

DANIELA DE SOUZA MACHADO

**MANUTENÇÃO E DOCUMENTAÇÃO DO PORTAL CORPORATIVO
DA 6ª REGIÃO DA PMMG**

Monografia apresentada ao Departamento de
Ciência da Computação da Universidade Federal
de Lavras como parte das exigências do curso de
Ciência da Computação para a obtenção do título
de Bacharel.

Orientador:

Prof. Heitor Augustus Xavier Costa

Co-Orientadores:

Prof. André Luiz Zambalde

Cap. Antônio Claret dos Santos

LAVRAS
MINAS GERAIS - BRASIL
2004

DANIELA DE SOUZA MACHADO

**MANUTENÇÃO E DOCUMENTAÇÃO DO PORTAL CORPORATIVO
DA 6ª REGIÃO DA PMMG**

Monografia apresentada ao Departamento de
Ciência da Computação da Universidade
Federal de Lavras como parte das exigências
do curso de Ciência da Computação para a
obtenção do título de Bacharel

APROVADA em 17 de Janeiro de 2005

Prof. André Luiz Zambalde

Cap. Antônio Claret dos Santos

Prof. Heitor Augustus Xavier Costa
(Orientador)

LAVRAS
MINAS GERAIS - BRASIL
2004

A Deus por sempre me fortalecer e me iluminar, aos meus queridos pais Laerte e Tereza, por terem preparado o terreno para o meu próprio caminho até a felicidade na vida, aos meus irmãos Wagner e Rafael, meus grandes companheiros e à uma pessoa muito especial Leonardo, por sempre estar ao meu lado.

Agradecimentos

A todos que me forneceram inestimável conhecimento além de apoio pessoal durante este percurso. Em especial, ao meu orientador, Prof. Heitor Augustus Xavier Costa, por seu auxílio e contribuição para mais esta etapa cumprida. Aos meus co-orientadores Cap. Antônio Claret dos Santos e Prof. André Luiz Zambalde por terem me ajudado neste último passo para minha formação.

Ao pessoal do Núcleo de Desenvolvimento de Projetos da 6ª Região da Polícia Militar que possibilitou um ambiente agradável para a realização deste trabalho.

Para concluir, minha profunda gratidão à minha família e aos muitos amigos que enriqueceram minha vida mais do que eu poderia expressar: Fernanda, Flávia, Adriana, meus colegas de classe e aos outros amigos que posso ter deixado de mencionar aqui pelo nome, mas que sempre trago no meu coração com respeito, gratidão e amor.

SUMÁRIO

| | |
|--|------------|
| LISTA DE FIGURAS..... | i |
| LISTA DE QUADROS..... | iii |
| RESUMO..... | iv |
| 1. INTRODUÇÃO..... | 1 |
| 1.1 MOTIVAÇÃO..... | 3 |
| 1.2 OBJETIVOS | 4 |
| 1.3 METODOLOGIA DE DESENVOLVIMENTO..... | 4 |
| 1.3.1 <i>Tecnologias Utilizadas</i> | 5 |
| 1.3.2 <i>Ambiente de Desenvolvimento</i> | 6 |
| 1.4 DESCRIÇÃO DOS CAPÍTULOS..... | 6 |
| 2. MANUTENÇÃO DE PRODUTOS DE SOFTWARE..... | 8 |
| 2.1 TIPOS DE MANUTENÇÃO DE SOFTWARE | 8 |
| 2.2 DIFICULDADES DA MANUTENÇÃO | 10 |
| 2.3 TECNOLOGIA DE SUPORTE À MANUTENÇÃO | 12 |
| 2.4 MODELOS DE PROCESSOS DE MANUTENÇÃO | 16 |
| 2.5 CARACTERÍSTICAS DA MODELAGEM DO PROCESSO DE MANUTENÇÃO.. | 19 |
| 2.6 RESUMO | 20 |
| 3. ENGENHARIA REVERSA..... | 22 |
| 3.1 CONCEITOS DE ENGENHARIA REVERSA..... | 23 |
| 3.2 VISÕES DE PRODUTOS DE SOFTWARE..... | 24 |
| 3.3 CATEGORIAS | 26 |
| 3.3.1 <i>Visualização de Código</i> | 26 |
| 3.3.2 <i>Entendimento de Produto de Software</i> | 27 |
| 3.4 RESUMO | 28 |
| 4. REENGENHARIA | 30 |
| 4.1 DEFINIÇÕES..... | 30 |
| 4.2 CATEGORIAS | 32 |
| 4.3 POR QUE REALIZAR A REENGENHARIA..... | 33 |
| 4.4 COMO REALIZAR A REENGENHARIA..... | 34 |
| 4.5 O PROCESSO DE REENGENHARIA..... | 35 |
| 4.6 FERRAMENTAS DE AUXÍLIO À REENGENHARIA..... | 37 |
| 4.7 RESUMO | 40 |

| | |
|--|-----------|
| 5. UML-BASED WEB ENGINEERING | 42 |
| 5.1 O QUE É UWE | 42 |
| 5.2 VISÃO GERAL..... | 43 |
| 5.3 RESUMO | 48 |
| 6. IMPLANTAÇÃO E DOCUMENTAÇÃO DO NOVO PORTAL CORPORATIVO DA 6ª RPM..... | 49 |
| 6.1 ARQUITETURA..... | 49 |
| 6.2 ESTRUTURA DO PRODUTO DE SOFTWARE..... | 51 |
| 6.3 DESIGN..... | 54 |
| 6.4 NOVA FUNCIONALIDADE | 57 |
| 6.5 MODELAGEM DO PRODUTO DE SOFTWARE | 61 |
| 6.5.1 <i>Modelo de Casos de Uso</i> | 61 |
| 6.5.2 <i>Modelo Conceitual</i> | 74 |
| 6.5.3 <i>Modelo de Navegação</i> | 76 |
| 6.5.4 <i>Modelo de Apresentação</i> | 80 |
| 6.6 RESUMO | 82 |
| 7. CONCLUSÃO E TRABALHOS FUTUROS | 83 |
| 7.1 CONCLUSÃO | 83 |
| 7.2 CONTRIBUIÇÕES..... | 84 |
| 7.3 TRABALHOS FUTUROS | 85 |
| REFERÊNCIAS BIBLIOGRÁFICAS..... | 87 |
| APÊNDICE A..... | 92 |

LISTA DE FIGURAS

| | |
|---|----|
| Figura 2.1: Relacionamentos no Ciclo de Desenvolvimento de Software..... | 14 |
| Figura 3.1: Visualizações de Software no Ciclo de Desenvolvimento..... | 25 |
| Figura 3.2: Categorias da engenharia reversa e suas visões..... | 28 |
| Figura 6.1: Página inicial do Portal Corporativo da 6ª RPM (seção Internet)..... | 52 |
| Figura 6.2: Página da Imprensa do Portal Corporativo da 6ª RPM..... | 53 |
| Figura 6.3: Página da Intranet do Portal Corporativo da 6ª RPM..... | 53 |
| Figura 6.4: Página da Administração do Portal Corporativo da 6ª RPM..... | 54 |
| Figura 6.5: Página para realizar cadastro/manutenção de Ocorrências..... | 56 |
| Figura 6.6: Diagrama dos subsistemas do Portal Corporativo da 6ª RPM..... | 62 |
| Figura 6.7: Diagrama de caso de uso do subsistema Grupos..... | 63 |
| Figura 6.8: Diagrama de caso de uso do subsistema Usuários..... | 63 |
| Figura 6.9: Diagrama de caso de uso do subsistema Usuários da Imprensa..... | 64 |
| Figura 6.10: Diagrama de caso de uso do subsistema BPIs..... | 64 |
| Figura 6.11: Diagrama de caso de uso do subsistema Notícias..... | 65 |
| Figura 6.12: Diagrama de caso de uso do subsistema Ocorrências..... | 65 |
| Figura 6.13: Diagrama de caso de uso do subsistema Desaparecidos..... | 66 |
| Figura 6.14: Diagrama de caso de uso do subsistema Procurados..... | 66 |
| Figura 6.15: Diagrama de caso de uso do subsistema <i>Links</i> | 66 |
| Figura 6.16: Diagrama de caso de uso do subsistema Enquetes..... | 67 |
| Figura 6.17: Diagrama de caso de uso do subsistema Fórum de Discussão..... | 67 |
| Figura 6.18: Diagrama de caso de uso do subsistema Mural de Recados..... | 68 |
| Figura 6.19: Diagrama de caso de uso do subsistema População..... | 68 |
| Figura 6.20: Diagrama de caso de uso do subsistema Ficha de Município..... | 69 |
| Figura 6.21: Diagrama de caso de uso do subsistema <i>Downloads</i> | 70 |
| Figura 6.22: Diagrama de caso de uso do subsistema Cidades..... | 71 |
| Figura 6.23: Diagrama de caso de uso do subsistema Pelotões PM..... | 71 |
| Figura 6.24: Diagrama de caso de uso do subsistema Companhias PM..... | 72 |
| Figura 6.25: Diagrama de caso de uso do subsistema Unidades Operacionais..... | 72 |
| Figura 6.26: Diagrama de caso de uso do subsistema Crimes..... | 73 |
| Figura 6.27: Diagrama de caso de uso do subsistema Efetivo..... | 73 |
| Figura 6.28: Diagrama de caso de uso do subsistema Treinamento..... | 74 |
| Figura 6.29: Modelo Conceitual do subsistema Unidades da Polícia Militar..... | 75 |
| Figura 6.30: Espaço do Modelo de Navegação do subsistema Unidades da Polícia Militar (Cadastro de Unidades Operacionais)..... | 77 |

| | |
|---|----|
| Figura 6.31: Estrutura do Modelo de Navegação com acessos primitivos do subsistema Unidades da Polícia Militar (Cadastro de Unidades Operacionais)..... | 79 |
| Figura 6.32: Modelo de Apresentação do subsistema Unidades da Polícia Militar (Cadastros de Unidades Operacionais)..... | 81 |

LISTA DE QUADROS

| | |
|---|----|
| Quadro 4.1: Escopo das informações utilizadas por ferramentas de reengenharia, as respectivas visões e outras saídas produzidas..... | 38 |
| Quadro 6.1: Representação da Hierarquia da 6ª RPM..... | 59 |

RESUMO

Título: Manutenção e Documentação do Portal Corporativo da 6ª Região da PMMG

A necessidade de realizar a manutenção de um produto de software é motivada principalmente por alguma insatisfação dos usuários com a sua utilização e, conseqüentemente, pela falha em sua usabilidade. Porém, o processo de manutenção muitas vezes torna-se complexo, quando a sua modelagem e a sua documentação não existem ou foram mal elaboradas. Assim, neste trabalho foi realizada a manutenção e documentação do Portal Corporativo da 6ª Região da Polícia Militar de Minas Gerais onde foram aplicados os conceitos de engenharia reversa de software e reengenharia de software e a modelagem UWE (*UML-based Web Engineering*).

Palavras-Chave: Portal Corporativo, Manutenção de Produtos de Software, Engenharia Reversa, Reengenharia, Modelagem UWE e Usabilidade.

ABSTRACT

Title: Maintenance and Documentation of the Corporative Portal of 6ª Region of the PMMG

The necessity of accomplish the maintenance of software product is motivated mainly by some dissatisfaction of the users with its use and, consequently, by the imperfection in its usability. However, the maintenance process many times becomes complex, when the modeling and documentation of this software do not exist or badly had been elaborated. Thus, in this work were accomplished the maintenance and documentation of the Corporative Portal of 6ª Region of Military Police of Minas Gerais where reverse engineering and software reengineering concepts and the UWE (*UML-based Web Engineering*) modeling had been applied.

Keywords: Corporative Portal, Software Products Maintenance, Reverse Engineering, Reengineering, UWE Modeling and Usability.

1. INTRODUÇÃO

Segundo [OSBORNE & CHIKOFSKY, 1990], a variedade de problemas que envolvem a manutenção de produtos software cresce constantemente, sendo que as soluções não acompanham essa evolução. Esses problemas são resultantes de código-fonte e documentação mal elaborados, além da falta de compreensão do produto de software.

A partir do momento em que um produto de software começa a ser utilizado, ele entra em um estado contínuo de mudança. Mesmo que tenha sido construído aplicando as melhores técnicas de projeto e codificação existentes, os produtos de software vão se tornando obsoletos em vista das novas tecnologias que são disponibilizadas.

Além das correções de erros (manutenção corretiva), as mudanças mais comuns que os produtos de software sofrem são migrações para novas plataformas, ajustes para mudanças de tecnologia de hardware ou sistema operacional (manutenção adaptativa) e extensões em sua funcionalidade para atender os usuários (manutenção perfectiva).

Em geral, essas mudanças são realizadas sem que haja preocupação com a arquitetura geral do produto de software, produzindo estruturas mal projetadas, documentação desatualizada e lógica e codificação ruins, sendo esses os focos que dificultam a sua manutenção [OSBORNE & CHIKOFSKY, 1990].

Quando o produto de software não é fácil de ser mantido, sendo, porém, de grande utilidade, ele deve ser reconstruído. Partindo-se do produto de software existente (via código-fonte, interface ou ambiente), é abstraída sua funcionalidade e são construídos o seu modelo de análise e o seu modelo de projeto. Esse processo é denominado engenharia reversa.

[RUGABER, 1992] afirma que a maior parte do esforço do desenvolvimento de produtos de software é gasto na sua manutenção, e não no desenvolvimento de produtos de software novos, e que grande parte do processo de manutenção é dirigido ao seu entendimento. Sendo assim, para melhorar o desenvolvimento de produtos de software, é necessário facilitar o seu processo de compreensão. A engenharia reversa aborda diretamente o problema de compreender o produto de software.

Sob a ótica do exposto acima, foi constatada a necessidade de realizar a manutenção do Portal Corporativo da 6ª Região da Polícia Militar de Minas Gerais (6ª RPM). Esse produto de software, conforme citado em [LEMOS, 2003], tem possibilitado um melhor gerenciamento de dados e de informações ocasionando a integração de toda a organização a uma base única de conhecimento, acessível através de um *browser*. Assim, pode-se ter acesso a base de conhecimento quando e onde estiver, facilitando a transformação de conhecimento em meios de ação, aumentando a agilidade decisória. Porém, para que o Portal faça diferença dentro de uma organização, ele deve ser usado constantemente, senão será como construir um *playground* em um condomínio só de idosos.

Sendo assim, para que estas informações continuem satisfazendo as necessidades dos usuários, é essencial que algumas alterações e adições sejam realizadas na funcionalidade deste produto de software.

Desta forma, este trabalho surge com o intuito de realizar a modelagem, documentando o produto de software existente, além de adaptá-lo às novas necessidades dos usuários. Para isso, os conceitos de engenharia reversa (*backward engineering*) e reengenharia de software, bem como um subconjunto de artefatos de software do padrão de modelagem *UML based Web Engineering* (UWE), são utilizados.

1.1 Motivação

A necessidade da mudança de um produto de software é motivada, principalmente, pela insatisfação dos usuários com a sua utilização e, conseqüentemente, pela falha em sua usabilidade.

Baseado nas características de usabilidade do produto de software em questão, surgem fatores de motivação para o desenvolvimento deste trabalho, os quais se referem às atividades de manutenção e de documentação do sistema de software.

Em relação à manutenção, motivações surgem devido às falhas e às necessidades apontadas por usuários do Portal Corporativo da 6ª RPM. Estas falhas não têm possibilitado o uso de forma satisfatória do produto de software, uma vez que algumas informações essenciais dentro da organização não estão presentes no Portal, devido à falta de funções que permitem sua manipulação. Ou seja, visto que estas informações são dinâmicas, a funcionalidade do portal deve permitir este dinamismo.

Além disso, inconsistências e falta de padronização de terminologias utilizadas na implementação estão presentes no produto de software como um todo, abrangendo tanto a base de dados (banco de dados) quanto o código fonte. Isto pode causar efeitos prejudiciais ao bom funcionamento do sistema, bem como à sua integridade.

Ainda em relação à manutenção, existe a necessidade de maior dinamismo do conteúdo de algumas páginas presentes no portal, para que as informações possam ser melhor manipuladas por quaisquer usuários responsáveis por estas operações de manipulação. Pois, uma vez presente este dinamismo, não há necessidade de alterar o código-fonte da página, possibilitando até mesmo usuários leigos realizar operações como: inclusão, alteração e remoção de informações. Há também a necessidade de modificar a

interface com o usuário, para que características de usabilidade estejam presentes no produto de software.

Por fim, em relação à documentação do produto de software, um fator de motivação surge devido à falta de modelos que deveriam ter sido gerados antes da sua construção, o que facilitaria sua manutenção e, posteriormente, sua reutilização. Assim, com o crescimento contínuo do Portal da 6ª RPM, está cada dia mais complexo realizar a sua manutenção e a reutilização dos programas ora construídos.

1.2 Objetivos

O objetivo deste trabalho é realizar a manutenção, que tem por finalidade auxiliar os usuários na utilização do Portal Corporativo da 6ª RPM, facilitando o seu aprendizado, a obtenção e a disponibilidade de informações necessárias de forma adequada e satisfatória.

Além disso, tem o objetivo de prover o Portal com uma documentação eficiente, através da modelagem e da descrição das entidades e da sua funcionalidade, para que futuras modificações possam ser realizadas sem deparar-se com problemas como, a dificuldade de sua compreensão.

1.3 Metodologia de Desenvolvimento

O trabalho foi desenvolvido com base nos conceitos de reengenharia e engenharia reversa de software.

Como definido por [PRESSMAN, 2001], a reengenharia, também chamada de recuperação ou renovação, recupera informações de projeto de um produto de software existente e usa essas informações para alterá-lo ou reconstituí-lo, preservando as funções existentes, ao mesmo tempo em que adiciona novas funções ao produto de software, em um esforço para melhorar sua qualidade global. Como parte do processo de reengenharia, a engenharia reversa de software é um processo de recuperação de projeto, consistindo em analisar um programa, na tentativa de criar uma representação, em um nível de abstração mais alto que o código-fonte.

Através do emprego destes conceitos, foi possível realizar a manutenção e a documentação do produto de software.

Todas as informações necessárias para a realização deste trabalho foram obtidas através de consultas a internet, a biblioteca da Universidade Federal de Lavras e a revistas científicas. Além disso, diversas entrevistas com os usuários do Portal foram realizadas, sendo de grande importância e fonte de subsídio.

1.3.1 Tecnologias Utilizadas

Para a realização da manutenção, foram utilizadas as ferramentas HTML (*Hipertext Markup Language*) e ASP (*Active Server Pages*). Para armazenamento dos dados foi utilizado o SGBD (Sistema Gerenciador de Banco de Dados) *SQLServer* da Microsoft. Para o armazenamento das páginas do portal, foi utilizado o servidor *Microsoft Internet Information Service* (IIS).

A documentação foi feita utilizando o padrão *UML-based Web Engineering* (UWE), construindo um subconjunto de seus artefatos de software, e foi realizada com a utilização da ferramenta CASE (*Computer Aided Software Engineering*) *Rational Rose Enterprise Edition*TM 2000.

1.3.2 Ambiente de Desenvolvimento

O presente trabalho foi desenvolvido, em sua maior parte, no Núcleo de Desenvolvimento de Projetos da 6ª Região da Polícia Militar, que tem sede na cidade de Lavras/MG e abrange uma totalidade de 141 cidades e 9 distritos. A 6ª RPM é dividida em seis comandos de Unidades Operacionais, sendo estes: o 8º Batalhão de Polícia Militar (8º BPM) com sede em Lavras, o 20º Batalhão de Polícia Militar (20º BPM) com sede em Pouso Alegre, o 24º Batalhão de Polícia Militar (24º BPM) com sede em Varginha, o 29º Batalhão de Polícia Militar (29º BPM) com sede em Poços de Caldas, a 5ª Companhia de Polícia Militar Independente (5ª CIA PM IND) com sede em Itajubá e a 14ª Companhia de Polícia Militar Independente (14ª CIA PM IND) com sede em São Lourenço.

Parte do desenvolvimento foi realizada nos laboratórios de Informática do Departamento de Ciência da Computação da Universidade Federal de Lavras – DCC/UFLA, utilizando o sistema operacional *Windows 2000 Professional*.

1.4 Descrição dos Capítulos

Este trabalho está organizado em sete capítulos. No capítulo 2, é apresentada uma visão geral sobre o processo de manutenção de software, expondo suas dificuldades, modelos e técnicas existentes para realizá-lo. No capítulo 3, são apresentados os conceitos de uma técnica empregada na manutenção de produtos de software conhecida como engenharia reversa, bem como suas características. No capítulo 4, é exposta outra técnica utilizada na manutenção de produtos de software chamada reengenharia, dando um enfoque maior nas definições de “o que” é a reengenharia - e na aplicabilidade dessa

atividade – e “por que” realizar a reengenharia de um produto de software. No capítulo 5, é exposta uma notação para modelagem de aplicações para *Web*, *UWE*. No capítulo 6 são apresentadas uma visão geral do produto de software, a nova funcionalidade implantada após a realização da manutenção e a modelagem do estado atual do Portal Corporativo da 6ª RPM. A conclusão obtida através da realização deste trabalho, suas contribuições para a corporação e sugestões de trabalhos futuros são apresentadas no capítulo 7.

2. MANUTENÇÃO DE PRODUTOS DE SOFTWARE

Este capítulo fornece uma visão geral de manutenção de produtos de software, as dificuldades em realizá-la, bem como as técnicas e os modelos existentes para realizar o processo de manutenção.

A seção 2.1 introduz os conceitos e as categorias de manutenção de software. A seção 2.2 aponta as dificuldades encontradas ao se deparar com a necessidade de realizar a manutenção. A seção 2.3 concentra na atual tecnologia de suporte a esta atividade. Modelos de processos de manutenção e suas características podem ser encontrados nas seções 2.4 e 2.5, respectivamente. Um breve sumário deste capítulo pode ser encontrado na seção 2.6.

2.1 Tipos de Manutenção de Software

Conforme citado em [CAPRETZ & CAPRETZ, 1996], a manutenção de produtos de software consiste de uma série de atividades requeridas para mantê-lo funcional após aceito e colocado em operação.

A preocupação atual sobre a manutenção é devida ao reconhecimento de que esta é a fase mais cara do ciclo de vida¹ de um produto de software. Além disso, a qualidade de reparos e atualizações no código-fonte é freqüentemente

¹ Amplo conjunto de estágios da engenharia de software que abrange: análise de requisitos, projeto, implementação, teste e manutenção de um produto de software [PRESSMAN, 2001].

pobre e pode comprometer a confiabilidade e o desempenho do produto de software.

Embora muitas atividades relacionadas à manutenção e ao desenvolvimento de produtos de software sejam similares, a manutenção possui características próprias, como detalhadas a seguir por [CAPRETZ & CAPRETZ, 1996]:

1. A manutenção é executada em um produto de software existente. Todas as mudanças introduzidas devem estar de acordo ou ser compatíveis com a sua arquitetura, o seu projeto e código-fonte;
2. A manutenção requer tipicamente que programadores gastem uma proporção significativa de seu tempo em tentar compreender como um produto de software é construído e como funciona;
3. A manutenção é geralmente ilimitada, continuando por muitos anos (enquanto seja economicamente viável), em contraste com o desenvolvimento, que é comprometido a um escala de tempo e a um orçamento;
4. Durante o desenvolvimento, dados de teste são criados. A manutenção pode usar estes dados e executar testes de regressão², ou, alternativamente, criar dados novos para testar adequadamente somente as mudanças e seus impactos no produto de software.

As atividades de manutenção são classificadas geralmente em quatro categorias [CAPRETZ & CAPRETZ, 1996]:

² Reexecução de algum subconjunto de testes que já foram conduzidos para garantir que as modificações não propagaram efeitos colaterais indesejáveis [PRESSMAN, 2001].

