

PROJETO DO SISTEMA: *delivery system*

Gustavo Henrique Borges Rosa¹
Jardel Rodrigues Marques De Lima²
Sérgio Augusto De Souza Moraes²

RESUMO

Com o alto índice de pedidos em empresas de fast-food surge a necessidade de um controle e automação no setor de entregas, possibilitando um melhor atendimento ao cliente e agregando uma entrega mais ágil e confiável, por esse motivo o Delivery System vêm trazer um gerenciamento de pedidos que conta com cadastro de clientes, produtos e pedidos, tal como um bom gerenciamento dos pedidos feitos anteriormente por um mesmo cliente juntamente com um relatório de entrega que traz mais segurança tanto para a empresa quanto para o cliente, o padrão de desenvolvimento utilizado foi o Web pois o mesmo pode ser acessado de qualquer lugar, basta ter acesso a internet e um navegador, possibilitando assim uma melhor administração e controle do processo, o presente projeto busca, de modo geral, uma melhor atendimento do cliente possibilitando a melhor interação e maior fidelização, cliente/empresa, onde no decorrer do projeto valida-se a hipótese de conclusão favorável onde todos os requisitos propostos foram implementados.

Palavras-chave: Web; fast-food; gerenciamento; segurança.

ABSTRACT

With the high demand in fast-food companies, there is a need for control and automation in the delivery sector, enabling a better customer service and adding a more agile and reliable delivery, which is why Delivery System comes with a management of orders that has customer, product and order registration, such as a good management of orders previously made by the same customer together with a delivery report that brings more security to both the company and the customer, the standard of development used was the Web because it can be accessed from anywhere, just have access to the internet and a browser, thus enabling better management and control of the process, this project seeks, in general, a better customer service enabling the best interaction customer / company, where

¹ Acadêmico do curso de Sistemas de Informação – Faculdade Atenas

² Docente do curso de Sistemas de Informação – Faculdade Atenas

during the course of the project the hypothesis of a favorable conclusion is validated where all the requirements were implemented.

Keywords: *Web; fast-food; management; security.*

INTRODUÇÃO

Sistemas automatizados estão cada vez mais presentes em empresas, seja um supermercado, restaurante, lanchonete, com isso surge a necessidade de um sistema que gerencie e automatize os pedidos emitidos por estas empresas.

É com sistemas de informação que uma organização reúne seus recursos humanos, suas informações, seus recursos tecnológicos afim de melhorar cada parte da organização, seja no setor de embalagem, seja no setor de pedidos ou entregas. (TURBAN, RAINER, JR. E POTTER, 2007).

As empresas procuram atender ao máximo as expectativas dos clientes, buscando oferecer todos os recursos que eles necessitam, que são poucos em mercado, no entanto as empresas têm dificuldade de alcançar os desejos de seus clientes pelo fato dos clientes não terem facilidade de se expressarem no que desejam, (CAMPOS, BRASIL 2007).

Por esse motivo surge-se a necessidade de um sistema automatizado aliado a uma boa logística tornando uma etapa existente dentro da empresa em um sistema movido por impulsos elétricos e com um profissional capacitado comando uma boa parte ou pequena parte do sistema em questão. (RIBEIRO, 2001).

Para a criação de um sistema automatizado faz-se o uso de um programa para o desenvolvimento, criação, que podem ser chamados de IDE ou ambiente de desenvolvimento, que usam linguagem de programação, uma linguagem computacional com regras de semântica e sintaxe que possibilitam a programação em alto nível de uma aplicação.

Com o auxílio de um sistema automatizado espera-se apresentar um sistema de controle de entregas, que armazena dados pertinentes a clientes e pedidos, gerando informações para a facilitação dos próximos atendimentos.

