
Sistema de Gerenciamento de Fretes para Motoristas Autônomos

Gustavo Silva Melo

PROJETO DE CONCLUSÃO DE CURSO

Data de Depósito: 21/06/2016

Assinatura: _____

Sistema de Gerenciamento de Fretes para Motoristas Autônomos

Gustavo Silva Melo

Saulo Alves Grego

Monografia apresentada ao Instituto Superior de Educação do UNIFOR/MG, como requisito parcial para obtenção do título de bacharel em Ciência da Computação, sob a orientação do Profº. Saulo Alves Grego.

Unifor-MG - Formiga
06/2016

Aos meus pais.

Agradecimentos

Ao professor e orientador Saulo Alves Grego, pela grande ajuda na conclusão deste projeto e ao decorrer de toda graduação, por não medir esforços e prover todo o auxílio necessário.

A minha família pelo apoio dado durante toda a fase de graduação.

Aos amigos que sempre estiveram do meu lado, e me deram força pra seguir e chegar até o final desta etapa.

“Você não pode abrir mão de algo que você realmente acredita por razões financeiras. Se você morrer na beira da estrada que assim seja. Mas pelo menos você sabe que tentou.”

Robert Plant

Sumário

1	Introdução	1
1.1	Considerações Iniciais	1
1.2	Objetivos do Trabalho	2
1.3	Estrutura da Monografia	2
2	Fundamentação Teórica	3
2.1	Considerações Iniciais	3
2.1.1	Sistema de informação gerencial	3
2.1.2	Enterprise resource planning (ERP)	4
2.1.3	PostgreSQL	4
2.1.4	Padrão Data Access Object	5
2.1.5	Hibernate	5
2.1.6	Modelo Model, View, Controller)	6
2.1.7	JavaServer Faces	6
2.1.8	Primefaces	7
2.1.9	DbWrench	7
2.1.10	JasperReports e iReport	7
2.2	Considerações Finais	8
3	Estudo da Arte	9
3.1	Considerações Iniciais	9
3.2	Trabalhos Relacionados	9
3.3	Considerações Finais	11
4	Metodologia e Desenvolvimento	13
4.1	Considerações Iniciais	13
4.2	Fases do Desenvolvimento	13
4.2.1	Descrição do Problema	13
4.2.2	Levantamento de Requisitos	15

4.2.3	Diagrama de Entidade Relacionamento (DER)	16
4.3	Desenvolvimento da Aplicação	16
4.3.1	Criação do Banco de dados	16
4.3.2	Conexão entre banco de dados e aplicação	17
4.3.3	Interface	18
4.3.4	Estrutura da aplicação seguindo os modelos MVC e DAO	18
4.3.5	Autenticação do Usuário	19
4.3.6	Mensagens do sistema	19
5	Funcionamento e Resultados	21
5.1	Considerações Iniciais	21
5.2	Sistema de Gerenciamento de fretes	21
5.2.1	Tela de Login	21
5.2.2	Tela Principal	22
5.2.3	Cadastros	23
5.2.4	fretes	26
5.3	Relatórios personalizados	28
5.4	Resultados	28
6	Trabalhos Futuros	31
6.1	Controle de manutenções	31
6.2	Orçamentos	31
7	Conclusões	33
7.1	Considerações Finais	33
A	Diagrama de entidade relacionamento	37
B	Relatórios de relação de clientes	39
C	Relatório de resumo de fretes	41
D	Questionário de avaliação	43

Resumo

MELO, G. S. *Sistema de Gerenciamento de Fretes*. Monografia (Graduação) — Centro Universitário de Formiga – Unifor-MG – Formiga, 2015.

No cenário atual, tornou-se cada vez mais importante um controle automatizado das informações de qualquer empreendimento. A informação tem sido cada vez mais valorizada, sendo de suma importância ter um controle seguro e um processamento eficiente, dando ao usuário segurança na análise para uma tomada de decisões. Devido a essa necessidade, surgiram os sistemas de informação gerenciais. São sistemas que fornecem informações necessárias para uma eficácia na gerência de uma organização. É um software que provê auxílio na tomada de decisões, disponibilizando as informações corretas para o usuário em tempo hábil. É uma parte integrante de estratégias empresariais, pois a comunicação e as informações são uma necessidade para qualquer organização. O projeto em questão, pode ser considerado como um sistema de informação gerencial, possuindo gerenciamentos de alto nível na manutenção de cadastro clientes e produtos, controle de fretes, ficha completa de veículos e motoristas. Foram desenvolvidos para a aplicação, vários relatórios visando um controle preciso nos processos efetuados pelos usuários do sistema, tornando-o completo e totalmente eficiente sobre o problema proposto.

Palavras-chave: Gerenciamento de Viagens, Java, PostgreSQL, JSF, Primefaces, Hibernate

Abstract

MELO, G. S. *System of Travel Management*. Monografia (Graduação) — Centro Universitário de Formiga – Unifor-MG – Formiga, 2015.

In the actual scenario, it becomes more important an automated control of the information in any enterprise. The information has been increasingly valued and it is of utmost importance to have a safe control and an efficient process of it, giving to the user the safety in the analysis to take decisions. Due to this necessity emerged the management information system. They are systems that give the necessary informations to a more effectiveness in the organization's management. It is a software that provides help on decisions-making, giving the correct informations to the user timely. It is an integrated part of business strategies, because communication and information are a necessity for any organization. The project in question can be considered a management information system, having high-level managements in the maintenance of the register of clients and products, freight control, complete vehicles record and drivers. Were developed for the application many reports in order to have a precise control of the processes performed by the system users making it complete and totally efficient for the proposed problem.

Palavras-chave: Travel management, Java, PostgreSQL, JSF, Primefaces, Hibernate

Introdução

O objetivo deste capítulo é mostrar ao leitor o contexto da idealização do projeto, e também os objetivos principais da aplicação em cumprir suas pretensões.

1.1 Considerações Iniciais

Sistemas de informação gerenciais (SIG) são aliados à gestão empresarial no ato de tomadas de decisões. A maneira como os dados são processados tornando-se informação é importante para que a decisão seja tomada conforme o ambiente em que é proposta. A importância do uso de sistemas de informação gerenciais vai além de coletar dados e transformá-los em informações. A qualidade da informação é uma característica que torna produtos e serviços mais valiosos para a organização.

A informação deve ser bem administrada e disposta, de acesso rápido e fácil para seu usuário, pois quanto mais próximo da informação, melhor será o tempo para análise e posteriormente efetuar uma tomada de decisão.

Um sistema de gerenciamento de fretes pode ser descrito como um sistema de informação gerencial. É um conjunto de ferramentas e operações referentes ao armazenamento e processamento de informações ligadas a um serviço de um empreendedor que presta serviços de fretes aos seus clientes.

O sistema consiste em um cadastro de informações necessárias para o gerenciamento das fretes feitas pelo empreendedor, como locais de carga e descarga, cadastro completo de clientes, produtos transportados, veículos, motoristas entre outros. O conjunto dessas informações previamente cadastradas dá o resultado final que é a viagem efetivada.

A proposta deste trabalho é oferecer um sistema para automação do controle das informações do usuário, onde foi vista uma necessidade de um gerenciamento automatizado devido a quantidade de dados que eram controlados manualmente, ajudando no resgate, organização

e preservação dos dados que anteriormente eram feitos por meios de anotações manuais em cadernos, onde muitos cálculos eram feitos erroneamente ou duplicados, havia uma dificuldade de resgate de informações e não havia um padrão de organização.

Durante o desenvolvimento, foi feito um contato direto com os usuários que seriam contemplados com a aplicação, levantando requisitos, fazendo testes e o uso da aplicação, para que o resultado final seja o mais fiel possível às necessidades. O foco do trabalho é o desenvolvimento de uma aplicação simples, de fácil uso e que atenda totalmente o que for preciso para o usuário.

1.2 Objetivos do Trabalho

O objetivo é oferecer um sistema de gerenciamento que atenda a grande parte das necessidades dos usuários para que foram desenvolvidos o sistema. A proposta é automatizar os seus registros, dando um maior controle e visão de suas informações. Com um controle automatizado do seu trabalho, pode ser poupado tempo com gerenciamento, informações mais precisas e rápidas, podendo solucionar falhas na gestão das informações feitas de formas manuais.

É de grande interesse também, o feedback dos usuários do sistema, que possuem posições diferentes diante a inovação, sendo resistentes a automação ou totalmente adeptos a ela. A maneira que será recepcionada a aplicação é extremamente relevante para o desenvolvedor.

Para fins acadêmicos, foram utilizadas várias tecnologias diferentes em conjunto, de forma a aprimorar o trabalho e aprofundar o conhecimento em cada uma delas. Foram associadas várias ferramentas, cada qual com sua função específica como front-end, gerenciamento de banco de dados, linguagem de programação, interligação entre banco e aplicação e etc.

1.3 Estrutura da Monografia

Esta monografia possui a seguinte estrutura: No capítulo 2 é mostrado um estudo para a seleção e uso das tecnologias envolvidas para o desenvolvimento do projeto. No capítulo 3 é feito um estudo a respeito de tecnologias já existentes para a solução do mesmo problema proposto para este trabalho. Para uma demonstração da maneira qual foi desenvolvida a aplicação, o capítulo 4 mostra quais foram os passos seguidos. Os resultados obtidos, são exibidos no capítulo 5 de acordo com o *feedback* dos usuários. Já o capítulo 6, mostra quais serão os trabalhos futuros para complementar o software desenvolvido. Por último, o capítulo 7 mostra um breve resumo das conclusões tomadas após a finalização do projeto.

Fundamentação Teórica

Nesta seção serão apresentadas as fundamentações teóricas deste trabalho, tais como ferramentas usadas. Serão descritas as ferramentas e suas características, juntamente com as técnicas que foram utilizadas para realização do projeto.

2.1 Considerações Iniciais

A primeira fase do projeto, consiste em escolher quais serão as tecnologias aplicadas no desenvolvimento. A seleção correta das ferramentas implica diretamente no seu desenvolvimento e conseqüentemente no seu resultado final. Deve ser observado o objetivo do projeto, a familiaridade do desenvolvedor com as ferramentas a serem utilizadas, a disponibilidade de material de estudo disponível, performance, adequação a aplicação dentre outros. As próximas seções serão dedicadas às tecnologias empregadas neste projeto.

2.1.1 Sistema de informação gerencial

O sistema de gerenciamento de fretes, devido as suas funções, pode ser enquadrado em um tipo de sistema de informação denominado como sistema de informação gerencial (SIG).

