

Automatização do gerenciamento de Fichas de Avaliação de Transgressões Disciplinares

Thalles Stellet Sanchez Sobral

Gabriel Teixeira Galam

Mario Bischoff Neto

Carlos Gabriel Agostinho Gonçalo Klink

Felipe Alves Manfrim

Resumo

Atualmente, a Força Aérea Brasileira vem se modernizando cada vez mais, adotando sistemas digitais para organização e gestão de seus recursos, de forma a economizar tempo, espaço e agilizar o gerenciamento de informações, necessário para o bom funcionamento de uma OM (Organização Militar). Entretanto, o sistema dedicado ao controle de punições dos militares ainda é "manual", sendo necessário um amplo espaço para o armazenamento de cada ficha individual de cada militar, atrasando toda e qualquer consulta nesses arquivos, além de gerar informações pouco precisas. Este projeto trata da elaboração de um banco de dados, possibilitando uma organização adequada destas informações, de forma a agilizar todo o processo de consulta e gerenciamento de dados.

Palavras-chave: banco de dados; RDAER; FATD; modernização.

1 Introdução

Recentemente, a FAB (Força Aérea Brasileira) vem adotando soluções tecnológicas para manutenção de sua capacidade operacional e de apoio ao Estado, por exemplo a aquisição do veículo aéreo não tripulado Hermes 900, para melhoria na vigilância durante os eventos da Copa do Mundo de 2012 e a automação dos testes de motores para aeronaves utilizados pelo PAMA (Parque de Material Aeronáutico). Mas além destes avanços, sistemas de apoio para controles de tarefas diárias da OM também estão evoluindo, utilizando-se mais meios eletrônicos do que as antigas pastas e arquivos.

Estes sistemas são hoje utilizados para o controle de material de uma base militar, o apoio didático para os alunos nas instituições de ensino militar, controle de férias, hospedagem nos hotéis de trânsito, entre outros. Alguns destes sistemas foram desenvolvidos

pelos próprios militares que atuam na área em que foi observado o problema, às vezes com um sucesso tão grande, que o sistema obteve maior reconhecimento e expansão em sua utilidade. Um exemplo claro dessa situação é o sistema MENTOR-SAGEM, foi implementado inicialmente na AFA, surgindo com o intuito de criar um controle das fichas de voo presentes na OM. Antes todo o gerenciamento das fichas de voos era feito fisicamente, com o uso de pastas e papéis. Os quadros dos aviadores eram conferidos por meio de plaquetas fixadas em quadro visível a todos. Caso um exercício de voo fosse interrompido e continuasse somente em outro momento era necessário procurar nos arquivos os papéis referentes àquela missão, conferindo individualmente cada caso.

Hoje, graças a um sistema de banco de dados, todo este gerenciamento é feito por meio de um sistema, onde cada militar consegue ver suas missões, fichas de voo e demais dados. Com isto, o tempo gasto procurando fichas desapareceu, ajudando o bom funcionamento da seção, e também o gerenciamento individual de cada piloto, visto que atualmente é possível conferir seu desempenho pessoal a qualquer momento, possibilitando também a identificação e correção de erros, vícios e demais problemas. O sucesso deste sistema foi tão grande que hoje ele é usado em diversas bases pelo Brasil.

Porém, apesar deste fluxo de modernização, o mesmo não ocorre com os controles de punição e disciplina para militares. Por definição, punição é algo que deve ser evitado pois é um castigo, uma recriminação feita por alguém sobre algum ato ou omissão cometida.

Entretanto, segundo o artigo 6 do RDAER (Regulamento de Disciplina da Força Aérea), uma punição dentro da força aérea visa o benefício do militar punido, pela sua reeducação ou para a organização a que pertence, pelo fortalecimento da disciplina e da justiça.

