

**DENISE FERNANDES
PAULO HENRIQUE HIRTH
ROMUALDO KRAMECK JUNIOR**

IDENTKIT - SISTEMA PARA DESENVOLVIMENTO DE RETRATOS FALADOS

Curitiba – Paraná
Dezembro de 2007

**DENISE FERNANDES
PAULO HENRIQUE HIRTH
ROMUALDO KRAMECK JUNIOR**

IDENTKIT - SISTEMA PARA DESENVOLVIMENTO DE RETRATOS FALADOS

Monografia apresentada como requisito parcial à obtenção do grau de Tecnólogo em Informática, curso de Tecnologia em Informática, Escola Técnica da Universidade Federal do Paraná.

Orientador : Prof Dieval Guizelini

Curitiba - Paraná
Dezembro de 2007

DEDICATÓRIA

Aos professores do curso de Tecnologia em Informática que não pouparam nenhum esforço para nos transmitir seu conhecimento no período em que estivemos na graduação.

AGRADECIMENTOS

Primeiramente a Deus, a todos que, direta ou indiretamente, contribuíram para a realização desta pesquisa, a nossos familiares que nos momentos nos quais mais precisamos estiveram nos dando apoio e nos motivando a prosseguir e especialmente aos professores que nos orientaram e ensinaram durante esta jornada, Prof DR Roberto e Prof Dieval.

SUMÁRIO

RESUMO.....	10
ABSTRACT.....	11
1. INTRODUÇÃO	12
2. OBJETIVO GERAL.....	12
3. OBJETIVOS ESPECÍFICOS.....	13
4. JUSTIFICATIVA.....	13
5. FUNDAMENTAÇÃO TEÓRICA.....	13
5.1. SOFTWARES COMERCIAIS PESQUISADOS	13
5.1.1. FotoCrim.....	13
5.1.2. Photo Composer	14
5.1.3. Griaule ICAO Face	15
5.1.4. Faces.....	15
5.1.5. Digital Physiomy	16
5.2. TECNOLOGIAS ESTUDAS.....	16
5.2.1. SVG.....	16
5.2.2. Apache Batik	16
5.2.3. Salamander	16
5.2.4. JAVA 2D.....	17
5.2.5. Hibernate.....	17
5.2.6. Design Patterns.....	18
5.2.7. MVC	19
5.2.8. PostgreSQL.....	20
5.2.9. JAVA	20
6. DESCRIÇÃO DO PRODUTO.....	21
6.1. IDENTKIT- SISTEMA DE CRIAÇÃO DE RETRATOS FALADOS.....	21
6.1.1. Retrato Falado.....	21
6.1.2. Montagem de um Retrato Falado.....	21
6.1.3. Biblioteca de Moldes	22
6.1.4. Produto Final	23
6.1.5. Configuração dos Moldes.....	23
6.1.6. Sobreposição de Imagens.....	24
6.1.7. Vetorização	25

6.1.8. SVG.....	25
6.1.9. Pesquisa de Moldes	26
6.1.10. Relatórios	27
6.1.11. Banco de Dados- PostgreSQL8.1	27
7. DIAGRAMAS	28
7.1. DIAGRAMA DE CASOS DE USO	28
7.2.ESPECIFICAÇÃO DE CASOS DE USO	28
7.3.DIAGRAMA DE CLASSE	77
7.4. DIAGRAM DE COMPONENTES.....	78
7.5. DIAGRAMA DE ATIVIDADES.....	79
7.6. DIAGRAMA DE SEQUÊNCIA TELA ESCOLHA.....	80
7.7. DIAGRAMA DE SEQUÊNCIA TELA CONFIGURAÇÃO	81
7.8. DIAGRAMA ENTIDADE RELACIONAMENTO.....	82
7.9. DICIONÁRIO DE DADOS	83
8. AMBIENTE UTILIZADO PARA O DESENVOLVIMENTO.....	87
9. REQUISITOS DO SISTEMA E SOFTWARE E HARDWARE.....	87
10. CONCLUSÃO	88
11. DIFICULDADE ENCONTRADAS	88
12. PROJETOS FUTUROS.....	89
13. GLOSSÁRIO	89
14. REFERÊNCIAS	90
ANEXO i.....	92
ANEXO ii.....	93

LISTA DE FIGURAS

FIGURA 1-BIBLIOTECA DE MOLDES	22
FIGURA 2-PRODUTO FINAL	23
FIGURA 3-CASO DE USO	28
FIGURA 4-TELA DESENHO	32
FIGURA 5-GUIA CABELO	34
FIGURA 6-CABELO	35
FIGURA 7-GUIA NARIZ/BOCA.....	37
FIGURA 8-NARIZ/BOCA	38
FIGURA 9-GUIA OLHO.....	40
FIGURA 10-OLHO.....	41
FIGURA 11-GUIA ROSTO	43
FIGURA 12-ROSTO	44
FIGURA 13-GUIA SOBRANCELHA.....	46
FIGURA 14-SOBRANCELHA	47
FIGURA 15-TELA CADASTRO	49
FIGURA 16-GUIA DE CONFIGURAÇÃO	51
FIGURA 17-RELATÓRIO.....	53
FIGURA 18-LAYOUT.....	55
FIGURA 19-PESQUISA	57
FIGURA 20-PESQUISAR	59
FIGURA 21-ALTERAR	60
FIGURA 22-PESQUISA DE USUÁRIO.....	62
FIGURA 23-EXCLUIR.....	62
FIGURA 24-TELA USUÁRIO JAVA	64
FIGURA 25-PESQUISA USUARIO2	66
FIGURA 26-TELA MOLDE	68
FIGURA 27-LISTA PEQUISA MOLDES.....	71
FIGURA 28-ALTERAR	71
FIGURA 29-LISTA PESQUISA MOLDES2	74
FIGURA 30-EXLCUIR MOLDE.....	74
FIGURA 31-PESQUISA MOLDES3.....	76

FIGURA 32-DIAGRAMA DE CLASSE	77
FIGURA 33-DIAGRAMA DE COMPONENTES	78
FIGURA 34-DIAGRAMA DE ATIVIDADES.....	79
FIGURA 35-DIAGRAMA DE SEQÜÊNCIA TELA ESCOLHA	80
FIGURA 36-DIAGRAMA DE SEQÜÊNCIA TELA CONFIGURAÇÃO	81
FIGURA 37-DIAGRAMA ENTIDADE RELACIONAMENTO	82

LISTA DE TABELAS

TABELA 1 – DESCRIÇÃO DA TABELA USUARIO	83
TABELA 2 – DESCRIÇÃO DA TABELA RETRATO	83
TABELA 3 – DESCRIÇÃO DA TABELA MOLDE_RETRATO.....	84
TABELA 4 – DESCRIÇÃO DA TABELA MOLDE.....	84
TABELA 5 – DESCRIÇÃO DA TABELA CARACTERISTICA.....	85
TABELA 6 – DESCRIÇÃO DA TABELA TIPO CARACTERISTICA.....	85
TABELA 7 – DESCRIÇÃO DA TABELA TIPO CARACTERISTICA TIPO MOLDE.....	85
TABELA 8 – DESCRIÇÃO DA TABELA TIPO MOLDE	86
TABELA 9 – DESCRIÇÃO DA TABELA VALOR	86
TABELA 10 – DESCRIÇÃO DA TABELA PERGUNTA.....	86

RESUMO

Esse projeto teve como finalidade criar um software que dê apoio a um especialista na área de criação de retratos falados, sendo assim, após estudos de como isso seria melhor implementado, surge um software que visa organizar uma biblioteca de moldes faciais humanos, e que tem a finalidade de agrupar e configurar essas partes do rosto, a fim de que o produto final seja uma representação fiel de uma figura humana. Utilizando conceitos de figuras vetorizadas, foi criado uma interface de desenho com muitos detalhes e possibilidades.

Palavras Chaves: Retrato falado, software, vetorização

ABSTRACT

This project was aimed at creating software that would support a specialist in the area of creating facial composites, so after studies of how this would be better implemented, there is a software that aims to organize a library of human facial templates, and has the purpose of group and configure these parts of the face so that the final product is a faithful representation of a human figure. Using concepts of vectorize figures, was created an design interface with many details and possibilities.

Key Words: facial composites, software, vectorize.

1. INTRODUÇÃO

O retrato falado tem sido considerado uma das mais importantes e eficazes ferramentas policiais para localização de marginais ou recuperação de pessoas desaparecidas. Entretanto a demanda gerada pela produção de retratos falados pelos Departamentos policiais das grandes médias ou pequenas cidades, geralmente, tem sido superior à capacidade de atendimento dos policiais com qualificação ou habilidade produzi-los.

Assim no intuito de suprir esta demanda, desenvolvem-se uma ferramenta alternativa para construção de retrato falado a qual permite que profissionais dotados de pouca habilidade artística, em curto tempo de treinamento possam produzir, em escala, um retrato falado superior, em qualidade, aos desenhos produzidos à mão-livre, e a fidelidade desejável nos resultados, que descrevem as características fisionômicas relatadas pelas vítimas ou testemunhas de delitos criminais.

O processo manual, devido a sua simplicidade de métodos, permite que o retrato falado manual seja gerado em qualquer ambiente e depende do simples deslocamento do entrevistador. Entretanto, o esforço para geração do desenho final é muito alto, sendo grande a perda de tempo e material até a conclusão do trabalho. Oferece baixa interação com o entrevistado e nenhuma interatividade com cadastros digitais. O sistema automatizado amplia a capacidade de atendimento da demanda interna de retratos falados e possui bom nível de interatividade com os usuários envolvidos. Sua eficiência está diretamente relacionada com a flexibilidade para modelagem facial e da capacidade de personalização dos moldes pré-armazenados.

Realizamos diversas reuniões com especialistas responsáveis por desenvolver retratos falados no Estado do Paraná, para extrair informações e adquirir experiência no assunto para desenvolver o software.

2. OBJETIVO GERAL

Desenvolver um sistema para montagem de Representações Faciais Humanas. Este é destinado às delegacias de polícia do Estado do Paraná e ao Instituto de Criminalística. Será uma ferramenta de apoio destinada a auxiliar o entrevistador. O sistema informatizará a construção da representação facial, aliado a algumas características físicas que poderão auxiliar na pesquisa por retratos já cadastrados.

3. OBJETIVOS ESPECÍFICOS

- Cadastro de representações faciais humanas desenvolvidos pelo usuário. O Cadastro da representação facial dará-se em duas etapas. Primeiro o usuário indicará as características físicas pertinentes à pessoa que será cadastrada. Depois, na segunda etapa ele montará a imagem da pessoa que se estiver tentando retratar.
- Disponibilizar moldes faciais. A apresentação dos moldes faciais terá como objetivo facilitar algumas lembranças para o entrevistado.

4. JUSTIFICATIVA

O sistema proposto visa desenvolver uma ferramenta para o auxiliar à Secretaria de Estado da Segurança Pública na montagem de retrato falado. Uma das principais formas utilizadas pela polícia na identificar criminosos é através da "representação facial humana" (também popularmente conhecida como "retrato falado").

5. FUNDAMENTAÇÃO TEÓRICA

O retrato falado é uma ferramenta que realiza um papel importante nas investigações policiais há muito tempo, tendo parte como prova-chave em vários casos. Porém pela dificuldade no processo de desenvolvimento, e pela escassez de profissionais capacitados de desenvolver retratos-falados com qualidade, essa ferramenta tem sido subutilizada.

A seguir são citados alguns produtos avaliados durante a pesquisa base deste trabalho.

5.1. SOFTWARES COMERCIAIS PESQUISADOS

5.1.1. FotoCrim

O software FotoCrim executa com eficiência a exaustiva e, por tantas vezes, infrutífera pesquisa de indivíduos em bases de dados de médio ou grande porte, considerando características físicas declaradas por testemunhas ou captadas por veículos de monitoração presentes. O sistema seleciona candidatos com melhor aproxima-se de parâmetros selecionados como argumento de pesquisa.

Desta forma, com recursos de paginação progressiva, a pesquisa visual dentre candidatos contidos no universo restrito do resultado processado, torna-se tarefa rápida e muito prática.

O sistema Fotocrim suporta recursos avançados de pesquisa de relacionamentos, com base em argumentação previa de ocorrências.

Estes argumentos podem ser personalizados de acordo com cenários e meio ambiente das pessoas relacionadas no universo considerado para as pesquisas.

A codificação utilizada para identificação dos moldes permite uma linguagem comum entre o retrato falado e o sistema Fotocrim. Os dados faciais-biométricos são registrados automaticamente no painel de integração, podendo, opcionalmente, receber dados complementares como estatura, idade, cicatrizes e outros. A desvantagem das exatas proporções exigidas no desenvolvimento de um modelo orientado para divulgação visual deixa de ser tão necessariamente exatas na execução automática utilizada no processamento desta integração

5.1.2. PhotoComposer

Possui um banco de Imagens Flexível , não incorpora um banco de imagens nativo, permitindo que o usuário incorpore tipos e costumes regionais, através de imagens abertas com extensão [bmp], [jpg] ou [gif]. Fácil de operar, o PhotoComposer oferece usabilidade e operação intuitiva, mesmo para aqueles sem experiência na produção de retratos falados ou maior experiência em sistemas automatizados. Partindo essencialmente de fotos de pessoas vivas (mas não exclusivamente), e objetos concretos, tratados com exclusivo acabamento gráfico, a seleção final do trabalho em muito se aproxima de uma fotografia, oferecendo grande realismo na qualidade do modelo final pretendido. Facilidades na edição, formatação e armazenamento dos trabalhos.

O PhotoComposer permite que os trabalhos finais sejam armazenados em formato BMP. Isto representa uma facilidade de leitura por qualquer programa editor de imagens para eventuais retoques ou tratamento gráfico desejado pelo usuário. Além disso, o sistema armazena os trabalhos em formato código, que referencia os moldes apenas armazenando os seus códigos e proporções utilizadas nos projetos faciais. O PhotoComposer é gerado de forma personalizada para seu usuário, permitindo o acesso apenas em seu computador e mediante a sua senha pessoal. O

desenvolvimento do programa considerou determinações ergonômicas de usabilidade e especificações comandadas por necessidades de policiais oriundos de corporações civil e militar, com grande experiência em trabalhos de desenvolvimento de retratos falados voltados para busca de indivíduos suspeitos ou foragidos.

A seleção do molde desejado é executada em um painel que oferece ótima visibilidade para o entrevistado, através de um simples clique do mouse sobre o molde desejado pelo entrevistador.

5.1.3. Griaule ICAO Face

Griaule ICAO Face é um Kit de Desenvolvimento de Software (SDK) que permite sistemas de identificação automática lidar com imagens faciais seguindo padrões internacionais. Imagens faciais digitalizadas são usadas por muitos sistemas, como em e-passaport. As dificuldades mais comuns ao lidar com as fotos são as diferenças entre as imagens, no tamanho, na posição da face, cor e contraste e uma série de outras variáveis. Griaule ICAO Face processa uma imagem digitalizada de um rosto para atender aos padrões exigidos pelo ICAO (Organização Internacional de Aviação Civil). O ICAO lidera os esforços para criar padrões para as novas tecnologias de passaporte eletrônico, incluindo um padrão para imagens faciais digitalizadas que garante requisitos fotográficos e digitais.

5.1.4. Faces

A tecnologia do FACES é utilizada por milhares de agências policiais todo o mundo - incluindo o FBI e CIA - para identificar, monitorar e deter suspeitos criminosos.

FACES 4,0 possui as seguintes características principais:

- Fácil de usar, qualquer oficial da polícia pode criar um retrato com um clique de um mouse;
- Ampla Base de Dados, possui uma base em torno de 4400 moldes incluindo novas características de latinos, africanos, americanos e asiáticos;

- Reconhecimento facial e identificação, as imagens de qualidade gerada pelo software permite ser integrado com tecnologias de reconhecimento facial;

5.1.5. Digital Physiomy

Digital Physiomy usa uma sofisticada rede neural para identificar correlações entre características faciais e características psicológicas. O programa determina uma pessoa de características psicológicas e apresenta uma análise detalhada ao caráter dessa pessoa em um formato gráfico. Em seguida, você poderá imprimir um relatório ou salvá-lo como uma página da web. Ele possui bibliotecas que incluem amostras de rostos humanos.

5.2. TECNOLOGIAS ESTUDADAS

5.2.1. SVG

SVG é a abreviatura de Scalable Vectorial Graphics que pode ser traduzido do [inglês](#) como [gráficos vectoriais](#) escaláveis. Trata-se de uma linguagem [XML](#) para descrever de forma vectorial desenhos e gráficos bidimensionais, quer de forma estática, quer dinâmica ou animada. Uma das principais características dos gráficos vectoriais, é que não perdem qualidade ao serem ampliados. A grande diferença entre o SVG e outros formatos vectoriais, é o fato de ser um formato aberto, não sendo propriedade de nenhuma empresa. Foi criado pela [World Wide Web Consortium](#), responsável pela definição de outros padrões, como o [HTML](#) e o [XHTML](#).

A criação deste formato foi baseada noutros já existentes: [CSS](#), [DOM](#), [JPEG](#), [PNG](#), [SMIL](#) e [XML](#). O formato SVG, neste momento, permite três tipos de objetos gráficos:

- formas geométricas vectoriais ;
- imagens [raster](#) / [bitmap](#);
- texto;

Quanto ao armazenamento, as imagens no formato SVG podem ser gravadas com a [compressão gzip](#), [sem perda de dados](#), podendo-se chamar neste caso, ficheiros SVGZ (assim como a [extensão](#)). Devido ao XML conter muitos dados redundantes, pode ser comprimido, resultando em

arquivos bastante menores. No entanto muitas vezes, o formato SVG já é por si menor que a sua versão em raster.

5.2.2. Apache Batik

Para podermos trabalhar com a linguagem SVG que foi atribuída ao software escolhemos usar um toolkit desenvolvido pela apache, aonde facilmente podemos gerar e manipular arquivos vetorizados em um ambiente JAVA. O Batik também nos dá a possibilidade de exportarmos o produto final que gerarmos em formato SVG, para algum formato raster, mais fácil de ser ajustado por outros editores gráficos, caso seja necessário.