1. **Manutenção Perfectiva:** acréscimo no produto de software, alterando seu comportamento funcional, resultante de uma mudança na intenção original ou nos requisitos;
2. **Manutenção Adaptativa:** alteração do produto de software em resposta a uma modificação no ambiente dos dados (formatos de entrada e saída) ou no ambiente de processamento (hardware ou software);
3. **Manutenção Corretiva:** correção de erros que causam saída incorreta ou finalização anormal do produto de software;
4. **Manutenção Preventiva:** atualização do sistema de software para antecipar problemas futuros; isto envolve melhorar a qualidade do produto de software e a sua documentação, ou outros fatores de qualidade de software. Modificações nesta atividade não afetam o comportamento funcional do produto de software.

Nem todas as modificações pertencem estritamente a uma categoria ou a outra. Por exemplo, a manutenção corretiva pode também requerer adições na funcionalidade (manutenção perfectiva) de um subsistema. Similarmente, um subsistema pode ser re-projetado para melhorar a manutenibilidade (manutenção preventiva), devido à incapacidade de corrigir uma falha persistente nele.

2.2 Dificuldades da Manutenção

Pode-se dizer que a manutenção de software é um processo de mudança para garantir a sobrevivência do produto de software. No entanto, segundo [SCHNEIDEWIND, 1987], o processo de manutenção é bastante árduo devido a 3 fatores, a saber:

- Não existem formas ainda eficazes de acompanhar os produtos de software gerados e o processo de sua geração na manutenção. Os modelos disponíveis para garantia da qualidade de software são bastante completos, porém muito focados no desenvolvimento de novos projetos;
- Dependendo de como o produto de software foi construído, a realização de mudanças em geral desencadeia um efeito “dominó” sobre o qual tem-se muita dificuldade de previsão. Isso se deve ao fato de não ter uma ferramenta apropriada para identificação de conseqüências que uma manutenção pode acarretar. Este efeito é citado também por [RAJLICH & SRIDHAR, 1996], o qual propõe um modelo que permite identificar a propagação da mudança;
- Em geral, a manutenção é apresentada na Engenharia de Software como um processo iniciado imediatamente após a implantação do produto de software. Isso não seria um problema, não fosse o pouco ou nenhum envolvimento das equipes de manutenção durante o processo de desenvolvimento do software.

Uma visão diferente é dada por [COUSIN & COLLOFELLO, 1992]. Eles acreditam que as equipes de manutenção poderiam resolver melhor todos os problemas de manutenção se dispusessem de documentação atualizada, treinamento contínuo e ferramentas automatizadas. Uma combinação destes três itens poderia ser um bom caminho para a continuidade da qualidade do software.

Focando mais diretamente a documentação do produto de software, muitos esforços têm sido feitos na tentativa de melhorar o processo de documentação. No estudo de [GARLAND & CALLISS, 1991], é discutido um método para incorporar informações sobre a solução de problemas em uma base de dados que contém todo o acompanhamento da manutenção.

Muitas das questões citadas levam a identificar que a falta de modelos de qualidade definidos especificamente para a manutenção ou adaptados para esta atividade, assim como a carência de ferramentas que permitam a implementação destes modelos, podem vir a deteriorar a qualidade do produto de software ao longo do processo de manutenção.

2.3 Tecnologia de Suporte à Manutenção

Para abordar adequadamente as técnicas de manutenção, deve-se primeiramente considerar três conceitos dependentes: a existência de um processo de desenvolvimento de produtos de software, a presença de um produto de software a ser analisado e a identificação de níveis de abstração³ [OSBORNE & CHIKOFSKY, 1990].

Qualquer que seja o processo de desenvolvimento, espera-se que haja interação entre seus estágios e, talvez, recursão. Em um processo de desenvolvimento, os estágios iniciais envolvem conceitos mais gerais, independentes da implementação, enquanto os estágios finais enfatizam os detalhes de implementação. O aumento de detalhes durante o processo de desenvolvimento conceitua os níveis de abstração. Estágios iniciais do produto de software planejam e definem requisitos de alto nível quando comparado com a própria implementação.

Essa comparação é importante para deixar claro que nível de abstração e grau de abstração são grandezas distintas. Segundo [OSBORNE &

³ Abstração é definida como a habilidade de ignorar os aspectos de assuntos não relevantes para o propósito em questão, tornando possível uma concentração maior nos assuntos principais.

CHIKOFSKY, 1990], enquanto o nível de abstração é um conceito que diferencia os estágios conceituais do projeto, o grau de abstração é intrínseco a cada estágio. A evolução através das fases do processo de desenvolvimento envolve transições do nível mais alto de abstração nos estágios iniciais, para nível mais baixo nos estágios posteriores. As informações podem ser representadas em qualquer estágio do desenvolvimento, seja de forma detalhada (baixo grau de abstração), seja de forma mais sucinta ou global (alto grau de abstração).

Para que as técnicas de manutenção (especificamente engenharia reversa e reengenharia) sejam descritas de forma simplificada, são tomadas como base apenas três fases do processo de desenvolvimento, com níveis de abstração bem diferenciados (conforme a Figura 2.1), a saber:

- **Requisitos:** especificação do problema a ser resolvido, incluindo objetivos, restrições e regras de negociação (o quê);
- **Projeto:** especificação da solução (como);
- **Implementação:** codificação, teste e adaptação ao sistema operacional (realização).

A técnica tradicional, que avança progressivamente pelas fases do processo de desenvolvimento, é denominada engenharia progressiva (*forward engineering*). A execução dessa técnica consiste em partir de projetos independentes da implementação (fase de análise), que possuem alto nível de abstração, indo em direção à implementação física do produto de software. Em outras palavras, engenharia progressiva segue a seqüência de desenvolvimento estabelecida na fase de análise e na fase de projeto, visando à obtenção do produto de software implementado [OSBORNE & CHIKOFSKY, 1990].

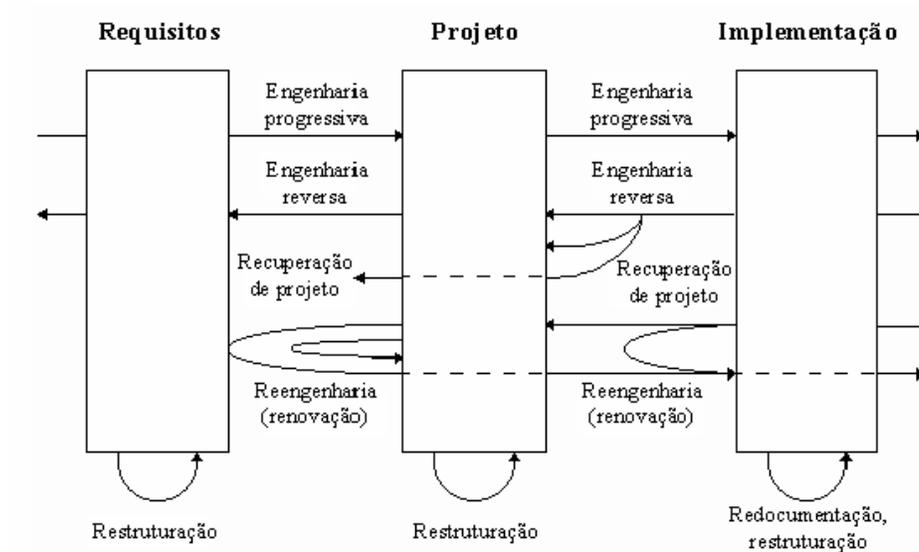


Figura 2.1: Relacionamentos no Ciclo de Desenvolvimento de Software [CHIKOFSKY & CROSS, 1990]

Na Figura 2.1, com exceção da engenharia progressiva, as demais transições entre as fases de desenvolvimento são tecnologias utilizadas na manutenção [CHIKOFSKY & CROSS, 1990], sendo assim definidas:

- **Redocumentação:** como uma subárea da engenharia reversa, é a criação ou a revisão de uma representação semanticamente equivalente, dentro do mesmo nível relativo de abstração, sendo que as formas resultantes de representação são consideradas como visões alternativas, utilizadas para uma melhor compreensão humana do produto de software analisado;
- **Recuperação de projeto:** é uma subárea da engenharia reversa na qual o conhecimento do domínio da aplicação, as informações externas e a dedução são adicionadas às observações referentes ao produto de software, para extraírem abstrações significativas de mais alto nível,

além daquelas obtidas através da observação direta do produto de software;

- **Reestruturação:** é a transformação de uma forma de representação, para outra no mesmo nível de abstração relativo, preservando o comportamento externo do produto de software (funcionalidade e semântica). Geralmente usada como uma forma de manutenção preventiva, a reestruturação é aplicada em produtos de software que tenham sido desenvolvidos de forma desestruturada, resultando uma representação que preserva as suas características, porém de forma mais bem estruturada;
- **Engenharia reversa:** é o processo de analisar um sistema com a finalidade de criar sua representação de uma forma diferente ou em um nível mais alto de abstração do que o código-fonte. Essa técnica é descrita detalhadamente no Capítulo 3.
- **Reengenharia:** é a reconstrução de algo do mundo real, tendo como propósito a busca por melhorias que permitam produzir algo de qualidade melhor ou comparável ao produto de software inicial. A reengenharia é detalhada no Capítulo 4.

No processo de manutenção, quando se trata de reconstruir um produto de software (ou seja, realizar sua reengenharia), é necessário que se proceda a sua engenharia reversa, a fim de obter o seu modelo de análise e o seu modelo de projeto. Esses modelos, com as devidas correções/alterações, serão o ponto de partida para a reengenharia.

O padrão IEEE P1219/D14 [SAGE, 1995] para manutenção define a reengenharia como um subconjunto da engenharia de software, composto por engenharia reversa e engenharia progressiva.

2.4 Modelos de Processos de Manutenção

Segundo [CAPRETZ & CAPRETZ, 1996], os modelos para desenvolvimento de produtos de software, como é o caso da maioria dos modelos de manutenção existentes, o representam em termos de fases. Por outro lado, os modelos de manutenção caracterizam o processo de manutenção somente da perspectiva do responsável pela manutenção. A estruturação das atividades de manutenção fornece um mecanismo útil para melhorar o processo. Entretanto, a aplicação destes modelos tem sido de benefício limitado em ajudar o processo de manutenção. Além disso, os modelos existentes não descrevem os processos reais que ocorrem durante a manutenção; eles somente podem fornecer meios de visualizar o processo em termos de produtos de software intermediários. Os marcos da fase poderiam ser associados com estes produtos, assim fornecendo um mecanismo pela qual a satisfação da gerência possa ser avaliada em relação aos requisitos gerais, orçamentos e programação.

Além disso, os modelos de manutenção, como modelos de desenvolvimento, devem fornecer um extenso auxílio para abranger inteiramente o processo. A modelagem do processo de produtos de software pode ser definida como uma metodologia que abrange uma tentativa de representação, análises detalhadas de potencialidades e a habilidade de fazer prognósticos a respeito dos efeitos das mudanças para um processo. Diversos objetivos têm sido citados como motivação para o desenvolvimento e a aplicação de modelos de processo. Estes incluem suporte para execução automatizada e controle, interação humana (tal como a orientação de execução), várias responsabilidades da gerência, compreensão e análise dos processos.

Assim, um modelo de processo de manutenção pode ser definido como a especificação de uma aproximação sistemática à manutenção [CAPRETZ & CAPRETZ, 1996]. Com a finalidade de estender a experiência e a tecnologia do

processo de desenvolvimento à manutenção, [CAPRETZ & CAPRETZ, 1996] citam quatro objetivos preliminares para o desenvolvimento de modelos de processo de software, a saber:

- Permitir uma comunicação eficaz, a respeito do processo aos gerentes, engenheiros de software e usuários;
- Facilitar o reuso do processo para permitir que um processo específico seja instanciado e executado de forma confiante e repetitiva através dos múltiplos projetos;
- Evolução do suporte ao processo servindo como um repositório para modificações, lições aprendidas, adaptando e analisando a eficácia das mudanças em um laboratório de ambientes simulados antes de executá-las. As decisões bem sucedidas devem então ser formalizadas e armazenadas como parte do modelo, de modo que possam consistentemente ser aplicadas no futuro;
- Facilitar o planejamento eficaz, o controle e a gerência operacional do processo. Isto é realizado através de uma melhor compreensão, treinamento, conformidade às definições do processo, simulações quantitativas, análise de potencialidades e definições e uso de medidas e métricas.

A fim de realizar estes objetivos, [CAPRETZ & CAPRETZ, 1996] afirmam que os modelos de processo devem possuir potencialidades em três categorias principais:

Um formalismo poderoso de representação é requerido para lidar com as complexidades reais dos processos organizacionais;

- Potencialidade de análises abrangentes, incluindo uma ampla variedade de testes nas áreas de consistência, integridade e precisão. Estas análises

são críticas em determinar a validade do modelo e do processo real que o modelo representa;

- Potencialidade de previsão que pode ser fornecida com a simulação que é integrada firmemente com a representação do modelo e análise das características.

Destas observações, uma lista de numerosas potencialidades desejáveis para uma tentativa ideal de modelar processos é apresentada a seguir [CAPRETZ & CAPRETZ, 1996]:

- Use uma aproximação altamente visual à representação da informação, tal como notações gráficas;
- Possibilite descrições resumidas, isto é, abrangente no escopo, contudo concisa na apresentação;
- Forneça suporte às múltiplas perspectivas complementares de um processo, tais como funcional, procedimental, organizacional e conceitual dos dados;
- Forneça suporte aos múltiplos níveis de abstração (por exemplo, decomposição hierárquica) para cada perspectiva;
- Ofereça uma sintaxe formalmente definida e semânticas, de modo que construções sejam computáveis;
- Forneça potencialidades de análises abrangentes. Isto envolveria testes nas categorias, tais como consistência, integridade e precisão;
- Facilite a simulação do comportamento do processo diretamente da representação;
- Forneça suporte à criação e à gerência de variações, versões e componentes reutilizáveis de modelos de processo;

- Forneça suporte à representação e à análise de restrições no processo, tal como regulamentos e padrões;
- Possibilite a representação das finalidades, objetivos e razões de componentes do processo e do processo total;
- Integre com outras aproximações que podem ser julgadas úteis;
- Faça um exame da função ativa na execução do processo;
- Ofereça ferramentas automatizadas que suportem a aproximação.

Com a disponibilidade de uma parcela das exigências descritas anteriormente, espera-se trazer benefícios substanciais aos processos de desenvolvimento e manutenção. Adicionalmente, estas exigências podem, no futuro, facilitar a evolução de processos de produtos de software de uma forma metódica e disciplinada.

2.5 Características da Modelagem do Processo de Manutenção

Conforme afirmam [BENNETT & HINLEY, 1992], os modelos de processos necessitam ter as seguintes características a fim de fornecer benefícios reais para a gerência de manutenção:

1. Uma cobertura detalhada do processo de manutenção, porém evitando a complexidade;
2. Um amplo escopo de tratamento de aspectos organizacionais, procedimentais e funcionais;
3. Identificação de objetos do mundo real por modelos (pedidos de mudança, registros de falhas);

4. Identificação de papéis explícitos das pessoas (gerente, engenheiro de software, usuário) que fazem parte do processo de manutenção;
5. Um formulário adequadamente esquematizado, de modo que pontos de medida e de controle possam ser estabelecidos;
6. Identificação do caminho no processo de comunicação, que não pode refletir uma seqüência de atividade ou uma hierarquia organizacional, por exemplo, mecanismos de controle de gerência;
7. Uma estrutura que guia gerentes nas suas atividades, por exemplo, como podem medir o desempenho do processo diante dos objetivos determinados, com o auxílio de um modelo de manutenção;
8. As características reutilizáveis, por exemplo, fornecem módulos de processos padrões que podem ser aperfeiçoados ou adaptados às circunstâncias individuais do projeto de manutenção tais como o processo de controle de pedido de mudança e o processo de liberação da mudança;
9. Flexibilidade de modo que os modelos possam ser rapidamente adaptados para cuidar de transformações de processos reais e de relacionamentos da mudança.

2.6 Resumo

Este capítulo apresentou uma visão geral da manutenção de produtos de software, abordando os problemas encontrados ao realizá-la.

Foi também exposto um exame dos modelos de processo de manutenção de produtos de software. Concluindo-se que, embora exista um número relativamente grande de modelos de manutenção de produtos de software, estes têm mostrado falhas que requerem pesquisas adicionais neste campo. As características relevantes da modelagem de manutenção de produtos de software

foram discutidas, a fim de introduzir a necessidade de gerar alternativas de aproximações para melhorar a manutenção de produtos de software legados.

3. ENGENHARIA REVERSA

Segundo [SALEH & BOUJARWAH, 1996], o mercado de produtos de software vem crescendo a cada dia e com ele o uso de técnicas de desenvolvimento, muitas vezes informais. Como resultado, a manutenção de tais produtos de software torna-se problemática, uma vez que a documentação associada a eles, na maioria das vezes, não está de acordo com o código-fonte implementado. Além disso, as constantes modificações e adições de novas características ao produto de software acarretam efeitos colaterais inesperados, que não estão presentes na documentação.

Muitas vezes, o engenheiro de software diante da manutenção do produto de software encontra uma documentação informal e incompleta, que não o reflete, tornando impossível o gerenciamento do processo de manutenção [SALEH & BOUJARWAH, 1996]. Neste caso, a Engenharia Reversa, com o propósito de recuperar as informações de projeto perdidas durante a fase de implementação e de documentar o seu real estado, pode auxiliar este processo de gerenciamento.

Na Seção 3.1, são discutidos alguns conceitos básicos sobre engenharia reversa, sendo apresentado em seguida, na Seção 3.2, visões de produtos de software com base em suas características e fases do ciclo de vida. Na Seção 3.3, são mostradas duas categorias de engenharia reversa. Um breve sumário deste capítulo pode ser encontrado na Seção 3.4.

3.1 Conceitos de Engenharia Reversa

O processo inverso à engenharia progressiva, caracterizado pelas atividades retroativas do ciclo de vida, que partem de um baixo nível de abstração para um alto nível de abstração, é conhecido como engenharia reversa [CHIKOFSKY & CROSS, 1990].

O termo “engenharia reversa” originou-se da análise de hardware [CHIKOFSKY & CROSS, 1990; REKOFF, 1985], que extraía o projeto a partir do produto final. Em geral, a engenharia reversa é aplicada para melhorar um produto ou para analisar um produto concorrente ou de um adversário (principalmente em situação militar ou de segurança nacional).

Aplicando o conceito inicial de engenharia reversa a produtos de software, muitas das técnicas utilizadas em hardware servem para obter uma compreensão básica do produto de software e da sua estrutura. Entretanto, enquanto o objetivo básico para hardware é duplicá-lo, os objetivos mais freqüentes para produtos de software são obter uma compreensão suficiente em nível de projeto para auxiliar a manutenção, fortalecer o seu crescimento e substituir o suporte [PIEKARSKI & QUINÁIA, 2000].

Para [PRESSMAN, 2001], a engenharia reversa de software é um processo de recuperação de projeto, consistindo em analisar um programa, na tentativa de criar uma representação, em um nível de abstração mais alto que o código fonte.

Segundo [BENEDUSI, 1992], pode-se definir engenharia reversa como uma coleção de teorias, metodologias e técnicas capazes de suportar a extração e a abstração de informações de um produto de software existente, produzindo documentos consistentes, seja a partir somente do código-fonte, seja através da adição de conhecimento e experiência que não podem ser automaticamente reconstruídos a partir do código-fonte.

Para [SAMUELSON, 1990], engenharia reversa é geralmente entendida como a ação de criar um conjunto de especificações funcionais para um produto de software, por alguém que não foi o projetista original, baseado na sua análise e nos seus componentes.

Segundo [WATERS & CHIKOFISKY, 1994], o processo de análise de um produto de software, para identificar seus componentes e seus inter-relacionamentos e para criar sua representação em outra forma, provavelmente em um nível mais alto de abstração que o código-fonte, constitui a engenharia reversa.

3.2 Visões de Produtos de Software

A partir da engenharia reversa e com base nos diferentes níveis e graus de abstração, o produto de software pode ser visualizado de diferentes maneiras [HARANDI & NING, 1990], a saber:

- **Visão em nível implementacional:** abstrai características da linguagem de programação e características específicas da implementação;
- **Visão em nível estrutural:** abstrai detalhes da linguagem de programação para revelar sua estrutura a partir de diferentes perspectivas. O resultado é uma representação explícita das dependências entre os seus componentes;
- **Visão em nível funcional:** abstrai a função de um componente, isto é, o que o componente faz. Essa visão relaciona partes do produto de software às suas funções, procurando revelar as relações lógicas entre elas (diferentemente das relações sintáticas ou das estruturais);

- **Visão em nível de domínio:** abstrai o contexto em que o produto de software está operando, ou seja, o porquê de desenvolvê-lo.

É relevante ressaltar que uma forma de representação extraída do código-fonte pode diferir de uma representação similar que foi desenvolvida no processo de engenharia progressiva. A forma extraída irá refletir a idiossincrasia da representação do código-fonte muito mais do que a representação original, que reflete a compreensão do problema pelo analista ou projetista.

A Figura 3.1 mostra a correspondência entre as categorias de visualização do produto de software e as diferentes atividades do seu ciclo de desenvolvimento.

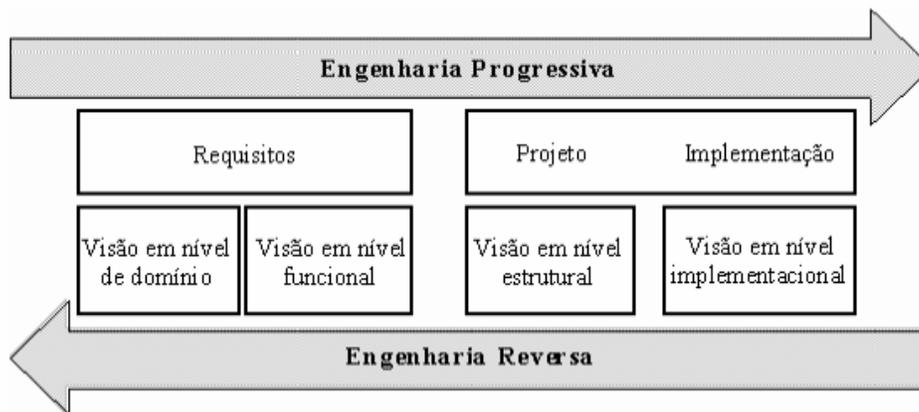


Figura 3.1: Visualizações de Software no Ciclo de Desenvolvimento [COSTA, 1997].

Muitas vezes, é necessário acrescentar às informações contidas no código-fonte, outras informações provenientes de conhecimentos e de experiências humanas, para obter visões diferenciadas do produto de software. Conforme o escopo das informações usadas, que resultam em um nível de

entendimento obtido do produto de software, pode-se formular uma categorização dos métodos de engenharia reversa.

3.3 Categorias

De acordo com o nível de entendimento obtido do produto de software e o escopo das informações usadas, duas categorias de engenharia reversa são definidas: visualização de código [OMAN, 1990] e entendimento de produto de software [CHIKOFFSKY & CROSS, 1990].

3.3.1 Visualização de Código

Também denominada redocumentação, a visualização de código é a criação ou revisão de representações semanticamente equivalentes em um mesmo nível de abstração [OMAN, 1990]. O processo de visualização de código cria as representações a partir de informações obtidas apenas da análise do código-fonte, embora a apresentação dessas informações possa se diversificar. As formas das representações são consideradas visões alternativas, cujo objetivo é melhorar a compreensão do produto de software.

A forma mais simples e mais antiga de engenharia reversa é a visualização de código. A intenção é recuperar a documentação que já existiu, ou que deveria ter existido, sobre o produto de software. A ênfase, de fato, é a criação de visões adicionais, especialmente visões gráficas, que não foram criadas durante o processo original de engenharia progressiva.

