

SGC - Sistema de Gerenciamento de Colaboradores

Trabalho de Conclusão de Curso
apresentado à UTFPR como requisito
parcial para obtenção do título de Tecnólogo
em Sistemas para internet.

**Curitiba
2012**

Marlon Jair da Silva

SGC – Sistema de Gerenciamento de Colaboradores

Trabalho de Conclusão de Curso
apresentado à UTFPR como requisito
parcial para obtenção do título de Tecnólogo
em Sistemas para internet.

Orientador:

Prof. Msc. Luiz Augusto Pelisson

**Curitiba
2012**

AGRADECIMENTOS

A Deus e a todos aqueles que ajudaram no desenvolvimento deste trabalho.

SUMÁRIO

| | |
|--|-----------|
| 1 INTRODUÇÃO..... | 13 |
| 1.1 Apresentação | 13 |
| 1.2 Justificativa da Escolha do Tema..... | 14 |
| 1.3 Objetivos do Trabalho | 14 |
| 1.3.1 Objetivos específicos | 15 |
| 1.4 Objetivos específicos | 15 |
| 2 LEVANTAMENTO BIBLIOGRÁFICO E ESTADO DA ARTE | 16 |
| 2.1 Base teórica..... | 16 |
| 2.2 Base teórica workflow | 17 |
| 2.3 Base de desenvolvimento | 18 |
| 3 METODOLOGIA..... | 19 |
| 3.1 Conteúdos da Metodologia | 19 |
| 3.2 Levantamento de requisitos | 19 |
| 3.3 Cronograma..... | 20 |
| 3.4 Projeto e implementação | 20 |
| 3.4.1 Estudo da tecnologias..... | 21 |
| 3.4.2 Modelagem do banco de dados..... | 22 |
| 3.4.3 Protótipo | 23 |
| 3.4.4 Desenvolvimento e integração dos serviços | 23 |
| 3.4.5 Teste geral..... | 24 |
| 3.4.6 Customização da aplicação | 24 |
| 3.5 Recursos empregados | 25 |
| 3.5.1 Recursos financeiros | 25 |
| 3.5.2 Recursos pessoais | 25 |
| 3.5.3 Recursos de hardware | 25 |
| 3.5.4 Recursos de software..... | 25 |
| 3.5.4.1 Softwares utilizados | 26 |
| 3.5.4.2 Tecnologias utilizadas..... | 28 |

| | |
|---|-----------|
| 4 RESULTADOS | 31 |
| 4.1 Descrição da arquitetura | 31 |
| 4.2 Requisitos funcionais e não funcionais | 32 |
| 4.2.1 Requisitos funcionais | 32 |
| 4.2.2 Requisitos não funcionais | 32 |
| 4.3 Casos de uso | 33 |
| 4.3.1 Caso de uso geral | 33 |
| 4.3.2 Autenticar | 34 |
| 4.3.3 Buscar funcionário | 35 |
| 4.3.4 Busca funcionário detalhado | 36 |
| 4.3.5 Listar equipe | 36 |
| 4.3.6 Transacionar colaborador | 37 |
| 4.3.7 Configurações de conta | 38 |
| 4.3.8 Transações pendentes | 38 |
| 4.3.9 Validar transações pendentes | 39 |
| 4.3.10 Área administrativa | 40 |
| 4.3.11 Gerenciar colaborador | 41 |
| 4.3.12 Gerenciar grupos | 42 |
| 4.3.13 Gerenciar cargos | 43 |
| 4.3.14 Gerenciar situações | 44 |
| 4.3.15 Controle de acesso | 45 |
| 4.4 Diagramas de seqüência | 46 |
| 4.4.1 Autenticação | 46 |
| 4.4.2 Buscar Colaborador | 47 |
| 4.4.3 Listar Equipe | 47 |
| 4.4.4 Novo Colaborador | 48 |
| 4.4.5 Alterar Colaborador | 48 |
| 4.4.6 Remover Colaborador | 49 |
| 4.4.7 Transacionar Colaborador | 49 |
| 4.4.8 Transação Colaborador Aceitar | 50 |
| 4.4.9 Transação Colaborador Recusar | 51 |
| 4.5 Diagramas de classes | 51 |

| | |
|---|-----------|
| 4.5.1 Descrição das Classes | 54 |
| 4.6 Diagrama entidade relacionamento | 55 |
| 4.7 Interfaces | 56 |
| 4.8 Workflow | 57 |
| 5 DISCUSSÃO..... | 60 |
| 5.1 Dificuldades encontradas | 60 |
| 6. CONCLUSÕES..... | 60 |
| 6.1 Contribuições..... | 61 |
| 6.2 Trabalhos Futuros | 61 |
| 7 REFERÊNCIAS | 65 |

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 - Controle de colaboradores modelo Excel (GVT 2010/2011). | 18 |
| Figura 2- Caso de uso 'geral' | 33 |
| Figura 3 – Caso de uso 'autenticar' | 34 |
| Figura 4 – Caso de uso 'buscar' | 35 |
| Figura 5 - Caso de uso colaborador detalhado' | 36 |
| Figura 6 - Caso de uso 'listar equipe' | 36 |
| Figura 7 – Caso de uso 'transacionar colaborador' | 37 |
| Figura 8 - Caso de uso 'configurações de conta' | 38 |
| Figura 9 - Caso de uso 'transações pendentes' | 38 |
| Figura 10 - Caso de uso 'validar transações pendentes' | 39 |
| Figura 11 - Caso de uso 'área administrativa' | 40 |
| Figura 12 - Caso de uso 'gerenciar colaborador' | 41 |
| Figura 13 - Caso de uso 'gerenciar grupos' | 42 |
| Figura 14 - Caso de uso 'gerenciar cargos' | 43 |
| Figura 15 - Caso de uso 'gerenciar situações' | 44 |
| Figura 16 - Caso de uso 'controle acesso' | 45 |
| Figura 17 - Diagrama de seqüência 'login' | 46 |
| Figura 18 - Diagrama de seqüência 'buscar colaborador' | 47 |
| Figura 19 - Diagrama de seqüência 'listar colaborador' | 47 |
| Figura 20 - Diagrama de seqüência 'novo colaborador' | 48 |
| Figura 21 - Diagrama de seqüência 'alterar colaborador' | 48 |
| Figura 22 - Diagrama de seqüência 'remover colaborador' | 49 |
| Figura 23 - Diagrama de seqüência 'transacionar colaborador' | 49 |
| Figura 24 - Diagrama de seqüência 'validar transação aceitar' | 50 |
| Figura 25 - Diagrama de seqüência 'valida transação recusar' | 51 |
| Figura 26 – Pacotes | 51 |
| Figura 27- Pacote 'controller' e suas respectivas classes | 52 |
| Figura 28 - Pacote 'model' e suas respectivas classes | 52 |
| Figura 29- Pacote 'dao' e suas respectivas classes | 53 |
| Figura 30 - Pacote 'utils' e suas respectivas classes | 54 |

| | |
|--|----|
| Figura 31 - (DER) Diagrama Entidade Relacionamento | 55 |
| Figura 32 - SGC área pública | 56 |
| Figura 33- SGC área administrativa..... | 56 |
| Figura 34 - SGC - lista de colaboradores..... | 56 |
| Figura 35 - Hierarquia | 57 |
| Figura 36 - Fluxo Requisitor..... | 58 |
| Figura 37 - Fluxo Receptor | 59 |
| Figura 38 -Fluxo Administrador..... | 59 |
| Figura 39- Menu FeedBack..... | 62 |
| Figura 40 - Menu Desligamento..... | 63 |
| Figura 41 - Menu Relatórios | 63 |
| Figura 42 - Menu Relatórios detalhados | 64 |

LISTA DE TABELAS

| | |
|--|----|
| Tabela 1 - Cronograma..... | 20 |
| Tabela 2 - Configuração do computador..... | 25 |

LISTA DE ABREVIATURAS E SIGLAS

AJAX: *Asynchronous JavaScript and XML*
CSS: *Cascading Style Sheets*
DAO: *Data Access Object*
DER: *Diagrama Entidade Relacionamento*
HTML: *HiperText Markup Language.*
IDE: *Integrated Development Environment*
J2EE: *Java 2 Platform, Enterprise Edition.*
JDK: *Java Development Kit*
JPA: *Java Persistence API*
JSF: *Java Server Faces*
JSP: *Java Server Pages*
JSTL: *JavaServer Pages Standard Tag Library*
JVM: *Java Virtual Machine*
CRM: *Customer Relationship Management*
GVT: *Global Village Telecom*
MVC: *Model-View-Controller*
SGBD: *Sistema Gerenciador de Banco de Dados*
SGC: *Sistema de gerenciamento de colaboradores.*

RESUMO

Com o surgimento de novas tecnologias o mundo digital vem crescendo em um ritmo acelerado fazendo com que os usuários tenham que se atualizar dia após dia, buscando melhores alternativas de softwares para otimizar o seu trabalho diário. Para que este ciclo ocorra, profissionais de tecnologia buscam sempre novas alternativas para inovar, facilitar, agilizar e flexibilizar o uso e o desenvolvimento destes sistemas, uma exemplo disso seria criar fluxos de trabalho para automatizar processos, os chamados workflow, que é definido como uma distribuição de tarefas entre dois ou mais participantes passando de um para o outro até a execução de uma ação.

Este trabalho implementa um sistema de gerenciamento de colaboradores, visando suprir todas as necessidades de controle de funcionários de uma organização, além de substituir controles antigos em papéis ou planilhas. Para desenvolvimento desta aplicação foram utilizados conceitos de engenharia de software e como base para programação foi utilizado linguagem de programação Java. Como resultado deste estudo foi desenvolvido um sistema de gerenciamento de colaboradores com interface Web.

Palavras chaves: Fluxo de trabalho, Automatizar processos, Sistema Web, Java.

ABSTRACT

With human evolution in the digital world, the development of new technologies has been increasing. These way users tend to update on a daily looking for better alternatives in software. In order to this cycle occur technology professionals constantly seer for new want to innovate, facilitate, streamline and flexibilize the use of this systems. hence creating workflows to automatize processes wich are defines as a distribution between two or more users going from one another until of execution of an action.

This work was developed with the aim of implementing a employee managing system, aiming to supply all employee control management needs of a company, besides it also substitutes old control measures on paper and spreadsheets. In order to develop this application, software engineering concepts were used and as programming base if was used java programming language. As a final result it was obtained a user management system with front end web.

Keywords: *workflow, automate processes, Workflow System, Web, Java.*

1 INTRODUÇÃO

O controle de colaboradores de uma empresa é uma necessidade real nos dias de hoje, tanto para controle de quadro, quanto medir o crescimento desta organização. O armazenamento de informações referentes aos colaboradores que já passaram pela empresa e aqueles que fazem parte do quadro de funcionários atual é uma rotina extremamente importante para as organizações.

O presente estudo foi desenvolvido em cima de uma situação real de uma concessionária do setor de telecomunicações. Nesta empresa, inicialmente o sistema de gerenciamento de colaboradores era baseado em uma planilha *Excel* que apesar de útil por um longo período, não era um meio mais seguro e ideal para armazenar informações, que também oferecia dificuldades para pesquisa quando havia um volume imenso de informações.

A proposta deste trabalho é voltada para controle de funcionários de uma organização sendo uma solução de software criada para sanar todos os problemas encontrados em sistemas antigos, controlados em papéis ou em planilhas, também servirá para automatizar toda entrada e saída de funcionários, assim como todas as transações internas ocorridas dentro de um ambiente corporativo.

1.1 Apresentação

Este projeto tem como foco central analisar, projetar e implementar um sistema para gestão de colaboradores seguindo como base um modelo *workflow* para um ambiente empresarial, com a finalidade de automatizar o processo de registro, transações e busca de informações de funcionários.

Este documento aborda vantagens e desvantagens do *SGC* – Sistema de Gerenciamento de Colaboradores, as métricas utilizadas para o planejamento, à metodologia utilizada no desenvolvimento do projeto e as tecnologias que foram utilizadas assim como seus resultados e conclusões.

1.2 Justificativa da Escolha do Tema

Este projeto visa minimizar problemas encontrados em ambientes empresariais no controle de funcionários, tendo em vista inovar, facilitar, agilizar, flexibilizar e o mais importante manter a confiabilidade das informações. Em contrapartida ao uso de um modelo planilha, este sistema inovador provê maior agilidade nos fluxos de processos de inserção e recuperação dos dados, assim como maior facilidade para acompanhamento das informações armazenadas.

O principal incentivo para o desenvolvimento deste trabalho foi à necessidade de implementação de um sistema para controle de colaboradores na área de *CRM (Customer Relationship Management)* da empresa *GVT(Global Village Telecom)*.

Utilizando como base teórica o livro de Tadeu Cruz, "*Workflow II A Tecnologia que Revolucionou os Processos*" e também o modelo de hierarquia da empresa, sendo assim foi construído um sistema padrão de gerenciamento de funcionários.

1.3 Objetivos do Trabalho

O objetivo principal deste projeto é a automatização de serviços, onde o sistema envolve gestores para executar uma ação: "a movimentação de um colaborador", ou seja para a transação de um colaborador há um fluxo de trabalho a ser seguido até que esta ação seja concluída. A intenção para o uso deste modelo de fluxo é suprir toda necessidade de atualização de dados de colaboradores em um ambiente corporativo assim como rapidez e confiabilidade da informação, facilitando assim os processos de inserção e recuperação dos dados.

1.3.1 Objetivos específicos

Este sistema que reconhece e faz a distinção de acordo com o cargo do usuário, por exemplo, um gerente, terá permissões diferentes de um analista. O sistema faz um reconhecimento através de matrícula e senha e mostra campos mais importantes para este usuário. Também faz um próximo reconhecimento caso o colaborador tente acessar a área administrativa, validando assim a matrícula do mesmo verificando se possui permissão ou não para acessar esta área.

1.4 Objetivos específicos

Este projeto é dividido nos seguintes capítulos, introdução, levantamento bibliográfico, metodologia, resultados alcançados, discussão e conclusão, abaixo seguem cada divisão detalhada:

a) Introdução: Apresenta o tema escolhido, as justificativas para sua escolha, assim como os objetivos específicos sobre este assunto e a organização do documento.

b) Levantamento bibliográfico e estado da arte: Apresenta a revisão da bibliografia, demonstrando conceitos teóricos que servem como base para o tema apresentado neste projeto.

c) Metodologia: Apresenta os métodos utilizados na construção deste projeto, assim como levantamento de requisitos, recursos pessoais, equipamentos e softwares utilizados.

d) Resultados: Apresenta os resultados obtidos durante o desenvolvimento do projeto, como os diagramas contidos no planejamento e o sistema final.

e) Discussão: Neste cenário discute e analisa os resultados obtidos.

f) Conclusão: É o encerramento do trabalho, citando o que foi construído no projeto.

2 LEVANTAMENTO BIBLIOGRÁFICO E ESTADO DA ARTE

Neste capítulo serão apresentados os conceitos teóricos deste projeto, enfatizando o uso do sistema de gerenciamento de colaboradores, assim como suas vantagens e desvantagens de cada cenário da aplicação.

2.1 Base teórica

Conforme citado no capítulo 1 desta obra o livro de Tadeu Cruz foi utilizado como base teórica para construção deste projeto.

Em seu livro, ele apontou que para realizar um sistema de *Workflow* assertivo é necessário que a área de Tecnologia e Negócio estejam juntas em todo o processo, também é necessário saber todas as atividades, processos, tarefas e outros elementos que compõem um processo. A partir disto os processos começaram a ficar mais rápidos, eficientes e seguros e a organização tende a ganhar.

Para combinar com o modelo proposto por Tadeu foi utilizado como base de desenvolvimento o modelo de hierarquia de funcionários da empresa *GVT*.

A partir deste modelo foi criado um banco de dados para armazenamento das informações assim como seus respectivos diagramas e por fim para manipulação destes dados foi construído um sistema de controle de funcionários com interface *Web*.

Existem muitas ferramentas de *workflow* disponíveis no mercado hoje, porém nenhuma com as características necessárias para transacionar funcionários de um ambiente corporativo, esta ferramenta é inovadora e única.