No decorrer do projeto foi estudada mais uma ferramenta que ajudaria na utilização de documentos SVG, porém, o batik se mostrou mais bem preparado para atender qualquer necessidade da equipe, e além disso a facilidade da implementação foi algo realmente surpreendente para toda a equipe. Após algumas tentativas falhas de se utilizar classes java puras, para tratamento de imagens, a vetorização com certeza foi algo que somou muito a qualidade do software final.

5.2.3. Salamander

Toolkit para tratamento de documentos SVG, foi descartado após integração do Apache Batik.

5.2.4. JAVA 2D

API utilizada para implementar a parte gráfica no começo da construção do software, foi totalmente descartada após problema na sobreposição de imagens raster, e após a integração do toolkit SVG Apache Batik.

5.2.5. Hibernate

O Hibernate é um [framework](#) para o [mapeamento objeto-relacional](#) escrito na linguagem [Java](#), mas também é disponível em [.Net](#) como o nome [NHibernate](#). Este programa facilita o mapeamento dos atributos entre uma base tradicional de dados relacionais e o modelo objeto de uma aplicação, mediante o uso de arquivos ([XML](#)) para estabelecer esta relação. Hibernate é um [software livre](#) de [código aberto](#) distribuído com a licença [LGPL](#). O objetivo do Hibernate é diminuir a complexidade entre os programas Java, baseado no modelo [orientado a objeto](#), que precisam

trabalhar com um banco de dados do [modelo relacional](#) (presente na maioria dos [SGDBs](#)). Em especial, no desenvolvimento de consultas e atualizações dos dados.

Sua principal característica é a transformação das classes em Java para tabelas de dados (e dos [tipos de dados](#) Java para os da [SQL](#)). O Hibernate gera as chamadas SQL e libera o desenvolvedor do trabalho manual da conversão dos dados resultante, mantendo o programa portátil para quaisquer bancos de dados SQL, porém causando um pequeno aumento no tempo de execução.

Nas questões relacionadas para o gerenciamento de transações e na tecnologia de acesso à base de dados são de responsabilidade de outros elementos na infraestrutura do programa. Apesar de existirem [API](#) no Hibernate para possuir operações de controle transacional, ele simplesmente delegará estas funções para a infraestrutura na qual foi instalada.

A HQL (Hibernate Query Language) é um dialeto [SQL](#) para o Hibernate. Ela é uma poderosa linguagem de consulta que se parece muito com a SQL, mas a HQL é totalmente orientada a objeto, incluindo os paradigmas de herança, polimorfismo e encapsulamento. No Hibernate, você pode escolher tanto usar a SQL quanto a HQL. Escolhendo a HQL, você poderá executar os pedidos SQL sobre as classes de persistência do Java ao invés de tabelas no banco de dados, aumentando, assim, a distância entre o desenvolvimento da [regras de negócio](#) e o banco de dados.

5.2.6. Design Patterns

Os padrões de projeto de software ou padrões de desenho de software, também muito conhecido pelo termo original em [inglês](#): Design Patterns, descrevem soluções para problemas recorrentes no desenvolvimento de sistemas de [software orientados a objetos](#). Um padrão de projeto estabelece um nome e define o problema, a solução, quando aplicar esta solução e suas conseqüências.

Os padrões de projeto visam facilitar a reutilização de soluções de desenho - isto é, soluções na fase de projeto do software, sem considerar reutilização de código. Também acarretam um vocabulário comum de desenho, facilitando comunicação, documentação e aprendizado dos sistemas de software.

No projeto foi utilizado o Data Access Object (DAO), que atualmente é o modelo preferido para persistir dados em aplicações é de [banco de dados](#) relacional. DAO (Data Access Object) é um padrão para persistência de dados que permite separar regras de negócio das regras de acesso a banco de dados. Numa aplicação que utilize a arquitetura [MVC](#), todas as funcionalidades de bancos de dados, tais como obter as conexões, mapear objetos [Java](#) para tipos de dados [SQL](#) ou executar comandos SQL, devem ser feitas por classes de DAO.

5.2.7. MVC

Model-view-controller (MVC) é um padrão de [arquitetura de software](#). Em aplicações complexas, que enviam uma série de dados para o usuário, o desenvolvedor freqüentemente necessita separar os dados (Model) da [interface](#) (View). Desta forma, alterações feitas na interface não afetarão a manipulação dos dados, e estes poderão ser reorganizados sem alterar a interface do usuário. O model-view-controller resolve este problema através da separação das tarefas de acesso aos dados e lógica do negócio da apresentação e interação com o usuário, introduzindo um componente entre os dois: o Controller. MVC é usado em [padrões de projeto de software](#), mas MVC abrange mais da arquitetura de uma aplicação do que é típico para um padrão de projeto.

É comum dividir a aplicação em camadas separadas: apresentação (interface), domínio e acesso a dados. Em MVC a camada de apresentação também é separada da view e da controller.

Model

O Modelo que é o objeto de aplicação, ou seja, onde fica toda a regra de negócio do sistema;

View

A visão, que trata da apresentação do objeto na tela, pode-se afirmar que é a interface de representação visual do modelo;

Controller

O controlador cuja função é controlar as interações do usuário com a interface. A maneira como são organizados os componentes dentro do paradigma MVC oferece uma grande confiabilidade.

5.2.8. PostgreSQL

PostgreSQL é um [sistema gerenciador de banco de dados objeto relacional](#) (SGBDOR), desenvolvido como projeto [software livre](#). Hoje, o PostgreSQL é um dos SGBD (Sistema Gerenciador de Bancos de Dados) de código aberto mais avançados, contando com recursos como: consultas complexas, chaves estrangeiras, integridade transacional, controle de concorrência multi-versão, suporte ao modelo híbrido objeto-relacional, [trigger](#), [views](#), [stored procedures](#) em várias linguagens.

5.2.9. JAVA

Java é uma [linguagem de programação orientada a objeto](#) desenvolvida na [década de 90](#) pelo programador [James Gosling](#), na empresa [Sun Microsystems](#). Diferentemente das linguagens convencionais, que são [compiladas](#) para [código nativo](#), a linguagem Java é compilada para um "[bytecode](#)" que é executado por uma [máquina virtual](#). A linguagem de programação Java é a linguagem convencional da [Plataforma Java](#), mas não sua única linguagem.

6. DESCRIÇÃO DO PRODUTO

6.1. IDENTKIT – SISTEMA DE CRIAÇÃO DE RETRATOS FALADOS.

A idéia inicial na criação do IdentKit, veio da necessidade de aumentar a área de abrangência que tange a montagem de retratos falados hoje em dia. Atualmente, a montagem de uma representação facial a partir do relato de uma vítima, é única e exclusivamente possível apenas com a presença de um especialista em desenho, esse por sua vez, necessita de um conhecimento praticamente único na modelagem de um rosto humano. Assim, se o conhecimento na modelagem de moldes faciais for incorporada a um software que gerencie uma biblioteca de imagens tratadas, todo esse processo de criação poderá ser facilitado e até mesmo incorporado a usuários menos experientes.

6.1.1. Retrato Falado

Item central na pesquisa do desenvolvimento do IdentKit, a representação facial humana pode ser descrita como:

É a descrição de um conjunto de elementos antropológicos, além daqueles referentes primariamente à face humana, e de quaisquer outras informações características, sinaléticas e secundárias, relatadas de maneira aproximada pela vítima de um crime, ou testemunha deste, com a finalidade de promover a identificação parcial do agressor, através do desenho artístico, técnico e científico, realizado por um perito nesta área de identificação humana.

(Jorge Luiz Werzbitzki).

6.1.2. Montagem de um Retrato Falado

Como relatado por Werzbitzki, é através de um desenho artístico que se chega ao produto final quando se trata da montagem de uma representação facial humana, então, um dos focos de estudo na criação do IdentKit, foi de como criar um ambiente de desenho, sem precisar realmente da parte artística de um especialista para efetua-lo. Sendo assim, após estudos com ferramentas já existentes, e contatos com especialistas da área, optamos por criar uma biblioteca de moldes faciais humanos,

onde cada peça se encaixaria nas outras subseqüentes, a fim de que o produto final seja um rosto humano.

6.1.3. Biblioteca de Moldes

A partir da orientação do especialista da área de criação de retratos falados do Instituto de Identificação da Polícia Civil do Paraná, conseguimos tratar fotos de rostos humanos, e, retirarmos todas as peças principais que formam a face humana. Sendo essas, os Olhos, Cabelo, o rosto, a sobrancelha, e por fim um molde conjunto de Nariz e Boca, sendo esse último, por orientação do especialista, ficará mesclado, pois uma vítima tem dificuldade em lembrar-se de como eram essas partes de um agressor, sem automaticamente elas estarem juntas. Os moldes faciais finais podem ser visualizados na figura 30.

Todos os moldes gerados para o sistema são figuras raster's ¹, e estão em formato PNG (Portable Network Graphics), além de ser um formato livre, ele é amplamente recomendado pela W3C, pois não perde qualidade quando se é comprimido, diferentemente do JPG, além de ser possível a exclusão do fundo da figura, para um melhor acabamento final do molde.

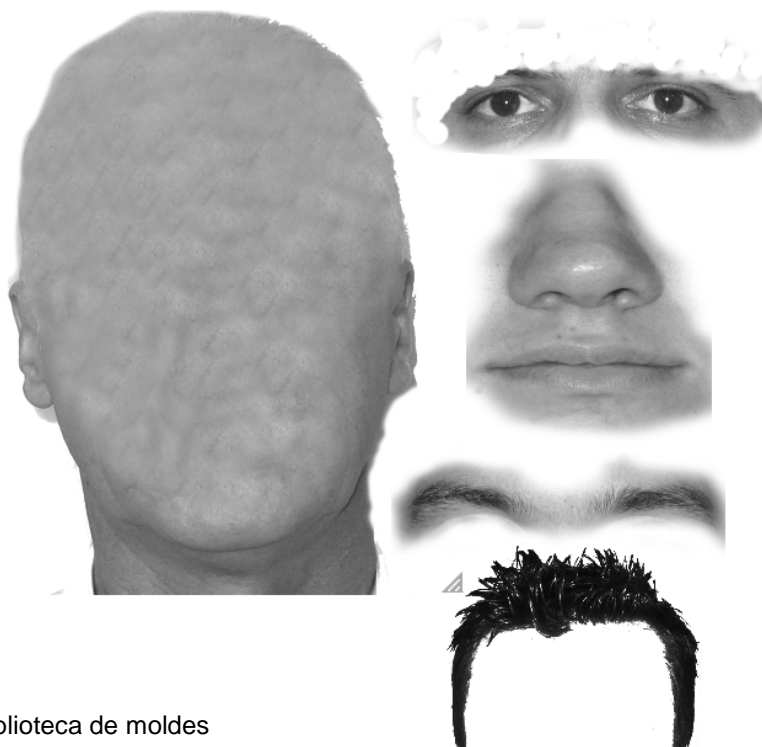


Fig. 1 Biblioteca de moldes

6.1.4. Produto Final

Após a biblioteca de moldes ser populada com um conjunto de no mínimo um molde de cada tipo, é possível montar uma representação facial humana, e isso ocorre com a escolha manual dos moldes de dentro da Tela “Biblioteca de moldes” do software. Após um tipo de molde ser escolhido, ele aparecerá automaticamente dentro da Tela de Desenho, onde é possível realizar pequenos ajustes na posição, tamanho e ordem das camadas em que o molde facial se encontra, sendo, assim, após cada um dos cinco tipos de moldes for escolhido, o produto final será uma fiel representação facial humana assim como se é visualizado na figura 29.



Fig.2 - Produto Final

6.1.5. Configuração dos Moldes.

Outra funcionalidade do IdentKit, diz respeito a Tela de configuração de moldes, aonde é possível o ajuste fino de cada um dos cinco tipos de moldes, sendo esses ajustes, descritos em:

- Altura do Molde: Ajusta a altura (height), do molde, podendo, tanto diminuir, como aumentar esse valor.
- Largura do Molde: Ajusta a largura (width), do molde, podendo, tanto diminuir, como aumentar esse valor.
- Posição x do molde: Ajusta a posição horizontal (x) do molde em relação a tela de desenho podendo, tanto diminuir, como aumentar esse valor.
- Posição y do molde: Ajusta a posição vertical (y) do molde em relação a tela de desenho, podendo, tanto diminuir, como aumentar esse valor.
- Ajuste da ordem da camada referente ao molde que está selecionado.
- Exclusão de um tipo molde da tela de desenho.
- Reverter a última alteração feita na escolha de um dos tipos de moldes.

6.1.6. Sobreposição de Imagens

Após os estudos iniciais de formação do software, ficou claro para os integrantes da equipe que seria necessário trabalhar com várias figuras dentro de uma tela de desenho em comum, sendo assim, a sobreposição de imagens seria algo imprescindível no desenvolvimento do projeto. E foi assim que nós optamos em trabalhar com a forma mais pura de sobreposição de imagens, que é usando o conceito de camadas que a vetorização nos proporciona.

Dentro do software, cada camada do desenho representa um tipo de molde, sendo assim cada molde é praticamente uma instância separada de todos os outros, e somente assim é possível tratarmos seus atributos, como posição e tamanho, separadamente, e sem nenhum tipo de interferência nos moldes adjacentes. Cada camada do desenho é dinâmica, e respondem a eventos do mouse, como `onClick`, e `onMouseOver`, assim, após a movimentação de um tipo de molde, a camada referente a esse desenho torna-se instantaneamente a primeira camada do conjunto, isso para proporcionar melhor visualização dos ajustes que serão feitos na figura, e é por essa característica, que foi necessário implementarmos ações de controle da ordem dessas camadas, como por exemplo, o botão “Molde anterior”, que tem a função de levar o atual molde escolhido para a posição diretamente anterior a sua inicial.

6.1.7. Vetorização

Quando se trabalha com o conceito de vetores em aplicação de desenho, cada objeto é uma entidade independente com propriedades, como tamanho e posição na tela, sendo essas, inclusas na definição de cada camada do desenho dentro do arquivo que gerencia todos esses objetos. Assim, no caso do IdentKit, optamos por usar o conceito de vetorização apenas para organizarmos dentro de uma tela de desenho todos os moldes faciais necessários para a produção de um retrato falado. Todo esse processo de gerenciamento dos moldes, tanto das figuras em si, como de seus atributos é feito por uma linguagem XML, que denomina-se, Scalable Vectorial Graphics, ou simplesmente, SVG.

6.1.8. SVG

SVG é uma linguagem que descreve um gráfico bi-dimensional dentro de um arquivo XML. Esse formato de arquivo permite o uso de três tipos gráficos de objetos, sendo eles:

- formas geométricas vectoriais, que nada mais são que descrições de pontos dentro de um ambiente, que ligadas por retas, forma um desenho vetorial;
- Textos em seu formato mais puro;
- E imagens raster.

Como descrito anteriormente, os moldes do IdentKit estão em formato PNG, portanto, são imagens rasters. Assim, a linguagem SVG faz todo o controle da adição e remoção dos moldes, através de ligações físicas entre os moldes do retrato falado. No momento da adição dos moldes, um atributo de identificação é passado, para futuramente, ser possível o resgate específico dessa camada do desenho.

Segue abaixo um exemplo de como fica a linha dentro do arquivo SVG, referente a inserção de um único molde.

```
<image x="100" y="50" width="400" xmlns:xlink="http://www.w3.org/1999/xlink"
xlink:href="../../temp/89.png" xlink:type="simple" height="600" id="Rosto"
preserveAspectRatio="none" />
```

Segue uma breve explicação de cada atributo a cima:

- <image>: Identifica que a linha representa um objeto imagem, e que terá um link para alguma imagem física, tanto em formato vetorial, quanto raster;
- X e Y : representa a posição horizontal e vertical do molde dentro da tela de desenho, o canto superior esquerdo da imagem é que dita a posição da imagem. Esse atributo recebe todos os valores inteiros, além de poder trabalhar com porcentagem.
- Width e Height: respectivamente, os valores de Largura e altura, correspondente ao da figura, Esses atributos só podem receber valores inteiros positivos, e assim como os anteriores, também podem trabalhar com valores descritos em porcentagem.
- xmlns:xlink : apenas um descritivo identificando o tipo de nó que será atribuído ao arquivo SVG
- xlink:href: Caminho que contém o arquivo físico em si, no caso do IdentKit, o lugar aonde o molde raster se encontra.
- xlink:type: Atributo que identifica o tipo de XLink utilizado, porém, em se tratando de imagens, esse valor será sempre “simple”;
- preserveAspectRatio: Define como será atribuído o ajuste do tamanho da imagem, se cada imagem irá aumentar proporcionalmente os valores de altura e largura, ou se eles irão trabalhar isolados, nesse casos nós optamos para o valor “none” em todos as figuras, pois assim temos maior liberdade em ajustar um molde específico atribuindo valores isolados, e não proporcionais.

O uso da linguagem SVG na aplicação ampliou as possibilidades de melhorias no software. Se compararmos alguns produtos no mercado que trabalham diretamente com figuras raster, o IdentKit se mostra superior a usar conceitos de vetorização. O exemplo mais pertinente a isso, é a liberdade de se poder aumentar ou diminuir uma imagem sem perder qualidade, e mesmo após um retrato ser salvo, todo ele pode ser reajustado sem nenhum tipo de dificuldade e obtendo um produto final exatamente igual se o fizesse desde o começo.

6.1.9. Pesquisa de Moldes

No momento do cadastro de um molde, é possível atribuir características a essa peça do retrato falado, para que assim seja possível uma pesquisa dentro de um

banco de dados vasto. Atualmente, é possível apenas a escolha de uma característica dentro da tela escolha, aonde todos os moldes com essa característica serão apresentados.

6.1.10. Relatórios

Juntamente a tela de configuração dos moldes, o IdentKit conta também com uma tela de cadastro, onde dados básicos relatados pela vítima serão prontamente preenchidos, aonde serão anexados junto ao relatório final que poderá ser impresso. O formato da impressão pode ser visto na figura 1.3.

6.1.11. Banco de Dados – PostgreSQL 8.1

O Banco de dados na aplicação tem a funcionalidade de guardar todas as informações referente aos retratos cadastrados, e isso se aplica desde aos campos de preenchimento de cadastro, até os arquivos SVG contendo todas as informações pertinentes ao desenho em si, e os próprios moldes em formato PNG. É por essa característica de trabalhar amplamente com arquivos do tipo ByteA que é necessário um acesso com resposta rápida e confiável do BD.

