições ao servidor adequado;
- Y: é a terceira dimensão da escalabilidade, a horizontal, denominada decomposição funcional e é responsável por dividir a aplicação em uma série de serviços. A cada serviço corresponde um conjunto de funções (gerenciamento de pedidos, gerenciamento de clientes e assim por diante).

Lembrando que, enquanto o eixo Z divide elementos semelhantes, o eixo Y divide elementos distintos.

### 3 dimensions to scaling



### 8.3 Implantar Microsserviços depende de um bom particionamento

A divisão, ou particionamento, do sistema deve ser bem elaborada, segue as abordagens que adotamos para implantar a Arquitetura de microsserviços:

- **Por verbos/casos de uso:** o caso *Checkout*, por exemplo, em que um serviço de conclusão de pedidos, ou *Checkout UI service*, implementa a interface com as pessoas que usam o sistema.
- **Por sinônimos/recursos:** como exemplo, considere que, para gerenciamento do catálogo de mercadorias, a empresa pode ter o *Catalog Service*. Nesse caso, o serviço se responsabiliza por todas as atividades que envolvem os recursos/entidades relacionados.

O importante é que cada serviço possua poucas responsabilidades, de acordo com o Princípio de Responsabilidade Única (SRP, *Single Responsibility Principle*), ou seja, um serviço exposto deve ter um único e claro papel dentro da Arquitetura.

Se for o caso em que um serviço possui mais de uma responsabilidade, deve-se aplicar algum particionamento, como citado acima.

As funcionalidades Unix constituem outro exemplo de modelagem de serviços, em que cada funcionalidade realiza somente uma operação definida (podendo, no entanto, ser combinada, por meio de *shell script*, com outras funcionalidades a fim de realizar atividades mais complexas). Ao longo dos anos, esse baixo acoplamento facilitou que diversas variações

### 9. Tamanho e desempenho

Caso a aplicação tenha processamento em lote ou execuções agendadas, essas operações devem, preferencialmente, ser realizadas em servidores separados dos servidores destinados a atender a requisições dos usuários. Devido aos custos de processamento destes recursos, a resposta do usuário pode ser onerada caso estejam no mesmo servidor.

As requisições devem ser atendidas o mais rápido possível. Processamentos de pedidos de usuários que gerem demora perceptível na *request* deve ter o seu processamento efetuado em *background* e o usuário deve ser notificado da conclusão da tarefa.

Preferir a separação do *front-end* e do *back-end*, visando facilitar a escalabilidade de recursos e o desacoplamento entre tecnologias com fins diferentes.

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

## 10. Requisitos de Qualidade

O Índice de Qualidade (IQ) será obtido a partir de indicadores de qualidade do código-fonte do software aferidos com o apoio da ferramenta SonarQube.. Alguns pontos que serão mais cuidadosamente analisados com base nos relatórios do sonar são: cobertura de testes, complexidade ciclomática, acoplamento e coesão.

O acesso ao SonarQube é livre para todos os desenvolvedores e deve ser incentivado dentro das equipes como meio de adequação e qualidade de código.

Além dos itens analisados pelo SonarQube, o código deverá ser submetido a testes de carga (estresse) pela ferramenta Apache JMeter. a qual demanda script JMeter capaz de produzir o referido teste.

Com a finalidade de garantir uma boa qualidade do código desenvolvido, os desenvolvedores devem usar o BDD (Design/Desenvolvimento guiado por comportamento), que em resumo é uma técnica voltada para o comportamento da aplicação e é comumente usada para testes.

Como forma de auxiliar no desenvolvimento dos códigos, podem ser utilizadas, bibliotecas que facilitem a atividade de teste, entre elas:

- **PHPUnit** - framework usado para escrever testes unitários de forma que permita a sua execução automática.
- **PHPUnit** – framework para executar testes de integração, auxiliando em testes que precisem de um banco ou recursos de container.
- **PHPUnit** – framework para executar validação de métricas CRAP (*Change Risk Analysis and Predictions*)
- **PHPCodeSniffer** – ferramenta para indicar violação às regras do padrão de estilo de codificação do código PSR-2.
- **PHPMD (PHP Mess Detector)** – ferramenta para indicar possíveis bugs ou uso indevido da linguagem.
- **PHPMD (PHP Magic Number Detector)** – ferramenta para detectar números mágicos no código.

- **PHPStan (PHP Static Analysis Tools)** – ferramenta para avaliar o código estaticamente revelando erros que são mostrados em tempo de compilação (execução).
- **PHPCPD (PHP Copy Past Detector)** – ferramenta para detectar bloco de código duplicado.
- **CHURN-PHP** – ferramenta para detectar classes que precisam ser refatoradas com base nas métricas de *cyclomatic complexity*.
- **Selenium** – ferramenta usada para mapear a navegação do usuário, usada para escrever os testes de aceitação.
- **Apache JMeter** – ferramenta para execução de carga estresse no código.

	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

## 11. Definições, acrônimos e abreviações

- *Backend* – Código executado no lado do servidor
- *Frontend* – Código executado no lado cliente
- Jenkins - <http://jenkins.capes.gov.br>
- Nexus - <http://nexus.capes.gov.br>
- Sonar - <http://sonar.capes.gov.br>
- GITLab- <http://git.capes.gov.br>
- Wiki - <http://wiki.capes.gov.br>
- Openshift - <http://openshift.capes.gov.br>
- Lumen - <https://lumen.laravel.com>
- Slim - <http://www.slimframework.com>
- Doctrine - <https://www.doctrine-project.org>
- Swagger - <https://swagger.io>
- Angular - <https://angular.io>
- React - <https://reactjs.org>
- Vuejs - <https://vuejs.org>

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

## 12. Referências

- [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy)
- [https://wiki.capes.gov.br/index.php/DTI:Visao\\_geral\\_tecnologias\\_integradas](https://wiki.capes.gov.br/index.php/DTI:Visao_geral_tecnologias_integradas)
- The Art of Computer Programming: D. E. Knuth, Addison-Wesley (volumes 1--3, 4A), 1998.
- Composer, disponível em: [getcomposer.org](http://getcomposer.org), acessado em: 11/03/2019
- PHP, disponível em: [php.net](http://php.net), acessado em 11/03/19
- PHP-FPM, disponível em: [http://php.net/manual/pt\\_BR/install.fpm.php](http://php.net/manual/pt_BR/install.fpm.php), acessado em: 22/03/2019
- PSR, disponível em: [www.php-fig.org/psr/](http://www.php-fig.org/psr/), acessado 20/03/2019
- PHPUnit, disponível em: [phpunit.de](http://phpunit.de) | [phpunit.readthedocs.io](http://phpunit.readthedocs.io), acessado em 11/03/19
- PHPCS, disponível em: [github.com/squizlabs/PHP\\_CodeSniffer/wiki](http://github.com/squizlabs/PHP_CodeSniffer/wiki), acessado em: 20/03/2019
- PHPMD, disponível em: [phpmd.org](http://phpmd.org), acessado em: 20/03/2019
- PHPMD, disponível em: [github.com/povils/phpmd](http://github.com/povils/phpmd), acessado em: 20/03/2019
- PHPStan, disponível em: [github.com/phpstan/phpstan](http://github.com/phpstan/phpstan), acessado em: 20/03/2019
- PHPCPD, disponível em: [github.com/sebastianbergmann/phpcpd](http://github.com/sebastianbergmann/phpcpd), acessado em: 20/03/2019
- CHRUN-PHP, disponível em: [github.com/bmitch/churn-php](http://github.com/bmitch/churn-php), acessado em: 20/03/2019
- Apache JMeter, disponível em: [jmeter.apache.org](http://jmeter.apache.org), acessado em: 20/03/2019
- Software Metrics for Code Coverage, disponível em: [phpunit.de/manual/6.5/en/code-coverage-analysis.html](http://phpunit.de/manual/6.5/en/code-coverage-analysis.html), acessado em: 20/03/2019
- Cyclomatic complexity, disponível em: [en.wikipedia.org/wiki/Cyclomatic\\_complexity](http://en.wikipedia.org/wiki/Cyclomatic_complexity), acessado em: 20/03/2019



- CRAP Metric Is a Thing And It Tells You About Risk in Your Code, disponível em: [blog.ndepend.com/crap-metric-thing-tells-risk-code/](http://blog.ndepend.com/crap-metric-thing-tells-risk-code/), acessado em: 20/03/2019
- How to Read and Improve the C.R.A.P Index of your code, disponível em: [opnsrce.github.io/how-to-read-and-improve-the-c-r-a-p-index-of-your-code](https://opnsrce.github.io/how-to-read-and-improve-the-c-r-a-p-index-of-your-code), acessado em: 20/03/2019
- How to Read / Improve C.R.A.P Index Calculated by PHP, disponível em: [stackoverflow.com/questions/4731774/how-to-read-improve-c-r-a-p-index-calculated-by-php](https://stackoverflow.com/questions/4731774/how-to-read-improve-c-r-a-p-index-calculated-by-php), acessado em: 20/03/2019

 <b>CAPES</b>	<b>Arquitetura de Referência – Guia PHP</b>	Versão: 1.1
		Data Emissão: 13/05/2019

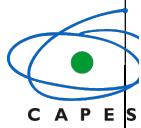
## Anexo A – Tecnologias

NOME	VERSÃO	DESCRIÇÃO
PHP	7.3+	Interpretador PHP
Composer	1.8+	Gerenciador de dependências
Lumen	5.8+	Micro framework baseado em Laravel
Slim	3+	Micro Framework PHP
Symfony	4.3+	Framework PHP
Doctrine (ORM)	2.6+	ORM PHP
Swagger-PHP	3+	Gerador de API Swagge <a href="http://zircote.com/swagger-php/">http://zircote.com/swagger-php/</a>
Codeception	2.1+	Framework para testes
PHPUnit	8+	Framework para testes
Monolog	1.24+	Componente de manipulação de Log <a href="https://github.com/Seldaek/monolog">https://github.com/Seldaek/monolog</a>
JWT	1.3+	Componente para manipulação de Token <a href="https://github.com/web-token/jwt-framework">https://github.com/web-token/jwt-framework</a>



**Anexo B – Componentes**

NOME	VERSÃO	DESCRIÇÃO
-	-	-



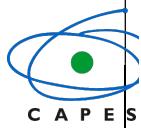
## ARQUITETURA DE REFERÊNCIA - GUIA PHYTON

**Versão: 1.1**

Data	Versão	Descrição	Responsável
11/03/2019	1.0	Criação do Guia Específico de Python	Harrysson Gilgamesh de Medeiros Nóbrega Thiago Adelino de Melo Valdecy Lourenço de Araújo Júnior
15/04/2019	1.0	Homologação do Guia Específico de Python	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Leonardo Morais Borges Théo Alves Monteiro Valdecy Lourenço de Araújo Júnior
13/05/2019	1.1	Alteração do quadro de métrica de qualidade no item 4 – Metas e restrições arquiteturais.	Alexandre Queiroz de Oliveira Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Théo Alves Monteiro Davi Souza Rafael Jesus Nirian Martins S. dos Santos

## Sumário

1	INTRODUÇÃO.....	5
2	PROpósito.....	5
3	ESCOPO.....	5
4	METAS E RESTRIÇÕES ARQUITETURAIS .....	5
4.1	CAMADA.....	7
4.1.1	CAMADA DE RECURSOS .....	7
4.1.2	CAMADA DE NEGÓCIOS .....	8
4.1.3	CAMADA DE ACESSO A DADOS .....	8
4.2	FRONTEND.....	8
4.3	BACKEND .....	9
4.4	PADRÕES .....	10
4.4.1	TECNOLOGIA .....	10
4.4.2	COMPONENTE.....	10
4.4.3	NOMENCLATURA .....	10
5	VISÃO LÓGICA.....	10
6	VISÃO DE IMPLEMENTAÇÃO .....	10
6.1	MICROSSERVIÇOS .....	11
6.2	DECOMPONDO APLICAÇÕES EM SERVIÇOS.....	12
6.3	IMPLANTAR MICROSSERVIÇOS DEPENDE DE UM BOM PARTICIONAMENTO .....	12
6.4	AS VANTAGENS DA ARQUITETURA DE MICROSSERVIÇOS .....	13
7	VISÃO DE PROCESSOS .....	14
8	VISÃO DE DADOS .....	14
9	VISÃO DE IMPLANTAÇÃO.....	14
10	TAMANHO E DESEMPENHO .....	14
11	REQUISITOS DE QUALIDADE .....	15
12	DEFINIÇÕES, ACRÔNIMOS E ABREVIAÇÕES.....	16
13	REFERÊNCIAS .....	17



<b>14</b>	<b>ANEXO A – TECNOLOGIAS</b>	<b>18</b>
<b>15</b>	<b>ANEXO B – COMPONENTES</b>	<b>18</b>

## 1 Introdução

Este documento é uma especialização do Guia Geral de Arquitetura e tem natureza subsidiária ao Guia Geral.