2.2 Base teórica workflow

Workflow ou fluxo de trabalho, em português, seu conceito é de uma seqüência de passos necessários para automatizar processos, de acordo com um conjunto de regras definidas, permitindo que estes possam ser transmitidos de uma pessoa para outra.

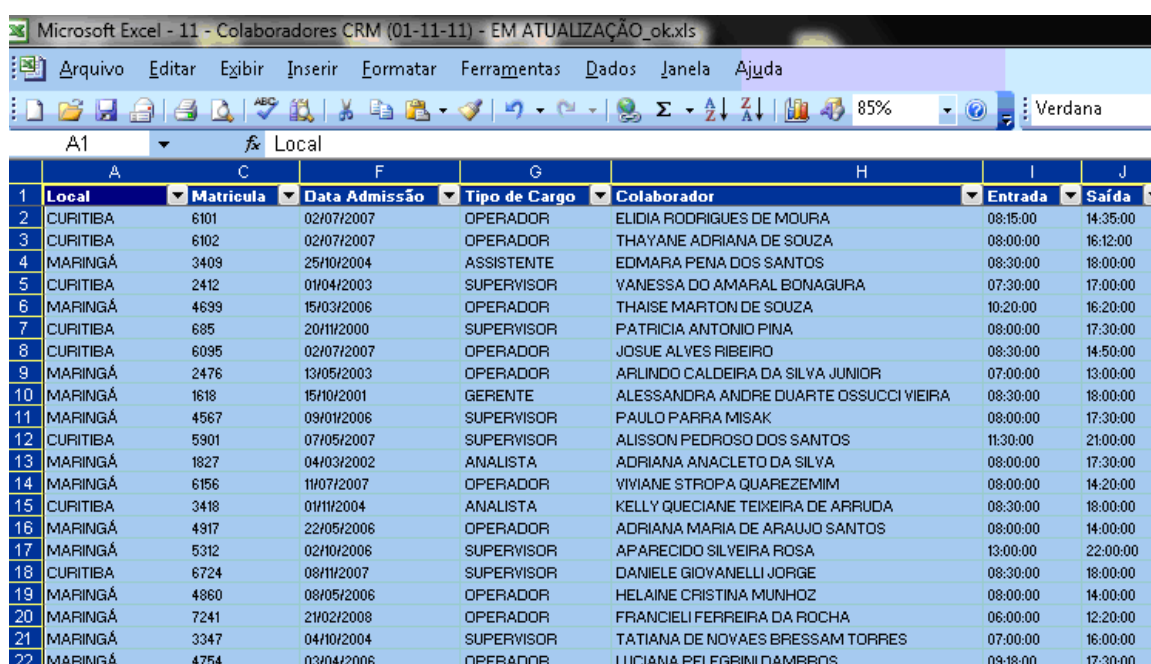
Segundo o projeto *Wide Workflow* é “um conjunto coordenado de atividades (seqüenciais ou paralelas) que são interligadas com o objetivo de alcançar uma meta comum”, sendo atividade conceituada como “uma descrição de um fragmento de trabalho que contribui para o cumprimento de um processo”. Assim, pode-se assumir que *workflow* é a divisão de um grande trabalho em várias tarefas menores, com pré-requisitos entre elas, que devem ser respeitados para o avanço da atividade (The wide workflow modeling language). Já segundo o *Workflow Management Coalition Workflow* é: “A automação de um processo de negócios, por inteiro ou em parte, durante o qual documentos, informações e tarefas são passadas de um participante para outro por ação respeitando um conjunto de regras *procedurais*”.

Segundo Tadeu, a mudança continua é a única certeza que temos o principal problema que toda organização enfrenta quanto a entender e executar processos advém do fato de que, ainda, hoje elas são estruturadas em funções. Se tivessem estruturas orientadas a processos, além de entendê-los com mais facilidade todos os empregados saberiam com segurança o próprio papel, suas responsabilidades e os resultados esperados de cada um deles. Esse tipo de estrutura é herança das idéias dos estudiosos que defendiam a tese de que uma atividade será realizada cada vez melhor se for repetidamente executada pelo mesmo trabalhador, sempre. Isso fez com que grupos de empregados, todos de uma mesma especialização, fossem concentrados em departamentos. Hoje a principal preocupação é definir as competências e o conhecimento necessário para fazer o trabalho e tomar decisões sobre como realizar cada tarefa. Embora possa parecer uma tecnologia que prescindia de tais capacidades, *Workflow*

é, pelo contrário, a tecnologia por excelência para dar ao trabalhador toda oportunidade de decidir o que fazer, pois a ferramenta fará o trabalho repetitivo e rotineiro e assumirá todas as tarefas que não requerem criatividade para serem executadas. (Uso e desuso de sistemas de *Workflow*.2006).

2.3 Base de desenvolvimento

Conforme citado no capítulo 1 desta obra o *SGC* (Sistema de Gerenciamento de Colaboradores) foi desenvolvido a partir do modelo antigo de armazenamento que era uma planilha em *Excel* conformem imagem abaixo.



| | A | C | F | G | H | I | J |
|----|----------|-----------|---------------|---------------|--|----------|----------|
| | Local | Matricula | Data Admissão | Tipo de Cargo | Colaborador | Entrada | Saída |
| 2 | CURITIBA | 6101 | 02/07/2007 | OPERADOR | ELIDIA RODRIGUES DE MOURA | 08:15:00 | 14:35:00 |
| 3 | CURITIBA | 6102 | 02/07/2007 | OPERADOR | THAYANE ADRIANA DE SOUZA | 08:00:00 | 16:12:00 |
| 4 | MARINGÁ | 3409 | 25/10/2004 | ASSISTENTE | EDMARA PENA DOS SANTOS | 08:30:00 | 18:00:00 |
| 5 | CURITIBA | 2412 | 01/04/2003 | SUPERVISOR | VANESSA DO AMARAL BONAGURA | 07:30:00 | 17:00:00 |
| 6 | MARINGÁ | 4639 | 15/03/2006 | OPERADOR | THAISE MARTON DE SOUZA | 10:20:00 | 16:20:00 |
| 7 | CURITIBA | 685 | 20/11/2000 | SUPERVISOR | PATRICIA ANTONIO PINA | 08:00:00 | 17:30:00 |
| 8 | CURITIBA | 6095 | 02/07/2007 | OPERADOR | JOSUE ALVES RIBEIRO | 08:30:00 | 14:50:00 |
| 9 | MARINGÁ | 2476 | 13/05/2003 | OPERADOR | ARLINDO CALDEIRA DA SILVA JUNIOR | 07:00:00 | 13:00:00 |
| 10 | MARINGÁ | 1618 | 15/10/2001 | GERENTE | ALESSANDRA ANDRE DUARTE OSSUCCI VIEIRA | 08:30:00 | 18:00:00 |
| 11 | MARINGÁ | 4567 | 09/01/2006 | SUPERVISOR | PAULO PARRA MISAK | 08:00:00 | 17:30:00 |
| 12 | CURITIBA | 5901 | 07/05/2007 | SUPERVISOR | ALISSON PEDROSO DOS SANTOS | 11:30:00 | 21:00:00 |
| 13 | MARINGÁ | 1827 | 04/03/2002 | ANALISTA | ADRIANA ANACLETO DA SILVA | 08:00:00 | 17:30:00 |
| 14 | MARINGÁ | 6156 | 11/07/2007 | OPERADOR | VIVIANE STROPA QUAREZEMIM | 08:00:00 | 14:20:00 |
| 15 | CURITIBA | 3418 | 01/11/2004 | ANALISTA | KELLY QUECIANE TEIXEIRA DE ARRUDA | 08:30:00 | 18:00:00 |
| 16 | MARINGÁ | 4917 | 22/05/2006 | OPERADOR | ADRIANA MARIA DE ARAUJO SANTOS | 08:00:00 | 14:00:00 |
| 17 | MARINGÁ | 5312 | 02/10/2006 | SUPERVISOR | APARECIDO SILVEIRA ROSA | 13:00:00 | 22:00:00 |
| 18 | CURITIBA | 6724 | 08/11/2007 | SUPERVISOR | DANIELE GIOVANELLI JORGE | 08:30:00 | 18:00:00 |
| 19 | MARINGÁ | 4860 | 08/05/2006 | OPERADOR | HELAINÉ CRISTINA MUNHOZ | 08:00:00 | 14:00:00 |
| 20 | MARINGÁ | 7241 | 21/02/2008 | OPERADOR | FRANCIELI FERREIRA DA ROCHA | 06:00:00 | 12:20:00 |
| 21 | MARINGÁ | 3347 | 04/10/2004 | SUPERVISOR | TATIANA DE NOVAES BRESSAM TORRES | 07:00:00 | 16:00:00 |
| 22 | MARINGÁ | 4754 | 03/04/2006 | OPERADOR | LUCIANA PELEGRINI DAMBROS | 09:18:00 | 17:30:00 |

Figura 1 - Controle de colaboradores modelo Excel (GVT 2010/2011).

A partir deste modelo foi criado um banco de dados para armazenamento das informações assim como seus respectivos diagramas. Por fim para manipulação destes dados foi construído um sistema de controle de funcionários com *interface Web*.

3 METODOLOGIA

3.1 Conteúdos da Metodologia

Neste capítulo é demonstrado a metodologia utilizada na construção deste trabalho que foi construído basicamente em três etapas:

- a) Levantamento dos requisitos;
- b) Cronograma;
- c) Projeto e implementação (Desenvolvimento, Integração e Implantação);

As etapas são descritas nas seções a seguir.

3.2 Levantamento de requisitos

Após o estudo inicial referente à regra de negócio da aplicação, o levantamento foi feito através de reuniões dentro da área do CRM da empresa GVT com a necessidade de implantação da ferramenta, buscando assim interpretar as maiores dificuldades para desenvolvimento de solução ideal de software. A partir de discussões foi possível reunir as informações necessárias para o projeto e implementação do banco de dados.

Além disso, foram desenvolvidos alguns diagramas e fluxogramas para interpretar todas as funcionalidades da ferramenta, o uso do software MS Visio foi essencial nesta etapa propiciando mais velocidade e entendimento no desenvolvimento do sistema futuro.

3.3 Cronograma

O cronograma foi criado após o levantamento de requisitos item 3.2 deste trabalho, assim foram divididas as tarefas e por fim gerado um cronograma a partir de dias úteis de cada mês.

A referencia entre os índices 1 ao 7 são referentes ao desenvolvimento do sistema e do 7 ao 10 referentes a documentação foram trabalhadas aproximadamente 3 horas por dia.

Tabela 1 - Cronograma

| Indice | Etapa | Previsão | Data Inicio | Data Fim | Predecessor |
|--------|--|------------------|-------------------|-------------------|-------------|
| 1 | Pesquisa das tecnologias a serem usadas. | 80 horas | 27/05/2011 | 04/07/2011 | |
| 2 | Estudo e desenvolvimento da regra de negocio | 40 horas | 04/07/2011 | 21/07/2011 | |
| 3 | Desenvolvimento do DER e scripts para criar base de dados | 100 horas | 21/07/2011 | 08/08/2011 | 1,2 |
| 4 | Desenvolvimento da aplicação | 150 horas | 08/08/2011 | 14/10/2011 | 1, 2, 3 |
| 5 | Pesquisa das tecnologias novas a serem usadas | 30 horas | 14/10/2011 | 28/10/2011 | |
| 6 | Desenvolvimento da aplicação utilizando novas tecnologias | 150 horas | 31/10/2011 | 06/01/2012 | 5 |
| 7 | Testes e refinamento | 60 horas | 08/01/2012 | 03/02/2012 | 6 |
| 8 | Documentação | 380 horas | 06/02/2012 | 01/06/2012 | 6 |
| 9 | Validação do projeto (montar apresentação) | 60 horas | 01/06/2012 | 20/06/2012 | 6 |
| 10 | Documentação Final (revisar confeccionar) | 40 horas | 20/06/2012 | 29/06/2012 | 9 |

3.4 Projeto e implementação

Nesta etapa, o desenvolvimento do projeto e a implementação do sistema foram feitas simultaneamente, ou seja, conforme análise de cada discussão com o cliente novas idéias e melhorias foram surgindo assim permitindo realizar mudanças em tempo de implementação, sendo que a cada resultado obtido foi possível estudar e implantar a melhor solução. Dessa maneira foram desenvolvidos vários protótipos para testar a aplicação e para tomadas de decisões no desenvolvimento do projeto final.

Abaixo, segue as fases e atividades desenvolvidas para alcançar o objetivo deste trabalho.

3.4.1 Estudo da tecnologias

Várias tecnologias foram estudadas para implementação do projeto, sendo que a maioria foi usada durante o curso de Tecnologia em Sistemas para Internet da UTFPR. Para concluir o projeto no melhor prazo possível, primeiro foi estudado qual o banco de dados que iria armazenar as informações da aplicação. Dentre eles os principais bancos estudados foram: Oracle, Mysql, Postgresql e Oracle Express. Nesta etapa foi analisado o comportamento de cada banco assim como sua sintaxe, podendo assim escolher o que mais se adequaria a aplicação a ser desenvolvida.

Quanto a aplicação as tecnologias utilizadas foram a linguagem Java, páginas JSP (Java Server Pages), HTML(HyperText Markup Language), XHTML (eXtensible Hypertext Markup Language), Framework JSF(Java Server Faces), Spring Security 3, a API (Hibernate) e a IDE Netbeans.

Dentre todos os objetivos, o primeiro passo foi o levantamento de requisitos, aspecto muito importante, pois com ele definido a assertividade no desenvolvimento do cronograma aumentaria.

Após isso foram estudadas outras tecnologias e aplicadas no desenvolvimento do projeto para solucionar algum problema ou alcançar objetivos.

No desenvolvimento utilizou-se JPA(Java Persistence API), além de algumas suítes de componentes para JSF como Primefaces, Icefaces e Richfaces, JSTL(JavaServer Pages Standard Tag Library), Servlets, Facelets, Hibernate, Spring, DAOS e comandos SQL;

Para construção do layout das páginas foram utilizados alguns componentes de CSS(Cascading Style Sheets), JavaScript e HTML(HyperText Markup Language).

Para o controle de versão utilizou-se Tortoise SVN.

No design da aplicação foram utilizados modelos de TCC, UML, Modelos de DER (Diagrama Entidade Relacionamento), MER (Modelo Entidade Relacionamento).

O resultado desta primeira fase foi à consolidação das informações e a escolha dos principais componentes a serem utilizados na estruturação do projeto, sendo assim os critérios para escolha foram medidos conforme os comportamentos e eficiência de cada tecnologia.

A principal dificuldade encontrada foi estudar o comportamento de cada tecnologia para assim poder escolher o pacote de componentes ideal para o desenvolvimento desta aplicação.

3.4.2 Modelagem do banco de dados

Depois do estudo das tecnologias o primeiro passo foi a modelagem do banco de dados, para isso foram utilizados conhecimento adquiridos durante o curso. Para criação do DER (Diagrama entidade relacionamento) foi instalado e utilizado o software DBDesigner onde permite a exportação para os bancos Mysql e Oracle. Foram gerados scripts de criação para os dois modelos de bancos de dados assim podendo a aplicação escolher este database. Nesta etapa foram instalados os softwares wamp para banco de dados Mysql e PLSQL Developer para manipular banco de dados Oracle, também neste momento foram estudados a sintaxe SQL e comportamento de cada banco de dados.

O resultado desta segunda fase foi a exploração de funcionalidades de cada banco de dados podendo assim escolher o que supra todas as necessidades desta aplicação.

As dificuldades encontradas nesta etapa foram estudar a sintaxe de cada banco e quais os diferenciais existentes entre eles.

3.4.3 Protótipo

Depois de toda a modelagem do banco de dados, o momento agora seria desenvolver alguns protótipos da aplicação, para aprofundamento no estudo da tecnologia, alguns softwares foram instalados nesta etapa para auxiliar no manuseio do projeto, por exemplo, a IDE Netbeans e Tortoise SVN para controle de versões.

Com isso foi criada o primeiro template de um código Java que permitiu a persistência de dados, as primeiras classes seguindo o padrão de desenvolvimento MVC e após isso as primeiras paginas para visualização da aplicação.