7. DIAGRAMAS

7.1. DIAGRAMA DE CASOS DE USO

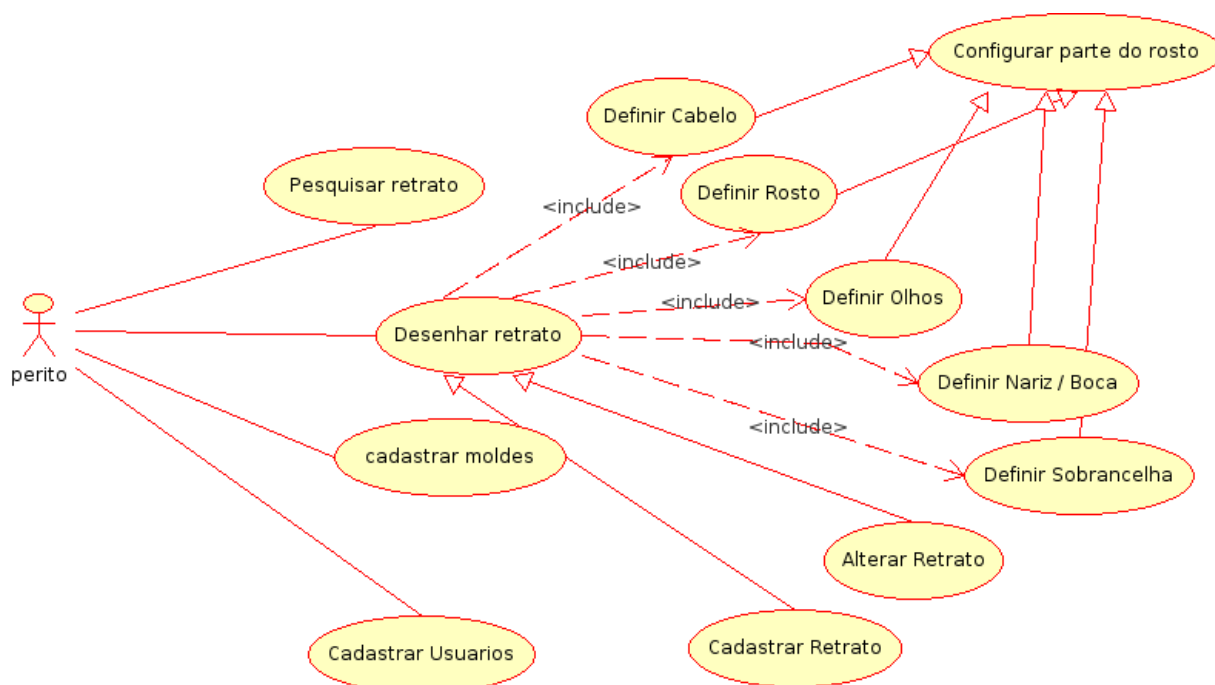


Fig 3 – Diagrama de Casos de Uso

7.2. ESPECIFICAÇÃO DE CASOS DE USO

DESENHAR RETRATO

Descrição

Este Use Case serve para a Tela de Desenho.

Pré-condições

Este use case pode iniciar somente se:

- Sistema estiver disponível;
- Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

- Ter salvo o retrato;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito abre a tela de desenho.

Fluxo principal

- O Sistema apresenta a tela de desenho; (T1);
- Perito trabalha com os moldes;(A1),(A2),(A3),(A4),(A5);
- Perito informa os dados do retratado na Guia Cadastro e clica em salvar; (E1),(A6),(A7),(A8),(A9),(A10),(A11)
- O Sistema salva as alterações e apresenta a mensagem informado o sucesso na alteração do registro.
- O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito adiciona Cabelo.

- Executa os procedimentos descritos no caso de uso Tela Desenho – Guia Cabelo;
- Retorna ao passo 2 do fluxo principal.

A2. Perito adiciona Nariz / Boca.

1. Executa os procedimentos descritos no caso de uso Tela Desenho – Guia Cabelo;
2. Retorna ao passo 2 do fluxo principal.

A3. Perito adiciona Olho.

- Executa os procedimentos descritos no caso de uso Tela Desenho – Guia Olho;
- Retorna ao passo 2 do fluxo principal.

A4. Perito adiciona Rosto.

1. Executa os procedimentos descritos no caso de uso Tela Desenho – Guia Rosto;
2. Retorna ao passo 2 do fluxo principal.

A5. Perito adiciona Sobrancelha.

1. Executa os procedimentos descritos no caso de uso Tela Desenho – Guia Sobrancelha;
2. Retorna ao passo 2 do fluxo principal.

A6. Perito clica em abrir.

- Executa os procedimentos descritos no caso de uso Tela Desenho – Abrir Retrato;
- O Caso de uso é finalizado.

A7. Perito clica em Novo.

- Executa os procedimentos descritos no caso de uso Tela Desenho – Novo Retrato;
- O Caso de uso é finalizado.

A8. Perito clica em Imprimir.

- Executa os procedimentos descritos no caso de uso Tela Desenho – Imprimir Retrato;
- O Caso de uso é finalizado.

A9. Perito clica em Cadastrar Molde.

- Executa os procedimentos descritos no caso de uso Cadastrar Molde;
- O Caso de uso é finalizado.

A10. Perito clica em Cadastrar Usuário.

- Executa os procedimentos descritos no caso de uso Cadastrar Usuário;
- O Caso de uso é finalizado.

A11. Perito clica em Sair.

1. Encerra a aplicação;
2. O Caso de uso é finalizado.

Exceções**E1. Perito não preenche um dos campos da aba cadastro e clica em salvar.**

- O Sistema informa o campo sem preenchimento;
- Retorna ao passo 2 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Desenho

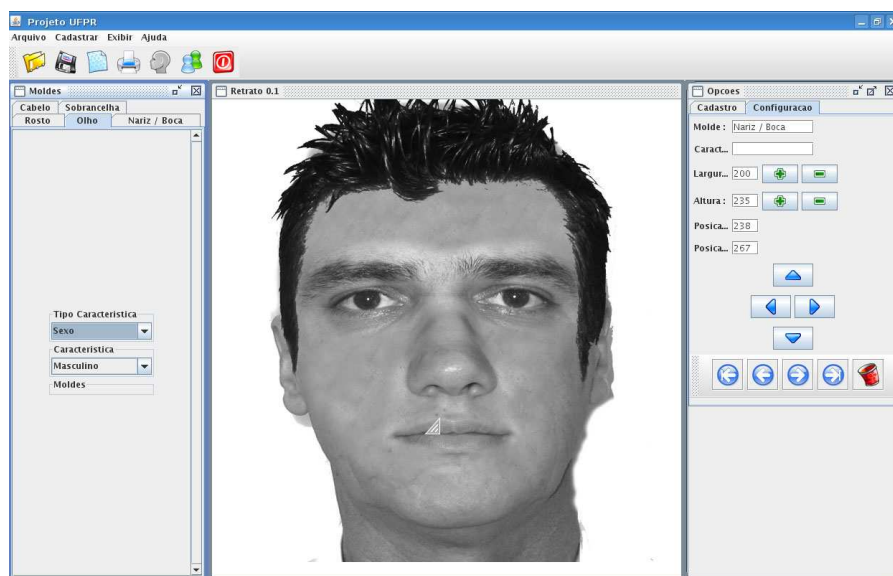


Fig. 4- Tela Desenho

UC – Definir Cabelo

Descrição

Este Caso de Uso serve para guiar o usuário na definição do cabelo.

Pré-condições

Este Caso de Uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Usuário logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os moldes do tipo cabelo existentes no sistema de acordo com os parâmetros informados pelo perito e adicionar o molde na área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Guia Cabelo” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Cabelo; (T1);
2. Perito seleciona o tipo de característica e a característica;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema adiciona o molde na área de desenho; (T2);
6. O Sistema executa o UC Configurar Molde;
7. O Caso de uso é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Desenho – Guia Cabelo



Fig. 5- Guia Cabelo

Tela 2: Tela Desenho – Cabelo

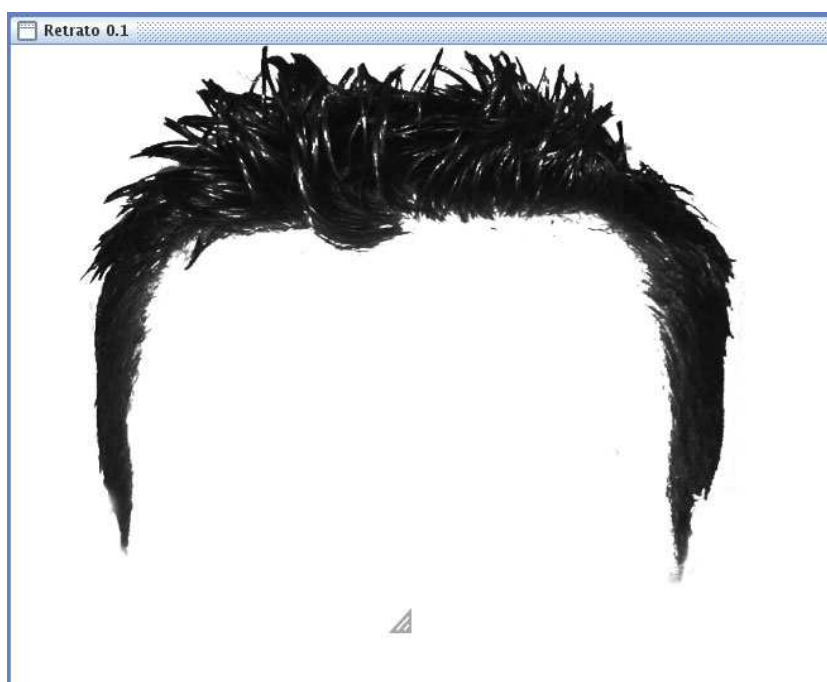


Fig. 6- Cabelo

UC – Definir Nariz Boca

Descrição

Este Use Case serve para guia Nariz / Boca.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os moldes existentes no sistema de acordo com os parâmetros informados pelo perito e adicionar o molde na área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Guia Nariz/Boca” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Nariz / Boca; (T1);
2. Perito seleciona o tipo de característica e a característica;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema adiciona o molde na área de desenho; (T2);
6. O Sistema executa o UC Configurar Molde;
7. O Caso de uso é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Desenho - Guia Nariz / Boca



Fig. 7- Guia Nariz/ Boca

Tela 2: Tela Desenho - Nariz / Boca

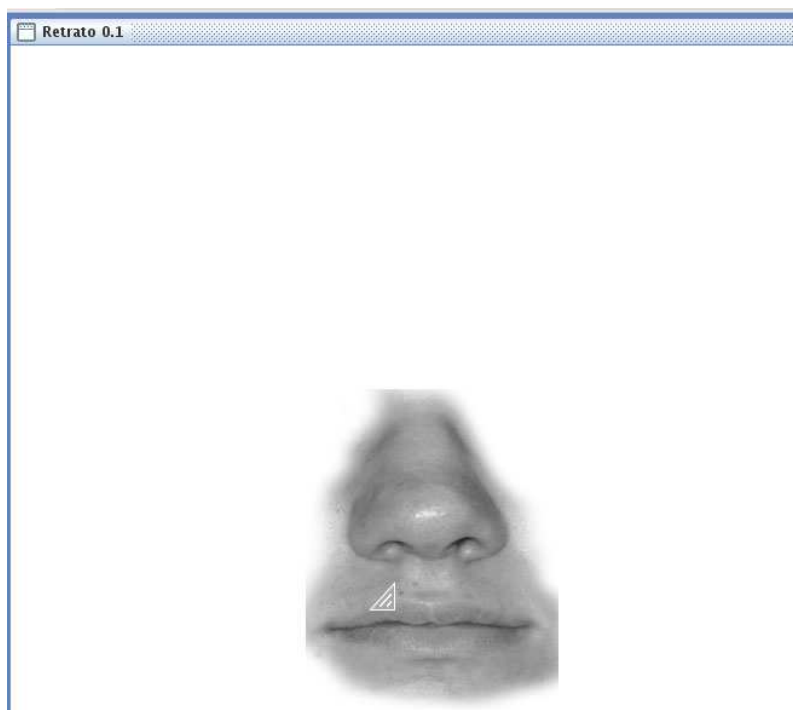


Fig. 8- Nariz/Boca

UC – Definir Olho

Descrição

Este Use Case serve para guia Olho.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os moldes existentes no sistema de acordo com os parâmetros informados pelo perito e adicionar o molde na área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Guia Olho” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Olho; (T1);
2. Perito seleciona o tipo de característica e a característica;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema adiciona o molde na área de desenho; (T2);
6. O Sistema executa o UC Configurar Molde;
7. O Caso de uso é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

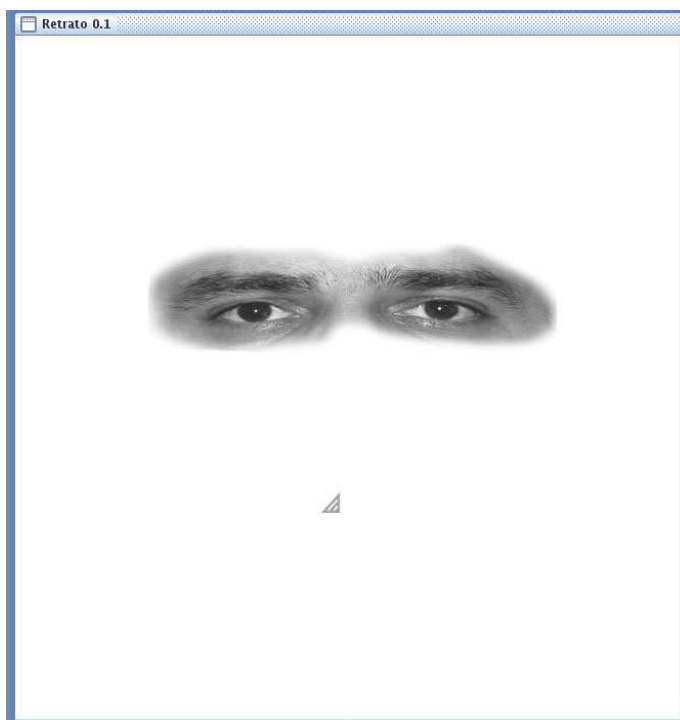
Não se aplica.

Tela:

Tela 1: Tela Desenho - Guia Olho



Fig. 9- Guia Olho

Tela 2: Tela Desenho – Olho**Fig. 10- Olho****UC – Definir Rosto****Descrição**

Este Use Case serve para guia Rosto.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os moldes existentes no sistema de acordo com os parâmetros informados pelo perito e adicionar o molde na área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Rosto” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Rosto; (T1);
2. Perito seleciona o tipo de característica e a característica;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema adiciona o molde na área de desenho (T2);
6. O Sistema executa o UC Configurar Molde;
7. O Caso de uso é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Desenho - Guia Rosto

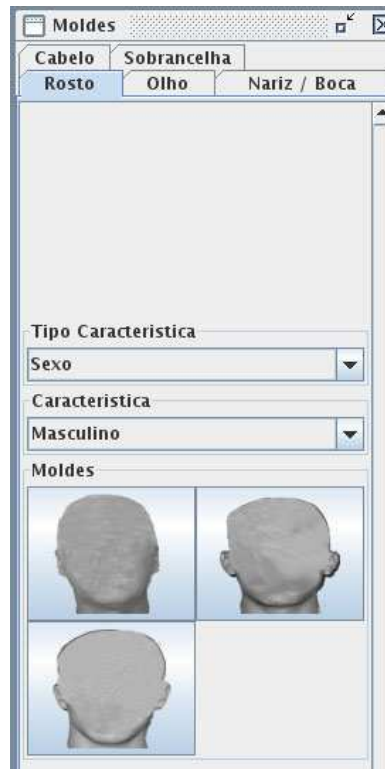


Fig. 11- Guia Rosto

Tela 2: Tela Desenho – Rosto

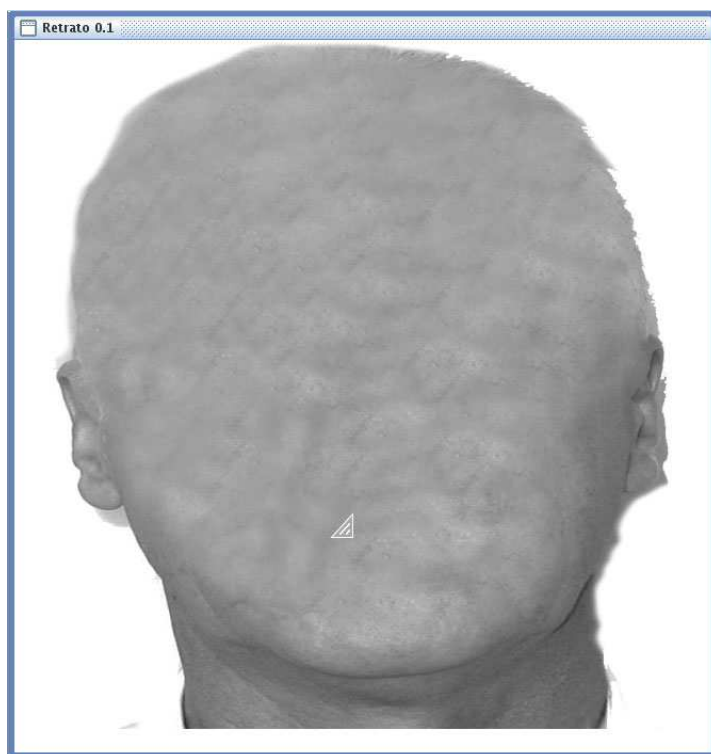


Fig. 12- Rosto

UC – Definir Sobrancelha

Descrição

Este Use Case serve para guia Sobrancelha.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os moldes existentes no sistema de acordo com os parâmetros informados pelo perito e adicionar o molde na área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Sobrancelha” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Sobrancelha; (I1);
2. Perito seleciona o tipo de característica e a característica;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema adiciona o molde na área de desenho (I2);
6. O Sistema executa o UC Configurar Molde;
7. O Caso de uso é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Desenho - Guia Sobrancelha

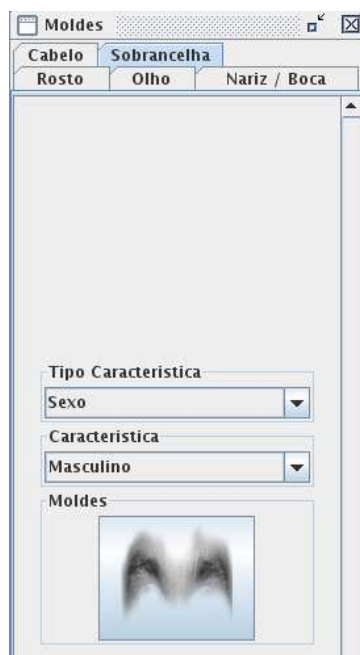


Fig. 13- Guia Sobrancelha

Tela 2: Tela Desenho – Sobrancelha

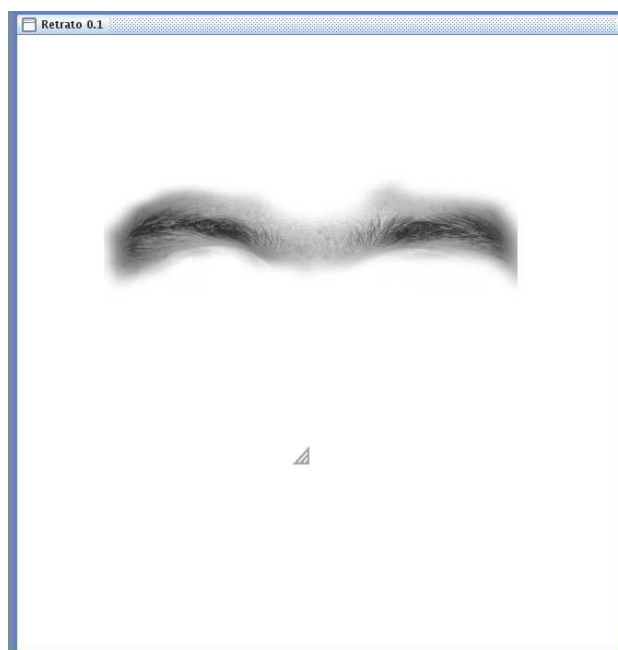


Fig. 14- Sobrancelha

UC – Cadastrar Retrato

Descrição

Este Caso de Uso serve para obter os dados do retratado.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
- Perito logado no sistema;

Pós-condições

Após o fim normal deste caso de Uso o sistema deve:

1. Ter cadastrado um novo molde no sistema de acordo com os parâmetros informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clicar na Guia “Cadastro” na tela de Desenho.