## 2 Propósito

Esse documento tem por propósito fornecer uma visão arquitetural abrangente para ser usada no desenvolvimento de sistemas. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da instituição. Assim, não tem a intenção de ser um documento definitivo de arquitetura para cada sistema individual. Ao invés disso, tenta traçar os objetivos de alto nível que levem a uma boa arquitetura de software. Cada sistema em particular deve ser analisado junto com a equipe de desenvolvimento e um dos arquitetos de sistemas da CAPES. Se os tópicos aqui apresentados forem seguidos, as chances de sucesso nos projetos tendem ser maiores.

## 3 Escopo

Fornecer uma visão arquitetural abrangente para ser usada no desenvolvimento de sistemas. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da instituição.

## 4 Metas e Restrições Arquiteturais

Na década de 70 os padrões de projetos era um conceito da arquitetura, descrito no livro *A Pattern Language* escrito pelo Christopher Alexander. Nos anos 80 o Kent Beck (o criador do *Extreme Programming*, *Test Driven Development* e um dos signatários originais do *Agile Manifesto*) e Ward Cunningham começaram a aplicar esse conceito na área da computação.

A arquitetura em camadas é um estilo arquitetural que propõe dividir um sistema em várias camadas de acordo com a responsabilidade de cada parte do software. Este estilo propõe que:

- As camadas tenham um propósito bem definido;
- Cada camada conheça apenas camadas abaixo dela;
- As camadas possam ser reaproveitáveis.

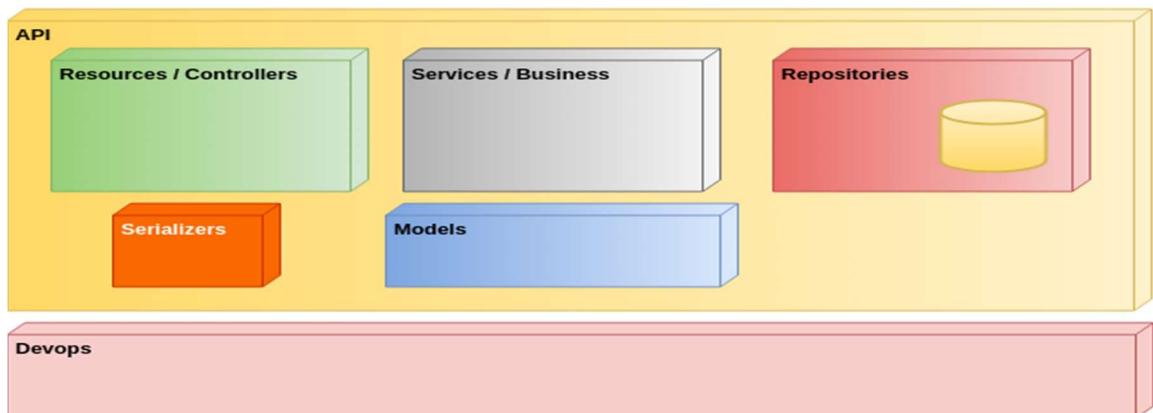


Figura 1 Arquitetura de aplicação Python

A separação de dados e comportamento leva a programação procedural e como consequência a perda de vantagens básicas da orientação a objetos como reutilização de código, encapsulamento, herança e polimorfismo. Esse modelo de programação é conhecido como modelo anêmico e devemos evitá-lo.

É preciso destacar também a importância da cobertura de testes, que traz várias vantagens para o projeto, como a maior qualidade dos sistemas entregues, facilidade de manutenção, diminuição de tempo para encontrar e corrigir problemas entre vários outros benefícios. Todos os novos sistemas devem ter suas principais funcionalidades testadas através de testes unitários ou de integração.

Ressalta-se que cada camada deve ser desenvolvida mantendo independente umas das outras, princípio da responsabilidade única. A camada de negócio não deverá ter seu comportamento diferido se passado os mesmos valores como parâmetros, não importando quantas vezes seja instanciada. Ainda, não deverá sofrer interferência de nenhum tipo acarretada pela tecnologia que seus consumidores adotem.

Os sistemas desenvolvidos deverão utilizar HTTPS nas comunicações com o usuário externo com a finalidade de garantir maior segurança no trâmite das informações.

Para promover a integração entre os sistemas e facilitar a usabilidade os sistemas deverão utilizar o segurança SSO como mecanismo de login único para as aplicações.

Os indicadores de qualidade que cada sistema deverá apresentar ao ser submetido à avaliação por meio de ferramenta Sonarqube, compreende em cada indicador uma meta a ser alcançada. No que tange ao código-fonte, o quadro a seguir relaciona os indicadores e sua respectiva meta.

Grupo	Indicador	Tipo Meta	Meta
PROJETO	Problemas confirmados	Unidades	= 0
	*Complexidade	Média total	<= 10
	Métodos	Média total	<= 3
	Índice de manutenibilidade	Nota	A
	Índice de confiabilidade	Nota	A
	Índice de segurança	Nota	A
	Taxa de dívida técnica	%	<= 2,5%
	Classes	Média total	<= 10
	Arquivos	Média total	<= 10
VIOLAÇÕES	Linhas duplicadas (%)	%	<= 4%
	Problemas impeditivos	Unidades	= 0
TESTE	Problemas críticos	Unidades	= 0
	Testes unitários ignorados	Unidades	= 0
	Sucesso em testes unitários (%)	%	= 100%
	Cobertura (camada de negócio)	%	>= 70%

\* Indicador não previsto no Sonarqube versão 6.7.1 LTS

Em alinhamento com o Guia Geral, a arquitetura de aplicações Python é projetada para o desenvolvimento em microsserviços e é agrupada em dois segmentos: API e DevOps.

- API é o agrupamento dos pontos de consumo da aplicação que podem ser consumidos por outros sistemas ou interfaces de usuários especializadas. Constitui a fronteira entre o *backend* e *frontend*.
- DevOps é o conjunto de *scripts* e artefatos necessários à automação do processo *build/deploy* da aplicação usando, no contexto da CAPES, o OpenShift.

## 4.1 Camada

Conforme mostrado na *Figura 1 Arquitetura de aplicação Python*, o primeiro seguimento da arquitetura, API, agrupa as camadas que compõe às aplicações nessa linguagem. Nesse contexto, cada camada exerce papel específico e bem definido no plano de codificação. Suas responsabilidades são agrupadas em 3 grupos:

- **Resources Layer / Controllers / Recursos:** Essa camada é responsável por separar a estrutura de acesso dos dados, da camada de apresentação do sistema. Ela centraliza a complexidade requisições e retornos da API Rest. Esta camada adere ao item 3.2.1.1.9 do guia geral.
- **Business Layer / Services / Serviços:** Responsável por centralizar e validar as regras negociais utilizadas para atingir o objetivo da aplicação. Esta camada também tem como objetivo centralizar o acesso a serviços externos, sendo estes em sua maioria serviços web Rest e SOAP. Como exemplo de serviço externo consumido, está o Cadastro de Pessoas, o Serviço de Documentos e o Serviço de Pessoas da Receita. Esta camada adere ao item 3.2.1.2.3 do guia geral.
- **Repositories Layer / Camada de banco de dados:** Representa o SGBD, ou local de persistência dos dados. Podendo ser um banco de dados relacional (Oracle, Postgresql) ou não relacional (CouchDB, MongoDB). Esta camada adere ao item 3.2.1.1.4 do guia geral.

### 4.1.1 Camada de recursos

Nessa camada devem ser implementadas o conjunto de rotinas e padrões estabelecidos e documentados da API, ela é o ponto de partida das funcionalidades do microsserviço.

Deve ser utilizado o Flask-RESTPlus, ou equivalente no Django de modo stateless, para descrever a API e expor sua documentação corretamente utilizando o Swagger.

Para cada namespace no *framework* Flask deve-se criar um Resource, que descreve e implementa o acesso à camada de negócios, processa os dados, serializa e retorna um objeto *HttpResponse* contendo o conteúdo do modelo serializado solicitado ou gera uma exceção como *Http404*.

Na inicialização do app é necessário adicionar esses resources criados.

#### 4.1.2 Camada de negócios

Todo acesso à camada de Banco de dados deve ser feito a partir da camada de negócio, é nela também que são verificadas e validadas as regras negociais, podendo emitir exceções se for o caso. Devem ser consumidos todos os tipos de serviços externos, independente se são serviços web usando Rest, ou SOAP. Essa camada tem como objetivo abstrair o consumo desses recursos, deixando visível para as outras camadas apenas os parâmetros de requisição e o retorno esperado.

Para as operações que envolvem manipulação de dados é nesta camada que devem ser controladas as transações.

#### 4.1.3 Camada de acesso a dados

Implementação do modelo de dados no Python definido em conjunto com a equipe de banco de dados e suas boas práticas. Os dados presentes nessa camada devem ser utilizados pela camada de acesso a dados.

Na CAPES os sistemas são totalmente isolados, cada um tem um esquema de banco de dados diferente, um sistema não deve ter acesso à dados de outro sistema via banco de dados, a maneira que estes dados devem ser compartilhados estão descritos no item "Camada de negócio" deste documento.

Todo acesso ao banco de dados no esquema do sistema é feito através da camada de acesso a dados. As consultas DML de recuperação, inclusão, remoção e modificação de informações em bancos de dados devem feitas através de uma técnica de mapeamento objeto relacional que permite fazer uma relação dos objetos com os dados que os mesmos representam (Object Relational Mapper) utilizando a ferramenta SQLAlchemy.

SQLAlchemy é mais famoso por seu mapeador objeto-relacional (ORM), um componente opcional que fornece o padrão de mapeamento de dados, onde as classes podem ser mapeadas para o banco de dados de várias maneiras - permitindo que o modelo de objeto e o esquema de banco de dados sejam desenvolvidos completamente limpo desde o início.

O SQLAlchemy considera o banco de dados como um mecanismo de álgebra relacional, não apenas uma coleção de tabelas. As linhas podem ser selecionadas não apenas de tabelas, mas também de junções e outras instruções selecionadas; qualquer uma dessas unidades pode ser composta em uma estrutura maior. A linguagem de expressão do SQLAlchemy baseia-se neste conceito a partir do seu núcleo.

A abordagem geral do SQLAlchemy para esses problemas é totalmente diferente da maioria das outras ferramentas SQL / ORM, baseadas em uma abordagem chamada de complementaridade; em vez de esconder os detalhes relacionais de objetos e SQL por trás de uma parede de automação, todos os processos são totalmente expostos em uma série de ferramentas transparentes e compostos. A biblioteca assume o trabalho de automatizar tarefas redundantes enquanto o desenvolvedor permanece no controle de como o banco de dados é organizado e como o SQL é construído.

### 4.2 Frontend

Não se aplica.

#### 4.3 Backend

Sem prejuízo à definição do Guia Geral 3.1.3. A seguir, é definido como uma aplicação Python será executada em ambiente CAPES.

Com o advento da automatização e devida inteligência na manutenção das aplicações, hoje, é esperado que a aplicação possa atender a picos de demandas com inicialização automática de novos processos, sem afetar seu comportamento.

A boa prática indica que processos de aplicações são *stateless* (não armazenam estado) e *share-nothing*. Quaisquer dados que precisem persistir devem ser armazenados em serviço de apoio *stateful* (armazena o estado), normalmente é usado uma base de dados. O objetivo final dessa prática não faz distinção se a aplicação é executada na máquina do desenvolvedor ou em produção, pois, nesse caso, o que muda é a quantidade de processos iniciados para atender as respectivas demandas.

É importante salientar: ao seguir a prática, uma aplicação não assume que, qualquer item armazenado em cache de memória ou no disco, estará disponível em futura solicitação ou *job* – com muitos processos de cada tipo rodando, são altas as chances de futura solicitação ser servida por processo diferente, até mesmo em servidor diferente. Mesmo quando, rodando em apenas um processo, um *restart* (desencadeado pelo *deploy* de um código, mudança de configuração, ou o ambiente de execução realocando o processo para localização física diferente) geralmente vai acabar com o estado local (memória e sistema de arquivos, por exemplo).

Durante o processo de desenvolvimento de uma aplicação é difícil imaginar o volume de requisição que ela terá no momento que for colocada em produção. Por outro lado, um serviço que suporte grandes volumes de uso é esperado nas soluções modernas. Nada é mais frustante que solicitar acesso a uma aplicação e ela não estar disponível. Sugere falta de cuidado e profissionalismo, na maioria dos casos. Quando a aplicação é colocada em produção, normalmente é dimensionada para determinada carga esperada, porém é importante que o serviço esteja pronto para escalar. A solução deve ser capaz de iniciar novos processos da mesma aplicação, caso necessário, sem afetar o produto.

Quando falamos de aplicações web, espera-se que mais de um processo atenda a todo tráfego requisitado para o serviço. Porém, tão importante quanto a habilidade de iniciar novos processos, a capacidade de um processo defeituoso terminar na mesma velocidade que iniciou, pois um processo que demora para finalizar pode comprometer toda solução, uma vez que ela pode ainda estar atendendo requisições de forma defeituosa.

Em resumo, podemos dizer que aplicações web deveriam ser capazes de remover rapidamente processos defeituosos. Com objetivo de evitar que o serviço prestado seja dependente das instâncias que o servem, a boa prática indica que as aplicações devem ser descartáveis, ou seja, desligar uma de suas instâncias não deve afetar a solução como um todo.