Todas as implementações ao gerar a persistência dos dados foram positivas, com isso foi possível desenvolver as primeiras paginas.

A principal dificuldade encontrada nesta etapa foi entender a regra de navegação de cada tecnologia, assim como manter e transitar os dados de uma pagina para outra.

3.4.4 Desenvolvimento e integração dos serviços

Depois de criado o protótipo o próximo passo seria o desenvolvimento da aplicação, ou seja, aplicar toda a regra de negócio. Nesta etapa integramos todos os serviços da aplicação com a base de dados, gerando classes de persistência. Assim que estabelecida a comunicação entre a aplicação e a base de dados outras funcionalidade puderam ser adicionados, para melhorar a usabilidade da ferramenta um dos componentes utilizados foi o Primefaces e gráficos dinâmicos (FusionCharts).

O resultado foi um sistema com todas as camadas concluídas segundo padrão MVC preparado para bateria de testes.

A principal dificuldade desta etapa foi a customização para gerar gráficos e integrar serviços.

3.4.5 Teste geral

Após o desenvolvimento foram realizados uma bateria de testes para garantir a integridade da ferramenta assim como sua confiabilidade nas informações. Os principais testes foram de inserção, atualização e remoção de informações.

Algumas falhas foram encontradas e anotadas passando para fase de correções em "customização da aplicação".

Nesta fase a dificuldade foi criar a classe Java para simular usuários no sistema e gerar bateria de testes para encontrar falhas.

3.4.6 Customização da aplicação

Após os testes da ferramenta, foram encontradas inconsistências na aplicação, podendo serem corrigidas todas em tempo de execução. Também teve de ser alterada a camada de visualização com objetivo de tornar a interface agradável ao usuário final, deixando cada tela mais intuitiva e clara possível.

A complexidade e customização de cada página passaram a ser muito grande, mesmo com o uso de bibliotecas ágeis para desenvolvimento. A solução foi criar dentro do mesmo projeto dois templates, um para área pública e outro para área administrativa.

Como resultado, após essa fase foi possível concluir o sistema.

A maior dificuldade desta fase foi criar um modelo ideal de pagina para a aplicação.

3.5 Recursos empregados

Os recursos utilizados neste projeto.

3.5.1 Recursos financeiros

Os recursos financeiros utilizados foram para impressão de documentos, os softwares utilizados possuem licença gratuita, alguns softwares possuem registros oficiais, como por exemplo, Windows 7, MS Visio, MS Word e os computadores utilizados eram de propriedade do acadêmico. Nenhum grande gasto financeiro foi necessário para o projeto.

3.5.2 Recursos pessoais

O aluno e professor orientador foram os 2 recursos pessoais envolvidos neste projeto.

3.5.3 Recursos de hardware

O Um computador foi utilizado para desenvolvimento do projeto:

Tabela 2 - Configuração do computador

| | |
|---------------------|--------------------------|
| Propriedade | Marlon Jair da Silva |
| Sistema Operacional | Microsoft Windows 7® |
| Memória | 2 GB |
| Disco Rígido | 500 GB |
| Processador | Intel Core i3 – 2.27 ghz |

3.5.4 Recursos de software

Para desenvolvimento deste projeto, foram utilizados diversos softwares, para poder apoiar a sua modelagem e implementação, a seguir segue descrição em tópicos de cada software.

3.5.4.1 Softwares utilizados

Nesta seção serão descritos os softwares utilizados para desenvolvimento do projeto.

a) Netbeans

IDE (Integrated Development Environment) Software gratuito para ambiente de desenvolvimento de diversas tecnologias, todas as ferramentas necessárias para criar ambiente de trabalho profissional, empresarial, web e aplicações de dispositivos moveis com plataforma Java, suporte para outras linguagens como C/C++, PHP, Java Script. Possui uma gama de recursos como palet para swing e depuradores de código para web e desktop. Versão atual 7.1.1, utilizada neste projeto.

b) Tortoise SVN

O Tortoise SVN software gratuito para controle de versões faz interface entre o usuário e as funcionalidades do Subversion. Facilitando o uso do Subversion.

c) Subversion

Software gratuito para ambiente Windows foi utilizado para controle de versões na implementação do projeto. Também pode ser integrada as IDEs mais conhecidas. Este programa também auxilia no controle de versão de pastas e documentos.

d) Oracle 10g Express

Software gratuito para banco de dados Oracle, é um SGBD (Sistema de gerenciamento de banco de dados) voltado para aplicações menores que

necessitem de no máximo 4GB de armazenamento, Possui ferramentas gráficas para o auxílio na criação de tabelas e manipulação de dados.

e) Apache Tomcat

Software gratuito é um servidor web multiplataforma, desenvolvido pela Apache Software Foundation. O Tomcat implementa tecnologias como o JSP, JSF utilizado neste projeto específico para aplicações Java

A versão utilizada neste projeto é a 7.0, versão atual e mais estável recomendada.

f) Wamp

Software gratuito para bancos de dados MySQL, desenvolvido pela Apache Software Foundation sua sigla significa W-Windows A-Apache M-MySQL e P-PHP ou seja são vários softwares instalados integrados para agilizar a utilização dos mesmos, possui interface gráfica com o usuário. A versão utilizada é a 5 versão atual.

g) DB Designer

Software gratuito para modelagem do banco de dados, criar tabelas, gerar scripts, desenhar todo o DER(Diagrama entidade relacionamento) da base de dados. Versão utilizada 4.0.5.6 por ser mais recente.

h) MS Visio

Para criar todos os fluxos de Workflow do projeto, diagramas de casos de uso, diagramas de classe, diagramas de sequencia.

Software com licença oficial. Versão MS Visio 2003.

i) MS Word

Para gerar toda a documentação do projeto. Software com licença oficial. Versão MS Word 2007.

j) MS Excel

Para gerar cronograma do projeto, e todas as tabelas da documentação. Software com licença oficial. Versão MS Excel 2007.

k) MySQL Query Browser

Programa gratuito para criar todas as consultas utilizadas no projeto. Área de interface entre o banco de dados Mysql.

l) Toad

Programa gratuito para criar todas as consultas utilizadas no projeto. Área de interface entre o banco de dados Oracle.

3.5.4.2 Tecnologias utilizadas

Nesta seção serão descritos as tecnologias utilizadas para desenvolvimento do projeto.

Java (linguagem de programação)

Tecnologia orientada a objetos, usado para desenvolver sistemas para desktop, sistemas web e sistemas embarcados. A plataforma possui divisões em J2SE que é desenvolvimento para Desktop, J2EE para ambientes empresarial e J2ME para dispositivos moveis. O Java possui o conceito de JVM(Java Virtual Machine) ou seja a linguagem de programação possui compatibilidade com qualquer plataforma que utilize JVM. Neste projeto foi utilizado o kit para desenvolvimento Java versão jdk1.7.0_03 com Java Enterprise Edition J2EE, no momento a mais recente do mercado.

b) Java Server Pages (JSP).

Java Server Pages é uma tecnologia usada para desenvolvimento de aplicações web, baseada em linguagem de programação Java, o JSP tem a vantagem de portabilidade entre plataformas permitindo assim execução em diferentes sistemas operacionais, além de trabalhar com Servlets. O JSP foi utilizando em conjunto com o JSF.

c) Java Server Faces (JSF)

O Java Server Faces é um framework MVC (Model View Controller) de aplicações Web, baseado no Java é a tecnologia padrão J2EE a sua idéia é simplificar o desenvolvimento de interfaces para o usuário. Com o uso deste framework é possível separar melhor as camadas da aplicação. A versão utilizada é a 2.0 no momento a mais recente do mercado.

d) Primefaces

Primefaces é uma biblioteca de componentes para Java Server Faces. É certamente uma das mais completas com mais de 200 componentes e foi uma das primeiras a estar totalmente convertida para JSF 2.0 a versão utilizada neste projeto foi a 3.2 no momento a mais recente do mercado.

e) JPA / Hibernate

O JPA é mais recente especificação criada para permitir persistência objeto-relacional com bancos de dados relacionais. Com essa API(Application Programming Interface) é possível transformar requisições de salvar ou consultar via classes Java carregadas de informações em comandos SQL que são enviados futuramente ao banco de dados. Versões utilizadas, JPA 2.0 e Hibernate 3.5, as mais recentes do mercado.

f) Java Script

Java Script é uma linguagem de programação dinâmica para web. Interagindo diretamente com o usuário o Java script permite adicionar componentes de interface dinâmicos na pagina, suportado em todos os navegadores é um dos principais componentes web e foi utilizado em todas as paginas deste trabalho.

h) CSS

CSS Cascading Style, é uma linguagem de estilos, utilizada para criar a apresentação da pagina no navegador. Utilizada para editar cores, posicionar elementos, estatizar fontes, em fim montar todo o layout da pagina.

i) XML

XML (eXtensible Markup Language) é uma linguagem de marcação criada pelo W3C, permite representar informações de maneira estruturada.

Essa tecnologia é, segundo a W3C, uma das linguagens mais utilizadas no mundo para compartilhar informações estruturadas entre softwares, computadores e pessoas, tanto localmente como via internet. Neste projeto o XML é utilizado principalmente para estruturar arquivos de configuração em conjunto com a linguagem Java.

Neste capítulo foram apresentados métodos utilizados de recursos de software, hardware, recursos financeiros. No capítulo a seguir serão apresentados os resultados referentes a este projeto.

4 RESULTADOS

Neste capítulo serão apresentados os resultados obtidos durante a elaboração deste trabalho, ou seja, toda a modelagem deste projeto.

4.1 Descrição da arquitetura

Este trabalho foi desenvolvido seguindo os conceitos de programação orientada a objeto no padrão MVC (Model View Controller) seguindo fortemente os padrões de desenvolvimento *Controller*, *Abstract Factory*, *DAO*, *Façade* que foram estudados e utilizados durante o curso. Para desenvolvimento deste trabalho, os programas mais utilizados foram: TortoiseSVN (controlador de versões) o compilador Netbeans IDE, MySQL Query Browser para consultas MySQL e Toad para manipular a base oracle.

Os frameworks utilizados foram JSF, JPA/Hibernate e Primefaces pois se integram facilmente dando velocidade de desenvolvimento da aplicação.

O projeto foi dividido em 4 camadas:

Modelo: Na camada de modelo ou *POJO*, trata-se de um objeto utilizado para implementar o modelo de domínio da aplicação, ou seja utilizado para trafegar um conjunto de valores entre as camadas e classes da aplicação.

View: Na camada de view contem todas as paginas da aplicação, estas são processadas do lado do cliente, assim como todas as funcionalidades da aplicação, ou seja, nela são processadas as funções JavaScript (Jquery) e o estilo da pagina (CSS).

Controller: Para cada pagina JSF deve haver um Managed Bean (MBean) que possui código Java de servidor que cuida de responder as requisições do usuário, ou seja é a lógica da aplicação da pagina web que o usuário esta utilizando.

Regra de Negócio: A camada de regra de negócio ou *Façade* é o ponto de chamada para utilizar funcionalidades de negócio da aplicação.

4.2 Requisitos funcionais e não funcionais

4.2.1 Requisitos funcionais

Neste capítulo serão abordados os requisitos funcionais do sistema.

Controle de acesso: o sistema deverá solicitar usuário e senha para autenticação.

Gerenciamento de acesso: cada usuário terá uma interface específica dependendo do seu nível de permissão.

Busca de colaboradores: Ao autenticar o usuário poderá buscar todos os colaboradores cadastrados na ferramenta, assim como todos os colaboradores desativados da ferramenta.

Listar Equipe: O usuário poderá visualizar a sua equipe de trabalho, em uma visão hierárquica ou de tabela.

Transação: Dependendo do nível da hierarquia o usuário poderá movimentar alterar ou desativar um colaborador a ele submisso.

O sistema deverá manter todos os logs de todas as transações efetuadas na ferramenta.

Área administrativa: O sistema deverá ter funcionalidades pra administração de usuários e controle de permissão dos mesmos. O usuário administrador deverá ter todos os privilégios para moderar o sistema.

4.2.2 Requisitos não funcionais

Neste capítulo serão abordados, requisitos não funcionais do sistema

O sistema deverá possuir mecanismos de segurança, autenticação de usuário e senha.

O sistema deverá oferecer um bom desempenho, ou seja, uma boa navegação e um menor tempo possível entre trocas de telas.

Sistema disponível para os principais navegadores utilizados hoje.

Interface intuitiva para que sua usabilidade seja de forma fácil e ágil para o usuário.

Manutenção rápida, ou seja, um sistema bem projetado e desenvolvido para reparos instantâneos.

4.3 Casos de uso

Os casos de uso descrevem um possível cenário entre as possíveis funcionalidades que o usuário poderá utilizar ao se interagir com o sistema.

4.3.1 Caso de uso geral

O diagrama abaixo representa o caso de uso geral do sistema.

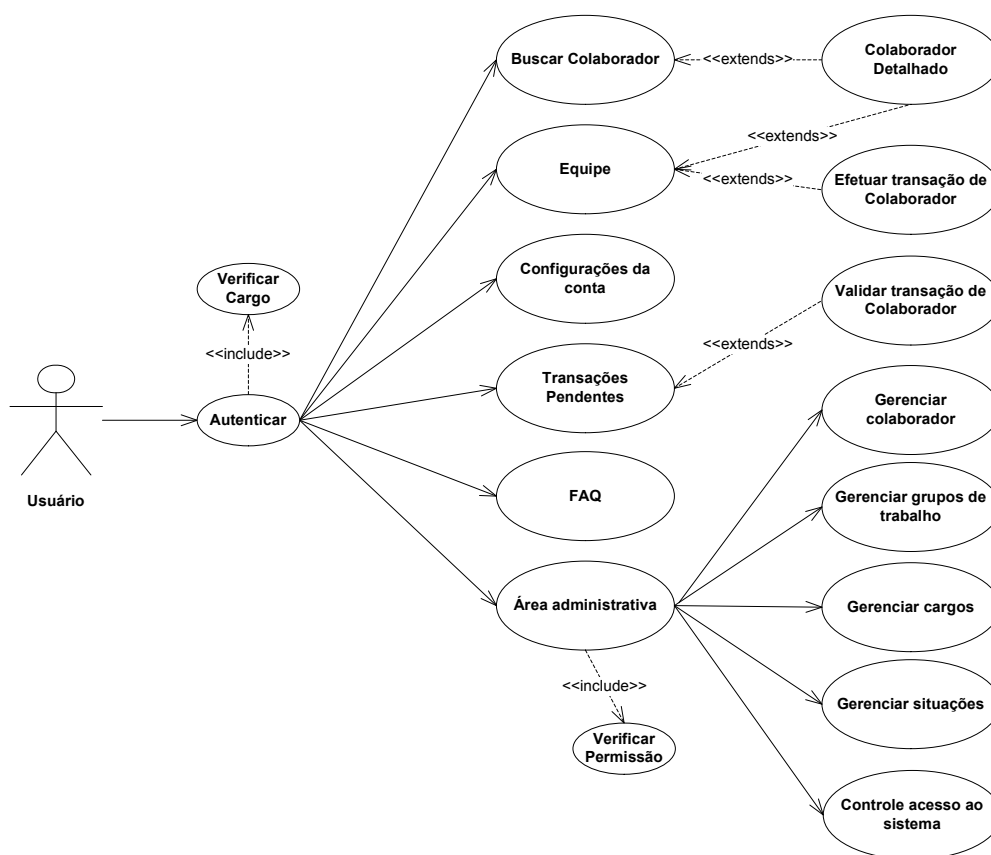


Figura 2- Caso de uso 'geral'

O diagrama representado pela Figura 2 tem como objetivo mostrar de maneira geral todas as funcionalidades do sistema. Nas seções a seguir serão apresentados cada diagrama detalhado.

4.3.2 Autenticar

O diagrama da figura 3 apresenta o caso de uso 'Autenticar'.