Fluxo principal

1. O Sistema apresenta a Guia Cadastro da tela de desenho; (T1);
2. Perito informa os dados;
3. O Sistema salva os dados informados e apresenta a mensagem confirmando a inserção dos dados realizadas com sucesso; (E1);
4. O Use Case é finalizado.

Fluxo alternativo

Não se aplica.

Exceções**E1. Perito não preenche um dos campos da tela e clica em salvar.**

1. O Sistema informa o campo sem preenchimento;
2. Retorna ao passo 2 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:**Tela 1: TelaCadastro**

The image shows a screenshot of a software window titled "Opcoes". It has two tabs: "Cadastro" (selected) and "Configuracao". The "Cadastro" tab contains the following fields:

- Nome: [Text Input]
- Apelido: [Text Input]
- Altura: [Dropdown Menu: --Selecione--]
- Peso: [Dropdown Menu: --Selecione--]
- Cor Da Pele: [Dropdown Menu: --Selecione--]
- Idade: [Dropdown Menu: --Selecione--]
- Sexo: Masculino Feminino
- Obs: [Text Area]

Fig. 15- Tela Cadastro

UC – Configurar Molde**Descrição**

Este Caso de Uso serve para guia Configurar Molde.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Um molde da área de desenho selecionado;
3. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Deixar o molde da largura, altura e na posição desejada, também poderá selecionar outros moldes da área de desenho;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Guia Configuração” na tela de Desenho, com um molde selecionado.

Fluxo principal

1. O Sistema apresenta a tela Desenho – Configuração; (T1);
2. Perito ajusta a largura, altura e posição do molde (A1),(A2),(A3),(A4),(A5);
3. O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito clica no Botão Primeiro (|<).

1. O Sistema envia o molde selecionado para a primeira camada ;
2. Retorna ao passo 2 do fluxo principal.

A2. Perito clica no Botão Anterior (<).

1. O Sistema envia o molde selecionado para a camada anterior;
2. Retorna ao passo 2 do fluxo principal.

A3. Perito clica no Botão Próximo(>).

1. O Sistema envia o molde selecionado para a próxima camada;
2. Retorna ao passo 2 do fluxo principal.

A4. Perito clica no Botão Último(>|).

1. O Sistema envia o molde selecionado para a ultima camada;
2. Retorna ao passo 2 do fluxo principal.

A5. Perito clica no ícone de exclusão.

1. O Sistema remove o molde selecionado da área de desenho;
2. Retorna ao passo 2 do fluxo principal.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

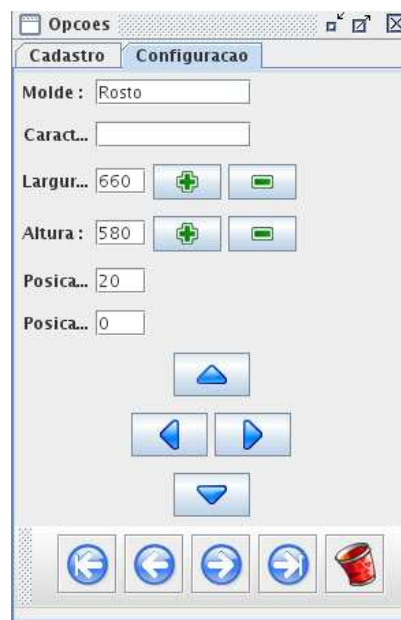
Tela:**Tela 1: Tela Desenho – Guia Configuração**

Fig. 16- Guia Configuração

UC – Imprimir Retrato

Descrição

Este Caso de Uso serve para imprimir retrato.

Pré-condições

Este Caso de Uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Imprimir o retrato;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clicar no ícone “Imprimir” na tela de desenho.

Fluxo principal

1. O Sistema apresenta a tela de desenho; (T1);
2. Perito clica imprimir;
3. O Sistema abre uma tela para visualizar a impressão;
4. O Use Case é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: TelaDesenho – Imprimir

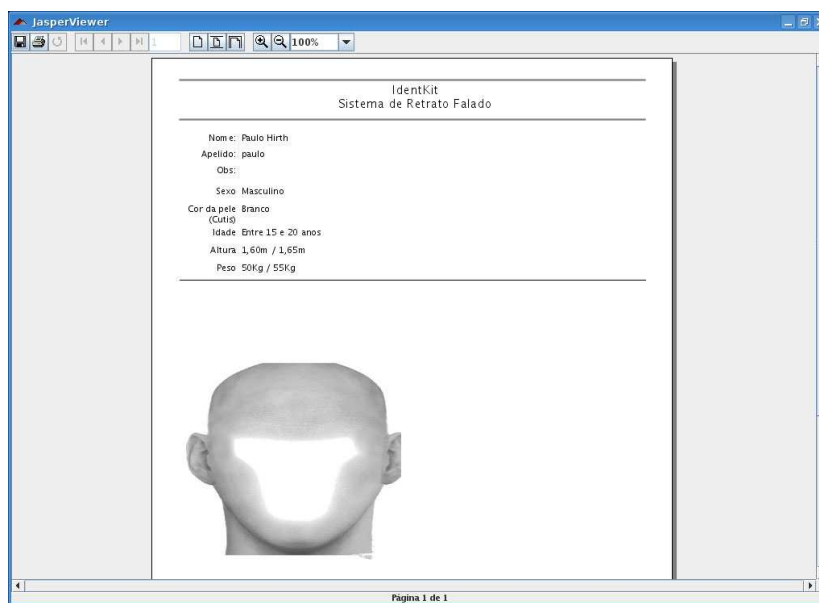


Fig. 17- Relatório

UC – Novo Retrato

Descrição

Este Caso de Uso serve para novo retrato.

Pré-condições

Este Caso de Uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Limpar a área de desenho para cadastrar um novo retrato;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clicar no ícone “Novo” na tela de desenho.

Fluxo principal

1. O Sistema apresenta a tela de desenho; (T1);
2. Perito clica novo;
3. O Sistema limpa a área de desenho, as guias de moldes e a guia de cadastro;
4. O Use Case é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: TelaDesenho



Fig. 18- Layout

UC – Pesquisar Retrato

Descrição

Este Caso de Uso serve para pesquisar retrato.

Pré-condições

Este Caso de Uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Listar os retratos existentes no sistema de acordo com os parâmetros de pesquisa informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clica no ícone “Abrir” na Tela de Desenho.

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de retrato; (T1);
2. Perito informa o usuário e/ou data do cadastro para filtro e clica em pesquisar; (E1);
3. O Sistema apresenta os retratos de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Use Case é finalizado.

Fluxo alternativo

Não se aplica.

Exceções**E1. Nenhum registro encontrado.**

1. O sistema apresenta a mensagem informando que não foram encontrados registros para os parâmetros informados pelo perito;
2. O Use Case é reiniciado.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Pesquisa

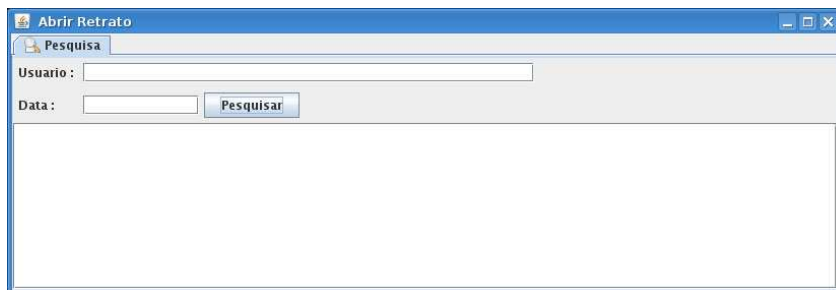


Fig. 19- Pesquisa

UC – Alterar Usuário

Descrição

Este Use Case serve para alterar usuários

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
- Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Alterar os dados do usuário do sistema;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Pesquisar” na tela usuário

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de usuário; (T1);
2. Usuário informa os dados para filtro e clica em pesquisar;
3. O Sistema apresenta os usuários de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no nome do usuário; (A1);
5. O Sistema apresenta os dados na tela de alteração (T2);
6. O Usuário altera os dados da tela de alteração e clica em Salva; (E1),(A2), (A3), (A4);
7. O Sistema salva as alterações e apresenta a mensagem informado o sucesso na alteração do registro.
8. O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito clica no nome do Usuário

1. Executa os procedimentos descritos no caso de uso de inclusão;
2. O Caso de uso é finalizado.

A2. Perito clica no ícone de exclusão.

1. Executa os procedimentos descritos no caso de uso de exclusão;
2. O Caso de uso é finalizado.

A3. Perito clica em Pesquisar.

1. Executa os procedimentos descritos no caso de uso de pesquisa de usuários;
2. O Caso de uso é finalizado.

A4. Perito clica em Limpar.

1. Executa os procedimentos descritos no caso de uso de inclusão;
2. Retorna ao passo 6 do fluxo principal.

Exceções**E1. Perito não preenche um dos campos da tela e clica em salvar.**

1. O Sistema informa o campo sem preenchimento;
2. Retorna ao passo 6 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:**Tela 1: Pesquisar**

Fig. 20- Pesquisar

Tela 2: Alterar



Fig. 21- Alterar

UC – Excluir Usuário

Descrição

Este Use Case serve para excluir usuários

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Excluir o usuário do sistema;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Pesquisar” na tela de usuário

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de usuários; (T1);
2. Usuário informa os dados para filtro e clica em pesquisar; (A1)
3. O Sistema apresenta os usuários de acordo com os parâmetros de pesquisa informados pelo usuário;
4. O Perito clica no nome do usuário;
5. O Sistema apresenta os dados na tela de alteração (T2);
6. O Usuário clica no ícone delete; (A1), (E1);
7. O Sistema excluir o registro solicitado e apresenta a mensagem informado o sucesso na exclusão do registro.
8. O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito clica em Limpar.

1. Executa os procedimentos descritos no caso de uso de inclusão.
2. O Caso de uso é finalizado.

Exceções

E1. Nenhum registro encontrado.

1. O sistema apresenta a mensagem informando que não foram encontrados registros para os parâmetros informados pelo usuário;
2. O Use Case é reiniciado.

Regras de negócio

R1: O Sistema não deverá permitir a exclusão de registros que contenham retratos cadastrados.

Tela:

Tela 1: Pesquisar



Fig. 22- Pesquisa Usuários

Tela 2: Excluir



Fig. 23- Excluir

UC – Incluir Usuário

Descrição

Este Use Case serve para cadastrar usuários

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Ter cadastrado um novo usuário no sistema de acordo com os parâmetros informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clicar em “Salvar” na tela de Usuários

Fluxo principal

1. O Sistema apresenta a tela de cadastro de usuários; (T1);
2. Perito informa os dados e clica salvar; (A1);
3. O Sistema salva os dados informados e apresenta a mensagem confirmando a inserção dos dados realizadas com sucesso; (E1,E2);
4. O Use Case é finalizado.

Fluxo alternativo

A1. Perito clica em Pesquisar.

1. O Sistema limpa os campos da tela;
2. Retorna ao passo 2 do fluxo principal.

Exceções

E1. Perito não preenche um dos campos da tela e clica em salvar.

1. O Sistema informa o campo sem preenchimento;

2. Retorna ao passo 2 do fluxo principal.

E2. Perito informa login já cadastrado e clica em salvar.

1. O Sistema informa que o login já está cadastrado e limpa o campo;
2. Retorna ao passo 2 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:

Tela 1: TelaUsuario.java



Fig. 24- Tela Usuário Java

UC – Pesquisar Usuário

Descrição

Este Use Case serve para pesquisar usuários

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Listar os usuários existentes no sistema de acordo com os parâmetros de pesquisa informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Pesquisar” na tela de usuário

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de usuários; (T1);
2. Usuário informa os dados para filtro e clica em pesquisar;
3. O Sistema apresenta os usuários de acordo com os parâmetros de pesquisa informados pelo usuário;
4. O Use Case é finalizado.

Fluxo alternativo

Não se aplica.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Pesquisa



Fig. 25- Pesquisa Usuário2

UC – Incluir Molde

Descrição

Este Use Case serve para cadastrar molde.

Pré-condições

Este use case pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Ter cadastrado um novo molde no sistema de acordo com os parâmetros informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O perito clicar em “Salvar” na tela de Moldes.

Fluxo principal

1. O Sistema apresenta a tela de cadastro de moldes; (T1);
2. Perito informa os dados e clica salvar; (A1),(A2),(A3),(A4);
3. O Sistema salva os dados informados e apresenta a mensagem confirmando a inserção dos dados realizadas com sucesso; (E1);
4. O Use Case é finalizado.

Fluxo alternativo

A1. Perito clica em Limpar.

1. O Sistema limpa os campos da tela;
2. Retorna ao passo 2 do fluxo principal.

A2. Perito clica no ícone Pesquisar.

1. Executa os procedimentos descritos no caso de uso de pesquisa de moldes;
2. O Caso de uso é finalizado.

A3. Perito clica no ícone Limpar.

1. Executa os procedimentos descritos no caso de uso de inclusão;
2. Retorna ao passo 6 do fluxo principal.

A4. Perito clica no ícone Sair.

1. Fecha a tela de molde
2. O Caso de uso é finalizado.

Exceções

E1. Perito não preenche um dos campos da tela e clica em salvar.

1. O Sistema informa o campo sem preenchimento;
2. Retorna ao passo 2 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Tela Molde

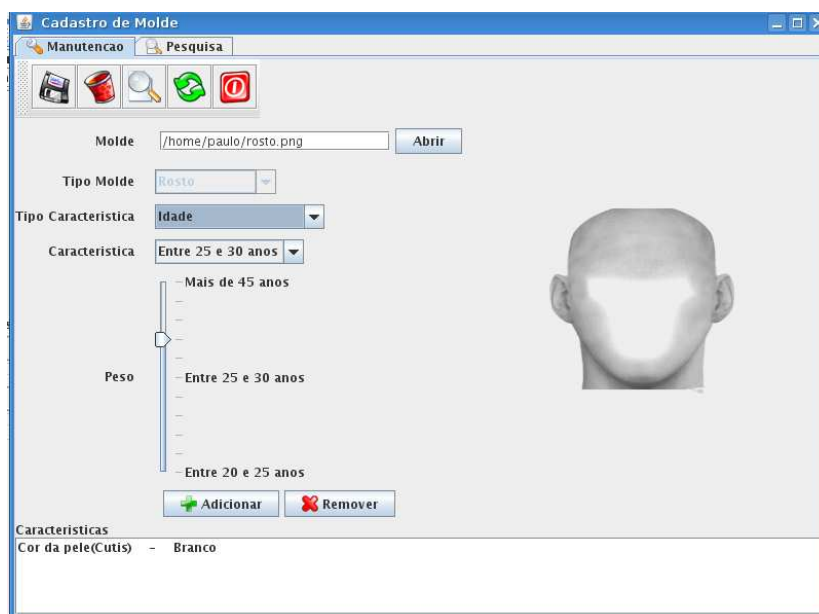


Fig. 26- Tela Molde

UC – Alterar Molde

Descrição

Este Caso de uso serve para alterar moldes.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Alterar os dados do molde do sistema;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Pesquisar” na tela molde.

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de moldes; (T1);
2. Perito informa os dados para filtro e clica em pesquisar;
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;
5. O Sistema apresenta os dados na tela de alteração (T2);
6. O Usuário altera os dados da tela de alteração e clica em Salvar; (E1), (A1),(A2), (A3),(A4);
7. O Sistema salva as alterações e apresenta a mensagem informado o sucesso na

alteração do registro.

8. O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito clica no ícone de exclusão.

1. Executa os procedimentos descritos no caso de uso Excluir Molde;
2. O Caso de uso é finalizado.

A2. Perito clica no ícone Pesquisar.

1. Executa os procedimentos descritos no caso de uso Pesquisar Moldes;
2. O Caso de uso é finalizado.

A3. Perito clica no ícone Limpar.

1. Executa os procedimentos descritos no caso de uso de inclusão;
2. Retorna ao passo 6 do fluxo principal.

A4. Perito clica no ícone Sair.