Outro detalhe importante é viabilizar que o código desligue “graciosamente” e reinicie sem erros. Assim, ao escutar um SIGTERM, o código deve terminar qualquer requisição em andamento e então desligar

o processo sem problemas e de forma rápida, permitindo, também, que seja rapidamente atendido por outro processo. Entendemos como desligamento “gracioso” uma aplicação capaz de auto finalizar sem danos à solução; ao receber sinal para desligar, imediatamente recusa novas requisições e apenas finaliza as tarefas pendentes em execução naquele momento. Implícito nesse modelo: as requisições HTTP são curtas (não mais que poucos segundos) e, nos casos de conexões longas, o cliente pode se reconectar automaticamente caso a conexão seja perdida.

A CAPES adotará as seguintes opções de *Deployment* para aplicações Python:

- **Standalone WSGI Containers**

Existem servidores populares escritos em Python que contêm aplicativos WSGI e servem HTTP. Esses servidores *stand alone* quando são executados e você pode fazer *proxy* para eles a partir do seu servidor web

- **Gunicorn**

Gunicorn um servidor WSGI HTTP para UNIX. É um modelo de trabalho portado a partir do projeto Unicorn do Ruby. Suporta eventlet e greenlet . Executar um aplicativo Flask neste servidor é bastante simples.

- **uWSGI**

O uWSGI é um servidor de aplicativos rápido escrito em C. É muito configurável, o que o torna mais complicado de configurar que o gunicorn.

## 4.4 Padrões

Definições adotadas pela CAPES com base em suas necessidades para criação, manutenção e evolução de seus sistemas.

### 4.4.1 Tecnologia

O Anexo A - Tecnologias elenca as tecnologias adotadas pela CAPES de forma não exaustiva para a pilha de desenvolvimento Python. A inclusão ou remoção de itens nele, deverá ser definido pela CAPES.

### 4.4.2 Componente

O Anexo B Componentes elenca os componentes adotados pela CAPES para a pilha de desenvolvimento Python. A inclusão ou remoção de itens nele, deverá ser definido pela CAPES.

### 4.4.3 Nomenclatura

A nomenclatura detalha o padrão PEP 8 adotado pela CAPES para o desenvolvimento em Python.

## 5 Visão lógica

Vide correlação no item 4.1.

## 6 Visão de implementação

Python é uma linguagem de programação poderosa e de fácil aprendizado. Ela possui estruturas de dados de alto nível eficientes, bem como adota uma abordagem simples e efetiva para

a programação orientada a objetos. Sua sintaxe elegante e tipagem dinâmica, em adição à sua natureza interpretada, tornam Python ideal para scripting e para o desenvolvimento rápido de aplicações em diversas áreas e na maioria das plataformas.

O interpretador de Python e sua extensa biblioteca padrão estão disponíveis na forma de código fonte ou binário para a maioria das plataformas a partir do site, <http://www.python.org/>, e deve ser distribuído livremente. No mesmo site estão disponíveis distribuições e referências para diversos módulos, programas, ferramentas e documentação adicional contribuídos por terceiros.

O interpretador de Python é facilmente extensível incorporando novas funções e tipos de dados implementados em C ou C++ (ou qualquer outra linguagem acessível a partir de C). Python também se adequa como linguagem de extensão para customizar aplicações.

Na comunidade Python existem uma grande quantidade de *frameworks* disponíveis para serem utilizadas, porém na CAPES foram adotados os frameworks Flask, utilizado para microsserviços, e Django, utilizado para portais.

O Django é um *framework full stack* de código aberto para desenvolvimento rápido para *web*, escrito em Python, que utiliza o padrão model-template-view. Foi criado originalmente como sistema para gerenciar um site jornalístico na cidade de Lawrence, no Kansas. Tornou-se um projeto de código aberto e foi publicado sob a licença BSD em 2005.

O Flask é um *microframework* para Python baseado em Werkzeug, Jinja 2 e é licenciado pela BSD, ideal para APIs, pois é totalmente desacoplado e facilmente plugado com outros *microframeworks*.

## 6.1 Microsserviços

Tradicionalmente, softwares é construído como uma estrutura fechada, com começo, meio e fim. Ou melhor, *back-end* e *front-end*.

Essa filosofia acaba gerando peças muito grandes (Arquitetura Monolítica), normalmente focadas em resolver grandes necessidades e problemas organizacionais. Se você quiser usar somente as funções que fazem gráficos do Excel para alguma outra aplicação, terá que abrir o Excel e carregar outras funcionalidades desnecessárias naquele momento.

A Arquitetura de Microsserviços, ao contrário das anteriores de Arquitetura Monolítica ou até SOA, destaca-se por explorar a ideia de granularidade, o que facilita a execução do próprio serviço e a adaptação às mudanças.

A ideia é dividir um determinado sistema em serviços açãoáveis e modulares, de modo que a união de pequenas partes realize um trabalho maior. Assim, os microsserviços permitem a integração entre vários serviços e a inserção de vários componentes no sistema.

## 6.2 Decompondo aplicações em serviços

A forma mais usual para escalar uma aplicação é executando várias cópias idênticas de aplicação (por meio de um balanceador de cargas), realizando um processo chamado decomposição.

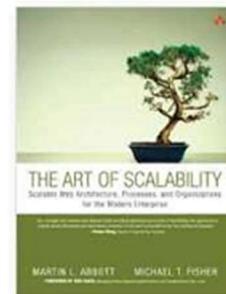
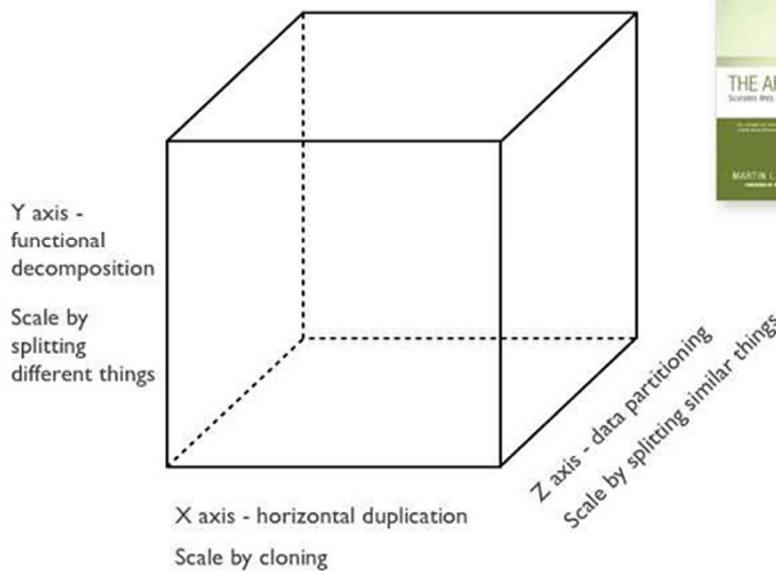
A finalidade da decomposição é resolver problemas comuns na arquitetura monolítica, implicando em que alguns serviços poderão ser muito pequenos enquanto outros serão significativamente maiores.

No conceito de três dimensões, é possível entender como os Microsserviços são úteis no particionamento e escalabilidade de uma aplicação monolítica. Temos três eixos de escalabilidade:

- X: referente à escalabilidade horizontal, para ampliar a capacidade e disponibilidade da aplicação (cada servidor executa uma cópia idêntica do código);
- Z: semelhante à do eixo X, mas requer a presença de um componente que se responsabilize pelo roteamento das requisições ao servidor adequado;
- Y: é a terceira dimensão da escalabilidade, a horizontal, denominada decomposição funcional e é responsável por dividir a aplicação em uma série de serviços. A cada serviço corresponde um conjunto de funções (gerenciamento de pedidos, gerenciamento de clientes e assim por diante).

Lembrando que, enquanto o eixo Z divide elementos semelhantes, o eixo Y divide elementos distintos.

## 3 dimensions to scaling



## 6.3 Implantar Microsserviços depende de um bom particionamento

A divisão, ou particionamento, do sistema deve ser bem elaborada, segue as abordagens que adotamos para implantar a Arquitetura de microsserviços:

- **Por verbos/casos de uso:** o caso *checkout*, por exemplo, em que um serviço de conclusão de pedidos, ou *Checkout UI service*, implementa a interface com as pessoas que usam o sistema.
- **Por sinônimos/recursos:** como exemplo, considere que, para gerenciamento do catálogo de mercadorias, a empresa pode ter o *Catalog Service*. Nesse caso, o serviço se responsabiliza por todas as atividades que envolvem os recursos/entidades relacionados.

O importante é que cada serviço possua poucas responsabilidades, de acordo com o Princípio de Responsabilidade Única (SRP, Single Responsibility Principle). Ou seja, um serviço exposto deve ter um único e claro papel dentro da Arquitetura.

Se for o caso em que um serviço possui mais de uma responsabilidade, deve-se aplicar algum particionamento, como citado acima.

As funcionalidades Unix constituem outro exemplo de modelagem de serviços, em que cada funcionalidade realiza somente uma operação definida (podendo, no entanto, ser combinada, por meio de *shell script*, com outras funcionalidades a fim de realizar atividades mais complexas). Ao longo dos anos, esse baixo acoplamento facilitou que diversas variações do sistema operacional fossem lançadas, como Ubuntu, Fedora, Solaris e tantas outras.

#### 6.4 As vantagens da Arquitetura de microsserviços

- Os desenvolvedores usufruem de liberdade maior para o desenvolvimento de serviços de modo independente;
- Implantação automática através de ferramentas de integração contínua e código aberto, como Hudson, Jenkins e outras;
- O contêiner web tem inicialização mais rápida;
- Possibilidade de utilizar códigos escritos em linguagens diferentes para diferentes serviços, usando uma “língua franca” para comunicação entre eles (como Json ou XML);
- Oportunidade para os desenvolvedores usarem as tecnologias mais atuais;
- Arquitetura de fácil compreensão e bastante adaptável às mudanças, o que favorece o aprendizado dos profissionais novatos, contribuindo para maior produtividade da equipe;
- Fácil ampliação e integração dos microsserviços com serviços terceirizados, através de APIs, por exemplo;
- Código organizado em função de capacidades de negócio, dando mais visão das ofertas e necessidades dos clientes;
- Mudanças necessárias poderão ser aplicadas somente sobre o serviço específico, sem necessidade de modificar todo o aplicativo. Atualizações de funcionalidades também passam a ser menos complexas;
- Gerenciamento otimizado das falhas (por exemplo, caso um serviço venha a falhar, os outros continuarão trabalhando).

## 7 Visão de processos

Não se aplica.

## 8 Visão de dados

As aplicações desenvolvidas na CAPES utilizam como mecanismo de persistência os SGBs relacionais. As aplicações em Python, fazem uso do *framework* SQLAlchemy para acessar os dados dos bancos. São necessários alguns cuidados a fim de garantir que a aplicação seja estável e não venha a causar problemas no acesso aos dados:

- Preferir o criteria ou Expression language do SQLAlchemy, ao SQL puro. Evitando dependência de banco.
- Usar corretamente o Lazy Loading, para trazer a maior quantidade de dados possíveis que realmente serão usados, com a finalidade de evitar 1+n consultas ao banco.
- Sempre que possível usar consultas paginadas para melhorar o desempenho.

## 9 Visão de implantação

Atualmente, os ambientes não produtivos e produtivos da Capes estão disponibilizados via Openshift, de modo que é possível baixar uma imagem de um contêiner Docker, homologada pela equipe de infraestrutura e iniciar o desenvolvimento da mesma, sem a necessidade que seja provido um ambiente específico para desenvolvimento ou servidor pré-instalado.

Deste modo, o desenvolvedor não necessita configurar ambiente, instalar servidor, ou configura-lo. Tudo está disponível direto na imagem Docker nos repositórios da Capes

Para a geração do pacote de *deploy* da aplicação, a *Pipeline* do GitLab se integra com o Jenkins. No próprio GitLab é feita a criação de uma TAG que represente o código usado nessa versão do sistema. A cada execução dessa tarefa é feita uma análise estática de código através do Sonar. E finalmente, o pacote gerado e disponibilizado no contêiner que seguirá todos os passos de aceitação para ser disponibilizado em ambiente produtivo.

Na *wiki*, existe uma página que detalha as configurações que devem ser feitas pelo desenvolvedor na sua máquina e no projeto. A página está disponível em [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy).

Cada projeto deverá ter um pacote chamado *devops* em seu repositório, mantido pela equipe de arquitetura e pelo líder técnico da equipe. Neste pacote deve conter os arquivos de CI e CD da aplicação.

## 10 Tamanho e Desempenho

Caso a aplicação tenha processamento em lote ou execuções agendadas, essas operações devem, preferencialmente, ser realizadas em servidores separados dos servidores destinados a atender a requisições dos usuários. Devido aos custos de processamento destes recursos a resposta do usuário pode ser onerada caso estejam no mesmo servidor.

As requisições devem ser atendidas o mais rápido possível. Processamentos de pedidos de usuários que gerem demora perceptível na *request* deve ter o seu processamento efetuado em *background* e o usuário deve ser notificado da conclusão da tarefa.