A visualização de código não transcende a visão em nível estrutural e não atribui significados ao produto de software analisado. Recuperações mais ambiciosas, tais como a função, os propósitos ou a essência do produto de

software, exigem um nível de entendimento maior e são definidas como seu entendimento.

3.3.2 Entendimento de Produto de Software

Nesta categoria de engenharia reversa, também denominada recuperação de projeto, o conhecimento do domínio das informações externas e as deduções são adicionados às observações feitas sobre o produto de software através do seu exame, de modo a obter informações com nível mais alto de abstração [CHIKOFFSKY & CROSS, 1990].

Segundo [BIGGERSTAFF, 1989], o entendimento de produto de software recria abstrações do projeto a partir de uma combinação de código-fonte, documentação existente do projeto (se disponível), experiências pessoais e conhecimentos gerais sobre o problema e o domínio de aplicação. Sintetizando, deve produzir todas as informações necessárias para entender completamente o que, como e por que o produto de software faz.

Entendimento de produto de software distingue-se de visualização de código porque objetiva entender o produto de software, ao invés de simplesmente fornecer visões alternativas para auxiliar o usuário a entendê-lo. Esse entendimento vai além do conhecimento em nível implementacional e estrutural, buscando obter o conhecimento em nível funcional e, até mesmo, em nível de domínio.

Um completo entendimento de produto de software busca reconstruir não somente a sua função, mas também o processo pelo qual foi desenvolvido. [RUGABER, 1990] enfatiza a importância da recuperação de decisões de projeto tomadas durante o desenvolvimento original para uma completa estrutura de entendimento.

A categoria de entendimento de produto de software é a forma mais crítica de engenharia reversa, porque tenta aproximar-se do raciocínio humano na busca do entendimento.

A Figura 3.2 apresenta a amplitude de alcance das categorias de engenharia reversa, relacionadas com o escopo das informações utilizadas (código-fonte ou base de conhecimento) e o nível de visualização pretendido (implementacional, estrutural, funcional e de domínio).

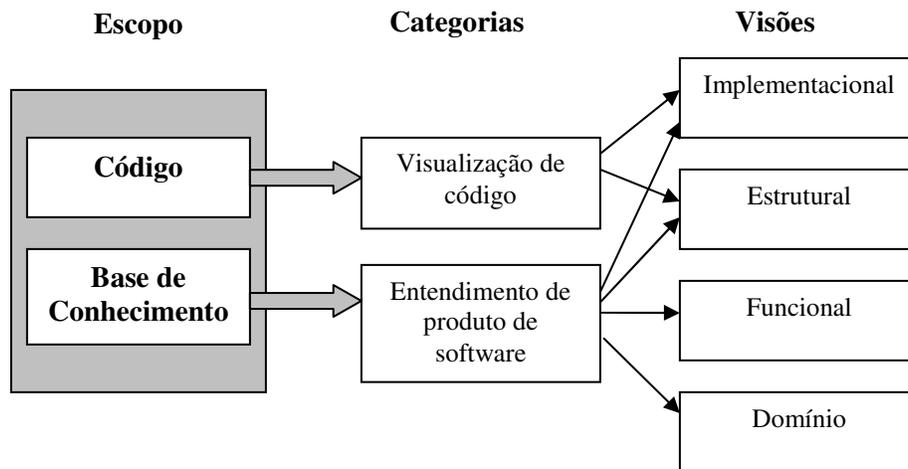


Figura 3.2: Categorias da engenharia reversa e suas visões [PIEKARSKI & QUINÁIA].

3.4 Resumo

Este capítulo abordou as definições de engenharia reversa, as visões de produtos de software e as suas categorias.

Pode ser observado que a engenharia reversa é de grande importância na manutenção de produtos de software, facilitando também sua futura

reengenharia. Unindo-se ao entendimento de produtos de software, a engenharia reversa produz excelentes resultados de grande utilidade para o engenheiro de software responsável por alterações, reuso e evolução do produto de software. Quando os modelos por ela obtidos, seguem o paradigma da orientação a objetos, ainda mais vantagens são oferecidas, principalmente quanto à facilidade de reengenharia com mudança de orientação do produto de software.

A engenharia reversa pode ser realizada com base nos diferentes métodos para desenvolvimento de produtos de software, sendo facilitada quando o método adotado apresenta transição suave entre as fases de análise e projeto e as fases de projeto e implementação. Por exemplo, se as fases de projeto e implementação possuem correspondência natural, fica mais fácil caminhar no sentido contrário, como é necessário fazer na engenharia reversa.

4. REENGENHARIA

A reengenharia pode ser usada para reestruturar e extrair componentes reutilizáveis, assim como fornecer novas visões de um produto de software e de sua documentação. A esperança é que a manutenibilidade e a adaptabilidade de produtos de software legados⁴ possam ser melhoradas. A base de toda a tentativa de reengenharia é tentar fazer produtos de software legados mais fáceis de manter.

Este capítulo fornece definições de alguns autores a respeito de reengenharia na seção 4.1. Categorias de reengenharia são apresentadas na seção 4.2. Como e porquê realizar a reengenharia de produtos de software são mostrados na seção 4.3 e 4.4, respectivamente. O processo de reengenharia é definido na seção 4.5 e as ferramentas de auxílio que podem ser utilizadas neste processo são apontadas na seção 4.6. Por fim, um breve resumo do capítulo é apresentado na seção 4.7.

4.1 Definições

Para [CHIKOFSKY & CROSS, 1990; IEEE CS-TCSE, 1997; GT-REG, 1998], a reengenharia, conhecida também como renovação ou reconstrução, é o exame e a alteração de um produto de software, para reconstituí-lo em uma nova forma e a subsequente implementação dessa nova forma. Um processo de

⁴ Grandes produtos de software que não se sabe como lidar, mas que são vitais em nas organizações [BENNETT, 1995].

reengenharia geralmente inclui alguma forma de engenharia reversa, seguida por uma forma de engenharia progressiva ou reestruturação.

Para [WARDEN, 1992], a reengenharia tem como principal objetivo melhorar um produto de software de alguma maneira, através de alterações significativas que proporcionem melhoria, porém, sem alterar suas funções. A extração automática da descrição de uma aplicação e sua implementação em outra linguagem não é considerada, segundo o autor, reengenharia, e sim tradução de código. Do mesmo modo que [CHIKOFSKY & CROSS, 1990], [WARDEN, 1992] considera que a reengenharia pode ser dividida em duas fases principais: a Engenharia Reversa e a Engenharia Progressiva; e cada uma destas fases pode ser dividida em uma série de atividades.

[PREMERLANI & BLAHA, 1995] citam que o objetivo da reengenharia é reutilizar automaticamente os esforços de desenvolvimento passados, objetivando reduzir custos de manutenção e melhoria na flexibilidade do produto de software.

Segundo [PRESSMAN, 2001], a reengenharia, também chamada de recuperação ou renovação, recupera informações de projeto de um produto de software existente e usa essas informações para alterá-lo ou reconstituí-lo, preservando as funções existentes, ao mesmo tempo em que adiciona novas funções, em um esforço para melhorar sua qualidade global.

Para [SOMMERVILLE, 1995], a reengenharia é descrita como a reorganização e a modificação de produtos de software existentes, parcial ou totalmente, para torná-los mais manuteníveis.

Das diversas definições citadas, percebe-se que existe clara distinção entre o desenvolvimento de um novo produto de software e reengenharia. A distinção está relacionada ao ponto de partida de cada um dos processos. O desenvolvimento de um novo produto de software (definido como engenharia progressiva por [CHIKOFSKY & CROSS, 1990]) inicia-se com a sua

especificação escrita, enquanto que a reengenharia inicia-se tomando como base um produto de software já desenvolvido.

Nota-se, também, que existe distinção entre os objetivos da reengenharia e os da engenharia reversa. O objetivo da reengenharia é reestruturar um produto de software atribuindo a ele características de manutenibilidade. O objetivo da engenharia reversa é derivar o projeto ou especificação de um produto de software, partindo-se de seu código-fonte [SOMMERVILLE, 1995]. Esta atividade é usada como parte do processo de reengenharia, pois fornece o entendimento do produto de software a ser reconstruído, como pôde ser visto no Capítulo 3.

4.2 Categorias

Em [SAGE, 1995], são indicadas diversas categorias de melhorias relacionadas à reengenharia, entre elas:

- **Reengenharia de processos administrativos:** é direcionada para alterações potenciais em todos os negócios ou processos organizacionais;
- **Reengenharia de processos produtivos:** consiste em modificar qualquer ciclo de processos padrão, que esteja em uso em uma dada organização, a fim de melhor acomodar as tecnologias novas e emergentes, bem como os requisitos dos clientes para um produto de software;
- **Reengenharia de produtos de software:** é o exame, o estudo, a captura e a modificação de mecanismos internos ou funcionalidade de um produto de software existente, visando reconstituí-lo em uma nova

forma e com novas características, freqüentemente para tomar vantagem das novas e emergentes tecnologias, mas sem grandes alterações na funcionalidade e propósito inerentes ao produto de software.

4.3 Por que Realizar a Reengenharia

Segundo [HAMMER & CHAMPY, 1994], a tendência de reengenharia dos processos das empresas é influenciada por fatores tais como: i) a necessidade de melhoria da qualidade dos serviços e produtos oferecidos; ii) a compressão das margens de lucro; iii) a redução do ciclo de vida dos produtos; iv) a diminuição da interferência dos governos e dos subsídios; v) a explosão tecnológica; vi) o rápido crescimento do conhecimento humano; vii) a maturidade dos mercados de consumo e; viii) a globalização da economia.

Outros fatores são relacionados com a complexidade das atividades empresariais, tais como: i) a busca por produtividade; ii) a flexibilidade frente às constantes mudanças; iii) a concentração no ramo de negócio; iv) os relacionamentos com clientes, com o meio ambiente e com os governos; v) o apoio de consultores; vi) a parcerização; e vii) a gestão e a remuneração dos recursos humanos por resultados. Todos esses pontos influenciam diretamente na reengenharia de software existente em empresas [HAMMER & CHAMPY, 1994]

Além disso, todos os produtos de software têm um tempo de vida limitado, sendo que cada alteração efetuada pode degenerar a sua estrutura, fazendo com que as manutenções se tornem cada vez mais difíceis e dispendiosas. Isso ocorre principalmente em produtos de software legado.

4.4 Como Realizar a Reengenharia

O processo de reengenharia é constituído de duas fases distintas. Na primeira, o produto de software objeto de reconstrução é “desmontado”, visando seu entendimento. Na segunda, o produto de software é reconstruído, na forma desejada, a partir do resultado da primeira fase, sendo incluídos os ajustes que se fizerem necessários.

O processo de Reengenharia pode ser traduzido como [JACOBSON & LINDSTRÖM, 1991]:

| |
|---|
| Reengenharia = Engenharia Reversa + Δ Engenharia Progressiva |
|---|

onde Δ pode ser de dois tipos:

- **Alterações parciais de funcionalidade:** ocorrem devido a mudanças nos negócios ou a necessidade do usuário;
- **Alterações de implementação:** ocorrem devido a alterações no ambiente de operação do software e/ou linguagem de implementação (protocolos, sistema operacional, portabilidade, linguagens, etc.).

Existem alguns pontos que devem ser considerados para um processo de reengenharia [WARDEN, 1992], tais como:

- Deve ser executado somente se existir um argumento aceitável da relação custo/benefício;
- Implica melhoria através de re-projeto;
- Deve remover projetos ruins, mas reconhecer e manter projetos bons e simples, mesmo que eles sejam desestruturados;

- A Engenharia Reversa é dirigida por tipos de problemas, os quais necessitam ser identificados;
- Os problemas são identificados e expressos como violações às técnicas de projeto estruturado e regras de programação, ou outras que o usuário pode definir;
- Ferramentas devem ser adequadas aos processos de reengenharia e não os processos adequados às ferramentas.

4.5 O Processo de Reengenharia

Em [JACOBSON & LINDSTRÖM, 1991], é declarado que para executar um processo de reengenharia de um produto de software, é necessário:

- **Realizar a engenharia reversa:** identificar como os componentes do produto de software se relacionam uns aos outros e criar uma descrição mais abstrata do produto de software. Um exemplo de identificação de relacionamentos entre componentes pode ser a identificação de dependências entre os arquivos e as funções, entre as funções e as descrições da base de dados, etc. Um exemplo da criação de uma descrição mais abstrata do sistema pode ser um diagrama de fluxo de dados (DFD) para as funções e o modelo entidade-relacionamento (MER) para as descrições da base de dados. Com esse primeiro passo, obtém-se um modelo abstrato, que mostra a funcionalidade do produto de software e um número de mapeamentos entre os diferentes níveis de abstração. Os mapeamentos compreendem as decisões de projeto que ocorrem quando se transforma uma representação abstrata em uma representação concreta;

- **Decidir sobre alterações na funcionalidade:** as alterações de funcionalidade são as alterações nos requisitos que o usuário determina que sejam implementadas no produto de software. Esse passo é executado utilizando-se as abstrações de mais alto nível, obtidas no passo anterior. Sem o modelo abstrato, é necessário decidir questões de alto nível utilizando comandos de baixo nível (por exemplo, um comando de alto nível tal como “Altere a associação entre as entidades X e Y” deveria ser traduzido para um outro de baixo nível, tal como “Adicione uma tabela que contenha as referencias entre X e Y”);
- **Re-projetar o produto de software:** parte-se das abstrações de alto nível, obtidas nos passos anteriores, para uma representação mais concreta, ou seja, executa-se a engenharia progressiva reimplementando o produto de software. Neste processo, deve-se levar em consideração as alterações de técnicas de implementação. Se apenas parte do produto de software for alterada, deve-se considerar questões sobre a integração/comunicação entre as suas partes velhas e novas.

Segundo [RAMAMOORTHY & TSAI, 1996], os métodos de engenharia reversa existentes até o momento não recuperam de modo automático todas as visões do produto de software. Isso acontece basicamente porque a fase de implementação caracterizada principalmente por códigos-fonte e descrição de arquivos não contém todas as informações essenciais para o processo de engenharia reversa, as quais são providas pela fase de análise. Dessa forma, há a necessidade da intervenção humana para extrair boas representações de projetos de produtos de software, principalmente em níveis mais altos de abstração [TANGORRA & CHIAROLLA, 1995].

4.6 Ferramentas de Auxílio à Reengenharia

Existem ferramentas com a finalidade de auxiliar a execução da reengenharia. A maioria das ferramentas é utilizada na etapa de engenharia reversa do produto de software a ser reconstruído.

Segundo [PRESSMAN, 1995], as ferramentas baseadas em engenharia reversa estão ainda “engatinhando”, ficando claro que as pesquisas na área de entendimento de código são muito importantes e muitas outras pesquisas ainda acontecerão.

As ferramentas devem ser adequadas aos processos de reengenharia, e não os processos adequados às ferramentas. As ferramentas de suporte disponíveis para auxiliar a reengenharia têm influencia sobre os custos de reengenharia.

Existem muitas ferramentas de reengenharia com aplicabilidade em produtos de software. O Quadro 4.1 apresenta algumas dessas ferramentas, mostrando de onde provem as informações de entrada e o que cada uma delas produz como resultado [PIEKARSKI & QUINÁIA, 2000].

Quadro 4.1: Escopo das informações utilizadas por ferramentas de reengenharia, as respectivas visões e outras saídas produzidas (continua)

| Ferramenta | Escopo das informações utilizadas | | Visões produzidas | | | | Outras saídas produzidas | Referências bibliográficas |
|------------|-----------------------------------|---|-------------------|------------|-----------|---------|--|-------------------------------|
| | Código | Base de conhecimento a partir de | Implementacional | Estrutural | Funcional | Domínio | | |
| Art | C | | √ | √ | | | | [TONELLA <i>et al</i> , 1996] |
| Decode | Cobol | | √ | √ | | | Base de conhecimento sobre o sistema | [CHIN & QUILICI, 1996] |
| Desire | C | Documentação. Biblioteca de componentes padrões de produtos de software. Informações informais. | √ | √ | √ | | | [BIGGERSTAFF, 1986] |
| Docket | Cobol e C | Documentação. Interação com usuário. | √ | √ | √ | √ | Base de conhecimento sobre o produto de software | [LAYZELL <i>et al</i> , 1995] |

Quadro 4.1: Escopo das informações utilizadas por ferramentas de reengenharia, as respectivas visões e outras saídas produzidas (continuação)

| Ferramenta | Escopo das informações utilizadas | | Visões produzidas | | | | Outras saídas produzidas | Referências bibliográficas |
|-----------------|-----------------------------------|---|-------------------|------------|-----------|---------|-------------------------------|-------------------------------|
| | Código | Base de conhecimento a partir de | Implementacional | Estrutural | Funcional | Domínio | | |
| Man-drake | Assembly | | √ | √ | | | | [MORRIS & FILMAN, 1996] |
| Newcomb | Cobol | | √ | √ | | | | [NEWCOMB & KOTIK, 1995] |
| Pat | Cobol | | √ | √ | | | | [NEWCOMB & KOTIK, 1995] |
| Re ² | Genérico | | √ | √ | | | | [CANFORA <i>et al</i> , 1994] |
| Redo | Cobol e Fortran | Biblioteca de componentes padrões de software | √ | √ | √ | | | [BENNETT, 1991] |
| Rigi | Cobol e C | | √ | √ | √ | | | [WONG <i>et al</i> , 1995] |
| Sneed | Cobol | Base de conhecimento do programador | √ | √ | √ | | Classes e métodos (CO-BOL-OO) | [SNEED, 1995] |
| Macs | C | Repositório com conhecimentos do domínio de aplicação | √ | √ | √ | | | [BENNETT, 1991] |

O escopo das informações que cada ferramenta utiliza é proveniente:

- **Do código fonte do produto de software:** cada ferramenta trabalha com código-fonte em uma determinada linguagem;
- **Da base de conhecimento do produto de software:** constituída por documentação existente, informações de usuários ou projetistas do produto de software e bibliotecas de componentes, entre outras.

Em relação aos resultados produzidos, quando apenas o código-fonte é utilizado como entrada, as visões fornecidas são, em geral, em nível de implementação e estrutural. Para se obter visões mais abstratas (funcional e de domínio), bem como outras saídas, são necessárias utilizar como entrada, além do código fonte, bases de conhecimento sobre o produto de software.

4.7 Resumo

Este capítulo apresenta um enfoque maior nas definições – “o que” é a reengenharia e outros termos relacionados a manutenção de produtos de software – e na aplicabilidade dessa atividade – “por que” realizar a reengenharia de um produto de software.

O “como”, ou seja, a forma de realizar a reengenharia em um produto de software depende de muitos fatores, relacionados, por exemplo, com a forma de trabalho da empresa, a origem do produto de software em questão e com o desenvolvimento de tecnologias para apoiar essa atividade. Algumas ferramentas disponíveis foram incluídas, com o intuito de fornecer uma visão geral de como executar a reengenharia de um produto de software.

A importância da reengenharia está em possibilitar que todo conhecimento agregado ao produto de software legado não seja perdido, o que aconteceria se a opção fosse pelo desenvolvimento de um novo produto de software.

O processo de extração do conhecimento de um produto de software legado é realizado pela engenharia reversa, que fornece visões mais abstratas do que o seu código-fonte e/ou alguma outra documentação possivelmente existente. Com essas visões, é possível compreender o produto de software e o escopo em que está inserido, possibilitando a produção do modelo de análise que servirá a reengenharia.

Da reengenharia de um produto de software resulta uma nova versão, que agrega toda a informação do produto de software legado, as inovações tecnológicas e a nova funcionalidade desejada. Dessa forma, a reengenharia não trata apenas de modernizá-lo, mas também adaptá-lo de acordo com as novas necessidades do processo em que está inserido.

5. UML-BASED WEB ENGINEERING

A *Web* tornou-se parte de nossa vida diária. Há cada vez mais aplicações da *Web* no nosso dia a dia e as aplicações estão se tornando maiores. Entretanto, modelar aplicações da *Web* é ainda uma disciplina nova, os métodos e as linguagens existentes, especialmente a UML, não suportam adequadamente a modelagem *Web*.

A aproximação *UML-based Web Engineering (UWE)* tenta resolver este problema introduzindo uma extensão ao UML (*Unified Modeling Language*), usando os seus mecanismos de extensão. Neste capítulo, será feito um exame no processo de modelagem UWE, apresentando como poderia ser usado para modelar aplicações *Web*.

Na seção 5.1 será definida a linguagem UWE, em seguida, suas características e suas atividades são apresentadas na seção 5.2. Um breve resumo pode ser encontrado na seção 5.3.

5.1 O que é UWE

A *Unified Modeling Language (UML)* é uma linguagem popular, padronizada para especificação, visualização, construção e documentação de sistemas de software [BOOCH, 1998]. É também um padrão na indústria e a linguagem mais geralmente usada em processos orientados a objetos da tecnologia de programação. Entretanto, sua sustentação para aplicações da *Web* é considerada insuficiente. Não há, por exemplo, nenhum “padrão” de elementos

de modelo que representam um *menu* ou um índice definido, nem elementos que representam trajetos da navegação entre locais diferentes da *Web*.

Uma solução possível é estender a UML. De fato, a UML é definida como auto-extensível, isto é, os mecanismos de extensão são definidos na própria UML. Com seus mecanismos de extensão, as soluções específicas para situações específicas, tais como sistemas de tempo real ou aplicações *Web*, podem ser desenvolvidas. Tais extensões são chamadas *Perfil* e são padronizadas pelo *Object Management Group* (OMG), <http://www.omg.org/>.

Um destes perfis para modelagem de aplicações *Web* é a *UML-based Web Engineering* (UWE) proposta por [KOCH, 2002]. Na UWE, elementos de modelagem (especialmente classes) são estendidos, novas interfaces são definidas e é introduzida uma nova abordagem sobre modelagem de aplicações *Web*.

De acordo com [KOCH, 2002], as principais diferenças entre um projeto normal, aplicações independentes e aplicações *Web* são:

1. A heterogeneidade do grupo de projetistas;
2. A estrutura composta de nós e *links*;
3. A necessidade de assistente de navegação;
4. Os conteúdos multimídia e a apresentação destes conteúdos em diferentes *browsers*.

5.2 Visão Geral

A UWE introduz uma abordagem sobre a modelagem incluindo três modelos, cada um focando um aspecto central das aplicações *Web*: conteúdo, navegação e apresentação.

Segundo [KOCH, 2001], as características principais da UWE são:

1. É uma abordagem inteiramente orientada a objeto;
 - Apresenta modelo de referência visual como apresentado no modelo da UML;
 - Suporta técnicas de modelagem visual;
 - Fornece perfil de extensão UML adaptado para aplicações hipermídia;
 - Define um processo de desenvolvimento que cobre todo o processo de criação de aplicações hipermídia.

Conforme citado em [KOCH, 2001], as características principais da técnica de modelagem UWE são:

- Suporta modelo visual e sistemático;
- Saídas hipermídia, como por exemplo, conteúdo, apresentação e navegação são tratados separadamente de acordo com a modelagem do usuário e as adaptações das saídas;
- Fornece um perfil da UML baseado em seus mecanismos de extensão e o utiliza para construir os modelos de análise e desenvolvimento.

As características principais do processo de desenvolvimento da abordagem UWE, de acordo com [KOCH, 2001], são:

- É orientado a objeto, baseado no processo iterativo e incremental;
- Molda o Processo Unificado (*Unified Process*, segundo [JACOBSON, BOOCH & RUMBAUGH, 1999]) para o desenvolvimento de aplicações hipermídia adaptáveis descrevendo

quais trabalhadores são requeridos, quais atividades eles devem realizar e quais produtos eles devem produzir;

- Estende os riscos do ciclo de desenvolvimento do Processo Unificado incluindo uma fase de manutenção;
- Adiciona processos de desenvolvimento de suporte *workflow* para gerenciamento de projetos e gerenciamento de qualidade;
- Muda o plano de gerenciamento do controle de qualidade incorporando *workflows* para validação de requisitos e verificação do projeto.