Infelizmente, este sistema atualmente é antiquado para o nível de gerenciamento necessário dentro de uma base militar. O sistema passa pelo mesmo problema citado anteriormente com controle de fichas de voo, ou seja, é dependente de papéis, arquivos e pastas. Mesmo que organizados de forma alfabética e pastas individuais, uma consulta nestes arquivos ainda leva tempo, além de ocupar um espaço físico muito grande da seção destinada a esta tarefa. Uma consulta, por exemplo, das principais ocorrências do último mês é extremamente trabalhosa, se não impossível, pois é necessário observar cada ficha individualmente. Além destas dificuldades, ainda é possível observar que, por ter uma gerência manual, erros como a não retirada dos arquivos de uma pasta pertencente a um militar que foi transferido ou mesmo o extravio desta podem vir a acontecer.

Este artigo tem como objetivo propor a utilização de um sistema de bancos de dados

simples, criado especificamente com o objetivo de armazenar e gerir as FATD's (Fichas de Avaliação de Transgressões Disciplinares) dos militares de uma base, de forma a ter um maior controle dos problemas e dificuldades enfrentados no dia a dia pelos militares lá sediados. Os resultados dos testes mostram um ganho significativo no tempo de pesquisa necessário no sistema, além de um controle mais apurado dos resultados. Com este controle é possível, por exemplo, a implementação de programas focados na resolução destes problemas, ou a criação de medidas restritivas para isso.

O artigo se organiza da seguinte forma: a Seção 2 descreve a história dos bancos de dados, sua evolução e influência no projeto. A Seção 3 apresenta os Materiais e Métodos, os softwares usados no desenvolvimento do projeto, destacando o método de desenvolvimento em que o protótipo foi construído. A Seção 4 discute a usabilidade do sistema. Na Seção 5, são apresentados os Resultados e Discussões deste trabalho. Por fim, na Seção 6, são apresentadas as conclusões obtidas.

2. Banco de Dados

A necessidade de administrar registros é decorrente de vários anos na história humana, de forma que quem consultar essas informações poderá avaliar seu histórico e adquirir conhecimento embasado, podendo encontrar possíveis soluções ou caminhos dentro de determinados problemas antes não observadas.

Nesse contexto a palavra registro, no dicionário significa, “Tomar nota, assentar o dito (por escrito ou na memória) para não olvidá-lo ou para servir de argumento quando necessário for.”, confirmando assim a importância das informações armazenadas e gerenciadas de forma eficaz.

Temos exemplos dessa afirmação no funcionamento do armazenamento de dados no Egito Antigo, onde eram registrados instruções, processos, dentre outras informações nos papiros, sendo eles enrolados e guardados em estojos especiais, facilitando assim sua organização, e conseqüentemente com boa organização se faz possível uma fácil consulta a essas informações.

Mas mesmo com sucesso nesse processo, surgiu a necessidade de melhorar, no sentido da flexibilidade dos dispositivos de armazenamento de registros, foi onde Pérgamo, aproximadamente 165 anos a.C. deu início a utilização dos pergaminhos, que eram feitos de pele de cabra ou carneiro, para registrar informações. Os pergaminhos, possuem mais liberdade em seu manuseio do que os papiros, além de mais flexibilidade e fáceis alterações

em

sua

forma.

Já com o crescimento populacional de alfabetizados aumentando, surgiu a necessidade de gerar cópias desses registros. Criar cópias à mão em grande escala, é uma tarefa árdua e exaustiva, ou seja, inviável levando em consideração a grande quantidade de leitores existentes. Foi onde surgiu as primeiras copiadoras, as copiadoras de argila, que tinha como função economizar tempo e facilitar o processo de gerar cópias.

Mais adiante, no Séc.XV houve a apresentação da tipografia, que veio para substituir as técnicas com argila, nesse método utilizava-se caracteres separados, sendo eles de metal, onde se mostrava flexibilidade no manuseio, além de ser capaz de produzir cópias em grande escala. A prova disso, foi um marco importante na história, as cópias da "Bíblia de 42 linhas".

No quesito arquivístico desse estudo, podemos ressaltar algumas passagens na história da civilização. Em Roma, na Grécia ou no Egito na antiguidade, já havia uma procedência de arquivamento, os chamados "fundos arquivísticos", que inclusive se respeita até a atualidade. Na Idade Moderna, a administração de documentos se tornou-se exclusivista, sendo acessível essas essas informações apenas ao governo, tendo fim nesse regime entre o final do Séc. XVIII e início do Séc. XX, onde houve a abertura da administração para o público .