CONCEITOS DE SISTEMAS DE INFORMAÇÃO E PROJETO DE SISTEMAS

Sistemas de Informação

Para se estabelecerem no mercado as organizações usam sistemas informatizados para melhoria de seus processos, assim Turban, Rainer, Jr. e Potter (2007) apresentam como

necessidades organizacionais a obtenção de informações corretas para os usuários no momento certo, com qualidade e formatos corretos, assim os sistemas de informação devem suprir as necessidades descritas e apresentar informações uteis. Logo, os sistemas de informações podem gerar informação em formato de dados e conhecimento. Nas palavras de Turban, Rainer, Jr. e Potter (2007) dados são um tipo de informação, que por sua vez, não estão trabalhadas ou dispostas de modo a gerar uma informação coerente, com um significado, já a informação é um conjunto de dados organizados de forma a gerar um valor para quem recebe, por fim, o conhecimento é a junção dados e/ou informações selecionados de modo a gerar uma informação válida que auxiliem na tomada de decisão.

As tecnologias da informação estão cada vez mais presentes dentro das organizações, seja ela pequena, média ou de grande porte, uma vez que a maioria dos processos já utiliza uma tecnologia da informação para melhor gerenciamento e aproveitamento de tempo organizacional, tanto é, que processos onde não possuem tecnologia da informação, são analisados para poderem ser sistematizados podendo assim melhorar o desempenho de tal processo, e possibilitando vantagens competitivas. (BALTZAN, PHILLIPS, 2012).

Nem todo sistema de informação utiliza um computador, em ressalva Turban, Rainer, Jr. e Potter (2007) ressalta que um sistema de informação baseado em um computador, utiliza os meios tecnológicos para realizar a maioria, ou todas as tarefas pretendidas, por outro lado, o termo sistema de informação, normalmente é usado para a definição de sistema de informação baseado em um computador.

Gestão de Projetos

No desenvolvimento de um sistema de informação se utiliza a gestão de projetos que, segundo Menezes (2009), um projeto nasce de uma necessidade de uma organização, que podem ser separadas em melhoria, em um produto, novo produto, melhoria interna, mudança organizacional e como definição clara, um projeto deve ter um início e fim bem definidos que nos leva a perceber um ciclo de vida em um desenvolvimento de projeto que é dividido em fases, I conceitual, II planejamento, III execução e IV conclusão.

É dentro da gestão de projetos que se define tudo que vai acontecer no seu projeto, levando-se em conta os riscos do projeto, o fator humano e tecnológico necessário para o desenvolvimento do projeto. Segundo Menezes (2009), o fator humano em um projeto é o fator mais delicado, pois cada pessoa tem suas particularidades: vontades, aspirações,

temperamento, caráter, uma equipe em um projeto deve ser formada aos poucos, procurando pessoas que vão realizar a sua função determinada em conjunto com os demais membros da equipe, fazendo o projeto fluir em bom estado. O fator humano em um projeto de sistema de informação é o fator mais importante do mesmo, pelo fato de todo o trabalho ser desenvolvido pelo capital humano.

Existem perguntas que um gerente de projetos, ou patrocinador do projeto, deve ter respostas bem elaboradas e satisfatórias para garantir o sucesso e aprovação do projeto. Neste contexto, Kerzner (2016) aponta algumas perguntas que devem ser respondidas para a aprovação e possível sucesso na elaboração de um projeto, tais como: A versão atual do sistema atende as necessidades da organização? Qual motivo leva o investimento nesse projeto? O custo do projeto será reintegrado a organização em tempo satisfatório?

Essas e outras perguntas são respondidas no início do planejamento do projeto, e aliado a gestão de projetos temos a engenharia de software que auxilia e documenta cada parte do desenvolvimento como diagramas e casos de uso.

Engenharia de Software

Junto com a gestão de projetos se encontra a engenharia de software, na qual irá se responsabilizar pelo desenvolvimento da solução em si, assim Sommerville (2011) ressalta que seu o foco da engenharia de software está em todos os aspectos do desenvolvimento do software, desde seu início até a sua manutenção, depois que o sistema já foi implantado.

Já Pressman (1995) relata que a engenharia de software é dividida em três elementos principais, métodos, ferramentas e procedimentos, sendo os métodos um amplo conjunto de tarefas de como fazer a construção do software envolvendo análise de requisitos do software, codificação, testes e manutenção, já as ferramentas proporcionam uma ajuda automatizada ou semi-automatizada em cima dos métodos conhecidos como CASE (Computer Aided Software Engineering) e por último, os procedimentos são os elos que mantém juntos os métodos e ferramentas, possibilitando um desenvolvimento racional, os procedimentos definem uma sequência em que os métodos deverão ser aplicados.