Preferir a separação do *frontend* e do *backend*, visando facilitar a escalabilidade de recursos e o desacoplamento entre tecnologias com fins diferentes.

## 11 Requisitos de Qualidade

Todos os sistemas passarão por análise estática de código através do Sonar. Alguns pontos que serão mais cuidadosamente analisados com base nos relatórios do sonar são: cobertura de testes, complexidade ciclomática, acoplamento e coesão.

O acesso ao Sonar é livre para todos os desenvolvedores e deve ser incentivado dentro das equipes como meio de adequação e qualidade de código.

Com a finalidade de garantir uma boa qualidade do código desenvolvido, os desenvolvedores devem usar o BDD (Design/Desenvolvimento guiado por comportamento), que em resumo é uma técnica voltada para o comportamento da aplicação e é comumente usada para testes.

Como forma de auxiliar no desenvolvimento dos códigos, podem ser utilizadas, bibliotecas que facilitem a atividade de teste, entre elas:

- **unittest** - pacote existente no python usado para escrever os testes de forma que permita a sua execução automática.
- **Selenium** - ferramenta usada para mapear a navegação do usuário, usada para escrever os testes de aceitação.
- **Doctest** - Os testes usando doctests testam a documentação de algo, buscando pedaços de códigos executáveis nos comentários.
- **Pytest** - ferramenta usada para executar os testes de integração, auxiliando em testes que precisem de um banco ou recursos de container.

## 12 Definições, Acrônimos e Abreviações

- Jenkins - <http://jenkins.capes.gov.br>
- Nexus - <http://nexus.capes.gov.br>
- Sonar - <http://sonar.capes.gov.br>
- GITLab- <http://git.capes.gov.br>
- Wiki - <http://wiki.capes.gov.br>
- Openshift - <http://openshift.capes.gov.br>
- Flask - <http://flask.pocoo.org/>
- SQLAlchemy - <https://www.sqlalchemy.org/>
- Flask-restplus - <https://flask-restplus.readthedocs.io/>
- Swagger - <https://swagger.io/>
- Angular - <https://angular.io/>
- React - <https://reactjs.org/>
- Vuejs - <https://vuejs.org/>
- Django - <https://www.djangoproject.com/>

## 13 Referências

- [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy)
- [https://wiki.capes.gov.br/index.php/DTI:Visao\\_geral\\_tecnologias\\_integradas](https://wiki.capes.gov.br/index.php/DTI:Visao_geral_tecnologias_integradas)
- Introdução à Programação em Mathematica (3a edição): J. Carmo, A. Sernadas, C. Sernadas, F. M. Dionísio, C. Caleiro, IST Press, 2014.
- Think Python: How to think like a computer scientist: A. Downey, Green Tea Press, 2012.
- Introduction to Computation and Programming Using Python (revised and expanded edition): J. V. Guttag, MIT Press, 2013.
- The Art of Computer Programming: D. E. Knuth, Addison-Wesley (volumes 1–3, 4A), 1998.
- Learning Python (fifth edition): M. Lutz, O'Reilly Media, 2013.
- Programação em Python: Introdução à programação utilizando múltiplos paradigmas: J. P. Martins, IST Press, 2015.
- Introdução à Programação em MatLab: J. Ramos, A. Sernadas e P. Mateus, DMIST, 2005.
- Learning IPython for Interactive Computing and Data Visualization: C. Rossant, Packt Publishing, 2013.
- Programação em Mathematica: A. Sernadas, C. Sernadas e J. Ramos, DMIST, 2003.
- Docker para desenvolvedores: Rafael Gomes, Leanpub, 2017

**14 Anexo A – Tecnologias**

Tecnologia	Versão
<b>python</b>	<b>&gt;=3.6.7</b>
<b>flask</b>	<b>&gt;=1.0.2</b>
<b>flask-restplus</b>	<b>&gt;=0.12.1</b>
<b>flask_restful</b>	<b>&gt;=0.3.6</b>
<b>flask_sqlalchemy</b>	<b>&gt;=2.3.1</b>
<b>flask_marshmallow</b>	<b>&gt;=0.8.0</b>
<b>marshmallow</b>	<b>&gt;=2.14.0</b>
<b>marshmallow-sqlalchemy</b>	<b>&gt;=0.13.2</b>
<b>cx_Oracle</b>	<b>&gt;=7.1</b>

**15 Anexo B – Componentes**

NOME	VERSÃO	DESCRIÇÃO
-	-	-



**ARQUITETURA DE REFERÊNCIA - GUIA FRONTEND**

**Versão: 1.1**



## Arquitetura de Referência – Guia Frontend

Versão: 1.1

Data Emissão:

03/06/2019

Data	Versão	Descrição	Responsável
06/05/2019	1.0	Criação do Guia de Frontend da Arquitetura de Referência	José Augusto de Jesus
15/05/2019	1.0	Homologação do Guia Geral	Carlos Alberto Rodrigues Santana José Arthur Souza de Macedo José Augusto de Jesus Leonardo Morais Borges Théo Alves Monteiro Valdecy Lourenço de Araújo Júnior
03/06/2019	1.1	Alteração do item 5.3 e Anexo B, incluindo os principais componentes CAPES.	Carlos Alberto Rodrigues Santana

---

## SUMÁRIO

1.	INTRODUÇÃO.....	4
2.	PROPÓSITO .....	4
3.	ESCOPO.....	4
4.	REPRESENTAÇÃO ARQUITETURAL.....	4
4.1	<i>SERVICE ORIENTED FRONT END ARCHITECTURE – SOFEA</i> .....	8
5.	METAS E RESTRIÇÕES ARQUITETURAIS .....	10
5.1	CAMADA .....	11
5.1.1	MÓDULO ( <i>MODULE</i> ).....	11
5.1.2	COMPONENTE ( <i>COMPONENT</i> ).....	12
5.1.2.1	HTML ( <i>TEMPLATE</i> ).....	12
5.1.2.2	SERVICE .....	12
5.1.2.3	FUNÇÕES .....	12
5.2	<i>FRONT-END</i> .....	13
5.3	COMPONENTE.....	13
6.	REQUISITOS DE QUALIDADE.....	14
7.	DEFINIÇÕES, ACRÔNIMOS E ABREVIACÕES .....	14
8.	REFERÊNCIAS .....	15
9.	ANEXO A – TECNOLOGIAS .....	16
10.	ANEXO B – COMPONENTES .....	16

## 1. Introdução

Este documento é uma especialização do **Guia Geral de Arquitetura** e tem natureza subsidiária ao Guia Geral.

## 2. Propósito

Apresentar arquitetura para o desenvolvimento e sustentação de *Front-end Web (Frontend)*. Detalhar os aspectos técnicos relativos à adoção de tecnologias, ao desenvolvimento e à implantação de componentes. Capturar e formalizar as principais decisões tomadas com relação à arquitetura de *Frontend*. Assim, não tem a intenção de ser um documento definitivo de arquitetura para cada sistema individual. Ao invés disso, tenta traçar os objetivos de alto nível que levem a uma boa arquitetura de software. Cada sistema em particular deve ser analisado junto com a equipe de desenvolvimento e um dos arquitetos de sistemas da CAPES. Se os tópicos aqui apresentados forem seguidos, as chances de sucesso nos projetos tendem ser maiores.

## 3. Escopo

Fornecer uma visão arquitetural usada no desenvolvimento de *Front-end*. Capturar e transmitir as decisões arquiteturais significativas que foram tomadas em relação ao desenvolvimento dos sistemas da CAPES.

Alinhar o desenvolvimento de *Front-end* a uma arquitetura baseada em componentes reutilizáveis, testáveis e com codificação dentro de padrões e guias de codificação de mercado.

Definir os parâmetros necessários a extensão deste documento.

## 4. Representação Arquitetural

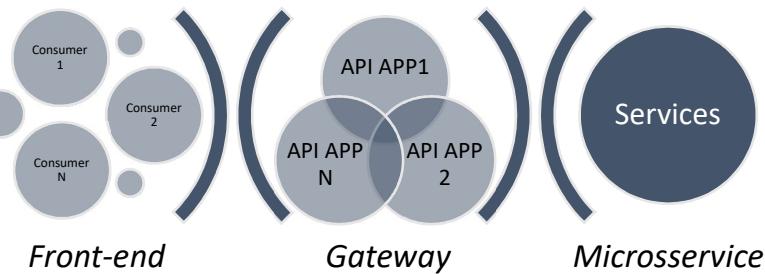


Figura 1 Fronteiras de responsabilidade das camadas

A *Figura 1* generaliza o padrão de projeto MVC, o qual, em sua essência, divide o desenvolvimento de software em três camadas: visão do cliente, controle de requisição e processamento de regra de negócio.

Objetivamente, este documento trata apenas da camada de visão do cliente (*View/User Interface*) e, devido à quantidade de *Framework* específico para ela, limita-se em definir premissas e restrições que devem ser atendidas na construção dessa camada.

O *Front-end* deverá ser funcionalmente independente do *Back-end*, mas dependente semanticamente.

Manter o isolamento de responsabilidade entre *Front-end* e *Back-end*. O grau de conformidade empregado entre *Front-end* e *Back-end* será definido pela API disponibilizada pelo serviço consumido.

A camada de *Front-end*, em direção ao *Back-end*, conhece apenas a API do serviço que deseja consumir conforme ilustrado na *Figura 1*.

Construir o *Front-end* sob a abordagem *Single Page Application* (SPA) empregando *template*, o qual deverá ser processado exclusivamente no lado do cliente.

Usar componentes para criar módulos e módulos para páginas. Cada componente deverá ser testável individualmente e em colaboração, quando compuser módulo. Em qualquer dos casos, *mock* ou *stub* deverão ser suficientes para que o componente funcione corretamente.

Priorizar o uso de *Dependency Injection* (Injeção de Dependências - DI) em detrimento de instâncias diretas no código de negócios. Nesse contexto, usar gerenciador de *DI* nativo do *Framework* em uso.

Construir componente estruturado por *template* HTML, o qual permita ter sua aparência especializada por folha de estilo própria sem que ela influencie nos demais componentes e tenha uma classe para gerenciamento de comportamento dos elementos que o compõe.

Codificar o *Front-end*, por conseguinte todos os seus componentes, usando TypeScript em sua última versão estável. Encapsular regras de negócios de *Front-end* em classe de serviço, a qual fará ligação entre o componente, *Front-end*, e o *Back-end*. A *Figura 2* ilustra o fluxo entre Componente, Serviço e *Back-end*.

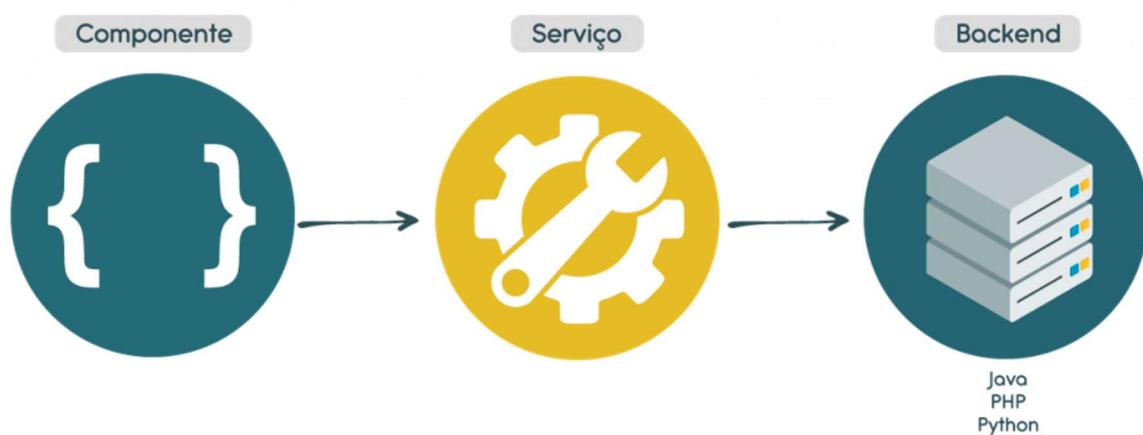


Figura 2 Comunicação entre componente, serviço e Back-end

Carregar a parte essencial do sistema para máquina do cliente. A partir desse ponto, cada requisição demanda apenas os dados que lhe são essencialmente necessários.

Realizar requisições ao *endpoint* através de sua API, via *Representational State Transfer* (REST), o qual retornará em formato *JavaScript Object Notation* (JSON) válido.

Empregar verbos HTTP para denotar o tipo de operação que se deseja executar contra um *endpoint*. A *Tabela 1* elenca os verbos possíveis para realizar requisições ao *endpoint*.

