

UNIVERSIDADE REGIONAL DE BLUMENAU
CENTRO DE CIÊNCIAS EXATAS E NATURAIS
CURSO DE CIÊNCIAS DA COMPUTAÇÃO – BACHARELADO

**SISTEMA DE ADVOCACIA COM AUTOMATIZAÇÃO DO
ANDAMENTO DOS PROCESSOS JURÍDICOS RECEBIDOS
DO SISTEMA PUSH ATRAVÉS DE E-MAIL**

MARCUS VINÍCIUS FELIPPI

BLUMENAU
2005

2005/2-17

MARCUS VINÍCIUS FELIPPI

**SISTEMA DE ADVOCACIA COM AUTOMATIZAÇÃO DO
ANDAMENTO DOS PROCESSOS JURÍDICOS RECEBIDOS
DO SISTEMA PUSH ATRAVÉS DE E-MAIL**

Trabalho de Conclusão de Curso submetido à
Universidade Regional de Blumenau para a
obtenção dos créditos na disciplina Trabalho
de Conclusão de Curso II do curso de Ciências
da Computação — Bacharelado.

Prof. Paulo Roberto Dias, Titulação - Orientador

**BLUMENAU
2005**

2005/2-17

**SISTEMA DE ADVOCACIA COM AUTOMATIZAÇÃO DO
ANDAMENTO DOS PROCESSOS JURÍDICOS RECEBIDOS
DO SISTEMA PUSH ATRAVÉS DE E-MAIL**

Por

MARCUS VINÍCIUS FELIPPI

Trabalho aprovado para obtenção dos créditos
na disciplina de Trabalho de Conclusão de
Curso II, pela banca examinadora formada
por:

Presidente:

Prof. Paulo Roberto Dias – Orientador, FURB

Membro:

Prof. Ricardo de Alencar Azambuja – FURB

Membro:

Prof. Wilson Pedro Carli – FURB

Blumenau, 16 de novembro de 2005

Dedico este trabalho a todos os meus amigos que deram força na realização deste, e aos meus colegas que compartilharam longas horas de companhia em aula, especialmente André Bohn, com o qual participei de muitos trabalhos em grupo.

AGRADECIMENTOS

Ao Prof. Paulo Roberto Dias, pela orientação e apoio na realização deste.

À minha mãe, que sempre acreditou na minha capacidade.

Aos meus amigos que me deram auxílio quando precisei e que acreditaram no meu potencial.

Ao meu amigo Jayson C. M. Rosa, advogado, que me auxiliou nas dúvidas de desenvolvimento deste trabalho.

“Quanto mais eu sei, mais eu sei que nada sei.”

Sócrates

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema de gestão para advocacia, tendo como foco principal a automatização das movimentações processuais recebidas do Sistema *Push* do TJSC através de *e-mail*. A partir destes *e-mails* são extraídas as movimentações processuais, sendo gravadas na base de dados do sistema, mantendo atualizadas as movimentações dos processos cadastrados. O sistema foi desenvolvido em *Java*, utilizando o banco de dados *PostgreSQL*.

Palavras-chave: Sistema de advocacia. Sistema *Push*. *Javamail*. Banco de dados *PostgreSQL*.

ABSTRACT

This paper presents the development of a management system for a law firm, mainly focusing the automation of the lawsuit history from the Push System of the TJSC through e-mail. From these e-mails, the lawsuit history is retrieved and recorded in a database, in order to keep it up-to-date. The system has been developed using Java and a PostgreSQL database.

Key-words: Law firm system. Push System. Javamail. PostgreSQL database.

LISTA DE QUADROS

Quadro 1 – Requisitos funcionais.....	26
Quadro 2 – Requisitos não funcionais.....	27
Quadro 3 – Método RetornaConteudo().....	42
Quadro 4 – Método RetornaConteudoEndereco().....	43

LISTA DE ILUSTRAÇÕES

Figura 1 – Pacotes dos diagramas de casos de uso.....	28
Figura 2 – Diagrama de casos de uso de cadastros.....	29
Figura 3 – Diagrama de casos de uso de movimentações	30
Figura 4 – Diagrama de casos de uso do financeiro	31
Figura 5 – Diagrama de casos de uso de relatórios	32
Figura 6 – Diagrama de casos de uso de configurações	32
Figura 7 – Diagrama de classes da atualização das movimentações dos processos.....	33
Figura 8 – Diagrama de atividades para o caso de uso importar movimentações.....	36
Figura 9 – Modelo lógico de dados	37
Figura 10 – Modelo físico de dados	38
Figura 11 – <i>Login</i> do sistema	43
Figura 12 – Mensagem de solicitação de verificação de <i>e-mail</i>	44
Figura 13 – Importação de movimentações processuais do servidor de <i>e-mails</i>	44
Figura 14 – Menu do sistema	45
Figura 15 – Cadastro de clientes.....	45
Figura 16 – Cadastro de advogados.....	46
Figura 17 – Cadastro de partes contrárias	46
Figura 18 – Cadastro de advogados contrários.....	47
Figura 19 – Cadastro de áreas.....	47
Figura 20 – Cadastro de comarcas.....	47
Figura 21 – Cadastro de processos	48
Figura 22 – Movimentações do processo	48
Figura 23 – Cadastro de usuários	49
Figura 24 – Agendamento de audiências.....	49
Figura 25 – Inserção de movimentação manualmente	50
Figura 26 – Agenda das movimentações.....	50
Figura 27 – Agenda de audiências.....	51
Figura 28 – Despesas de processos	51
Figura 29 – Lançamento de honorários	52
Figura 30 – Baixa de honorários	52
Figura 31 – Tela de filtros do relatório de clientes.....	52

Figura 32 – Relatório de clientes visualizado.....	53
Figura 33 – Tela de filtros do relatório de advogados.....	53
Figura 34 – Relatório de advogados visualizado.....	54
Figura 35 – Tela de filtros do relatório de movimentações de processos.....	54
Figura 36 – Relatório de movimentações de processos visualizado	55
Figura 37 – Tela de filtros do relatório de situação de honorários	55
Figura 38 – Relatório de situação de honorários visualizado.....	56
Figura 39 – Alteração de senha	56
Figura 40 – Alteração de <i>e-mails</i>	57
Figura 41 – Alteração de <i>tokens</i>	57
Figura 42 – Consulta de clientes.....	58

LISTA DE SIGLAS

API – Application Program Interface
ASCII - American Standard Code for Information Interchange
ASP – Active Server Pages
FTP - File Transport Protocol
IETF – Internet Engineering Task Force
IMAP - Internet Mail Access Protocol
JVM - Java Virtual Machine
MIME - Multipurpose Internet Mail Extensions
NNTP – Network News Transfer Protocol
POP - Post Office Protocol
POP3 - Post Office Protocol 3
RFC822 - Request for Comments 822
RFC977 - Request for Comments 977
SGBD - Sistema Gerenciador de Banco de Dados
SMTP - Simple Mail Transfer Protocol
STJ - Superior Tribunal de Justiça
TCP - Transport Control Protocol
TJSC – Tribunal de Justiça de Santa Catarina
UML - Unified Modeling Language

SUMÁRIO

1 INTRODUÇÃO.....	14
1.1 OBJETIVOS DO TRABALHO	15
1.2 ESTRUTURA DO TRABALHO	16
2 FUNDAMENTAÇÃO TEÓRICA	17
2.1 SISTEMA PUSH.....	17
2.2 DIREITO PROCESSUAL.....	18
2.3 PROTOCOLOS DE E-MAIL.....	19
2.3.1 SMTP	19
2.3.2 IMAP	20
2.3.3 POP3	21
2.3.4 NNTP	21
2.4 FORMATO DE MENSAGEM DE CORREIO ELETRÔNICO	22
2.5 TRABALHOS CORRELATOS	23
3 DESENVOLVIMENTO DO TRABALHO	25
3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO.....	25
3.2 ESPECIFICAÇÃO	27
3.2.1 Diagramas de casos de uso.....	27
3.2.1.1 Diagrama de casos de uso de cadastros	28
3.2.1.2 Diagrama de casos de uso de movimentações	29
3.2.1.3 Diagrama de casos de uso do financeiro.....	30
3.2.1.4 Diagrama de casos de uso de relatórios	31
3.2.1.5 Diagrama de casos de uso de configurações.....	32
3.2.2 Diagrama de classes	33
3.2.2.1 Classe ImportaMovimentacoes.....	34
3.2.2.2 Classe AtualizaMovimentacao	34
3.2.3 Diagrama de atividades	35
3.2.4 Modelo lógico de dados	37
3.2.5 Modelo físico de dados	38
3.3 IMPLEMENTAÇÃO	39
3.3.1 Técnicas e ferramentas utilizadas.....	39
3.3.1.1 API JavaMail	40

3.3.1.2 Método RetornaConteudo()	41
3.3.1.3 Método RetornaConteudoEndereco()	42
3.3.2 Operacionalidade da implementação	43
3.4 RESULTADOS E DISCUSSÃO	58
4 CONCLUSÕES	60
4.1 EXTENSÕES	61
REFERÊNCIAS BIBLIOGRÁFICAS	62
APÊNDICE A – Descrição de casos de uso	64
APÊNDICE B – Dicionário de dados	77

1 INTRODUÇÃO

Hoje em dia, nas organizações em geral, o número de informações é cada vez maior e as mudanças ocorrem em grande velocidade. Estas informações necessitam, cada vez mais, ser armazenadas e recuperadas de uma maneira rápida e eficiente.

Em um escritório de advocacia há uma grande quantidade de papéis, dificultando o trabalho do profissional da área. Muitas vezes, para encontrar uma simples informação são necessárias horas de procura, e se não tiver uma boa organização física, essa procura pode ser ainda maior. Esse trabalho pode ser otimizado com o uso de um sistema computacional que contenha grande parte destas informações.

De acordo com Baeta (1983, p. 25), em um escritório de advocacia, o computador tem especial importância, pois permite a execução automática de tarefas, tendo um papel fundamental na execução e organização das tarefas do dia-a-dia.

Hoje, existem diversos aplicativos para a área de advocacia, que facilitam a organização das informações, agilizando o trabalho e possibilitando um melhor desempenho nos escritórios, resultando em maior produtividade. No entanto, quando há movimentação no andamento de um processo, esses aplicativos não atualizam automaticamente a base de dados. Para manter essa base atualizada, é necessário acessar o *site* que pertence ao Tribunal de Justiça de Santa Catarina (TJSC) (<http://www.tj.sc.gov.br>) e pesquisar as movimentações ocorridas para o processo desejado. Em um escritório com muitos processos em andamento, será necessário acessar cada um, através do *site*, para verificar se houve alguma atualização, resultando em desperdício de tempo.

Para suprir essa necessidade, foi criado um sistema chamado *Push*, um correio eletrônico automático de andamentos processuais. Para acessar esse serviço, o advogado deve entrar no *site* do TJSC e cadastrar seu *e-mail* para determinado processo. Assim, sempre que

houver movimentação no andamento do processo, um *e-mail* será disparado para avisar ao advogado sobre tal movimentação. Esse serviço de envio de *e-mail* é sempre realizado durante a noite, devido ao excesso de tráfego na rede causado pelo Sistema *Push*. Porém, para o advogado ter a movimentação do processo atualizada em seu aplicativo, terá de fazer manualmente, copiando o texto da movimentação a partir do *e-mail* (SUPERIOR TRIBUNAL DE JUSTIÇA, 2004).

Considerando o problema apresentado neste trabalho, foi desenvolvido um sistema para gestão de advocacia que captura informações de *e-mails* originados do Sistema *Push*. Para cada *e-mail* é capturada a movimentação do processo, a data que ocorreu a movimentação e o *link* da página onde se encontram as movimentações do processo. Essas informações capturadas do *e-mail* são gravadas na base de dados do sistema automaticamente. Esse sistema contém uma agenda indicando quais os processos tiveram movimentações a partir de uma data informada.

1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é fornecer um sistema de gestão para escritórios de advocacia, com a atualização automática do andamento dos processos que são enviados pelo Sistema *Push* através de *e-mail*.

Os objetivos específicos do trabalho são:

- a) capturar *e-mails* provenientes do Sistema *Push* com as movimentações dos processos e atualizar a base de dados com essas movimentações;
- b) manter agenda atualizada com as movimentações de cada processo;
- c) manter cadastros de advogados, clientes e processos para o funcionamento do sistema;

- d) gerar relatório com as movimentações dos processos.

1.2 ESTRUTURA DO TRABALHO

O Capítulo 1 apresenta a introdução do trabalho.

No capítulo 2 são realizadas revisões sobre a fundamentação teórica referente ao trabalho.

No capítulo 3 é apresentada a especificação do sistema, seguida da implementação do mesmo, mostrando sua operacionalidade.

Por último, no capítulo 4, é apresentada a conclusão do trabalho.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados alguns conceitos referentes a este trabalho. É apresentada uma breve descrição do que é o Sistema *Push* e sobre direito processual. Após são abordados alguns dos mais importantes protocolos de *e-mail*, seguido de informações sobre o formato de mensagens de *e-mail*. Por último, tem-se uma seção sobre trabalhos correlatos.

2.1 SISTEMA PUSH

De acordo com o Superior Tribunal de Justiça (STJ) (SUPERIOR TRIBUNAL DE JUSTIÇA, 2004), o Sistema *Push* é um sistema de envio automático de informações por *e-mail* aos usuários cadastrados. Toda vez que há alguma movimentação de um dado processo, o advogado interessado que se cadastrar na página do referido tribunal, recebe automaticamente um *e-mail* informando a movimentação ocorrida.

O Sistema *Push* tem as seguintes funções:

- a) acompanhamento processual: envia ao usuário cadastrado um *e-mail* comunicando os andamentos do dia quando houver movimentação de algum dos processos incluídos na lista de processos cadastrados pelo usuário;
- b) notícias do STJ: as notícias publicadas no *site* do STJ são enviadas por *e-mail* aos usuários cadastrados;
- c) informativos de jurisprudência do STJ: os informativos de jurisprudência do STJ são enviados por *e-mail* aos usuários cadastrados.

2.2 DIREITO PROCESSUAL

De acordo com Moreira (1994, p. 3), o exame sistemático de qualquer legislação revela a influência de princípios, refletindo a atuação de ideologias, de orientações filosóficas, políticas, econômicas, sociais e até religiosas. Em um ambiente como o brasileiro, a junção de tendências diversas impede que a opção por um princípio assumira caráter absoluto. Apenas em poucos casos poderá ser apontado qual o princípio é dominante no processo civil brasileiro, prevalecendo uma combinação variável e dando margens a interpretações divergentes.

Conforme Moreira (1994, p. 7), o direito brasileiro distingue três espécies de processos, conforme a natureza e o fim da atividade realizada em juízo:

- a) processo de conhecimento: destinado à prolação de sentença em que o órgão judicial aceitará ou rejeitará o pedido para que se reconheça ou crie um determinado efeito jurídico a favor do autor;
- b) processo de execução: visa a realização concreta daquilo que se contenha na sentença condenatória ou em algum título extrajudicial a ela equiparado por lei. Trata-se de praticar atos concretos, com o objetivo de ajustar a situação de acordo com que ela deve ser, nos termos da sentença ou de acordo com outro título a que a lei atribui eficácia semelhante;
- c) processo cautelar: por meio do qual se tomam providências provisórias, ordenadas a assegurar a eficácia prática das medidas próprias de qualquer das duas outras espécies.

Conforme Marques (2004), nos processos complexos as audiências costumam demorar vários dias ou várias semanas, de acordo com o que as partes querem provar. A tramitação de processos em que não há acordo é muito demorada, chegando a alguns anos. A não existência praticamente de limites às partes na produção das provas prejudica inclusive os juízes, pois

eles, à medida que as provas vão se desenvolvendo, têm de fazer anotações, para quando as partes terminarem os debates orais, sentenciarem de imediato.

2.3 PROTOCOLOS DE E-MAIL

Existem vários protocolos de *e-mail*, utilizados nas camadas de aplicação, que estão no correio eletrônico. Entre os principais estão *Simple Mail Transfer Protocol* (SMTP), *Internet Mail Access Protocol* (IMAP), *Post Office Protocol* (POP) e *Network News Transfer Protocol* (NNTP), que são abordados nas próximas seções.