1. Fecha a tela de cadastro de molde
2. O Caso de uso é finalizado.

Exceções

E1. Perito não preenche um dos campos da tela e clica em salvar.

1. O Sistema informa o campo sem preenchimento;
2. Retorna ao passo 6 do fluxo principal.

Regras de negócio

Não se aplica.

Tela:

Tela 1: Pesquisar

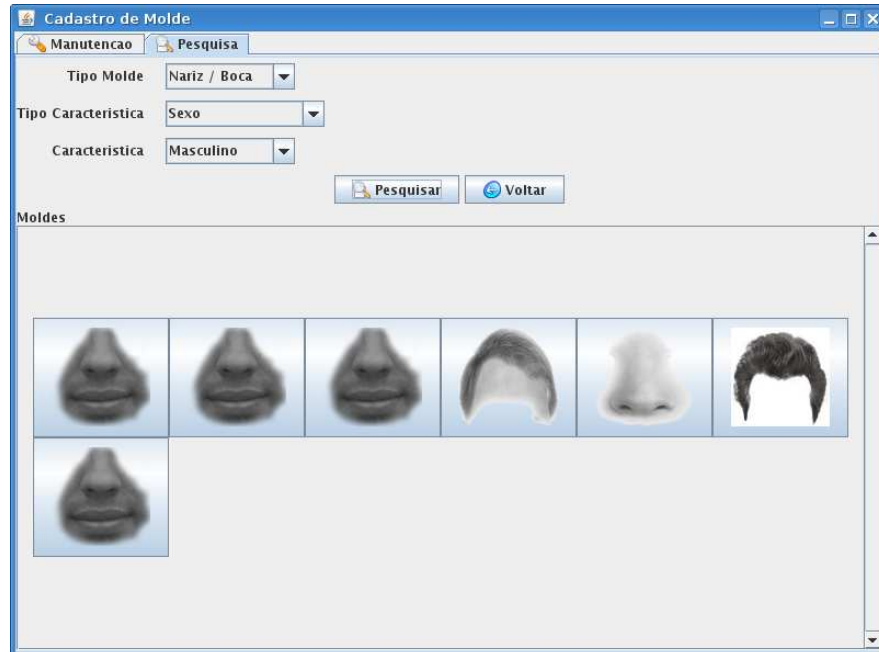


Fig. 27- Lista Pesquisa Moldes

Tela 2: Alterar

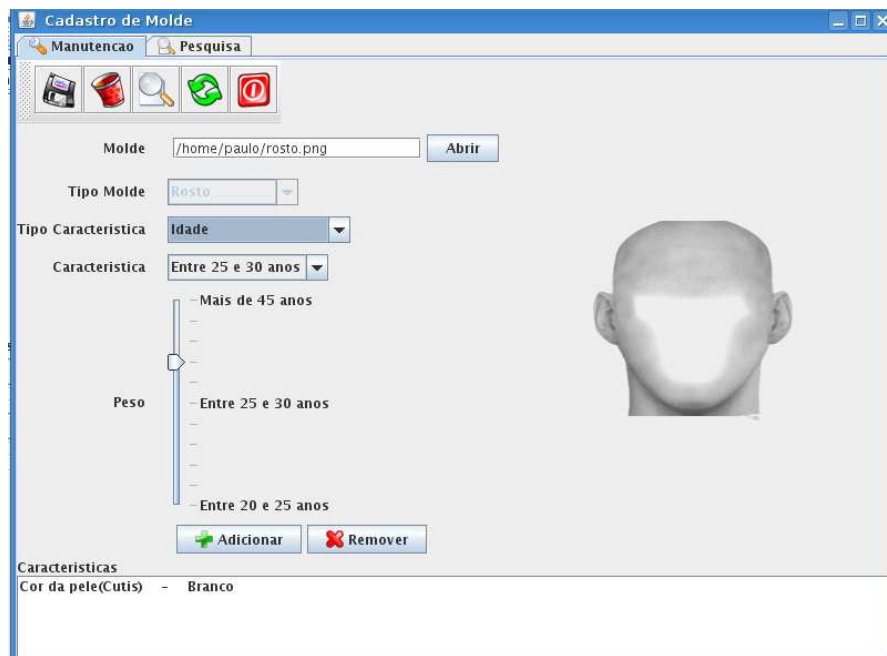


Fig. 28- Alterar

UC – Excluir Molde

Descrição

Este Caso de Uso serve para excluir molde.

Pré-condições

Este caso de uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste use case o sistema deve:

1. Excluir o molde do sistema;

Ator

Perito

Estímulo

Este caso de uso deve começar quando:

O usuário Seleciona “Pesquisar” na tela de molde.

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de moldes; (T1);
2. Perito informa os dados para filtro e clica em pesquisar; (A1)
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Perito clica no molde;

5. O Sistema apresenta os dados na tela de alteração (T2);
6. O Usuário clica no ícone delete; (A1), (A2),(A3),(E1);
7. O Sistema excluir o registro solicitado e apresenta a mensagem informado o sucesso na exclusão do registro.
8. O Caso de uso é finalizado.

Fluxo alternativo

A1. Perito clica em Limpar.

1. Executa os procedimentos descritos no caso de uso Incluir Molde.
2. O Caso de uso é finalizado.

A2. Perito clica no ícone Pesquisar.

1. Executa os procedimentos descritos no caso de uso Pesquisa Moldes;
2. O Caso de uso é finalizado.

A3. Perito clica no ícone Sair.

1. Fecha a tela de molde.
2. O Caso de uso é finalizado.

Exceções

E1. Nenhum registro encontrado.

1. O sistema apresenta a mensagem informando que não foram encontrados registros para os parâmetros informados pelo usuário;
2. O Caso de Uso é reiniciado.

Regras de negócio

R1: O Sistema não deverá permitir a exclusão de registros que contenham retratos cadastrados.

Tela:

Tela 1: Pesquisar

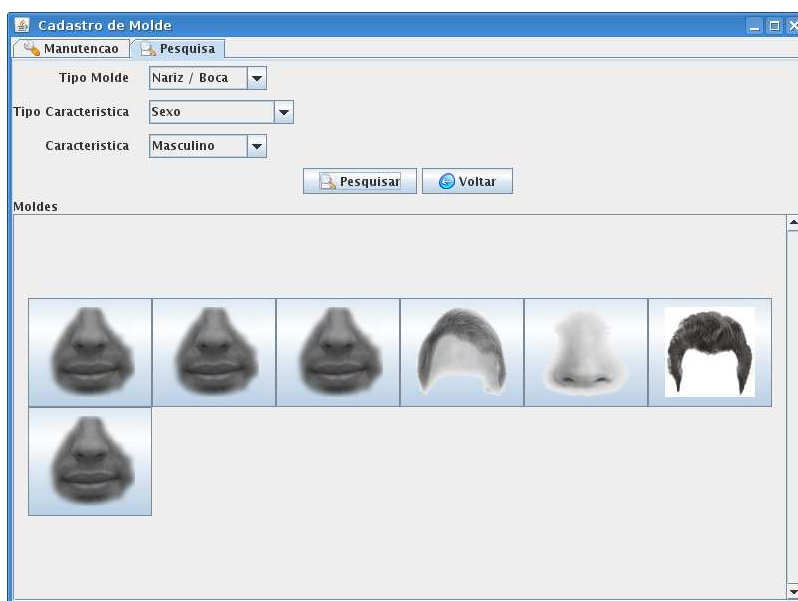


Fig. 29- Lista Pesquisa Moldes2

Tela 2: Excluir

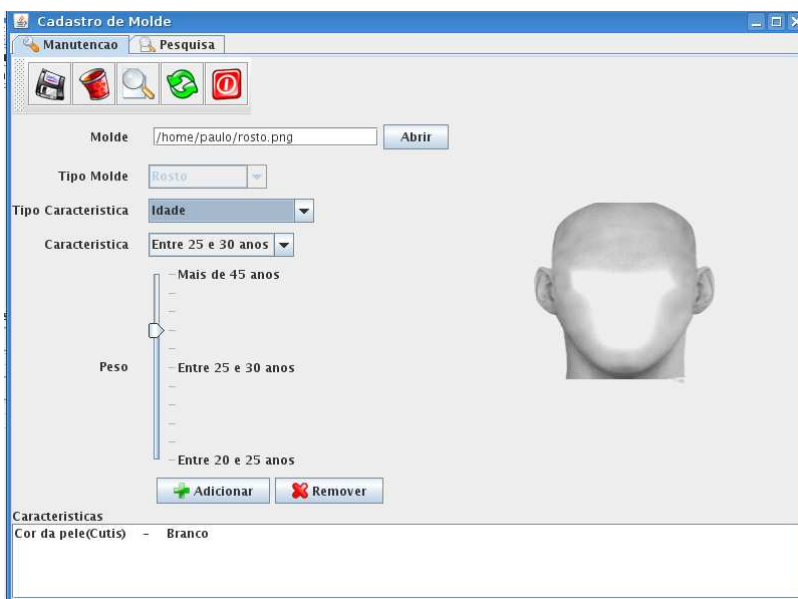


Fig. 30- Excluir Molde

UC – Pesquisar Molde

Descrição

Este Caso de Uso serve para pesquisar molde.

Pré-condições

Este Caso de Uso pode iniciar somente se:

1. Sistema estiver disponível;
2. Perito logado no sistema;
3. Perito tiver permissão de acesso;

Pós-condições

Após o fim normal deste caso de uso o sistema deve:

1. Listar os moldes existentes no sistema de acordo com os parâmetros de pesquisa informados pelo perito;

Ator

Perito

Estímulo

Este use case deve começar quando:

O usuário Seleciona “Pesquisar” na tela de moldes.

Fluxo principal

1. O Sistema apresenta a tela de pesquisa de moldes; (T1);
2. Perito informa os dados para filtro e clica em pesquisar; (A1);
3. O Sistema apresenta os moldes de acordo com os parâmetros de pesquisa informados pelo perito;
4. O Use Case é finalizado.

Fluxo alternativo

A1. Perito clica no ícone voltar

1. Executa os procedimentos descritos no caso de uso de inclusão;
2. O caso de uso é finalizado.

Exceções

Não se aplica.

Regras de negócio

Não se aplica.