Verbo HTTP	Descrição
<b>POST (Create)</b>	Criar recurso Alguma alteração poderá ser produzida por uma requisição PUT
<b>GET (Read)</b>	Recuperar informação de um recurso ou uma coleção deles Nenhuma alteração será produzida por uma requisição GET
<b>PUT(Update/Replace)</b>	Atualizar total ou parcialmente um ou mais recursos Substituir um ou mais recursos Alguma alteração poderá ser produzida por uma requisição PUT
<b>DELETE (Delete)</b>	Excluir um ou mais recursos Alguma alteração poderá ser produzida por uma requisição DELETE

*Tabela 1 Emprego de verbos HTTP*

A *Tabela 1* tem o objetivo de correlacionar os verbos HTTP às operações. As especificidades requeridas pelo uso de cada verbo deverão ser observadas conforme definição no documento *REST API Best Practices***Erro! Indicador não definido..**

Usar código HTTP para interpretar o *status* da operação solicitada. Esta diretriz tem alinhamento direto com o *Back-end*, pois, é nesta camada que os códigos listados na *Tabela 2* são definidos e retornados ao *Front-end*.

Código HTTP	Objetivo
<b>200</b>	Ok Informar que a requisição fora atendida adequadamente
<b>204</b>	Ok Informar que a requisição fora atendida adequadamente, mas não retorna nenhum conteúdo
<b>304</b>	Not Modified Informar que o recurso solicitado não sofreu modificação desde a última requisição
<b>400</b>	Bad request Informar ocorrência de erro negocial
<b>401</b>	Unauthorized Informar que o recurso solicitado requer autenticação
<b>403</b>	Forbidden Informar que a credencial apresentada não permite o uso do recurso solicitado
<b>404</b>	Not found Informar que o recurso solicitado não foi encontrado
<b>412</b>	Precondition failed Informar erro de validação dos dados enviados
<b>500</b>	Internal Server Error Informar erro não tratado no servidor

*Tabela 2 Código HTTP usados pela API*

O emprego de *REST* deverá observar as seguintes restrições, as quais estão sucintamente descritas na *Tabela 3*. A versão completa e detalhada de cada uma das restrições apresentadas na Tabela 3, poderá ser conferida no documento *REST API Best Practices*<sup>1</sup>.

Princípio	Objetivo
<b>Uniform Interface</b>	Simplificar e desacoplar a arquitetura Evoluir partes da aplicação de forma independente
<b>Resource-Based</b>	Identificar recursos (serviço) univocamente Retornar ao cliente apenas o estritamente necessário Evitar desperdício de recurso de rede
<b>Manipulation of Resources Through Representations</b>	Permitir que o cliente tenha informações suficientes para modificar ou excluir um recurso no servidor, desde que possua permissão para tanto
<b>Self-descriptive Messages</b>	Incluir informações suficientes para descrever como processar a mensagem em cada mensagem
<b>Stateless</b>	Permitir extração de todas as informações de estado da própria requisição, as quais serão enviadas como parte da parte da URI
<b>Cacheable</b>	Indicar, explicitamente, a capacidade de <i>cacheamento ou não de um recurso</i>
<b>Client–server</b>	Definir interfaces uniformes entre <i>Cliente</i> e <i>Servidor</i> Separar responsabilidades de cliente no cliente e de servidor no servidor. Exemplo: Ao cliente, jamais, deverá ser atribuído a responsabilidade de armazenamento de dados e ao servidor, não cabe depender da versão de navegador para que sua regra de negócio funcione corretamente <sup>2</sup>
<b>Layered system</b>	Limitar o conhecimento do cliente sobre o local real de armazenamento do recurso por ele requisitado. Cabe-lhe apenas saber que o recurso está disponível em um dado <i>endpoint</i> e não onde o endpoint está hospedado

*Tabela 3 Restrições Arquiteturais do REST*

1 A versão completa do documento que detalha cada uma das restrições está disponível em:

<https://tinyurl.com/rest-espect>

2 Salvo quando a própria regra de negócio versar sobre isso.

#### 4.1 Service Oriented Front End Architecture – SOFEA

Todas as premissas e restrições apresentadas no item 4 formam a base para o desenvolvimento de software aplicando o paradigma, adotado pela CAPES, **SOFEA**.

Nesse modelo, logo na primeira requisição, todo o HTML, *templates*, JavaScript e CSS, **essenciais e necessários** à execução da aplicação são enviados para o cliente e, a partir daí requisições subsequentes e sob demanda são realizadas para complementar as funcionalidades da aplicação.

A Figura 3 ilustra de forma geral o processo de inicialização de uma aplicação sob a arquitetura SOFEA.

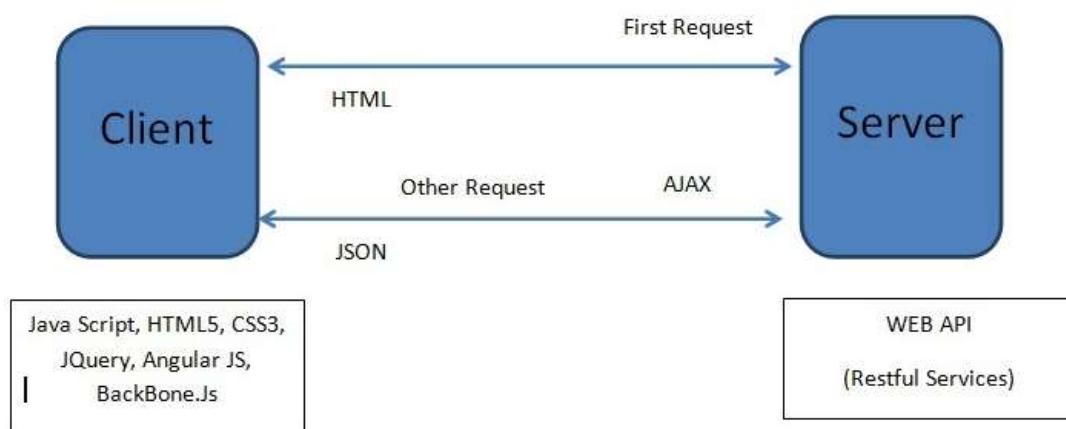


Figura 3 Visão geral de funcionamento de aplicação SOFEA

Uma das grandes dificuldades em se criar aplicações cliente/servidor web é gerenciar a expectativa do *download* inicial de grandes arquivos (HTML/CSS/JavaScript) necessários à inicialização da aplicação. A aplicação *Front-end* deve ter sempre em foco a experiência do usuário. Com isso, ter estratégias bem definidas para que o cliente seja capaz de interagir com o sistema o mais rápido possível.

Cada aplicação deverá ter um *bootstrap* mínimo e o menor possível para iniciar aplicação de modo a coloca-la operacional.

Deve atentar para o fato que, em uma primeira análise, aplicações em arquitetura SOFEA se confunde, em certa medida, com aplicações que utilizam o fluxo Ajax.

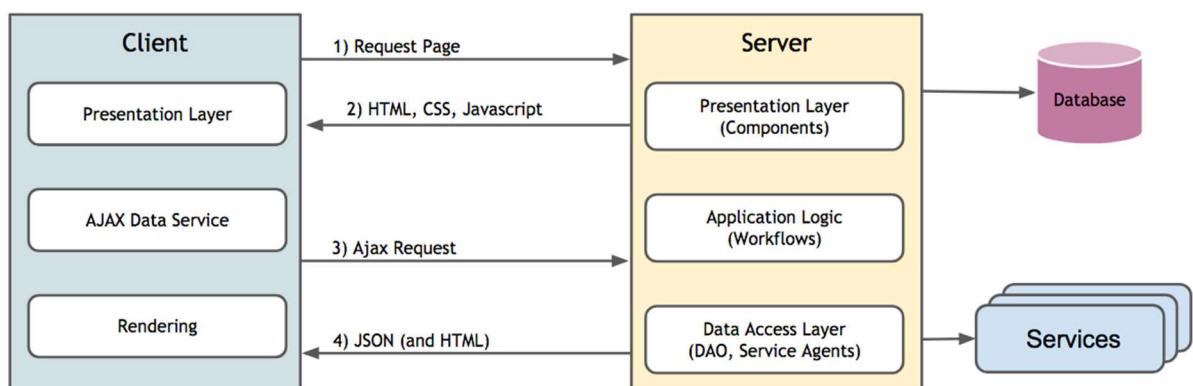


Figura 4 Arquitetura de aplicação web usando fluxo Ajax

Conforme observa-se na Figura 4, em uma aplicação clássica usando Ajax para baixar os *templates*, a camada de apresentação, responsável por gerenciar o que será mostrado ao usuário, fica parte no cliente e parte no servidor. Outro ponto é a camada que gerencia o fluxo de trabalho ou de eventos de tela (*workflow*), ela fica inteiramente no servidor e conforme vai sendo solicitada, o servidor faz o processamento da regra de negócios e também o de *workflow*.

Sob o prisma de aplicações SOFEA, também conhecida por *Thin Server Architecture*, preconiza que a lógica de renderização deverá ser exclusiva do cliente, eliminando do servidor a inteligência de geração de *templates* ou processamentos de *views*.

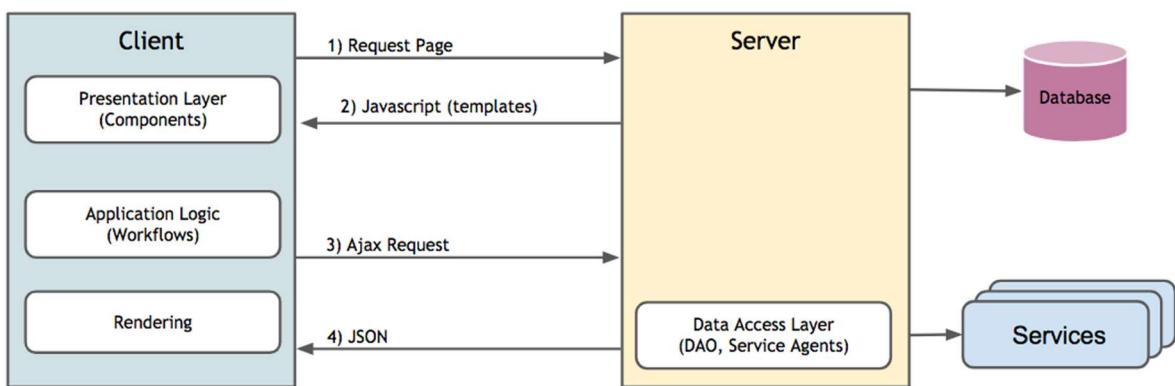


Figura 5 Arquitetura de aplicação Single-page Application

Com SOFEA, toda a responsabilidade de gerenciamento de apresentação fica do lado do cliente. Ela visa alcançar 3 objetivos principais:

- O programador Back-end pode focar na lógica de negócios da aplicação
- A aplicação torna-se menos complexa, uma vez que o *Back-end* e *Front-end* são separados
- Comunicação entre Cliente e Servidor usa um protocolo, sendo que cada um pode exportar, importar e apresentar dados para outro sistema (REST)

A arquitetura de aplicações em SOFEA permite em grande medida a separação de conteúdo estático do dinâmico. Isso também permite aliviar a carga dos servidores e diminuir o consumo de banda. Com essa diretriz em pauta, o Front-end deverá ser projetado para ter partes consumidas de diferentes provedores de conteúdo (*Content Delivery Network – CDN*).

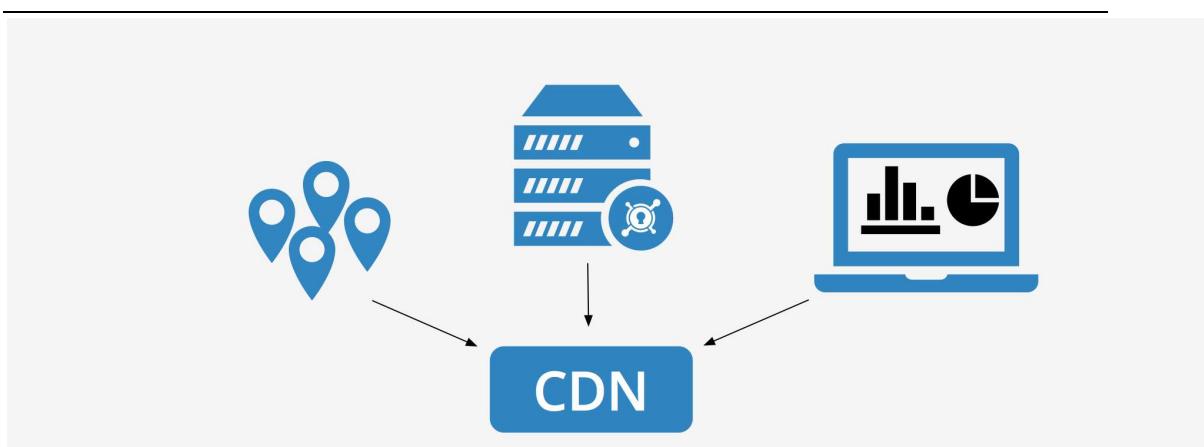


Figura 6 Arquitetura CDN

Nessa arquitetura, aplicações, clientes e serviços podem demandar conteúdo de uma CDN, basta que haja permissão suficiente para tal. Isto implica que o *Front-end* deverá prever mecanismos de consumo tanto de serviços de autenticação quanto identificar-se para outros serviços – saber gerir de *token* quando necessário.