Segundo [KOCH, 2001], os principais produtos produzidos pelo método de desenvolvimento da UWE são:

- Um modelo de casos de uso que captura os requisitos do sistema;
- Um modelo conceitual para o conteúdo;
- Um modelo de usuário;
- Um modelo de navegação que compreende um modelo do espaço de navegação;
- Um modelo da estrutura de navegação;
- Um modelo de apresentação que compreende modelos estáticos e dinâmicos (estrutura do modelo de apresentação, fluxo do modelo de apresentação, modelo abstrato do usuário da interface e modelo do ciclo de vida do objeto);
- Um modelo de adaptação.

A UWE inicia com uma fase de análise de requisitos identificando casos de uso. O resultado da análise de requisitos é chamado modelo de casos de uso em UWE.

Segundo [KOCH, 2002], no desenvolvimento de processos UWE, uma aplicação *Web* deve ser modelada seguindo os seguintes modelos: Modelo Conceitual, Modelo de Navegação e Modelo de Apresentação.

No **Modelo Conceitual**, as classes e os objetos participantes do sistema e as relações entre eles são modeladas. Exatamente como nos sistemas tradicionais orientados a objeto, a modelagem é realizada da mesma forma, ou seja, encontrar as classes, definir estrutura de herança, especificar restrições, etc. Elementos do modelo usados no modelo conceitual são: Classes Conceituais, Pacotes e Associações. Pacotes e Associações são como na UML não estendida. Classe Conceitual é uma subclasse da classe por adição de um atributo, se a navegação é relevante ou não. Isto indica se esta classe conceitual é relevante para o modelo de navegação.

No **Modelo de Navegação**, a estrutura de navegação da aplicação *Web* é modelada. Para cada classe conceitual relevante para a navegação no modelo conceitual, existe uma classe de navegação adicionada para este modelo. Associações entre classes de navegação são adicionadas se suas classes conceituais são conectadas umas com as outras por associações no modelo conceitual. Uma associação deve ser adicionada para representar os caminhos de navegação de uma classe de navegação para outra. Além disso, acessos primitivos são adicionados para modelar as possibilidades para o usuário poder navegar na aplicação. Elemento de navegação é a forma genérica da classe de navegação. Cada elemento de navegação representa um nó da navegação *Web*.

Existem quatro tipos de acessos primitivos definidos: i) um índice tem itens de índice, o qual todos apontam para instâncias da mesma classe de navegação; ii) *guided tours* fornecem acesso seqüencial para instâncias de uma

classe de navegação; iii) consultas são usadas para selecionar algumas instâncias de uma classe de navegação; e iv) *menus* possuem itens de *menus*, os quais possuem pontos para instâncias de diferentes tipos de elementos de navegação. Acessos primitivos conectam elementos da navegação com cada outro e são necessários para descrever a estrutura de navegação de uma aplicação *Web* completa.

No **Modelo de Apresentação**, a estrutura de apresentação da aplicação *Web* é modelada. Para cada elemento apresentável no modelo de apresentação, existe uma classe de apresentação adicionada ao Modelo de Apresentação. Classes de apresentação podem ser colocadas em *frames*⁵, itens de apresentação como Textos, Imagens, *Links*, entre outros devem ser adicionados às classes de apresentação.

O Modelo de Apresentação é a representação de onde e como os objetos da navegação e os acessos primitivos serão apresentados para o usuário. O projeto da apresentação suporta a transformação do modelo da estrutura da navegação em um conjunto de modelos que mostram o local estático dos objetos visíveis para o usuário, o esquema de representação destes objetos (páginas no projeto da aplicação *Web*) e dos seus comportamentos dinâmicos. O esquema da representação é similar à técnica de desenvolvimento usada por alguns projetistas de interface. O projeto de apresentação focaliza a organização estrutural da apresentação, como por exemplo, textos, imagens, formulários e *menus* e não a aparência física em termos de formatos especiais, cores, etc. Deste modo, decisões são tomadas durante o desenvolvimento de um protótipo da interface ou em uma fase de implementação.

⁵ Múltiplas telas em um documento.

5.3 Resumo

Neste capítulo, foram mostradas as atividades de projeto da aproximação *UML-based Web Engineering (UWE)* que por meio de sua extensão introduz tarefas de modelagem, destacando um papel importante dentro da modelagem de aplicações para *Web*. Foram também apresentadas as definições de UWE, bem como suas características.

6. IMPLANTAÇÃO E DOCUMENTAÇÃO DO NOVO PORTAL CORPORATIVO DA 6ª RPM

O Portal Corporativo da 6ª RPM foi desenvolvido com o intuito de integrar as informações e todo o pessoal que o compõe. A manutenção e a documentação do Portal surgiram com o propósito de satisfazer as necessidades apontadas pelos usuários do produto de software e possibilitar sua melhor utilização dentro da instituição e pelo público em geral, através de serviços oferecidos pela Polícia Militar de Minas Gerais.

Este capítulo fornece uma visão de todo o produto de software, a nova funcionalidade implantada no Portal e sua modelagem. A seção 6.1 descreve a arquitetura que constitui o produto de software e a estrutura do Portal é descrita na seção 6.2. As seções 6.3 e 6.4 apresentam o novo *design* aplicado às páginas e a nova funcionalidade do Portal Corporativo da 6ª RPM, respectivamente. A seção 6.5 apresenta a modelagem do produto de software e um breve resumo do capítulo pode ser encontrado na seção 6.6.

6.1 Arquitetura

O Portal Corporativo da 6ª RPM consiste de uma aplicação cliente-servidor que usa a internet para sua execução. Uma aplicação cliente-servidor consiste na divisão de processos entre estações clientes e servidores, com a finalidade de buscar melhor desempenho, menor tempo de resposta e maior

facilidade de manutenção. Assim, no produto de software, terão processos que serão executados na máquina do cliente e outros que serão executados no servidor *Web*. Estes processos executados no servidor geralmente estão relacionados a páginas dinâmicas ou de acesso ao banco de dados, que são executados e enviados novamente para o cliente, porém agora de uma forma fácil para interpretação pelo *browser* do cliente.

Como o produto de software é executado em navegadores *Web*, as linguagens utilizadas no seu desenvolvimento foram: HTML (*Hipertext Markup Language*) e ASP (*Active Server Pages*). A linguagem HTML é usada para o desenvolvimento de páginas estáticas que são geralmente executadas no cliente, enquanto que a linguagem ASP foi utilizada para a páginas dinâmicas, executadas no servidor, e que permitem o acesso ao banco de dados. As páginas que compõem o Portal Corporativo foram hospedadas no servidor *Microsoft Information Server* (IIS) e é neste servidor que configurações foram feitas para a execução correta da aplicação, através da criação de diretórios virtuais, definição de permissões de acesso entre outras.

O produto de software realiza consultas SQL para acessar informações no Sistema de Banco de Dados Relacional utilizado, ou seja, o *Microsoft SQLServer*, via ODBC (*Open Database Connectivity*). O driver de banco de dados de ponte ODBC para *SQLServer* permite que qualquer programa acesse qualquer fonte de dados *SQLServer*.

Os requisitos para a utilização do produto de software são:

- Acesso à internet;
- *Browser Web*;
- Sistema operacional Windows (95, 98, ME e NT) ou Linux;

6.2 Estrutura do Produto de Software

A necessidade de integrar as informações presentes na corporação da Sexta Região da Polícia Militar levou a criar uma ferramenta que fornecesse um ponto único de acesso onde as pessoas poderiam obter informações e serviços úteis de caráter corporativo, além da criação de meios eficientes de comunicação. O portal visou também atender ao público em geral, através de serviços como o Net Denúncia e o Fale Conosco, além de disponibilizar dicas e informações como eventos, concursos e projetos.

Justamente para diferenciar o que é de acesso geral e de acesso privado, foi desenvolvido um portal tal que fosse dividido em três seções: uma para o público geral, outra para os membros da Polícia Militar de Minas Gerais e uma terceira para o pessoal da Imprensa. Na seção geral, há um *link* que proverá acesso à seção privada dos usuários da Polícia Militar, denominada Intranet, e outro *link* que dará acesso a seção privada destinada a Imprensa. Somente usuários cadastrados terão o direito de acessar a Intranet ou a seção Imprensa do Portal. A Intranet por sua vez é dividida em dois módulos: o módulo usuário privado, ou seja, o usuário que possui acesso a Intranet e o módulo usuário administrador, o qual possui também a permissão de realizar operações sobre as informações disponibilizadas pelo Portal. A página principal de cada seção pode ser vista nas Figuras 6.1, 6.2, 6.3 e 6.4, sendo respectivamente, seção usuário geral (Internet), seção usuário privado da Imprensa, seção usuário privado (Intranet) e módulo Administração.



Figura 6.1 – Página inicial do Portal Corporativo da 6ª RPM (seção Internet)



Figura 6.2 – Página da Imprensa do Portal Corporativo da 6ª RPM

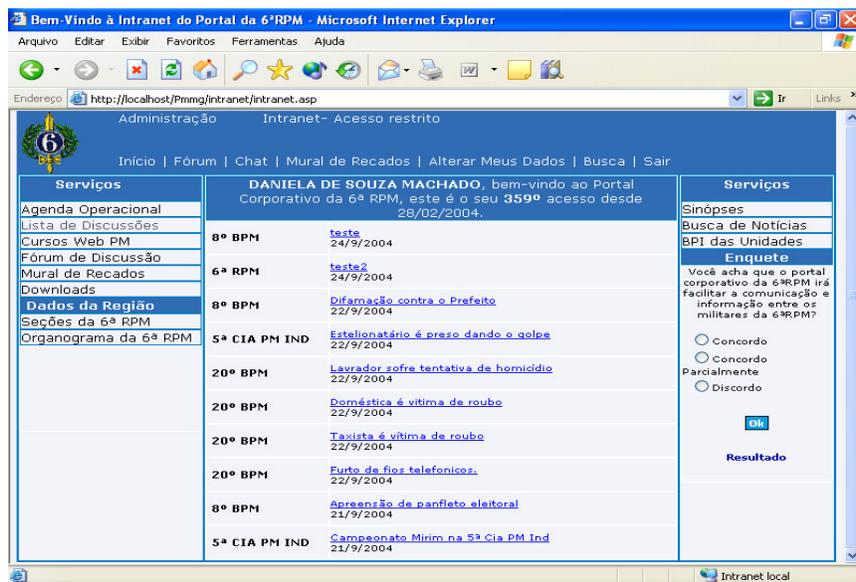


Figura 6.3 – Página da Intranet do Portal Corporativo da 6ª RPM

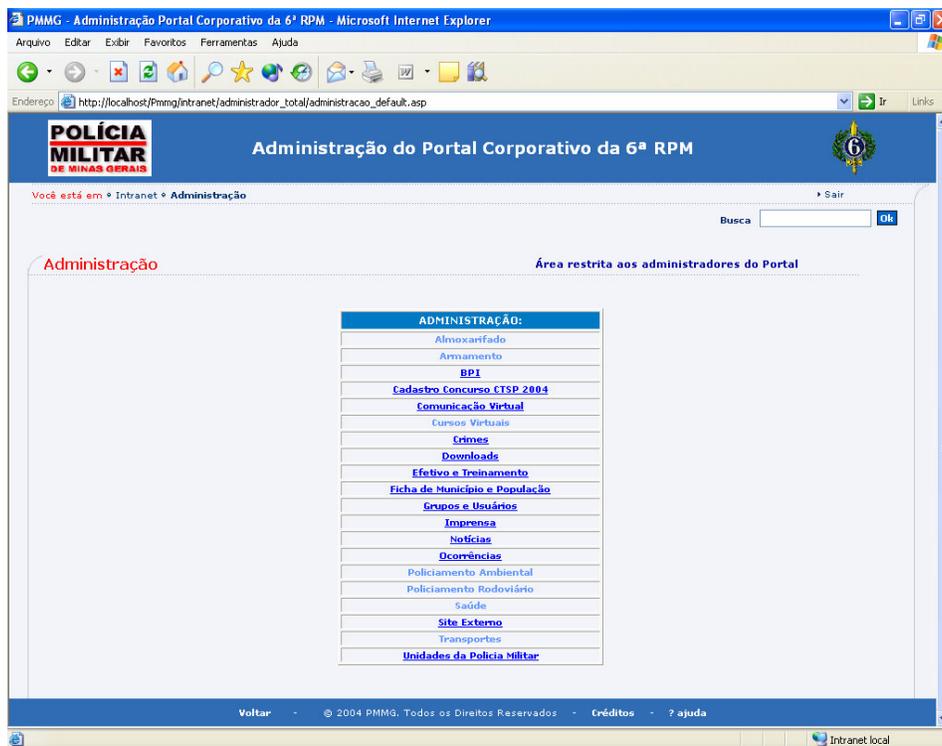


Figura 6.4 – Página da Administração do Portal Corporativo da 6ª RPM

6.3 Design

Embora a tecnologia de portais corporativos seja recente, vários são os benefícios apontados por fornecedores e consultores de informática associados a sua aplicação nas instituições. Dentre esses benefícios, destacam-se a facilidade de acesso às informações distribuídas e a facilidade de comunicação corporativa. Porém, para conseguir concretizar esse benefício, é fundamental que o projeto do Portal Corporativo leve em consideração a interação dos usuários com sua interface. Sua capacidade de facilitar o acesso dos usuários às informações e aos

meios de comunicação corporativa está intrinsecamente relacionada à facilidade de uso, satisfação do usuário, aprendizagem, capacidade de memorização, isto é, à usabilidade de sua interface. Baseado nisso, foi feito o re-design das páginas de tal maneira que facilitasse a utilização do Portal pelo usuário.

A falta de uso de padrões, inconsistências de *hiperlinks* e redundância de páginas foram os fatores motivadores que levaram a adoção de um novo modelo para as páginas que compõem o Portal. O modelo aplicado no *design* do Portal Corporativo foi a utilização de quadros (*frames*), este modelo tem como característica a divisão da página em quadros independentes, garantindo assim menor redundância de páginas e maior velocidade ao carrega-las, diminuindo, assim o tráfego de dados na rede. Uma vez que as operações efetuadas afetarão somente algum(s) quadro(s) da página, ou seja, não será necessário carregar novamente a página toda, mas apenas porções dela.

As cores presentes no *design* foram escolhidas de modo a manter o padrão com o Portal Corporativo da Polícia Militar de Belo Horizonte e as fontes dos textos foi cuidadosamente selecionadas de modo a facilitar a visualização pelo usuário.

Um exemplo de uma página desenvolvida com a utilização de quadros pode ser visualizado na figura 6.5.

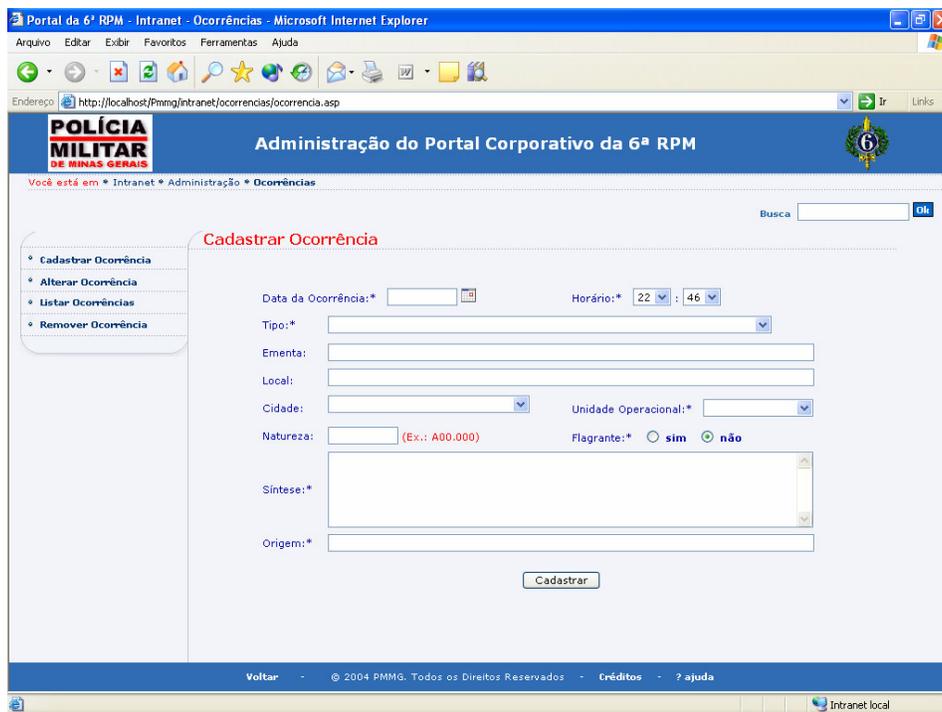


Figura 6.5 – Página para realizar cadastro/manutenção de Ocorrências.

Como pode ser observado na figura 6.5, a página mostra-se dividida em cinco regiões. Na região superior da página encontra-se o cabeçalho da página, que contém o logotipo da Polícia Militar de Minas Gerais e da 6ª RPM. A região central (conteúdo da página) é dividida em três porções: superior, esquerda e direita. Na porção superior do conteúdo da página encontra-se um conjunto de *hiperlinks* dispostos de forma hierárquica possibilitando ao usuário o retorno para páginas de níveis superiores a página corrente. Por exemplo: a página da Intranet do portal é um nível superior à página de Administração do portal e esta é superior a de Administração de Ocorrências. A porção esquerda do conteúdo da página apresenta um *menu* de opções de possíveis operações sobre o cadastro de ocorrência, ou seja, os *hiperlinks* do *menu* alteram apenas o quadro a direita.

Na porção direita do conteúdo da página, são visualizadas as informações referentes a cada operação escolhida no *menu* de opções. Por fim, na região inferior da página, encontra-se o rodapé da página contendo *hiperlinks* que levarão a página anterior à página corrente, bem como as páginas de ajuda e créditos do Portal Corporativo da 6ª RPM.

6.4 Nova Funcionalidade

Neste projeto de pesquisa, a funcionalidade oferecida pelo Portal foi definida a partir dos seus objetivos e do estudo das tarefas que os usuários realizam através dele. Desta forma, o objetivo central da manutenção e documentação do Portal Corporativo da 6ª RPM pode ser descrito como: criar mecanismos para fornecer maior dinamismo às informações presentes no Portal, incluir funções para tratar as novas informações disponibilizadas pelo produto de software e gerar uma documentação abrangente do Portal.

A nova funcionalidade do Portal permite a manipulação de informações que anteriormente não eram tratadas pelo produto de software. Estas informações foram apontadas pelos usuários como importante para o conhecimento e sua manipulação pelos membros da corporação. Portanto, a nova funcionalidade permite a realização dos seguintes cadastros e/ou manutenção: arquivos para *download* através do Portal, enquetes, *links* úteis, Companhia PM, Pelotão PM, mensagens a serem exibidas no fórum de discussão, mensagens a serem exibidas do mural de recados e organograma estrutural da 6ª RPM.

Para que fosse possível tornar dinâmico o conteúdo das páginas, houve a necessidade de criar uma nova base de dados. Uma vez que o banco de dados

utilizado anteriormente à manutenção apresentava alguns dados redundantes e outros que antes não estavam presentes na base de dados, mas que seriam fundamentais para a realização desta tarefa.

Depois de finalizada esta fase, foram reestruturados os códigos de todas as páginas que compõem o Portal, de forma a fazer com que acessassem a nova base de dados. No momento da fase de reestruturação, foram desenvolvidos mecanismos para permitir o dinamismo do conteúdo das páginas de forma a prever futuras alterações sobre os dados. Um exemplo de mecanismo adotado nas páginas é a atribuição de conteúdo dinâmico a componentes da página denominados *lists* e *menus*, ou seja, os valores que preenchem estes componentes são obtidos através de consultas na base de dados.

Uma importante função do Portal é a possibilidade de alterar a hierarquia das unidades que compõem a Sexta Região da Polícia Militar de Minas Gerais. Esta hierarquia é definida da seguinte forma: no nível mais alto estão as Regiões da Polícia Militar (RPM), logo abaixo as Unidades Operacionais (UEOP) – Batalhões da Polícia Militar e Companhias da Polícia Militar Independentes, em seguida Companhias da Polícia Militar, abaixo os Pelotões da Polícia Militar, Destacamentos e Sub-destacamentos. O quadro 6.1 visa a dar uma melhor compreensão desta estrutura hierárquica, tomando como exemplo a 6ª Região da Polícia Militar, o 8º Batalhão da Polícia Militar, suas Companhias PM, Pelotões PM, Grupos PM (Destacamentos e Sub-Destacamentos), bem como suas respectivas cidades sede representadas entre parênteses.

Quadro 6.1: Representação da Hierarquia da 6ª RPM

| RPM | UEOP | COMPANHIA PM | PELOTÃO PM | GRUPO PM |
|-------------------|--------------------|---|--|--|
| 6ªRPM (Lavras) | 8º BPM (Lavras) | 8ª Cia PM Espz Rv e MAmb (Lavras) | 1º Pel PM Espz Rv (Lavras) | Lavras Oliveira |
| | | | 2º Pel PM Espz MAmb (Lavras) | Lavras Aiuruoca Andrelândia Campo Belo Oliveira |
| | | 54ª Cia PM (Lavras) | (04 Pel PM) (01 Pel PChq) (Lavras) | Lavras |
| | | 59ª Cia PM (Oliveira) | 1º Pel PM (Oliveira) | Oliveira <i>Morro do Ferro</i> São Francisco de Paula Carmo da Mata |
| | | | 2º Pel PM (Bom Sucesso) | Bom Sucesso Ibituruna |
| | | | 3º Pel PM (Carmópolis de Minas) | Carmópolis de Minas Passa Tempo Piracema |
| | | 106ª Cia Ens/Trein (Lavras) | | |
| | | 112ª Cia PM (Perdões) | 1º Pel PM (Perdões) | Perdões Ribeirão Vermelho Cana Verde Santo Antônio do Amparo |
| | | | 2º Pel PM (Nepomuceno) | Nepomuceno |
| | | | 3º Pel PM (Itumirim) | Itumirim <i>Macuco de Minas</i> Carrancas Ingaí Itutinga Luminárias |

Quadro 6.1: Representação da Hierarquia da 6ª RPM (continuação)

| RPM | UEOP | COMPANHIA PM | PELOTÃO PM | GRUPO PM |
|-------------------|--------------------|------------------------------|----------------------------|--|
| 6ªRPM (Lavras) | 8º BPM (Lavras) | 112ª Cia PM (Perdões) | 4º Pel PM (Ijaci) | Ijaci |
| | | 140ª Cia PM (Andrelândia) | 1º Pel PM (Andrelândia) | Andrelândia Arantina Minduri São Vicente de Minas |
| | | | 2º Pel PM (Aiuruoca) | Aiuruoca Bocaina de Minas Carvalhos Liberdade Passa Vinte Seritinga Serranos |
| | | 161ª Cia PM (Campo Belo) | 1º Pel PM (Campo Belo) | Campo Belo |
| | | | 2º Pel PM (Campo Belo) | Campo Belo Aguanil Cristais Santana do Jacaré |
| | | | 3º Pel PM (Candeias) | Candeias |

Devido a esta estrutura ser passível de alterações ao longo do tempo, ou seja, uma Companhia PM pode vir a tornar-se uma Unidade Operacional e, por sua vez, conter suas próprias Companhias PM, Pelotões PM e Grupos PM que antes eram subordinados a outra Unidade Operacional, foi necessária a criação de mecanismos que tratassem estas possíveis modificações.

A partir disto, será possível, após uma mudança da hierarquia, cadastrar novas informações seguindo a atual disposição da estrutura assim como manter os dados cadastrados anteriormente.