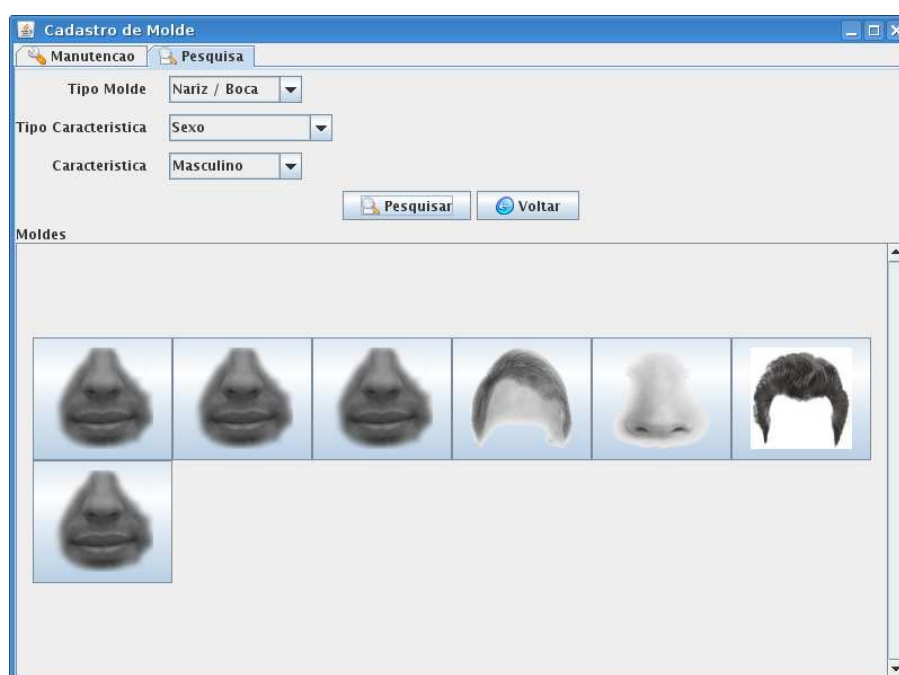
Tela:**Tela 1: Pesquisa**

Fig. 31- Pesquisa Moldes 3

7.3. DIAGRAMA DE CLASSES

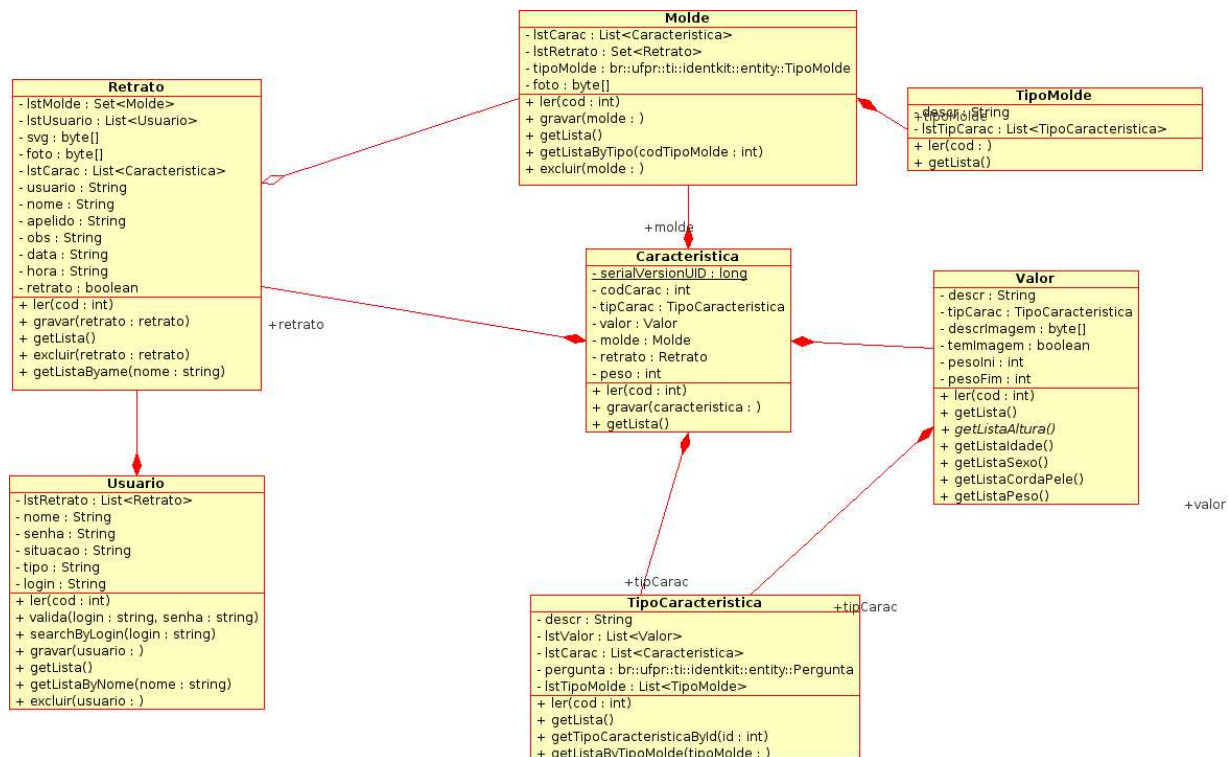


Fig. 32 -Diagrama de Classes

7.4. DIAGRAMA DE COMPONENTES

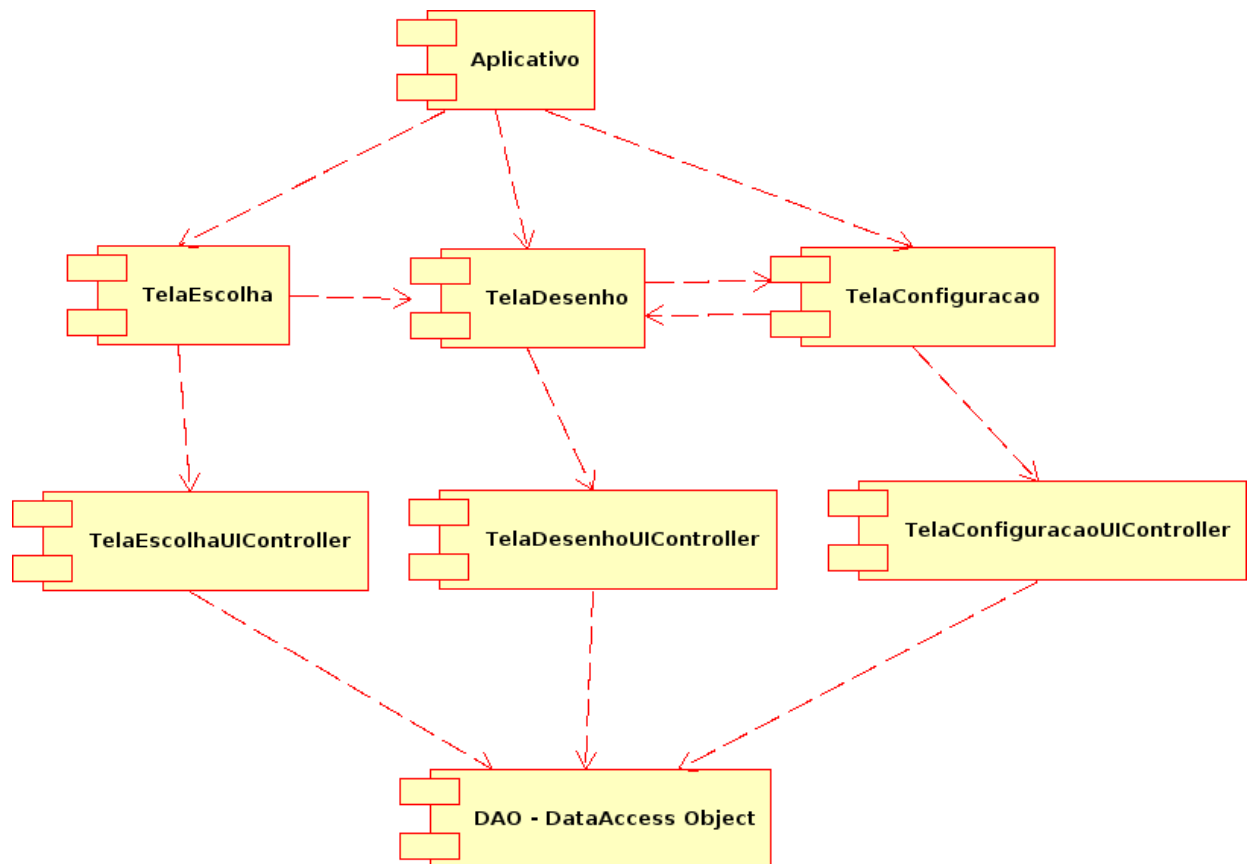


Fig.31- Diagrama de Componentes

7.5. DIAGRAMA DE ATIVIDADES – Definir Parte do Rosto

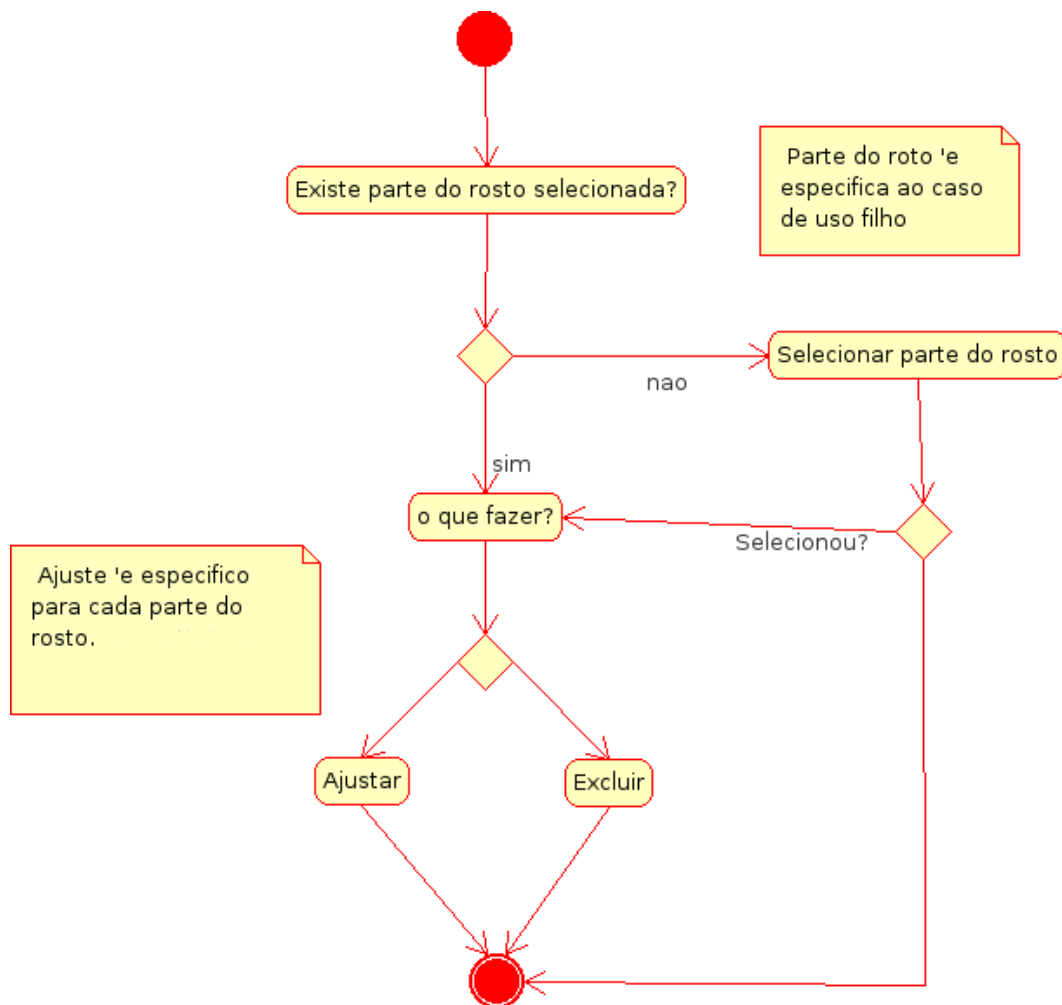


Fig.34- Diagrama de Atividades

7.6. DIAGRAMAS DE SEQÜÊNCIA TELA ESCOLHA

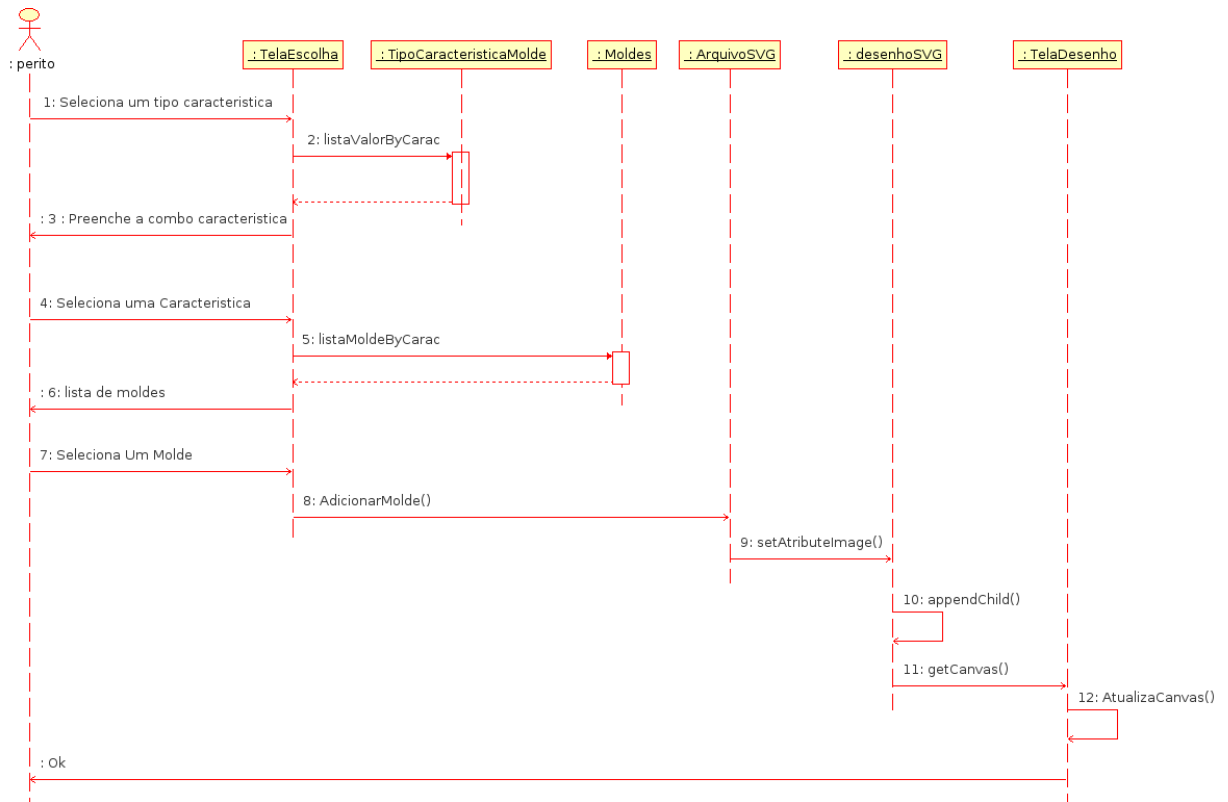


Fig. 35- Diagrama de Seqüência Tela Escolha

7.7. DIAGRAMAS DE SEQÜÊNCIA TELA CONFIGURAÇÃO

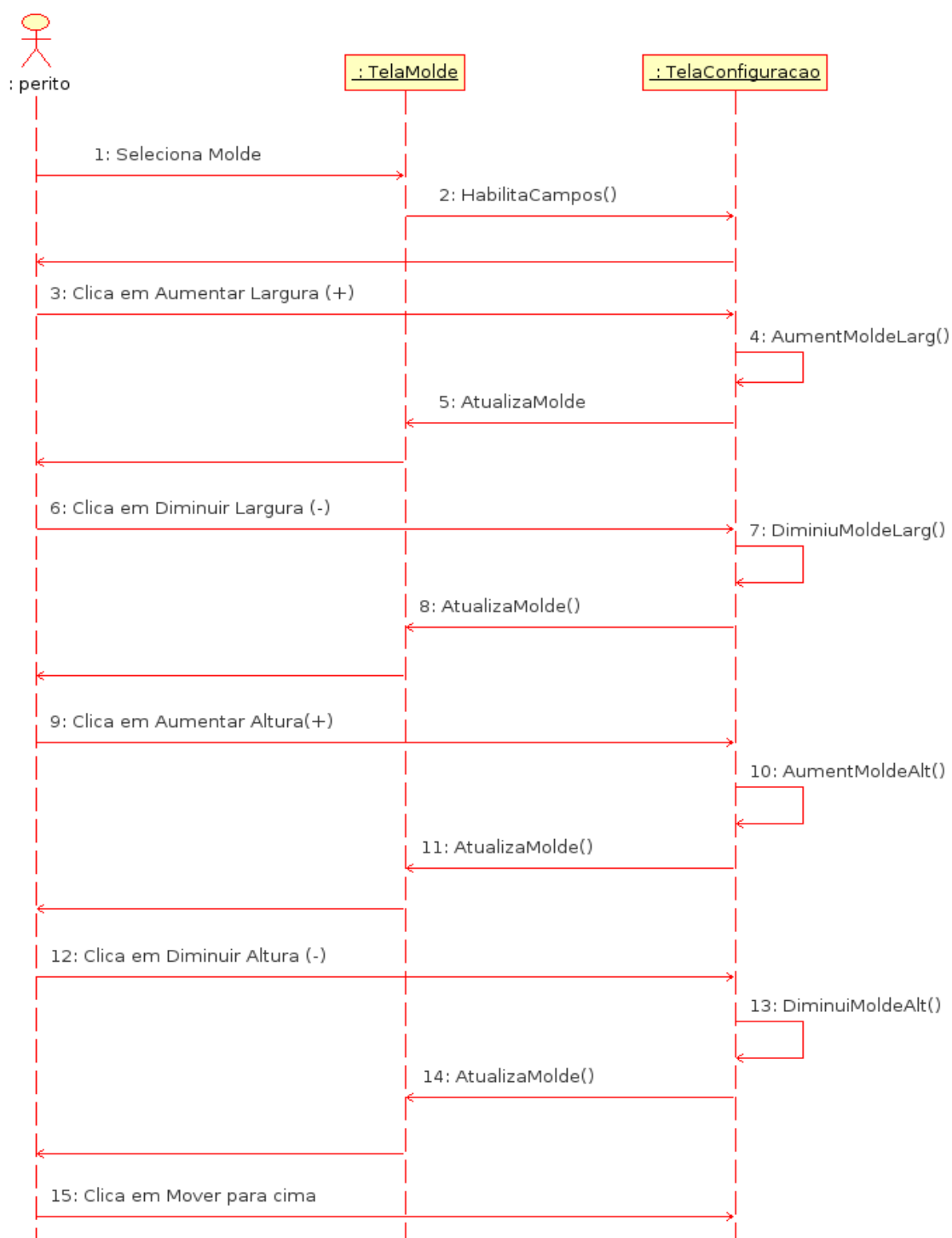


Fig. 36- Diagrama de Seqüência Tela Configuração

7.8. DIAGRAMA ENTIDADE RELACIONAMENTO

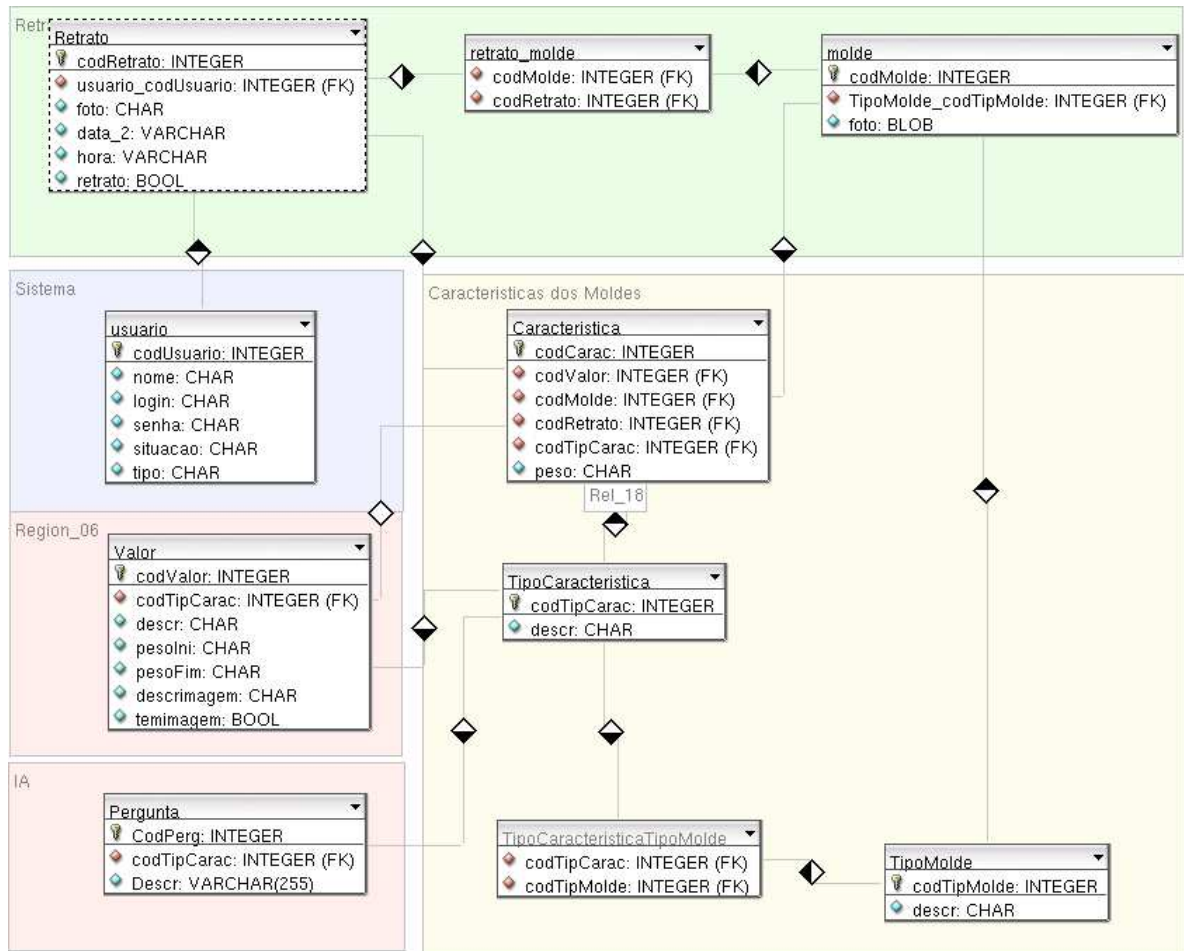


Fig. 37. Diagrama Entidade Relacionamento

7.9. DICIONÁRIO DE DADOS

TABELA		Usuario		
DESCRIÇÃO		Tabela que armazena os usuários do sistema		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO	
cod_usuario	serial	Primary Key Not Null	Código único identificador do usuário	
nome	CHAR		Nome completo do Usuário	
login	CHAR		Apelido utilizado para acessar o sistema	
senha	CHAR		Combinação para acessar o sistema	
situacao	CHAR		Status do usuário	
tipo	CHAR		Indicativa se o usuário é administrador ou usuário normal	

Tabela 1 – Descrição da Tabela Usuario

TABELA		Retrato		
DESCRIÇÃO		Tabela que contém os retratos do sistema.		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO	
cod_retrato	serial	Primary Key Not Null	Código único identificador do retrato	
Usuário_cod_usuario	INTEGER	Foreign Key	Código de referência a tabela usuario	
svg	CHAR		Contém o arquivo svg manipulado na aplicação	
foto	CHAR		Contém foto de pessoas cadastradas no sistema	
data	VARCHAR		Contém a data de cadastro da foto ou retrato	

hora	VARCHAR		Contém a hora cadastro da foto ou retrato
retrato	BOOL		Campo que diferencia um retrato de uma foto

Tabela 2 – Descrição da Tabela Retrato

TABELA	Molde_Retrato		
DESCRIÇÃO	Tabela de Relacionamento entre o Retrato e o Molde		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
Cod_molde	INTEGER	Primary Key Foreign Key Not Null	Código de referência a tabela molde
retrato_cod_retrato	INTEGER	Primary Key Foreign Key	Código de referência a tabela retrato

Tabela 3 – Descrição da Tabela Molde_Retrato

TABELA	Molde		
DESCRIÇÃO	Tabela que armazena moldes		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_molde	serial	Primary Key Not Null	Código do molde.
cod_tipo_molde	INTEGER		Código do tipo molde
imagem	BLOB		Imagem do molde

Tabela 4 – Descrição da Tabela molde

TABELA	Caracteristica		
DESCRIÇÃO	Tabela que armazena as caracteristicas		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_caracterisitica	INTEGER	<ul style="list-style-type: none"> • Primary Key • Not Null 	Código único identificador da caracteristica
cod_retrato	INTEGER	<ul style="list-style-type: none"> • Not Null 	Código de referência a tabela retrato
cod_valor	INTEGER		Código de referência a tabela valor
cod_tipo_caracteristica	INTEGER		Código de referência a tabela tipoCaracteristica
cod_molde	INTEGER		Código de referência a tabela molde
descricao	INTEGER		Descrição da caracteristica
peso	INTEGER		Peso cadastrado para a mesma

Tabela 5 – Descrição da Tabela caracteristica

TABELA	TipoCaracteristica		
DESCRIÇÃO	Tabela que armazena os tipo de características dos retratos e moldes		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_tipo_caracteristica	INTEGER	Primary Key Foreign Key Not Null	Código único identificador do tipoCaracteristica
descricao	CHAR		Descrição do tipoCaracteristica

Tabela 6 – Descrição da Tabela TipoCaracteristica

TABELA	TipoCaracteristicaTipoMolde		
DESCRIÇÃO	Tabela de relacionamento entre TipoCaracteristica e TipoMolde		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_tipo_caracteristica	INTEGER		Código de referência a tabela tipoCaracteristica
cod_tipo_molde	INTEGER		Código de referência a tabela tipoMolde

Tabela 7 – Descrição da Tabela TipoCaracteristicaTipoMolde

TABELA	TipoMolde		
DESCRIÇÃO	Tabela que armazena os tipos de Moldes que o sistema possui.		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_tipo_molde	INTEGER	Primary Key Foreign Key Not Null	Código único identificador do tipoMolde
descricao	CHAR		Descrição do Tipo do Molde

Tabela 8 – Descrição da Tabela TipoMolde

TABELA	Valor		
DESCRIÇÃO	Tabela que armazena os valores de cada característica pode possuir.		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_valor	INTEGER	Primary Key Foreign Key Not Null	Código único identificador do valor
cod_tipo_caracteristica	INTEGER		Código de referência a tabela tipoCaracteristica
descricao	CHAR		Descrição do valor
pesolni	CHAR		Peso inicial que a característica pode assumir
pesoFim	CHAR		Peso final que a característica pode assumir
decrImagem	CHAR		Descrição da imagem
temimagem	BOOL		Campo que identifica se possui imagem

Tabela 9 – Descrição da Tabela Valor

TABELA	Pergunta		
DESCRIÇÃO	Tabela que armazena as perguntas utilizadas pelo sistema.		
NOME DO CAMPO	TIPO DE DADO	RESTRIÇÃO	DESCRIÇÃO
cod_perg	INTEGER		Código único identificador da

			pergunta
cod_tipo_caracteristica	INTEGER		Código de referência a tabela tipoCaracteristica
descr	VARCHAR		Contém a pergunta feita pelo sistema

Tabela 10 – Descrição da Tabela Pergunta

8. AMBIENTE UTILIZADO PARA O DESENVOLVIMENTO

Denise

- Hardware: Pentium 4 3.0GHz, 512 MB memória RAM, HD 80 GB, conexão 200 kpbs.