Na atualidade em organizações de grande porte principalmente, a quantidade de dados armazenados é grande, e continua crescendo com o decorrer do tempo, e empresa IBM não é diferente. Nas Décadas de 60 e 70, a IBM se deparou com um grande problema, o gasto que estava tendo para contratar pessoas que fizessem o armazenamento, organização e consulta de seus registros. Tendo em visto essa necessidade vários pesquisadores estudaram soluções para resolvê-la, foi quando Ted Codd (um dos pesquisadores da IBM), publicou um artigo sobre bancos de dados relacionais, usando álgebra relacional, permitindo que usuários sem muito conhecimento técnico pudessem buscar informações através de comandos em inglês, mas devido a complicação matemática e natureza técnica do artigo, o projeto não foi executado de imediato, mas a IBM mesmo assim montou uma equipe para estudo e execução do projeto, denominado System R.

Em 1976 Honeywell Information Systems Inc., concebeu o primeiro sistema de banco de dados comercial, embasado nos estudos de Ted Codd, e logo depois surgiu também para concorrer, o modelo da Oracle.

Nos anos 80 finalmente apareceu o modelo da IBM através do SQL/DS, evoluindo o System R para DB2(banco de dados desenvolvida pela IBM), dando assim início a linguagem SQL (Structured Query Language).

Já na Déc. de 90 houve o surgimento de outros bancos, por exemplos, SQL Server,

MySQL, Paradox, entre outros. Entre 1989 e 1990, também surgiu o conceito de banco de dados orientado a objeto, para suprir algumas necessidades existentes em algumas áreas científicas.

Com o sucesso da linguagem em 1986, o ANSI (Instituto Americano de Padrões) padronizou sua implementação, logo no ano seguinte o padrão foi adotado pela OSI (Organização Internacional de Padronização) (Lima, Allyn. Padrão SQL e sua Evolução). Desde então a linguagem SQL vêm evoluindo, por exemplo o SQL-92, a cada nova versão que surge se implementa padronizações. Dessa forma SQL é a linguagem para Banco de Dados com mais força no mercado.

Originalmente a linguagem SQL era chamada de SEQUEL. E como antes visto, foi concebida com o intuito de manipular informações dentro do modelo relacional em um banco de dados. SQL é uma linguagem declarativa, ou seja, seu funcionamento depende de descrever o problema a ser resolvido, e não uma solução, sendo o contrário de algumas linguagens, por exemplo, a linguagem C.

Atualmente a linguagem SQL possui várias versões e também várias variações (ou como também são chamadas, extensões), onde cada uma tem suas particularidades, isso depende do SGDB (Sistema de Gerenciamento de Banco de Dados), alguns exemplo são, MySQL, Microsoft SQL Server, PostgreSQL, entre muitos outros.

2 Materiais e Métodos

O software livre para o gerenciamento de banco de dados LibreOffice Base¹ foi inicialmente escolhido para o primeiro protótipo do projeto. Este software pode ser utilizado tanto por usuários iniciantes quanto avançados, permitindo suporte nativo para alguns dos motores de bancos de dados mais utilizados atualmente, tais como Adabas D², MS Access³, Firebird⁴ e PostgreSQL⁵. Ele também tem suporte embutido para os mecanismos JDBC (*Java DataBase Connectivity*) e ODBC (*Open Database Connectivity*), o que permite uma ampla gama de conexões com outros mecanismos de bancos de dados.

O sistema foi desenvolvido no software Workbench MySQL⁶. Este software foi escolhido pela sua ampla versatilidade e ser compatível com os plugins necessários para o

¹ <https://pt-br.libreoffice.org/descubra/base/>

² <http://www.adabas.com/>

³ <https://products.office.com/pt-br/access>

⁴ <https://firebirdsql.org/>

⁵ <https://www.postgresql.org/>

⁶ <https://www.mysql.com/products/workbench/>

desenvolvimento do protótipo. A capacidade de armazenamento transacional disponível também foi levada em conta, visto que até mais de um usuário pode utilizar o sistema simultaneamente. A facilidade com que é possível manipular instruções SQL (*Structured Query Language*) foi essencial para o projeto, pois essa característica permite a inclusão das instruções em diversos sistemas. Escrito em C e C++ o Workbench tem suporte para APIs (*Application Programming Interface*) das linguagens de programação C, C++, Eiffel, Java, Perl, PHP, Python, Ruby e Tcl.