Um SIG é um tipo de sistema que dá suporte às funções de planejamento, controle e organização. Fornece informações seguras e em tempo hábil para que o usuário seja auxiliado nas tomadas de decisões. O SIG é representado por um conjunto de subsistemas trabalhando de forma integrada capaz de gerar informações que podem ser necessárias no processo de tomada de decisão (DAYANA, 2008).

O propósito básico de um SIG é dar auxílio para o alcance de metas, provendo aos administradores detalhes sobre as operações regulares da organização, de maneira que seja possível controlar, organizar e fazer um planejamento mais eficiente (BAZZOTTI; GARCIA, 2007). Pode ser definido como um conjunto de tecnologias que disponibilizam os meios necessários para o processamento inteligente dos dados disponíveis. É voltado para coleta, armazenagem,

recuperação e processamento de informações necessárias pelo profissional na execução das suas atividades. É o processo de transformação de dados brutos em informações processadas que são utilizadas na estrutura decisória da empresa, proporcionando a sustentação administrativa para uma otimização dos resultados esperados (BATISTA, 2006).

Um SIG é desenvolvido para atender as necessidades específicas de cada projeto, como no caso deste trabalho, um sistema para fazer o controle automatizado de viagens.

2.1.2 Enterprise resource planning (ERP)

Um ERP (*enterprise resource planning*) é apresentado como uma solução para a maioria dos problemas gerenciais. É um sistema genérico capaz de integrar todas as informações que giram dentro da empresa em uma única base de dados. A implantação de um sistema ERP afeta a empresa em todas as suas dimensões, sejam culturais ou tecnológicas. Todo o controle da empresa é feito por meio do sistema.

Na adoção de um sistema ERP o objetivo não é somente colocar em produção o software desenvolvido, mas sim uma reformulação dos processos da empresa usando a tecnologia da informação. Vai além da mudança tecnológica, implicando um processo de mudança organizacional. A implantação compreende o processo de adoção do ERP envolvendo seleção, aquisição e testes, onde o sistema foi previamente planejado, passando por uma análise de requisitos da empresa onde o sistema deve estar de acordo com a orientação estratégica da mesma. O sucesso da implantação depende de um alinhamento entre o software, cultura, objetivos e negócios da empresa (MENDES; FILHO, 2002).

Os resultados esperados com a adoção de um ERP são perceptíveis após certo tempo de uso. Uma das vantagens de um ERP, é a integração de vários módulos informatizados que antes eram executados de forma separada. Dessa maneira, a empresa deixa de operar como se fossem várias ilhas informatizadas, porém independentes, mudando para um modelo integrado melhorando a utilização dos recursos internos trazendo economia para a empresa.

Outra vantagem observada, é a documentação e contabilização de documentos definindo bem as regras de negócios permitindo que pontos críticos sejam controlados mais rigorosamente. Pode também ser citado, principalmente para empresas de pequeno e médio porte, uma oportunidade de subir na escala tecnológica (MENDES; FILHO, 2002).

A implantação de um ERP contribui para que se tenha um controle maior sobre todas as informações recolhidas. Com uma base centralizada de dados, todos têm acesso as informações e os dados somente precisam ser inseridos uma vez. Isso proporciona confiabilidade e integridade do sistema, desde que os dados estejam atualizados e de acordo com a realidade (MENDES; FILHO, 2002).

2.1.3 PostgreSQL

Um sistema de gerenciamento de banco de dados (SGBD) é um sistema computadorizado de manutenção de registros. O banco de dados no entanto, pode ser comparado a um armário de arquivos, ou seja, é um repositório computadorizado para uma coleção de arquivos de informações (DATE, 2004). Para este trabalho, foi escolhido o SGBD *PostgreSQL*. O *PostgreSQL*

é um sistema gerenciador de banco de dados de código fonte aberto. Possui mais de 15 anos de desenvolvimento ativo e uma arquitetura com forte reputação de confiabilidade, integridade de dados e conformidade a padrões.

Opera em todos os grandes sistemas operacionais, tais como *Windows, GNU/Linux, Unix (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64)*. Tem suporte completo a *triggers, functions, procedures*, chaves estrangeiras e junções. Inclui a maior parte dos tipos de dados do ISO SQL:1999, suportando também armazenamento de objetos binários e mídias como figuras, sons ou vídeos (PostgreSQL, 2016).

2.1.4 Padrão Data Access Object

O padrão data access object (DAO) é um padrão incluído no ambiente Java JEE para tornar mais simples e desacoplar a interação das aplicações Java com a API JDBC. Todas as comunicações feitas com o mecanismo de persistência são feitas por um mediador DAO. Esse objeto faz o mapeamento das informações transportadas em objetos para instruções da API de persistência (neste trabalho, o *Hibernate*) e mapeia os resultados obtidos por ela de volta aos mesmos objetos de transporte.

Toda a lógica de execução e mapeamento das instruções é feito dentro do objeto DAO, isolando desta maneira da API de persistência completamente. O objeto DAO é responsável pela operação do mecanismo de persistência em nome da aplicação, executando as operações com o banco de dados, que são invocadas diretamente passando a objetos com as informações a serem processadas (Taboarda, Sérgio, 2016).

2.1.5 Hibernate

O *Hibernate* é um projeto de *framework*, que visa ser uma solução completa para o problema de gerenciamento de dados persistentes em Java. Ele faz a mediação da interação do aplicativo com um banco de dados relacional, deixando livre o desenvolvedor para se concentrar nas regras de negócio. No *Hibernate*, não é obrigatório seguir muitas regras específicas e padrões de projeto ao escrever sua lógica de negócios e classes persistentes, sendo assim, o *Hibernate* integra sem maiores problemas com a maioria das novas aplicações existentes e não requer muitas alterações ao resto da aplicação (BAUER; KING, 2005).

O *Hibernate* realiza o mapeamento do objeto relacional escrito em Java, onde o seu principal objetivo é diminuir o grau de complexidade no desenvolvimento de aplicações que trabalham com banco de dados relacional, realizando a intermediação entre a aplicação e o banco de dados, dando liberdade ao programador de trabalhar com menos instruções SQL para recuperar ou inserir dados da sua aplicação. O *Hibernate* faz o mapeamento do objeto relacional, onde as tabelas do banco de dados são representadas em formas de classes na aplicação, e as operações de interação com o banco de dados são realizadas através dos métodos do *Hibernate*.

O uso do *Hibernate* possibilita grande facilidade ao programador, além de diminuir o uso de sentenças SQL, provê a facilidade de migração para a maioria de SGBDs disponíveis no mercado, modificando apenas as configurações do *framework*, não tendo a necessidade de alterar linhas de código dentro da aplicação. Também facilita o desenvolvimento, por trabalhar de forma

orientada a objetos, detecta qualquer alteração feita na estrutura do banco de dados, gerando as alterações nas classes da aplicação. Realiza *cache* nas operações, deixando a aplicação mais rápida, além de ser um *framework* totalmente gratuito (Souza, Nailson, 2012).

2.1.6 Modelo Model, View, Controller)

O sistema foi desenvolvido com o padrão model, view, controller (MVC) juntamente com o DAO. O MVC é um padrão de desenvolvimento muito útil para arquitetura de sistemas web. No MVC a entrada do usuário, a modelagem e a parte visual são totalmente separados e tratados por três tipos de objeto, cada um especializado na sua tarefa.

No *Model*, é administrado o comportamento dos dados, o domínio da aplicação, a resposta a requisições e a instruções. A *View* representa a parte visual, as páginas que serão exibidas e onde o usuário irá inserir as informações. Finalmente no *Controller* é onde são tratadas as regras de negócio, os métodos implementados, tratando a entrada e saída de dados (BURBECK, 1992).

Utilizar o modelo MVC traz muitos benefícios para o desenvolvimento da aplicação. As camadas são bem definidas, facilitam a manutenção e o entendimento do código, pois cada camada tem sua função específica. Novas funcionalidades são adicionadas facilmente e novas características podem ser adicionadas rapidamente. Também permite que os desenvolvedores trabalhem simultaneamente em camadas separadas, onde uma não afeta a outra diretamente.

A figura 2.1 ilustra o como é o funcionamento do modelo MVC.

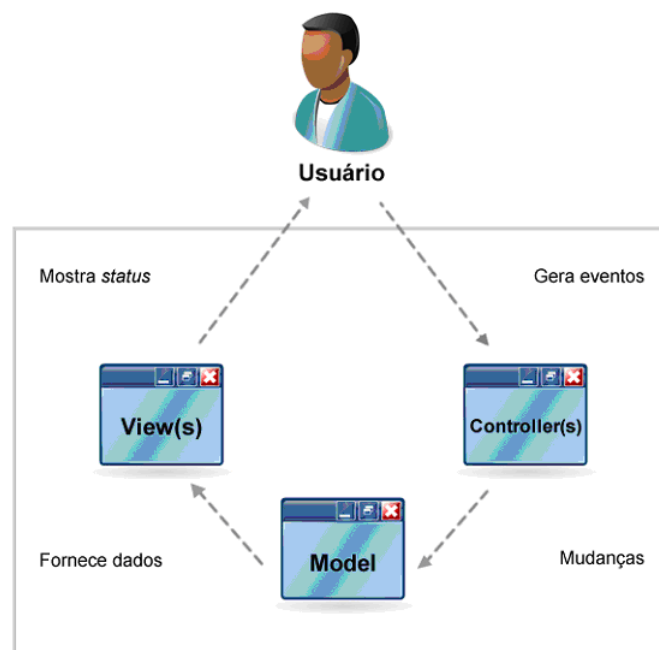


Figura 2.1: Modelo MVC

2.1.7 JavaServer Faces

JavaServer Faces (JSF) é uma tecnologia para aplicações web em Java, construída com a experiência adquirida com o *JavaServer Pages* (JSP), e numerosas aplicações web e *frameworks*

(BERGSTEN, 2001). É um modelo de programação dirigido a eventos, abstraindo os detalhes da manipulação dos eventos e organização dos componentes, o que permite o programador se concentrar na lógica de programação. O JSF incorpora características de um *framework* MVC para web. No JSF, o controle é constituído pelo *FacesServlet*, arquivos de configuração e um conjunto de manipuladores de ações e observadores de eventos. O *FacesServlet* é o responsável por receber as requisições web, redirecionando-as para o modelo e remetendo uma resposta (PITANGA, 2008).