Paulo

- Hardware:
Athlon 64 3200, 512MB memoria RAM, HD 80 GB, conexao 300 kbps.
Turion 64 2.0GHz, 512MB, HD 80GB, conexao 300kbps.

Romualdo

- Hardware: Sempron 2300+, 1,25Gb Memória RAM, HD 120 Gb, Conexão 300 kbps

9. REQUISITOS DO SISTEMA SOFTWARE E HARDWARE

Características do Hardware cliente:

Mínimo:

Pentium III 1000 Mhz

512 MB RAM

10 GB HD

Desejável:

Intel core 2 duo 2.4 Ghz

2 GB RAM

320 GB HD

Softwares Utilizados no Desenvolvimento:

Netbeans – Editor free para JAVA.

Eclipse – Editor free para JAVA.

PGAdmin – Software free para administração do PostGreSQL

DBDesigner – Software free para modelagem do DER do Banco de Dados.

Umbrello – Software free usado na modelagem UML

10. CONCLUSÃO

Este trabalho apresentou um software para auxiliar no desenvolvimento de retrato falado. A ferramenta apresentada neste trabalho age como um facilitador, ampliando a capacidade de atendimento da demanda interna de retratos falados. Além da produção da ferramenta e sua documentação de suporte, cabe salientar a pesquisa realizada e registrada neste trabalho, bem como o conhecimento gerado com a utilização das novas tecnologias estudadas.

11. DIFICULDADES ENCONTRADAS

Acreditamos que a maior dificuldade foi desenvolver um projeto inteiro tendo que desenvolver todas as fases do projeto. A utilização do SVG também foi um fator que dificultou a programação no início, pelo fato de se tratar de uma tecnologia desconhecida pelos integrantes da equipe. Outro ponto que a equipe acabou tendo dificuldades foi na linguagem de programação JAVA, pois nem todos os membros da equipe trabalhavam com a linguagem. Para o desenvolvimento do software precisávamos de especialistas na área de retrato falado, devido a escassez desses profissionais encontramos bastante dificuldade de encontrar um especialista que estivesse disposto a trabalhar em conjunto com a equipe.

12. PROJETOS FUTUROS

A partir do momento que a vetorização foi incorporada ao IdentKit um horizonte amplo de melhorias nos foram apresentadas. Uma das principais, seria aplicar a vetorização aos atuais moldes raster, com uma inovação desse nível a aplicação ganharia em maleabilidade na formação do retrato, pois o usuário poderá interagir com os próprios moldes, a fim de fazer pequenos ajustes dentro de cada parte do retrato. Com moldes em formato SVG, também será possível exportar o produto final do retrato sem ter a necessidade de incorporar imagens físicas no mesmo path, assim, com apenas um arquivo final, todo o retrato poderá ser aberto em outras aplicações de edição de vetores, e ajustada.

Desde o começo, toda a equipe se conscientizou que não poderíamos fazer um software a altura dos atuais editores gráficos que estão no mercado, então a partir disso que surgiu a idéia de ter opções para exportar os retratos para aplicação de editoração gráfica mais complexas, como o GIMP, ou até mesmo o ADOBE PhotoShop, para um ajuste fino do produto final.

Todos o processo de montagem de representações faciais humanas também poderá ser melhorado, pois não houve um estudo aprofundado na técnica que é aplicada nas policias do estado, assim a partir do momento que algum especialista utilizar o software, muitas melhorias, que vão desde o posicionamento de janelas, até mesmo novas implementações que não foram imaginadas pelo grupo, poderão ser estudadas.

E finalmente, ao término do projeto, descobrimos que após um ano de estudo aprofundado em várias áreas técnicas, não conseguimos um software final que atendesse um mercado tão importante como é a criação de retratos falados do estado, porém adquirimos um conhecimento de tal nível que se o projeto continuar de algum modo, com estudos aprofundados na área de negócio, o resultado final será algo que não tem precedentes na história nacional.

13. GLOSSÁRIO

Retrato falado – nome comum dado a Representação Facial Humana.

BMP – formato de imagem raster. (BITMAP)

PNG - formato de imagem raster. (Portable Network Graphics)

JPEG – formato de imagem raster (Joint Photographic Experts Group)

BITMAP – Mapa de bits.

Apache - A Apache Software Foundation (ASF) é uma organização sem fins lucrativos criada para suportar os projetos Apache, onde o carro chefe é o servidor Apache, mas também projeto como o XML graphics que subdivide-se em Apache Batik, Apache FOP e Apache XML Graphics Commons.

Toolkit – conjunto de métodos utilizados para auxiliar o desenvolvimento do software.

Framework - Estrutura de suporte definida em que um outro projeto de software pode ser organizado e desenvolvido

LGPL (Lesser General Public License) - Licença de software livre

SGDBs – Sistemas gerenciadores de Banco de Dados.

SQL - Linguagem de pesquisa declarativa para banco de dados relacional

API - Conjunto de rotinas e padrões estabelecidos por um software

Raster - imagens que contém a descrição de cada pixel, em oposição aos gráficos vectoriais.

W3C - World Wide Web Consortium é um consórcio de empresas de tecnologia voltada a cultura WEB.

XML- (eXtensible Markup Language) - Recomendação da W3C para gerar linguagens de marcação para necessidades especiais.

ByteA – Formato de arquivo binários dentro do PostgreSQL

14. REFERÊNCIAS

1-[FOTOCRIM] Software. Disponível em: <
www.citynet.com.br/retratofalado/fotocrim.htm > Acesso em: 16 jan. 2007.

2-[PERITOCRIMINAL] Pesquisa em bases fotográficas, a partir do retrato falado. Disponível em: <www.peritocriminal.net/artigos/retrfalado.htm> Acesso em: 20 jan. 2007.

3-[PHOTOCOMPOSER] Software. Disponível em: <
www.erbitt.com.br/photocomposer.htm>Acesso em: 20 jan. 2007.

4-[ICAO] Software Disponível em: < www.griaule.com/page/pt-br/gricaoface_sdk>Acesso em: 1 fev. 2007.

5-[FACES] Software. Disponível em:
<<http://www.pimall.com/nais/faces.html>>Acesso em: 15 fev. 2007.

6-[PHYSIOGNOMY] Software. Disponível em:
<<http://www.uniphiz.com/physiognomy.htm>>Acesso em: 20 fev. 2007

7-[BATIK] JAVA toolkit for SVG applications. Disponível em: <
<http://xmlgraphics.apache.org/batik/>>Acesso em: 18 abr. 2007.

8-[BATIK] JAVA toolkit for SVG applications. Disponível em: <
[Http://en.wikipedia.org/wiki/Batik_%28software%29](http://en.wikipedia.org/wiki/Batik_%28software%29)>Acesso em: 18 abr. 2007.

9-[PORTAL JAVA] Fórum sobre a tecnologia **JAVA**. Disponível em:
<<http://www.portaljava.com/>. >Acesso em: 16 set. 2007

10-[POSTGRESQL] Site do desenvolvedor do banco de dados **PostgreSQL**.
Disponível em: <<http://www.postgresql.org/>>Acesso em: 28 set. 2005.

11-[HIBERNATE] Framework para o mapeamento objeto-relacional. Disponível em: < <http://www.hibernate.org/>> Acesso em: 15 nov. 2007.

12-[MVC] Padrão de arquitetura de software. Disponível em: <<http://pt.wikipedia.org/wiki/MVC>. > Acesso em: 14 set. 2007.

ANEXO I – Parecer do cliente

Declaro que participamos do projeto para "representação facial humana" com os seguintes representantes:

- Equipe de projeto da UFPR formada por Anderson Pazza Mello, Denise Fernandes, Paulo Henrique Hirth, Romualdo Krameck Junior, Thiago Henrique Podja e Rafael Fernandes de Medeiros.

- CELEPAR - Jair Fernandes e Debora Ruedell

- Funcionários ligados a Secretaria de Estado da Segurança Pública, Instituto de Criminalística e Polícia Civil.

Trata-se da primeira etapa do projeto que visa desenvolver ferramentas para o apoio à Secretaria de Estado da Segurança Pública na identificação de criminosos. Uma das principais formas utilizadas pela polícia na identificar criminosos é através da "representação facial humana" (também popularmente conhecida como "retrato falado"). O processo exige do policial (especialista) um conhecimento profundo em desenho. Além disso, esta tarefa exige uma refinada habilidade para extrair informações das vítimas. Hoje, nos quadros de recursos humanos da segurança pública, há poucas pessoas com este conhecimento.

A identificação de criminosos é um processo fundamental para o trabalho da polícia. A captura e o isolamento social (detenção) destes criminosos evita novos crimes e, principalmente, salva a vida de muitas pessoas possíveis vítimas de assassinos.

Nos últimos anos, a CELEPAR evoluiu muito na informatização da área de segurança pública. Há vários sistemas transacionais que apoiam as atividades nesta área, gerando um grande volume de informações nos bancos de dados. Neste momento, há uma forte necessidade de investir em soluções especialistas que agreguem tecnologia aos processos e, integradas às demais soluções existentes, aumentem a eficiência e a produtividade da atividade policial.

Há poucas soluções conhecidas no mercado voltadas para a "representação facial humana". Os softwares oriundos de outros países possuem fortes características regionais, dificultando a utilização em nossa região. Outro aspecto importante é a necessidade de integração e aderência da solução com

o ambiente de software, hardware e banco de dados existente no Governo do estado.

Após análise deste cenário, a equipe de trabalho da UFPR pesquisou profundamente o conhecimento do especialista na elaboração de um "retrato falado". O objetivo do projeto é propor uma solução capaz de incorporar tecnologia ao processo, construindo, assim, uma ferramenta apoiada no conhecimento do especialista para aumentar a qualidade e multiplicar a capacidade de produzir o produto final.

A solução alcançada pela equipe, apesar das dificuldades existentes, atingiu plenamente aos objetivos. Com o repasse deste conhecimento e o software desenvolvido, a CELEPAR poderá dar continuidade a solução realizando as adaptações necessárias para o funcionamento integrado aos demais sistemas da área de Segurança Pública. Será possível, por exemplo, vincular o retrato gerado com todas as informações do boletim de ocorrência. Para isso, posteriormente, deverá ser assinado um convênio formalizando o repasse deste trabalho.

ANEXO II - Plano do Projeto

I Introdução

1. Propósito deste documento

1. Consolidar o planejamento para o projeto em um único conjunto de documentos que será usado para administrá-lo;
2. Prover meios para atualizar o plano, bem como o andamento do projeto;
3. Regulamentar e oficializar a proposta de criação de um Sistema para desenvolvimento de Representações Faciais Humanas¹;
4. Servir de base para futuras manutenções;
5. Estabelecer as estimativas, o cronograma e organização do pessoal;
6. Estabelecer regras para o gerenciamento de documentos relativos ao desenvolvimento do projeto;

¹ Representação Facial Humana é o termo utilizado atualmente em substituição ao termo “Retrato – Falado”.

7. Delimitar o escopo do projeto: limitações, restrições, funcionalidades e recursos;
8. Estabelecer claramente o escopo do produto, objeto do projeto a ser desenvolvido.

2. Objetivos do Projeto

2.1 Objetivo Geral.

Desenvolver um sistema para montagem de Representações Faciais Humanas. Este é destinado às delegacias de polícia do Estado do Paraná e ao Instituto de Criminalística. Será uma ferramenta de apoio destinada a auxiliar o entrevistador. O sistema informatizará a construção da representação facial, aliado a algumas características físicas que poderão auxiliar na pesquisa por retratos já cadastrados.

2.2 Objetivos Específicos.

2.2.1 Funcionalidades do Sistema.

- Cadastro de representações faciais humanas desenvolvidos pelo usuário. O Cadastro da representação facial dará-se em duas etapas. Primeiro o usuário indicará as características físicas pertinentes à pessoa que será cadastrada. Depois, na segunda etapa ele montará a imagem da pessoa que se estiver tentando retratar.
- Disponibilizar moldes faciais. A apresentação dos moldes faciais terá como objetivo facilitar algumas lembranças para o entrevistado.

2.3 Questões de Desempenho.

2.3.1 Desenvolvimento do projeto

Para acompanhar o desenvolvimento do projeto, será utilizada a análise do cronograma inicial proposto para o desenvolvimento do projeto. Com isso, poder-se-ão mensurar e quantificar os esforços, analisando se os mesmos foram utilizados a contento e atendendo o que foi projetado inicialmente.

Após a implementação, a homologação do sistema por parte do cliente será um fator decisivo para que se possa analisar o resultado da qualidade do produto final a ser entregue e se este atende às especificações estabelecidas no presente plano de projeto.

2.3.4 Restrições Técnicas e Administrativas

- Conceitos novos para a equipe;

- Domínio da tecnologia a ser utilizada;
- Falta de disponibilidade do cliente;
- Integração do sistema com outros já existentes;
- Conceitos de processamento de imagens;

II Estimativas de projeto

1. Dados históricos usados nas estimativas

Para estimar os prazos durante a fase de análise, foi levada em consideração a experiência de cada componente da equipe na questão de desenvolvimento de software, projetos corporativos, o conhecimento do negócio, o estudo de ferramentas existentes para construção de retrato-falado, além do grau de dificuldade das atividades e aplicações similares na mesma plataforma.

2. Técnicas de Estimativas

- Definição das ferramentas a serem utilizadas;
- Divisão do projeto em fases;
- Divisão das fases em tarefas;
- Entrevistas com o cliente;
- Levantamento de dados históricos de projetos semelhantes;
- Levantamento de requisitos;

3. Estimativas

Produto para automatizar o processo de construção de um retrato falado, para assim agilizar e facilitar esta atividade. Além de prover alguns recursos que complementam as funcionalidades já existentes.

I) Fases do projeto:

- Levantamento de requisitos;
- Análise;
- Implementação;
- Documentação;
- Apresentação.

II) Tarefas das fases:

Levantamento de requisitos:

- Realização de entrevista com o cliente para definição do escopo do projeto e funcionalidades do sistema;
- Elaboração do plano de projeto.

Análise:

- Elaboração dos diagramas UML do sistema: casos de uso, diagrama de classes, DER, diagramas de seqüência, diagramas de atividades;
- Elaboração de descrição resumida do sistema, tendo por base as definições dos diagramas UML.

Implementação:

- Escolha e instalação das ferramentas que serão utilizadas no desenvolvimento do sistema;
- Criação das telas;
- Criação do banco de dados;
- Codificação do sistema;
- Execução de testes.

Documentação:

- Elaboração de descrição detalhada do projeto, com base no resultado do sistema após a codificação e documentação do sistema;
- Revisão e atualização dos diagramas UML, com base nas alterações que se fizerem necessárias durante o desenvolvimento do sistema;
- Elaboração dos manuais de auxílio para utilização do sistema;
- Impressão de todo o material.

Apresentação:

- Elaboração da apresentação do projeto, que será mostrada à banca de avaliação;
- Entrega do material aos componentes da banca de avaliação;
- Apresentação do sistema à banca de avaliação.

III) Previsão de duração das fases

- Levantamento de requisitos: 112 dias;
- Análise: 43 dias;
- Implementação: 190 dias;
- Documentação: 127 dias;
- Apresentação: 1 dia.

IV) Previsão de duração do projeto

- Início: janeiro/2007;
- Término: dezembro/2007;
- Apresentação: dezembro/2007;
- Duração total: 272 dias.

III Riscos do projeto

1. Propósito do Plano de Gerenciamento dos Riscos

O Plano de gerenciamento dos riscos tem os seguintes propósitos:

- Classificar os riscos, considerando-os como baixo, médio ou alto, no que se refere a probabilidade de acontecer;
- Categorizar os riscos conforme seu grau de severidade;
- Identificar os sintomas que indicam que o risco está por se materializar;
- Desenvolver procedimentos de evitar os riscos;

- Definir procedimentos dentro do projeto de identificação de novos riscos e a reavaliação dos já identificados.

2. Procedimentos para Identificação de novos Riscos.

- Toda reunião contará com uma reavaliação dos riscos já discutidos;
- Esta etapa consiste em identificar possíveis sintomas possíveis problemas e atualizar/reavaliar os riscos já encontrados.

3. Análise dos riscos

3.1 Ambiente Interno ao Projeto

Falta de Infra-Estrutura (Severidade: Alta; Risco: Baixo)

Uma infra-estrutura (equipamentos e softwares utilizados para o desenvolvimento do projeto) inadequada para o desenvolvimento do projeto, sem dúvida comprometeria a realização de etapas do projeto ou até mesmo do projeto como um todo.