O sistema foi desenvolvido usando a metodologia de desenvolvimento de software Cascata, conforme apresentado na Figura 1.



Etapas de desenvolvimento. Figura 1.

Desenvolvido em 5 etapas, onde primeiramente ocorreu uma entrevista com os responsáveis pelo controle das FATD's da AFA, que foi exposto a ideia do projeto e discutido as possibilidades que dele derivariam. Atualmente, a organização das fichas é feita dentro de armários, as quais estão divididas por diferentes seções (setores dentro da OM). Tal organização torna o levantamento de dados de um militar específico demorado, sendo necessário às vezes dias para o levantamento dos dados, visto que ao longo do tempo o militar

muda de seção e as fichas não o acompanham. Assim, o entrevistado sugeriu a inclusão de um “resumo” de todas as punições que o militar já sofreu, permitindo ao militar avaliador dar uma punição coerente com histórico do acusado.

Na etapa 2, o desenvolvimento do projeto teve início com a criação das tabelas. Por ser um sistema de armazenamento pensado especificamente para FATD’s, uma única entidade (objeto que possui características) foi gerada, chamada de Ficha de Punição. A entidade é caracterizada por diferentes atributos, os quais representam os campos que compõem a ficha em papel. Posteriormente à modelagem da estrutura do banco de dados, isto é, criação das tabelas, foram selecionados os *softwares* a serem usados no desenvolvimento do protótipo.

A etapa 3 teve como objetivo a criação da interface inicial, elaborada para ser usada com o software LibreOffice Base. A interface foi baseado na FATD existente, tentando incorporar em uma única tela todo seu conteúdo, de forma que se assemelhasse mais com a ficha física.

Com a interface concluída teve início a quarta etapa do projeto, a qual corresponde à primeira sessão de testes. Nesta etapa, os testes foram feitos pelos próprios autores deste trabalho.

The image shows a screenshot of a web-based form interface for recording disciplinary actions (FATD). The interface is set against a dark blue background. At the top, there are two input fields for 'MILITAR ARROLADO' and 'AUTORIDADE QUE APURA A TRANSGRESSÃO DISCIPLINAR', both with a placeholder text 'Posto\Graduação\Nome Completo\SARAM\OM\Sec'. Below these are two more input fields for 'TESTEMUNHAS (Se Necessário)'. To the right of the top section, there are four checkboxes with corresponding labels: 'Devolveu sem o FATD devidamente preenchido.', 'Recusou-se a assinar a FATD.', 'Devolveu fora do prazo o FATD.', and 'Não devolveu o FATD.'. Below the input fields, there are four large, empty rectangular boxes arranged in a 2x2 grid. The top-left box is labeled 'RELATO DO FATO', the top-right 'DECISÃO DA AUTORIDADE QUE APLICA A PUNIÇÃO DISCIPLINAR', the bottom-left 'JUSTIFICATIVAS / ALEGAÇÕES DE DEFESA', and the bottom-right 'SOLUÇÃO DA AUTORIDADE QUE APURA A TRANSGRESSÃO DISCIPLINAR'. A small navigation icon is visible in the top right corner.

Interface inicial no LibreOffice Base. Figura 2.

Na etapa 5, os problemas identificados durante a fase de teste foram analisados e a melhor solução encontrada implementada. Nos testes, defeitos foram encontrados em alguns campos da tabela do banco de dados e, conseqüentemente, na interface. Além disso,