2.1.8 Primefaces

O *Primefaces* é uma ferramenta que possibilita implementar uma interface gráfica para sistemas web que pode ser denominada como uma interface rica. É um *framework* *opensource* utilizado junto a tecnologia JSF no desenvolvimento de aplicações web. É uma biblioteca de componentes diversos que compões a parte *front-end* do sistema (MOENSTER; BONETTI,). O *Primefaces* possui mais de cem componentes dos mais diversos tipos, como entrada de texto, botões, tabelas, painéis, menus, dentre outros, podendo trabalhar com aplicações web comuns até componentes *touchscreen*. O *Primefaces* também possibilita que a aplicação tenha um *layout* responsivo de maneira simples e muito eficiente (Primefaces.org, 2016).

2.1.9 DbWrench

DbWrench é uma ferramenta interessante e muito útil no processo de documentação de um sistema. Consiste em uma aplicação que se conecta diretamente no banco de dados já criado, gerando o diagrama de entidade relacionamento da aplicação. É uma ferramenta multiplataforma, podendo ser usada em sistemas operacionais *Windows*, *Linux* e também *MAC*. Possui suporte para alguns SGBDs conhecidos, como *MySQL*, *Oracle*, *SqlServer* e *PostgreSQL* (Dbwrench, 2016).

Uma grande vantagem do uso dessa ferramenta, é ter sempre a documentação do banco de dados em dia. No ato do desenvolvimento, o programador pode ter que fazer algumas pequenas alterações na estrutura do banco de dados. A ferramenta atualiza o diagrama de entidade relacionamento em poucos segundos, deixando sempre atualizada a documentação do banco, mostrando tabelas, campos, chaves estrangeiras, tipos de dados, tamanho de campos, chaves primárias e todas as informações essenciais.

2.1.10 JasperReports e iReport

O *JasperReports* é um *framework* utilizado para geração de relatórios. É uma ferramenta de fonte aberta e gratuita, podendo ser utilizada livremente. As principais funcionalidades do *JasperReports* são de exportar relatórios para diversos formatos diferentes como PDF, XML, HTML etc. Aceita também vários tipos de dados, conexão com banco de dados, sessão do *Hibernate* ou coleção de objetos da memória. Permite também o uso de diagramas, gráficos e até código de barras (Martins, Marcelo, 2010). A versão utilizada para o desenvolvimento, foi a 5.0.6.

Utilizado em conjunto com o *JasperReport*, foi desenvolvido o *iReport*, que é uma ferramenta

que possibilita a montagem de relatórios de forma gráfica, permitindo que o relatório seja montado através de uma palheta de opções, com uma interface amigável e um sistema onde o usuário usa o sistema de “arrasta e solta” para a montagem visual do relatório. Essa função põe fim a um problema, que era a dificuldade em se escrever um *layout* totalmente em XML (Martins, Marcelo, 2010).

A ferramenta também possui um conjunto de *templates* que trazem modelos já prontos e também possibilita que sejam criados temas personalizados (Martins, Marcelo, 2010).

2.2 Considerações Finais

Um dos principais objetivos deste trabalho, além de atender a necessidade do usuário, é o uso de tecnologias diferentes associadas em um projeto, tendo cada uma a sua função específica explorada e aprofundada para um resultado comum ao final do projeto. As tecnologias usadas podem ser consideradas atuais, sendo ainda utilizadas e aprimoradas em suas respectivas comunidades, existindo sempre uma versão atual para seu uso, o que facilita a busca por documentação e tutoriais de uso. Outro ponto positivo, é a possibilidade de implementar novas funcionalidades com facilidade, já que, sempre existem novas ferramentas a serem utilizadas. O uso de tecnologias modernas torna o sistema contemporâneo, tanto na sua interface quanto no seu uso.

Estudo da Arte

Este capítulo irá apresentar os ganhos do gerenciamento de fretes automatizado, e também o estudo da arte, feito em busca de ferramentas similares disponíveis no mercado.

3.1 Considerações Iniciais

Atualmente o gerenciamento e controle de informações têm sido informatizado nos mais amplos campos de atuação. Desde o uso pessoal, até grandes Sistemas ERP, que podem controlar um grupo de empresas enorme. Utilizar um sistema automatizado para o armazenamento e processamento das informações, é uma solução quase que inevitável para todos os tipos de negócio por diversos motivos. Os ganhos são vários, como integração de processos e informações, acompanhamento de tendência, maior agilidade, confiabilidade e segurança entre outros.

Sistemas para gerenciamento de fretes podem ser encontrados com facilidade na internet, porém, integrados em grandes ERPs ou voltado para grandes empresários, preparados para controles financeiros complexos, emissão de documentos fiscais, controle contábil e fiscal. O maior problema, é o custo de implantação desses sistemas, que para um trabalhador autônomo, foge da realidade.

Esta aplicação tem como o foco, àquele que é o dono do próprio pequeno negócio, mas deseja ter um controle automatizado de suas informações, deixando de lado anotações manuais, que por muitas vezes podem ser falhas ou inseguras.

O projeto foi desenvolvido pensando nesse tipo de usuário, de forma simples, auto explicativa, mas com o objetivo final bem definido.

3.2 Trabalhos Relacionados

É importante ressaltar, que ao fazer o estudo de trabalhos relacionados, podem ser abrangidos vários trabalhos onde um sistema de informação é implantado com o intuito de fazer a

gestão, armazenamento e processamento de dados, transformando-as em informações refinadas para o usuário, dando auxílio na administração e tomada de decisões dentro de uma organização. A utilidade sempre será a mesma, porém em cenários diferentes, cada um atendendo a uma atividade específica exercida pela organização que irá fazer o uso do sistema.

O estudo de (FESTA; ASSUMPÇÃO, 2012) é uma análise do impacto do desempenho logístico com o uso de tecnologia da informação para apoio em gerenciamento de transportes. Foi um estudo realizado com base em dados coletados por meio de estudos de caso com operadores logísticos que fazem transporte de eletrodomésticos da Zona Franca de Manaus a São Paulo. Foi observado, em duas transportadoras com mais de 30 anos de mercado, o quanto o uso do sistema de gerenciamento de transportes colabora pela melhoria do serviço prestado, focando nos fluxos de operações operacionais para controle gerencial.

O sistema oferece condições para um melhor controle da produtividade no uso dos ativos físicos da transportadora e a racionalidade de sua equipe de gestão, reduzindo o tempo gasto no planejamento e montagem de carga e também na programação de entregas, incluindo também a negociação dos contratos. A melhoria foi possível pelo controle de suas operações pelo uso de dados precisos de frete, cliente, frota, viagem em relação às condições contratadas. Um ponto importante notado foi a confiabilidade na transmissão de informações que é garantida pelo uso de troca de informações eletronicamente e também de documentos, agilizando a liberação de mercadorias.

O sistema BSoft, é um exemplo de ERP específico, para controle de transportes que oferece domínio sobre qualquer movimentação necessária para administrar uma empresa. O sistema conta com os módulos essenciais para uma empresa que presta serviço de transportes, sendo um sistema completo e abrangente, atendendo os níveis operacionais e gerenciais, suprimindo também as necessidades fiscais. Trata-se de um ERP e deve ser contratado para que seja feito o uso, mas é possível fazer o *download* de uma versão para avaliação.

O sistema conta com controles completos e específicos para uma empresa de transportes, como: controle de rotas, cadastro completo de veículos, controle de pneus, despesas, acertos financeiros entre outros. Uma função essencial para uma empresa de transportes, é a emissão de documentos eletrônicos, como notas fiscais, conhecimentos de transporte eletrônicos e manifesto de documentos fiscais (BSoft, 2016).

Pode ser citado, o sistema comercial Delphi SGA como um ERP preparado para controle de fretes. É um sistema de informação gerencial de grande porte, abrangendo todo o controle de uma empresa. Dentre as diversas funcionalidades, o sistema possui um controle de fretes efetuados pelos veículos da empresa transportadora. Por se tratar de um sistema contábil e fiscal, todo o controle de fretes é feito com base na emissão de conhecimento de transporte eletrônico (CTe), que é um documento exigido por lei, para que uma empresa transporte de cargas de qualquer mercadoria por transporte terrestre

No ato da emissão do CTe, o sistema armazena informações essenciais para o controle de uma viagem, como: origem, destino, carga, informações do veículo e motorista, quantidade e também valores. É gerado um lançamento financeiro, onde podem ser controlados o que foi

recebido e o que há para receber pela empresa. O sistema proporciona, dessa maneira, todas as informações necessárias para que a administração faça o controle de suas operações (Delphisga, 2016).

Em aplicativos móveis, para realizar a função específica deste trabalho, que é o gerenciamento de fretes, foi encontrado apenas um aplicativo com uma função similar a este projeto. A função do aplicativo é apenas o cálculo da viagem com base da distância, consumo, quantidade e valor do combustível. Não existe armazenamento de dados e nem controle cadastral de clientes e fretes efetuados (APP4ALL, 2016).

3.3 Considerações Finais

É fato que, quando analisados os trabalhos sobre resultados de implantações de sistemas de informação, o ganho organizacional de uma empresa é notável, devido à organização e facilidade na recuperação das informações. Nota-se também, em pesquisa, que não existem ferramentas que trabalhem especificamente como a desenvolvida neste trabalho. Quando se procura um sistema para exercer essa função, grande parte dos resultados são ERPs, voltados para empresas e com controles além dos necessários para o usuário foco deste trabalho.

Devido a complexidade e o tamanho destes sistemas, o valor se torna muito elevado, ficando inviável para um transportador autônomo fazer seu próprio controle. Uma ferramenta simples e funcional, como é a proposta do trabalho, seria uma solução viável para atender as suas necessidades, por ser simples de usar e focar no que é necessário, que é o controle de fretes e todos os cadastros envolvidos.

Metodologia e Desenvolvimento

O objetivo deste capítulo é mostrar como foi o processo de desenvolvimento do sistema.

4.1 Considerações Iniciais

Considerando o desenvolvimento de um software, é importante fazer uma avaliação do ambiente de atuação do sistema.

Um bom planejamento faz com que a aplicação se torne bem feita e consistente, sendo assim, durante o processo de desenvolvimento, não são necessárias alterações na estrutura do sistema, somente algumas adaptações caso seja preciso.

É essencial, uma análise de requisitos detalhada, escolha das ferramentas corretas e um banco de dados bem arquitetado para evitar futuros transtornos durante o desenvolvimento

4.2 Fases do Desenvolvimento

Esta seção tem como objetivo explicar as fases realizadas antes de se iniciar o processo de desenvolvimento do sistema.