Dentro da engenharia de software existem ciclos, um deles é o levantamento de requisitos que, segundo Sommerville (2011), é a descrição do que o sistema deve fazer, o que ele deve oferecer ao seu usuário, suas restrições de funcionamento, sendo que os requisitos são separados em requisitos funcionais e não funcionais, nesse sentido, os funcionais definem como o sistema deverá reagir a entradas específicas e como o sistema deverá se comportar

em determinadas situações, já os não funcionais, são restrições aos serviços ou funções oferecidos pelo sistema, que podem ser de tempo, restrições de processos e/ou restrições de regras de negócio. Sendo que o levantamento de requisitos, é sem dúvidas, a parte mais importante do projeto, onde se define por completo o que o sistema em questão deve proporcionar.

Levantamento de Requisitos

O levantamento de requisitos é uma das fases mais importantes no desenvolvimento de um software, pois segundo Balbé (2011), um time de desenvolvimento é encarregado de levantar tudo que o sistema deve conter para seu bom funcionamento e, normalmente, os requisitos são extraídos com o domínio do negócio e na existência de um sistema antigo em funcionamento, a equipe de levantamento deve se preocupar em desenvolver algo novo, não se restringindo somente ao sistema antigo.

Já Quiterio (2012) ressalta que a análise de requisitos é vital para determinar o sucesso ou fracasso do projeto, sendo que os requisitos devem estar de acordo com a necessidade do cliente pelo fato que uma análise bem elaborada implica diretamente no custo final do projeto e a aceitação do cliente.

Ainda nas palavras de Quiterio (2012) a análise de requisitos se divide em requisitos funcionais e requisitos não funcionais, os funcionais são exatamente o que o sistema deve fazer, como ele deve agir e responder as ações do usuário, já os não funcionais são restrições e propriedades do sistema, tais como tempo de resposta de uma função, alocação de espaço em disco, alocação de memória, tempo gasto no processador.

Já Pressman (2016) ressalta em seu livro que levantamento de requisitos pode ser feito de forma ágil, encaixando-o junto com o desenvolvimento de aplicações ágeis, e que desse modo, o levantamento de requisitos é feito em partes, usando o conceito dividir para conquistar, de modo que os envolvidos no desenvolvimento se juntam e levantam uma parte dos requisitos e desenvolvem posteriormente, fazem um novo levantamento de requisitos gerando mais uma parte a ser desenvolvida.

O desenvolvimento ágil veio com a necessidade do desenvolvimento de aplicações com um menor custo de tempo para atender as necessidades das empresas.

Metodologia de Desenvolvimento Ágil

Uma metodologia de desenvolvimento é um guia contendo regras e técnicas para auxiliar no desenvolvimento de um software, entre esses guias existem as metodologias de desenvolvimento tradicionais e as ágeis que podem ser definidas como processos de software, que segundo Soares (2004) são atividades e resultados gerando processos, que juntos geram um produto final proveniente das técnicas utilizadas, as quais existem várias e dentre essas podemos citar as tradicionais, como as orientadas a documentações e que não conseguem atender os padrões de desenvolvimento das empresas, perante isso surge a necessidade de processos mais rápidos para o desenvolvimento.

Perante essa necessidade surge-se a de metodologia de desenvolvimento ágil, que segundo Pressman (2016) é a junção do desenvolvimento tradicional, com a constante alteração do software, custo mais aceitável para o cliente, equipe mais unida e cada membro tendo diversos papéis dentro da equipe, já o desenvolvimento é separado trabalhando em pequenas partes e depois juntando-as e apresentando ao cliente, esse método traz o cliente para dentro do desenvolvimento, tendo participação efetiva na tomada de decisão e alteração no software.

Dentro do desenvolvimento ágil existem diversos métodos, como o Extreme Programming – XP e o Scrum, sendo que os mesmos seguem o princípio do desenvolvimento ágil no seu desenvolvimento.

Extreme Programming – XP

Dentro das metodologias ágeis encontra-se o extreme programming (XP) onde Teles (2017) ressalta que é uma metodologia onde os requisitos são alterados com frequência durante o desenvolvimento, o cliente é um elemento constante, acompanhando a cada dia novas mudanças no projeto, aprovando e/ou mudando funcionalidades já presentes. Com isso, o cliente consegue definir o que vai agregar mais valor à sua empresa, posteriormente passando essas informações para a equipe de desenvolvimento.