6.5 Modelagem do Produto de Software

Nas seções a seguir, é apresentada a modelagem do Portal Corporativo da 6ª RPM. Como descrito no Capítulo 1, Seção 3.1 (Tecnologias Utilizadas), foi utilizado o perfil de modelagem para aplicações *Web* UWE (*UML-based Web Engineering*). Este procedimento oferece vários recursos para a modelagem de aplicações *Web*, bem como técnicas de modelagem visual e o desenvolvimento pode ser descrito como um processo iterativo.

6.5.1 Modelo de Casos de Uso

A partir da aplicação de engenharia reversa apresentada no Capítulo 3 (Engenharia Reversa de Software), foi possível recuperar as informações do projeto e documentar o real estado do Portal, através da construção do modelo de casos de uso (*use cases*) do produto de software.

Neste trabalho, foi adotado um padrão para este modelo o qual é representado apenas por um caso de uso que abrange cada subsistema do Portal. Portanto, cada caso de uso abrangerá informações sobre todas as operações que podem ser realizadas em cada subsistema, estas operações são: cadastrar novas informações, alterar os dados, listar as informações cadastradas e removê-las. Este padrão foi adotado devido ao grande número de casos de usos existentes e para que pudesse obter um conhecimento geral do produto de software, porém de forma resumida.

Os subsistemas presentes no Portal podem ser visualizados na figura 6.6. Este diagrama apresenta todos os subsistemas do produto de software, bem como a relação entre eles.

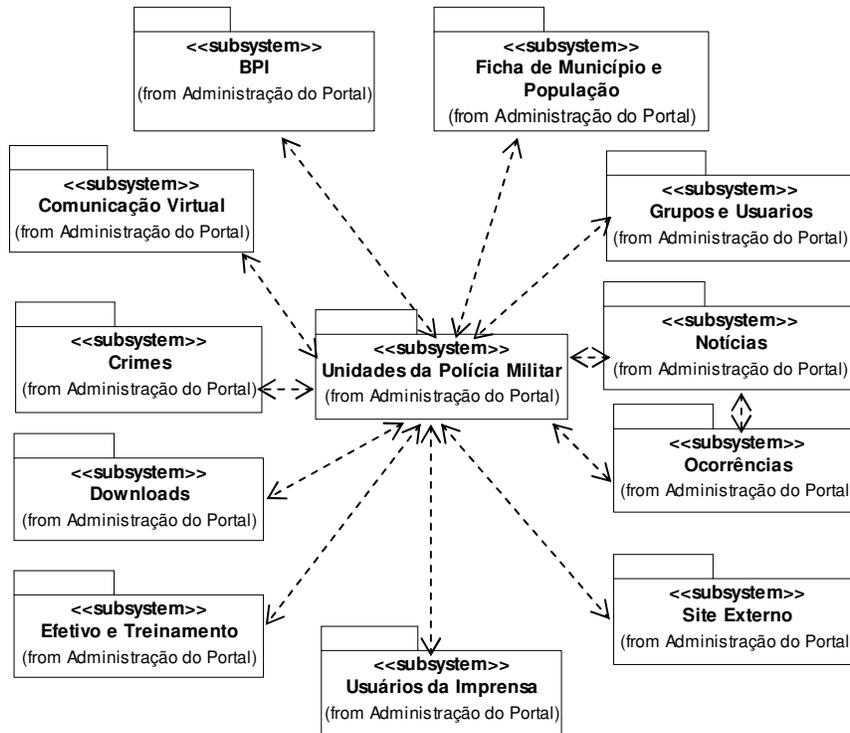


Figura 6.6: Diagrama dos subsistemas do Portal Corporativo da 6ª RPM

Os diagramas de caso de uso do Portal Corporativo são apresentados nas figuras a seguir. A descrição de cada caso de uso pode ser encontrada no apêndice A.

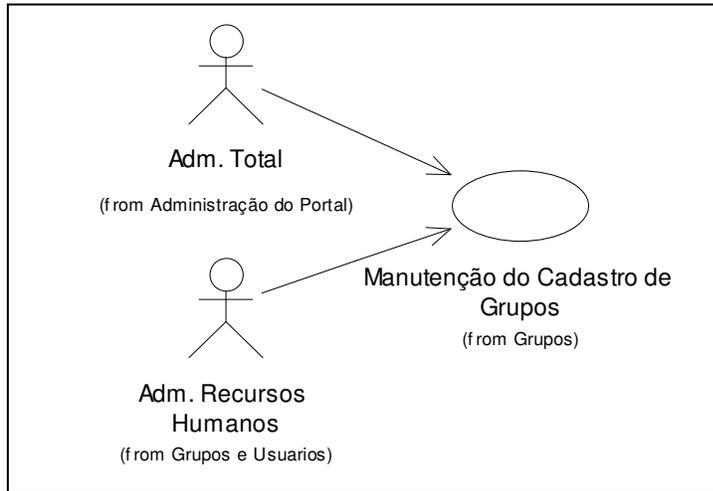


Figura 6.7: Diagrama de caso de uso do subsistema Grupos

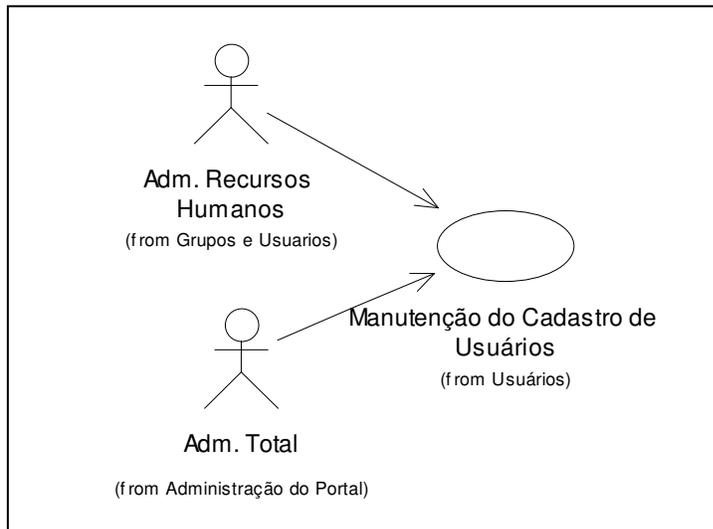


Figura 6.8: Diagrama de caso de uso do subsistema Usuários

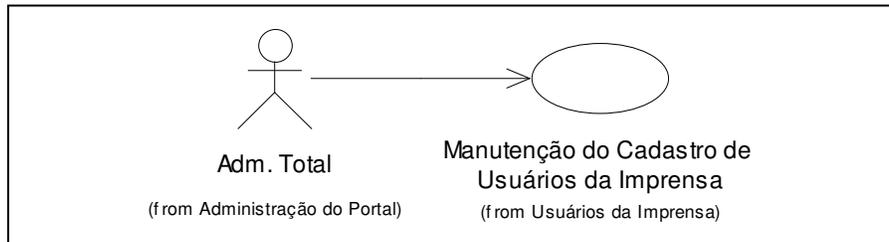


Figura 6.9: Diagrama de caso de uso do subsistema Usuários da Imprensa

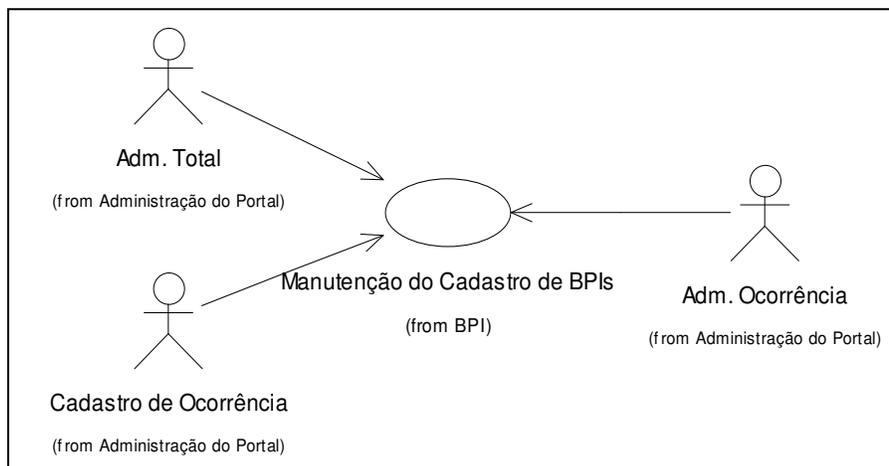


Figura 6.10: Diagrama de caso de uso do subsistema BPIs

Na figura 6.10 os atores Cadastro de Ocorrência e Adm. Ocorrência são responsáveis apenas pela realização de cadastro e/ou manutenção de BPIs de sua Unidade Operacional.

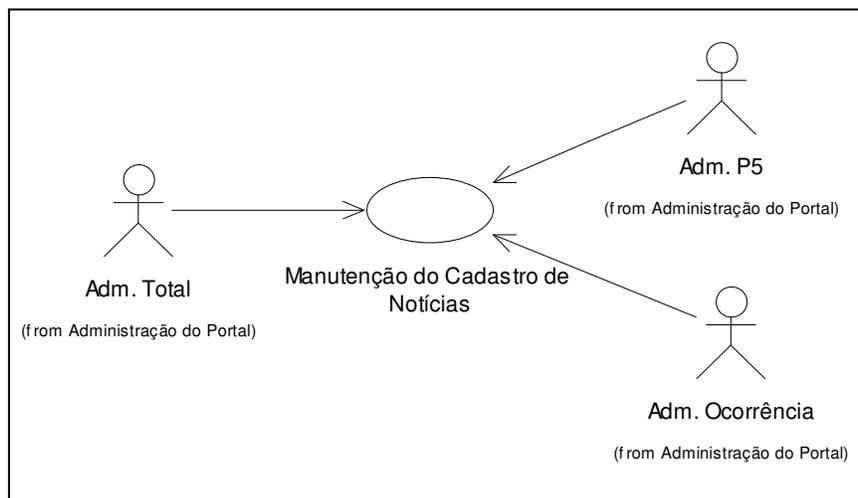


Figura 6.11: Diagrama de caso de uso do subsistema Notícias

Na figura 6.11, os atores Adm. P5 e Adm. Ocorrência são responsáveis apenas pela realização de cadastro e/ou manutenção de Notícias de sua Unidade Operacional.

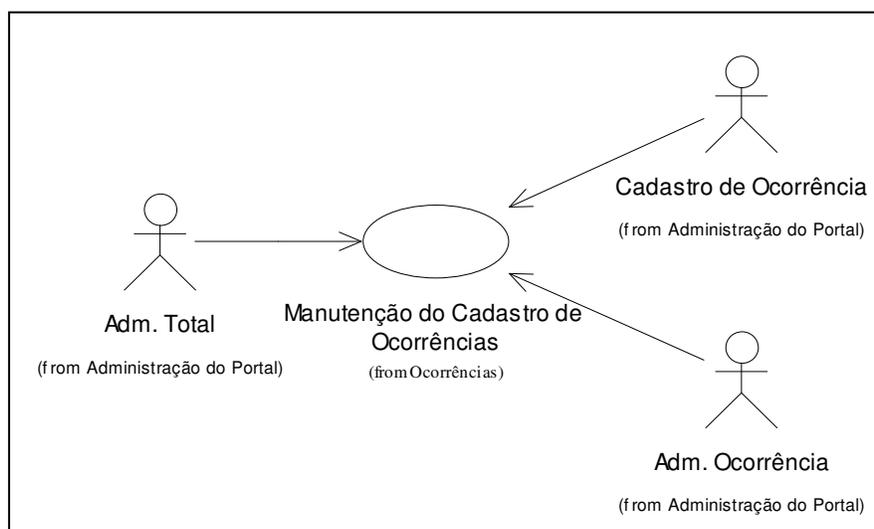


Figura 6.12: Diagrama de caso de uso do subsistema Ocorrências

Na figura 6.12 os atores Cadastro Ocorrência e Adm. Ocorrência são responsáveis apenas pela realização de cadastro e/ou manutenção de Ocorrências de sua Unidade Operacional.

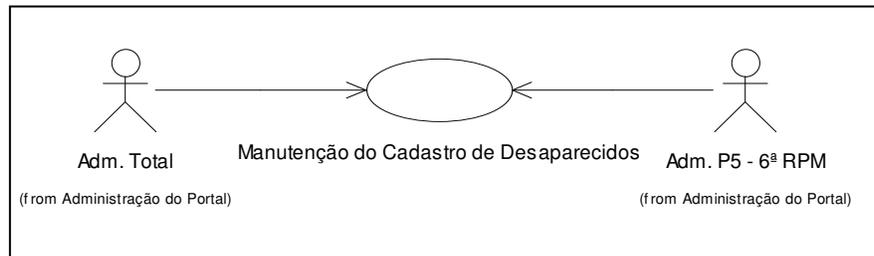


Figura 6.13: Diagrama de caso de uso do subsistema Desaparecidos

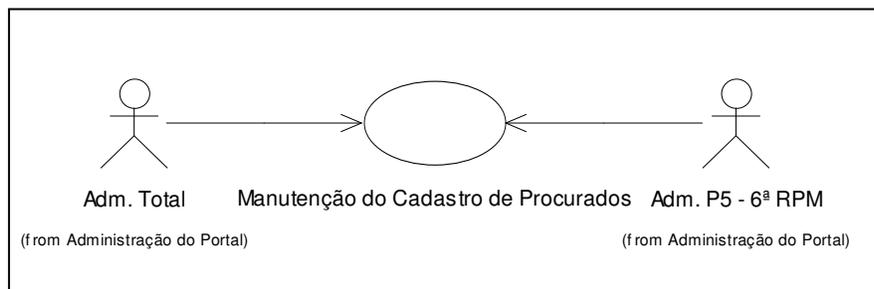


Figura 6.14: Diagrama de caso de uso do subsistema Procurados

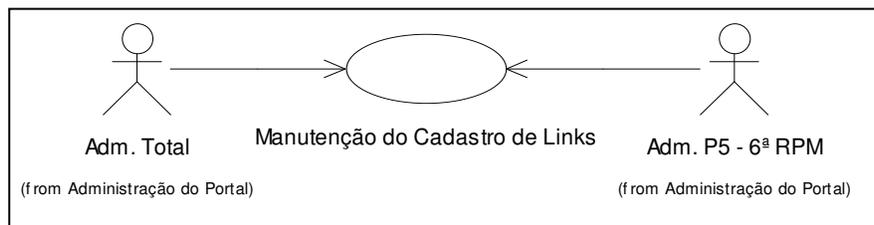


Figura 6.15: Diagrama de caso de uso do subsistema Links

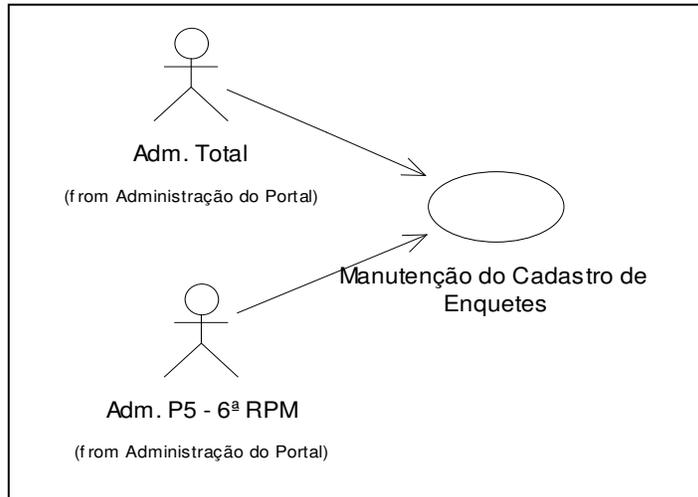


Figura 6.16: Diagrama de caso de uso do subsistema Enquetes

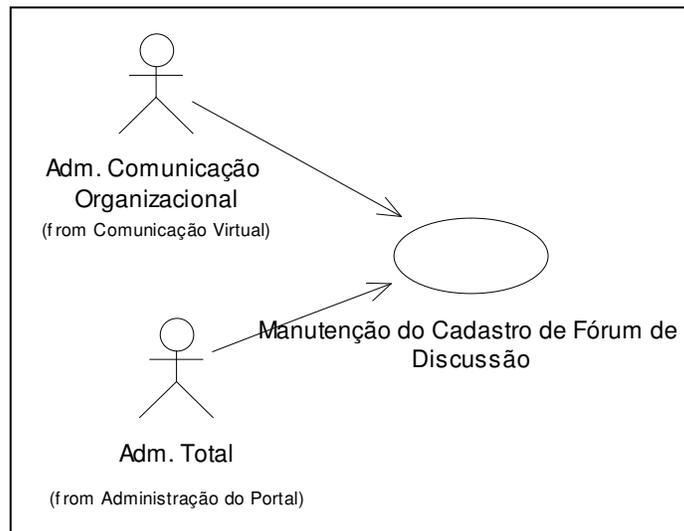


Figura 6.17: Diagrama de caso de uso do subsistema Fórum de Discussão

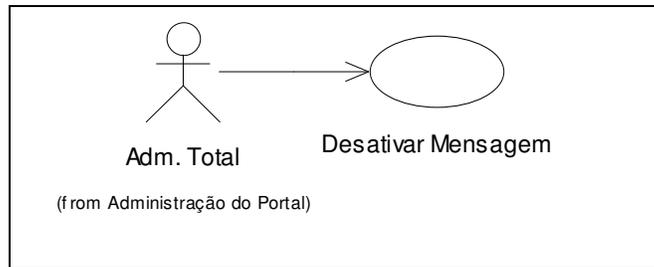


Figura 6.18: Diagrama de caso de uso do subsistema Mural de Recados

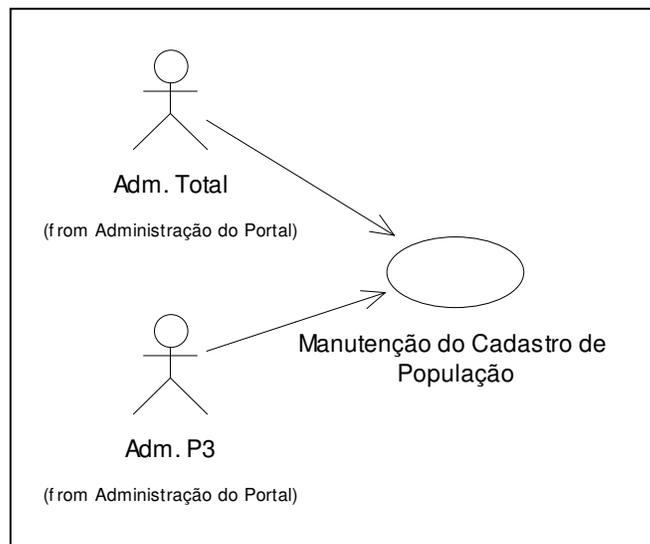


Figura 6.19: Diagrama de caso de uso do subsistema População

Na figura 6.19 o ator Adm. P3 é responsável apenas pela realização de cadastro e/ou manutenção de População de sua Unidade Operacional.

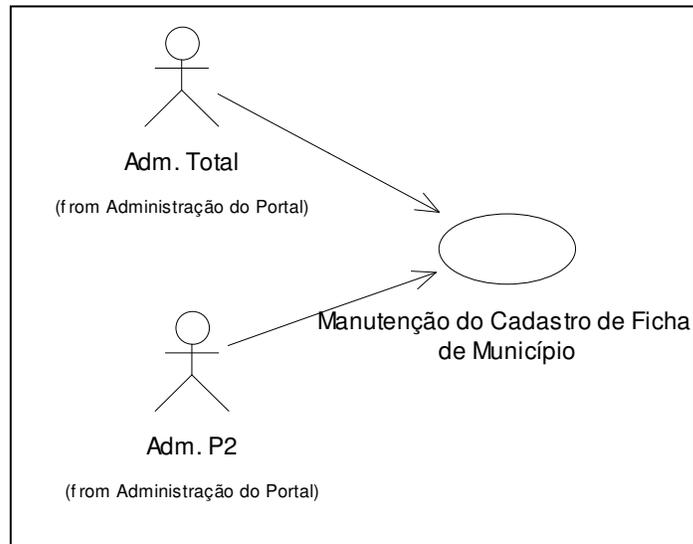


Figura 6.20: Diagrama de caso de uso do subsistema Ficha de Município

Na figura 6.20 o ator Adm. P2 são responsáveis apenas pela realização de cadastro e/ou manutenção de Ficha de Município de sua Unidade Operacional.

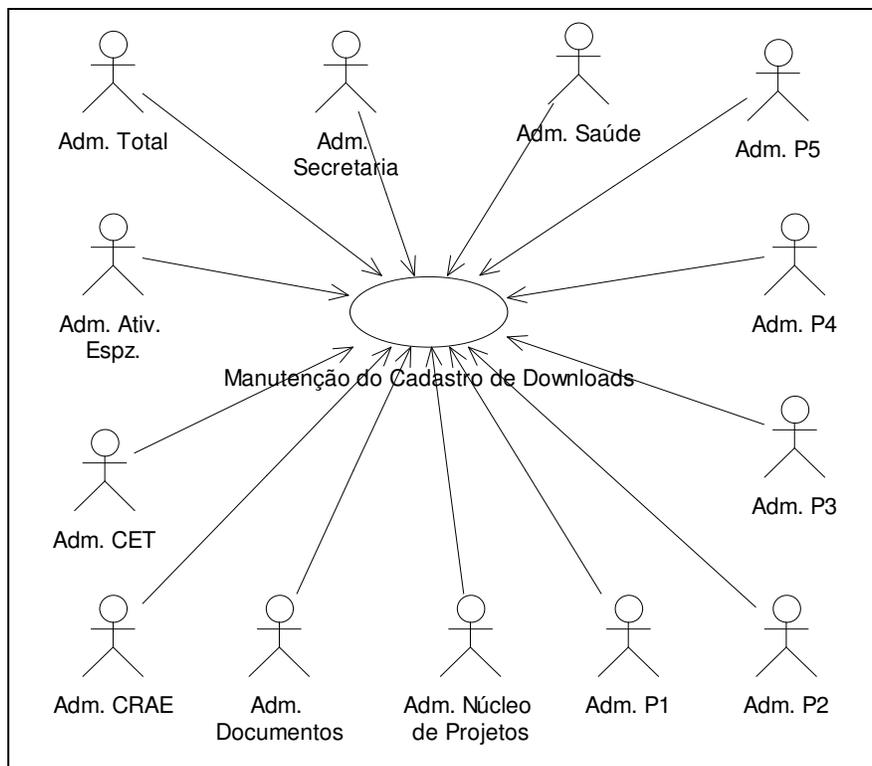


Figura 6.21: Diagrama de caso de uso do subsistema *Download*

Na figura 6.21 os atores Adm. Núcleo de Projetos e Adm. Saúde apenas tem permissão para efetuar o cadastro e/ou manutenção de arquivos para *download* da 6ª RPM, os atores Adm. Ativ. Espz. e Adm. CET têm permissão para efetuar o cadastro e/ou manutenção de arquivos para *download* de seu Batalhão de Polícia Militar ou Companhia de Polícia Militar Independente. Os demais atores são responsáveis pela realização de cadastro e/ou manutenção de Ficha de Município de sua Unidade Operacional.