4.2.1 Descrição do Problema

Foi escolhido um caminhoneiro autônomo, como caso de uso para ser utilizado o sistema. São duas pessoas que fazem o controle dos fretes, sendo então os usuários da aplicação.

Todo o controle de viagens era feito de forma manual, com anotações em cadernos e agendas, com cálculos feitos a mão. Apesar do volume de informações não ser grande, existe dificuldade para o usuário organizar as informações de uma maneira eficiente, em que seja fácil resgatar os dados quando preciso.

As imagens 4.2 e 4.1 como são controladas as informações. A imagem 4.2 mostra uma agenda telefônica, enquanto a imagem 4.1 representa o controle de fretes efetuados.

Pode-se notar a precariedade da documentação das informações, tanto quanto cadastro de clientes, endereços, telefones e informações pessoais. Não existe um critério de organização, para os cadastros pessoais nem mesmo por ordem alfabética. Para o controle de fretes, não existe separação para o que foi recebido ou não, assim como organização por data de realização do serviço ou cliente.

A maior motivação para realização deste projeto, partiu deste confuso processo de documentação. A partir deste ponto, foi sugerido o desenvolvimento de um software para automatizar este controle cadastral, onde as informações são centralizadas em uma base informatizada, dando controle mais preciso e fazendo com que a recuperação de informações seja mais eficiente, sem falhas e economizando tempo de trabalho.

Handwritten ledger entries for freight control:

- 18/04/16 = compra - Rodolfo = 104 sacos = \$41,00 A.V.
- 4264,00 = Cheque \$4.250,00 - faturas = 19/04/16 \$1400,00
- 19/04/16 = compra Menor - Dinheiro - 100 sacos = \$42,00 A.V. = 42
- + 20 sacos = \$40,00 A.V. - Registro
- T.M.F. = Paim - 13/04/16: 1713,07. Perfino + clp = 2136
- Paulo H. Silva - Paim - 25/04/16: 300,00 Adriano
- Paulo S. Ribeiro - Paim - 25/04/16: 3.983,00 Rodney 33
- Marcelo Viçentini - Silva - 02/05/16: 837,20 Adriano - Paim
- Adriano - Paim - 11/05/16: 199,00 Adriano - Paim
- João Ferreira - Paim - 01/05/16: 2.135,00 Rodney
- Wellington de Oliveira - Paim - 07/05/16: 998,00 Adriano - Paim
- 26/04/16 = compra - Dinheiro (Carla) - 4200 A.V. 190 sacos =
- 09/06/16 = compra - João Marco - 600 sacos = \$50
- 30.000,00 = Dinheiro = 9.000,00 - Cheques \$21000,00
- Adriano - Paim - 13/06/16 - Paim
- Adriano - Paim - 14/06/16 - Paim

Figura 4.1: Controle de fretes

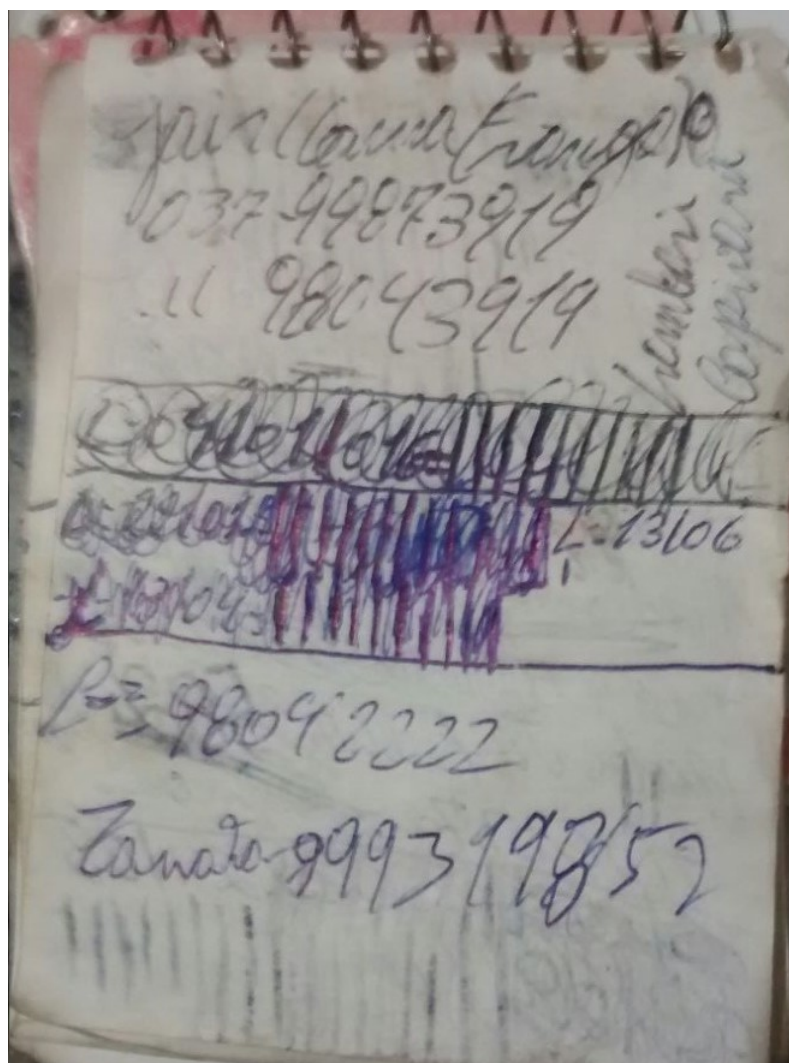


Figura 4.2: Registros telefônicos

Várias outros cadastros e controles, como preço de produtos, cadastro de veículos, locais e cidades também estão disponíveis para facilitar o trabalho dos usuários.

4.2.2 Levantamento de Requisitos

A primeira parte a ser feita no desenvolvimento de um software, é o levantamento de requisitos. Um requisito é considerado como uma especificação de uma característica ou propriedade, na qual o sistema possua ou deva exercer, assim como a sua restrição operacional. Podem ser definidos por algumas classificações, como: requisitos de negócio, funcionais, não funcionais, etc. Um levantamento de requisitos bem feito será necessário para ter uma fundação firme para realização das atividades durante o desenvolvimento. (Andrade, Márcio, 2016)

O passo inicial para detectar as necessidades do projeto, foi reunir com os futuros usuários para entender como era feito o controle, participar dos processos e saber quais as principais prioridades para o desenvolvimento do sistema. O resultado pode ser observado na tabela 4.1.

Tabela 4.1: Requisitos do Sistema

Requisito	Prioridade
Controle de viagens	Alta
Cadastro de clientes	Alta
Cadastro de veículo	Alta
Cadastro de motorista	Alta
Cadastro de mercadorias	Alta
Cadastro de locais	Alta
Histórico de preços	Média
Cadastro de cidades	Média
Controle de orçamentos	Baixa
Controle de manutenção	Baixa

4.2.3 Diagrama de Entidade Relacionamento (DER)

Após levantados os requisitos, o passo seguinte foi a modelagem do banco de dados, formando um diagrama de entidade relacionamento. Um modelo entidade relacionamento é uma representação dos dados para descrever os aspectos e informações do banco de dados do sistema. O modelo captura as necessidades organizacionais em termos de armazenamento de dados, independentemente da implementação (HEUSER, 2009).

Para a criação do modelo, é necessário o conhecimento do funcionamento do sistema, os atributos necessários, tabelas a serem usadas e a relação entre elas. Desde a criação do modelo, já se faz necessário o conhecimento da interação das tabelas por meio de suas chaves estrangeiras, sabendo assim quais as referências que as tabelas farão de uma para a outra. O diagrama de entidade relacionamento se encontra no apêndice A.

4.3 Desenvolvimento da Aplicação

Após a análise de requisitos e modelo do banco de dados pronto, o processo de implementação pode ser iniciado.

4.3.1 Criação do Banco de dados

Com base no modelo de entidade relacionamento, a aplicação pode começar a tomar forma. O primeiro passo é a criação do banco de dados no *PostgreSQL*.

O *PostgreSQL* permite que as tabelas sejam criadas de forma visual, não sendo necessário o uso de sentenças SQL, facilitando a modelagem do banco tornando mais rápida e eficiente na configuração das tabelas, tamanho de campos e tipos de dados. Um passo importante na criação das tabelas, é a definição das chaves tanto primárias quanto estrangeiras pois são essenciais para um funcionamento correto da aplicação.

4.3.2 Conexão entre banco de dados e aplicação

Após a criação do banco de dados dentro do *PostgreSQL*, antes de começar a codificar, é necessário fazer a conexão do banco de dados com a aplicação. É uma etapa simples gerenciada pelo próprio *Netbeans* através do assistente de nova conexão com banco de dados, como mostra a figura 4.3.