observou-se uma carência de recursos disponíveis no software selecionado inicialmente para o desenvolvimento (i.e., LibreOffice Base), sendo necessária a migração desse sistema para o MySQL WorkBench. Para tanto, os autores elaboraram um código para atender aos requisitos identificados inicialmente neste trabalho. Na tabela FichaPunicao, foi criada uma chave primária, denominada ID, visando identificar unicamente cada registro da tabela e, assim, controlando as informações inseridas. Em outras palavras, esse controle permite que cada ficha seja única no banco, impedindo repetições nos registros. Para atender a necessidade do entrevistado, o campo “saram”, que faz referência ao número SARAM (Subdiretoria de Aplicação dos Recursos da Assistência Médico-Hospitalar), é utilizado como uma identificação única do militar, tendo em vista que todo militar deve possuir esse número para manter-se em acompanhamento médico por todo o período que permanecer militar da ativa ou reserva. O uso do saram possibilita, por exemplo, que o algoritmo percorra todos os registros e faça uma somatória de todos os campos iguais, possibilitando a contagem e visualização dos campos “grau_incriminatorio” e “tipo_punicao”. O “grau_incriminatorio” indica o tipo acusação que o militar arrolado sofreu. Por exemplo, caso o militar tenha o uniforme em desalinho, ele recebe o grau incriminatório de “transgressão”, pois fere o art.[...], mas caso ele tenha cometido deserção, ele já recebe o grau de “crime militar”, e assim por diante. O campo “tipo_punicao” define qual foi o tipo de punição que o militar acusado sofreu. Por exemplo, “delta” (Detenção), “papa” (Prisão), “repreensão verbal”, “repreensão por escrito”, entre outros.

O problema na tabela foi encontrado em dois dos atributos (justificativa e descricao_dos_fatos), visto que o tamanho reservado para o armazenamento dos dados mostrou-se menor do que necessário para o preenchimento dos campos. Porém, ao aumentar o tamanho desse campo, tornou-se inviável manter os campos no espaço de tela disponível. A resolução deste problema deu origem a mais uma etapa de desenvolvimento.

A ideia da criação de uma interface para o primeiro protótipo foi abandonada depois da última etapa de testes. Foi observado que, mesmo através de instruções SQL, o sistema foi capaz de alcançar os objetivos propostos inicialmente, relegando a interface para o campo de projetos futuros.