2.3.1 SMTP

De acordo com Lima (2004, p. 12), SMTP é um protocolo *American Standard Code for Information Interchange* (ASCII) simples. Depois de estabelecida a conexão através do *Transport Control Protocol* (TCP) na porta 25, a máquina que envia a mensagem fica executando como cliente, esperando conexão da máquina que está operando como servidor. O servidor envia sua identidade e informa se está preparado para receber o *e-mail*. Caso não esteja, o cliente libera a conexão e tenta novamente depois. Se o servidor estiver pronto, o cliente informa o destinatário e o remetente. Se o destinatário for válido, o servidor dá ao cliente o cabeçalho da mensagem, então, o cliente envia a mensagem e o servidor termina a transmissão.

Conforme Lima (2004, p. 13), o padrão SMTP utiliza o conceito de *spooling*, utilizando uma fila para organizar qual mensagem deve ser entregue e permitindo que o *e-mail* seja enviado de uma aplicação local para uma aplicação SMTP, com armazenamento do *e-mail* em algum dispositivo ou memória. O servidor checka se alguma mensagem está

disponível para entrega e, se estiver, atende a entrega. Caso o usuário não esteja disponível para entrega, o servidor tenta entregar depois. Eventualmente, se o *e-mail* não pôde ser entregue, ele poderá ser descartado e retornado para a origem. Isso é conhecido como *end-to-end delivery system*, porque o servidor é que contata o servidor de entrega para fazê-la, e mantém o *e-mail* por um determinado período de tempo até que tenha sido entregue.

2.3.2 IMAP

Lima (2004, p. 13), define IMAP como um método para acessar *e-mails* que são mantidos em um servidor de *e-mail* compartilhado, permitindo que um programa de *e-mail* cliente acesse remotamente as mensagens como se elas estivessem localmente.

Conforme Lima (2004, p. 14), os principais objetivos do IMAP são:

- a) ser totalmente compatível com o padrão de mensagens na *internet*, e o *Multipurpose Internet Mail Extensions* (MIME);
- b) permitir o acesso às mensagens e gerenciamento em mais de um servidor;
- c) permitir acesso sem menos eficiência que outros protocolos de acesso a arquivos;
- d) dar suporte aos modos de acesso *on-line*, *off-line* e desconcertado;
- e) suportar acesso concorrente para caixas de *e-mail* compartilhadas;
- f) o *software* cliente não precisa conhecer o formato do arquivo armazenado.

O protocolo inclui operações para:

- a) criação, exclusão e renomeação para *mailboxes*;
- b) checagem de mensagens novas;
- c) remoção permanente de mensagens;
- d) alteração e limpeza de *flags*;
- e) servidor baseado na especificação *Request for Comments 822* (RFC822) e nos tipos

MIME;

- f) pesquisa e seleção dos atributos da mensagem e textos com eficiência.

2.3.3 POP3

De acordo com Lima (2004, p. 14), o POP é largamente utilizado, trabalhando melhor quando tem somente um computador servidor, desde que seja desenvolvido para dar suporte ao acesso de mensagens *off-line*, ou seja, as mensagens são copiadas e então excluídas do servidor. Esse método de acesso não é compatível com acesso de múltiplos computadores, ao menos, que todas as máquinas compartilhem um sistema de arquivos comum. No momento, o protocolo está na versão 3.0, conhecido como *Post Office Protocol 3* (POP3).

Conforme Kurose e Ross (2003, p. 93), o POP3 é um protocolo bastante simples, sendo sua funcionalidade limitada. O POP3 começa quando o agente usuário abre uma conexão TCP com o servidor de correio na porta 110. Com a conexão estabelecida, ele segue por 3 fases: autorização, transação e atualização. Na primeira fase o agente usuário envia um nome de usuário e uma senha para autenticar a leitura da correspondência do usuário. Na segunda fase, a de transação, é feita a recuperação das mensagens. Nesta fase também é possível marcar as mensagens que devem ser excluídas, remover marcas de apagar mensagens e obter estatísticas do correio. A terceira fase, a de atualização, ocorre após o usuário sair da aplicação, encerrando a sessão POP3. Neste momento, o servidor de correio apaga as mensagens que foram marcadas.

2.3.4 NNTP

De acordo com Lima (2004, p. 15), o protocolo de transferência de notícias NNTP está

documentado na *Request for Comments 977* (RFC977). É um protocolo em que o cliente e o servidor trocam mensagens como o SMTP, mas com facilidades de grupos de notícias e transferência de artigos entre os servidores.

2.4 FORMATO DE MENSAGEM DE CORREIO ELETRÔNICO

Conforme Comer (2001, p. 351), uma mensagem de correio eletrônico tem um formato simples. Consiste em um texto ASCII separado em duas partes por uma linha em branco. A primeira parte, chamada de cabeçalho (*header*), contém informações sobre a mensagem como o remetente, os receptores pretendidos, a data que a mensagem foi enviada e o formato do conteúdo. A segunda parte, conhecida como corpo, contém o texto da mensagem.

Embora o corpo da mensagem possa conter texto arbitrário, o cabeçalho segue uma forma padrão que o *software* de *e-mail* usa ao enviar ou receber uma mensagem. Cada linha de um cabeçalho começa com uma palavra-chave seguida por dois pontos e informações adicionais, onde que, a palavra-chave indica ao *software* de *e-mail* como interpretar o resto da linha. Algumas palavras-chaves são necessárias em cada cabeçalho de *e-mail*, outras são opcionais.

De acordo com Comer (2001, p. 353), o sistema de *e-mail* da *internet* original foi projetado para tratar somente texto. Pesquisadores que achavam o *e-mail* útil criaram esquemas para permitir que fosse usado *e-mail* para transferir dados arbitrários. Em geral, todos os esquemas codificam os dados de uma forma textual, que pode ser enviada em uma mensagem de correio. A mensagem, ao chegar ao destino, tem seu corpo extraído e convertido novamente para a forma arbitrária.

Para ajudar a coordenar os vários esquemas inventados para a codificação de dados arbitrários, o *Internet Engineering Task Force* (IETF) inventou o MIME, que não dita um

único padrão para codificação de dados arbitrários. Ao usar o MIME, um remetente de uma mensagem inclui dados no cabeçalho para especificar que a mensagem segue o formato MIME, assim como linhas adicionais no corpo para especificar o tipo dos dados e a codificação. Ele permite a um remetente dividir uma mensagem em várias partes e especificar a codificação de cada uma independente, dando muita flexibilidade na construção de mensagens eletrônicas.

2.5 TRABALHOS CORRELATOS

Foram analisados quatro trabalhos correlatos descritos a seguir.

O primeiro tem como objetivo implementar um protótipo de um *software* aplicativo para escritório de advocacia com recursos voltados à *internet* utilizando *Active Server Pages* (ASP). Possibilita o advogado enviar informações referentes ao andamento de processos para os clientes que têm acesso à *internet*, assim como os clientes podem acessar a página do advogado e ver informações referente ao andamento dos processos (RADUENZ, 2000).

O segundo trata de um *software* utilizando os protocolos POP e *File Transport Protocol* (FTP), que automatiza o processo de busca de mensagens recebidas via *e-mail* com arquivos anexados, enviando os anexos através do protocolo FTP, para as pastas destino informadas no *software*. Sua implementação foi feita no ambiente *Delphi* (REGIS, 2000).

O penúltimo tem como objetivo principal a especificação e a implementação de um protótipo de *software* para notificação de recebimento de novos *e-mails* e o fornecimento de algumas informações adicionais sobre eles, como hora de chegada, assunto e presença de anexo. O desenvolvimento foi baseado no protocolo IMAP (KNORST, 2000).

O último, denominado Agente Jurídico, é um *software* que permite aos advogados e departamentos jurídicos realizarem o acompanhamento e controle de seus processos em todas

as fases e instâncias. Com o Agente Jurídico tem-se o cadastro dos clientes e das partes envolvidas, visualização de prazos processuais, gráficos estatísticos e informações financeiras detalhadas sobre todas as despesas e honorários processuais (PONTE SOFTWARE, 2004).

3 DESENVOLVIMENTO DO TRABALHO

Neste capítulo, são descritos os principais requisitos, a especificação e a implementação do sistema. No final, são apresentados os resultados e discussão.

De acordo com os objetivos propostos para este trabalho, desenvolveu-se um sistema de gestão para advocacia, que busca no servidor de *e-mail* cadastrado no sistema, através do protocolo POP3, os *e-mails* provenientes do Sistema *Push* e a partir destes *e-mails*, faz a atualização das movimentações processuais para os processos cadastrados no sistema.

3.1 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

O quadro 1 apresenta os requisitos funcionais do sistema desenvolvido, com suas funcionalidades definidas.

Requisitos Funcionais
RF01: O sistema deverá manter cadastro de clientes.
RF02: O sistema deverá manter cadastro de advogados.
RF03: O sistema deverá manter cadastro de partes contrárias.
RF04: O sistema deverá manter cadastro de advogados contrários.
RF05: O sistema deverá manter cadastro de áreas.
RF06: O sistema deverá manter cadastro de comarcas.
RF07: O sistema deverá manter cadastro de processos.
RF08: O sistema deverá manter cadastro de usuários.
RF09: O sistema deverá permitir agendar audiências.
RF10: o sistema deverá permitir verificação dos <i>e-mails</i> originados do Sistema <i>Push</i> e atualizar a movimentação do processo correspondente.
RF11: O sistema deverá permitir informar movimentações de processos, como contingência.
RF12: O usuário poderá acessar a agenda das últimas movimentações ocorridas para os processos.
RF13: O usuário poderá acessar a agenda de audiências.
RF14: O sistema deverá permitir informar despesas dos processos.
RF15: O sistema deverá manter lançamentos de honorários.
RF16: O sistema deverá permitir fazer baixa de honorários.
RF17: O usuário poderá gerar relatório de clientes.
RF18: O usuário poderá gerar relatório de advogados.
RF19: O usuário poderá gerar relatório de movimentações de processos por clientes e advogados.
RF20: O usuário poderá gerar relatório de honorários pagos e pendentes.
RF21: O sistema deverá manter configuração de <i>e-mails</i> utilizados na importação.
RF22: O sistema deverá manter configuração dos <i>tokens</i> utilizados na importação.

Quadro 1 – Requisitos funcionais

O quadro 2 apresenta os requisitos não funcionais do sistema desenvolvido.

Requisitos Não Funcionais
RNF01: O sistema será desenvolvido com a linguagem <i>java</i> .
RNF02: O sistema deverá permitir alterar <i>e-mail</i> monitorado e <i>e-mail</i> que envia movimentações processuais pelo Sistema <i>Push</i> .
RNF03: O sistema armazenará os dados no banco de dados <i>PostGreSQL</i> .
RNF04: O sistema utilizará o protocolo de <i>e-mail</i> POP3 para ler as mensagens do servidor de <i>e-mails</i> .
RNF05: O sistema deverá ter acesso protegido por senha.

Quadro 2 – Requisitos não funcionais

3.2 ESPECIFICAÇÃO

Na seqüência são apresentados alguns dos diagramas da *Unified Modeling Language* (UML), para representar o sistema desenvolvido, como o diagrama de casos de uso, o diagrama de classes, o diagrama de atividades e o modelo físico de dados.

De acordo com Lima (2005, p. 32), a UML é uma linguagem gráfica para análise, especificação e construção de sistemas de *software*, usada para representar projetos orientados a objeto utilizando uma notação comum. É extensível e independente de processos ou linguagens de programação, sem deixar de expressar-se claramente para os seus usuários e desenvolvedores, pois utiliza uma notação padrão para todos os ambientes e empresas.

3.2.1 Diagramas de casos de uso

O diagrama de casos de uso tem como objetivo descrever um modelo funcional de alto nível do sistema em projeto, procurando identificar os usuários e representar o sistema segundo sua visão. Os casos de uso são representados em todas as fases da construção de um

sistema, no início, durante e no final. Inicialmente auxilia o analista na definição dos requisitos do sistema, decorrentes das necessidades dos usuários. Durante as fases de *design* e construção, ajudam a criar visões, permitindo entendimento dos processos (DEBONI, 2003).

A construção dos diagramas de casos de uso deste trabalho foi feita com a ferramenta *Enterprise Architect*.

A seguir serão expostos os diagramas de caso de uso do sistema implementado. Para um melhor entendimento, os diagramas são divididos de acordo com as funcionalidades em comum: cadastros, movimentações, financeiro, relatórios e configurações. Os pacotes dos diagramas de casos de uso são apresentados na figura 1.

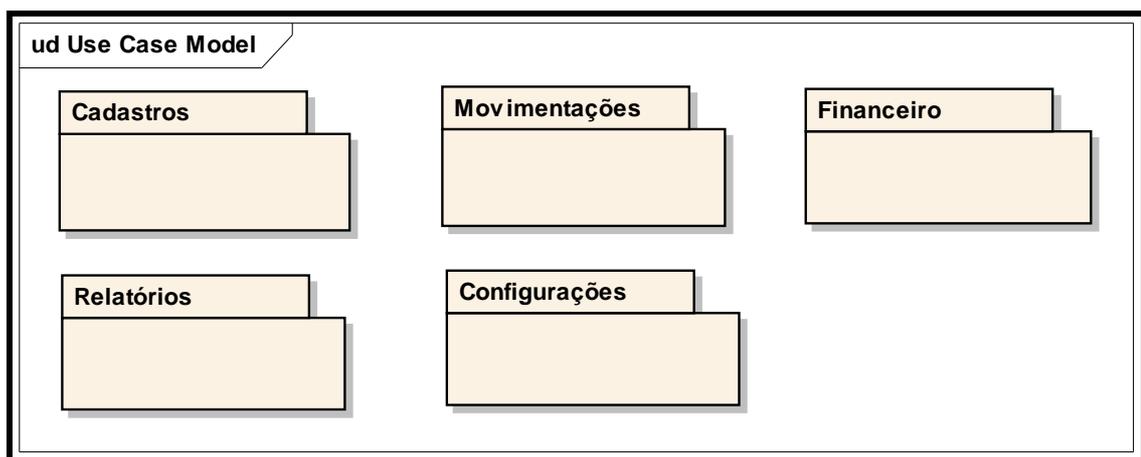


Figura 1 – Pacotes dos diagramas de casos de uso

3.2.1.1 Diagrama de casos de uso de cadastros

A figura 2 apresenta o diagrama de casos de uso de cadastros. Os casos de usos apresentados neste diagrama são ligados a funcionalidades cadastrais para a manutenção das informações no sistema.