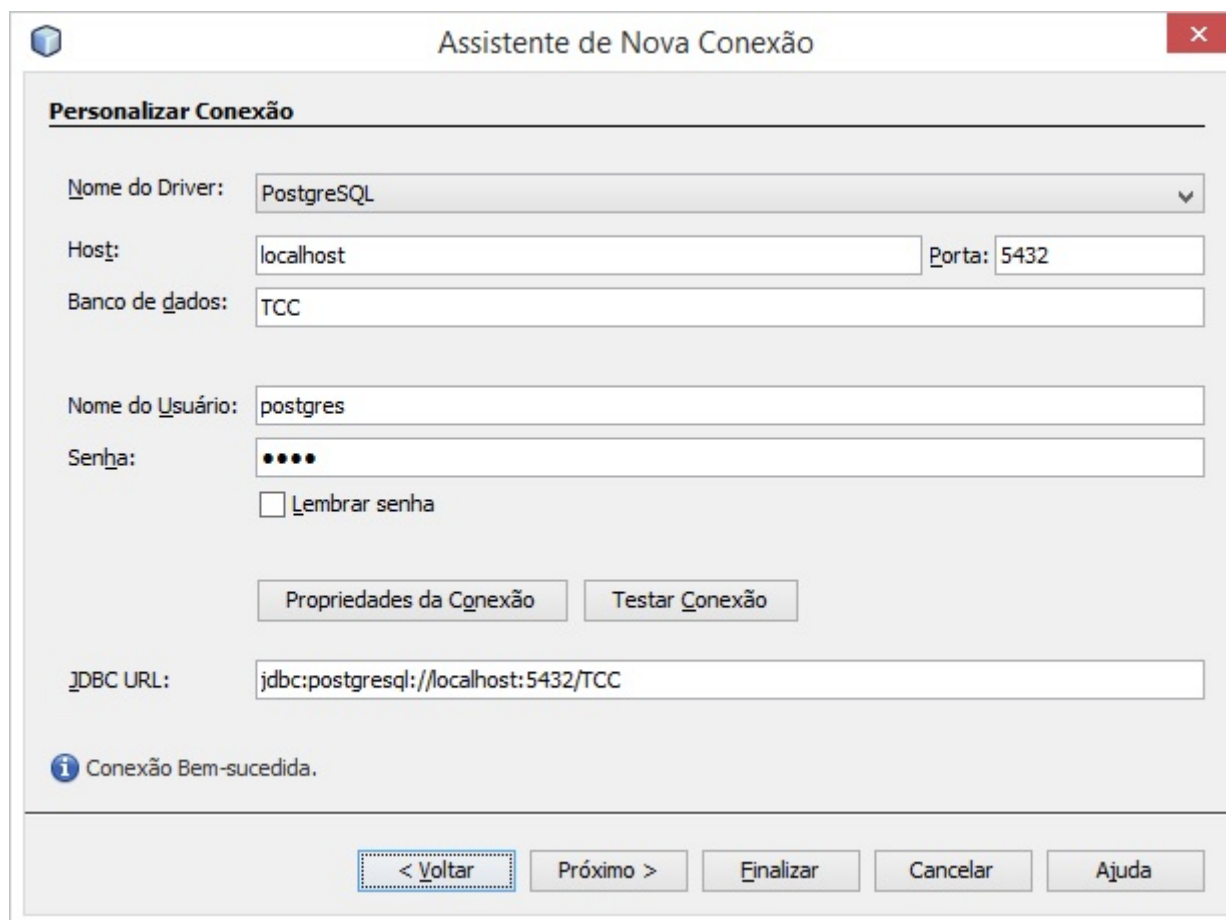


Figura 4.3: Assistente de Conexão ao Banco de Dados

São necessários para realizar a conexão o driver de conexão com o *PostgreSQL* e os dados da conexão. Feita a conexão com o banco, o próximo passo é a criação das classes java que fazem a representação das tabelas do banco dentro da aplicação. Para a criação das tabelas, o *Netbeans* juntamente com o *Hibernate*, geram automaticamente as classes correspondentes ao banco de dados. São necessárias as classes na aplicação, pois o *Hibernate* trata as tabelas do banco de dados de maneira orientada a objetos, sendo que as classes são a representação do banco dentro da aplicação.

Além da criação das tabelas, faz-se necessário, o mapeamento das classes dentro do arquivo "hibernate config", que é um arquivo xml de configuração gerado pelo *Hibernate* para gerenciar a conexão com o banco de dados. Este mapeamento tem como função mostrar para a aplicação, qual classe está relacionada a qual tabela dentro do banco de dados, como mostra a figura 4.4


```

3  <hibernate-configuration>
4  <session-factory>
5      <property name="hibernate.dialect">org.hibernate.dialect.PostgreSQLDialect<
6      <property name="hibernate.connection.driver_class">org.postgresql.Driver</p
7      <property name="hibernate.connection.url">jdbc:postgresql://localhost:5432/
8      <property name="hibernate.connection.username">postgres</property>
9      <property name="hibernate.connection.password">root</property>
10     <property name="hibernate.current_session_context_class">thread</property>
11     <mapping class="modelo.Cidade"/>
12     <mapping class="modelo.Cliente"/>
13     <mapping class="modelo.Estado"/>
14     <mapping class="modelo.HistoricoProduto"/>
15     <mapping class="modelo.ItemViagem"/>
16     <mapping class="modelo.Local"/>
17     <mapping class="modelo.Marca"/>
18     <mapping class="modelo.Modelo"/>
19     <mapping class="modelo.Motorista"/>
20     <mapping class="modelo.Pais"/>
21     <mapping class="modelo.Produto"/>
22     <mapping class="modelo.Unidade"/>
23     <mapping class="modelo.Usuario"/>
24     <mapping class="modelo.Veiculo"/>
25     <mapping class="modelo.Viagem"/>
26
27  </session-factory>
28  </hibernate-configuration>

```

Figura 4.4: Hibernate Config

4.3.3 Interface

Antes de serem desenvolvidas as funcionalidades do sistema, é criada uma interface para todas as telas de cadastro. O sistema possui uma padronização de todas as telas de cadastro e exibição dos dados. A parte visual utiliza de um arquivo “estilos.css”, onde é feita toda a parte de formatação de textos e cores do *template*, como cabeçalho, menu e rodapé.

Ao utilizar o primefaces, para que seja usado um *template*, basta que seja feita uma chamada no início do desenvolvimento de cada página, para que seja aplicado o *template* desejado.

Junto ao *template*, para tornar o sistema com uma interface simples e de fácil manuseio, é utilizado um tema pré-disponibilizado pelo primefaces, que dá forma e colorização a todos os componentes utilizados.

4.3.4 Estrutura da aplicação seguindo os modelos MVC e DAO

Como citado no capítulo 2, o desenvolvimento do software segue os padrões MVC e DAO para sua construção. Segundo este padrão, o código é dividido em pacotes para facilitar a codificação e manutenção.

As operações com o banco de dados são feitas exclusivamente dentro das classes do pacote DAO. As sentenças para incluir, remover e modificar são feitas separadamente às regras de

negócios para que seja mais fácil o trabalho de desenvolvimento.

Dentro de cada pacote estão as respectivas classes correspondentes a cada um dos conceitos do padrão MVC: Modelo, Visão e Controle.

As classes correspondentes ao banco de dados compõe o pacote modelo, onde não são alteradas manualmente. Caso haja alguma alteração no banco de dados, é necessária a exclusão de todo o pacote, e em seguida gerar novamente as classes de acordo com o banco de dados.

A pasta de páginas web é composta pela visão, onde estão todas os componentes visuais, como *templates*, imagens, ícones, CSS e as páginas web que irão interagir com o usuário.

O pacote controle é responsável pelos *Managed Beans*, responsáveis pelas funções da aplicação assim como seus métodos e regras de negócio. É dentro deles que a parte funcional do software é desenvolvido.

Durante o desenvolvimento da aplicação, deve ser seguida uma ordem para o desenvolvimento dos componentes. Primeiramente são criadas as classes do pacote modelo, que são geradas automaticamente pelo *Netbeans*, a partir da conexão com o banco de dados. O passo seguinte, é a criação das classes do pacote DAO, seguido da criação dos managed beans da classe controle e por final é feita a parte visual da aplicação, que são as telas exibidas para a interação com o usuário.

Esta ordem deve ser respeitada para que o desenvolvimento seja mais fácil e evite erros, pois cada passo de desenvolvimento descrito acima, depende do anterior para que seja concluído.

4.3.5 Autenticação do Usuário

Para o uso do sistema, faz-se necessário um controle de segurança, onde, somente àquele que possui o conhecimento de seu usuário e senha correspondentes têm acesso às informações. O método trabalha com um aninhamento de instruções, que checam se o usuário e senhas estão corretos, só assim será concedido o acesso ao usuário. Primeiro é verificado se foram digitadas as informações pedidas. Atendendo a essa condição, é consultado o nome de usuário. Uma vez que, o usuário sendo encontrado, verifica-se a senha, que se for correta dará acesso ao sistema.

4.3.6 Mensagens do sistema

O sistema possui dois tipos de mensagens para as alterações feitas pelo usuário, exibindo quando é feita alguma alteração no sistema se ela foi bem sucedida ou se houve alguma falha, como por exemplo, um campo obrigatório que não foi digitado. São mensagens flutuantes na tela que já são disponibilizadas pela linguagem. As figuras mostram as mensagens de erro 4.5 e de sucesso 4.6

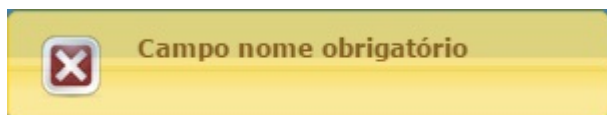


Figura 4.5: Mensagem de erro

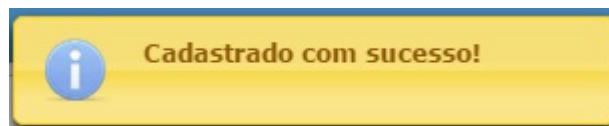


Figura 4.6: Mensagem de sucesso

Para facilitar a chamada de mensagens, foi criado dentro do pacote Util do projeto uma classe Java “UtilMensagens” que contém dois métodos chamados “mensagemErro” e “mensagemSucesso”. Quando necessário, é chamado o método respectivo à mensagem que deseja ser exibida ao usuário, passando como parâmetro uma string que contém o texto a ser exibido na mensagem. Feito isso, não é necessário implementar todo o método para que seja mostrada uma mensagem, fazendo assim, somente uma chamada e passando a mensagem como parâmetro.

Funcionamento e Resultados

Este capítulo irá demonstrar o uso do sistema e exibir os resultados obtidos com o desenvolvimento do projeto proposto. Serão exibidos o funcionamento do sistema e a impressão do usuário sobre o mesmo.

5.1 Considerações Iniciais

Após a conclusão do desenvolvimento do sistema, é importante que seja passado por uma fase de testes para comprovar o funcionamento correto do mesmo. A fase de testes foi feita com o uso do sistema, a fim de encontrar algum mal funcionamento ou ajuste necessário.

5.2 Sistema de Gerenciamento de fretes

Após a modelagem e desenvolvimento, o sistema que propõe a solução para os problemas do usuário em questão, pode começar a ser utilizado em fase de testes.

5.2.1 Tela de Login

Para iniciar o uso do sistema, é necessário que o usuário entre com seu usuário e senha, pré determinados pelo desenvolvedor. A figura 5.1, mostra a tela de autenticação do sistema.



The image shows a login form with a blue header bar containing the word "Autenticação". Below the header, there are two white input fields with blue borders. The first field is labeled "Login" and the second is labeled "Senha". Below these fields is a blue button with the word "Confirmar" in white text. The entire form is enclosed in a rounded rectangle with a subtle shadow.

Figura 5.1: Tela de login

5.2.2 Tela Principal

Após efetuado o login, o usuário é redirecionado para a página principal do sistema. A partir da tela principal, o usuário pode acessar as funcionalidades disponíveis como exibido na figura 5.2. São exibidas a opção home que leva ao menu principal, cadastros onde existe um submenu disponibilizando telas como cadastro de motoristas, clientes, locais e etc. Também existem as opções para cadastro de fretes emissão de relatórios.