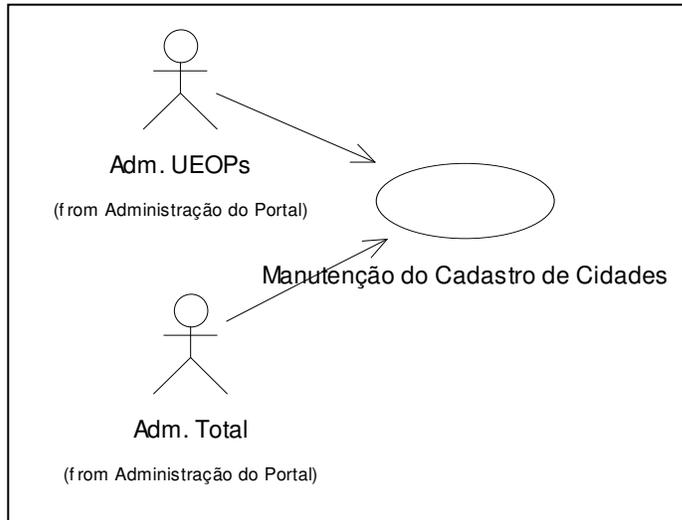


Figura 6.22: Diagrama de caso de uso do subsistema Cidades

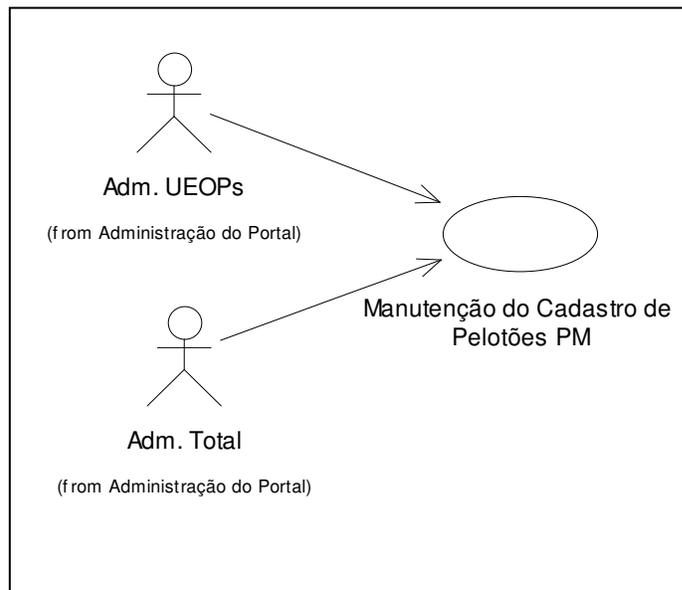


Figura 6.23: Diagrama de caso de uso do subsistema Pelotões PM

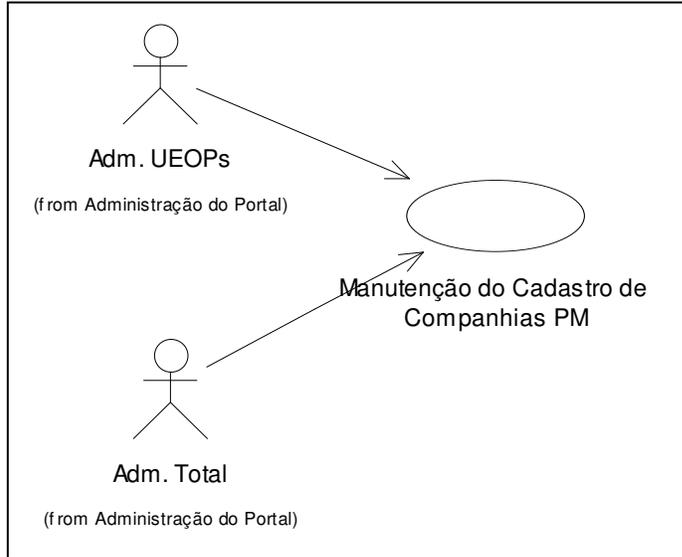


Figura 6.24: Diagrama de caso de uso do subsistema Companhias PM

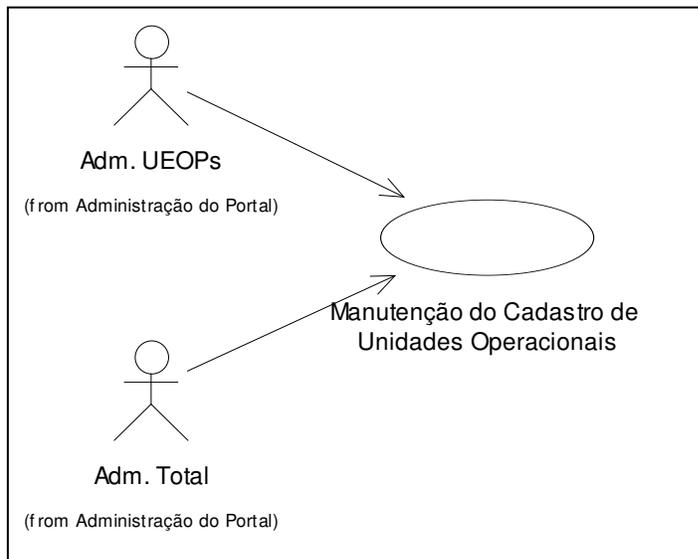


Figura 6.25: Diagrama de caso de uso do subsistema Unidades Operacionais

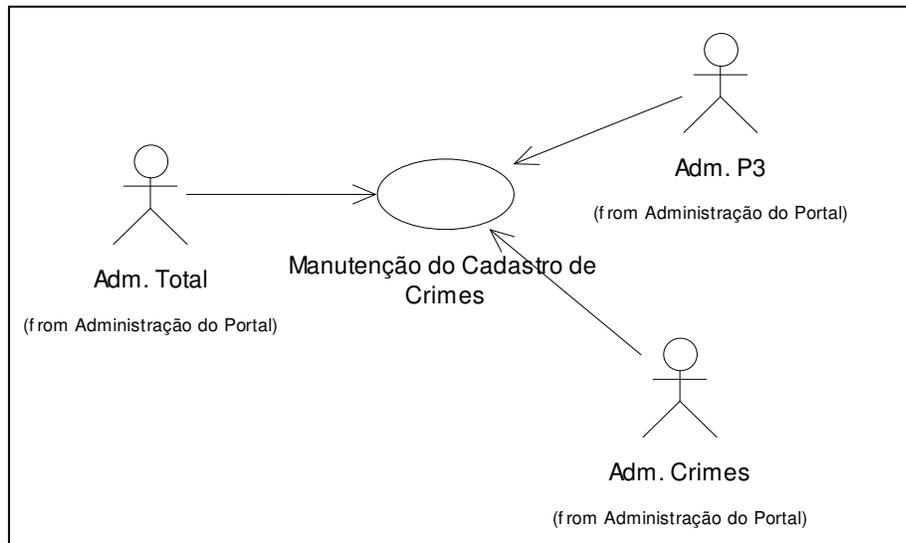


Figura 6.26: Diagrama de caso de uso do subsistema Crimes

Na figura 6.26 o ator Adm. Crimes é responsável apenas por realizar o cadastro e/ou manutenção de Crimes de sua cidade, enquanto é permitido ao ator Adm. P3 a realização de cadastro e/ou manutenção de Crimes de toda a sua Unidade Operacional.

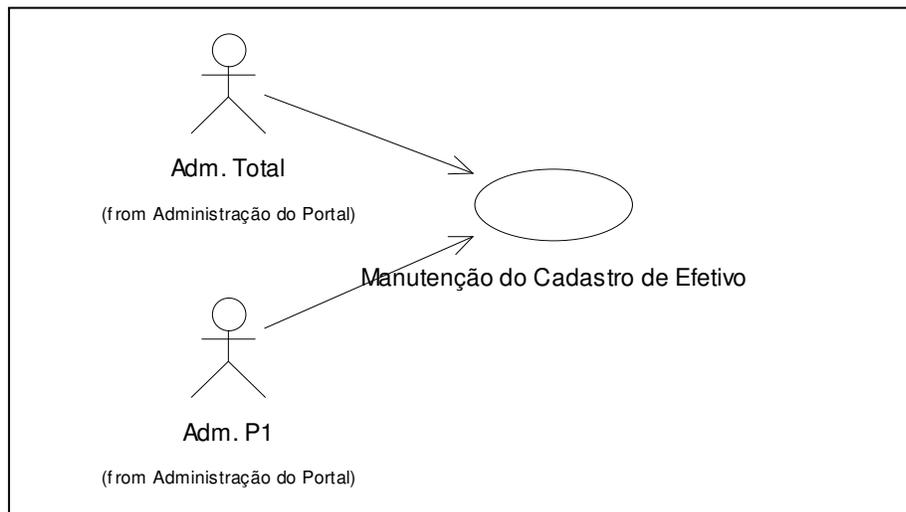


Figura 6.27: Diagrama de caso de uso do subsistema Efetivo

Na figura 6.27 o ator Adm. P1 são responsáveis apenas pela realização de cadastro e/ou manutenção de Efetivo de sua Unidade Operacional.

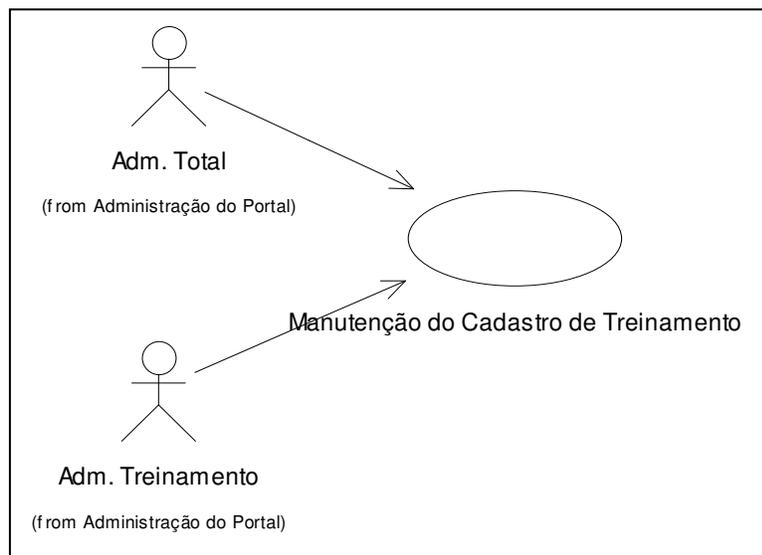


Figura 6.28: Diagrama de caso de uso do subsistema Treinamento

Na figura 6.28 o ator Adm. Treinamento é responsável apenas pela realização de cadastro e/ou manutenção de Efetivo de sua Unidade Operacional.

6.5.2 Modelo Conceitual

De acordo com [KOCH, 2001], os principais elementos utilizados no modelo conceitual são as classes e as associações entre elas. Graficamente, as classes são representadas como na notação UML, ou seja, são descritas por um nome, atributos e métodos. Segundo [KOCH, 2001], os métodos podem ser omitidos do modelo conceitual, pois eles contêm informações adicionais usadas pelo funcionamento do conteúdo adaptativo da aplicação, isto é, para apresentar conteúdos adicionais ao usuário de acordo com o estado corrente da aplicação. As

classes definidas nesta etapa são fundamentais para a construção do modelo de navegação.

O Portal Corporativo da 6ª RPM possui inúmeras classes cada qual pertencendo a um subsistema. Um exemplo de modelo conceitual do subsistema Unidades da Polícia Militar e dentro deste os subsistemas cadastro de Unidades Operacionais, Companhias PM, Pelotões PM e Cidades é apresentado na figura 6.29.

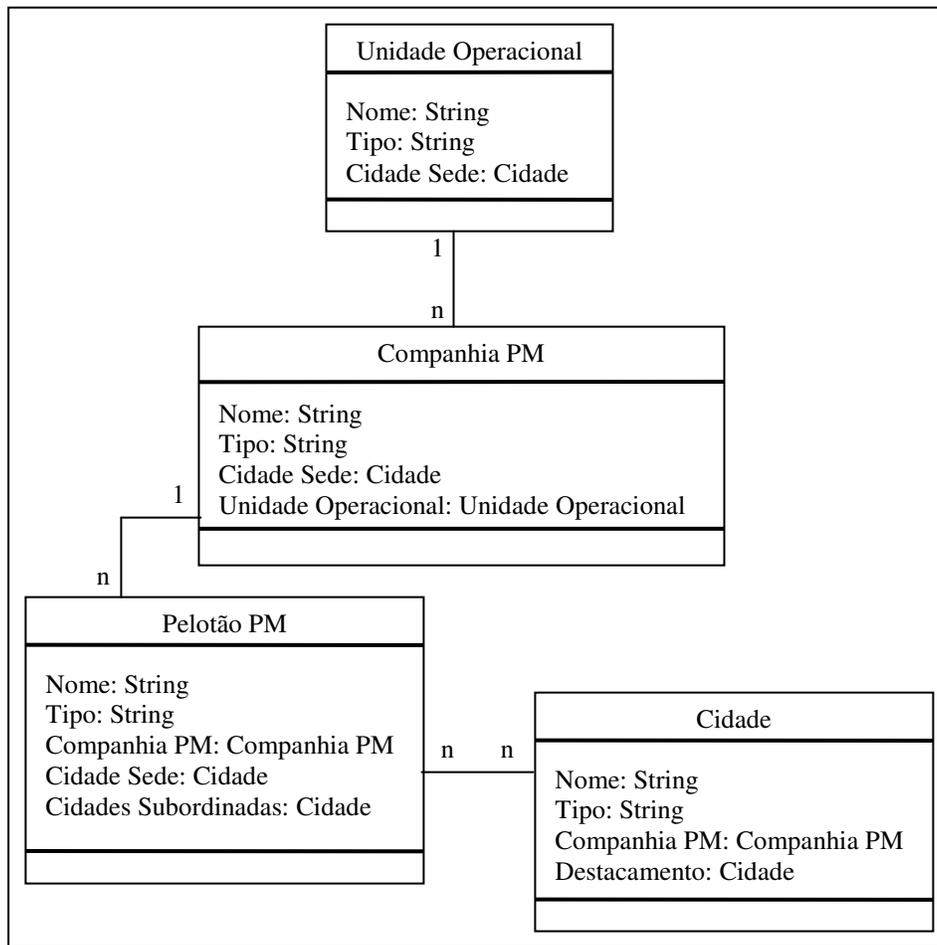


Figura 6.29: Modelo Conceitual do subsistema Unidades da Polícia Militar

6.5.3 Modelo de Navegação

Segundo [KOCH, 2002], o modelo de navegação de aplicações *Web* compreende a construção de dois modelos de navegação: o espaço do modelo de navegação e a estrutura do modelo de navegação.

Um exemplo de espaço do modelo de navegação, apresentado na Figura 6.30, inclui as classes do subsistema Unidades da Polícia Militar e os objetos que podem ser visitados durante a navegação. Os principais elementos para este modelo são: as páginas principais de cada seção e o módulo do sistema, chamada de classes de navegação e a direção da navegabilidade, ou seja, a direção do caminho das classes de navegação para as demais classes do subsistema, sem incluir páginas adicionais. No caso, as classes do subsistema Unidades da Polícia Militar podem ser alcançadas a partir da página principal do Portal (seção Internet), depois acessando a página da seção Intranet, e por fim através da página do módulo Administração do Portal.

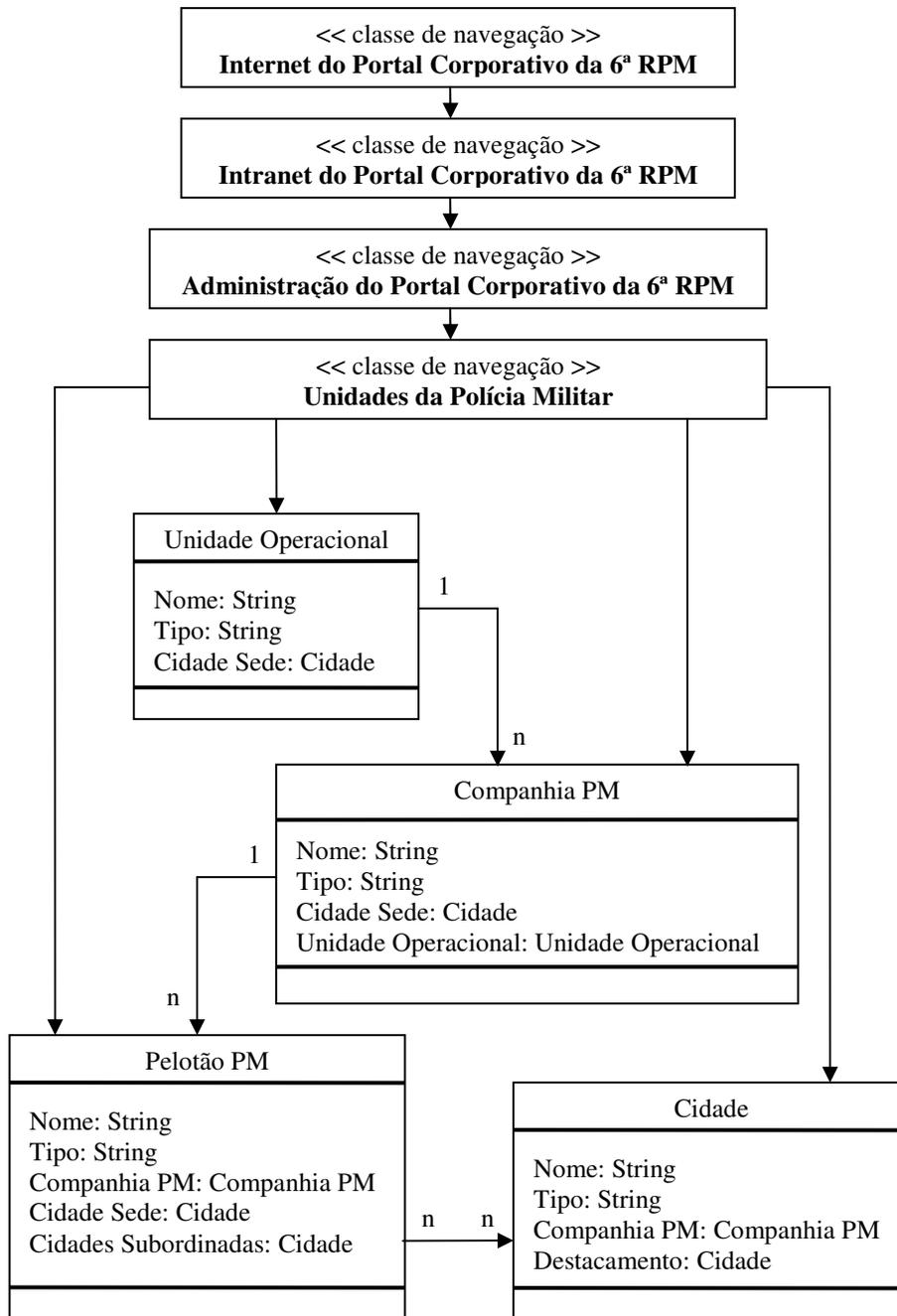


Figura 6.30: Espaço do Modelo de Navegação do subsistema Unidades da Polícia Militar

A estrutura do modelo de navegação é construída com base no espaço de navegação. São adicionados os caminhos secundários existentes entre a classe de navegação e as classes do Portal. Ou seja, são adicionadas as telas secundárias da aplicação, incluindo os acessos primitivos (*menus*, índices, consultas, etc.) que definem o tipo de acesso e as funções das telas.

A Figura 6.31 apresenta a estrutura do modelo de navegação com os acessos primitivos e as telas secundárias do subsistema Unidades da Polícia Militar apresentando apenas o cadastro de Unidades Operacionais por questões de visualização. É possível perceber que a tela principal da administração do Portal Corporativo da 6ª RPM possui um *menu* com todos os subsistemas, estes são: BPI, Comunicação Virtual, Crimes, *Downloads*, Efetivo e Treinamento, Ficha de Município e População, Grupos e Usuários, Imprensa, Notícias, Ocorrências, Site Externo e Unidades da Polícia Militar. A partir do *menu* do subsistema Unidades da Polícia Militar, é obtida uma página com um novo *menu* com as opções para realizar a manutenção do cadastro de Unidades Operacionais, Companhias PM, Pelotões PM e Cidades. A partir do *menu* Unidades Operacionais é possível efetuar o cadastro, alterações, listagem e remoção de Unidades Operacionais.

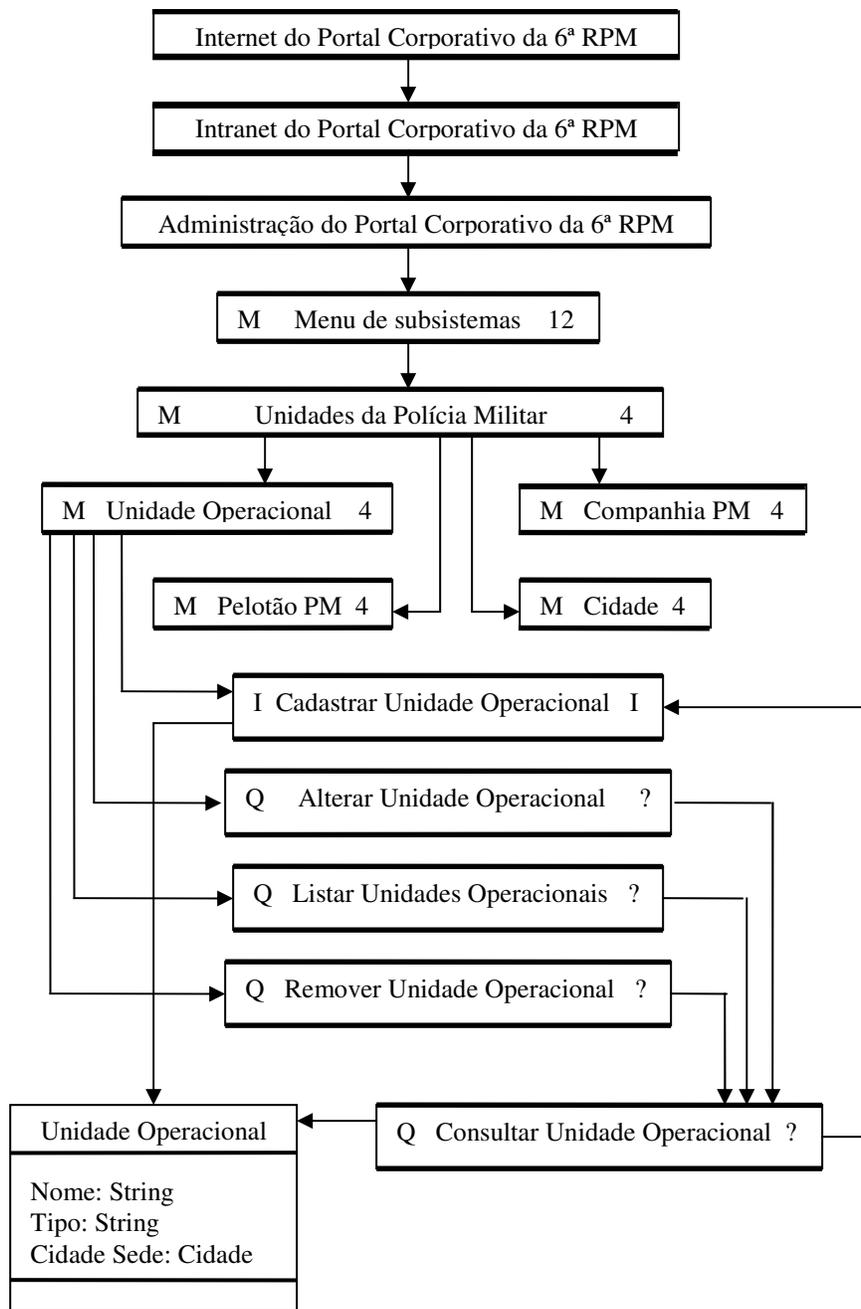


Figura 6.31: Estrutura do Modelo de Navegação com acessos primitivos do subsistema Unidades da Polícia Militar (Cadastro de Unidades Operacionais)

6.5.4 Modelo de Apresentação

Depois de adicionar os acessos primitivos e organizar as associações no modelo de navegação, é preciso criar o modelo de apresentação. Nesta modelagem, foi criada uma classe de aplicação para cada elemento de navegação apresentado no modelo navegacional e conectada as classes adjacentes por composição. O diagrama de apresentação do subsistema Unidades da Polícia Militar é mostrado na Figura 6.32. Segundo [KOCH, 2002], este diagrama é semelhante à interface vista pelo usuário. Além disso, não é necessário colocar em uma classe de apresentação os atributos e as operações desta classe. A letra “P” ao lado de cada componente representa uma classe de apresentação.