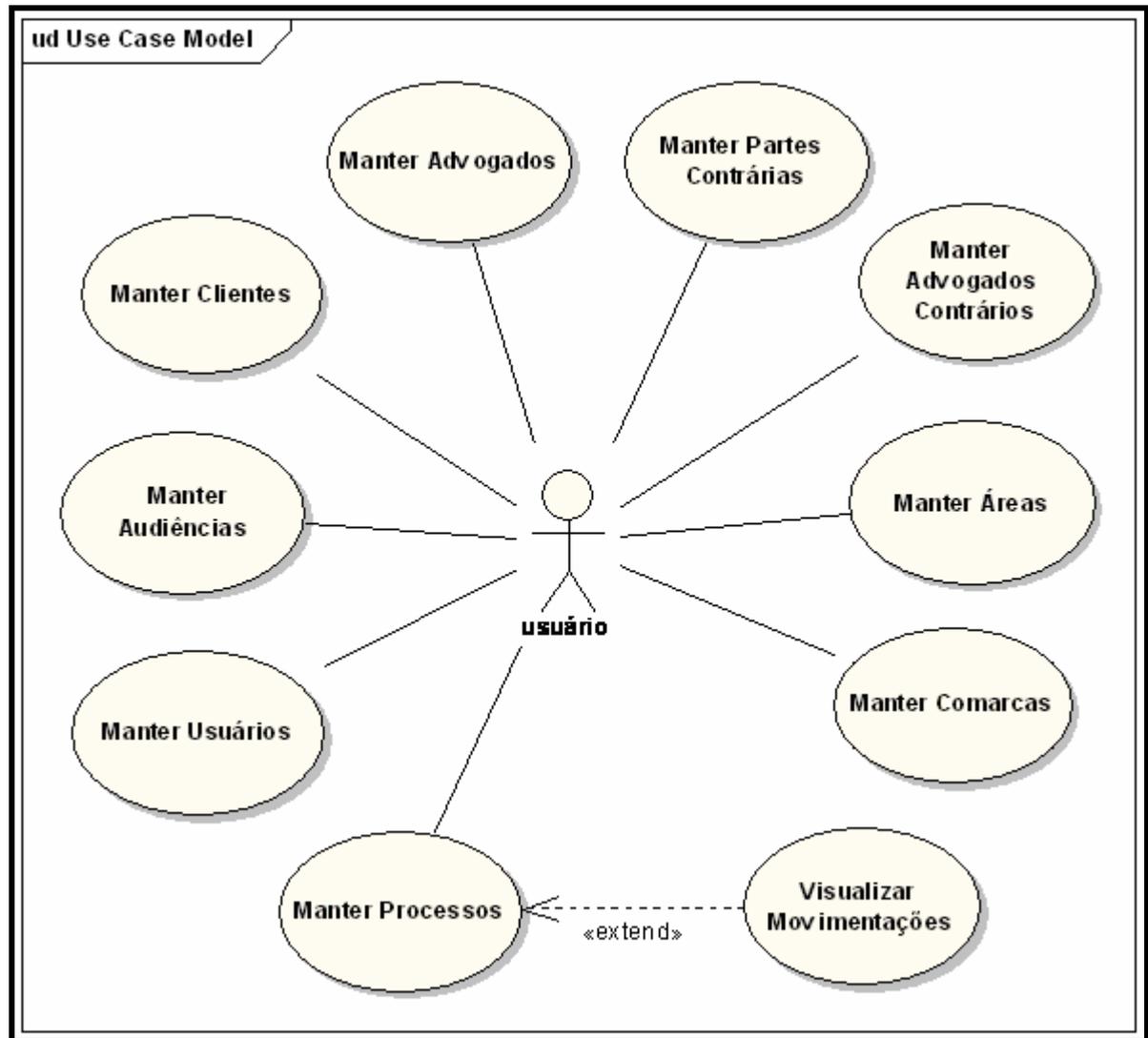


Figura 2 – Diagrama de casos de uso de cadastros

3.2.1.2 Diagrama de casos de uso de movimentações

A figura 3 apresenta o diagrama de casos de uso de movimentações. Os casos de uso apresentados neste diagrama são referentes a movimentações e consulta da agenda de audiências.

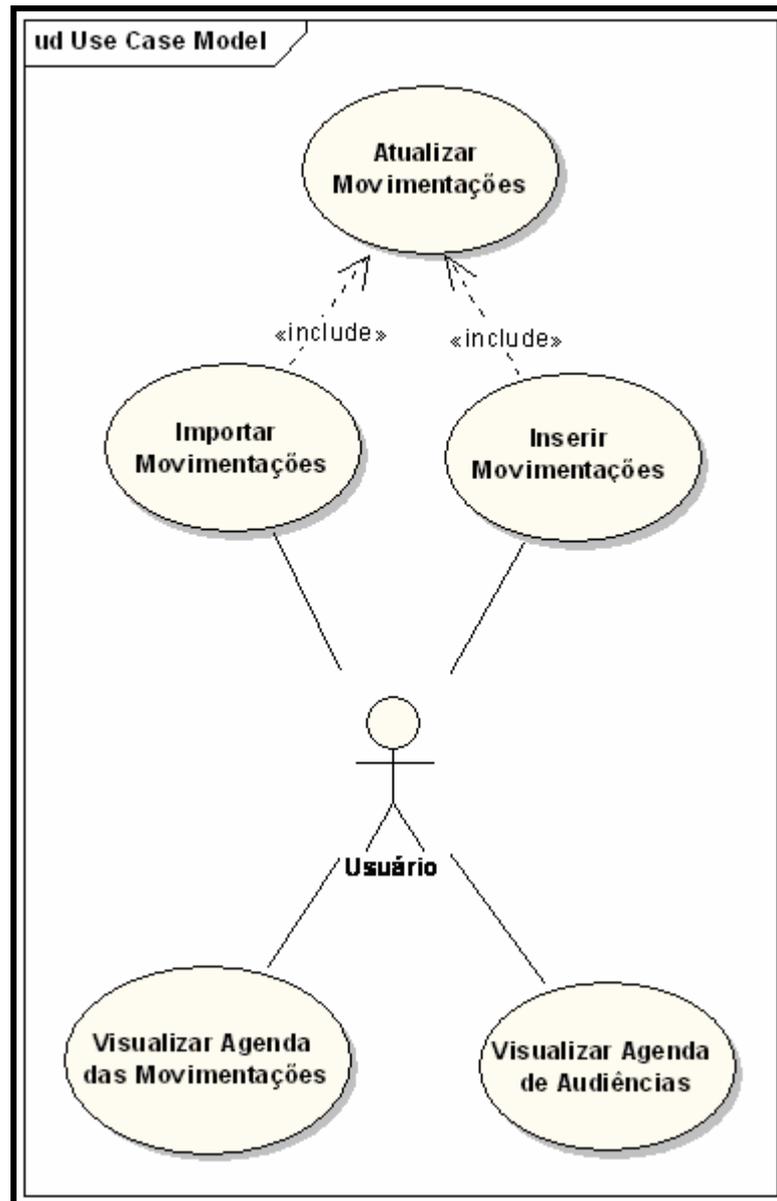


Figura 3 – Diagrama de casos de uso de movimentações

3.2.1.3 Diagrama de casos de uso do financeiro

A figura 4 apresenta o diagrama de casos de uso do financeiro. Os casos de uso apresentados neste diagrama são referentes às funcionalidades ligadas à parte financeira do sistema.

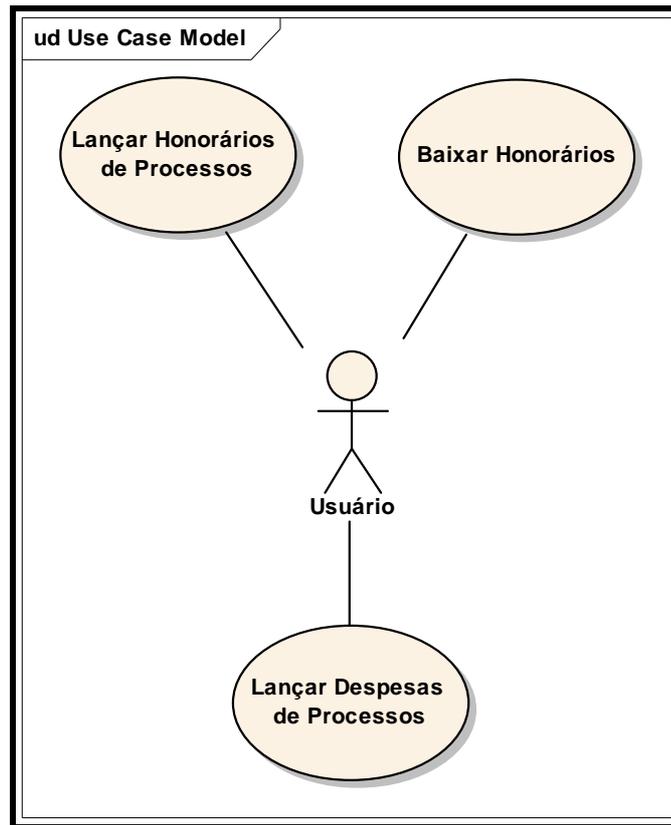


Figura 4 – Diagrama de casos de uso do financeiro

3.2.1.4 Diagrama de casos de uso de relatórios

A figura 5 apresenta o diagrama de casos de uso dos relatórios disponíveis no sistema.

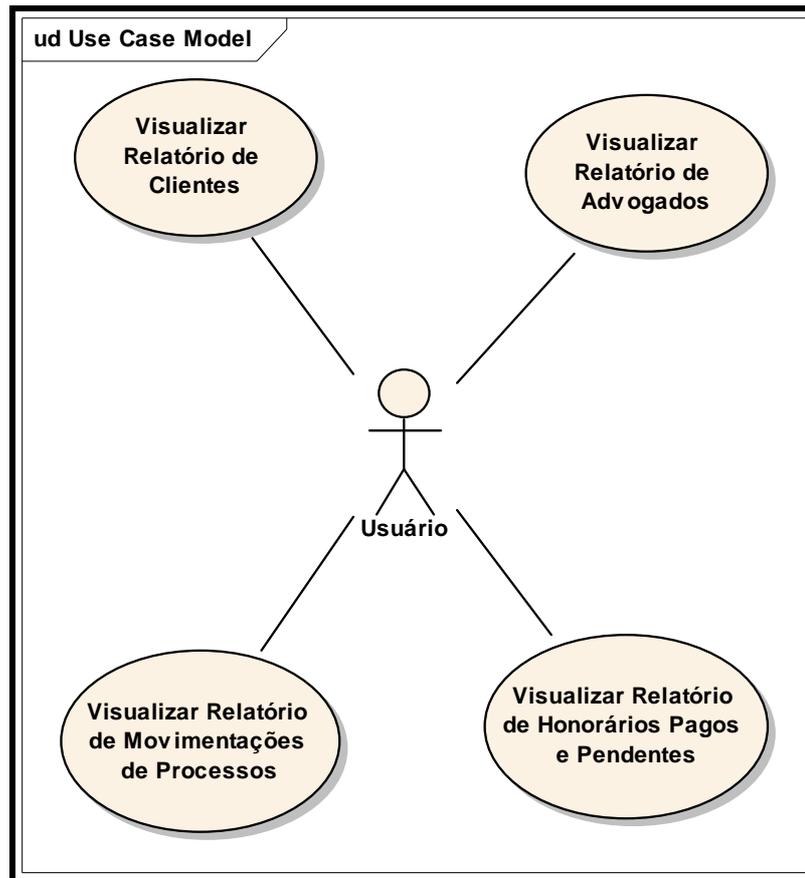


Figura 5 – Diagrama de casos de uso de relatórios

3.2.1.5 Diagrama de casos de uso de configurações

A figura 6 apresenta o diagrama de casos de uso de configurações utilizadas para a importação dos *e-mails* provenientes do Sistema *Push*.

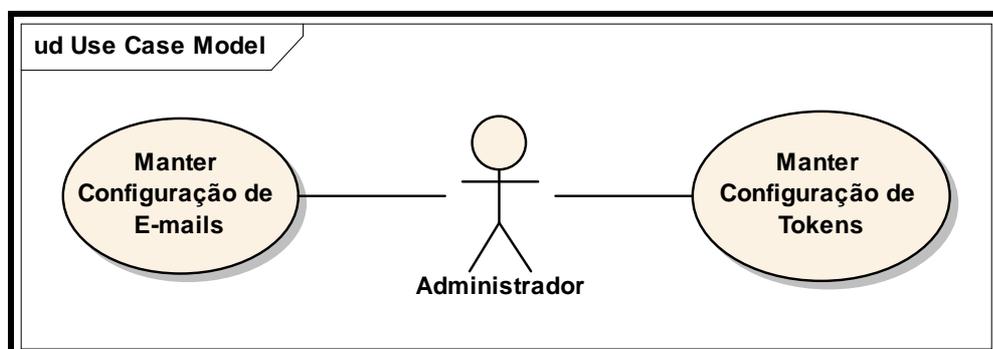


Figura 6 – Diagrama de casos de uso de configurações

3.2.2 Diagrama de classes

Deboni (2003, p. 73), define que classes são matrizes de objetos, que identificam grupos de elementos que compartilham as mesmas propriedades. Na UML, as classes são identificadas por retângulos divididos horizontalmente em três partes, classificadas em nome da classe, lista de atributos e lista de operações.

O diagrama de classes exibido na figura 7 apresenta as duas principais classes do sistema. Essas classes são responsáveis pela atualização das movimentações processuais, fazendo a verificação do *e-mail* e gravação das informações extraídas.

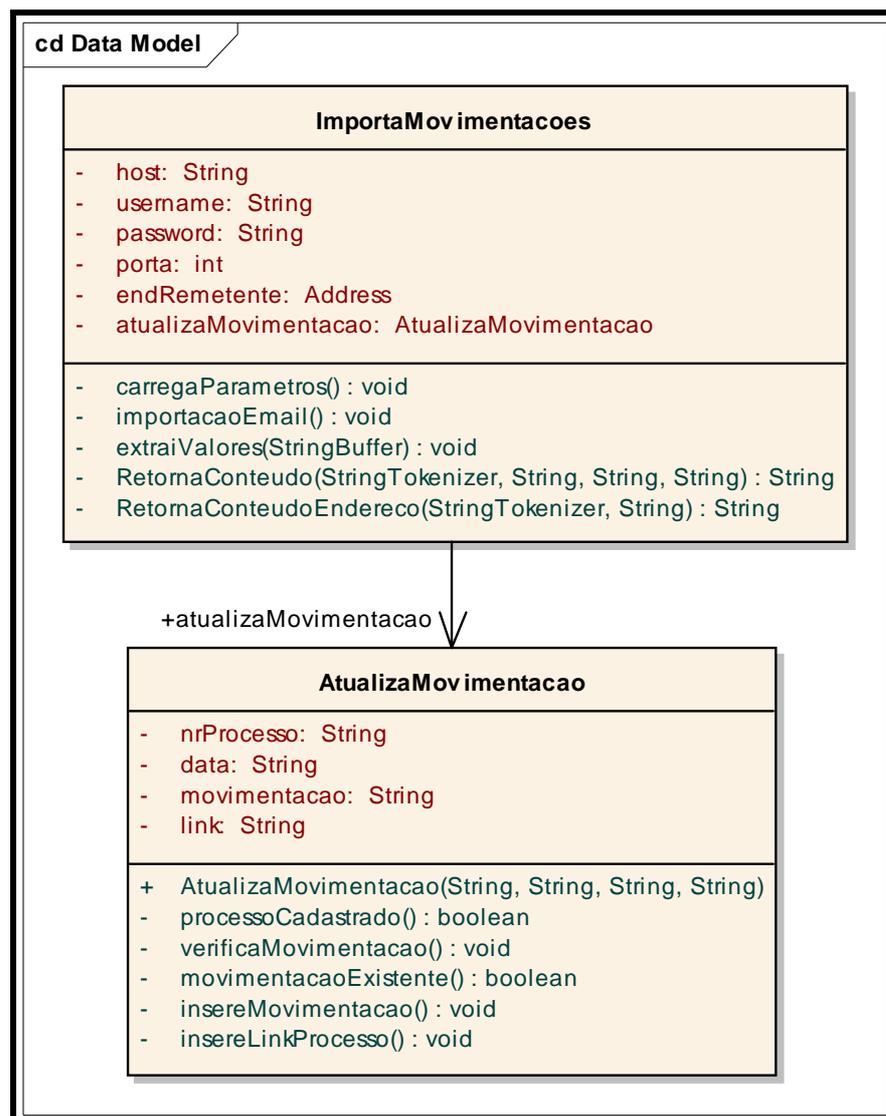


Figura 7 – Diagrama de classes da atualização das movimentações dos processos

A seguir é feita uma descrição das classes apresentadas no diagrama da figura 7.

3.2.2.1 Classe ImportaMovimentacoes

Esta classe é responsável pela importação dos dados dos *e-mails*, fazendo a conexão com o servidor de *e-mails* onde estão armazenados os *e-mails* enviados pelo Sistema *Push*.

Para fazer a conexão, primeiramente esta classe lê os dados para conexão, que estão armazenados na tabela de parâmetros da base de dados. Com esses dados é possível conectar-se ao servidor e procurar por *e-mails* enviados pelo Sistema *Push*.

A cada *e-mail* encontrado, é feita uma varredura e são extraídos os dados desejados da movimentação que serão gravados na base de dados. O formato da mensagem enviada pelo Sistema *Push* do TJSC é texto puro. Os dados que esta classe captura são: número do processo, data da movimentação, descrição da movimentação e *link* do *site* onde se encontram todas as movimentações do processo em questão.

Após ter todos os dados do *e-mail* capturados, serão passados para um objeto da classe *AtualizaMovimentacao*, que é responsável pela gravação dos dados.