Figura 5.2: Tela inicial

5.2.3 Cadastros

O sistema tem como objetivo, o cadastro e controle de fretes efetuadas. Para que essa viagem seja incluída no sistema, faz-se necessário uma série de cadastros para que ela possa ser efetuada. Estes cadastros, além de utilizados para a realização de uma frete, também se tornam úteis para controle de algumas informações importantes para o usuário, como por exemplo, o valor dos produtos, informações sobre os veículos, cadastro de motoristas, ficha de clientes e locais de carga e descarga.

A figura 5.3, representa a exibição da lista de clientes cadastrados.

Clientes								
Código	Nome	Endereço	Nº	Bairro	Cidade	Celular	Editar	Excluir
9	CLIENTE 1	ENDERECO 1	N1	BAIRRO 1	Abadia de Goiás	(11) 1 1111-1111		
10	CLIENTE 2	ENDERECO 2	N2	BAIRRO 2	Abadia dos Dourados	(22) 2 2222-2222		
11	CLIENTE 3	ENDERECO 3	N3	BAIRRO 3	Abadiânia	(11) 1 1111-1111		
12	CLIENTE 4	ENDERECO 4	N4	BAIRRO 4	Abaeté	(11) 1 1111-1111		
13	CLIENTE 5	ENDERECO 5	N5	BAIRRO 5	Abaetetuba	(11) 1 1111-1111		
14	CLIENTE 6	ENDERECO 6	N6	BAIRRO 6	Abaiara	(11) 1 1111-1111		
15	CLIENTE 7	ENDERECO 7	N7	BAIRRO 7	Abaira	(11) 1 1111-1111		
16	CLIENTE 8	ENDERECO 8	N8	BAIRRO 8	Abaré	(11) 1 1111-1111		
17	CLIENTE 9	ENDERECO 9	N9	BAIRRO 9	Abatiá	(11) 1 1111-1111		
18	CLIENTE 10	ENDERECO 10	N10	BAIRRO 10	Abdon Batista	(11) 1 1111-1111		

Figura 5.3: Tabela de clientes já cadastrados

Como um exemplo de tela de cadastro, pode-se observar a figura 5.4 que representa a tela para a inclusão de um novo motorista no sistema. Pode-se notar que campos como CPF e telefone possuem máscaras pré definidas no sistema, onde a formatação dos dados é salva em acordo com o seu formato correto.

O campo data utiliza de um componente do *Primefaces* que mostra um calendário, exibindo a data no formato correto, permitindo que o usuário navegue por ele para encontrar a data desejada. Para o cadastro de cidade, é usado um componente que recupera as informações do banco de dados e as dispõe para a seleção no campo para o usuário, basta selecionar a opção desejada. Todas as cidades foram introduzidas no banco durante o desenvolvimento de acordo com a tabela do IBGE, onde todas as cidades brasileiras estão listadas.

Podem ser vistos outros exemplos de telas do sistema na figura 5.5 e também figura 5.6.

SGV - Sistema de Gerenciamento de Viagens

Home Cadastro Viagens Relatórios

Nome: MOTORISTA CPF: 111.111.111-1 Endereço: ENDEREÇO Número: N01 Bairro: BAIRRO

Cidade: Lábrea Telefone: (11) 1111-1111 Celular: (11) 1 1111-1111 CNH: 111111111111 Categoria: AB Nascimento: 23/05/16

Email: EMAIL@EMAIL.COM Observação: OBSERVACAO

May 2016

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31				

Figura 5.4: Tela para cadastro de motoristas

SGV - Sistema de Gerenciamento de Viagens

Home Cadastro Viagens Relatórios

Nome: CLIENTE Endereço: ENDEREÇO Número: N01 Bairro: BAIRRO Complemento: COMPLEMENTO

Cidade: Pacaembu CPF: CNPJ: 11.111.111/1111-11 Telefone: (11) 1111-1111 Celular: (11) 1 1111-11 Email: EMAIL@EMAIL.COM

Figura 5.5: Tela para cadastro de clientes

Produtos					
Código	Descrição	Unidade	Preço	Editar	Excluir
7	MILHO	TN	R\$ 60,00		
8	AGUA	GL	R\$ 4,00		
9	SOJA	SC	R\$ 100,00		

Usuário: root

Figura 5.6: Lista de produtos cadastrados

5.2.4 fretes

A principal função deste projeto, está concentrada na tela representada pela figura 5.7. Nela todas as informações anteriormente cadastradas serão utilizadas para o registro de um novo frete efetuado.

SGV - Sistema de Gerenciamento de Viagens

Home Cadastro Viagens Relatórios

Local Origem: FAZENDA 7 Veículo: BBB-2222 Motorista: MOTORISTA 1 Data Inicial: 02/05/16 Data Final: 10/05/16

Cliente: Produto: Local Destino: KM Inicial: 0 KM Final: 0 Distância: 0

Quantidade: 0,00 Repasse: ☒ Sim ☐ Não Compra: 0,00 Venda: 0,00 Calcular: ☒ Km ☐ Peso Unitário: 0,00 Total Item: 0,00 Total Viagem: 490,00

Financeiro: Aberto Data Recebimento: 03/05/16

Código	Descrição	Cliente	Destino	Quantidade	Total Frete	Excluir
8	AGUA	CLIENTE 4	FAZENDA 2	0,00	490,00	

Usuário: root

Figura 5.7: Tela para inclusão de nova frete

A tela se divide em duas partes, sendo uma delas contendo os itens gerais da frete como origem, veículo, motorista, data de início e encerramento.

A segunda parte, contém informações específicas para cada item (cada frete pode conter mais de um item). Essa divisão é necessária pelo seguinte fato: em um mesmo lugar de carregamento, pode ser feita a carga de dois produtos a serem entregues em dois lugares diferentes, apesar de ser um caso de uso não muito comum. O usuário deve informar os dados da frete, em seguida do item. Ao ser inserido o item na frete, os seus respectivos dados serão exibidos na tabela de itens, indicando que pertencem a frete. Caso haja mais de um item, devem ser introduzidas as informações do novo item, até que todos sejam incluídos. Feito isso, poderá ser salva a frete em questão.

Para o cálculo do valor total da frete, são utilizados vários critérios diferentes. O frete pode ser calculado pela distância percorrida pelo veículo, multiplicando assim a quilometragem pelo valor unitário informado. Outra forma de cálculo se dá pela quantidade transportada, onde é cobrado um valor para cada unidade transportada. Também existe a possibilidade de ser uma mercadoria com valor de repasse, que é feita da seguinte maneira: o transportador compra a mercadoria e faz a revenda para o cliente, cobrando assim um valor de repasse. Para isso, é feito o cálculo com base na multiplicação da quantidade da mercadoria pelo valor de venda, resultando no valor a ser recebido pela frete. O sistema pede ao usuário que informe o valor de compra da mercadoria, que é o valor pago por ela e também o valor de venda, que será o valor cobrado do cliente. O valor de compra é armazenado na frete para critério de informações e relatórios, caso seja necessário futuramente uma busca por informações de fretes já finalizadas.

O valor total da frete consiste no somatório do valor de cada item transportado.

Para evitar erros nos cálculos, o sistema controla quais campos são habilitados para que o usuário informe somente os dados necessários. O campo distância por exemplo, não pode ser editado, existe uma função no sistema que faz o cálculo com base na quilometragem de origem e destino resultando na distância total.

Na tela também existem dois componentes que servem para controle de como serão feitos os cálculos de cada item. A lógica de funcionamento é feita da seguinte maneira: caso esteja habilitada a opção de frete por repasse, o campo para valor unitário é desativado, pois não será necessário no cálculo. Da mesma maneira, ocorre quando a opção de frete por repasse não é selecionada, abrindo assim duas possibilidades: cálculo por distância ou cálculo por quantidade. Para este fim, existe a opção de cálculo por quilometragem e peso. Selecionada a opção, o cálculo do valor será feito de acordo com o seu respectivo modo.

5.3 Relatórios personalizados

Uma *feature* de grande utilidade desenvolvida para este trabalho, é a possibilidade de que sejam solicitados relatórios com informações personalizadas. Utilizando a ferramenta *iReport*, o desenvolvedor pode criar facilmente relatórios para que sejam resgatadas informações do banco, dispondo-as ao usuário de maneira fácil a ser interpretada.

Esta função dá ao sistema grande versabilidade e amplia as funções do sistema, já que na criação de um relatório podem ser filtradas informações de diferentes maneiras. É possível usar totalizadores, filtros, ordenação por diferentes maneiras, controles mais precisos. Qualquer dado inserido no sistema, pode ser refinado e transformado em informação valiosa.

Nos apêndices B e C estão disponíveis dois exemplos de relatórios diferentes que fazem parte do sistema. O relatório do apêndice B mostra a relação de clientes cadastrados no sistema e seus dados mais importantes, tornando fácil a consulta às informações necessárias e também permite a impressão das mesmas, caso seja necessário.

O apêndice C mostra um resumo das fretes feitas, contendo informações como cliente, data e valor total. O relatório também possui um totalizador, onde é um somatório do valor de todas as fretes.

A possibilidade de criar relatórios personalizados, amplia o campo de uso do cliente, pois é uma maneira de filtrar as informações desejadas da maneira que seja necessária. Dessa forma, o usuário pode ter qualquer dado inserido no sistema de forma organizada e rápida, podendo explorar todas informações salvas, aproveitando-as da melhor forma possível.

Devido a facilidade de uso da ferramenta *iReport*, o desenvolvedor pode corresponder a solicitação do usuário em tempo hábil, pois a criação de um relatório para o sistema se torna uma tarefa simples de ser efetuada.

5.4 Resultados

Para a avaliação dos resultados do sistema, foi aplicado um questionário que pode ser visto no apêndice D, o qual foi baseado no trabalho de (ANA, 2000). Foram aplicados os questionários

em dois usuários do sistema, ambos com faixa etária superior a 45 anos, com o primeiro grau completo de escolaridade atingida. Ambos os usuários não possuem nenhuma experiência no uso de computadores e não haviam usado nenhum tipo de software, e apenas um dos usuários fazem uso de smartphone.