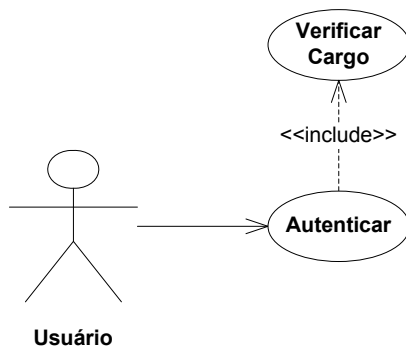


Figura 3 – Caso de uso 'autenticar'

Para acessar o sistema o usuário necessita de um login e uma senha. O caso de uso 'Autenticar' verifica se as credenciais do usuário estão cadastradas no sistema e qual o tipo de visualização este usuário poderá ter. Os menus de atualizações, inserções no sistema são de acordo com o usuário, por exemplo, um gerente tem permissões diferentes de um analista, ou seja, os acessos estão restritos de acordo com o cargo do funcionário.

Pré-condições o usuário deverá ser cadastrado no sistema.

Pós-condições o sistema valida as credenciais do usuário liberando ou não o acesso a aplicação.

4.3.3 Buscar funcionário

O diagrama representado pela Figura 4 apresenta o caso de uso 'Buscar'.

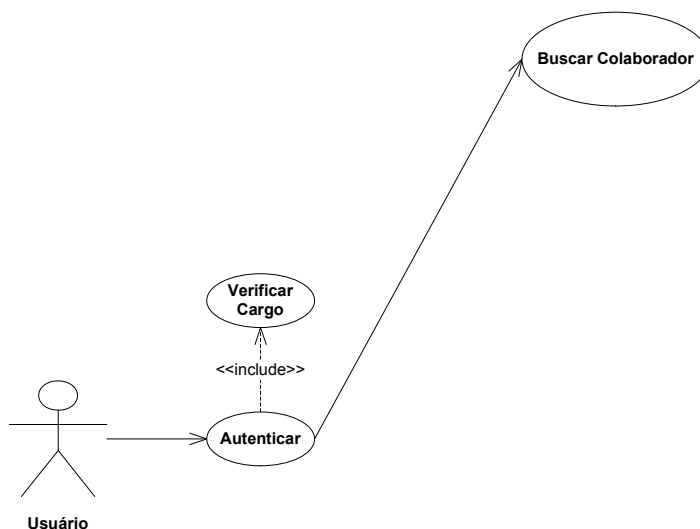


Figura 4 – Caso de uso 'buscar'

Ao buscar um colaborador cadastrado no sistema o usuário poderá refinar a busca ou clicar sob no nome de um colaborador para exibir suas informações detalhadas.

Pré-condições o usuário deverá tentar localizar um ou mais colaboradores cadastrados no sistema.

Pós-condições o sistema deverá exibir para os usuários a lista de colaboradores procurados com possibilidades de refinamento ou exibir todas as informações detalhadas daquele colaborador.

4.3.4 Busca funcionário detalhado

O diagrama representado pela Figura 5 apresenta o caso de uso buscar colaborador detalhado.

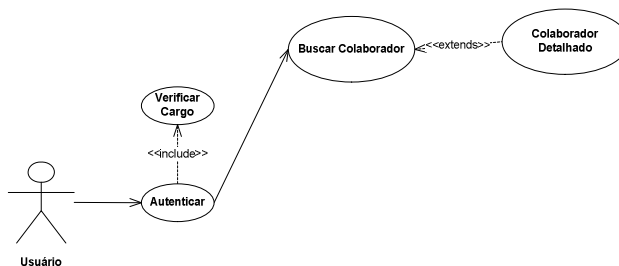


Figura 5 - Caso de uso colaborador detalhado'

Ao clicar sob no nome de um colaborador o sistema irá exibir as informações detalhadas daquele funcionário.

Pré-condições: o usuário devesse clicar sob o nome de um colaborador.

Pós-condições: o sistema devesse mostrar as informações detalhadas daquele colaborador.

4.3.5 Listar equipe

O diagrama representado pela Figura 6 apresenta o caso de uso Listar equipe.

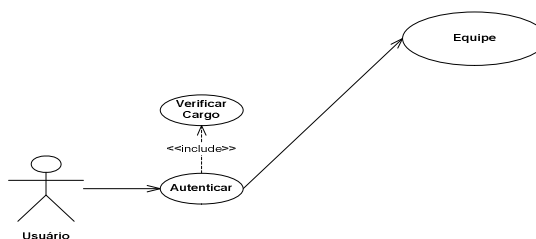


Figura 6 - Caso de uso 'listar equipe'

Ao listar equipe o sistema deverá mostrar para o usuário logado apenas sua equipe de trabalho.

Pré-condições: o usuário deverá acessar o menu 'Listar Equipe'.

Pós-condições: o sistema deverá mostrar as informações da equipe do usuário logado, restringindo o acesso ao menu de opções caso o usuário não seja gestor desta equipe.

4.3.6 Transacionar colaborador

O diagrama representado pela Figura 7 apresenta o caso de uso Transacionar colaborador.

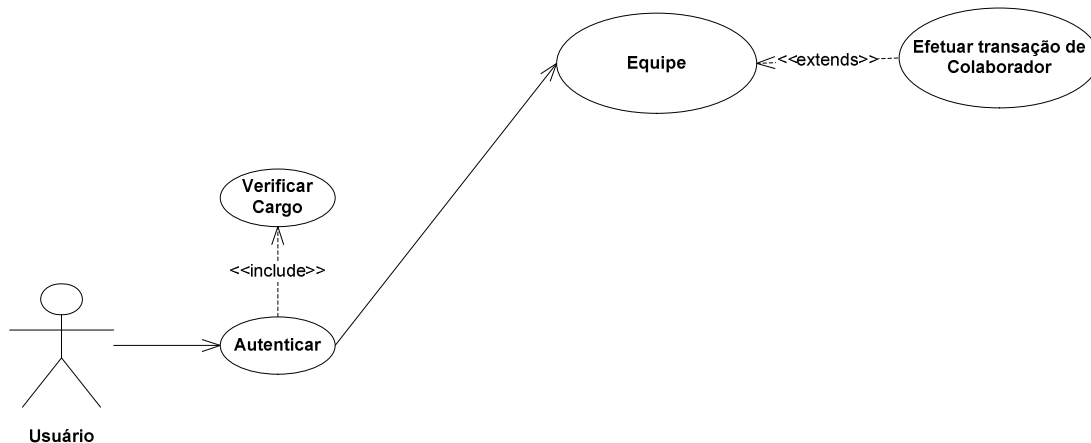


Figura 7 – Caso de uso ‘transacionar colaborador’

Ao listar equipe o sistema deverá mostrar para o usuário logado apenas sua equipe de trabalho, e se o usuário for gestor desta equipe ele poderá movimentar o colaborador para outra equipe ele também poderá desligar este colaborador ou alterar as informações detalhadas deste colaborador.

Pré-condições: o usuário deverá ser gestor, em seguida acessar o menu ‘Listar Equipe’ e clicar sob um colaborador.

Pós-condições: o sistema deverá mostrar as informações a serem atualizadas do colaborador selecionado.

4.3.7 Configurações de conta

O diagrama representado pela Figura 8 apresenta o caso de uso Configurações de conta.

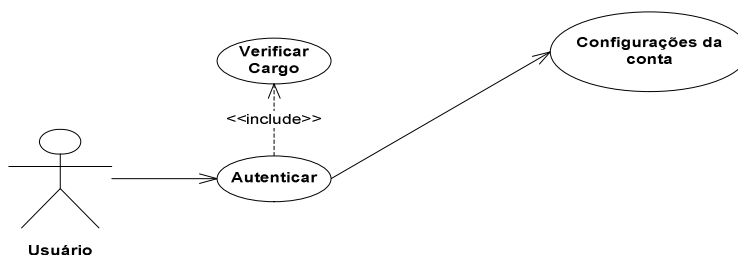


Figura 8 - Caso de uso 'configurações de conta'

Ao acessar configurações de conta o usuário poderá trocar sua senha de acesso ao sistema.

Pré-condições: o usuário deverá acessar o menu 'Configurar conta'.

Pós-condições: o sistema deverá mostrar as credenciais de acesso ao sistema, permitindo assim a alteração da senha.

4.3.8 Transações pendentes

O diagrama representado pela Figura 9 apresenta o caso de uso Transações Pendentes.

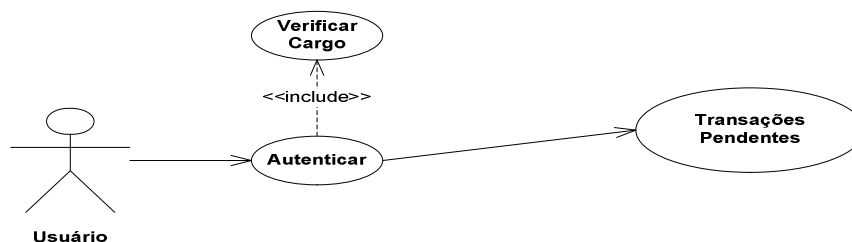


Figura 9 - Caso de uso 'transações pendentes'

Ao acessar o menu transações pendentes e caso usuário seja gestor, o mesmo irá visualizar todas as transações que estão ocorrendo em sua equipe.

Pré-condições: o usuário deverá ser gestor e acessar o menu de transações pendentes.

Pós-condições: o sistema deverá mostrar todas as transações pendentes na qual o usuário está participando.

4.3.9 Validar transações pendentes

O diagrama representado pela Figura 10 apresenta o caso de uso Validar transações pendentes.

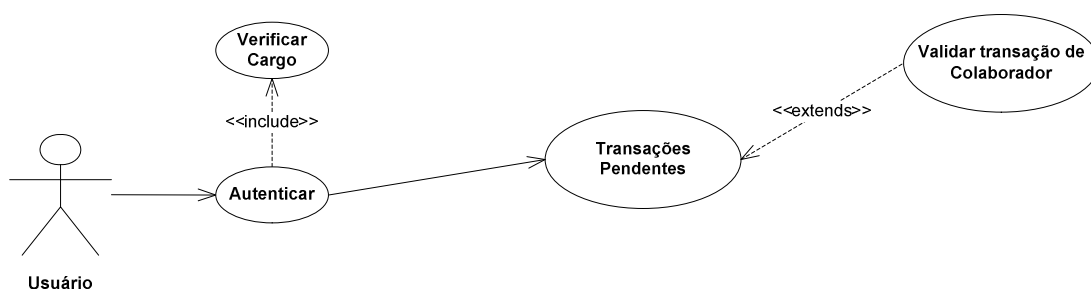


Figura 10 - Caso de uso 'validar transações pendentes'

Ao acessar o menu transações pendentes e caso usuário seja gestor, o mesmo irá visualizar todas as transações que estão ocorrendo em sua equipe, podendo aceitar ou negar caso transação dependa de sua participação.

Pré-condições: o usuário deverá ser gestor e acessar o menu de transações pendentes.

Pós-condições: o sistema deverá mostrar todas as transações pendentes na qual o usuário está participando.

4.3.10 Área administrativa

O diagrama representado pela Figura 11 apresenta o caso de uso Área administrativa.

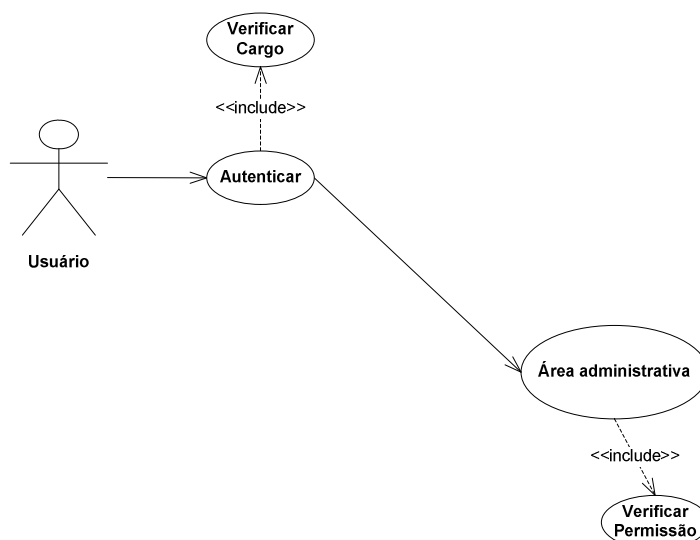


Figura 11 - Caso de uso 'área administrativa'

Ao acessar o sistema o usuário poderá acessar a área administrativa caso o mesmo esteja cadastrado como administrador da ferramenta.

Pré-condições: o usuário deverá ser administrador e acessar a área administrativa.

Pós-condições: o sistema deverá mostrar todas as opções administrativas da ferramenta.

4.3.11 Gerenciar colaborador

O diagrama representado pela Figura 12 apresenta o caso de uso Gerenciar colaborador.

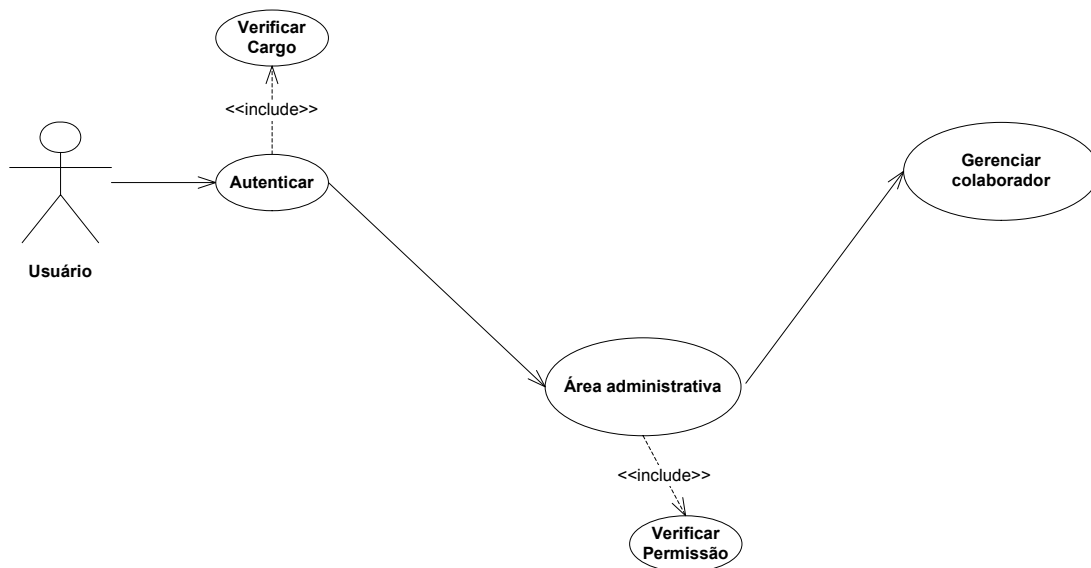


Figura 12 - Caso de uso 'gerenciar colaborador'

Ao acessar a área administrativa o usuário poderá cadastrar, atualizar ou remover um colaborador.

Pré-condições: o usuário deverá acessar o menu gerenciar colaborador e selecionar um colaborador ou cadastrar um novo.

Pós-condições: o sistema deverá mostrar todas as informações que o usuário poderá cadastrar, alterar ou remover um colaborador.

4.3.12 Gerenciar grupos

O diagrama representado pela Figura 13 apresenta o caso de uso Gerenciar grupos.