3.2.2.2 Classe AtualizaMovimentacao

Esta classe é responsável pela gravação dos dados capturados do *e-mail*. Porém, nem todas as solicitações de gravação serão atendidas. A primeira etapa a ser feita é verificar se o processo da movimentação está cadastrado, se não estiver, a movimentação será ignorada e o objeto termina seu processamento.

Estando o processo cadastrado, será verificado se a movimentação já não foi gravada na base de dados, caso já exista, como ocorre com o processo não cadastrado, a

movimentação é ignorada e o objeto termina seu processamento.

O próximo passo será gravar os dados referentes à movimentação. O número do processo, a data da movimentação e a descrição da movimentação serão gravados na tabela movimentacoes. A tabela processos será atualizada com o *link* capturado do *e-mail*.

3.2.3 Diagrama de atividades

De acordo com Menezes (2002, p. 228), um diagrama de atividades é um tipo especial de diagrama de estados onde são representados os estados de uma atividade, ao invés dos estados de um objeto. Diferentemente dos diagramas de estados que são orientados a eventos, os diagramas de atividades são orientados a fluxos. Eles podem ser vistos como uma extensão dos fluxogramas, onde, além de existir a semântica de um fluxograma, possui notação para representar ações concorrentes, juntamente com a sua sincronização.

Conforme Menezes (2002, p. 228), os elementos de um diagrama de atividades podem ser divididos em dois grupos: os utilizados para representar fluxos de controle seqüenciais e os utilizados para representar fluxos de controle paralelos.

Na figura 8 é apresentado o diagrama de atividades do caso de uso importar movimentações.

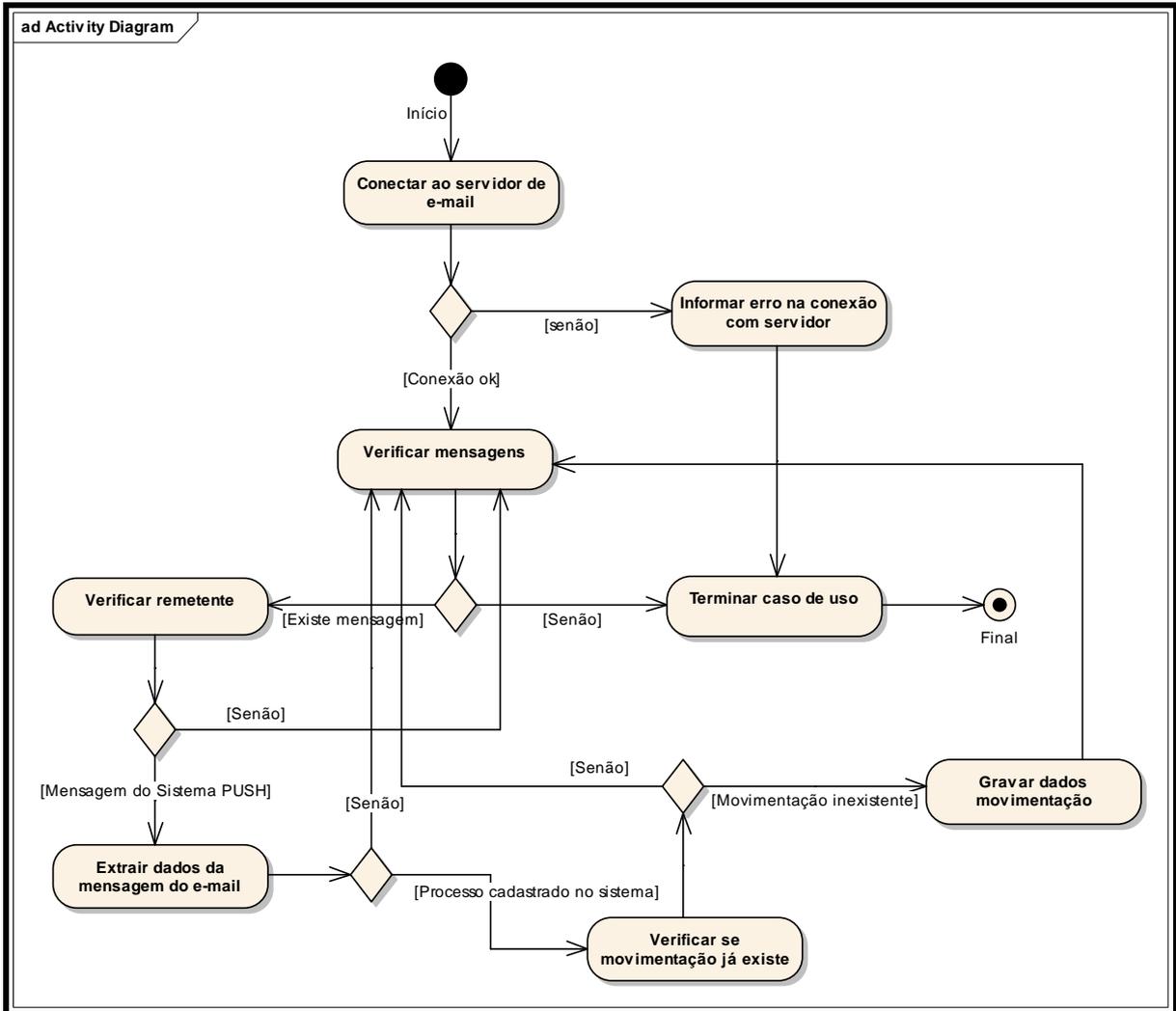


Figura 8 – Diagrama de atividades para o caso de uso importar movimentações

3.2.4 Modelo lógico de dados

A figura 9 apresenta o modelo lógico de dados do sistema, gerado com a ferramenta *PowerDesigner*.

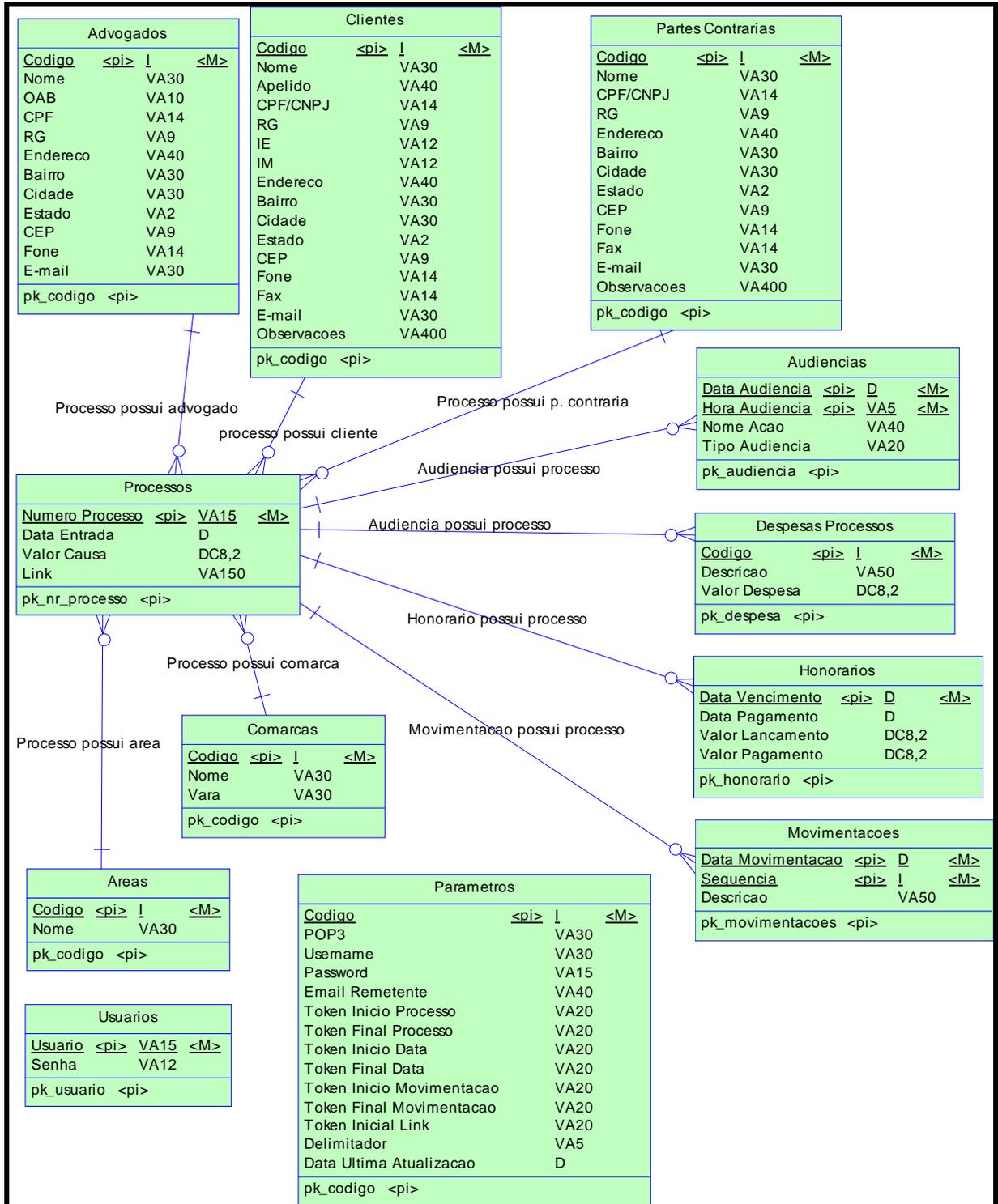


Figura 9 – Modelo lógico de dados

3.2.5 Modelo físico de dados

A figura 10 apresenta o modelo físico de dados do sistema, gerado com a ferramenta *PowerDesigner*.

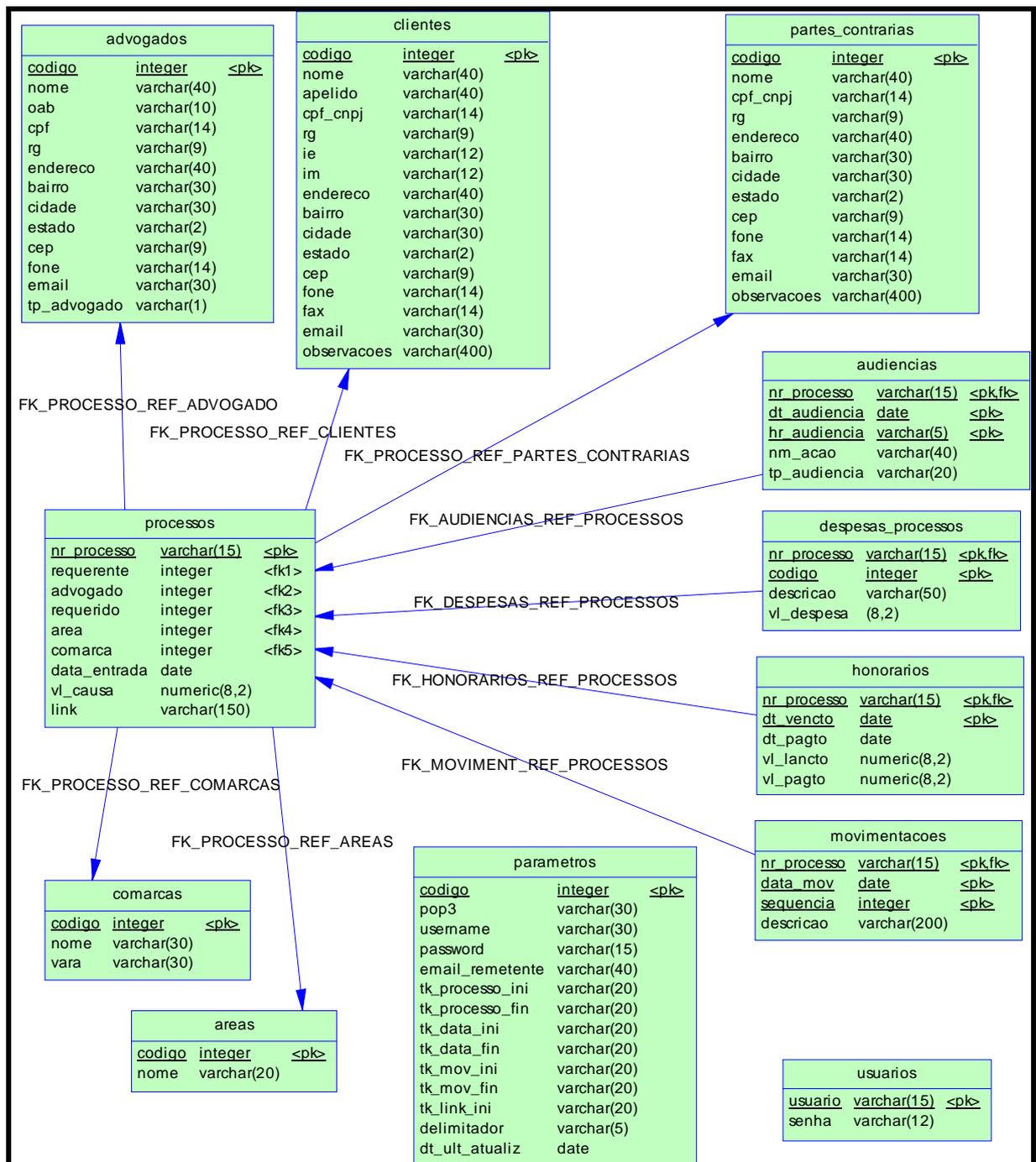


Figura 10 – Modelo físico de dados

3.3 IMPLEMENTAÇÃO

O sistema desenvolvido tem como propósito auxiliar na gestão de um escritório de advocacia, tendo como ponto diferencial, a atualização automática do andamento dos processos que são enviados pelo Sistema *Push* através de *e-mail*.

Nesta seção são abordados detalhes relativos à implementação, demonstrando sua operacionalidade e ferramentas utilizadas no desenvolvimento.

3.3.1 Técnicas e ferramentas utilizadas

O ambiente de desenvolvimento utilizado para a implementação foi o *JCreator Pro 2.00* com a *Java Virtual Machine (JVM)*. Para o armazenamento dos dados foi utilizado o Sistema Gerenciador de Banco de Dados (SGBD) *PostgreSQL 8.0*.

Segundo Alecrim (2004), o *PostgreSQL* é um banco de dados relacional e orientado a objetos. Seu grande triunfo é possuir recursos de bancos de grande porte, com o diferencial de ser gratuito.

Para a implementação do acesso ao servidor de *e-mail*, optou-se na utilização da *Application Program Interface (API) JavaMail*, utilizando o protocolo POP3. A API *JavaMail* é descrita na seção 3.3.1.1.

Na captura das informações da mensagem de *e-mail*, que possui seu formato em texto puro, foram implementados dois métodos: *RetornaConteudo()* e *RetornaConteudoEndereco()*. Na implementação dos dois métodos foi utilizada a classe *StringTokenizer* que faz parte do pacote *java.util* da JVM. Esta classe possui um método chamado *hasMoreTokens()* que possibilita a extração de cada palavra que faz parte de um texto. A implementação dos métodos *RetornaConteudo()* e *RetornaConteudoEndereco()* é apresentada nas seções 3.3.1.2 e

3.3.1.3, respectivamente.

3.3.1.1 API JavaMail

De acordo com Lima (2004, p. 5), *JavaMail* é um conjunto abstrato de classes compondo um *FrameWork* para envio, recebimento e manipulação de *e-mails*. Esse pacote da *Sun* contém implementações para protocolos POP3, IMAP e SMTP. O *FrameWork* permite criar aplicações de *e-mail* com muita facilidade, sem um conhecimento profundo sobre *e-mails*. Estão disponíveis métodos e classes que permitem acessar pastas de mensagens, fazer *download* de mensagens, envio com anexos e filtros de mensagens.

A API *JavaMail* tem como objetivos principais:

- a) ser simples e fácil de implementar;
- b) usar conceitos e interfaces que facilitam a integração com outras API *Java*;
- c) dar suporte ao desenvolvimento de aplicações de *e-mail* robustas manipulando uma variedade complexa de formatos de *e-mails*, tipos de dados, acessos e transportes de dados;
- d) permitir que desenvolvedores estendam suas classes para suas implementações particulares de sistemas de mensagens.