E o XP conta com quatro valores fundamentais, que nas palavras de Kuhn (2004) definem as atitudes e as prioridades da equipe, e começa-se com o feedback constante entre o cliente e a equipe de desenvolvimento, onde o feedback que define se o projeto está indo de acordo com as necessidades do cliente, têm se também a comunicação onde ela é intensa tanto entre os membros da equipe quanto entre a equipe e o cliente, por outro lado, possui-se a

simplicidade que proporciona uma entrega de produtos ao cliente de forma simples e funcional, implementando todas as funções propostas pelo cliente e, por último, a coragem, a qual não pode faltar nesse desenvolvimento, onde ela é motivadora e a equipe desenvolve um software incremental, todos da equipe tem acesso ao código fonte, integra várias partes do sistema todos os dias, unir a equipe de desenvolvimento com o cliente e modelar e documentar apenas quando for extremamente necessário.

Em contrapartida, encontra-se o scrum, outra metodologia de desenvolvimento ágil que pode ser outra alternativa de metodologia.

Scrum

O Scrum é um framework de desenvolvimento ágil que não se aplica somente ao desenvolvimento de programas de computador, sendo que Sutherland (2016) ressalta que foi usando em diversas outras ocasiões, tais como na fabricação de um carro, ensino em sala de aula, construção de foguetes e até no gerenciamento de uma lavanderia, o scrum pode ser utilizado em sistemas evolucionários, autocorretivos e adaptativos.

Ainda sobre o scrum, Cruz (2013) conta que o scrum é um framework que gerencia projetos ágeis e tem como foco a entrega de resultados no menor tempo possível, o time scrum é dividido em time menores responsáveis por ciclos de chamados de sprints, e a cada sprint um problema é resolvido e apresentado ao cliente, dentro de um sprint todos do time participam até o término do sprint e a apresentação para o cliente.

Contudo, o scrum é usado e projeto de maior amplitude de tempo e recursos, não sendo recomendado para projetos menores, por esse fato o projeto descrito nesse trabalho será executado em XP.

E com a crescente demanda das organizações por sistemas tecnológicos surge a necessidade de sistemas automatizados para o gerenciamento de pedidos nas empresas de fast-food.

LOGÍSTICA DE PEDIDOS E AUTOMAÇÃO

Logística

A logística, de um modo geral, trata do produto desde seus primórdios passando informação e constituindo um fluxo de informações que vai do produtor ao consumidor, final

ou intermediário, levando o produto ou serviço para o cliente final independente onde ele esteja com o máximo de qualidade, seja para empresas já atuantes no mercado ou novas empresas que estão nascendo agora. MOURA (2006)

Com o nascimento de uma empresa vêm junto a necessidade de atrair novos clientes e manter o bom atendimento com os já existentes, com isso se dá o conceito de logística empresarial que Ballou (2009) descreve como um estudo administrativo que visa a distribuição de serviços e consumidores de modo que tenham seus bens e/ou serviços onde desejarem e quando desejarem.

Nas palavras de Bowersox et al. (2014) a logística começa na gestão do pedido e vai até o material da embalagem, visando uma sincronização perfeita para um melhor atendimento do cliente, no entanto conseguir tal feito é difícil, algumas vezes impossível, porém quando uma organização chega a dominar completamente a logística, de seu patamar inicial até a entrega do produto ou serviço ao cliente, consegue-se um destaque frente aos seus concorrentes que dificilmente vão conseguir realizar tal ato.

Uma vez que, dentro das necessidades dos clientes por produtos e serviços se tem a elaboração dos pedidos, nos quais consistem na formalização de uma necessidade/desejo a um determinado fornecedor, o qual deve elaborar um plano de entrega para seu cliente para obter o máximo de satisfação do mesmo.

Logística de Pedidos

Para ter uma satisfação do seu cliente necessita-se de uma logística de pedidos junto com um sistema automatizado que deve entregar ao seu cliente um produto de qualidade, que nas palavras de Ballou (2009) um produto pode ser do tipo intangível ou físico, em qualquer um dos produtos o mesmo deve apresentar características relevantes para seu cliente, no caso do produto intangível ele deve apresentar fatores como qualidade, conveniência e descrição, já no físico os também existem fatores como peso, volume, forma.