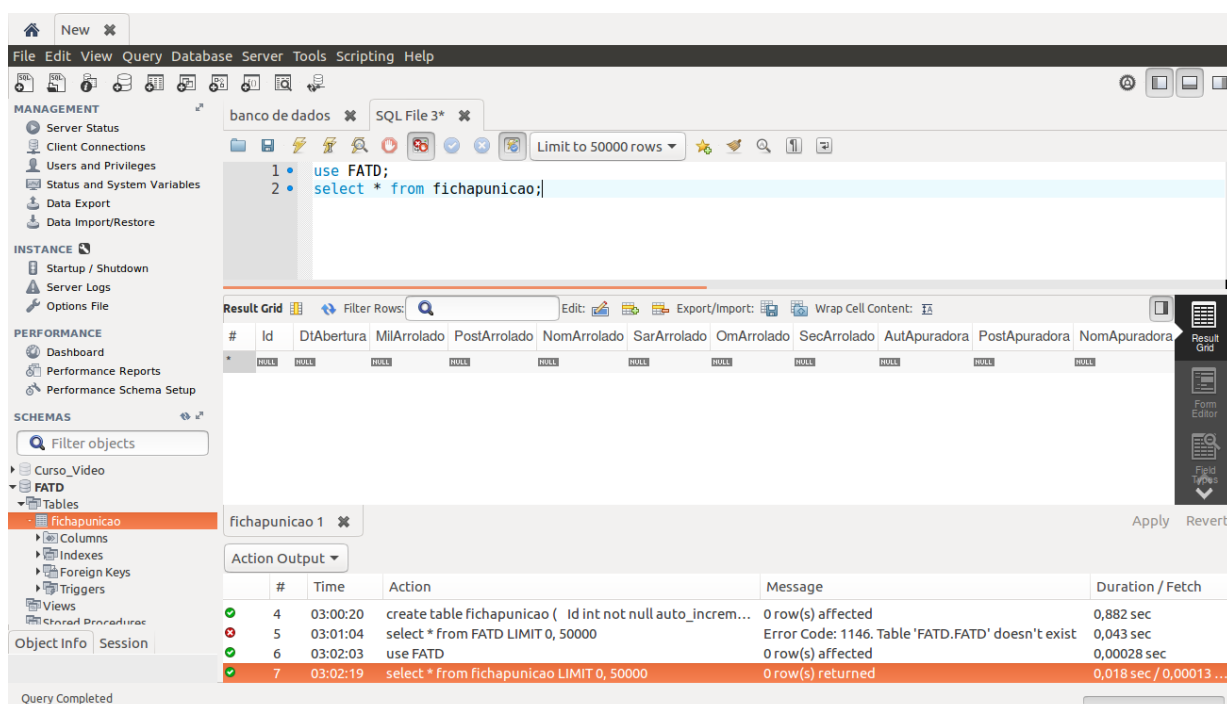
3 Métodos de uso

Todo o gerenciamento das FATDs é realizado por meio de instruções SQL, digitadas no próprio SGBD MySQL Workbench ou em outro SGDB da preferência do usuário, com as

devidas exclusividades de cada plataforma. Dessa forma, possibilita-se a versatilidade entre os softwares em relação ao sistema.

Os comandos em SQL devem ser escritos na aba QUERY de forma a consultar os registros buscando responder o problema em questão. Por exemplo, se o usuário desejar visualizar todos os registros do campo “x” da tabela “y”, ele irá digitar a instrução SQL e o SGBD retornará a “resposta” da requisição, ou seja, ele mostrará todos os registros do campo “x” da tabela “y” do banco de dados. Caso o comando seja inválido ou se a sintaxe do comando estiver incorreta, o SGBD retornará um erro.

A linguagem SQL é dividida em grupos que indicam qual a funcionalidade de cada comando. Um exemplo seria o comando SELECT, que pertence ao grupo DQL (*Data Query Language*), sua função é selecionar valores de determinados campos e retorná-los para o usuário requerente. De forma geral, o usuário utiliza os grupos DQL e DML (*Data Manipulation Language*), englobam as funções de alterar dados do sistema e consulta-los.



Interface do MySQL Workbench. Figura 3.

4 Resultado de Análise

Graças ao sistema uma pesquisa que anteriormente demorava dias pode ser feita em alguns segundos. Por meio do histórico de punições, é possível realizar uma avaliação mais direta e rápida do militar arrolado, permitindo uma agilização dos trâmites burocráticos. O espaço de armazenamento necessário para as FATD's não foi alterado, pois por questões legislativas ainda é necessário registrar de maneira física as punições ocorridas. Porém isso

pode ser alterado baseando-se no processo que ocorre com as FOBS (Ficha de Observação) dos cadetes da Força Aérea, onde todo armazenamento é digital.

Analisando as principais ocorrências, é possível ainda propor um programa de “intervenção” em determinado setor da OM, onde um problema se repete com frequência. Por exemplo, um programa explicando os males do alcoolismo em algum setor onde exista uma grande ocorrência de militares chegando embriagados para o serviço.

Uma das vantagens desse sistema é a possibilidade de uma apuração mais eficiente de um determinado setor ou grupo de militares que apresenta com frequência o mesmo relato de transgressão. Por exemplo, os militares de um determinado setor (manutenção) que têm um histórico de desalinhamento de uniforme (uso de cobertura não prevista), sugere a possibilidade de que o problema em questão tenha a ver com o setor ou o serviço prestado (corte de grama) e não com um militar específico.

5 Conclusões

O protótipo final do projeto visa ser utilizado para o armazenamento e o controle das FATD's, consulta de punições para os militares da OM. Além disso, o sistema também permite a solução de problemas específicos, como uma análise mais detalhada dos registros. O banco de dados pode ser expandido para aceitar outras atribuições, tais como o controle de todas as alterações dos militares, ou integrando-o diretamente com outros sistemas de forma a permitir que o militar seja ou não convocado para determinadas missões, receba ou não uma promoção, entre outros.

Com o uso de IA (Inteligência Artificial), o reconhecimento de padrões, que atualmente é feito por usuários, pode ser realizado usando algoritmos como o KNN (*K-Nearest Neighbors*), para identificar os problemas que mais se repetem, ou algoritmos de Machine Learning, ou até mesmo novas soluções.