A arquitetura *JavaMail* está dividida em três camadas, como disposto a seguir:

- a) *Abstract Layer* – classes e interfaces que dão suporte à manipulação de *e-mails*.
Essa camada também suporta tipos de dados e interage com acesso a mensagens e protocolos de transporte;
- b) *Internet Implementation Layer* – utiliza os padrões de *internet* RFC822 e MIME;
- c) *Java Activation FrameWork* – *FrameWork* que encapsula e manipula vários tipos de dados de mensagens.

3.3.1.2 Método RetornaConteudo()

O método `RetornaConteudo()` é responsável pela extração do número do processo, data da movimentação e descrição da movimentação. Para cada informação a ser extraída é chamado este método, ou seja, será chamado três vezes, para capturar o número do processo, a data da movimentação e a descrição da movimentação. Para esse método, é passado quatro parâmetros, o primeiro é uma instância da classe *StringTokenizer* que contém a mensagem, o segundo parâmetro delimita o início do conteúdo a ser extraído após encontrar o *token* que inicia o conteúdo pretendido, e os dois últimos limitam o início e fim do conteúdo pretendido.

O método vai analisar cada *token* encontrado pelo método *hasMoreTokens* da Classe *StringTokenizer* até encontrar o *token* que indica o início do conteúdo, então verifica se o próximo *token* é o delimitador de início de conteúdo, como por exemplo “:” (dois pontos), caso não seja, o *token* encontrado anteriormente é descartado e o método continua a procura pelo *token* de início do conteúdo, caso seja, os próximos *tokens* encontrados são armazenados em uma variável, até que seja encontrado o *token* de fim do conteúdo. Para o *token* de fim ser válido, segue a mesma regra usada para o *token* de início, ou seja, deve ser precedido pelo *token* delimitador, senão será descartado seguindo a extração de *tokens*.

No quadro 3 é apresentado o código do método `RetornaConteudo()`.

```

private String RetornaConteudo(StringTokenizer st, String dlm, String t1, String t2){
    String resultado = "", tokenAnt = "", tk = "";
    int controle = 0; // 1 --> achou token t1
                    // 2 --> achou delimitador após t1, concatena primeiro token
                    // 3 --> concatena outros tokens
                    // 4 --> achou token final
                    // 5 --> achou token final seguido do delimitador

    while (st.hasMoreTokens()){
        // pega próximo token
        tk = st.nextToken();
        //verifica se achou o primeiro token
        if ((tk.equalsIgnoreCase(t1))&&(controle == 0)){ // achou o primeiro token
            controle = 1;
        }
        else {
            if (controle > 0){
                if (controle == 1){
                    if (tk.equals(dlm)){ // achou delimitador após primeiro token
                        controle = 2;
                    }
                    else {
                        controle = 0; // se não achou delimitador após primeiro token, não é válido
                    }
                }
                else {
                    if (controle == 2){ // acumula primeiro token
                        resultado = tk;
                        controle = 3;
                    }
                    else {
                        if (tk.equalsIgnoreCase(t2)){ // verifica se se encontrou t2
                            if (controle == 4){ // Já encontrou token final anteriormente
                                resultado = resultado + " "+tokenAnt;
                            }
                            controle = 4; // achou token final
                            tokenAnt = tk;
                            continue; // vai para próximo token
                        }

                        if (controle == 3){ // concatena outros tokens
                            resultado = resultado + " "+tk;
                        }
                        else {
                            if (controle == 4){
                                if (tk.equals(dlm)){ // na segunda vez que passa com controle = 4
                                    controle = 5; // achou delimitador após token final, sai fora
                                    break;
                                }
                                else {
                                    resultado = resultado + " "+tokenAnt+" "+tk;
                                    controle = 3;
                                }
                            }
                        }
                    }
                }
            }
        }
    }
    return resultado;
}

```

Quadro 3 – Método RetornaConteudo()

3.3.1.3 Método RetornaConteudoEndereco()

O método RetornaConteudoEndereco() tem a função de extrair o *link* onde estão as movimentações no *site* do TJSC. Para esse método é passado como parâmetro uma instância da classe *StringTokenizer* e o *token* que indica o início do *link*. O método analisa cada *token* até encontrar o *token* de início, quando encontrar, o próximo *token* é o *link*.

No quadro 4 é apresentado o código do método RetornaConteudoEndereco().

```
private String RetornaConteudoEndereco(StringTokenizer st, String tokenEnd){
    String tk = "";
    while (st.hasMoreTokens()){
        // pega próximo token
        tk = st.nextToken();
        //verifica se achou token
        if ((tk.equalsIgnoreCase(tokenEnd))){ // achou o primeiro token
            tk = st.nextToken();
            break; // quebra loop
        }
    }
    return tk;
}
```

Quadro 4 – Método RetornaConteudoEndereco()

3.3.2 Operacionalidade da implementação

Para um melhor entendimento das funcionalidades, as principais telas são apresentadas a seguir, com explicação de suas funcionalidades.

O primeiro passo ao acessar o sistema é efetuar o *login* informando usuário e senha, como apresentado na figura 11. Na implantação do sistema há um usuário *default*, que será utilizado para acessar o sistema até a criação dos usuários reais.

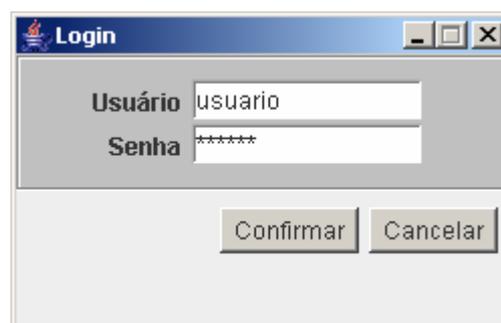


Figura 11 – Login do sistema

Após efetuar o login, o sistema verifica se na data corrente já foi executada a verificação dos *e-mails* do Sistema *Push*. Caso ainda não foi executada, uma mensagem pedirá confirmação para que seja executado, como apresentado na figura 12. Se optar por não executar no momento, a cada acesso ao sistema será solicitado, até que a verificação seja

feita. Se optar em executar a importação no momento, será aberta a tela, como apresentado na figura 13. Esta operação também poderá ser executada a qualquer momento, pelo sistema, acessando o menu Movimentações e escolhendo o item Importar Movimentações.



Figura 12 – Mensagem de solicitação de verificação de *e-mail*



Figura 13 – Importação de movimentações processuais do servidor de *e-mails*

O menu do sistema pode ser visualizado na figura 14. Está dividido em:

- a) cadastros: possui funcionalidades cadastrais e contém os itens Clientes, Advogados, Partes Contrárias, Advogados Contrários, Áreas, Comarcas, Processos, Usuários e Agendar Audiências;
- b) movimentações: este menu tem como itens Importar Movimentações, Inserir Movimentações, Agenda das Movimentações e Agenda de Audiências;
- c) financeiro: possui funcionalidades ligadas a parte financeira do sistema e contém os itens Despesas de Processos, Lançamentos de Honorários e Baixa de Honorários;
- d) relatórios: contém os relatórios do sistema. Possui os itens Clientes, Advogados, Movimentação de Processos e Honorários Pagos/Pendentes;
- e) configurações: possui funcionalidade para alteração de senhas e opções para

manter dados utilizados na importação das movimentações dos processos armazenados no servidor. Possui as funcionalidades Alteração de Senha, Alteração de *E-mails* e Alteração de *Tokens*.



Figura 14 – Menu do sistema

Os clientes podem ser cadastrados através da tela de cadastro de clientes, a qual é apresentada na figura 15.

A screenshot of a software application window titled "Cadastro de Clientes". The window contains a form with the following fields and values:

- Código:** 3
- Nome / Razão Social:** Marcus Vinicius Felippi
- Apelido / Fantasia:** Vinicius
- CPF / CNPJ:** 022.734.284.52
- RG:** 3.458.782
- IE:** (empty)
- IM:** (empty)
- Endereço:** Rua Tenente Carlos Deeke, 45
- Bairro:** Velha
- Cidade:** Blumenau
- Estado:** SC
- CEP:** 89032-347
- Fone:** (47)330-7588
- Fax:** (47)330-7588
- E-mail:** vinicius@inf.furb.br
- Observações:** (empty text area)

At the bottom of the form, there are four buttons: "Salvar", "Excluir", "Cancelar", and "Fechar".

Figura 15 – Cadastro de clientes

A tela de cadastro de advogados é apresentada na figura 16.

Código	1		
Nome	Jayson Cássio Muller Rosa		
OAB	0142875		
CPF	154.854.658-78	RG	5.684.857
Endereço	Rua 7 de Setembro, 550		
Bairro	Centro		
Cidade	Blumenau	Estado	SC
CEP	89010-542		
Fone	(47)326-8597		
E-mail	jaysonrosa@terra.com.br		

Salvar Excluir Cancelar Fechar

Figura 16 – Cadastro de advogados

A tela de partes contrárias tem como objetivo cadastrar a pessoa oposta ao cliente no processo. Esta tela é apresentada na figura 17.

Código	1		
Nome / Razão Social	João da Silva		
CPF / CNPJ	023.842.584-85	RG	2.568.452
Endereço	Rua XV de Novembro, 523		
Bairro	Centro		
Cidade	Blumenau	Estado	SC
CEP	89542-054		
Fone	(47)322-8554	Fax	() -
E-mail	joaosilva@hotmail.com		
Observações			

Salvar Excluir Cancelar Fechar

Figura 17 – Cadastro de partes contrárias

Na figura 18 é apresentada a tela de cadastro de advogados contrários.

Cadastro de Advogados Contrários

Código 2

Nome Pedro de Souza Mello

OAB 5488658

CPF 542.895.748-32 RG 5.478.823

Endereço Rua General Osório, 250

Bairro Velha

Cidade Blumenau Estado SC

CEP 89523-110

Fone (47)328-5489

E-mail pedrosm@gmail.com

Salvar Excluir Cancelar Fechar

Figura 18 – Cadastro de advogados contrários

Na figura 19 é apresentada a tela de cadastro de áreas.

Cadastro de Áreas

Código 2

Nome Cível

Salvar Excluir Cancelar Fechar

Figura 19 – Cadastro de áreas

A tela de cadastro de comarcas é apresentada na figura 20.

Cadastro de Comarcas

Código 2

Nome Blumenau

Vara 2ª Vara Cível

Salvar Excluir Cancelar Fechar

Figura 20 – Cadastro de comarcas

Para fazer o cadastro de um processo é necessário estar previamente cadastrado o advogado, o cliente, a parte contrária, o advogado contrário, a área e a comarca. Na figura 21, é apresentada a tela de cadastro de processos. Nesta mesma tela é possível visualizar as movimentações ocorridas para o processo, basta clicar no botão Movimentações para abrir a tela de movimentações apresentada na figura 22.

Figura 21 – Cadastro de processos

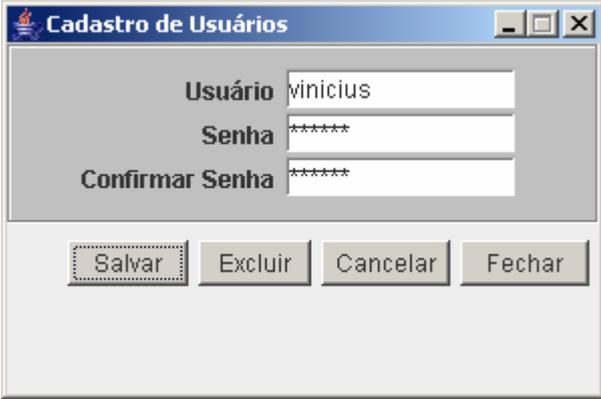
Data	Movimentação
13/10/2005	Certificado decurso de prazo
14/10/2005	Concluso para despacho
14/10/2005	Aguardando envio para o Juiz
18/10/2005	Recebimento
18/10/2005	Despacho outros
04/11/2005	Aguardando publicação
10/11/2005	Publicação de edital para intimação de advogado
16/11/2005	Carga ao Advogado
16/11/2005	Aguardando envio para o Advogado

Para maiores detalhes acesse o link abaixo:

<http://tjsc5.tj.sc.gov.br/cpopg/pcpoSelecaoPG.jsp?cbPesquisa=NUMPROC&dePesquisa=149.03.005352-1&cdForo=149>

Figura 22 – Movimentações do processo

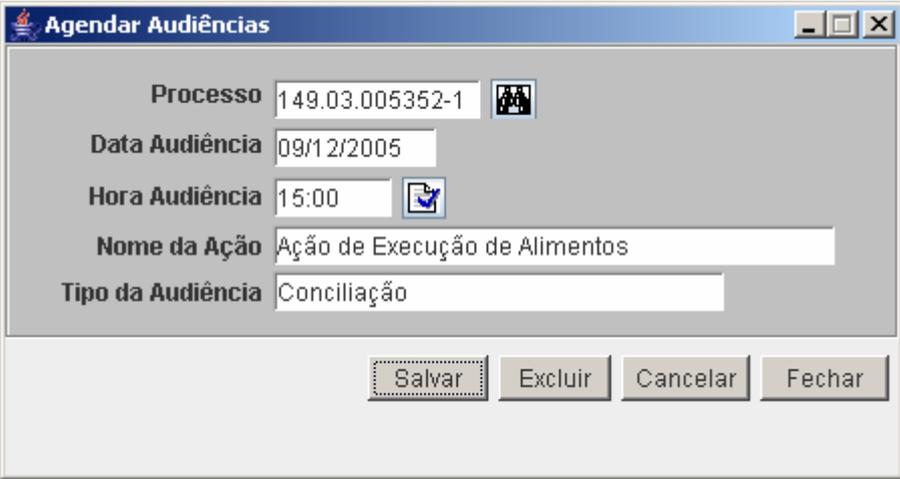
Na figura 23 é apresentada a tela de cadastro de usuários.



A janela 'Cadastro de Usuários' possui um cabeçalho com o ícone de uma chama e o título 'Cadastro de Usuários'. O formulário contém três campos de texto: 'Usuário' com o valor 'vinicius', 'Senha' com '*****' e 'Confirmar Senha' com '*****'. Abaixo dos campos, há quatro botões: 'Salvar' (destacado com uma borda pontilhada), 'Excluir', 'Cancelar' e 'Fechar'.

Figura 23 – Cadastro de usuários

Na figura 24 é apresentada a tela de agendamento de audiências.



A janela 'Agendar Audiências' possui um cabeçalho com o ícone de uma chama e o título 'Agendar Audiências'. O formulário contém cinco campos: 'Processo' com '149.03.005352-1' e ícone de pessoas; 'Data Audiência' com '09/12/2005'; 'Hora Audiência' com '15:00' e ícone de relógio; 'Nome da Ação' com 'Ação de Execução de Alimentos'; e 'Tipo da Audiência' com 'Conciliação'. Abaixo dos campos, há quatro botões: 'Salvar' (destacado com uma borda pontilhada), 'Excluir', 'Cancelar' e 'Fechar'.

Figura 24 – Agendamento de audiências

No menu Movimentações encontra-se a tela de importação de movimentações de processos, apresentada na figura 13. Ela é responsável pela busca de movimentações que está no servidor de *e-mails*.

Na figura 25 é apresentada a tela de inserção de movimentação manualmente. Esta opção serve apenas como contingência, caso ocorra algum problema na importação da movimentação do *e-mail*. Nesta tela é possível inserir movimentações para processos cadastrados, digitando-se a data e a descrição da movimentação.

Inserção de Movimentação

Processo: 070.01.001845-2

Data Movimentação: 16/11/2005

Movimentação: Publicação de edital para intimação de advogado

Salvar Cancelar Fechar

Figura 25 – Inserção de movimentação manualmente

Na figura 26 é apresentada a agenda de movimentações dos processos, partindo de uma data informada. São apresentadas todas as movimentações dos processos ocorridos a partir da data desejada. As movimentações são listadas em ordem decrescente de data.