## 5. Metas e Restrições Arquiteturais

Os indicadores de qualidade que cada sistema deverá apresentar ao ser submetido à avaliação por meio de ferramenta Sonarqube, compreende em cada indicador uma meta a ser alcançada. No que tange ao código-fonte, o quadro a seguir relaciona os indicadores e sua respectiva meta.

Grupo	Indicador	Tipo Meta	Meta
PROJETO	Problemas confirmados	Unidades	= 0
	Complexidade	Média total	<= 10
	Métodos	Média total	<= 3
	Índice de manutenibilidade	Nota	A
	Índice de confiabilidade	Nota	A
	Índice de segurança	Nota	A
	Taxa de dívida técnica	%	<= 2,5%
	Classes	Média total	<= 10
	Arquivos	Média total	<= 10
VIOLAÇÕES	Linhas duplicadas (%)	%	<= 4%
	Problemas impeditivos	Unidades	= 0
TESTE	Problemas críticos	Unidades	= 0
	Testes unitários ignorados	Unidades	= 0
	Sucesso em testes unitários (%)	%	= 100%
	Cobertura (camada de negócio)	%	>= 70%

Os projetos devem ser aderentes ao modelo de acessibilidade do governo eletrônico (EMAG), especialmente quanto a identificação dos campos e mensagens de ajuda para navegação na interface.

## 5.1 Camada

Os itens apresentados a seguir deverão ser entendidos como camadas, salvo direcionamento contrário, mesmo que em outras linguagens tenha entendimento diverso. Ainda, salvo menção explícita, sempre será referenciada apenas *Front-end* e embora sejam adotadas referências da arquitetura Angular, elas poderão ser reproduzidas por qualquer outro Framework desde que atendidas as especificações deste documento.

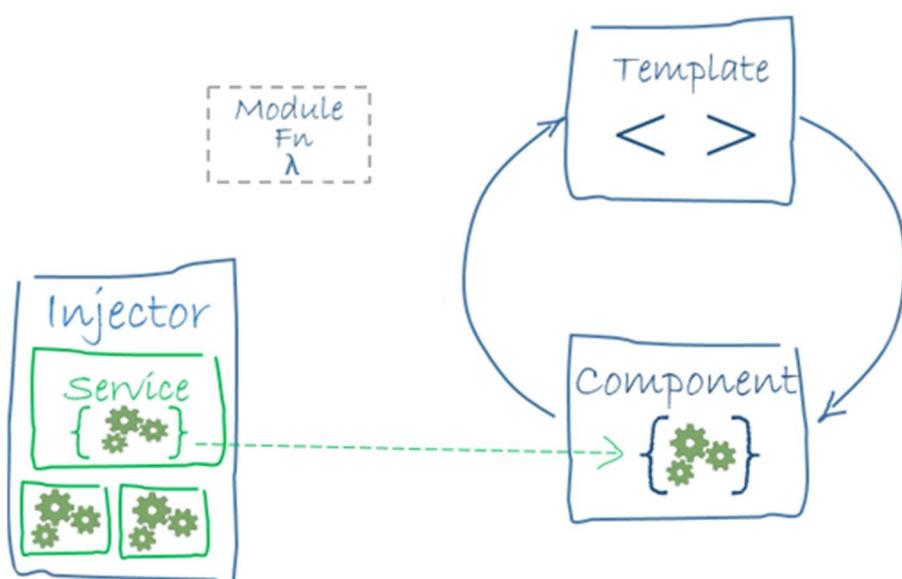


Figura 7 Arquitetura de componentes

### 5.1.1 Módulo (Module)

Agrupar e coordenar um conjunto de um ou mais componentes relacionados a uma mesma solução.

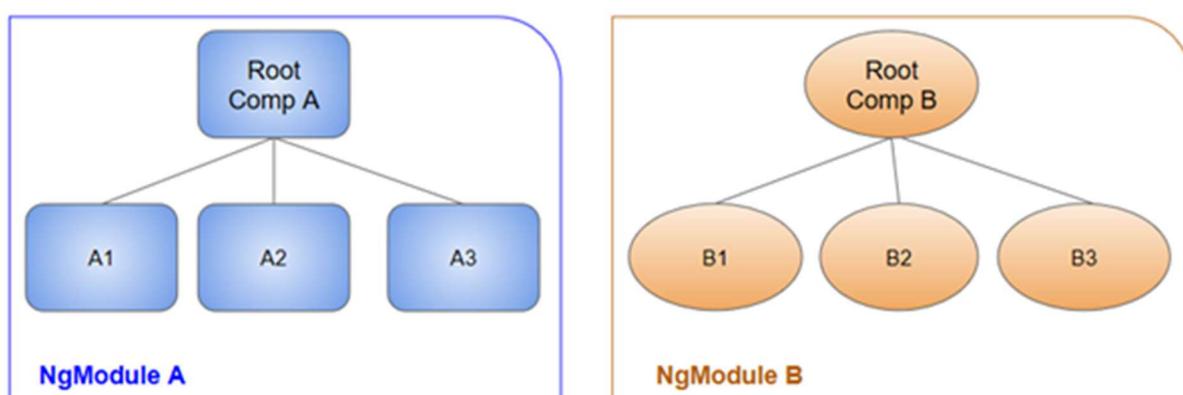


Figura 8 Estrutura de Módulo

Conforme ilustrado na *Figura 8*, um módulo, *Root Comp A*, contém um e somente componente raiz, mas pode interagir com vários componentes. Por sua vez, uma aplicação pode ser composta por vários módulos.

Um componente pode usar tanto seus *templates*, *views*, quanto de outros componentes, vide *Figura 9*.

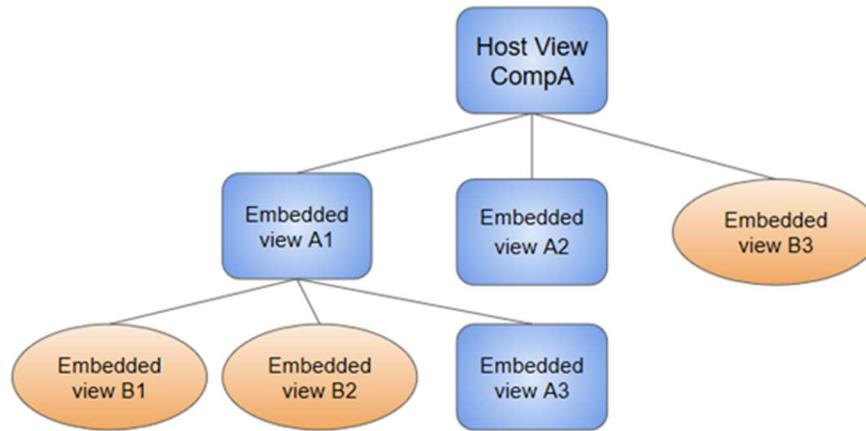


Figura 9 Esquema de uso de template

### 5.1.2 Componente (Component)

Controlar parte da tela, *view*. Cada componente controla apenas eventos, renderização e demais comportamentos de elementos relacionados a sua própria *view*. Um componente poderá se comunicar com outros por meio de mensagens.

A estrutura básica de um componente é composta, mas não limitada a: *Template*, *Service*, Funções.

#### 5.1.2.1 HTML (Template)

Representação visual do conteúdo manipulado pelo componente. O *template* deverá ter arquivo próprio para agrupar todo HTML necessário à sua representação.

#### 5.1.2.2 Service

Um serviço pode estar vinculado a um componente, nesse caso, deverá ser vinculado a própria estrutura do componente ou servir a vários componentes distintos, quando assim, deverá ser estruturado de forma global. Ter classe própria e objetivo específico.

#### 5.1.2.3 Funções

Agrupar código que não se enquadre em uma *service*, mas se faz necessário a um componente.

## 5.2 Front-end

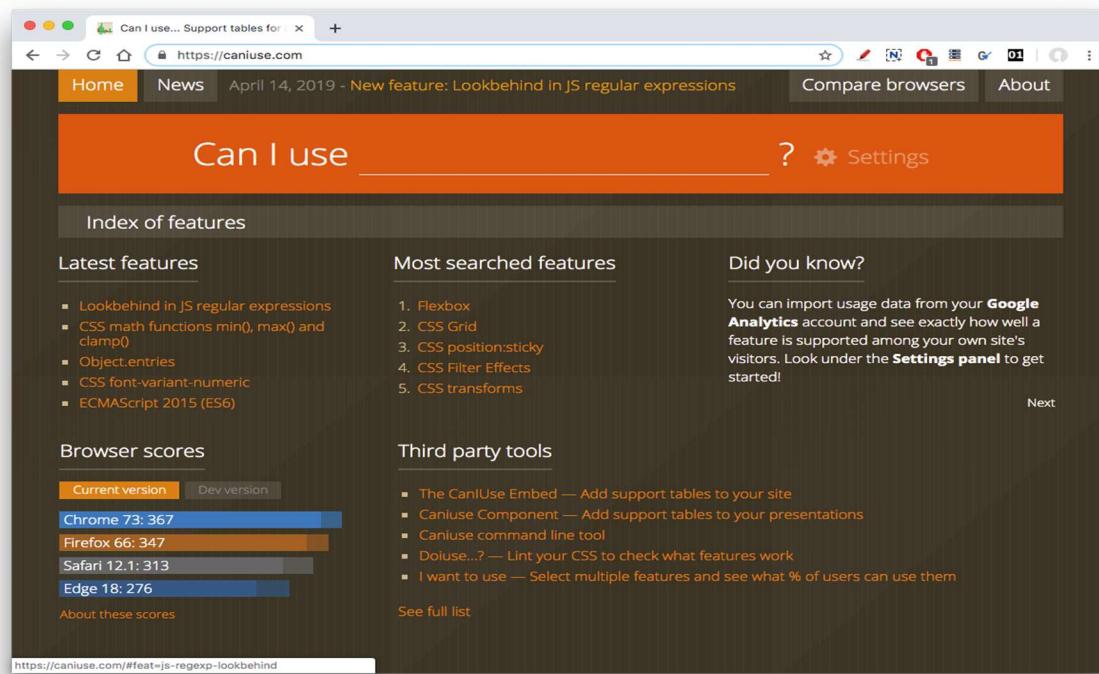
Esta seção define a lista de clientes, navegadores, que a camada de Front-end deverá suportar.

A *Tabela 4* lista os navegadores que deverão ser suportados pelas aplicações Front-end CAPES. Isso implica que todas as *tags HTML*, *CSS* e funcionalidades *Javascript* empregadas na construção de telas e comportamentos/eventos deverão ter *status* de *Supported* por seus respectivos fabricantes.

Navegador Desktop	Versão mínima
Google Chrome	73
Mozilla Firefox	66
Apple Safari	12
Microsoft Edge	12

*Tabela 4* Lista de clientes suportados pelas aplicações CAPES

A verificação do suporte de *tag HTML*, *CSS* e *Javascript* pode ser realizada em: <https://caniuse.com>.



The screenshot shows the CanIUse.com homepage. The main header says "Can I use" and has a search bar. Below the header, there are several sections: "Index of features", "Latest features" (listing Lookbehind in JS regular expressions, CSS math functions min(), max() and clamp(), Object.entries, CSS font-variant-numeric, and ECMAScript 2015 (ES6)), "Most searched features" (listing Flexbox, CSS Grid, CSS position:sticky, CSS Filter Effects, and CSS transforms), "Did you know?", "Browser scores" (showing Chrome 73: 367, Firefox 66: 347, Safari 12.1: 313, and Edge 18: 276), and "Third party tools" (listing The CanIUse Embed, Caniuse Component, Caniuse command line tool, Doluse...?, and I want to use). The URL in the address bar is https://caniuse.com/#feat=js-regexp-lookbehind.

*Figura 10* URL caniuse

## 5.3 Componente

A CAPES já possui alguns componentes de estrutura básica para auxiliar o desenvolvimento da interface do sistema.



---

Os componentes desenvolvidos apenas complementam a arquitetura de componentes já existente nas aplicações, de forma a acelerar o desenvolvimento, entretanto podem ser substituídos caso seja necessário.

Os componentes desenvolvidos auxiliam o suporte à validação de campos de tela, processamento de notificações, logging, autenticação e acesso a serviços de backend via rest.

### 6. Requisitos de Qualidade

O Índice de Qualidade (IQ) será obtido a partir de indicadores de qualidade do código-fonte do software aferidos com o apoio da ferramenta SonarQube. Alguns pontos que serão mais cuidadosamente analisados com base nos relatórios do sonar são: cobertura de testes, complexidade ciclomática, acoplamento e coesão.

O acesso ao SonarQube é livre para todos os desenvolvedores e deve ser incentivado dentro das equipes como meio de adequação e qualidade de código.

Além dos itens analisados pelo SonarQube, o código deverá ser submetido a testes de carga (estresse) pela ferramenta Apache JMeter a qual demanda script JMeter capaz de produzir o referido teste.