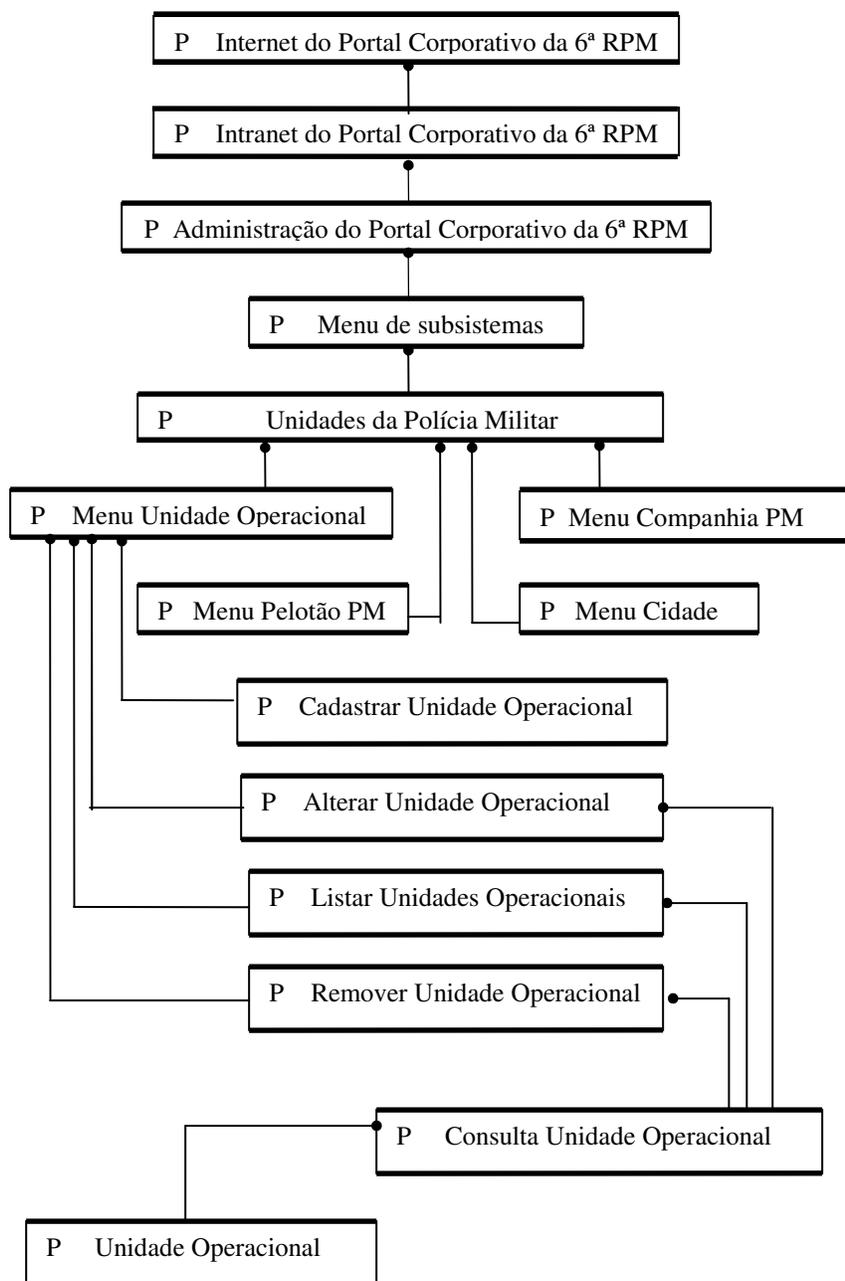


Figura 6.32: Modelo de Apresentação do subsistema Unidades da Polícia Militar (Cadastro de Unidades Operacionais)

6.6 Resumo

Neste capítulo foram apresentadas a nova funcionalidade do Portal Corporativo da 6ª RPM e sua modelagem, explicitando os casos de uso do produto de software a fim de permitir uma visão geral de todo o Portal. Além disso, foi demonstrada a estrutura organizacional da Polícia Militar de Minas Gerais, sobre a qual todo o produto de software foi baseado durante seu desenvolvimento.

7. CONCLUSÃO E TRABALHOS FUTUROS

Este trabalho forneceu ao seu término a documentação e a modelagem do produto de software Portal Corporativo da 6ª RPM, bem como proporcionou uma melhoria na sua estrutura uma vez que desenvolveu métodos que proporcionaram melhor usabilidade, além de criar funções que manipulassem novas informações que anteriormente não eram tratadas e disponibilizadas pelo Portal.

Algumas propostas futuras são sugeridas para que o Portal possa trazer maiores contribuições aos seus usuários e à corporação da Polícia Militar de Minas Gerais.

7.1 Conclusão

Um forte motivo para a mudança em um produto de software se refere às suas falhas de usabilidade e falta de funcionalidade que trate todas as informações importantes para a organização. Essa mudança pode ser realizada através da reengenharia de software e, para o sucesso dessa reengenharia, esta deve ser precedida de um planejamento [SILVA & DÍSCOLA, 2003].

Segundo o mesmo autor, o planejamento da reengenharia deve estabelecer justificativas e objetivos para a mudança no produto de software, determinar as atividades necessárias na reengenharia e estudar a sua viabilidade como forma de mudança do produto de software existente em uma organização. Nota-se que os processos de planejamento de reengenharia existentes

mencionam a necessidade de conhecer os objetivos e causas para a mudança, e os desejos e objetivos dos usuários para responder perguntas sobre quais requisitos estão com problema, quais mudanças devem ser realizadas e o que os usuários esperam do produto de software.

A manutenção realizada no produto de software Portal Corporativo da 6ª RPM forneceu nova funcionalidade e o reestruturou de forma a atender aos requisitos levantados durante o planejamento da reengenharia. Através deste trabalho foi possível proporcionar um avanço na comunicação dentro da corporação, a fim de afetar positivamente os processos de geração, difusão e armazenamento de conhecimento da Polícia Militar de Minas Gerais.

A partir da aplicação da engenharia reversa, foi possível recuperar informações de projeto perdidas durante a fase de desenvolvimento e documentar o real estado do produto de software. A documentação gerada irá auxiliar futuros processos de manutenção, além de ficar mais simples continuar a sua construção.

7.2 Contribuições

Os portais corporativos visam gerar o fluxo de conhecimento e informação significativa e personalizada dentro dos limites da organização, mudando completamente a forma como informações, atividades e responsabilidades são compartilhadas em um ambiente organizacional. Ao mesmo tempo, buscam gerar conexões entre pessoas e entre pessoas e informação, assim fomentam a criação de conhecimento, inovação e reutilização de conhecimento explicitado, e a localização de pessoas que podem aplicar seu conhecimento tácito em situações específicas de negócios [BARBOSA, 2000].

Sob esta ótica este trabalho proporcionou a reestruturação e a manutenção de um produto de software, Portal Corporativo da 6ª RPM, de forma a criar uma rede de inteligência com o objetivo de prover de forma fácil e rápida o acesso a dados, informações e conhecimento relevantes para a realização das funções presentes na organização e ainda criar melhores meios de comunicação interconectando todos os seus membros. É esperado que o Portal Corporativo da 6ª RPM fomente a criação, o gerenciamento e o compartilhamento de conhecimento, tanto individual como coletivo, incentivando os membros da organização a aprender, a criar e a inovar cada vez mais.

7.3 Trabalhos Futuros

As sugestões para pesquisas e implementações futuras se baseiam em estudos sobre a integração de conhecimentos e informações de todas as Regiões que compõem a Polícia Militar de Minas Gerais, a fim de que experiências possam ser trocadas entre seus membros assim como dados importantes possam ser levantados sobre a corporação.

É sugerido também que a gerência do Portal realize treinamentos e palestras de conscientização para os integrantes da corporação, pois o aumento de produtividade está intrinsecamente ligado ao nível de experiência e qualificação de seus membros.

Seria de grande valia também que fossem criados mecanismos geradores de relatórios das informações contidas no produto de software, possibilitando a realização de levantamentos e análises dos dados cadastrados, fornecendo a corporação fontes de conhecimento para a tomada de decisões.

Ao se pensar em deficiências estruturais e de hardware, foi possível observar e prever que uma rede de computadores ineficiente pode ser um dos

maiores problemas para o sucesso da implantação do portal, logo é extremamente recomendável um estudo que aperfeiçoe as tecnologias de rede e de hardware presentes.

REFERÊNCIAS BIBLIOGRÁFICAS

- BARBOSA, L. **Gestão do conhecimento é o grande desafio**, 2000.
- BENEDUSI, P. & CIMITILE, A. & CARLINI, U. **Reverse Engineering Processes, Design Document Production, and Structure Charts**. *Journal Systems and Software*, v. 19, 1992, p. 225-245.
- BENNETT, K. H. **Automated Support of Software Maintenance. Information and Software Technology**, v. 33, n.1, 1991, p. 74-85.
- BENNETT, K. H. & HINLEY, D. S. **Using a model to manage the software maintenance process**. In *Proceedings of the Conference on Software Maintenance*, 1992, pp. 174-182, Flórida, 1992.
- BENNETT, K. H. **Legacy Systems: Coping with success**. *IEEE Software*, 12(1), pp. 19-23, 1995.
- BIGGERSTAFF, T. J. **Design Recovery for Maintenance and Reuse**. *IEEE Computer*, v. 22, n. 7, 1989, p. 36-49.
- BOOCH, G.; Rumbaugh, J. & Jacobson, I. **The Unified Modeling Language User Guide**, ADDISON-WESLEY, 1998.
- CANFORA, G.; CIMITILE, A.; MUNRO, M. **RE2: Reverse-engineering and Reuse Reengineering**. *Journal of Software Maintenance: Research and Practice*, v. 6, n. 2, 1994, p. 53-72.
- CAPRETZ, Luiz F. & CAPRETZ, Miriam A. M. **Object-Oriented Software: Design and Maintenance**. University of Pittsburgh, USA. Series Editor S K Chang: *Series on Software Engineering and Knowledge Engineering*, Vol. 6, 1996.
- CHIKOFSKY, E. J. & CROSS II, J. H. **Reverse Engineering and Design Recovery: A Taxonomy**. *IEEE Software*, v. 7, n. 1, 1990, p. 13-17.
- CHIN, D. N. & QUILICI, A. **DECODE: A Co-operative Program Understanding Environment**. *Journal of Software Maintenance: Research and Practice*, v. 8, n. 1, 1996, p. 3-33.

- COSTA, R. M. **Aplicação do Método de Engenharia Reversa FUSION-RE/ na Recuperação da Funcionalidade da Ferramenta de Teste PROTEUM.** São Carlos: ICMSC-USP, 1997. Dissertação (mestrado).
- COUSIN, L. & COLLOFELLO, J. S. **A task-based approach to improving the software maintenance process.** International Conference on Software Maintenance (ICSM) – IEEE Orlando, Florida Nov., 9-12, 1992.
- GALL, H. & KLOSCH, R. **Finding Objects in Procedural Programs: An alternative Approach.** In: Proceedings of Second Working Conference on Reverse Engineering. Monterey, EUA: 1995, p. 208-216.
- GARLAND, J. & CALLISS, F., **Improved change tracking for software maintenance.** International Conference on Software Maintenance (ICSM) – IEEE Sorrento, Italy Oct., 15-17, 1991.
- GT-REG (Georgia Tech's - Reverse Engineering Group). **Glossary of Reengineering Terms.** Georgia, 1998. Disponível em: <http://www.cc.gatech.edu/reverse/glossary.html>. Acesso em 15 de junho de 2004.
- HAMMER, M. & CHAMPY, J. **Reengenharia: revolucionando a empresa em função dos clientes, da concorrência e das grandes mudanças da gerencia.** Rio de Janeiro: Campus, 1994.
- HARANDI, M. T. & NING, J. Q. **Knowledge-Based Program Analysis.** IEEE Software v. 7, n. 1, 1990, p. 74-81.
- IEEE CS-TCSE (IEEE Computer Society - Technical Council on Software Engineering). **Reengineering & Reverse Engineering Terminology.** Washington, 1997. Disponível em: <http://www.tcse.org/revengt/taxonomy.html>. Acesso em 15 de junho de 2004.
- IEEE Std 610.12-1990. **IEEE Standard Glossary of Software Engineering Terminology.** IEEE, 1994.
- JACOBSON, I. & LINDSTRÖM, F. **Reengineering of old systems to an object-oriented architecture.** SIGPLAN Notices, v. 26, n. 11, 1991, p. 340-350.
- JACOBSON, I.; BOOCH, G. & RUMBAUGH, J. **The Unified Software Development Process.** Addison-Wesley, 1999. 463p.

- KOCH, Nora Parcus. **Software Engineering for Adaptive Hypermedia Systems**. München: Ludwig-Maximilians-Universität München, 2001.
- KOCH, Nora Parcus. **CASE Support For Modeling Web Applications**. München: Ludwig-Maximilians-Universität München, 2002.
- KOCH, Nora Parcus e KRAUS, Andreas. **The Expressive Power of UML-based Web Engineering**. München, Germany: Ludwig-Maximilians - Universität, 2002.
- KRÄMER, C.; PRECHELT, L. **Design Recovery by Automated Search for Structural Design Patterns in Object-Oriented Software**. In: Proceedings of Third Working Conference on Reverse Engineering, Monterey, EUA, 1996, p. 208-215.
- LAYZELL, P. J.; FREEMAN, M. J.; BENEDUSI, P. **Improving Reverse-engineering through the Use of Multiple Knowledge Sources**. Journal of Software Maintenance: Research and Practice, v. 7, n. 4, 1995, p. 279-299.
- LEMOS, Renato A. **Estudo de Conceitos e Características do Portal Corporativo da 6ª RPM: Um Estudo Preliminar**. Lavras, UFLA, 2003. Monografia (graduação).
- MORRIS, P.; FILMAN, R. **Mandrake: A Tool for Reverse-Engineering IBM Assembly Code**. In: Proceedings of Third Working Conference on Reverse Engineering, Monterey, EUA, 1996, p. 57-66.
- NEWCOMB, P. e KOTIK G. **Reengineering Procedural Into Object-Oriented Systems**. In: Proceedings of Second Conference on Reverse Engineering, Toronto, Canada, 1995, p. 237-249.
- OMAN, P.W. **Maintenance Tools**. IEEE Software, v. 7, n. 3, 1990, p. 59-65.
- OSBORNE, W. M. & CHIKOFFSKY, E. J. **Fitting Pieces to the Maintenance Puzzle**. IEEE Software, v. 7, n. 1, 1990, p. 11-12.
- PREMERLANI, W. J. & BLAHA, M. R. **An Approach for Reverse Engineering of Relational Databases**. Communications of the ACM, v. 37, n. 5, 1994, p. 42-49.
- PIEKARSKI, A. E. T & QUINÁIA, M. A. **Reengenharia de software: o que, por quê e como**. Departamento de Informática – UNICENTRO. Guarapuava, Paraná, 2000.

- PRESSMAN, Roger. S. **Engenharia de Software**. São Paulo, Makron Books, 1995.
- PRESSMAN, Roger. S. **Engenharia de Software**. São Paulo, Makron Books, 5ª Ed., 2001.
- RAJLICH, V. & SRIDHAR, R., **VIFOR 2: A tool for browsing and documentation**. International Conference on Software Maintenance (ICSM) – IEEE Monterey, California Nov., 4-8, 1996.
- REKOFF Jr., M. G. **On Reverse Engineering**. IEEE Transaction on Systems, Man, and Cybernetics, v. 15, n. 2, março/abril, 1985.
- RAMAMOORTHY, C. V. & TSAI, W. **Advances in Software Engineering**. IEEE Computer, v. 29, n. 10, 1996, p. 47-58.
- RUGABER, S. et al. **Recognizing Design Decision in Programs**. IEEE Software, v. 7, n. 1, 1990, p. 46-54.
- RUGABER, S. **Program Comprehension for Reverse Engineering**. In: AAAI Workshop on AI and Automated Program Understanding, San Jose, California, p. 106-110. July 1992. Disponível em: <http://www.cc.gatech.edu/reverse/papers.html>. Acesso em 15 de junho de 2004.
- SAGE, A. P. **Systems Engineering and Systems Management for Reengineering**. Journal Systems and Software, v. 30, n. 1, 1995, p. 3-25.
- SALEH, K. & BOUJARWAH, A. **Communications Software Reverse Engineering: A Semi Automatic Approach**. Information and Software Technology, Oxford, n.38, p.379-390, 1996.
- SAMUELSON, P. **Reverse-Engineering Someone Else's Software: Is It Legal?** IEEE Software, v.1, n. 1, 1990, p. 90-96.
- SCHNEIDEWIND, N. F. **Introduction to the Special Section on Software Maintenance. The State of Software Maintenance**. IEEE Transactions on Software Engineering, 1987. pp.303-310.
- SILVA, J. C. & DÍSCOLA, S. L. **Processos de Planejamento da Reengenharia de Software Apoiados por Princípios de Usabilidade**. Universidade Federal de São Carlos. São Carlos, São Paulo, 2003.

- SNEED, H. M. **Migration of Procedurally Oriented COBOL Programs in an Object-Oriented Architecture.** In: Proceedings of Conference on Software Maintenance, Orlando, EUA, 1992, p. 105-116.
- SOMMERVILLE, I. **Software Engineering.** International Computer Science Series. 5ª Edição. Reading: Addison-Wesley, 1995.
- TANGORRA, F. & CHIAROLLA, D. **A methodology for reverse engineering hierarchical databases.** Information and Software Technology, v. 37, n. 4, 1995, p. 225-231.
- TONELLA, P. et al. **Augmenting Pattern-Based Architectural Recovery with Flow Analysis: Mosaic - A Case Study.** In: Proceedings of Third Working Conference on Reverse Engineering, Monterey, EUA, 1996, p. 198-208.
- WARDEN, R. **Re-engineering - A Practical Methodology With Commercial Applications.** In: Applied Information Technology 12 (Software Reuse and Reverse Engineering in Practice). (P. A. V. Hall, ed.) - Chapman@Hall, 1992.
- WATERS, R. C. & CHIKOFFSKY, E. J. **Reverse Engineering: Progress Along Many Dimensions.** Communications of the ACM, v. 37, n. 5, 1994, p. 23-24.
- WONG, K. et al. **Structural Redocumentation: A Case Study.** IEEE Software, v. 12, n. 1, 1995, p. 46-54.

APÊNDICE A

Descrição dos Casos de Uso

A seguir é apresentada a descrição dos casos de usos do Portal Corporativo da 6ª RPM.

| |
|---|
| ID caso de Uso: 01 |
| Nome: Manutenção do Cadastro de Grupos |
| Casos de uso utilizados: 0 |
| Ator(es): Administrador Total e Administrador de Recursos Humanos |
| Descrição: Permite realizar atividades para a manutenção do cadastro de grupos de permissões de acesso a seção Intranet do produto de software. Estas atividades são: cadastrar um novo grupo, alterar os dados de grupos existentes, listar informações destes grupos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um grupo são: nome do grupo, sua descrição, a que unidade operacional pertencerá e se este grupo será um grupo com permissão de acesso ao módulo Administração do produto de software. |
| Execução normal: <ul style="list-style-type: none">▪ O usuário escolhe a opção de administrar grupos e usuários e em seguida grupos;▪ O sistema verifica se o usuário possui permissão para realizar a operação;▪ O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de grupos.▪ O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade. |
| Execução anormal: <ul style="list-style-type: none">▪ O produto de software irá retornar uma mensagem de erro se:<ul style="list-style-type: none">- O usuário não possui permissão para tal funcionalidade do sistema;- Houve uma falha na conexão com o banco de dados devido a problemas com o servidor <i>Web</i> ou o SGBD. |

ID caso de uso: 02

Nome: Manutenção do Cadastro de Usuários

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de Recursos Humanos

Descrição: Permite realizar atividades para a manutenção do cadastro de usuários. Estas atividades são: cadastrar um novo usuário, alterar os dados de usuários existentes, listar informações destes usuários e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um usuário são: nome, sexo, data de nascimento, CPF, RG, e-mail, cargo, *login*, senha, graduação, especialização, unidade operacional que pertence e a que grupos de permissão de acesso pertencerá. Uma vez realizado o cadastro de um usuário este terá acesso a Intranet e a funcionalidades do produto de software as quais possuem permissões de acesso para os grupos que este usuário pertence.

Execução normal:

- O usuário escolhe a opção de administrar grupos e usuários e em seguida usuários;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de usuários.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 03

Nome: Manutenção do Cadastro de Usuários da Imprensa

Casos de uso utilizados: 0

Ator(es): Administrador Total

Descrição: Permite realizar atividades para a manutenção do cadastro de usuários da imprensa. Estas atividades são: cadastrar um novo usuário da imprensa, alterar os dados de usuários da imprensa existentes, listar informações destes usuários e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um usuário da imprensa são: nome, cidade, órgão, e-mail, *login*, senha e unidade operacional que pertence. Uma vez realizada o cadastro de um usuário da imprensa este terá acesso a seção Internet e a funcionalidades do produto de software as quais possuem permissões de acesso para usuários da imprensa.

Execução normal:

- O usuário escolhe a opção de administrar usuários da imprensa;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de usuários da imprensa.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 04

Nome: Manutenção do Cadastro de BPIs

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Ocorrência, Cadastro de Ocorrência - 8º BPM, Cadastro de Ocorrência - 20º BPM, Cadastro de Ocorrência - 24º BPM, Cadastro de Ocorrência - 29º BPM, Cadastro de Ocorrência - 5ª CIA PM IND e Cadastro de Ocorrência - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de BPIs (Boletins Periódicos de Inteligência). Estas atividades são: cadastrar um novo BPI, alterar os dados de BPIs existentes, listar informações destes BPIs e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um BPI são: título do BPI, arquivo do BPI, unidade operacional referente ao BPI.

Execução normal:

- O usuário escolhe a opção de administrar BPIs;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de BPIs.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 05

Nome: Manutenção do Cadastro de Notícias

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Ocorrência, Administrador de P5 - 8º BPM, Administrador de P5 - 20º BPM, Administrador de P5 - 24º BPM, Administrador de P5 - 29º BPM, Administrador de P5 - 5ª CIA PM IND e Administrador de P5 - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de Notícias. Estas atividades são: cadastrar uma nova notícia, alterar os dados de notícias existentes, listar informações destas notícias e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma notícia são: título da notícia, foto, síntese da notícia, cidade, unidade operacional responsável pela notícia.

Execução normal:

- O usuário escolhe a opção de administrar notícias;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de notícias.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 06

Nome: Manutenção do Cadastro de Ocorrências

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Ocorrência, Administrador de Ocorrência - 8º BPM, Administrador de Ocorrência - 20º BPM, Administrador de Ocorrência - 24º BPM, Administrador de Ocorrência - 29º BPM, Administrador de Ocorrência - 5ª CIA PM IND e Administrador de Ocorrência - 14ª CIA PM IND, Cadastro de Ocorrência - 8º BPM, Cadastro de Ocorrência - 20º BPM, Cadastro de Ocorrência - 24º BPM, Cadastro de Ocorrência - 29º BPM, Cadastro de Ocorrência - 5ª CIA PM IND e Cadastro de Ocorrência - 14ª CIA PM IND, Administrador de P5 - 6ª RPM, Administrador de P5 - 8º BPM, Administrador de P5 - 20º BPM, Administrador de P5 - 24º BPM, Administrador de P5 - 29º BPM, Administrador de P5 - 5ª CIA PM IND e Administrador de P5 - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de ocorrências. Estas atividades são: cadastrar uma nova ocorrência, alterar os dados de ocorrências existentes, listar informações destas ocorrências e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma ocorrência são: data da ocorrência, horário, tipo da ocorrência, ementa, síntese, local da ocorrência, cidade, unidade operacional, natureza, flagrante e origem.