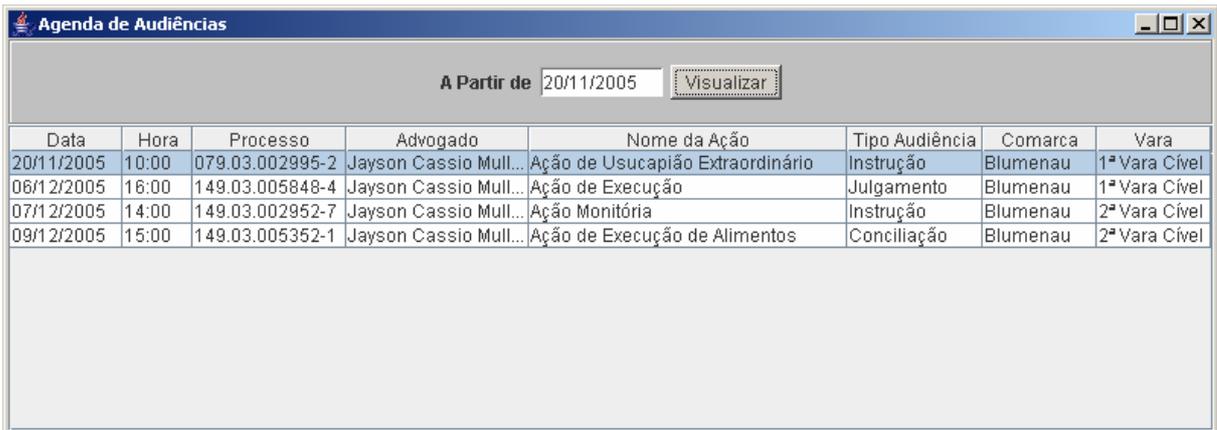
Agenda de Movimentações

A Partir de 01/10/2005 Visualizar

Data	Processo	Movimentação
18/11/2005	149.03.005848-4	Processo arquivado definitivamente
16/11/2005	149.03.005352-1	Carga ao Advogado
16/11/2005	149.03.005848-4	Certificado trânsito em julgado
16/11/2005	149.03.005352-1	Aguardando envio para o Advogado
11/11/2005	149.03.005848-4	Juntada de carta precatória
10/11/2005	149.03.005352-1	Publicação de edital para intimação de advogado
08/11/2005	149.03.002952-7	Processo suspenso
08/11/2005	149.03.002952-7	Decisão determinando suspensão (Cível)
07/11/2005	079.03.002995-2	Processo suspenso
07/11/2005	070.01.001754-2	Decisão determinando suspensão (Cível)
04/11/2005	149.03.005352-1	Aguardando publicação
27/10/2005	149.03.002952-7	Juntada de mandado

Figura 26 – Agenda das movimentações

Na figura 27 é apresentada a agenda das audiências informadas na tela da figura 24. Na tela da figura 27 é informado a partir de que data deseja-se listar as audiências. As audiências são listadas em ordem cronológica de data.

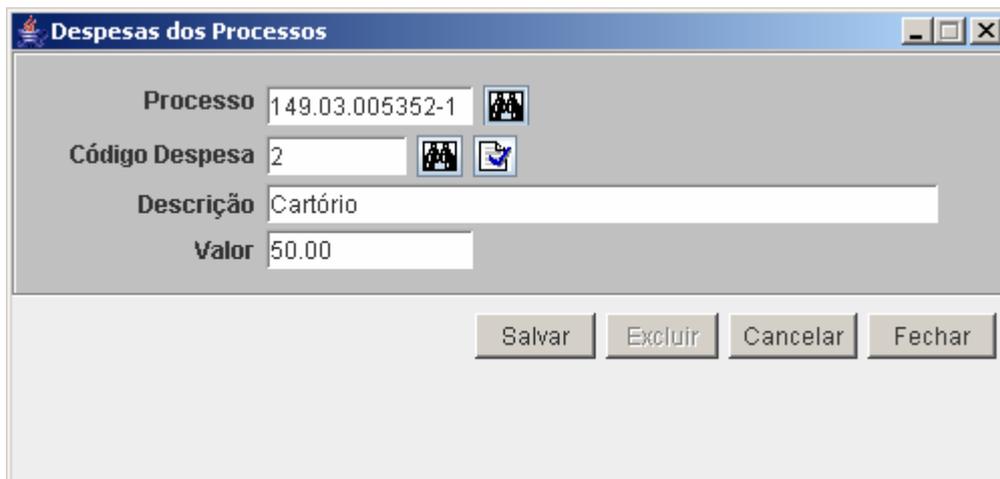


The screenshot shows a window titled "Agenda de Audiências" with a search filter "A Partir de 20/11/2005" and a "Visualizar" button. Below is a table with the following data:

Data	Hora	Processo	Advogado	Nome da Ação	Tipo Audiência	Comarca	Vara
20/11/2005	10:00	079.03.002995-2	Jayson Cassio Mull...	Ação de Usucapião Extraordinário	Instrução	Blumenau	1ª Vara Cível
06/12/2005	16:00	149.03.005848-4	Jayson Cassio Mull...	Ação de Execução	Julgamento	Blumenau	1ª Vara Cível
07/12/2005	14:00	149.03.002952-7	Jayson Cassio Mull...	Ação Monitória	Instrução	Blumenau	2ª Vara Cível
09/12/2005	15:00	149.03.005352-1	Jayson Cassio Mull...	Ação de Execução de Alimentos	Conciliação	Blumenau	2ª Vara Cível

Figura 27 – Agenda de audiências

No menu financeiro encontra-se uma funcionalidade para lançamento de despesas gastas com processos. Há uma consulta de processos para facilitar seu uso e uma consulta de despesas lançadas para o processo informado. Se não for informado processo, não será possível consultar os lançamentos. Na figura 28 pode-se ver a tela da funcionalidade.



The screenshot shows a window titled "Despesas dos Processos" with the following fields and buttons:

- Processo:** 149.03.005352-1
- Código Despesa:** 2
- Descrição:** Cartório
- Valor:** 50.00

Buttons at the bottom: Salvar, Excluir, Cancelar, Fechar.

Figura 28 – Despesas de processos

Outra funcionalidade encontrada no menu financeiro é a tela de lançamento de honorários. Há uma consulta de processos para facilitar seu uso e uma consulta para ver os honorários já lançados para o processo. Nesta funcionalidade, o sistema não permitirá lançar honorários em que sua soma seja maior do que o valor da causa informada no cadastro do processo. Também não poderá ser alterado ou excluído honorário já pago (figura 29).



The screenshot shows a dialog box titled "Lançamento de Honorários". It contains four input fields: "Processo" with the value "149.03.005352-1", "Data Vencimento" with "17/11/2005", "Valor" with "150.00", and "Saldo a Lançar" with "350.00". There are two small icons to the right of the "Processo" and "Data Vencimento" fields. At the bottom, there are four buttons: "Salvar", "Excluir", "Cancelar", and "Fechar".

Figura 29 – Lançamento de honorários

Na figura 30 é apresentada a tela de baixa de honorários. O sistema não permitirá que o usuário faça manutenção em um honorário já pago.



The screenshot shows a dialog box titled "Baixa de Honorários". It contains four input fields: "Processo" with "149.03.005352-1", "Data Vencimento" with "17/11/2005", "Data Pagamento" with "17/11/2005", and "Valor" with "150.00". There are two small icons to the right of the "Processo" and "Data Vencimento" fields. At the bottom, there are three buttons: "Salvar", "Cancelar", and "Fechar".

Figura 30 – Baixa de honorários

Nas figuras 31 e 32 são apresentados a tela de filtros e o relatório de clientes gerado, respectivamente.



The screenshot shows a dialog box titled "Relatório de Clientes". It has two input fields for "Cliente" with values "0" and "99999" separated by the letter "a". At the bottom, there are two buttons: "Visualizar" and "Fechar".

Figura 31 – Tela de filtros do relatório de clientes

Relatório de Clientes Data: 02/11/2005

Seleção - Cliente: 0 a 99999

Código	Nome/Razão Social	Bairro	CPF/CNPJ
Endereço		Estado	CEP
Cidade		E-mail	
Fone	Fax		
8	Marcus Vinícius Felippi	Água Verde	022.734.259-37
Rua Tenente Carlos Deeke, S/N		SC	89032-347
Blumenau		viniciusf@al.furb.br	
(47)330-7798	(47)330-7798		
9	Juca de Souza	Centro	052.854.562-45
Alameda Rio Branco, 452		SC	89054-221
Blumenau		jucasouza@gmail.com	
(47)322-5896	() -		

Page 1 of 1

Figura 32 – Relatório de clientes visualizado

Nas figuras 33 e 34 são apresentados a tela de filtros e o relatório de advogados gerado, respectivamente.

Relatório de Advogados

Advogado 0 a 99999

Visualizar Fechar

Figura 33 – Tela de filtros do relatório de advogados

Seleção - Advogado: 0 a 99999		Data: 02/11/2005	
Código	Nome	OAB	
CPF	RG		
Endereço	Bairro		
Cidade	Estado	CEP	
Fone	E-mail		
1	Jayson Cássio Muller Rosa	123456789	
022.734.259-37	3.448.552		
Rua das Palmeiras, 299	Vila Nova		
Blumenau	SC	89032-347	
(47)330-7798	jaysonrosa@hotmail.com		
6	Carlos Honesto	5421578	
125.454.124-54	5.324.852		
Rua da palmeiras, 50	Centro		
Blumenau	SC	89541-245	
(47)322-8564	carloshonesto@ibest.com.br		

Page 1 of 1

Figura 34 – Relatório de advogados visualizado

Nas figuras 35 e 36 são apresentados a tela de filtros e o relatório de movimentações de processos, respectivamente. Este relatório contém opção de listar o relatório por filtro de clientes ou advogados e o período que deseja ver as movimentações.

Relatório de Movimentação dos Processos

Opção **Advogado** ▼

Advogado 1 a 10

Período 15/10/2005 a 30/10/2005

Visualizar Fechar

Figura 35 – Tela de filtros do relatório de movimentações de processos

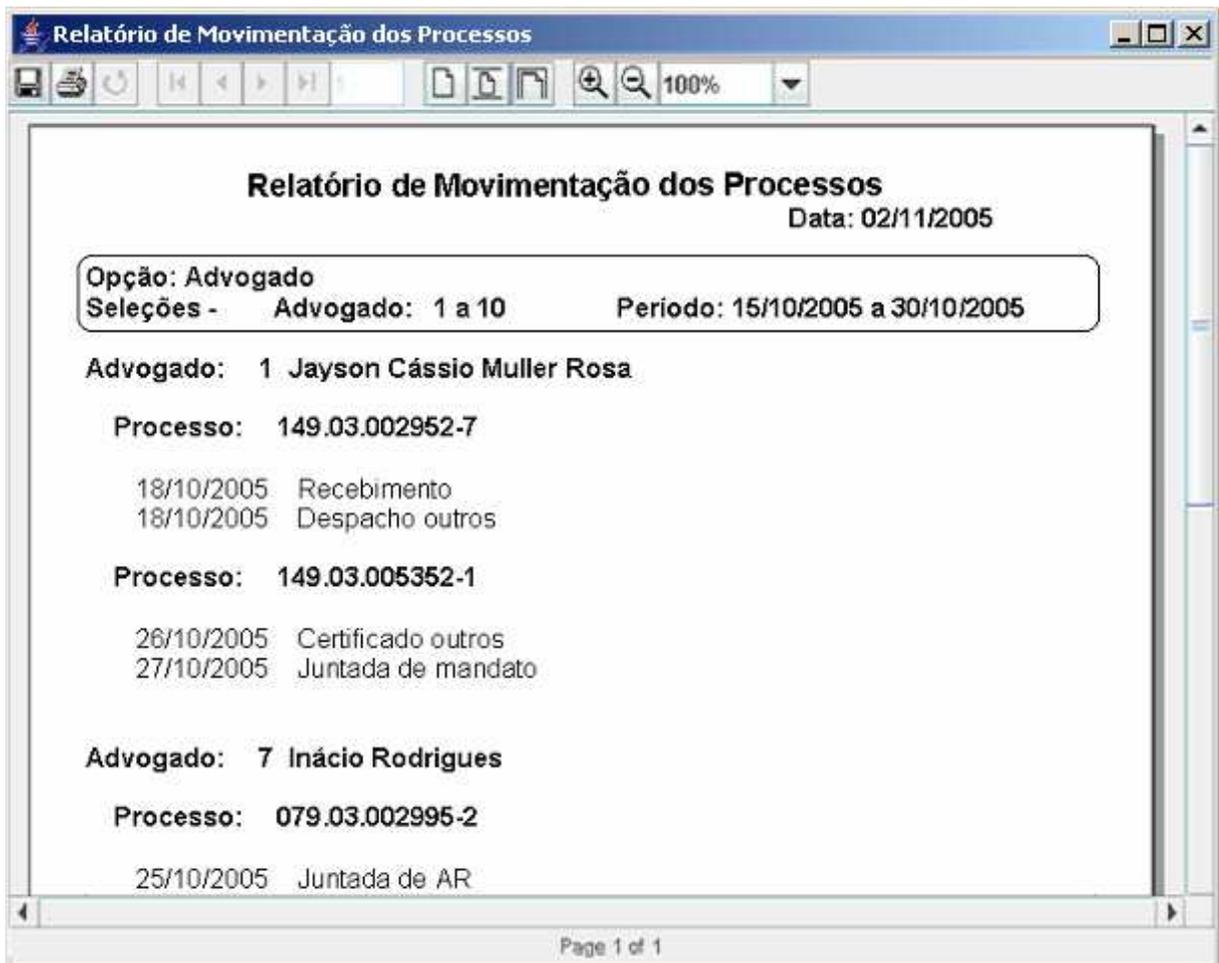


Figura 36 – Relatório de movimentações de processos visualizado

Nas figuras 37 e 38 são apresentados a tela de filtros e o relatório de honorários pagos e pendentes, respectivamente. Este relatório contém opção de listar o relatório por filtro de honorários pagos ou pendentes e o período que deseja visualizar.

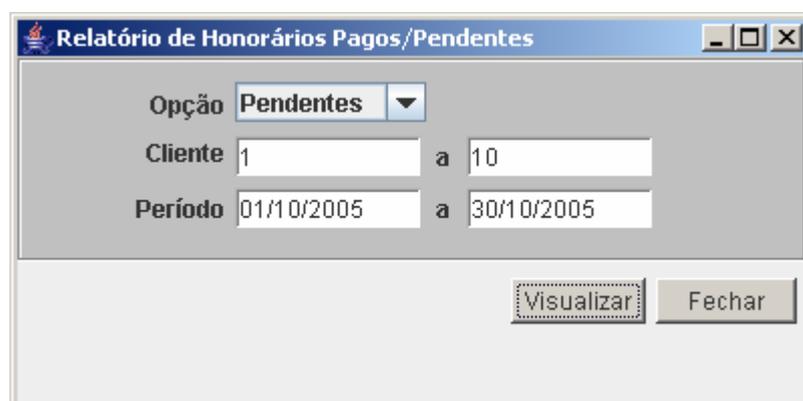


Figura 37 – Tela de filtros do relatório de situação de honorários

Relatório de Honorários Pagos/Pendentes
Data: 02/11/2005

Opção: Pendentes
Seleções - Cliente: 1 a 10 Período: 01/10/2005 a 30/10/2005

Vencimento	Valor
Cliente: 8 Marcus Vinícius Felippi	
Processo: 149.03.005352-1	Data Entrada: 06/06/2005
01/10/2005	150.00
15/10/2005	100.00
Subtotal	250.00
Total do Cliente	250.00
Cliente: 9 Juca de Souza	
Processo: 149.03.002952-7	Data Entrada: 08/10/2005
10/10/2005	100.00
25/10/2005	50.00
Subtotal	150.00

Figura 38 – Relatório de situação de honorários visualizado

A figura 39 apresenta a tela de alteração de senhas dos usuários cadastrados. Esta tela é exclusivamente para alteração de senha de usuários já cadastrados.

Alteração de Senha

Usuário: vinicius

Senha Atual: *****

Senha Nova: *****

Confirmar Senha: *****

Salvar Cancelar Fechar

Figura 39 – Alteração de senha

Na figura 40 é apresentada a tela de alteração de *e-mails*. Esta funcionalidade tem o objetivo de manter atualizados os dados de *e-mails* utilizados na importação dos *e-mails* de movimentações.