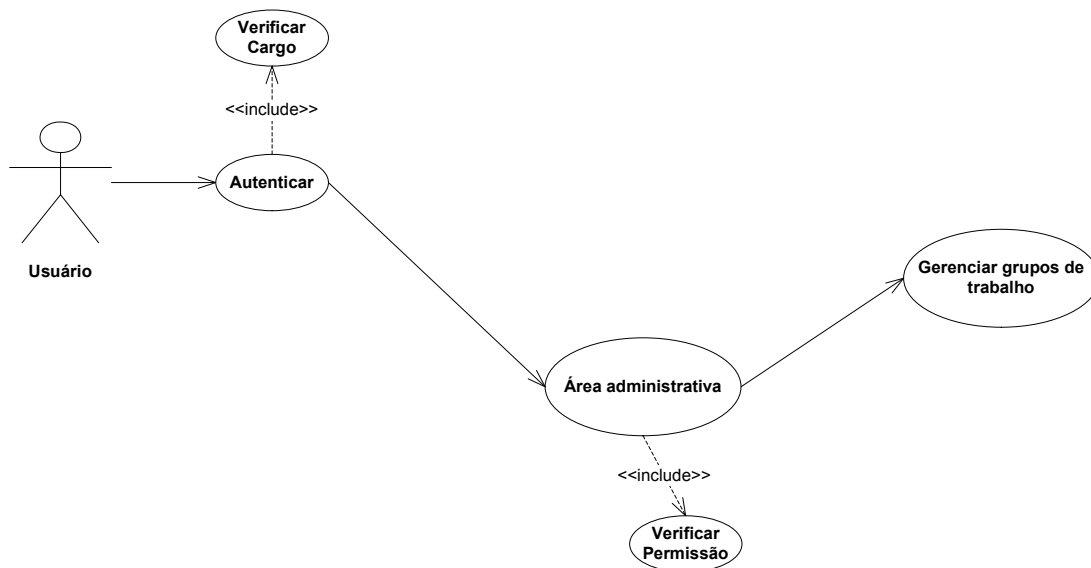


Figura 13 - Caso de uso 'gerenciar grupos'

Ao acessar a área administrativa o usuário poderá gerenciar as equipes de trabalho, podendo cadastrar, atualizar ou remover uma equipe.

Pré-condições: o usuário deverá acessar o menu gerenciar grupos e selecionar uma equipe para alterar, remover ou cadastrar uma nova.

Pós-condições: o sistema deverá mostrar todas as informações que o usuário poderá cadastrar, alterar ou remover um grupo.

4.3.13 Gerenciar cargos

O diagrama representado pela Figura 14 apresenta o caso de uso Gerenciar cargos.

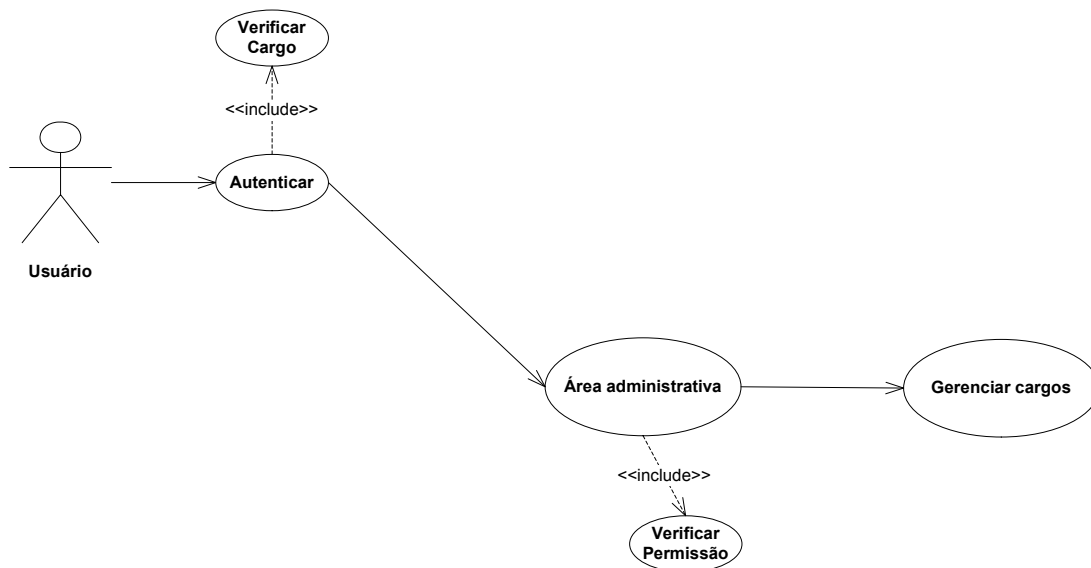


Figura 14 - Caso de uso 'gerenciar cargos'

Ao acessar a área administrativa o usuário poderá gerenciar os cargos dos funcionários podendo cadastrar, atualizar ou remover um cargo.

Pré-condições: o usuário deverá acessar o menu gerenciar cargos e selecionar um cargo para alterar, remover ou cadastrar uma novo.

Pós-condições: o sistema deverá mostrar todas as informações que o usuário poderá cadastrar, alterar ou remover um cargo.

4.3.14 Gerenciar situações

O diagrama representado pela Figura 15 apresenta o caso de uso Gerenciar situações

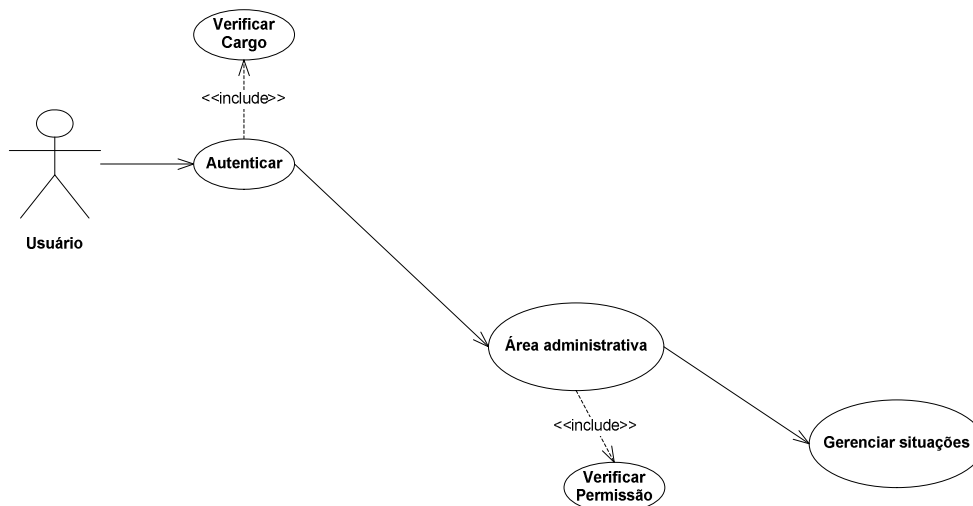


Figura 15 - Caso de uso 'gerenciar situações'

Ao acessar a área administrativa o usuário poderá gerenciar as situações de um funcionário, podendo cadastrar, atualizar ou remover uma situação.

Pré-condições: o usuário deverá acessar o menu gerenciar situações e selecionar uma situação para alterar, remover ou cadastrar uma nova.

Pós-condições: o sistema deverá mostrar todas as informações que o usuário poderá cadastrar, alterar ou remover uma situação.

4.3.15 Controle de acesso

O diagrama representado pela Figura 16 apresenta o caso de uso Controle de acesso.

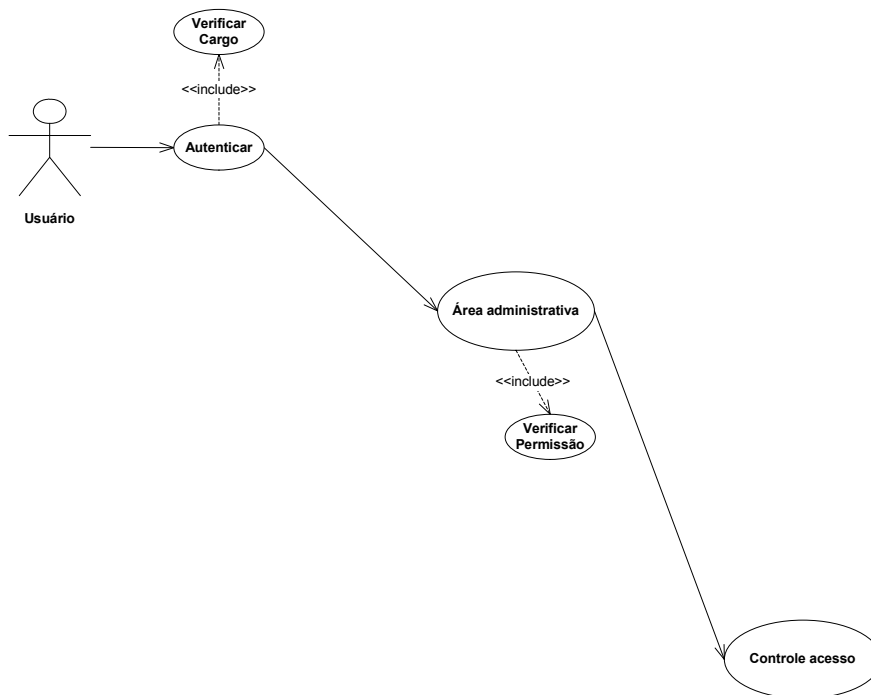


Figura 16 - Caso de uso 'controle acesso'

Ao acessar a área administrativa o usuário poderá gerenciar as permissões de controle de acesso ao sistema.

Pré-condições: o usuário deverá acessar o menu controle de acesso.

Pós-condições: o sistema deverá mostrar a lista de todos os colaboradores onde será possível conceder ou negar permissões de administrador a um funcionário.

4.4 Diagramas de seqüência

Nesta seção são descritos os diagramas de sequencia desta aplicação, para demonstrar e detalhar o fluxo de ações deste sistema.

4.4.1 Autenticação

A figura 17 apresenta o diagrama de seqüência referente à autenticação do sistema.

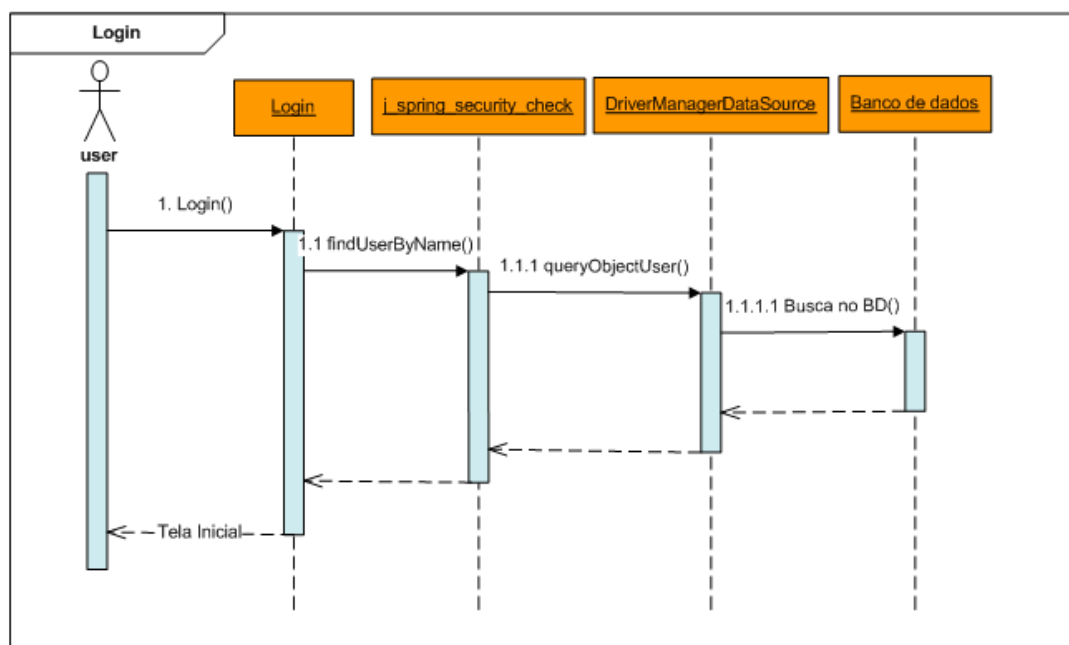


Figura 17 - Diagrama de seqüência 'login'

Após autenticação do usuário, o sistema irá reconhecê-lo e por fim liberar ou restringir acesso em alguns menus do sistema, conforme cada cargo/perfil de cada usuário.

4.4.2 Buscar Colaborador

A figura 18 apresenta o diagrama de seqüência referente a busca de um colaborador no sistema.

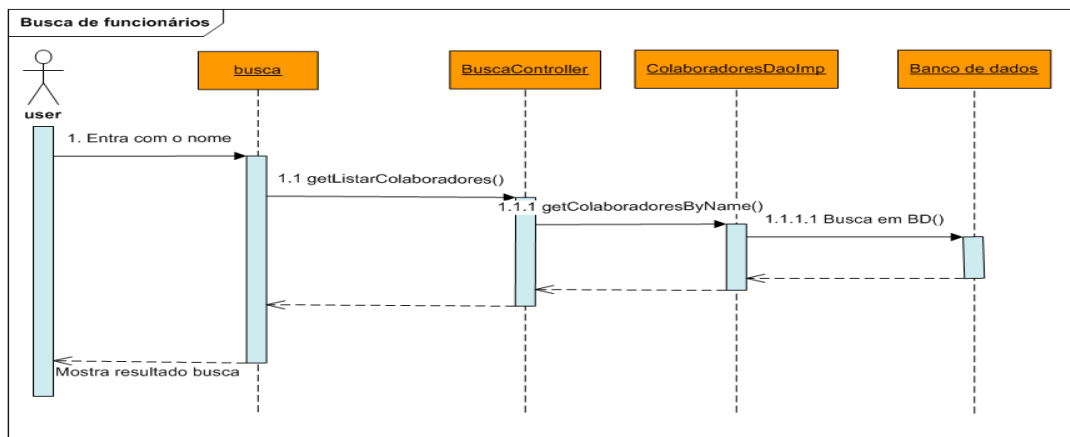


Figura 18 - Diagrama de seqüência 'buscar colaborador'

Após autenticação no sistema o usuário poderá procurar um colaborador cadastrado na base de dados.

4.4.3 Listar Equipe

A figura 19 apresenta o diagrama de seqüência referente a listar equipe do colaborador logado no sistema.

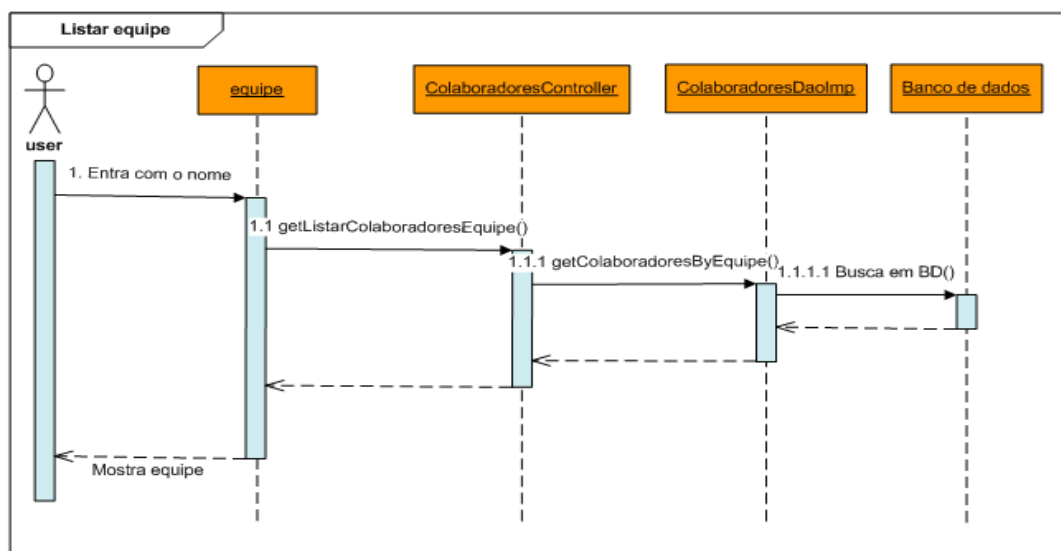


Figura 19 - Diagrama de seqüência 'listar colaborador'

Após autenticação no sistema o usuário poderá listar sua equipe de trabalho.

4.4.4 Novo Colaborador

A figura 20 apresenta o diagrama de seqüência referente a adição de um novo colaborador no sistema.