Quanto ao uso do sistema, podem ser traçados dois perfis ligeiramente diferentes. Um dos usuários demonstrou uma facilidade maior quanto à operação do software, e conseqüentemente, seus resultados demonstraram uma maior satisfação quanto ao uso. Este usuário em questão, mostrou que apesar de não ter contato nenhum com computadores, possuía grande habilidade para o uso, e logo após um rápido treinamento conseguiu dominar o sistema e fazer seu uso sem necessidade de mais instruções. O usuário em questão, é o que faz uso de smartphone, que mostra uma adaptação maior quanto ao uso de softwares.

O segundo usuário, mostrou ter dificuldades para a operação do sistema, demandando uma atenção maior durante as instruções iniciais de como deveriam ser feitos os procedimentos. Inicialmente, o usuário demonstrou resistência quanto à utilidade do software, insistindo que não era necessário. Porém, após um certo período de aprendizagem, o usuário conseguiu operar com menos dificuldades e teve uma aceitação maior para o uso do sistema, concordando que seria um avanço importante para seu empreendimento.

Apesar da diferença entre as primeiras impressões dos usuários, o resultado final é unânime e extremamente satisfatório. As questões relacionadas a satisfação do uso e expectativas quanto à solução dos problemas tiveram resultados extremamente positivos, principalmente em relação ao controle e cálculo de fretes. Os usuários ficaram impressionados com a rapidez e a confiabilidade dos resultados.

Outro ponto positivo, foi o *feedback* relacionado a interface e navegação no sistema. Durante o desenvolvimento, foi uma grande preocupação desenvolver uma aplicação com uma interface fácil de usar e auto explicativa, onde o usuário consiga absorver rapidamente quais são as funções dos botões, posicionamento dos menus, como realizar consultas e edições nos cadastros e etc. Ambos os usuários demonstraram respostas positivas nestes quesitos.

Em resumo, pode-se concluir que as primeiras impressões dos usuários não foram boas, demonstrando resistência para fazer a mudança das anotações manuais para o sistema, porém, após o período de aprendizagem e treinamento, e visto os resultados, ambos concordaram plenamente em automatizar seus processos totalmente, abandonando seus métodos de controle manuais.

Trabalhos Futuros

Como mostrado na tabela 4.1, existem duas funções que foram declaradas como prioridade baixa para o sistema, não sendo de extrema urgência no período de uso inicial do software, que são o controle de manutenção de veículos e orçamentos

6.1 Controle de manutenções

O controle de manutenções de veículos tem como objetivo o agendamento de manutenções programadas aos veículos da frota. Esta funcionalidade, dá a opção para o usuário programar quando deve ser feita uma manutenção de rotina, ou registrar manutenções feitas sem programação.

Para o controle de frotas, é uma opção necessária, pois organiza o controle e também torna mais fácil o agendamento. Manutenções veiculares programadas são feitas com base na quilometragem do veículo, que ao ultrapassar uma determinada quilometragem, faz-se necessária. O sistema irá operar com base neste valor. Ao introduzir a quilometragem de uma viagem para certo veículo, o sistema verifica se existe manutenção programada para o mesmo com a quilometragem introduzida na viagem. Caso exista, ele informa ao usuário que deve ser feita a manutenção agendada.

6.2 Orçamentos

Juntamente com o controle de manutenções, faz-se necessária a opção de orçamentos dos materiais utilizados nas manutenções, como por exemplo, mão de obras e peças. Para isso, é preciso que seja implementado um cadastro de peças juntamente aos orçamentos. O usuário faz o cadastro dos itens necessários e ao fazer o orçamento, inclui o material com o preço de cada estabelecimento, para que possa ser feita a comparação de preços cotados posteriormente.

No controle de manutenções, o sistema permitirá a seleção de um orçamento aprovado para

realizar a mesma. Após a manutenção ser marcada como realizada, o usuário terá um controle financeiro do valor gasto com cada veículo, o que é uma informação de extrema importância para a organização, podendo ser utilizada posteriormente.

A figura 6.1 é uma representação do projeto inicial de como será a modelagem do banco de dados da extensão da aplicação a ser desenvolvida como continuidade do trabalho.

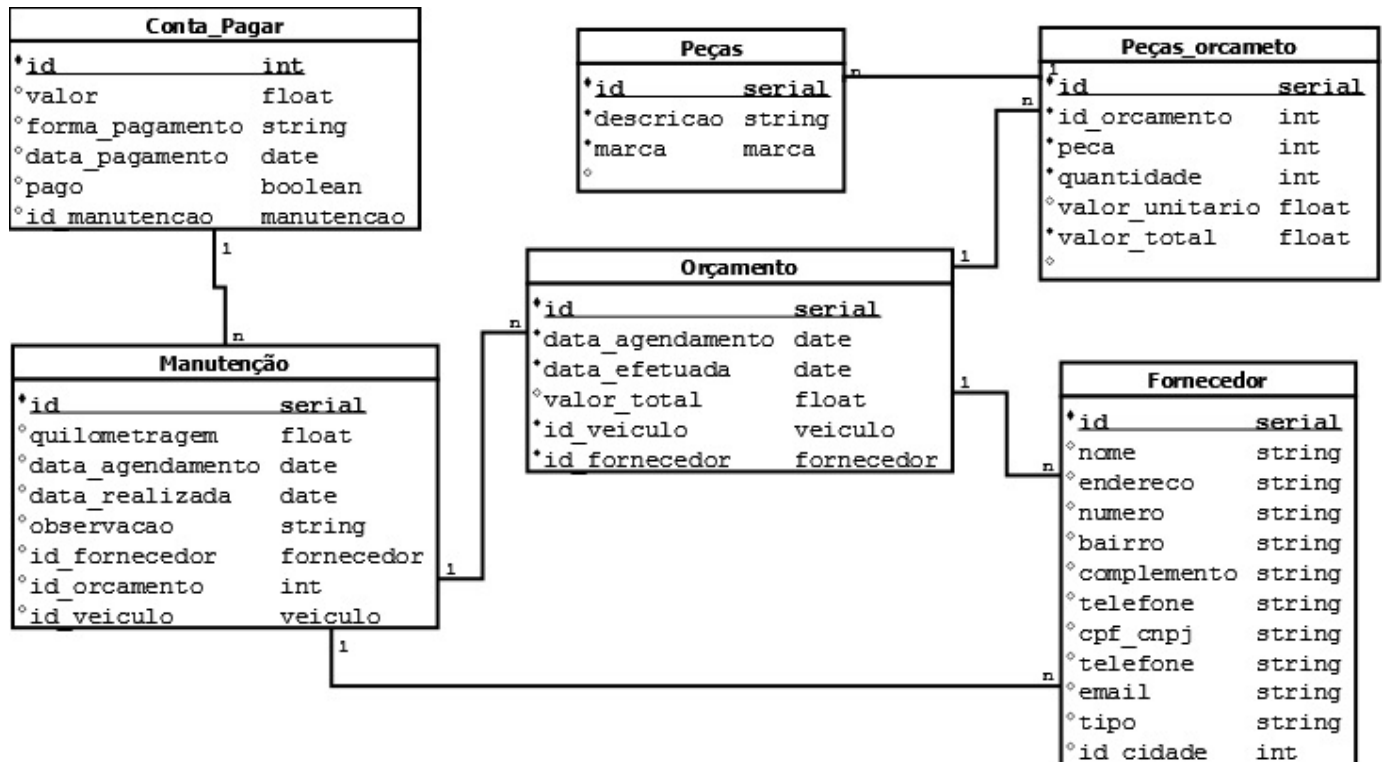


Figura 6.1: Esboço do diagrama entidade relacionamento do módulo de manutenções e orçamentos

Conclusões

Este capítulo tem como objetivo, um resumo de todas as conclusões obtidas com o desenvolvimento do projeto.

7.1 Considerações Finais

Após todo o desenvolvimento do projeto, pode-se concluir que com o levantamento de dados e requisitos feitos previamente, que as expectativas foram atendidas. Pôde-se confirmar que as necessidades requisitadas pelos usuários foram amplamente correspondidas.

O uso de um SIG, neste caso em particular um sistema para gerenciamento de viagens, mostrou ser uma solução eficiente para o controle cadastral. O sistema soluciona um problema, que é a falta de um software simples e específico para este tipo de empreendedor autônomo, visto que a contratação de um ERP completo torna-se inviável devido ao alto custo de implantação e manutenção. Além do alto valor de custo, um ERP possui funções desnecessárias que não seriam utilizadas, como controles contábeis e emissão de documentos fiscais.

Outra funcionalidade bastante útil, que é a criação de relatórios personalizados, torna o sistema versátil e permite que o usuário solicite qualquer tipo de informação com base dos dados cadastrados no sistema.

Além das funcionalidades já disponíveis, com a realização dos trabalhos futuros propostos no capítulo 6.1, além das necessidades essenciais do empreendedor, serão introduzidos novos controles que também serão importantes para o gerenciamento do empreendedor.

Referências Bibliográficas

- ANA, M. Roteiro de avaliação de usuário sistema de controle de horas trabalhadas scoht 1.0. *Compact Software*, 2000.
- Andrade, Márcio. *A importância do levantamento de requisitos no sucesso dos projetos de software*. 2016. Data de acesso: 24 mai. 2016. Disponível em: <<http://www.linhadecodigo.com.br/artigo/1685/a-importancia-do-levantamento-de-requisitos-no-sucesso-dos-projetos-de-software.aspx>>.
- APP4ALL. *Frete Autônomo (calculadora)*. 2016. Data de acesso: 24 mai. 2016. Disponível em: <https://play.google.com/store/apps/details?id=app4all.frete&hl=pt_BR>.
- BATISTA, E. de O. *Sistemas de informação: o uso consciente da tecnologia para o gerenciamento*. [S.l.]: Saraiva, 2006.
- BAUER, C.; KING, G. *Hibernate in action*. Manning Greenwich CT, 2005.
- BAZZOTTI, C.; GARCIA, E. A importância do sistema de informação gerencial para tomada de decisões. *VI Seminário Unioeste-2008*. Disponível em:<<http://www.unioeste.br/campi/cascavel/ccsa/VISeminario/ArtigosapresentadosemComunica/ART3aimportanciadesistemadeinformacaogerencialparatomadadedeciso.es.pdf>> Acessado em, v. 7, 2007.
- BERGSTEN, H. *JavaServer Pages Pocket Reference*. [S.l.]: "O'Reilly Media, Inc.", 2001.
- BSoft. *Controle de Transportes*. 2016. Data de acesso: 17 mai. 2016. Disponível em: <<http://www.bsoft.com.br/software-controle-transportadoras>>.
- BURBECK, S. Applications programming in smalltalk-80 (tm): How to use model-view-controller (mvc). *Smalltalk-80 v2*, v. 5, 1992.
- DATE, C. J. *Introdução a sistemas de bancos de dados*. [S.l.]: Elsevier Brasil, 2004.
- DAYANA, K. Tópicos emergentes do sistema de informação gerencial. <http://www.administradores.com.br/artigos/tecnologia/sistema-de-informacao-gerencial>, 2008.
- Dbwrench. *DbWrench, Database Design and Synchronization Software*. 2016. Data de acesso: 23 mai. 2016. Disponível em: <<http://www.dbwrench.com/>>.