A janela 'Alteração de E-mails' contém os seguintes campos e botões:

Pop3	pop.al.furb.br
Usuário	viniciusf
Senha	*****
E-mail Remetente	tjscpusth@tj.sc.gov.br

Botões: Salvar, Fechar

Figura 40 – Alteração de *e-mails*

Na figura 41 é apresentada a tela de alteração de *tokens*. Estes *tokens* são utilizados como delimitadores para capturar os dados corretos de cada *e-mail* enviado pelo Sistema *Push*. Eles só devem ser alterados se o *layout* do *e-mail* mudar, não necessitando de nova implementação para a captura das informações. Porém, dependendo da alteração ocorrida no *layout* do *e-mail*, alterar os *tokens* não será suficiente para manter a integridade na importação.

A janela 'Alteração de Tokens' contém os seguintes campos e botões:

Início Processo	Processo	Fim Processo	Data
Início Data	Data	Fim Data	Movimentação
Início Movimentação	Movimentação	Fim Movimentação	Vara
Início Link	endereço:		
Delimitador	:		

Botões: Salvar, Fechar

Figura 41 – Alteração de *tokens*

O sistema também permite consulta à informações cadastradas. Um exemplo é a tela de consulta de clientes, a qual pode ser visualizada na figura 42.



Código	Nome	CPF/CNPJ
1	Marcus Vinícius Felippi	022.734.284-48
2	Mário da Cruz	542.485.542-88
3	Paulo Rdrigues	485.894.211.56
4	João Luiz	548.235.451-85
5	Márcio Pereira	845.654.598-54
6	Jorge Dias	154.542.875-21

Figura 42 – Consulta de clientes

3.4 RESULTADOS E DISCUSSÃO

Este trabalho apresentou o desenvolvimento de um sistema para escritórios de advocacia com a automatização do andamento dos processos enviados pelo Sistema *Push* através de *e-mail*. Contendo uma agenda com as últimas movimentações dos processos, que permite ao advogado tomar as ações necessárias para a agilidade do processo.

Em Raduenz (2000), foi apresentado um protótipo de um aplicativo para advocacia com recursos voltados à *internet* através do uso da linguagem ASP. No entanto, este protótipo teve como principal objetivo permitir que clientes consultem a situação dos processos na página do protótipo.

Os trabalhos desenvolvidos por Regis (2000) e Knorst (2000) tiveram como objetivo apenas o monitoramento de *e-mails*. No trabalho desenvolvido, além da captura de dados de *e-mails* provenientes do Sistema *Push*, foi desenvolvido um sistema de advocacia, que tem sua base de dados atualizada com as informações referentes às movimentações dos processos.

Um grande diferencial em relação aos sistemas de advocacia pesquisados, como o Agente Jurídico, é que as movimentações dos processos são gravadas automaticamente no

sistema ao verificar as mensagens existentes no servidor de *e-mails*, não precisando digitar, ou copiar e colar do *e-mail* para o aplicativo a movimentação de um determinado processo, dando mais agilidade e segurança ao sistema.

4 CONCLUSÕES

Atualmente, agilizar os processos em qualquer negócio é sempre um grande objetivo a ser atingido. Em um escritório de advocacia não pode ser diferente, por esse motivo, que um sistema computacional é essencial para dar mais agilidade e organização no armazenamento das informações e na tomada de decisão.

Este trabalho apresentou o desenvolvimento de um sistema de gestão para advocacia, tendo como seu principal foco e diferencial a captura das movimentações dos processos recebidos através de *e-mail*.

O maior objetivo deste trabalho foi manter as movimentações dos processos atualizadas sem que isso fosse feito manualmente. Essas movimentações são informações importantes, ficando disponíveis para consulta na base de dados em qualquer momento em que o usuário necessitar, tendo uma visão mais ampla de como ocorreu o andamento dos processos cadastrados no sistema.

A tecnologia utilizada para a conexão com o servidor de *e-mail* e captura dos *e-mails* desejados foi uma API do *Java* chamada *Javamail*. Esta API possui recursos para conexão ao servidor, pesquisa por remetente e leitura de mensagens. Ela mostrou-se muito eficiente e prática.

Para a captura das movimentações dos processos existentes nos *e-mails*, não encontrou-se nenhuma técnica existente que atendesse ao esperado. No entanto, foi possível criar uma rotina que analisa o texto do *e-mail*, *token a token*, retirando as informações necessárias para a atualização do sistema.

De maneira geral, conclui-se que o trabalho contribui para o desenvolvimento de sistemas que capturam informações de *e-mails*, não só na área de advocacia, mas em qualquer área desejada, onde informações capturadas de *e-mails* sejam importantes.

4.1 EXTENSÕES

Como sugestão de extensão deste trabalho pode-se estudar a viabilidade de buscar as movimentações diretamente no *site* do TJSC, ao invés de capturar as movimentações a partir dos *e-mails* enviados pelo Sistema *Push*. Isso tornaria o sistema bem mais prático e flexível, pois o usuário não necessitaria cadastrar os processos no *site* do Sistema *Push*.

REFERÊNCIAS BIBLIOGRÁFICAS

ALECRIM, Emerson. **Banco de dados PostgreSQL e MySQL**. [S. l.], [2004?]. Disponível em: <<http://www.infowester.com/col191204.php>>. Acesso em: 01 nov. 2005.

BAETA, Márcio N. **O microcomputador no escritório: modernização da advocacia**. Brasília: Jarbex, 1983.

COMER, Douglas E. **Redes de computadores e internet**. 2. ed. Porto Alegre: Bookman, 2001.

DEBONI, José Eduardo Zindel. **Modelagem orientada a objetos com UML**. São Paulo: Futura, 2003.

KNORST, Alexandre Eduardo. **Protótipo de software para notificação de recebimento de novos e-mails, baseado no protocolo IMAP**. 2000. 81 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

KUROSE, James F.; ROSS, Keith W. **Redes de computadores e a internet: uma nova abordagem**. São Paulo: Pearson Education do Brasil, 2003.

LIMA, Adilson da Silva. **UML 2.0: do requisito à solução**. São Paulo: Érica, 2005.

LIMA, Johnatan Daniel Fromholz. **JavaMail: guia prático**. Florianópolis: VisualBooks, 2004.

MARQUES, Luiz Guilherme. **Temas de direito: brasileiro e estrangeiro**. [S. l.], [2004?]. Disponível em: <<http://www.artnet.com.br/~lgm/down1.doc>>. Acesso em: 17 out. 2005.

MENEZES, Eduardo Diatahy Bezerra. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Campus, 2002.

MOREIRA, José Carlos Barbosa. **Temas de direito processual**. 5. ed. São Paulo: Saraiva, 1994.

PONTE SOFTWARE. **Agente jurídico**. [S. l.], [2004?]. Disponível em: <<http://www.pontesoftware.com.br/agente/framesetagente.htm>>. Acesso em: 11 nov. 2005.

RADUENZ, Gilmar Vereano. **Protótipo de aplicativo para escritório de advocacia com acesso a internet baseado em ASP**. 2000. 50 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

REGIS, Eduardo Jean. **Protótipo de software para distribuição de arquivos recebidos por e-mail via internet**. 2000. 62 f. Trabalho de Conclusão de Curso (Bacharelado em Ciências da Computação) – Centro de Ciências Exatas e Naturais, Universidade Regional de Blumenau, Blumenau.

SUPERIOR TRIBUNAL DE JUSTIÇA. **Sistema Push**. Brasília, [2004?]. Disponível em: <<http://www.stj.gov.br/webstj/processo/Push/Ajuda.asp>>. Acesso em: 10 nov. 2005.

APÊNDICE A – Descrição de casos de uso

Manter Clientes: inclusão, alteração, exclusão e consulta de clientes	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de clientes. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de um novo cliente. b. O sistema apresenta uma tela em branco para o preenchimento dos dados do cliente. c. O Usuário fornece os dados do novo cliente. d. O sistema inclui o novo cliente. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa o cliente que deseja alterar. b. O usuário altera um ou mais dados referentes a um cliente e solicita a sua atualização. c. O sistema altera os dados do cliente. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa o cliente que deseja excluir. b. Se o cliente pode ser excluído, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta dos clientes cadastrados. b. O sistema apresenta a lista de todos os clientes cadastrados, permitindo que o usuário selecione o cliente desejado. c. O sistema mostra os detalhes do cliente selecionado e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de cliente utilizado em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Cliente cadastrado, alterado ou excluído.
Regras do negócio	Não há.

Manter Advogados: inclusão, alteração, exclusão e consulta de advogados	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de advogados. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de um novo advogado. b. O sistema apresenta uma tela em branco para o preenchimento dos dados do advogado. c. O Usuário fornece os dados do novo advogado. d. O sistema inclui o novo advogado. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa o advogado que deseja alterar. b. O usuário altera um ou mais dados referentes a um advogado e solicita a sua atualização. c. O sistema altera os dados do advogado. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa o advogado que deseja excluir. b. Se o advogado pode ser excluído, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta dos clientes cadastrados. b. O sistema apresenta a lista de todos os advogados cadastrados, permitindo que o usuário selecione o advogado desejado. c. O sistema mostra os detalhes do advogado selecionado e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de advogado utilizado em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Advogado cadastrado, alterado ou excluído.
Regras do negócio	Não há.

Manter Partes Contrárias: inclusão, alteração, exclusão e consulta de partes contrárias	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de partes contrárias. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O sistema apresenta uma tela em branco para o preenchimento dos dados da parte contrária. b. O Usuário fornece os dados da nova parte contrária. c. O sistema inclui a nova parte contrária. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa a parte contrária que deseja alterar. b. O usuário altera um ou mais dados referentes a uma parte contrária e solicita a sua atualização. c. O sistema altera os dados da parte contrária. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa a parte contrária que deseja excluir. b. Se a parte contrária pode ser excluída, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta das partes contrárias cadastradas. b. O sistema apresenta a lista de todas as partes contrárias cadastradas, permitindo que o usuário selecione a parte contrária desejada. c. O sistema mostra os detalhes da parte contrária selecionada e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de parte contrária utilizada em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Parte contrária cadastrada, alterada ou excluída.
Regras do negócio	Não há.

Manter Advogados Contrários: inclusão, alteração, exclusão e consulta de advogados contrários	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de advogados contrários. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de um novo advogado contrário. b. O sistema apresenta uma tela em branco para o preenchimento dos dados do advogado contrário. c. O Usuário fornece os dados do novo advogado contrário. d. O sistema inclui o novo advogado contrário. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa o advogado contrário que deseja alterar. b. O usuário altera um ou mais dados referentes a um advogado contrário e solicita a sua atualização. c. O sistema altera os dados do advogado contrário. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa o advogado contrário que deseja excluir. b. Se o advogado contrário pode ser excluído, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta dos advogados contrários cadastrados. b. O sistema apresenta a lista de todos os advogados contrários cadastrados, permitindo que o usuário selecione o advogado contrário desejado. c. O sistema mostra os detalhes do advogado contrário selecionado e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de advogado contrário utilizado em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Advogado contrário cadastrado, alterado ou excluído.
Regras do negócio	Não há.

Manter Áreas: inclusão, alteração, exclusão e consulta de áreas	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de áreas. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de uma nova área. b. O sistema apresenta uma tela em branco para o preenchimento dos dados da área. c. O Usuário fornece os dados da nova área. d. O sistema inclui a nova área. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa a área que deseja alterar. b. O usuário altera um ou mais dados referentes a uma área e solicita a sua atualização. c. O sistema altera os dados da área. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa a área que deseja excluir. b. Se a área pode ser excluída, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta das áreas cadastradas. b. O sistema apresenta a lista de todas as áreas cadastradas, permitindo que o usuário selecione a área desejada. c. O sistema mostra os detalhes da área selecionada e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de área utilizada em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Área cadastrada, alterada ou excluída.
Regras do negócio	Não há.

Manter Comarcas: inclusão, alteração, exclusão e consulta de comarcas	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de comarcas. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de uma nova comarca. b. O sistema apresenta uma tela em branco para o preenchimento dos dados da comarca. c. O Usuário fornece os dados da nova comarca. d. O sistema inclui a nova comarca. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa a comarca que deseja alterar. b. O usuário altera um ou mais dados referentes a uma comarca e solicita a sua atualização. c. O sistema altera os dados da comarca. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa a comarca que deseja excluir. b. Se a comarca pode ser excluída, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta das comarcas cadastradas. b. O sistema apresenta a lista de todas as comarcas cadastradas, permitindo que o usuário selecione a comarca desejada. c. O sistema mostra os detalhes da comarca selecionada e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de comarca utilizada em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Comarca cadastrada, alterada ou excluída.
Regras do negócio	Não há.

Manter Processos: inclusão, alteração, exclusão e consulta de processos	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de processos. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de um novo processo. b. O sistema apresenta uma tela em branco para o preenchimento dos dados do processo. c. O Usuário fornece os dados do novo processo. d. O sistema inclui o novo processo <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa o processo que deseja alterar. b. O usuário altera um ou mais dados referentes a um processo e solicita a sua atualização. c. O sistema altera os dados do processo. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa o processo que deseja excluir. b. Se o processo pode ser excluído, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta dos processos cadastrados. b. O sistema apresenta a lista de todos os processos cadastrados, permitindo que o usuário selecione a processo desejado. c. O sistema mostra os detalhes do processo selecionado e retorna ao passo 3. <p>Fluxo (3): Visualizar movimentações</p> <ol style="list-style-type: none"> a. O usuário solicita a visualização das movimentações ocorridas para o processo. b. O sistema apresenta a lista das movimentações.
Fluxo de exceção	<p>Fluxo (3): Exclusão de processo utilizado em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a honorários, movimentações, despesas ou audiências, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Processo cadastrado, alterado ou excluído.
Regras do negócio	Não há.

Manter Usuários: inclusão, alteração, exclusão e consulta de usuários	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de usuários. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de um novo usuário. b. O sistema apresenta uma tela em branco para o preenchimento dos dados do usuário. c. O Usuário fornece os dados do novo usuário. d. O sistema inclui o novo usuário. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa o usuário que deseja alterar. b. O usuário altera um ou mais dados referentes a um usuário e solicita a sua atualização. c. O sistema altera os dados do usuário. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa o usuário que deseja excluir. b. Se o usuário pode ser excluído, o sistema realiza a exclusão; caso contrário, o sistema reporta o fato. <p>Fluxo (3): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a consulta dos usuários cadastrados. b. O sistema apresenta a lista de todos os usuários cadastrados, permitindo que o usuário selecione o usuário desejado. c. O sistema mostra os detalhes do usuário selecionado e retorna ao passo 3.
Fluxo de exceção	<p>Fluxo (3): Exclusão de usuário utilizado em outras funcionalidades</p> <ol style="list-style-type: none"> a. Em caso de tentativa de exclusão de registro vinculado a um processo, não será permitida a exclusão. b. O sistema retorna ao passo 3.
Pós-condições	Usuário cadastrado, alterado ou excluído.
Regras do negócio	Não há.

Manter Audiências: inclusão, alteração e exclusão de audiências	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de audiências. 2. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão e finalização do caso de uso. 3. O usuário executa uma das operações. 4. O caso de uso retorna para o passo 2.
Fluxo alternativo	<p>Fluxo (3): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita a inclusão de uma nova audiência. b. O sistema apresenta uma tela em branco para o preenchimento dos dados da audiência. c. O Usuário fornece os dados da nova audiência. d. O sistema inclui a nova audiência. <p>Fluxo (3): Alteração</p> <ol style="list-style-type: none"> a. O usuário informa a audiência que deseja alterar. b. O usuário altera um ou mais dados referentes a uma audiência e solicita a sua atualização. c. O sistema altera os dados da audiência. <p>Fluxo (3): Exclusão</p> <ol style="list-style-type: none"> a. O usuário informa a audiência que deseja excluir. b. O sistema realiza a exclusão.
Fluxo de exceção	Não há.
Pós-condições	Audiência cadastrada, alterada ou excluída.
Regras do negócio	Não há.