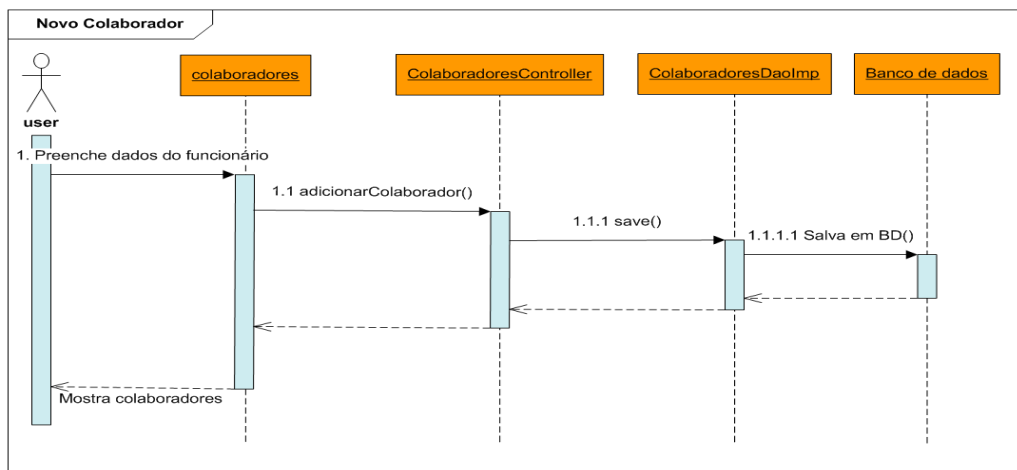


Figura 20 - Diagrama de seqüência 'novo colaborador'

Após autenticação no sistema o usuário poderá cadastrar um colaborador no sistema.

4.4.5 Alterar Colaborador

A figura 21 apresenta o diagrama de seqüência referente à alteração de um colaborador no sistema.

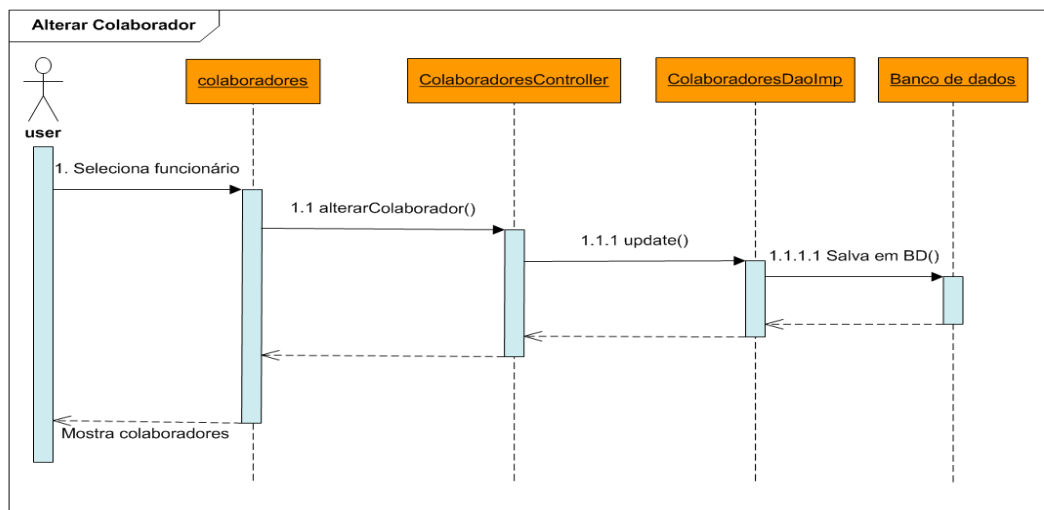


Figura 21 - Diagrama de seqüência 'alterar colaborador'

Após autenticação no sistema o usuário poderá alterar um colaborador cadastrado na base de dados.

4.4.6 Remover Colaborador

A figura 22 apresenta o diagrama de seqüência referente a remoção de um colaborador no sistema.

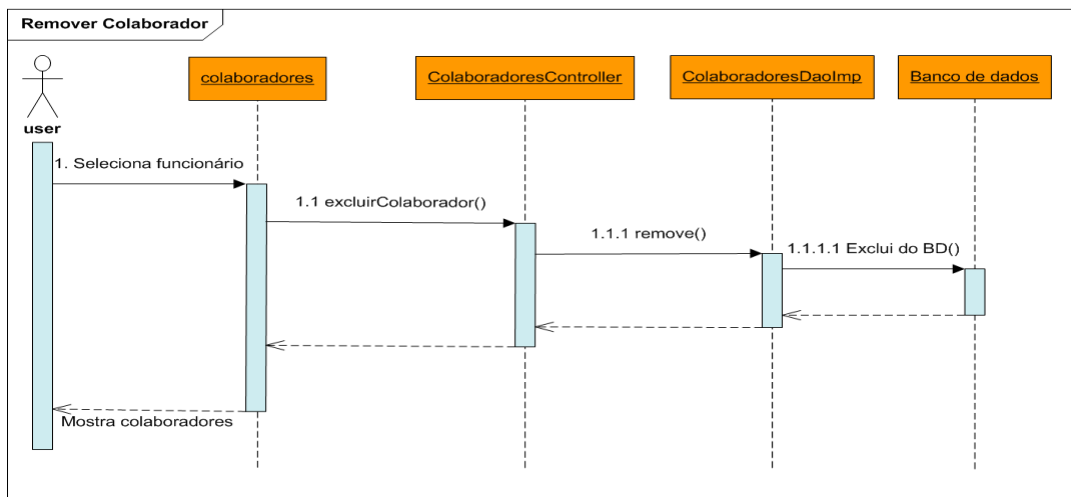


Figura 22 - Diagrama de seqüência 'remover colaborador'

Após autenticação no sistema o usuário poderá remover um colaborador cadastrado na base de dados.

4.4.7 Transacionar Colaborador

A figura 23 apresenta o diagrama de seqüência referente a transação de um colaborador no sistema.

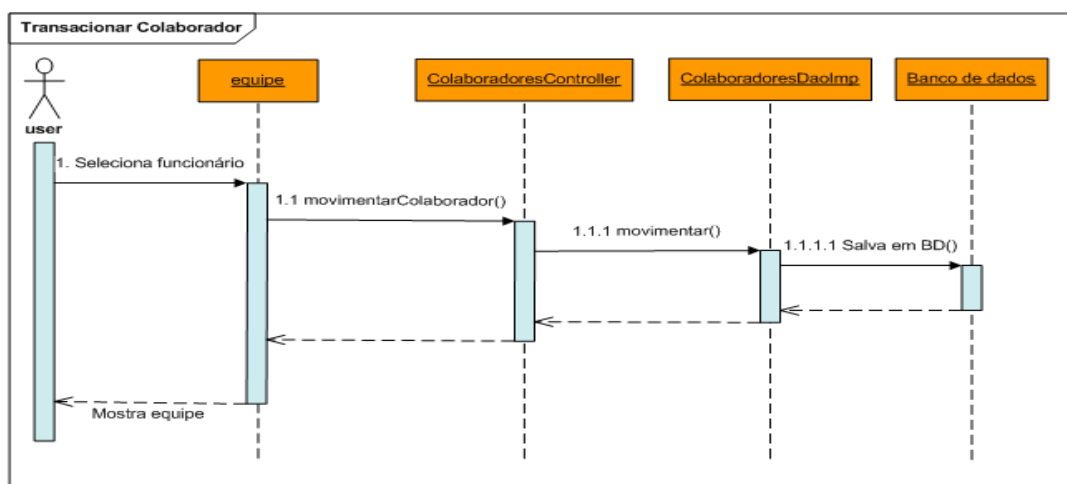


Figura 23 - Diagrama de seqüência 'transacionar colaborador'

Após autenticação no sistema o usuário poderá transacionar um colaborador cadastrado na base de dados.

4.4.8 Transação Colaborador Aceitar

A figura 24 apresenta o diagrama de seqüência referente ao aceite de uma transação de um colaborador a ser realizada no sistema.

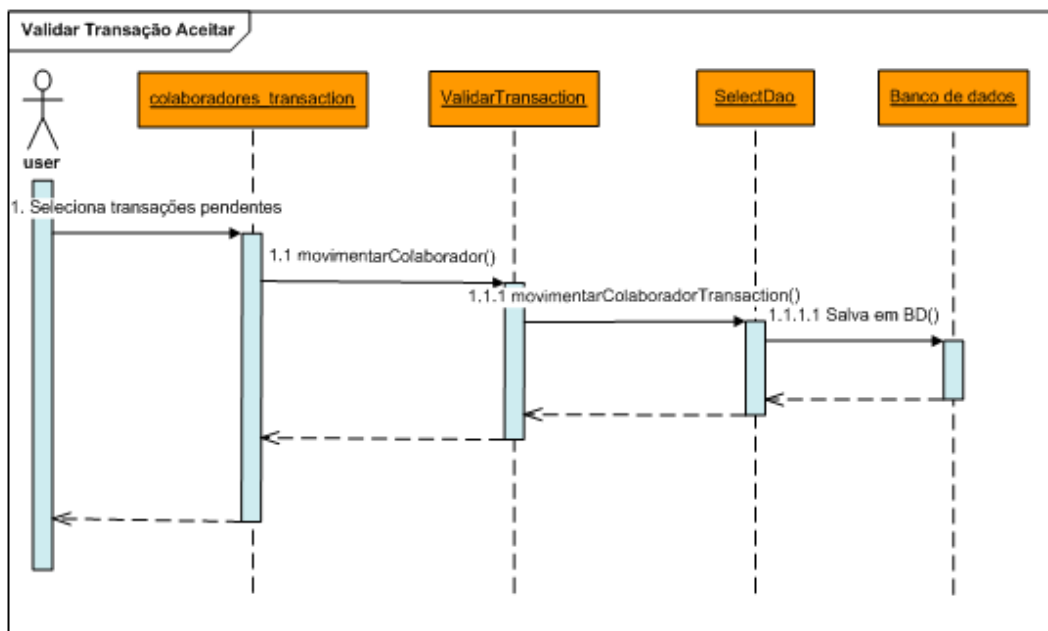


Figura 24 - Diagrama de seqüência 'validar transação aceitar'

Após uma transação de um colaborador feita por um gestor de origem será enviado para o gestor de destino uma notificação no sistema que o mesmo possui uma transação pendente, onde poderá aceitar, assim o colaborador deixara a equipe de origem e passara a ser da equipe de destino.

4.4.9 Transação Colaborador Recusar

A figura 25 apresenta o diagrama de seqüência referente a recusa de uma transação de um colaborador no sistema.

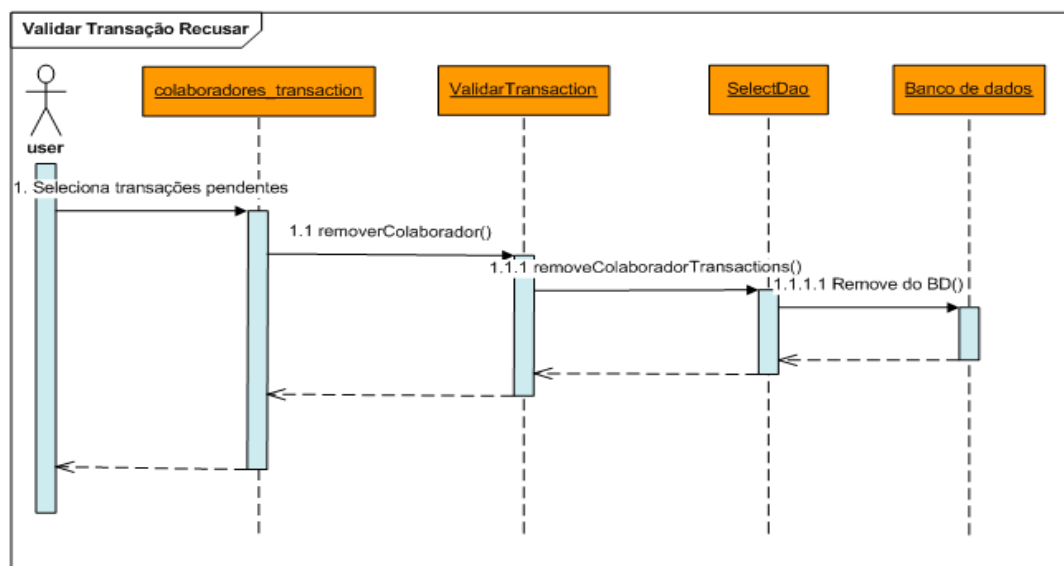


Figura 25 - Diagrama de seqüência 'valida transação recusar'

Após autenticação no sistema o usuário poderá procurar um colaborador cadastrado na base de dados.

4.5 Diagramas de classes

Nesta seção serão apresentados os principais classes e pacotes deste sistema, cada classe dentro de seu respectivo pacote.

A figura 26 representa os pacotes Model, Controller, Dao e Utils da aplicação.

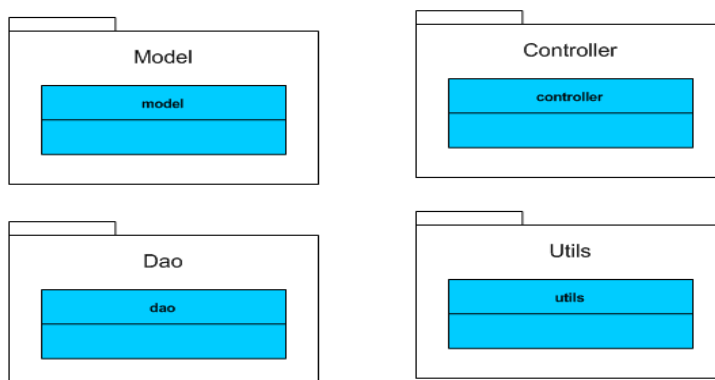


Figura 26 – Pacotes

A figura 27 representa o pacote 'controller' e suas respectivas classes.

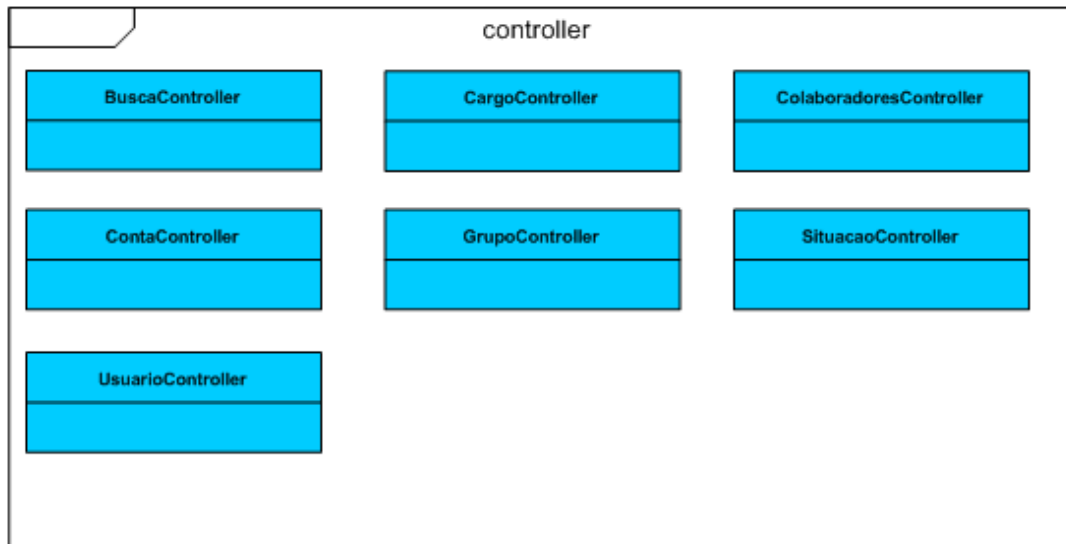


Figura 27- Pacote 'controller' e suas respectivas classes

A figura 28 representa o pacote 'model' e suas respectivas classes.