### 7. Definições, acrônimos e abreviações

- Backend – Código executado no lado do servidor
- *Frontend* – Código executado no lado cliente
- Jenkins – <http://jenkins.capes.gov.br>
- Nexus – <http://nexus.capes.gov.br>
- Sonar – <http://sonar.capes.gov.br>
- GITLab – <http://git.capes.gov.br>
- Wiki – <http://wiki.capes.gov.br>
- Angular – <https://angular.io>
- React – <https://reactjs.org>
- Vuejs – <https://vuejs.org>
- JSON – <https://json.org>

---

**8. Referências**

- [https://wiki.capes.gov.br/index.php/DTI:Politica\\_de\\_Geracao\\_de\\_Builds\\_para\\_Deploy](https://wiki.capes.gov.br/index.php/DTI:Politica_de_Geracao_de_Builds_para_Deploy)
- [https://wiki.capes.gov.br/index.php/DTI:Visao\\_geral\\_tecnologias\\_integradas](https://wiki.capes.gov.br/index.php/DTI:Visao_geral_tecnologias_integradas)
- Crie aplicações com Angular: O novo framework do Google, Thiago Guedes, Casa do código
- Guia Front-End: O caminho das pedras para ser um dev Front-end, Diego Eis, Casa do código
- Building Isomorphic JavaScript Apps: From Concept to Implementation to Real-World Solutions, Maxime Najim, Jason Strimpel, O'Reilly
- Angular, disponível em: <https://angular.io>, acessado em: 08/04/2019
- Angular + Jasmine, disponível em: <https://medium.com/angularbr/angular-5-criando-testes-com-jasmine-175170612ed8>, acessado em: 15/04/2019
- SPA, disponível em: <https://blog.pshrmn.com/entry/how-single-page-applications-work>, acessado em: 10/04/2019
- REST, disponível em: <https://restfulapi.net>, acessado em: 10/04/2019
- REST - Restrições, disponível em: 10/03/2019  
<https://www.restapitutorial.com/resources.html>, acessado em: 15/04/2019
- REST - Princípios e boas práticas, disponível em: <https://blog.caelum.com.br/rest-principios-e-boas-praticas/>, acessado em: 10/04/2019
- MCV, disponível em: <http://heim.ifi.uio.no/~trygver/themes/mvc/mvc-index.html>, acessado em: 15/04/2019
- JSON, disponível em: <http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf>, acessado em: 08/04/2019
- IoC, disponível em: [https://pt.wikipedia.org/wiki/Inversão\\_de\\_controle](https://pt.wikipedia.org/wiki/Inversão_de_controle), acessado em: 08/04/2019
- CDN, disponível em: <https://www.keycdn.com/what-is-a-cdn>, acessado em: 09/04/2019
- HTML5, disponível em: <https://html5forwebdesigners.com/history/index.html>, acessado em: 24/04/2019
- HTTP Status, disponível em: <https://developer.mozilla.org/pt-BR/docs/Web/HTTP>Status>, acessado em: 19/04/2019
- HTTP Methods, disponível em: <https://www.restapitutorial.com/lessons/httpmethods.html>, acessado em: 19/04/2019
- MVC/MVP/MVVM, disponível em: <https://medium.com/@FilipeFNunes/android-mvc-x-mvp-x-mvvm-qual-pattern-utilizar-parte-1-3defc5c89af>, acessado em: 22/04/2019
- SOFEA, disponível em: [dzone.com/articles/service-oriented-front-end-architecture-sofea](https://dzone.com/articles/service-oriented-front-end-architecture-sofea), acesso em: 08/04/2019

**9. Anexo A – Tecnologias**

NOME	VERSÃO	DESCRIÇÃO
PHP	7.3+	Interpretador PHP

**10. Anexo B – Componentes**

NOME	VERSÃO	DESCRIÇÃO
@capes/ngx	3.1.2	Este projeto agrupa todas as dependências necessárias para a montagem de projetos angulares.
@capes/ngx-validation	3.1.1	Contém regras de validação para uso em formulários reativos.
@capes/ngx-auth	3.1.1	Contém os códigos necessários para o processamento de autenticação e autorização.
@capes/ngx-base	3.1.1	Contém as classes de base (template method) padronizando a construção de interfaces em angular.
@capes/ngx-rest	3.1.1	Contém as classes de base para a construção de serviços de comunicação via rest com o backend.
@capes/ngx-ui-forms	3.1.1	Este pacote tem os componentes usados para processamento de formulários.
@capes/ngx-forms	3.1.1	Componente que cria controles para formulários reativos.
@capes/ngx-notification	3.1.1	Este pacote permite o uso de notificações em aplicações, trata-se de um componente do pacote @capes/ngx e deve



## Arquitetura de Referência – Guia Frontend

Versão: 1.1

Data Emissão:

03/06/2019

NOME	VERSÃO	DESCRIÇÃO
		ser instalado pelo @capes/ngx.
@capes/ngx-logging	3.1.1	Contém os componentes para gerenciamento de logging.
@capes/ngx-ui-base	3.1.1	Contém classes base para construção de componentes de interface.
@capes/ngx-ui-mask	3.1.1	Este pacote adiciona suporte de máscaras em aplicações angulares.
@capes/ngx-memory-storage	3.1.1	Memory storage resolve o problema de abertura de abas no browser.
@capes/ngx-utils	3.1.1	Contém utilitários a programação.



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE M - MODELO DE ORDEM DE SERVIÇO

##### 1. GRUPO 1 - DESENVOLVIMENTO DE SISTEMAS

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
Ordem de Serviço – Contrato nº / _____ Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.	Nº da OS: _____
<b>Tipo de OS:</b>	<input type="checkbox"/> Projeto <input type="checkbox"/> Manutenção
<b>Nome do Projeto/Sistema(s):</b>	
<b>Data de Abertura:</b>	
<b>Data Prevista de Término:</b>	
<b>Quantitativo de Postos de Trabalho:</b>	<input type="checkbox"/> Analista Programador nível Sênior - Java <input type="checkbox"/> Analista Programador nível Pleno- Java <input type="checkbox"/> Analista Programador nível Júnior- Java <input type="checkbox"/> Analista Programador nível Sênior - PHP <input type="checkbox"/> Analista Programador nível Pleno - PHP <input type="checkbox"/> Analista Programador nível Júnior - PHP <input type="checkbox"/> Analista Programador nível Sênior - Python <input type="checkbox"/> Analista Programador nível Pleno - Python <input type="checkbox"/> Analista Programador nível Júnior - Python
<b>Valor estimado (R\$):</b>	
<b>Representantes da CAPES:</b>	Nome - Matrícula (Fiscal Requisitante)  Nome - Matrícula (Gestor do Contrato)
<b>Ciência do representante da empresa:</b>	Nome – Cargo na empresa

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR
---

## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

Aditivo nº _____ à Ordem de Serviço – Contrato nº _____	Nº da OS: _____
Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.	
Tipo de OS:	( ) Projeto ( ) Manutenção
Data do Aditivo:	
	<input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Sênior - Java <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Pleno - Java <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Júnior- Java <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Sênior - PHP <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Pleno - PHP <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Júnior - PHP <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Sênior - Python <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Pleno - Python <input type="checkbox"/> Aumento de posto(s) de trabalho de Analista Programador nível Júnior - Python  <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Sênior - Java <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Pleno - Java <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Júnior - Java <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Sênior - PHP <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Pleno - PHP <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Júnior - PHP <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Sênior - Python <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Pleno - Python <input type="checkbox"/> Redução de posto(s) de trabalho de Analista Programador nível Júnior - Python  <input type="checkbox"/> Prorrogação do Término da OS para ____/____/____ <input type="checkbox"/> Antecipação do Término da OS para ____/____/____
Valor estimado atualizado (R\$):	

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR	
<b>Representantes da CAPES:</b>	Nome - Matrícula (Fiscal Requisitante)
	Nome - Matrícula (Gestor ou Fiscal do Contrato)
<b>Ciência do representante da empresa:</b>	Nome – Cargo na empresa

## 2. GRUPO 2 - MENSURAÇÃO DE SISTEMA

COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR					
Ordem de Serviço – Contrato nº / _____ Objeto: Serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.	Nº da OS: _____				
<b>Tipos de Contagem:</b>	<input type="checkbox"/> Validação e/ou Auditagem <input type="checkbox"/> Estimada <input type="checkbox"/> Detalhada				
<b>Produtos/serviços esperados:</b>					
<b>Data de Abertura:</b>					
<b>Data Prevista de Término:</b>					
<b>Formas de avaliação dos serviços executados:</b>					
<b>Estimativa de cronograma orçamentário:</b>	<b>Serviço</b>	<b>Qtde</b>	<b>Valor (R\$)</b>	<b>Deflator</b>	<b>Valor final (R\$)</b>
	<b>Total:</b>				
<b>Representantes da CAPES:</b>	Nome - Matrícula (Fiscal Requisitante)				
	Nome - Matrícula (Gestor ou Fiscal do Contrato)				
<b>Ciência do representante da empresa:</b>	Nome – Cargo na empresa				



COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

**ENCARTE TÉCNICO CONTRATAÇÃO - TI**

**ENCARTE N - MODELO DE DECLARAÇÃO DE CONTRATOS FIRMADOS COM A INICIATIVA PRIVADA E COM A ADMINISTRAÇÃO PÚBLICA**

Declaramos que a empresa \_\_\_\_\_, inscrita no CNPJ (MF) nº \_\_\_\_\_, possui os seguintes contratos firmados com a iniciativa privada e administração pública:

Órgão/Empresa	Nº Contrato/ Aditivo	Vigência		Valor Total Atual do Contrato
		Início	Término	
Valor Total dos Contratos (Atualizados)				

Local e data

\_\_\_\_\_  
Assinatura e carimbo do emissor

**COMPROVAÇÃO DA SUBCONDIÇÃO 14.12.2.2 - item 'd'.**

Cálculo demonstrativo visando comprovar que o patrimônio líquido é igual ou superior a 1/12 (um doze avos) do valor dos contratos firmados com a administração pública e com a iniciativa privada.

Valor do Patrimônio Líquido x 12 > 1 Valor total dos contratos

Obs.: Esse resultado deverá ser superior a 1.

**COMPROVAÇÃO DA CONDIÇÃO 14.12.2.3.**

Cálculo demonstrativo da variação percentual do valor total constante na declaração de contratos firmados com a iniciativa privada e com a Administração Pública em relação à receita bruta informada na DRE.

$(\text{Valor da receita bruta} - \text{Valor total dos contratos}) \times 100 = X\% \text{ Valor da receita bruta}$

Obs.: Caso o percentual encontrado seja maior que 10% (positivo ou negativo), a licitante deverá apresentar as devidas justificativas.



**COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR****ENCARTE TÉCNICO CONTRATAÇÃO - TI****ENCARTE O - MODELO DE ATESTADO DE VISITA TÉCNICA**

Objeto: Contratação de serviços de desenvolvimento e manutenção de software com práticas ágeis e mensuração de sistemas de informação.

<b>Identificação da Empresa</b>	
<b>Data:</b>	
<b>Empresa Licitante:</b>	
<b>CNPJ:</b>	

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (Capes) atesta que a empresa acima identificada, por meio de seu representante técnico abaixo qualificado, efetuou, nesta data, vistoria nas dependências da sede da Capes, com o objetivo de conhecer o ambiente técnico e tecnológico, bem como as condições e detalhes da prestação de serviços licitados pelo Edital em referência. A empresa licitante declara que todas as dúvidas foram sanadas havendo concordância com os termos de Edital referenciado.

<b>Identificação do Representante da Empresa</b>	
<b>Nome:</b>	
<b>Cargo:</b>	
<b>CPF ou RG:</b>	

<b>Representante da Capes</b>	
<b>Carimbo e assinatura</b>	<dd/mm/aaaa>



## COORDENAÇÃO DE APERFEIÇOAMENTO DE PESSOAL DE NÍVEL SUPERIOR

### ENCARTE TÉCNICO CONTRATAÇÃO - TI

#### ENCARTE P - MODELO DE TERMO DE CONFIDENCIALIDADE E SIGILO

A **Coordenação de Aperfeiçoamento de Pessoal de Nível Superior**, sediada em Brasília, Setor Bancário Norte, Quadra 2, Bloco L, CNPJ n.º 00889834/0001-08, doravante denominado **CONTRATANTE**, e, de outro lado, a *<NOME DA EMPRESA>*, sediada em *<ENDEREÇO>*, CNPJ n.º *<CNPJ>*, doravante denominada **CONTRATADA**.

CONSIDERANDO que, em razão do **CONTRATO N.º XX/20XX** doravante denominado **CONTRATO PRINCIPAL**, a **CONTRATADA** poderá ter acesso a informações sigilosas do **CONTRATANTE**;

CONSIDERANDO a necessidade de ajustar as condições de revelação destas informações sigilosas, bem como definir as regras para o seu uso e proteção;

CONSIDERANDO o disposto na Política de Segurança da Informação da **CONTRATANTE**;

**Resolvem celebrar o presente TERMO DE CONFIDENCIALIDADE, SIGILO E COMPROMISSO, doravante TERMO, vinculado ao CONTRATO PRINCIPAL, mediante as seguintes cláusulas e condições:**

#### Cláusula Primeira – DO OBJETO

Constitui objeto deste TERMO o estabelecimento de condições específicas para regulamentar as obrigações a serem observadas pela **CONTRATADA**, no que diz respeito ao trato de informações sigilosas, disponibilizadas pela **CONTRATANTE**, por força dos procedimentos necessários para a execução do objeto do **CONTRATO PRINCIPAL** celebrado entre as partes e em acordo com o que dispõem a a Lei 12.527, de 18/11/2011 e os Decretos 7.724, de 16/05/2012 e 7.845, de 14/11/2012, que regulamentam os procedimentos para acesso e tratamento de informação classificada em qualquer grau de sigilo.