Ainda segundo o autor, qualquer produto que se adquira têm-se uma classificação, os produtos são classificados em bens de consumo e bens industriais, os bens de consumo, por sua vez, são separados em bens de convivência, uso especial e comparação, neste contexto os bens de convivência são os mais procurados pelos clientes, são aqueles bens que os clientes não pesquisam muito antes de comprar, são os bens que eles compram com mais frequência, normalmente focados em bens alimentícios, que por sua vez possuem um ciclo de vida, sendo

que não se vende o volume máximo no ato do lançamento, tendo que ter um controle de volume de produtos disponíveis para os clientes.

Diversas empresas pecam na entrega de seus produtos ou serviços gerando reclamações e perdendo vários clientes com esse critério, um dos principais motivos é o atraso nas entregas outro é produtos que chegam danificados na casa do cliente, contudo o bom gerenciamento dos pedidos pode ser feito com entregas pontuais, boas condições do produto e documentação de entrega impecável garantindo ao cliente a entrega no tempo mais rápido possível com um produto em boa qualidade e em boas condições de consumo. BOWERSOX (2014).

Para que um cliente receba seu produto com qualidade, Gomes (2004) ressalta que uma boa entrega de produto ou serviço necessita de diversos fatores que implicam em uma entrega ao destinatário com sucesso, tais como densidade, tamanho, grau de fragilidade, elaboração de um plano de entrega, tempo estimado para entrega, outros fatores de risco que podem ocorrer no transporte, contudo com uma boa logística consegue-se alcançar um grande número de clientes totalmente satisfeitos com suas encomendas.

Um bom diferencial junto a concorrência é a embalagem que leva seu produto, onde Ballou (2009) ressalta que existe vários tipos de embalagens, e que cada uma leva uma característica específica, já no ramo alimentício as características marcantes são em chamar a atenção do seu cliente pelo jeito da embalagem, visando em total proteção do produto que a embalagem carrega.

Para se chegar a uma boa logística com os pedidos surge-se a necessidade de um sistema informatizado que vai automatizar e controlar todas as informações pertinentes a cada pedido e cliente.

Processos Informatizados

No período da revolução industrial a grande maioria dos processos das empresas eram feitos de forma manual, pelo homem, e com a chegada da tecnologia da informação nas indústrias os processos começaram a ser informatizados, começaram a ser realizados totalmente ou parcialmente por uma máquina, previamente controlada por um ser humano, gerando uma passagem da sociedade industrial para uma sociedade de informação, e nessa sociedade que se iniciou as pequenas empresas começaram a informatizar processos, procurando uma passagem do trabalho manual para um sistema computacional gerando

facilidade e melhor controle das informações dentro da empresa uma vez que a informação é um bem de altíssimo valor dentro da empresa. (SOUZA, 2004).

Por esse motivo necessita-se desenvolver um sistema automatizado que ajude o gerenciamento de entregas das empresas de fast-food.

Automação

A palavra automação vem do latim Automatus e tem como significado “mover-se por si” que se resume em um processo anteriormente feito manualmente que foi substituído por um mecanismo ou aplicação que a executa de forma mais automática, sendo que por outro lado pode ser definido como automação um processo feito por ser humano de forma repetitiva e com o auxílio de alguma tecnologia tornar esse processo repetitivo uma rotina de uma aplicação, melhorando o desempenho do trabalhador. Assim, a automação ganhou relevância nos anos de 1949 e 1950 quando uma máquina realizava operações previamente programadas dispensando a intervenção direta do homem, e a palavra automação ganhou mais força com a revolução industrial onde grandes máquinas automatizaram setores de produção mesmo sendo muito frágeis e necessitando de manutenção constante. ROSÁRIO (2009).

Assim nas palavras de Ribeiro (2001), sistema automatizado é um sistema que funciona a partir de impulsos elétricos ou magnéticos, contendo pouca interação humana, agilizando os processos que originalmente era feito sem auxílio de um sistema.