Importar Movimentações: importar movimentações de processos armazenadas no servidor de <i>e-mails</i>	
Pré-condições	<ol style="list-style-type: none"> 1. O usuário está identificado pelo sistema. 2. O sistema possui o cadastro do <i>e-mail</i> remetente das mensagens do Sistema <i>Push</i> e os dados do servidor de <i>e-mail</i> onde as mensagens a serem importadas estão armazenadas. 3. O sistema possui o cadastro dos <i>tokens</i> que delimitam as informações que serão extraídas dos <i>e-mails</i>. 4. O usuário está conectado à <i>internet</i>.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário acessa o sistema pela primeira vez no dia. 2. O sistema questiona se deseja fazer a importação das movimentações processuais. 3. O usuário solicita a importação das movimentações processuais no <i>e-mail</i>. 4. O sistema conecta-se ao servidor de <i>e-mails</i> e faz a atualização das movimentações processuais.
Fluxo alternativo	O usuário executa a importação através de opção do menu do sistema.
Fluxo de exceção	<p>Fluxo (4): Usuário não conectado à internet</p> <ol style="list-style-type: none"> a. O sistema aborta a importação e reporta o fato. <p>Fluxo (4): O processo não está cadastrado no sistema ou a movimentação já foi importada</p> <ol style="list-style-type: none"> a. O sistema ignora a mensagem do <i>e-mail</i> e não grava a movimentação. b. O sistema procura por outra mensagem de <i>e-mail</i>.
Pós-condições	Movimentações, dos processos cadastrados no sistema, armazenadas na base de dados e disponíveis para consulta.
Regras do negócio	Não há.

Inserir Movimentações: inserir movimentações manualmente caso houver algum problema na importação de alguma movimentação dos <i>e-mails</i>	
Pré-condições	1. O usuário está identificado pelo sistema. 2. O processo a ser inserida a movimentação está cadastrado no sistema.
Fluxo principal	3. O usuário informa o processo ou seleciona através da consulta. 4. O usuário informa data e descrição da movimentação e solicita gravação. 5. O sistema atualiza a movimentação na base de dados.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Movimentação inserida na base de dados e disponível para consulta.
Regras do negócio	Não há.

Visualizar Agenda das Movimentações: visualizar as últimas movimentações dos processos ocorridas a partir de uma data informada	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	1. O usuário informa a partir de que data deseja visualizar as movimentações. 2. O sistema visualiza as movimentações para o período selecionado mostrando na ordem decrescente por data.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Visualização das últimas movimentações processuais ocorridas a partir da data informada, em ordem decrescente.
Regras do negócio	Não há.

Visualizar Agenda de Audiências: visualizar agenda de audiências a partir de uma data informada	
Pré-condições	O usuário está identificado pelo sistema.
Fluxo principal	1. O usuário informa a partir de que data deseja visualizar as audiências. 2. O sistema visualiza as audiências para o período selecionado.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Visualização das audiências agendadas a partir da data informada.
Regras do negócio	Não há.

Lançar Honorários de Processos: lançamento de honorários dos processos	
Pré-condições	<ol style="list-style-type: none"> 1. O usuário está identificado pelo sistema. 2. O processo do lançamento está cadastrado no sistema com valor da causa informado.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de lançamentos de honorários. 2. O usuário informa o número do processo e a data de vencimento do honorário. 3. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 4. O usuário executa uma das operações.
Fluxo alternativo	<p>Fluxo (4): Inclusão</p> <ol style="list-style-type: none"> a. O usuário solicita o lançamento de um novo honorário. b. O sistema apresenta a tela para informar o valor do honorário. c. O usuário fornece o valor do lançamento. d. O sistema inclui o lançamento. <p>Fluxo (4): Alteração</p> <ol style="list-style-type: none"> a. O usuário solicita a alteração de um honorário. b. O sistema apresenta o honorário cadastrado. c. O usuário informa novo valor. d. O sistema atualiza o valor do honorário. <p>Fluxo (4): Exclusão</p> <ol style="list-style-type: none"> a. O usuário solicita exclusão de um honorário cadastrado. b. O sistema exclui o honorário. <p>Fluxo (4): Consulta</p> <ol style="list-style-type: none"> a. O usuário solicita a visualização dos honorários do processo. b. O sistema apresenta a lista de honorários lançados para o processo, permitindo que o usuário selecione o lançamento desejado. c. O sistema apresenta os dados do lançamento e retorna para o passo 4.
Fluxo de exceção	Se o honorário já estiver pago, o sistema alerta que o honorário já foi pago e não permite acessar o registro.
Pós-condições	Honorário lançado.
Regras do negócio	A soma dos valores de honorários lançados para um processo não pode ultrapassar o valor da causa informado no cadastro do processo.

Baixar Honorários: baixa de honorários lançados para os processos	
Pré-condições	<ol style="list-style-type: none"> 1. O usuário está identificado pelo sistema 2. O honorário está lançado no sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a baixa de honorários. 2. O usuário informa o número do processo e a data de vencimento do honorário. 3. O sistema carrega o registro. 4. O usuário informa a data do pagamento. 5. O sistema faz a baixa do honorário.
Fluxo alternativo	Não há.
Fluxo de exceção	Se o honorário já foi baixado, o sistema alerta que o honorário já foi baixado e não permite acessar o registro.
Pós-condições	Honorário baixado.
Regras do negócio	Só é permitido baixar o valor integral do honorário. Para baixar um valor menor, o lançamento deve ser excluído e lançado novamente com as devidas parcelas e valores.

Lançar Despesas de Processos: lançamento de despesas de processos	
Pré-condições	<ol style="list-style-type: none"> 1. O usuário está identificado pelo sistema. 2. O processo do lançamento está cadastrado no sistema.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário solicita a manutenção de lançamentos de despesas. 2. O usuário informa o número do processo e o código da despesa. 3. O sistema apresenta as operações que podem ser realizadas: inclusão, alteração, exclusão, consulta e finalização do caso de uso. 4. O usuário executa uma das operações.
Fluxo alternativo	<p>Fluxo (4): Inclusão</p> <ol style="list-style-type: none"> e. O usuário solicita o lançamento de uma nova despesa. f. O sistema apresenta a tela para informar o valor da despesa. g. O usuário fornece a descrição e o valor da despesa. h. O sistema inclui o lançamento. <p>Fluxo (4): Alteração</p> <ol style="list-style-type: none"> e. O usuário solicita a alteração de um lançamento de despesa. f. O sistema apresenta o lançamento cadastrado. g. O usuário altera os dados que deseja. h. O sistema atualiza os dados do lançamento. <p>Fluxo (4): Exclusão</p> <ol style="list-style-type: none"> c. O usuário solicita exclusão de um lançamento cadastrado. d. O sistema exclui o lançamento. <p>Fluxo (4): Consulta</p> <ol style="list-style-type: none"> d. O usuário solicita a visualização dos lançamentos das despesas. e. O sistema apresenta a lista de despesas lançadas para o processo, permitindo que o usuário selecione o lançamento desejado. f. O sistema apresenta os dados do lançamento e retorna para o passo 4.
Fluxo de exceção	Não há.
Pós-condições	Despesa lançada.
Regras do negócio	Não há.

Visualizar Relatório de Clientes: visualizar clientes cadastrados no sistema	
Pré-condições	Não há.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário informa cliente inicial e final. 2. O sistema visualiza os clientes da seleção.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Relatório de clientes visualizado.
Regras do negócio	Não há.

Visualizar Relatório de Advogados: visualizar advogados cadastrados no sistema	
Pré-condições	Não há.
Fluxo principal	<ol style="list-style-type: none"> 1. O usuário informa advogado inicial e final. 2. O sistema visualiza os advogados da seleção.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Relatório de advogados visualizado.
Regras do negócio	Não há.

Visualizar Relatório de Movimentações de Processos: visualizar movimentações dos processos por advogados e por clientes	
Pré-condições	Não há.
Fluxo principal	1. O usuário escolhe opção do relatório por cliente ou por advogado e informa o intervalo de código e período desejados. 2. O sistema visualiza as movimentações da seleção.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Relatório de movimentações visualizado.
Regras do negócio	Não há.

Visualizar Relatório de Honorários Pagos e Pendentes: visualizar honorários pagos e pendentes	
Pré-condições	Não há.
Fluxo principal	1. O usuário escolhe opção do relatório por honorários pagos ou pendentes e informa o intervalo de clientes e período desejados. 2. O sistema visualiza os honorários da seleção.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Relatório de honorários visualizado.
Regras do negócio	Não há.
Manter Configuração de E-mails: configurar <i>e-mail</i> monitorado e <i>e-mail</i> do Sistema <i>Push</i> que envia as movimentações	
Pré-condições	Não há.
Fluxo principal	1. O usuário informa campos que deseja alterar. 2. O sistema atualiza campos na base de dados.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	Dados dos <i>e-mails</i> atualizados.
Regras do negócio	Não há.

Manter Configuração de Tokens: configurar <i>tokens</i> que delimitam os dados a serem importados dos <i>e-mails</i>	
Pré-condições	Não há.
Fluxo principal	1. O usuário informa os <i>tokens</i> que deseja alterar. 2. O sistema atualiza os <i>tokens</i> na base de dados.
Fluxo alternativo	Não há.
Fluxo de exceção	Não há.
Pós-condições	<i>Tokens</i> atualizados.
Regras do negócio	Não há.

APÊNDICE B – Dicionário de dados

Advogados

Nome da Coluna	Descrição	Tipo	PK	FK
adv_codigo	Código	SERIAL	Yes	No
adv_nome	Nome	VARCHAR(40)	No	No
adv_oab	OAB	VARCHAR(10)	No	No
adv_cpf	CPF	VARCHAR(14)	No	No
adv_rg	RG	VARCHAR(9)	No	No
adv_endereco	Endereço	VARCHAR(40)	No	No
adv_bairro	Bairro	VARCHAR(30)	No	No
adv_cidade	Cidade	VARCHAR(30)	No	No
adv_estado	Estado	VARCHAR(2)	No	No
adv_cep	CEP	VARCHAR(9)	No	No
adv_fone	Fone	VARCHAR(14)	No	No
adv_email	E-mail	VARCHAR(30)	No	No
adv_tp_advogado	Tipo de advogado	VARCHAR(1)	No	No

Áreas

Nome da Coluna	Descrição	Tipo	PK	FK
are_codigo	Código	SERIAL	Yes	No
are_nome	Nome	VARCHAR(20)	No	No

Clientes

Nome da Coluna	Descrição	Tipo	PK	FK
cli_codigo	Código	SERIAL	Yes	No
cli_nome	Nome	VARCHAR(40)	No	No
cli_apelido	Apelido	VARCHAR(40)	No	No
cli_cpf_cnpj	CPF/CNPJ	VARCHAR(18)	No	No
cli_rg	RG	VARCHAR(09)	No	No
cli_ie	Inscrição estadual	VARCHAR(12)	No	No
cli_im	Inscrição municipal	VARCHAR(12)	No	No
cli_endereco	Endereço	VARCHAR(40)	No	No
cli_bairro	Bairro	VARCHAR(30)	No	No
cli_cidade	Cidade	VARCHAR(30)	No	No
cli_estado	Estado	VARCHAR(02)	No	No
cli_cep	CEP	VARCHAR(09)	No	No
cli_fone	Fone	VARCHAR(14)	No	No
cli_fax	Fax	VARCHAR(14)	No	No
cli_email	E-mail	VARCHAR(30)	No	No
cli_observacoes	Observações	VARCHAR(400)	No	No

Comarcas

Nome da Coluna	Descrição	Tipo	PK	FK
com_codigo	Código	SERIAL	Yes	No
com_nome	Nome	VARCHAR(30)	No	No
com_vara	Vara	VARCHAR(30)	No	No

Partes contrárias

Nome da Coluna	Descrição	Tipo	PK	FK
pac_codigo	Código	SERIAL	Yes	No
pac_nome	Nome	VARCHAR(40)	No	No
pac_cpf_cnpj	CPF/CNPJ	VARCHAR(18)	No	No
pac_rg	RG	VARCHAR(09)	No	No
pac_endereco	Endereço	VARCHAR(40)	No	No
pac_bairro	Bairro	VARCHAR(30)	No	No
pac_cidade	Cidade	VARCHAR(30)	No	No
pac_estado	Estado	VARCHAR(02)	No	No
pac_cep	CEP	VARCHAR(09)	No	No
pac_fone	Fone	VARCHAR(14)	No	No
pac_fax	Fax	VARCHAR(14)	No	No
pac_email	E-mail	VARCHAR(30)	No	No
pac_observacoes	Observações	VARCHAR(400)	No	No

Usuários

Nome da Coluna	Descrição	Tipo	PK	FK
usu_usuario	Usuário	VARCHAR(15)	Yes	No
usu_senha	Senha	VARCHAR(12)	No	No

Parâmetros

Nome da Coluna	Descrição	Tipo	PK	FK
par_codigo	Código	INTEGER	Yes	No
par_pop3	POP3	VARCHAR(30)	No	No
par_username	Usuário	VARCHAR(30)	No	No
par_password	Senha	VARCHAR(15)	No	No
par_email_remetente	E-mail do remetente	VARCHAR(40)	No	No
parTk_processo_ini	Token inicial delimitador do processo	VARCHAR(20)	No	No
parTk_processo_fin	Token final delimitador do processo	VARCHAR(20)	No	No
parTk_data_ini	Token inicial delimitador da data	VARCHAR(20)	No	No
parTk_data_fin	Token final delimitador da data	VARCHAR(20)	No	No
parTk_mov_ini	Token inicial delimitador moviment.	VARCHAR(20)	No	No
parTk_mov_fin	Token final delimitador moviment.	VARCHAR(20)	No	No
parTk_link_ini	Token inicial delimitador do link	VARCHAR(20)	No	No
par_delimitador	Token delimitador de linhas	VARCHAR(5)	No	No
par_dt_ult_atualiz	Data da última atualização moviment.	DATE	No	No

Processos

Nome da Coluna	Descrição	Tipo	PK	FK
pro_nr_processo	Número do processo	VARCHAR(15)	Yes	No
pro_advogado	Código do advogado	INTEGER	No	Yes
pro_requerente	Código do cliente	INTEGER	No	Yes
pro_requerido	Código da parte contraria ao cliente	INTEGER	No	Yes
pro_area	Código da área	INTEGER	No	Yes
pro_comarca	Código da comarca	INTEGER	No	Yes
pro_advogado_contrario	Código do advogado contrário	INTEGER	No	Yes
pro_data_entrada	Data de entrada do processo	DATE	No	No
pro_vl_causa	Valor da causa	NUMERIC(10,2)	No	No
pro_link	Link das movimentações do processo	VARCHAR(150)	No	No

Honorários

Nome da Coluna	Descrição	Tipo	PK	FK
hon_nr_processo	Número do processo	VARCHAR(15)	Yes	Yes
hon_dt_vencdo	Data de vencimento	DATE	Yes	No
hon_dt_pagto	Data do pagamento	DATE	No	No
hon_vl_lancto	Valor do lançamento	NUMERIC(10,2)	No	No
hon_vl_pagto	Valor pago	NUMERIC(10,2)	No	No

Despesas dos processos

Nome da Coluna	Descrição	Tipo	PK	FK
des_nr_processo	Número do processo	VARCHAR(15)	Yes	Yes
des_codigo	Código da despesa	INTEGER	Yes	No
des_descricao	Descrição da despesa	VARCHAR(50)	No	No
des_vl_despesa	Valor da despesa	NUMERIC(10,2)	No	No

Movimentações

Nome da Coluna	Descrição	Tipo	PK	FK
mov_nr_processo	Número do processo	VARCHAR(15)	Yes	Yes
mov_data_mov	Data da Movimentação	DATE	Yes	No
mov_sequencia	Seqüência	INTEGER	Yes	No
mov_descricao	Descrição da movimentação	VARCHAR(200)	No	No

Audiências

Nome da Coluna	Descrição	Tipo	PK	FK
aud_nr_processo	Numero do processo	VARCHAR(15)	Yes	Yes
aud_dt_audiencia	Data da audiência	DATE	Yes	No
aud_hr_audiencia	Hora da audiência	VARCHAR(5)	Yes	No
aud_nm_acao	Nome da ação	VARCHAR(50)	No	No
aud_tp_audiencia	Tipo de audiência	VARCHAR(30)	No	No