#### Cláusula Segunda – DOS CONCEITOS E DEFINIÇÕES

Para os efeitos deste TERMO, são estabelecidos os seguintes conceitos e definições:

**Informação:** dados, processados ou não, que podem ser utilizados para produção e transmissão de conhecimento, contidos em qualquer meio, suporte ou formato.

**Informação Pública ou Ostensiva:** são aquelas cujo acesso é irrestrito, obtida por divulgação pública ou por meio de canais autorizados pela **CONTRATANTE**.

**Informações Sensíveis:** são todos os conhecimentos estratégicos que, em função de seu potencial no aproveitamento de oportunidades ou desenvolvimento nos ramos econômico, político, científico, tecnológico, militar e social, possam beneficiar a Sociedade e o Estado brasileiros.

**Informações Sigilosas:** aquela submetida temporariamente à restrição de acesso público em razão de sua impescindibilidade para a segurança da sociedade e do Estado.

**Contrato Principal:** contrato celebrado entre as partes, ao qual este TERMO se vincula.

### **Cláusula Terceira – DAS INFORMAÇÕES SIGILOSAS**

Serão consideradas como informação sigilosa, toda e qualquer informação escrita ou oral, revelada a outra parte, contendo ou não a expressão confidencial e/ou reservada. O TERMO informação abrangerá toda informação escrita, verbal, ou em linguagem computacional em qualquer nível, ou de qualquer outro modo apresentada, tangível ou intangível, podendo incluir, mas não se limitando a: know-how, técnicas, especificações, relatórios, compilações, código fonte de programas de computador na íntegra ou em partes, fórmulas, desenhos, cópias, modelos, amostras de ideias, aspectos financeiros e econômicos, definições, informações sobre as atividades da CONTRATANTE e/ou quaisquer informações técnicas/comerciais relacionadas/resultantes ou não ao CONTRATO PRINCIPAL, doravante denominados INFORMAÇÕES, a que diretamente ou pelos seus empregados, a CONTRATADA venha a ter acesso, conhecimento ou que venha a lhe ser confiada durante e em razão das atuações de execução do CONTRATO PRINCIPAL celebrado entre as partes.

Parágrafo Primeiro – Comprometem-se, as partes, a não revelar, copiar, transmitir, reproduzir, utilizar, transportar ou dar conhecimento, em hipótese alguma, a terceiros, bem como a não permitir que qualquer empregado envolvido direta ou indiretamente na execução do CONTRATO PRINCIPAL, em qualquer nível hierárquico de sua estrutura organizacional e sob quaisquer alegações, faça uso dessas informações, que se restringem estritamente ao cumprimento do CONTRATO PRINCIPAL.

Parágrafo Segundo – As partes deverão cuidar para que as informações sigilosas fiquem restritas ao conhecimento das pessoas que estejam diretamente envolvidas nas atividades relacionadas à execução do objeto do CONTRATO PRINCIPAL.

Parágrafo Terceiro – As obrigações constantes deste TERMO não serão aplicadas às INFORMAÇÕES que:

- I - Sejam comprovadamente de domínio público no momento da revelação;
- II - Tenham sido comprovadas e legitimamente recebidas de terceiros, estranhos ao presente TERMO;
- III - Sejam reveladas em razão de requisição judicial ou outra determinação válida do Governo, somente até a extensão de tais ordens, desde que as partes cumpram qualquer medida de proteção pertinente e tenham sido notificadas sobre a existência de tal ordem, previamente e por escrito, dando a esta, na medida do possível, tempo hábil para pleitear medidas de proteção que julgar cabíveis.

### **Cláusula Quarta – DOS LIMITES DO SIGILO**

As obrigações constantes deste TERMO não serão aplicadas às INFORMAÇÕES que:

- I – Sejam comprovadamente de domínio público no momento da revelação, exceto se tal fato decorrer de ato ou omissão da CONTRATADA;
- II – Tenham sido comprovadas e legitimamente recebidas de terceiros, estranhos ao presente TERMO;
- III – Sejam reveladas em razão de requisição judicial ou outra determinação válida do Governo, somente até a extensão de tais ordens, desde que as partes cumpram qualquer medida de proteção pertinente e tenham sido notificadas sobre a existência de tal ordem, previamente e por escrito, dando a esta, na medida do possível, tempo hábil para pleitear medidas de proteção que julgar cabíveis.

### **Cláusula Quinta – DOS DIREITOS E OBRIGAÇÕES**

As partes se comprometem e se obrigam a utilizar a informação sigilosa revelada pela outra parte exclusivamente para os propósitos da execução do CONTRATO PRINCIPAL, em conformidade com o

disposto neste TERMO.

Parágrafo Primeiro – A CONTRATADA se compromete a não efetuar qualquer tipo de cópia da informação sigilosa sem o consentimento expresso e prévio da CONTRATANTE.

Parágrafo Segundo – A CONTRATADA compromete-se a dar ciência e obter o aceite formal da direção e empregados que atuarão direta ou indiretamente na execução do CONTRATO PRINCIPAL sobre a existência deste TERMO bem como da natureza sigilosa das informações.

I – A CONTRATADA deverá firmar acordos por escrito com seus empregados visando garantir o cumprimento de todas as disposições do presente TERMO e dará ciência à CONTRATANTE dos documentos comprobatórios.

Parágrafo Terceiro – A CONTRATADA obriga-se a tomar todas as medidas necessárias à proteção da informação sigilosa da CONTRATANTE, bem como evitar e prevenir a revelação a terceiros, exceto se devidamente autorizado por escrito pela CONTRATANTE.

Parágrafo Quinto – A CONTRATADA obriga-se por si, sua controladora, suas controladas, coligadas, representantes, procuradores, sócios, acionistas e cotistas, por terceiros eventualmente consultados, seus empregados, contratados e subcontratados, assim como por quaisquer outras pessoas vinculadas à CONTRATADA, direta ou indiretamente, a manter sigilo, bem como a limitar a utilização das informações disponibilizadas em face da execução do CONTRATO PRINCIPAL.

Parágrafo Sexto - A CONTRATADA, na forma disposta no parágrafo primeiro, acima, também se obriga a:

I – Não discutir perante terceiros, usar, divulgar, revelar, ceder a qualquer título ou dispor das informações, no território brasileiro ou no exterior, para nenhuma pessoa, física ou jurídica, e para nenhuma outra finalidade que não seja exclusivamente relacionada ao objetivo aqui referido, cumprindo-lhe adotar cautelas e precauções adequadas no sentido de impedir o uso indevido por qualquer pessoa que, por qualquer razão, tenha acesso a elas;

II – Responsabilizar-se por impedir, por qualquer meio em direito admitido, arcando com todos os custos do impedimento, mesmo judiciais, inclusive as despesas processuais e outras despesas derivadas, a divulgação ou utilização das Informações Proprietárias por seus agentes, representantes ou por terceiros;

III – Comunicar à CONTRATANTE, de imediato, de forma expressa e antes de qualquer divulgação, caso tenha que revelar qualquer uma das informações, por determinação judicial ou ordem de atendimento obrigatório determinado por órgão competente; e

IV – Identificar as pessoas que, em nome da CONTRATADA, terão acesso às informações sigilosas.

## **Cláusula Sexta – DA VIGÊNCIA**

O presente TERMO tem natureza irrevogável e irretratável, permanecendo em vigor desde a data de sua assinatura até expirar o prazo de classificação da informação a que a CONTRATADA teve acesso em razão do CONTRATO PRINCIPAL.

## **Cláusula Sétima – DAS PENALIDADES**

A quebra do sigilo e/ou da confidencialidade das informações, devidamente comprovada, possibilitará a imediata aplicação de penalidades previstas conforme disposições contratuais e legislações em vigor que tratam desse assunto, podendo até culminar na rescisão do CONTRATO PRINCIPAL firmado entre as PARTES. Neste caso, a CONTRATADA, estará sujeita, por ação ou omissão, ao pagamento ou recomposição de todas as perdas e danos sofridos pela CONTRATANTE, inclusive as de ordem moral, bem como as de responsabilidades civil e criminal, as quais serão apuradas em regular processo administrativo ou judicial, sem prejuízo das demais sanções legais cabíveis, conforme Art. 87 da Lei nº. 8.666/93.

## **Cláusula Oitava – DISPOSIÇÕES GERAIS**

Esse Termo de Confidencialidade é parte integrante e inseparável do CONTRATO PRINCIPAL.

Parágrafo Primeiro – Surgindo divergências quanto à interpretação do disposto neste instrumento, ou quanto à execução das obrigações dele decorrentes, ou constatando-se casos omissos, as partes buscarão solucionar as divergências de acordo com os princípios de boa fé, da eqüidade, da razoabilidade, da economicidade e da moralidade.

Parágrafo Segundo – O disposto no presente TERMO prevalecerá sempre em caso de dúvida e, salvo expressa determinação em contrário, sobre eventuais disposições constantes de outros instrumentos conexos firmados entre as partes quanto ao sigilo de informações, tal como aqui definidas.

Parágrafo Terceiro – Ao assinar o presente instrumento, a CONTRATADA manifesta sua concordância no sentido de que:

I – A CONTRATANTE terá o direito de, a qualquer tempo e sob qualquer motivo, auditar e monitorar as atividades da CONTRATADA;

II – A CONTRATADA deverá disponibilizar, sempre que solicitadas formalmente pela CONTRATANTE, todas as informações requeridas pertinentes ao CONTRATO PRINCIPAL.

III – A omissão ou tolerância das partes, em exigir o estrito cumprimento das condições estabelecidas neste instrumento, não constituirá novação ou renúncia, nem afetará os direitos, que poderão ser exercidos a qualquer tempo;

IV – Todas as condições, TERMOS e obrigações ora constituídos serão regidos pela legislação e regulamentação brasileiras pertinentes;

V – O presente TERMO somente poderá ser alterado mediante TERMO aditivo firmado pelas partes;

VI – Alterações do número, natureza e quantidade das informações disponibilizadas para a CONTRATADA não descharacterizarão ou reduzirão o compromisso e as obrigações pactuadas neste TERMO, que permanecerá válido e com todos seus efeitos legais em qualquer uma das situações tipificadas neste instrumento;

VII – O acréscimo, complementação, substituição ou esclarecimento de qualquer uma das informações disponibilizadas para a CONTRATADA, serão incorporados a este TERMO, passando a fazer dele parte integrante, para todos os fins e efeitos, recebendo também a mesma proteção descrita para as informações iniciais disponibilizadas, sendo necessário a formalização de TERMO aditivo a CONTRATO PRINCIPAL;

VIII – Este TERMO não deve ser interpretado como criação ou envolvimento das Partes, ou suas filiadas, nem em obrigação de divulgar Informações Sigilosas para a outra Parte, nem como obrigação de celebrarem qualquer outro acordo entre si.

### **Cláusula Nona – DO FORO**

A CONTRATANTE elege o foro da Seção Judiciária do Distrito Federal, onde está localizada a sede da CONTRATANTE, para dirimir quaisquer dúvidas originadas do presente TERMO, com renúncia a qualquer outro.

E, por assim estarem justas e estabelecidas as condições, o presente TERMO DE COMPROMISSO DE MANUTENÇÃO DE SIGILO é assinado pelas partes em 2 vias de igual teor e um só efeito.

De acordo,

---

**CONTRATANTE**

SIAPE: XXXXXXXXXXXX

\_\_\_\_ / \_\_\_\_ / \_\_\_\_

---

**Preposto da CONTRATADA**

CPF: XXXXXXXXXXXX

---

\_\_\_\_/\_\_\_\_/\_\_\_\_\_

---

**TESTEMUNHA**

CPF: XXXXXXXXXXXX

---

\_\_\_\_/\_\_\_\_/\_\_\_\_\_

---

**TESTEMUNHA**

CPF: XXXXXXXXXXXX

---

\_\_\_\_/\_\_\_\_/\_\_\_\_\_



**MINISTÉRIO DA EDUCAÇÃO  
FUNDAÇÃO CAPES  
DIRETORIA DE TECNOLOGIA DA INFORMAÇÃO  
MODELO DE TERMO DE CIENCIA**

<b>Contrato Nº</b>			
<b>Objeto:</b>			
<b>Contratante:</b>			
<b>Gestor do Contrato:</b>		<b>Matr.:</b>	
<b>Contratada:</b>		<b>CNPJ:</b>	
<b>Preposto da Contratada:</b>		<b>CPF:</b>	

Por este instrumento, os funcionários abaixo-assinados declaram ter ciência e conhecer o teor do Termo de Compromisso de Manutenção de Sigilo e as normas de segurança vigentes na Contratante.

, de de 20

Ciência,

CONTRATADA

---

<Nome> Matrícula: <Matr>

---

<Nome> Matrícula: <Matr>