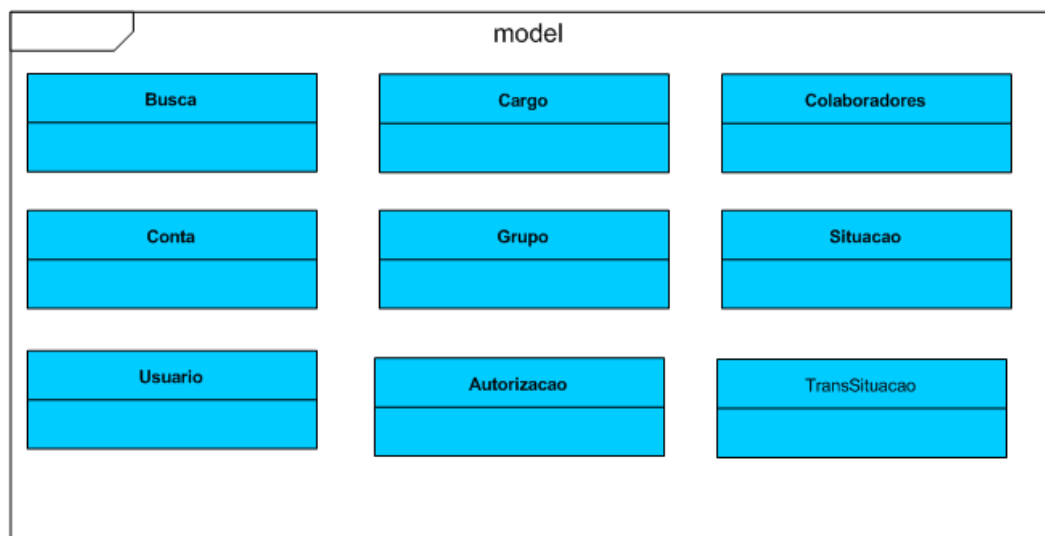


Figura 28 - Pacote 'model' e suas respectivas classes

A figura 29 representa o pacote 'dao' e suas respectivas classes.

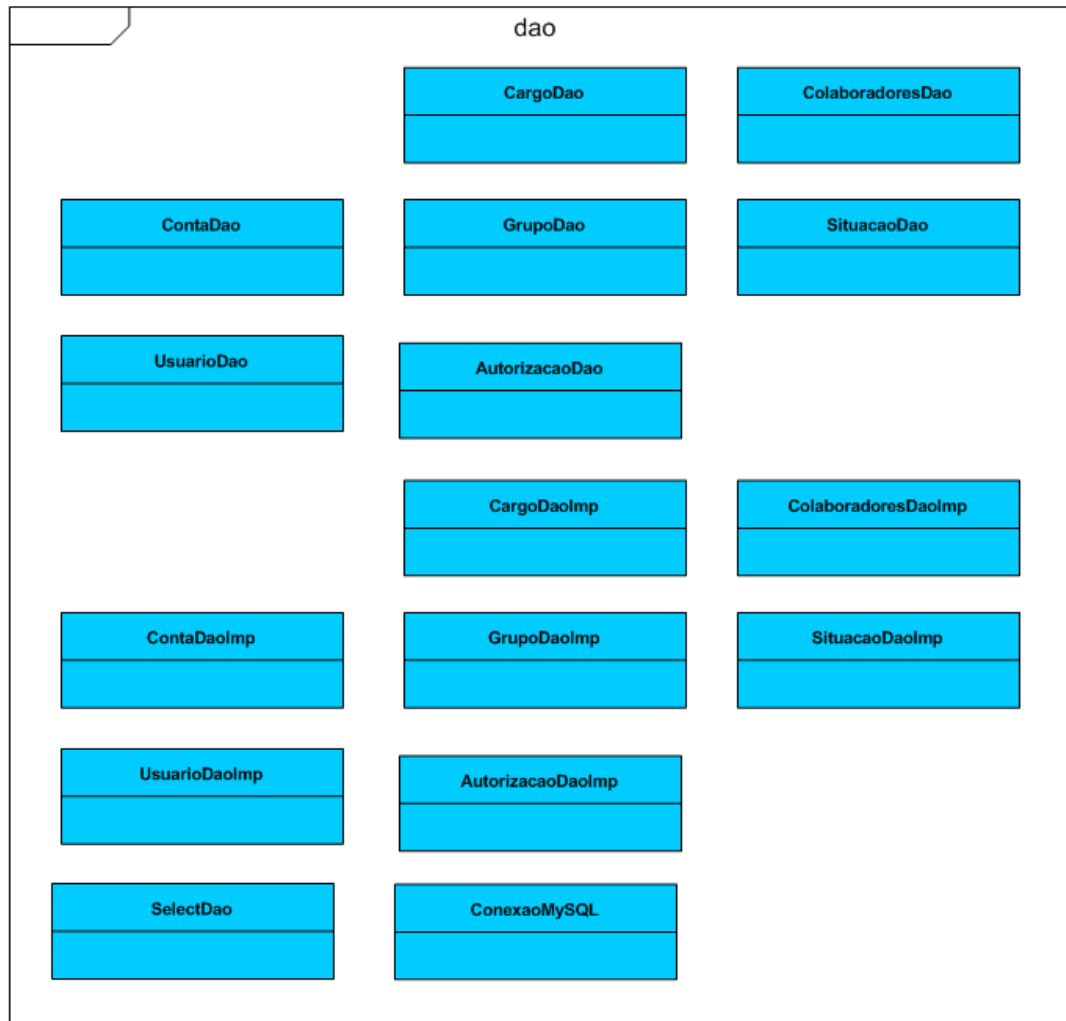


Figura 29- Pacote 'dao' e suas respectivas classes

A figura 30 apresenta o pacote 'utils' e suas respectivas classes.

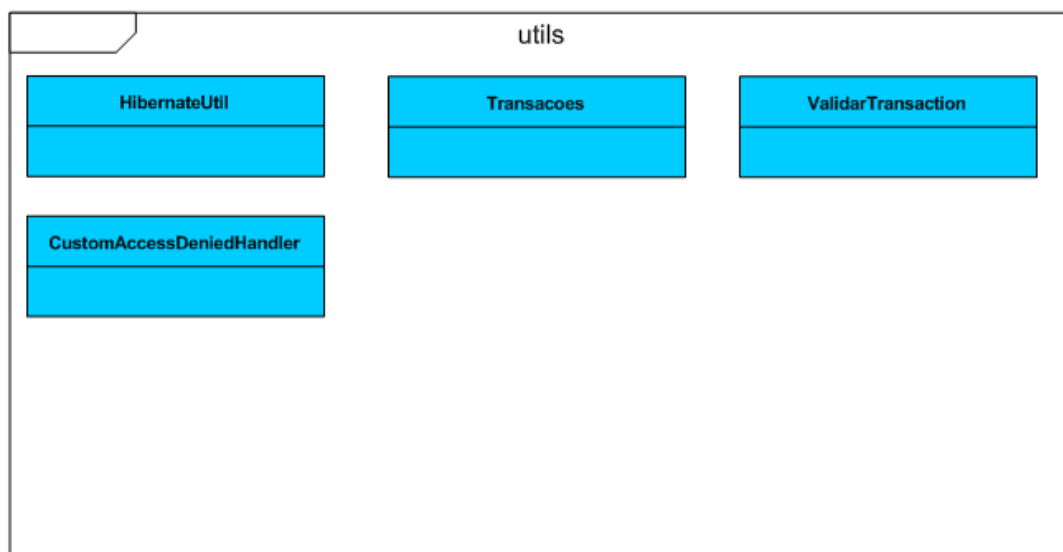


Figura 30 - Pacote 'utils' e suas respectivas classes

4.5.1 Descrição das Classes

Esta seção descreve as principais classes do sistema.

A notação <tabela> refere-se a maneira genérica a classes de mesma função na qual difere entre si por parte do nome.

Por exemplo: CargoController, ColaboradorController são classes que possuem a mesma função porém trabalham com tabelas diferentes da base de dados.

a) <tabela>Controller

A lista de classes Controller cumpre o papel de fazer a comunicação entre as páginas da aplicação (views) e as classes Dao que por sua vez fazem comunicação com o banco de dados. As classes controller são vistas como classes de controle, sendo que o seu principal papel é fazer chamadas das classes Dao e enviar as respostas para a página. As principais funções são buscar, atualizar, inserir e remover.

4.7 Interfaces

Nesta seção serão apresentadas as principais interfaces deste sistema.

A figura 32 demonstra a tela de acesso pública a aplicação.

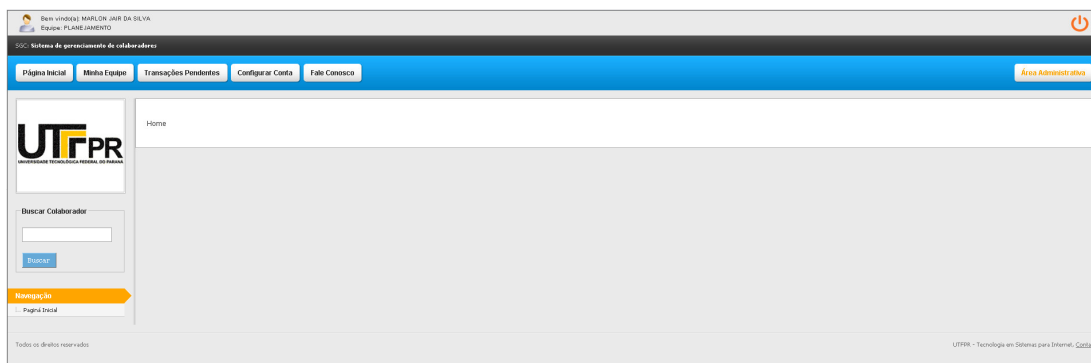


Figura 32 - SGC área pública

A figura 33 demonstra a tela de acesso administrativo a aplicação.

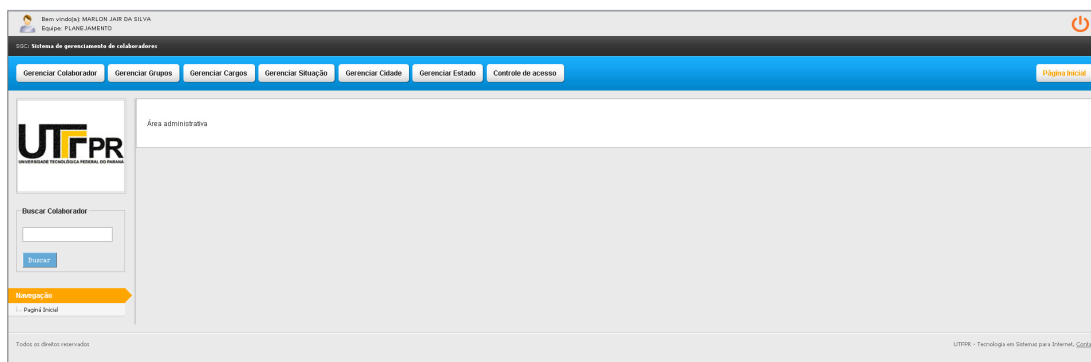


Figura 33- SGC área administrativa

A figura 34 demonstra a tela de todos os colaboradores da aplicação.

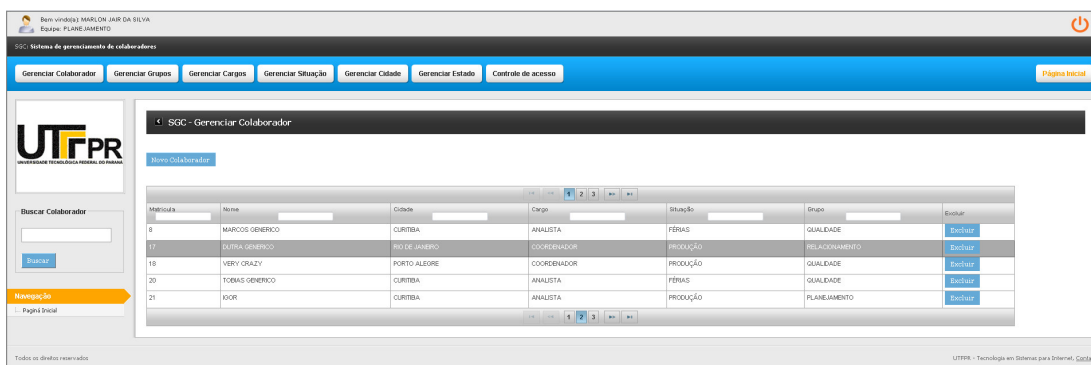


Figura 34 - SGC - lista de colaboradores

4.8 Workflow

Nesta seção serão apresentados os fluxos de trabalho de funcionamento deste sistema.

A figura 35 apresenta a hierarquia de colaboradores do SGC, é a partir dela que os fluxos de trabalhos passam a funcionar, ou seja caso a empresa possua esta hierarquia de cargos de colaboradores o sistema já poderá entrar em produção.

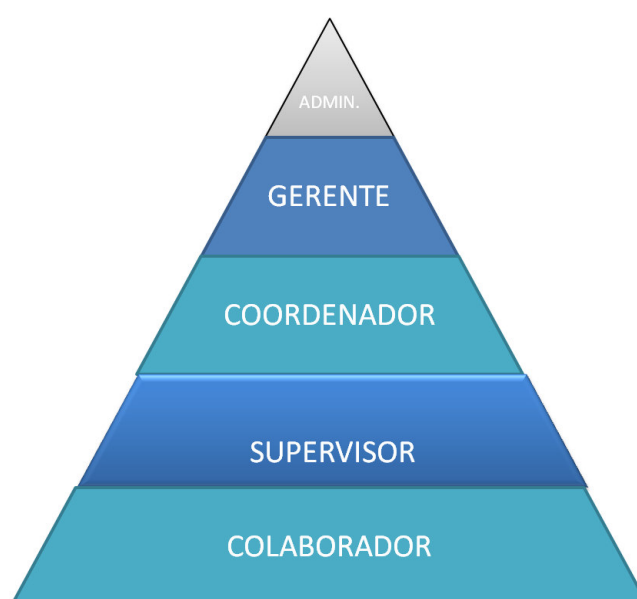


Figura 35 - Hierarquia

Para efetivar a troca de um colaborador de hierarquia e(ou) grupo de trabalho as ações deverão passar pelos fluxos descritos abaixo

A figura 36 apresenta o fluxo de transação REQUISITOR.

Ação: O requisitor (supervisor, coordenador, gerente) seleciona um colaborador de sua equipe envia este colaborador para um destino (supervisor, coordenador, gerente) e grupo de trabalho. Assim alterando o status para “Aguardando aprovação do destino”.

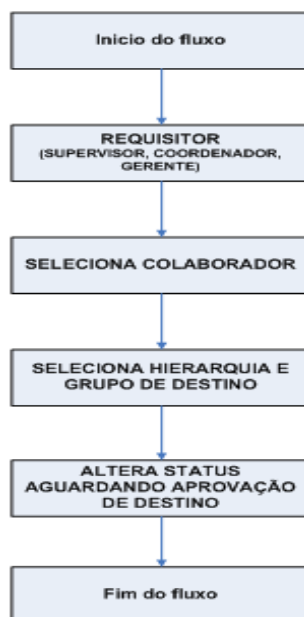


Figura 36 - Fluxo Requisitor

A figura 37 apresenta o fluxo de transação RECEPTOR.

Ação: O receptor (supervisor, coordenador, gerente) poderá aprovar ou negar uma transação onde estiver envolvido. Assim alterando o status para “Aguardando aprovação do administrador”.

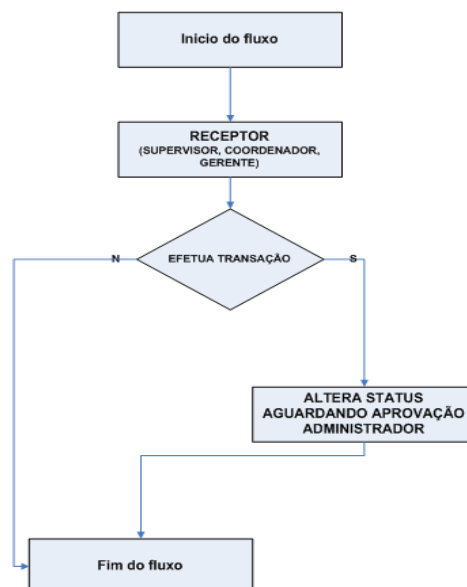


Figura 37 - Fluxo Receptor

A figura 38 apresenta o fluxo de transação ADMINISTRADOR.

Ação: O administrador poderá aprovar ou negar uma transação somente após este aceite o colaborador passara a ser da equipe de destino.