Execução normal:

- O usuário escolhe a opção de administrar ocorrências;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de ocorrências.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 07

Nome: Manutenção do Cadastro de Desaparecidos

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de P5 - 6ª RPM

Descrição: Permite realizar atividades para a manutenção do cadastro de pessoas desaparecidas. Estas atividades são: cadastrar um novo desaparecido, alterar os dados dos desaparecidos cadastrados, listar informações destes desaparecidos e removê-los do sistema. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um desaparecido são: foto do desaparecido, nome e sua descrição. Uma vez cadastrado um desaparecido seus dados estarão disponíveis para acesso na seção Internet do produto de software.

Execução normal:

- O usuário escolhe a opção de administrar o Site Externo e em seguida em Desaparecidos;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de desaparecidos.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 08

Nome: Manutenção do Cadastro de Procurados

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de P5 - 6ª RPM

Descrição: Permite realizar atividades para a manutenção do cadastro de pessoas procuradas. Estas atividades são: cadastrar um novo procurado, alterar os dados dos procurados cadastrados, listar informações destes procurados e removê-los do sistema. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um procurado são: foto do procurado, nome e sua descrição. Uma vez cadastrado um procurado, seus dados estarão disponíveis para acesso na seção Internet do produto de software.

Execução normal:

- O usuário escolhe a opção de administrar o Site Externo e em seguida em Procurados;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de procurados.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 09

Nome: Manutenção do Cadastro de *Links*

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de P5 - 6ª RPM

Descrição: Permite realizar atividades para a manutenção do cadastro de *links* que 6ª RPM julga como úteis a seus usuários. Estas atividades são: cadastrar um novo *link*, alterar os dados de *links* existentes, listar informações destes *links* e removê-los do sistema. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um link são: título da página referente ao *link* e *hiperlink*. Uma vez cadastrado um *link* seus dados estarão disponíveis para acesso na seção Internet do produto de software.

Execução normal:

- O usuário escolhe a opção de administrar o Site Externo e em seguida em *Links*;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *links*.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 10

Nome: Manutenção do Cadastro de Enquetes

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de P5 - 6ª RPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de enquetes. Estas atividades são: cadastrar uma nova enquete, alterar os dados de enquetes existentes, listar informações destas enquetes e removê-las do sistema. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma enquete são: título, pergunta da enquete, alternativas (em número menor ou igual a 5) e em que seção do produto de software (Internet e/ou Intranet) a enquete será exibida.

Execução normal:

- O usuário escolhe a opção de administrar Comunicação Virtual e em seguida em Enquete;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de enquetes.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 11

Nome: Manutenção do Cadastro de Fórum de Discussão

Casos de uso utilizados: 0

Ator(es): Administrador Total e Administrador de Comunicação Organizacional.

Descrição: Permite realizar atividades para a manutenção do cadastro de fórum de discussão. Estas atividades são: cadastrar um novo grupo de discussão, alterar os dados de grupos de discussão existentes, listar informações destes grupos, removê-los do sistema e desativar mensagens postadas no fórum de discussão. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um grupo de discussão são: nome do grupo, descrição, e se este grupo trata de um assunto que é sub-tópico de um tema principal anteriormente cadastrado no sistema.

Execução normal:

- O usuário escolhe a opção de administrar Comunicação Virtual e em seguida Fórum de Discussão;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de fórum de discussão.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 12

Nome: Manutenção do Mural de Recados

Casos de uso utilizados: 0

Ator(es): Administrador Total

Descrição: Permite excluir mensagens que o usuário não deseja mais exibir no mural de recados, porém desativando-as e não removendo estas mensagens do banco de dados do sistema.

Execução normal:

- O usuário escolhe a opção de administrar Comunicação Virtual e em seguida Mural de Recados;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as mensagens postadas no mural de recados.
- O usuário seleciona qual(is) mensagem(s) deseja desativar e o produto de software efetua a operação.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 13

Nome: Manutenção do Cadastro de População

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de P3 – 6ª RPM, Administrador de P3 - 8º BPM, Administrador de P3 - 20º BPM, Administrador de P3 - 24º BPM, Administrador de P3 - 29º BPM, Administrador de P3 - 5ª CIA PM IND e Administrador de P3 - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de população. Estas atividades são: cadastrar a população de um município, alterar a população de municípios existentes no sistema, listar informações sobre a população de municípios e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro da população de um município são: nome do município, ano e número de habitantes do município.

Execução normal:

- O usuário escolhe a opção de administrar Ficha de Município e População e em seguida População;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de população.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 14**Nome:** Manutenção do Cadastro de Ficha de Município**Casos de uso utilizados:** 0**Ator(es):** Administrador Total, Administrador de P2 - 6ª RPM, Administrador de P2 - 8º BPM, Administrador de P2 - 20º BPM, Administrador de P2 - 24º BPM, Administrador de P2 - 29º BPM, Administrador de P2 - 5ª CIA PM IND e Administrador de P2 - 14ª CIA PM IND.**Descrição:** Permite realizar atividades para a manutenção do cadastro de ficha de município. Estas atividades são: cadastrar a ficha informativa de um município, alterar fichas de municípios existentes no sistema, listar a ficha informativa dos municípios pertencentes a 6ª RPM e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro da ficha de um município são: Nome do Prefeito, Partido do Prefeito, Endereço do Prefeito, Telefone do Prefeito, Celular do Prefeito, Nome da Esposa do Prefeito, Nome do Vice-Prefeito, Partido do Vice-Prefeito, Endereço do Vice-Prefeito, Telefone do Vice-Prefeito, Celular do Vice-Prefeito, Nome da Esposa do Vice-Prefeito, Nome do Presidente da Câmara, Partido do Presidente da Câmara, Nome do Juiz de Direito, Nome do Promotor de Justiça, Nome do Comandante da Fração PM, Telefone do Comandante da Fração PM, E-mail do Comandante da Fração PM, Telefone da Fração PM, Nome do Delegado Regional, Telefone do Delegado Regional, Celular do Delegado Regional, Nome do Delegado da Comarca, Telefone do Delegado da Comarca, Celular do Delegado da Comarca, Exército, Marinha, Aeronáutica, Nome do Delegado da Polícia Federal, Telefone do Delegado da Polícia Federal, Celular do Delegado da Polícia Federal, E-mail do Delegado da Polícia Federal, Nome do Vigário, Partido do 1º Presidente do Diretório Político, Partido do 2º Presidente do Diretório Político, Partido do 3º Presidente do Diretório Político, Partido do 4º Presidente do Diretório Político, Partido do 5º Presidente do Diretório Político, Partido do 6º Presidente do Diretório Político, Partido do 7º Presidente do Diretório Político, Nome do 1º Presidente do Diretório Político, Nome do 2º Presidente do Diretório Político, Nome do 3º Presidente do Diretório Político, Nome do 4º Presidente do Diretório Político, Nome do 5º Presidente do Diretório Político, Nome do 6º Presidente do Diretório Político, Nome do 7º Presidente do Diretório Político, Celular do 2º Presidente do Diretório

Político, Telefone do 1º Presidente do Diretório Político, Telefone do 2º Presidente do Diretório Político, Telefone do 3º Presidente do Diretório Político, Telefone do 4º Presidente do Diretório Político, Telefone do 5º Presidente do Diretório Político, Telefone do 6º Presidente do Diretório Político, Nome do 7º Presidente do Diretório Político, Celular do 1º Diretório Político, Celular do 3º Presidente do Diretório Político, Celular do 4º Presidente do Diretório Político, Celular do 5º Presidente do Diretório Político, Celular do 6º Presidente do Diretório Político, Celular do 7º Presidente do Diretório Político, Itinerária da Via de Acesso a BH, Observações da Via de Acesso a BH, Via de Acesso a BH Pavimentado, Distância Terrestre de BH, Distância Aérea de BH, Endereço da Prefeitura Municipal, Telefone da Prefeitura Municipal, Nome do Chefe de Gabinete, Celular do Chefe de Gabinete, Nome do Chefe Cerimonial, Celular do Chefe Cerimonial, Nome do Presidente da Microrregião, Celular do Presidente da Microrregião, Prefeito da Microrregião, Endereço da Sede da Microrregião, Município da Microrregião, Telefone da Microrregião, Descrição da Fração PM, Endereço Fração PM, Telefone da Fração PM, População Urbana, População Rural, População Censo, Área do Município em km², Principais Fontes de Economia, Universidades, Principais Indústrias, Total de Escolas Estaduais, Total de Escolas Municipais, Total de Escolas Particulares, Aeroporto, Sala Vip, Comprimento da Pista, Largura da Pista, Balizado, Cercado, Homologado, Hangar, Biruta, Capacidade de Aeronaves, Opera Instrumentos, Telefone do Homologado, Celular do Responsável pelo Homologado, Aeroporto Próximo ao Município, Distância do Município, Via de acesso ao Aeroporto Próximo Pavimentado, Sala Vip, Comprimento da Pista, Largura da Pista, Balizado, Homologado, Cercado, Hangar, Biruta, Capacidade de Aeronaves, Opera Instrumentos, Telefone do Homologado, Celular do Responsável pelo Homologado, Agência Bancária, Rede Hospitalar, Rede Hoteleira, Locadora de Veículos, Nome da Locadora de Veículos, Campo de Futebol Gramado, Campo de Futebol Cercado, Capacidade do Campo de Futebol, Parque de Exposição, DER, Opção de DER Mais Próxima, COPASA, CEMIG, TELEMAR, Correios, SAAE, Telemig Celular, Voltagem Elétrica, Rádios, TVs, Jornais, COMDEC, Partido Majoritário, Deputados Estaduais Majoritários, Deputados Federais Majoritários, Candidato do Governo Apoiado, Total de Integrantes da Câmara do PMDB, Total de Integrantes da Câmara do PFL, Total de Integrantes da Câmara do PSDB, Total de Integrantes da Câmara do PT, Total de Integrantes da Câmara do PDT, Total de Integrantes da Câmara do PRONA, Total de Integrantes da Câmara do PV, Total de Integrantes da Câmara do PL, Total de Integrantes da Câmara do PSB, Total de Integrantes da Câmara de Outros Partidos, Entidade de Classe, Ficha de Município preenchida por, Responsável pelos dados, Data do Cadastro da Ficha de Município.

Execução normal:

- O usuário escolhe a opção de administrar Ficha de Município e População e em seguida Ficha de Município;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de ficha de município.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 15

Nome: Manutenção do Cadastro de *Downloads* da 6ª RPM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador da CRAE, Administrador de Documentos - 6ª RPM, Administrador do Núcleo de Projetos, Administrador de P1 - 6ª RPM, Administrador de P2 - 6ª RPM, Administrador de P3 - 6ª RPM, Administrador de P4 - 6ª RPM, Administrador de P5 - 6ª RPM, Administrador de Saúde, Administrador da Secretaria - 6ª RPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* da 6ª RPM. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* da 6ª RPM;
 - O sistema verifica se o usuário possui permissão para realizar a operação;
 - O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* da 6ª RPM.
- O usuário escolhe qual operação deseja efetuar e o produto de software

- retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 16

Nome: Manutenção do Cadastro de *Downloads* do 8ª BPM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas - 8º BPM, Administrador da CET - 8º BPM, Administrador de Documentos - 8º BPM, Administrador de P1 - 8º BPM, Administrador de P2 - 8º BPM, Administrador de P3 - 8º BPM, Administrador de P4 - 8º BPM, Administrador de P5 - 8º BPM, Administrador da Secretaria - 8º BPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* do 8ª BPM. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* do 8º BPM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* do 8º BPM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 17

Nome: Manutenção do Cadastro de *Downloads* do 20ª BPM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas - 20º BPM, Administrador da CET - 20º BPM, Administrador de Documentos - 20º BPM, Administrador de P1 - 20º BPM, Administrador de P2 - 20º BPM, Administrador de P3 - 20º BPM, Administrador de P4 - 20º BPM, Administrador de P5 - 20º BPM, Administrador da Secretaria - 20º BPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* do 20ª BPM. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* do 20º BPM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* do 20º BPM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 18

Nome: Manutenção do Cadastro de *Downloads* do 24ª BPM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas -24º BPM, Administrador da CET - 24º BPM, Administrador de Documentos -24º BPM, Administrador de P1 - 24º BPM, Administrador de P2 - 24º BPM, Administrador de P3 - 24º BPM, Administrador de P4 - 24º BPM, Administrador de P5 - 24º BPM, Administrador da Secretaria - 24º BPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* do 20ª BPM. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* do 24º BPM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* do 20º BPM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 19

Nome: Manutenção do Cadastro de *Downloads* do 29ª BPM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas - 29º BPM, Administrador da CET - 29º BPM, Administrador de Documentos - 29º BPM, Administrador de P1 - 29º BPM, Administrador de P2 - 29º BPM, Administrador de P3 - 29º BPM, Administrador de P4 - 29º BPM, Administrador de P5 - 29º BPM, Administrador da Secretaria - 29º BPM.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* do 29ª BPM. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* do 29º BPM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* do 29º BPM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 20

Nome: Manutenção do Cadastro de *Downloads* da 5ª CIA PM IND

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas - 5ª CIA PM IND, Administrador da CET - 5ª CIA PM IND, Administrador de Documentos - 5ª CIA PM IND, Administrador de P1 - 5ª CIA PM IND, Administrador de P2 - 5ª CIA PM IND, Administrador de P3 - 5ª CIA PM IND, Administrador de P4 - 5ª CIA PM IND, Administrador de P5 - 5ª CIA PM IND, Administrador da Secretaria - 5ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* da 5ª CIA PM IND. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* da 5ª CIA PM IND;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* da 5ª CIA PM IND.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 21

Nome: Manutenção do Cadastro de *Downloads* da 14ª CIA PM IND

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Atividades Especializadas - 14ª CIA PM IND, Administrador de CET - 14ª CIA PM IND, Administrador de Documentos - 14ª CIA PM IND, Administrador de P1 - 14ª CIA PM IND, Administrador de P2 - 14ª CIA PM IND, Administrador de P3 - 14ª CIA PM IND, Administrador - P4 da 14ª CIA PM IND, Administrador de P5 - 14ª CIA PM IND, Administrador da Secretaria - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de *downloads* da 14ª CIA PM IND. Estas atividades são: cadastrar um arquivo para *download*, alterar os dados de arquivos existentes, listar informações destes arquivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um arquivo para *download* são: arquivo, título do documento, seções do Estado Maior a qual será destinada o documento para *download*.

Execução normal:

- O usuário escolhe a opção de administrar *downloads* da 14ª CIA PM IND;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de *downloads* da 14ª CIA PM IND.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 22

Nome: Manutenção do Cadastro de Cidades

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de UEOPs.

Descrição: Permite realizar atividades para a manutenção do cadastro das cidades pertencentes a 6ª RPM. Estas atividades são: cadastrar uma cidade, alterar os dados de cidades existentes no sistema, listar informações destas cidades e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma cidade são: nome, classificação como integrante do Grupo PM (sub-destacamento ou destacamento), a que destacamento é subordinada no caso de ser classificada como sub-destacamento.

Execução normal:

- O usuário escolhe a opção de administrar Unidades da Polícia Militar e em seguida Cidade;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de cidade.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 23

Nome: Manutenção do Cadastro de Pelotões PM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de UEOPs.

Descrição: Permite realizar atividades para a manutenção do cadastro dos Pelotões PM pertencentes a 6ª RPM. Estas atividades são: cadastrar um Pelotão PM, alterar os dados de Pelotões PM existentes no sistema, listar informações destes Pelotões PM e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de um Pelotão PM são: nome, tipo (Pelotão de Choque, Pelotão de Polícia Militar, Pelotão PM Especial, Pelotão PM Especializado Ambiental, Pelotão PM Especializado Rodoviário, Pelotão PM Especializado Rodoviário e Ambiental), a que Companhia PM é subordinado, cidade sede do Pelotão PM e cidades onde o Pelotão PM atua.

Execução normal:

- O usuário escolhe a opção de administrar Unidades da Polícia Militar e em seguida Pelotão PM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de Pelotão PM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 24

Nome: Manutenção do Cadastro de Companhias PM

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de UEOPs.

Descrição: Permite realizar atividades para a manutenção do cadastro das Companhias PM pertencentes a 6ª RPM. Estas atividades são: cadastrar uma Companhia PM, alterar os dados de Companhias PM existentes no sistema, listar informações destas Companhias PM e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma Companhia PM são: nome, tipo (Companhia de Ensino e Treinamento, Companhia de Polícia Militar, Companhia PM Especial, Companhia PM Especializada Ambiental, Companhia PM Especializada Rodoviária, Companhia PM Especializada Rodoviária e Ambiental), a que Unidade Operacional é subordinada e cidade sede da Companhia PM.

Execução normal:

- O usuário escolhe a opção de administrar Unidades da Polícia Militar e em seguida Companhia PM;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de Companhia PM.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 25

Nome: Manutenção do Cadastro de Unidades Operacionais

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de UEOPs.

Descrição: Permite realizar atividades para a manutenção do cadastro das Unidades Operacionais pertencentes a 6ª RPM. Estas atividades são: cadastrar uma Unidade Operacional, alterar os dados de Unidades Operacionais existentes no sistema, listar informações destas Unidades Operacionais e removê-las. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de uma Unidade Operacional são: nome, tipo (Comando Regional de Polícia Militar, Batalhão de Polícia Militar e Companhia PM Independente) e cidade sede da Unidade Operacional.

Execução normal:

- O usuário escolhe a opção de administrar Unidades da Polícia Militar e em seguida Unidade Operacional;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de Unidade Operacional.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 26**Nome:** Manutenção do Cadastro de Crimes**Casos de uso utilizados:** 0

Ator(es): Administrador Total, Administrador de P3 – 6ª RPM, Administrador de P3 - 8º BPM, Administrador de P3 - 20º BPM, Administrador de P3 - 24º BPM, Administrador de P3 - 29º BPM, Administrador de P3 - 5ª CIA PM IND, Administrador de P3 - 14ª CIA PM IND, Administrador de Crimes – 6ª RPM, Administrador de Crimes - 8º BPM, Administrador de Crimes - 20º BPM, Administrador de Crimes - 24º BPM, Administrador de Crimes - 29º BPM, Administrador de Crimes - 5ª CIA PM IND e Administrador de Crimes - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de crimes. Estas atividades são: cadastrar crimes, alterar os dados de crimes existentes, listar informações destes crimes e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de crimes são: cidade de ocorrência do crime, Pelotão PM que atua na cidade, período de ocorrência do crime, grupo da classe do crime. Uma vez preenchido os campos da página com os dados citados anteriormente, uma tabela será apresentada com a classes e respectivas sub-classes do grupo da classe do crime selecionado na qual será preenchida a quantidade de crimes ocorridos de cada sub-classe.

Execução normal:

- O usuário escolhe a opção de administrar Crimes;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de crimes.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 27

Nome: Manutenção do Cadastro de Efetivo

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de P1 – 6ª RPM, Administrador de P1 - 8º BPM, Administrador de P1 - 20º BPM, Administrador de P1 - 24º BPM, Administrador de P1 - 29º BPM, Administrador de P1 - 5ª CIA PM IND e Administrador de P1 - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de efetivo. Estas atividades são: cadastrar o efetivo de uma cidade, alterar os dados de efetivos existentes, listar informações destes efetivos e removê-los. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de efetivo são: cidade, tipo de efetivo presente na cidade (Efetivo total da cidade, Policiamento Especializado, Policiamento Ostensivo, Efetivo da Administração da 6ª RPM, Efetivo da Administração de Unidades Operacionais, Efetivo de Companhia PM e Efetivo de Pelotão PM) e a quantidade de efetivo de cada posto ou graduação da Polícia Militar.

Execução normal:

- O usuário escolhe a opção de administrar Efetivo e Treinamento e em seguida Efetivo;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de efetivo.
- O usuário escolhe qual operação deseja efetuar e o produto de software retorna uma tela referente a esta atividade.

Execução anormal:

- O produto de software irá retornar uma mensagem de erro se:
 - O usuário não possui permissão para tal funcionalidade do sistema;
 - Houve uma falha na conexão com o banco de dados devido a problemas com o servidor *Web* ou o SGBD.

ID caso de uso: 28

Nome: Manutenção do Cadastro de Treinamento

Casos de uso utilizados: 0

Ator(es): Administrador Total, Administrador de Treinamento – 6ª RPM, Administrador de Treinamento - 8º BPM, Administrador de Treinamento - 20º BPM, Administrador de Treinamento - 24º BPM, Administrador de Treinamento - 29º BPM, Administrador de Treinamento - 5ª CIA PM IND e Administrador de Treinamento - 14ª CIA PM IND.

Descrição: Permite realizar atividades para a manutenção do cadastro de treinamento. Estas atividades são: cadastrar o efetivo avaliado no treinamento, alterar os dados do efetivo avaliado, cadastrar resultados de avaliação de treinamento, alterar resultados de avaliações, cadastrar gastos com munição em treinamento, alterar gastos com munição, cadastrar recursos gastos com treinamento ou avaliação e alterar recursos gastos com treinamento ou avaliação. Os dados armazenados no banco de dados do sistema ao efetuar o cadastro de efetivo avaliado são: período do treinamento (trimestre), ano, quantidade de efetivo disponível e avaliado no trimestre e até o trimestre de cada treinamento (TPB, TEF, TCAF, TC) e de cada unidade operacional. Ao efetuar o cadastro de resultados de avaliação os seguintes dados são necessários: período do treinamento (trimestre), ano, quantidade de efetivo aprovado e reprovado de cada treinamento (TPB, TEF, TCAF, TC) e de cada unidade operacional. No cadastro de gastos com munição os dados inseridos no banco de dados do sistema são: período do treinamento (trimestre), ano, quantidade de munição gasta no treinamento e avaliação de cada tipo de arma (Calibre 12, Calibre 38, Calibre 40, Calibre 7,62 mm e Calibre 9 mm) e de cada unidade operacional. Ao efetuar o cadastro de gastos com avaliação ou treinamento os seguintes dados são necessários: período do treinamento (trimestre), ano, quantidade de recursos gastos com diárias, materiais de consumo, inscrições ou cursos de cada unidade operacional.

Execução normal:

- O usuário escolhe a opção de administrar Efetivo e Treinamento e em seguida Treinamento;
- O sistema verifica se o usuário possui permissão para realizar a operação;
- O produto de software retorna uma tela com as atividades que podem ser realizadas sobre o cadastro de treinamento.

RESUMO

Título: Manutenção e Documentação do Portal Corporativo da 6ª Região da PMMG

A necessidade de realizar a manutenção de um produto de software é motivada principalmente por alguma insatisfação dos usuários com a sua utilização e, conseqüentemente, pela falha em sua usabilidade. Porém, o processo de manutenção de produtos de software muitas vezes torna-se complexo, quando a sua modelagem e a sua documentação não existem ou foram mal elaboradas. Sob esta ótica, foi constatada a necessidade de realizar a manutenção do Portal Corporativo da 6ª Região da Polícia Militar de Minas Gerais (6ª RPM). Porém, para que esta ferramenta continue satisfazendo as necessidades dos membros da corporação, é essencial que algumas alterações e adições sejam realizadas na funcionalidade deste produto de software. Principalmente no que se refere ao dinamismo do conteúdo das páginas que compõem o Portal e, à sua interface, fazendo com que características de usabilidade estejam presentes no produto de software. Assim, neste trabalho foram realizadas a manutenção e documentação do Portal Corporativo da 6ª Região da Polícia Militar de Minas Gerais onde foram aplicados os conceitos de engenharia reversa de software e reengenharia de software e a modelagem UWE (*UML-based Web Engineering*).