Delphisga. *Recursos Delphisga*. 2016. Data de acesso: 17 mai. 2016. Disponível em: <<http://delphionline.com.br/solucoes/>>.

FESTA, E.; ASSUMPÇÃO, M. R. P. Uso da tecnologia de informação e desempenho logístico na cadeia produtiva de eletroeletrônicos-doi: [http://dx. doi. org/10.15600/2238-1252/rct.v17n33p7-23](http://dx.doi.org/10.15600/2238-1252/rct.v17n33p7-23). *Revista de Ciência & Tecnologia*, v. 17, n. 33, p. 7–23, 2012.

HEUSER, C. A. *Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS*. [S.l.]: Bookman Editora, 2009.

Martins, Marcelo. *Relatórios em Java - JasperReports e iReport*. 2010. Data de acesso: 01 jun. 2016. Disponível em: <<http://www.k19.com.br/artigos/relatorios-em-java-jasperreports-e-irepor>>.

MENDES, J. V.; FILHO, E. E. Sistemas integrados de gestão erp em pequenas empresas: um confronto entre o referencial teórico e a prática empresarial. *Revista Gestão & Produção*, SciELO Brasil, v. 9, n. 3, p. 277–296, 2002.

MOENSTER, A. S.; BONETTI, T. P. Um estudo sobre os frameworks jsf e primefaces no desenvolvimento de software web.

PITANGA, T. Javaserwer faces: A mais nova tecnologia java para desenvolvimento web. *UFRN*, 2008.

PostgreSQL. *Sobre o PostgreSQL*. 2016. Data de acesso: 02 mai. 2016. Disponível em: <<https://www.postgresql.org.br/sobre>>.

Primefaces.org. *Why Primefaces?* 2016. Data de acesso: 02 mai. 2016. Disponível em: <<http://www.primefaces.org/whyprimefaces>>.

Souza, Nailson. *O que e Hibernate?* 2012. Data de acesso: 02 mai. 2016. Disponível em: <<http://blog.naison.com.br/java/o-que-e-hibernate>>.

Taboarda, Sérgio. *DAO*. 2016. Data de acesso: 02 mai. 2016. Disponível em: <<https://sergiotaborda.wordpress.com/desenvolvimento-de-software/java/patterns/dao/>>.

Diagrama de entidade relacionamento

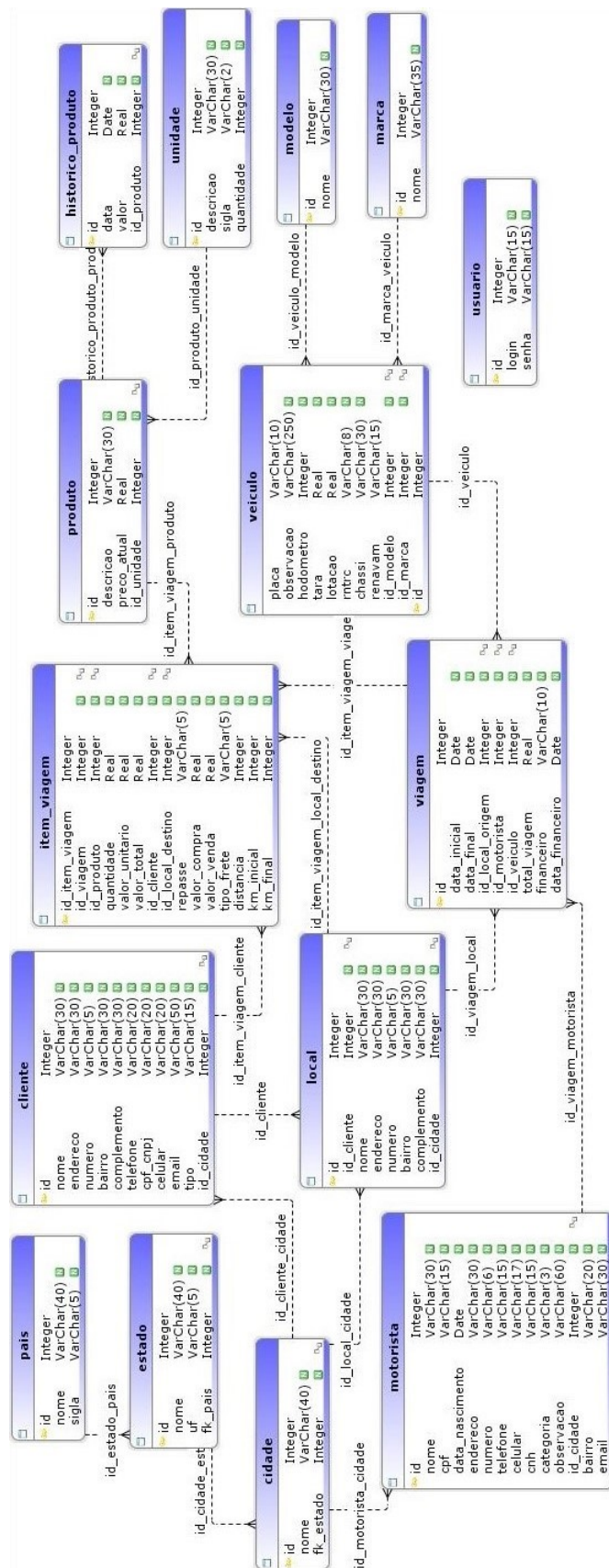


Figura A.1: Diagrama Entidade Relacionamento

Relatórios de relação de clientes

Relação de Clientes

Código	Nome	Endereço	Bairro	Cidade	Telefone	Email
9	Fernando Souza	Av. Pedro Sales, 325	Centro	Formiga	(37)9251-1231	fernandosouza@gmail.com
10	Maria Paula Rodrigues	Rua Lisboa, 231	Sagrada	Belo Horizonte	(31)9312-9332	mariapaularodrigues@gmail.
11	Vanuza dos Santos	Rua Osório Sebastião, 12	Centro	Formiga	(37)3829-0241	vanuza_santos@terra.com.br
12	Eduardo Gomes de Oliveira	Av. Evaristo Gomes Guerra, 65	Alvorada	Formiga	(37)8324-9834	dudu_gomes@hotmail.com
13	Maurício de Oliveira Neto	Rua Três, 234	Quinzinho	Formiga	(37)3321-9982	mmauricioliveira@gmail.com
14	Maria Rosângela da Silva	Rua Dr. Paulo Menicuci, 682	S.C. de Jesus	Formiga	(37)9843-9901	mariarosasilva@yahoo.com.br
15	Anderson Luiz de Melo	Rua Anibal Teodoro, 278	Centro	Varginha	(35)3324-1126	adersonmelo@gmail.com
16	Marcos Paulo Furlan	Praça Dr. Augusto Souza, 20	Areias Brancas	Formiga	(37)3322-0084	furlan_marcos@hotmail.com
17	Carlos Chagas	Rua José Moreira Primeiro, 89	Caiçara	Belo Horizonte	(31)9435-0901	carloschagas@gmail.com
18	Lúcia da Silva	Rua Dr. Samuel Gamon, 763	Santo André	Belo Horizonte	(31)9677-2117	luciasilva1986@yahoo.com.br
19	Antônio Francisco Santos	Rua Alberto Boari, 256	Burititis	Belo Horizonte	(31)9211-7659	toninho_santos@yahoo.com.br
20	Virgínia Calixto Almeida	Rua Antônio Hermeto, 867	Centro	Formiga	(37)3329-8665	vivi_calixto@gmail.com

Relatório de resumo de fretes

Relatório de Viagens

Código	Data Inicial	Data Final	Local	Motorista	Placa	Total Viagem
7	01/04/2016	04/04/2016	Fazenda Rosa Mística	Antônio Rabelo de Melo	HBJ-2111	9000,00
8	01/04/2016	26/04/2016	Fazenda Ponte Alta	Mauro Vieira Carvalho	HRZ-8357	98901,00
10	02/05/2016	23/05/2016	Fazenda Ponte Alta	Antônio Rabelo de Melo	HBJ-2111	1000,00
14	02/05/2016	10/05/2016	Fazenda Rosa Mística	Mauro Vieira Carvalho	HRZ-8357	490,00
Total Geral:						109391,00

Questionário de avaliação



Questionário de Avaliação do Sistema

Responda as questões pessoais para a avaliação do perfil do usuário:

1) Qual a sua faixa etária?

- a) 18-25
- b) 26-40
- c) 40-55
- d) Mais que 55

2) Qual opção melhor define seu grau de instrução?

- a) Primeiro grau
- b) Segundo grau
- c) Ensino superior

3) Possui experiência no uso de sistemas de computador?

- a) Sim
- b) Não

4) Você faz uso de smartphone?

- a) Sim
- b) Não

7) Como você descreveria seu nível atual de experiência com computadores?

- a) Nenhum (Nunca utilizei um aplicativo de computador).
- b) Baixo (Ocasionalmente utilizo um aplicativo de computador).
- c) Moderado (Frequentemente utilizo um aplicativo de computador mas não me considero um expert).
- d) Alto (Considero-me um expert em pelo menos dois aplicativos de computador).

Após feita o uso e a avaliação do sistema, favor responda as questões propostas de acordo com a experiência de uso.

		Ruim		Mediano		Bom	
		0	1	2	3	4	5
1	Satisfação em relação ao uso do sistema						
2	Em relação a sua expectativa ao que o sistema propôs fazer						
3	Os lay-outs das telas foram úteis						
4	Quantidade de informação mostrada na tela						
5	Organização da informação						
6	Seqüência de Telas						
7	Mensagens que apareceram nas telas						
8	Instruções para comandos ou funções						
9	Instruções para correção de erros						
11	Aprendizado para operar o sistema						
12	Tempo para aprender a usar o sistema						
13	Descoberta de novas funcionalidades						
14	Lembrança de nomes e usos de comandos						
15	Tarefas podem ser executadas de maneira direta						
16	Número de passos para executar uma tarefa						
17	Falhas de sistema ocorreram						