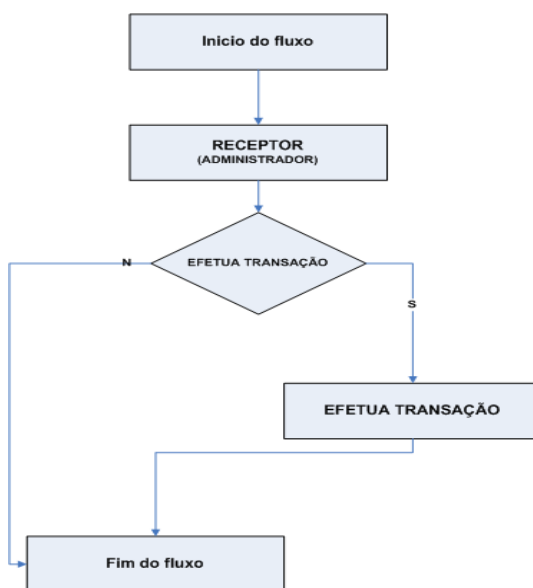


Figura 38 - Fluxo Administrador

5 DISCUSSÃO

O SGC foi construído com o intuito de facilitar o trabalho de controle de funcionários de uma organização, mantendo uma base histórica de dados e um sistema de acesso rápido as informações, conforme a necessidade de busca do usuário. Além disso o sistema oferece um sistema de Workflow para movimentações de funcionários de um setor para outro.

Este presente trabalho foi implantado na empresa GVT como primeiro cliente do projeto.

5.1 Dificuldades encontradas

A principal dificuldade encontrada no projeto foi a assertividade no cronograma, pois conforme foi feita cada reunião com o cliente final varias analises e sugestões foram surgindo e implementadas em tempo real ocasionando atrasos em entregas de algumas tarefas.

6. CONCLUSÕES

Durante todas as fases deste projeto, foram procuradas várias alternativas para o desenvolvimento do software de Gerenciamento de Colaboradores, sempre visando alcançar o melhor resultado possível. Sendo assim varias tecnologias web foram estudadas no desenvolvimento da aplicação e muitas destas utilizadas para suprir uma necessidade ou agregar tecnologia a uma parte do sistema.

Sendo assim podemos concluir que houve um grande aprendizado em tecnologia web, pois a busca de novas tecnologias foi constante. Para garantir que a aplicação funcionasse perfeitamente, foram disparados testes unitários em cada tela para assim evitar qualquer tipo de falha. Também foi feito o estudo de comportamento de cada framework, para analisar melhor desempenho e produtividade possível deste sistema, após isso foram escolhidos as tecnologias que mais se adaptaram ao resultado final esperado e assim deram se inicio ao desenvolvimento do SGC, não

podemos esquecer que parte para base teórica deste projeto foram utilizados algumas literaturas de fluxos de trabalho, o que nos levou a melhores resultados e soluções possíveis durante todo o desenvolvimento da ferramenta.

O diferencial desta ferramenta em comparação a outras existentes no mercado é a troca entre funcionários por Workflow, ou seja, cada tipo de gestor tem um papel fundamental dentro do sistema, o seu fluxo de trabalho entrará em ação para efetuar uma transação podendo dar início, ou caso haja necessidade sua aprovação para dar continuidade a transação já existente para que esta ação seja totalmente efetivada. O ganho efetivo para as organizações ao aderir o SGC é imenso, um exemplo foi a empresa GVT onde necessitavam desta ferramenta para controle de relatórios, histórico do percurso de cada colaborador dentro da empresa, sistema onde varias pessoas pudessem atualizar ao mesmo tempo, velocidade, confiabilidade, integridade das informações.

6.1 Contribuições

Este trabalho contribui com as organizações que necessitem de um programa de gerenciamento de funcionários e principalmente por ser um sistema que facilite as ações de registro e busca de informações para o acompanhamento do quadro de funcionários.

O sistema proposto em conjunto com a base de dados construída neste projeto contribui para agilizar um ambiente corporativo interno de uma organização além de guardar um histórico de todas as ações ocorridas.

Desta forma o sistema é útil para áreas de RH, CRM em que transações em funcionários sejam constantes.

6.2 Trabalhos Futuros

Este trabalho abre algumas possibilidades de continuação como citado a seguir:

- rever a análise de requisitos para adicionar funcionalidades;
- aplicar melhorias ao design da aplicação;

- adicionar um sistema de pesquisa de colaboradores, com opções de pesquisa avançada;
- aprimorar a segurança das informações contidas no sistema, utilizando criptografia de a, por exemplo;

Algumas destas melhorias já foram implantadas dentro do cliente GVT além de outras adaptações conforme solicitação do cliente, por exemplo, modulo de FeedBack de um colaboradores, e modulo de TurnOver que é área de desligamento daquele colaborador conforme descritos entre as figuras 39 a 42.

A figura 39 apresenta a tela de feedback de um colaborador. Nesta tela o gestor poderá aplicar uma advertência ou uma suspensão a um de seus colaboradores.

The screenshot shows the GVT system interface for a user named MARLON JAIR DA SILVA. The interface is divided into several sections:

- Header:** "Bem vindo(a): MARLON JAIR DA SILVA", "Equipe: MIS", "Ações Disciplinares", and "MARLON JAIR DA SILVA".
- Left Sidebar:** Contains a "Home" button, the GVT logo, a "Buscar Colaborador" section with a "Buscar avançada" checkbox and a "Buscar" button, and a "Navegação" menu with options for "Colaboradores", "TurnOver", and "Histórico".
- Main Content Area:** Features a table titled "Ações disciplinares:" with columns "Tipo", "Motivo", and "Total". The table lists two actions: "SUSPENSÃO" with "DISTRATO CLIENTE" (Total: 1) and "ADV VERBAL" with "ABSENTEISMO" (Total: 1). Below the table is a calendar for "Julho 2012" with columns for days of the week (Domingo to Sabado) and rows for dates (1 to 31). A blue bar labeled "ABSENTEISMO" is visible on the calendar grid, spanning from the 9th to the 11th.
- Right Sidebar:** Includes a "Relatório TO" button and a section labeled "OPÇÕES" with various icons.

Figura 39- Menu Feedback

A figura 40 apresenta a tela de desligamento de um colaborador. Nesta tela o gestor poderá desligar um de seus funcionários.

Bem vindo(a): MARLON JAIR DA SILVA
Equipe: MIS

Desligar Colaborador

Colaborador: MARLON JAIR DA SILVA Matrícula: 15601

Grupo: MIS SubGrupo:

Gerencia: ALEXANDRE PIRES DA SILVA Coordenador: MICHAELLE HANNINGER MEGA

Supervisor: Cargo: ANALISTA

Preencha o formulário abaixo

Voluntário [] Involuntário []

Motivo:

Outros Motivos:

Desempenho / Comportamento:

Outros Desempenho / Comportamento:

Data desligamento CRM:

Observação:

Figura 40 - Menu Desligamento

A figura 41 apresenta a tela de relatórios de desligamentos diários feitos pelo sistema onde o usuário poderá gerar um relatório dinâmico de todos os desligamentos feitos.

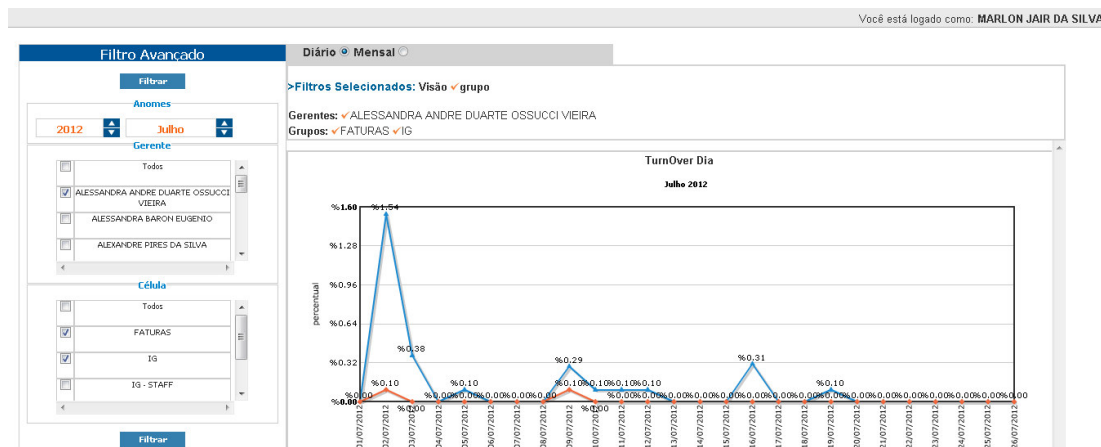


Figura 41 - Menu Relatórios

A figura 42 apresenta a tela de relatórios de desligamentos mensais feitos pelo sistema onde o usuário poderá gerar um relatório dinâmico de todos os desligamentos feitos sumarizados por mês.

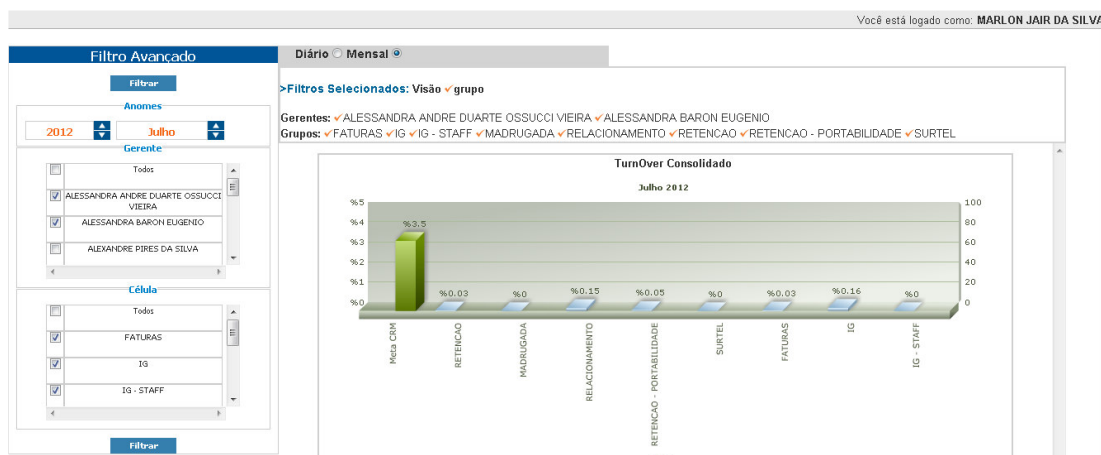


Figura 42 - Menu Relatórios detalhados

Este projeto pode também servir como base para outros trabalhos do mesmo gênero, tanto para estudantes da área de informática.

7 REFERÊNCIAS

LUCKOW, D. H.; MELO, A. A. **Java Para a WEB**. 1a. edição, 2010.

DEITEL, H. M.; DEITEL, P. J. **Java como programar**; tradução e revisão técnica Carlos Arthur Lang Lisbôa. 4a. edição, 2003.

Apache.org. **Apache tomcat**. Disponível em <<http://tomcat.apache.org/>>. Acesso em 1 de novembro de 2011.

Deitel, Harvey m. **Java como programar**. Editora pearson prentice hall, São Paulo, Brasil, 2005.

Gomes, Yuri Marx p. **Java na web com jsf**. Editora ciência moderna, Rio de Janeiro, Brasil, 2008.

Bayern, Shawn. **Jstl in action**. Editora oreilly & assoc, 2002

Sauvé, jacques. **Jstl - jsp standard tag library** disponível em <<http://www.dsc.ufcg.edu.br/~jacques/cursos/daca/html/jstl/jstl.htm>>. Acesso em 24 de Agosto de 2011.

Temple, André. Jsp, **servlets e j2ee** disponível em [Http://www.inf.ufsc.br/~bosco/downloads/livro-jsp-servlets-j2ee.pdf](http://www.inf.ufsc.br/~bosco/downloads/livro-jsp-servlets-j2ee.pdf)>. Acesso em 25 de Agosto de 2011.

Matena, Vlada. **Java transaction api(jta)**. San antonio rd palo alto, ca 94303,usa, 1999.

Cheung, susan. Sun microsystems inc. **Java transaction api (jta)** Disponível em<http://download.oracle.com/auth/otn-pub/jcp/7286-jta-1.0.1-spec-oth-jspec/jtaspec1_0_1.pdf?e=1314577131&h=fa81c50cac4a322bdc374ac6600de5de>. Acesso em 25 de Agosto de 2011.

Apache.org. **Apache maven project**. Disponível em <<http://maven.apache.org/what-is-maven.html>>. Acesso em 1 de Novembro de 2011.

Tigris.org. **Tortoise subversion**. Disponível em <<http://tortoisesvn.tigris.org>>. Acesso em 1 de novembro de 2011.

W3c. Cascading **style sheets**. Disponível em <<http://www.w3.org/style/css/>>. Acesso em 7 de dezembro de 2011.

Gratuita do banco de dados Oracle – **Oracle express edition 10g**. Disponível em <[http://www.linhadecodigo.com.br/artigo/1196/oracle-freewareconhe%a7a-a-nova-edi%a7%a3o-gratuita-do-banco-](http://www.linhadecodigo.com.br/artigo/1196/oracle-freewareconhe%a7a-a-nova-edi%a7%a3o-gratuita-do-banco)

de-dados-oracle-%e2%80%93-oracle-express-edition-10g.aspx>. Acesso em 7 de setembro de 2010.

Oracle.com. **Java server faces technology**. Disponível em <<http://www.oracle.com/technetwork/java/javaee/javaserverfaces-139869.html>>. Acesso em 02 de fevereiro de 2012.

Oracle.com. **Java server pages technology**. Disponível em <<http://www.oracle.com/technetwork/java/javaee/jsp/index.html>>. Acesso em 02 de fevereiro de 2012.

Javascript.com. **Javascript.com the definitive javascript resource**. Disponível em <<http://www.javascript.com/>>. Acesso em 7 de setembro de 2010.

Linhares. M. **Introdução ao hibernate**. Disponível em <http://www.guj.com.br/content/articles/hibernate/intruducacao_hibernate3_guj.pdf>. Acesso em 7 de março de 2012

Linha de código, oracle freeware – **Oracle.com. Java technology**. Disponível em <<http://www.oracle.com/us/technologies/java/index.html>>. Acesso em 2 de abril de 2012.

The wide workflow modeling language: Disponível em <<http://dis.sema.es/projects/wide/documents/4080-2.pdf>>. Acesso em: 12 de Dezembro, 2011.

Workflow management coalition. **The workflow reference model**. Disponível em <<Http://www.wfmc.org>> Acesso em 05 de Maio, 2012.

Workflow **Definição de Processos** Disponível Em <Http://www.arquivar.com.br/espaco_profissional/sala_leitura/artigos/definicao_de_processos_de_workflow.pdf> Acesso em 11 de Junho de 2012.

Cruz, Tadeu **Workflow || A Tecnologia que Revolucionou processos** 1° edição , 2004.

Cruz, Tadeu **Uso e desuso de sistemas de Workflow** 1° edição , 2006.

Aalst,van der, **Wil Gestão de Workflows** 1° edição , 2004.

Primefaces, **Componentes de primefaces**: disponível em <<http://primefaces.org/>> Acesso em 5 de Abril de 2012

Spring Security, **Guide to Spring Security** Disponível em

< <http://www.mularien.com/blog/2008/07/07/5-minute-guide-to-spring-security/>> Acesso em 2 de Março de 2012.

JavaScript, **Script Brasil** disponível em < <http://www.scriptbrasil.com.br/>> acesso em 3 de Agosto de 2011.

JPA, **Java persistence api** disponível em < <http://www.oracle.com/technetwork/articles/javaee/jpa-137156.html>> acesso em 22 de Agosto de 2011

Hibernate, **Java Hibernate**, disponível em < <http://www.hibernate.org/>> acesso em 20 de Agosto de 2011.

Autorização

Autorizo a reprodução e/ou divulgação total ou parcial da presente obra, por qualquer meio convencional ou eletrônico, desde que citada a fonte.

Nome do autor: Marlon Jair da Silva

Assinatura do autor: _____

Instituição: Universidade Tecnológica Federal do Paraná

Local: Curitiba, Paraná

E-mail: marlonspinf@gmail.com