Falta de disponibilidade dos integrantes da equipe (Severidade: Alta; Risco: Médio)

Os integrantes da equipe que realizam este projeto possuem atividades paralelas ao desenvolvimento do projeto, ou seja, se não dedicarem de uma maneira organizada e produtiva o tempo que possuem, com uma divisão de tarefas muito bem estruturada o sucesso deste trabalho poderá ser comprometido.

Utilização de um Base de Imagens Faciais (Severidade: Alta; Risco: Alto)

Sem uma base com quantidade relevante de imagens que representam a face humana o projeto torna-se inviável.

Compreensão das Técnicas utilizadas para o reconhecimento facial humano (Severidade: Alta; Risco: Baixo)

Conhecer os procedimentos utilizados atualmente para o desenvolvimento de representações faciais humanas, é importante para produzir um sistema intuitivo que facilitaria o manuseio por parte dos usuários.

Problemas com a pesquisa das tecnologias para que serão utilizadas no desenvolvimento do projeto (Severidade: Alta; Risco: Baixo)

Levando em conta tudo o que existe para desenvolvimento de sistemas atualmente, escolher tecnologias que tem projeção e são simples de se utilizar, é uma tarefa que não pode falhar. Escolhas erradas ou precipitadas nesta etapa do projeto podem levar ao insucesso do mesmo.

Dificuldade na implementação de funcionalidades do sistema (Severidade: Alta; Risco: Baixo)

Conhecer as funcionalidades do sistema por completo, é importante para não ocorrer surpresas durante a implementação das mesmas.

3.2 Ambiente externo do projeto

Aceitação da tecnologia escolhida pelo cliente (Severidade: Média; Risco: Médio)

O cliente deve possuir recursos para manutenção no sistema que será desenvolvido, ou seja, as ferramentas que serão escolhidas, podem sofrer resistência por parte do cliente.

Expectativa excedente do que será desenvolvido (Severidade: Média; Risco: Médio)

Depois que o sistema estiver modelado, alterações em regras de negócio podem comprometer o tempo de desenvolvimento que foi proposto.

4. Administração dos riscos

4.1 Atividades para mitigar os riscos

- Auto-Estudo constante;
- Monitoramento do cronograma;
- Reuniões com o professor orientador semanalmente;
- Reuniões mensais com o cliente e orientador em conjunto;
- Comunicação freqüente com o professor orientador via e-mail para esclarecimento de dúvidas;
- Redundância de repositório de dados, de modo que o trabalho desenvolvido não fique centralizado em um único computador;
- Planejamento e cumprimento de cronograma fixo, acordado no início do projeto;
- Execução de testes durante a codificação e após o término de cada construção dos sistemas;
- Ater-se ao escopo do projeto especificado no início.

4.2 Procedimentos de monitoração dos riscos

- Reuniões periódicas da equipe;
- Análise do Plano Geral de Projeto;
- Análise freqüente do calendário;
- Reuniões com o cliente e feedback do professor orientador.

IV Cronograma

1. Work Breakdown Structure (WBS)

Ver anexo I

2. Rede de Tarefas (Pert CPM)

Ver anexo II

3. Gráfico de Gantt

Ver anexo III

V Recursos do projeto

1. Pessoal

O projeto será desenvolvido pela equipe, contando com o apoio da Celepar para a disponibilização dos padrões adotados no desenvolvimento, base de dados e funcionalidades do sistema já existente. Os integrantes da equipe são:

- **Denise Fernandes:** Conhecimento em Java, JSP, Cobol , CICS, C, PostGres e DB2. Trabalha atualmente com desenvolvimento no HSBC Bank Brasil.
- **Paulo Henrique Hirth:** Conhecimentos em Java, PHP, Mysql, MsSQL, Oracle, JavaScript, Ajax, Windows/Linux. Atualmente desempregado.
- **Romualdo Krameck Junior:** Conhecimento em Delphi, Java, PHP e MySQL. Trabalha com desenvolvimento de Sistemas de Informação em Delphi.

2. Hardware e Software

A aplicação será executada em um computador (servidor) instalado nas dependências do cliente. Este servidor servirá de apoio para os destes durante a fase de implementação. O desenvolvimento nas estações de trabalho se dará com a utilização dos microcomputadores da equipe, valendo-se dos seguintes softwares e hardwares, na linguagem JSP/Java versão 1.5:

2.1 Softwares

- DBDesigner;
- IDE Eclipse;
- Java;
- JavaDoc;
- Jude;
- Internet Explorer;
- Mozilla Firefox;
- PostgreSQL;

2.2 Hardwares

Denise

- Softwares: Utilitários comuns*
- Hardware: Pentium 4 3.0GHz, 512 MB memória RAM, HD 80 GB, conexão 200 kpbs.

Paulo

- Softwares: Utilitários comuns*
- Hardware:

Athlon 64 3200, 512MB memoria RAM, HD 80 GB, conexao 300 kbps.

Turion 64 2.0GHz, 512MB, HD 80GB, conexao 300kbps.

Romualdo

- Softwares: Utilitários comuns*.
- Hardware: Sempron 2300+, 1,25Gb Memória RAM, HD 120 Gb, Conexão 300 kbps

* Os softwares classificados como **utilitários comuns** são aqueles que todos os desenvolvedores terão instalados em seus computadores, a saber: DBTools 4, Ferramentas auxiliares (envio de e-mail, comunicador instantâneo, ferramentas de escritório Office e e-books para pesquisa).

VI Organização do pessoal

1. Estrutura da equipe

A equipe terá uma divisão de papéis de modo que **todos estarão envolvidos em todas as atividades**, porém, haverá um “líder” para cada atividade específica, uma espécie de coordenador, que deverá verificar o andamento do projeto dentro da atividade na qual este é responsável.

- Gerente do projeto:
 - Atividades: Controlar prazos, tarefas e cronogramas; agendar reuniões; contato com o cliente; contato com o professor orientador; gerenciar os meios para o desenvolvimento (hardware, software, IDE's utilizadas, versões); manter o foco da equipe; exibir resultados; pesquisas das tecnologias.
 - Membro responsável: **Denise**
- Programação:
 - Atividades: Coordenar as atividades técnicas referentes à programação; estabelecer/adotar padrões de codificação; frameworks utilizados; *refactoring*; homologação; implementação de funcionalidades; controle de versões; testes.
 - Membro responsável: **Romualdo**
- Modelagem:
 - Atividades: Coordenar a modelagem do sistema no tocante a: diagramas de caso de uso, classes e seqüência; modelo Entidade-Relacionamento; interface do sistema (telas); estrutura do banco de dados.
 - Membro responsável: **Paulo**

- Documentação:
 - Atividades: Desenvolvimento do HELP do sistema; atualização do plano de projeto; atas; relatório de atividades mensais; pesquisa das tecnologias; documentação do código fonte.
 - Membro responsável: **Romualdo / Denise**

2. Relatórios administrativos

Serão produzidos pela equipe um relatórios mensais (ver anexo IV), contendo as atividades desenvolvidas pelo grupo e se estas estão dentro do cronograma proposto inicialmente. Os relatórios serão elaborados de maneira sucinta, as atividades e datas nas quais estas foram realizadas, erros observados, sugestões/dificuldades e expectativas para as próximas iterações.

VII Mecanismos de tracking (rastreamento) e controle

1. Propósitos dos indicadores de desempenho

Os propósitos dos indicadores de desempenho do projeto são:

- Servir de base para o acompanhamento geral do andamento do projeto;
- Identificar as causas potenciais de desvios;
- Antecipar prováveis tendências;
- Auxiliar tomadas de decisões.

2. Indicadores do projeto

Baseadas nos relatórios gerados haverá reuniões mensais administrativas, separadas em 3 (três) categorias, a saber: **reuniões internas da equipe**, para discussão de assuntos técnicos, divisão de trabalhos, implementação e solução de dúvidas; **reuniões com o cliente**, para avaliação do sistema quanto ao progresso deste e apresentação de resultados, bem como definição de novas metas para a próxima reunião, definição de prioridades e esclarecimentos de dúvidas quanto ao negócio; e, **reuniões com o professor orientador**, para o esclarecimento de dúvidas técnicas e de implementação, sugestões para as próximas iterações e acompanhamento da evolução do sistema. Essas reuniões serão previamente agendadas e realizadas nas datas marcadas, salvo em exceções extraordinárias.

Além das reuniões, serão feitas análises freqüentes do Plano Geral de Projeto, para avaliar se o desenvolvimento não está fugindo do escopo inicial acertado. O controle do cronograma será feito utilizando o software Microsoft Project 2003.

Anexos.

Anexo I – Work Breakdown Structure (WBS)

		Task Name	Início	Término	Predecessoras	Duração	Nomes de recursos
1		Planejamento e Coleta de Requisitos	Qui 18/1/07	Qui 21/6/07		112 dias?	
2		Site do Projeto	Qua 3/1/07	Sex 12/1/07		8 dias	Paulo
3		Plano do Projeto	Qua 3/1/07	Sex 30/3/07		63 dias	Denise
4		Entrega Plano de projeto	Sáb 31/3/07	Sáb 31/3/07	3	1 dia	Anderson
5		Criação do DER	Seg 2/4/07	Qua 30/5/07	4	43 dias?	Paulo
6		Criação do Diagrama de classes	Seg 2/4/07	Qua 30/5/07	4	43 dias?	Paulo
7		Criação do Diagrama de Casos de Uso	Seg 2/4/07	Qua 30/5/07	4	43 dias?	Paulo
8		Criação do Fluxos de Eventos	Seg 2/4/07	Qua 30/5/07	4	43 dias?	Paulo
9		Criação do Diagrama de Sequencia	Seg 2/4/07	Qua 30/5/07	4	43 dias?	Paulo
10		Estudo de ferramentas de manipulação de imagens	Ter 13/3/07	Seg 13/8/07		111 dias?	Todos
11		Apresentação do Protótipo	Seg 27/8/07	Qua 5/9/07		8 dias?	Todos
12		Construção da interface	Seg 2/4/07	Seg 4/6/07	4	46 dias?	Romualdo
13		Desenvolvimento do Sistema	Seg 2/4/07	Sex 21/12/07	4	190 dias?	Todos
14		Documentação do Sistema	Seg 2/4/07	Ter 25/9/07		127 dias?	Romualdo;Denise
15		Especificação do Sistema	Seg 2/4/07	Seg 25/6/07	4	61 dias?	Todos
16		Apresentação do Projeto	Seg 24/12/07	Seg 24/12/07	13	1 dia?	
17		+ Reunião da equipe	Qui 18/1/07	Qui 13/12/07		237 dias?	Todos
58		+ Reunião com o Orientador	Qua 21/3/07	Qui 13/12/07		193 dias	Todos

Figura 1 – Work Breakdown Structure (WBS)

Anexo II – Rede de Tarefas (Pert CPM)

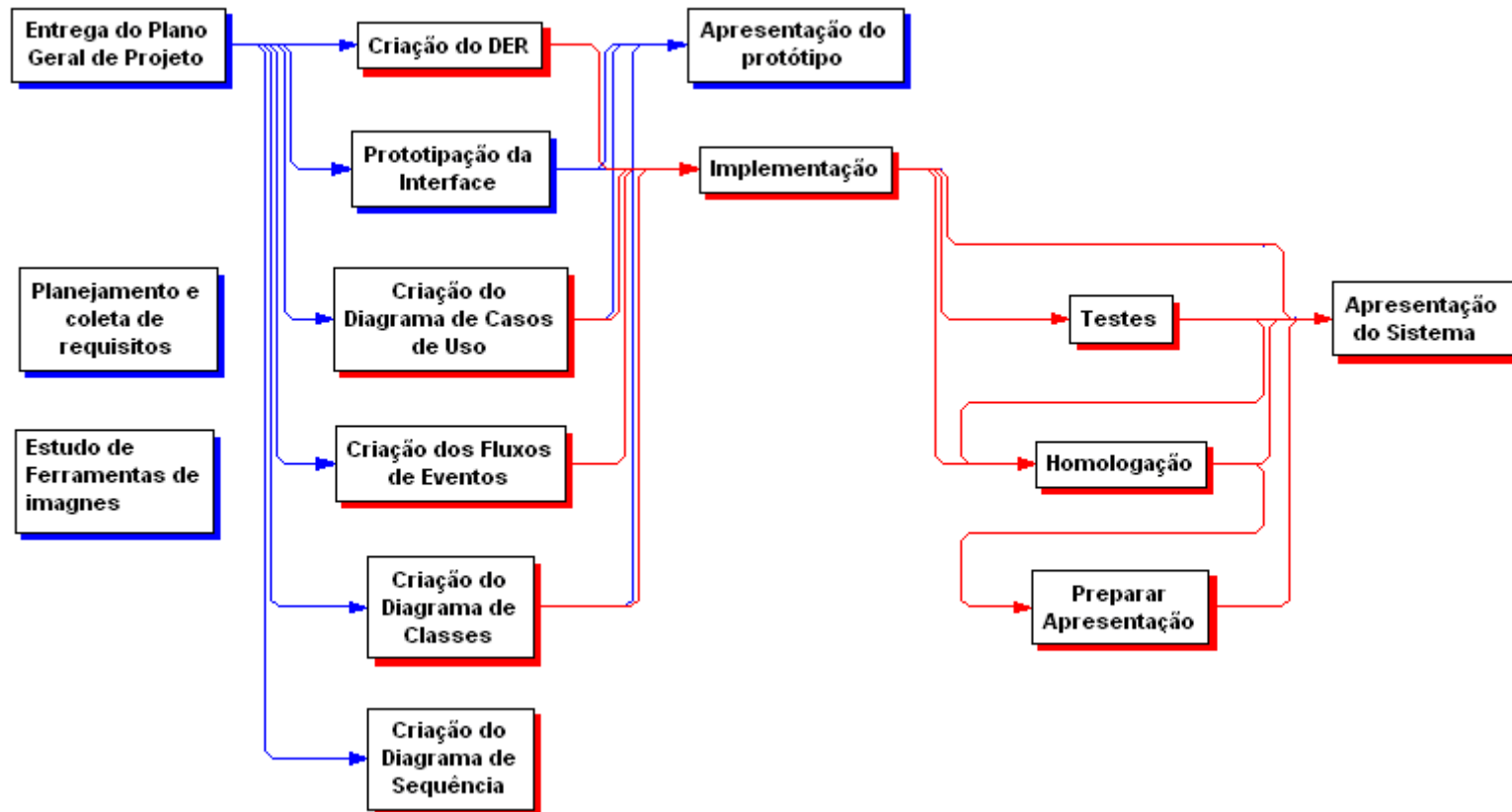


Figura 2 - Rede de tarefas PERT / CPM

Anexo III - Gráfico de Gantt com as atividades macro.

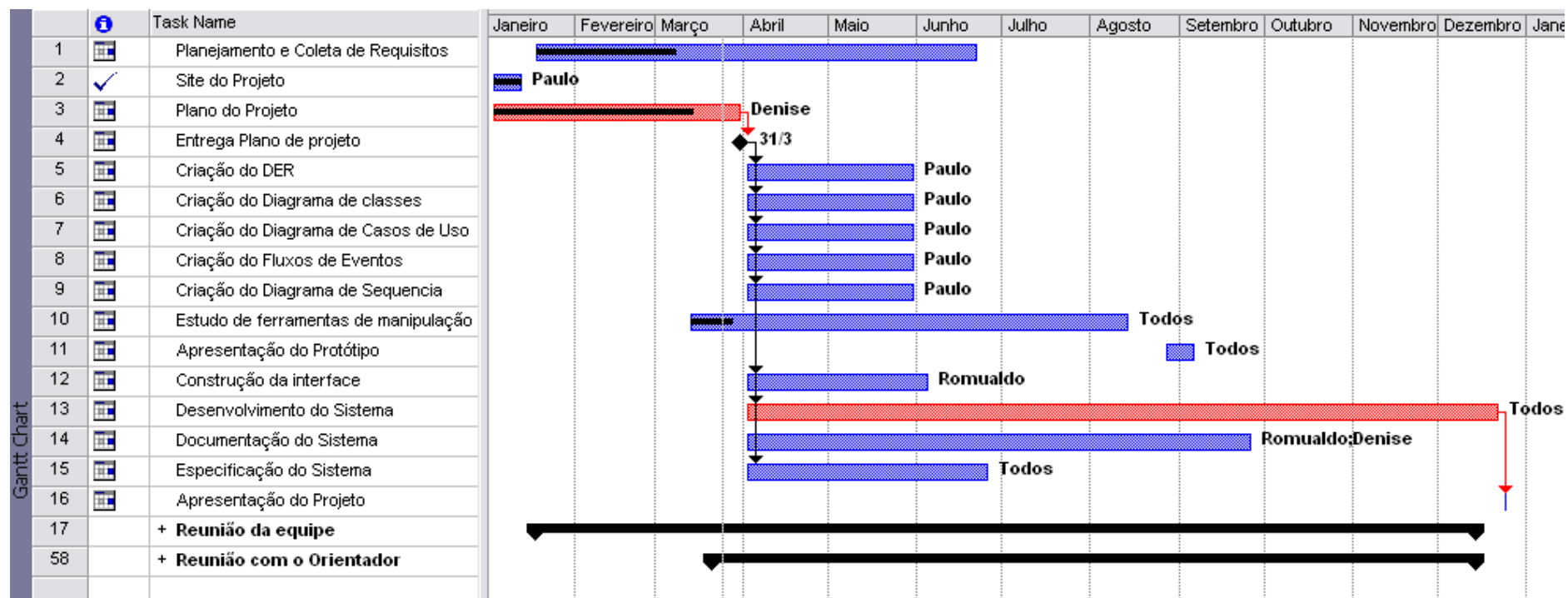


Figura 3 – Planejamento: Gráfico de Gantt.

REFERÊNCIAS BIBLIOGRÁFICAS

1-PRESSMAN, Roger S. **Engenharia de software**. São Paulo: Makron Books, 1995.

2-Projeto Notarium. 2005. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Informática), Setor Escola Técnica, Universidade Federal do Paraná, Curitiba, 2006.

3-Projeto DEMS Workflow. 2005. Trabalho de Conclusão de Curso (Graduação em Tecnologia em Informática), Setor Escola Técnica, Universidade Federal do Paraná, Curitiba, 2006.

4-QUADROS, Moacir. **Gerência de projetos de software: técnicas e ferramentas**. Florianópolis: Bookstore Livraria Ltda (Visual Books), 2002.