O plugin JDBC pode ser usado para a integração do banco de dados com diferentes sistemas. Esta API permite a portabilidade e a legibilidade para o sistema, sendo uma característica necessária para atingir o maior número de usuários. O JDBC⁷ tem um amplo uso corporativo e *Web*, com uma segurança que permite a armazenagem de dados a longo prazo de forma consistente. Além disso, o editor de imagens de código aberto GIMP⁸ (GNU Image

⁷ <http://www.oracle.com/technetwork/database/features/jdbc/index-091264.html>

⁸ <https://www.gimp.org/>

Manipulation Program) pode ser usado no desenvolvimento das imagens dos botões e ícones do projeto. Disponível para os sistemas operacionais Windows, GNU/Linux, OS X, entre outros. Este programa gratuito conta com uma grande base de apoiadores, permitindo uma aprendizagem rápida, graças aos fóruns e tutoriais disponíveis.

A aplicação do projeto desenvolvido pode garantir uma economia de tempo e recursos para a força aérea brasileira, cada vez mais relevante no quadro nacional. Os benefícios associados aos militares envolvidos no processo confirmam a premissa do regulamento disciplinar da aeronáutica: as punições visam o melhoramento do militar da instituição e a boa manutenção da justiça.

6 Bibliografia

BRASIL. Decreto nº 76322, de 22 de setembro de 1975 . Aprova o Regulamento Militar da Aeronáutica (RDAER) . Diário Oficial, Brasília, DF, 22 set. 1975, 154º da Independência de 87º da República..

Militares da FAB criam processo de automação do banco de provas de motores do PAMA-SP. Força Aérea Brasileira. Disponível em:
<<http://www.fab.mil.br/noticias/mostra/30512/NOTAER%20-%20Militares%20da%20FAB%20criam%20processo%20de%20automa%C3%A7%C3%A3o%20do%20banco%20de%20provas%20de%20motores%20do%20PAMA-SP>>. Acesso em: 04 mai. 2018.

Hermes 900 reforça a capacidade da FAB no reconhecimento eletrônico. Força Aérea Brasileira. Disponível em:
<<http://www.fab.mil.br/noticias/mostra/18093/REAPARELHAMENTO-%E2%80%93-Hermes-900-refor%C3%A7a-capacidade-operacional-da-FAB-no-reconhecimento-eletr%C3%B4nico>>. Acesso em: 04 mai. 2018.

Significado de Punição. Dicio. Disponível em: <<https://www.dicio.com.br/punicao/>>. Acesso em: 04 mai. 2018.

Significado de Registro. Dicionário do Aurélio. Disponível em:

<<https://dicionariodoaurelio.com/registro>>. Acesso em: 04 mai. 2018.

FRAGOSO, Graça Maria; DUARTE, Rogério. Livro, leitura, biblioteca ... uma história sem fim. Revista ACB: Biblioteconomia em Santa Catarina, Florianópolis, v. 9, p. 01-02, 2004.

CALDERON, Wilmara Rodrigues; CORNELSEN, Julce Mary; PAVEZI, Neiva; LOPES, Maria Aparecida. O processo de gestão documental e da informação arquivística no ambiente universitário. Ciência da Informação, Brasília, v. 33, n. 3, p. 01, set./dez. 2004.

COSTA, Adriana. Ferramenta de Gestão de Dados Históricos. 2013. Dissertação de mestrado integrado em Engenharia de Comunicações - Universidade do Minho, Escola de Engenharia, Braga/Guimarães, Portugal.

GALASSI, Carla; FÓRMICE, César; Dionysio, Rosana. Modelagem de Banco de Dados. Taquaritinga: Clube de Autores, 2009.

Liguagem SQL. .org. Disponível em: <<http://principo.org/linguagem-sql-nice-pereira-tuani-zanatta.html>>. Acesso em: 04 mai. 2018.

LIMA, Allyn Grey de Almeida. Padrão SQL e sua evolução. Disponível em: <<http://www.ic.unicamp.br/~geovane/mo410-091/Ch05-PadraoSQL-art.pdf>>. Acesso em: 04 mai. 2018.

TOP 10 principais SGDBs do mercado global!. Bencode. Disponível em: <<https://bencode.com.br/principais-sgbd/>>. Acesso em: 04 mai. 2018.