Sabe-se que a automação é de grande importância para a melhoria das operações logísticas, nas quais utilizam de hardware e software interligados para uma melhor otimização e agilidade dos processos.

FERRAMENTAS PARA O GERENCIAMENTO DE PEDIDOS

Linguagem de Programação

No desenvolvimento de um sistema informatizado utiliza-se uma linguagem de programação que segundo Gotardo (2015), é um conjunto de padrões que se usa para passar informações para um computador programável que inclui regras sintáticas e semânticas, sendo que as regras de sintática se diz respeito ao modo de escrita já a semântica ao conteúdo em si, com a escolha de uma linguagem de programação você pode especificar quais dados

um computador pode usar, como serão tratados, armazenados e transmitidos, quando se faz o uso de uma linguagem de programação é gerado um código fonte que contém um conjunto de palavras obedecendo as regras de sintática e semântica da linguagem em questão.

Uma linguagem de programação é composta por um algoritmo, conjunto de instruções bem detalhadas que vão levar a um determinado fim e todo algoritmo possui características que são indispensáveis como ter fim, não possui ambiguidade, receber e gerar informações, algoritmos não se voltam somente para o uso computacional, eles também pode ser definidos como receitas de bolo, como trocar um pneu de carro, em suma tudo que possua um passo a passo de como se fazer junto com informações que precisam ser preenchidas para chegar a um fim desejado. (EGYPTO, 2004).

Java

Em linguagem de programação existe diversas linguagens tais como C, C++, C#, Java, Python entre outras, contudo a linguagem Java vem se destacando há anos e segundo Dietel (2010), Java foi uma linguagem de programação criada em 1991 porém a demanda do mercado não foi tão vantajosa quanto a Sun Microsystems estava esperando, contudo houve uma explosão da web fazendo o Java ganhar mercado novamente, foi então em 1995 que, em uma conferência, o Java foi anunciado oficialmente pela Sun. Java foi originalmente criado para o desenvolvimento do projeto de um controle remoto, Luckow e Melo (2010) relata que o primeiro projeto em Java foi para um controle remoto chamado de *7 (StarSeven) que possuía uma interface gráfica sensível ao toque e que conseguia interagir com diversos equipamentos, a linguagem em questão surgiu da necessidade de controlar internamente o equipamento.

Java como muitas outras linguagens é orientada a objetos Barnes e Kolling (2009) ressalta que classe é um conjunto de objetos generalizando uma parte de codificação, e os objetos são dispostos dentro das classes, um exemplo em seu livro é o carro que fica no lugar da classe, um carro generalizado, e as suas características ficam como os objetos cor, potência, velocidade máxima.

NetBeans

Para se desenvolver em qualquer linguagem de programação necessita-se de um ambiente de desenvolvimento conhecido como IDE, logo temos o NetBeans que Luckow e

Melo (2010) nos mostra que é uma IDE de desenvolvimento oficial da Sun Microsystems que permite o desenvolvimento de aplicações em diversas linguagens de programação, o NetBeans é a IDE oficial para do desenvolvimento em Java tornando o trabalho com a linguagem mais satisfatório, contudo existem ferramentas de desenvolvimento que são criadas por instituições privadas tais como o Eclipse, Apache Tomcat e MySQL que a comunidade contribui constantemente para sua melhoria.

Eclipse

Já o eclipsei desenvolvido pela IBM e posteriormente foi entregue a comunidade de desenvolvedores tornando-se uma IDE de código aberto, quando se baixa ela já vem configurada para a programação em Java, porém por ser de código aberto veio a criação de plug-ins, feitos pela comunidade, para o desenvolvimento nas mais diversas linguagens de programação, e juntamente com o Eclipse deve-se usar um servidor de aplicação para execução de projetos, um desses serviços é o Apache Tomcat que é um contêiner e um servidor de aplicações web que suporta a execução de Java Servlet e JavaServer Pages permitindo assim o Java funcionar em ambiente web. (LUCKOW E MELO 2010).

Java Server Faces – JSF

Falando em desenvolvimento web contando com o auxílio do Eclipse e do servidor de aplicações Apache Tomcat podemos ainda, utilizar ferramentas que auxiliam o desenvolvimento web como o JavaServer Faces (JSF) que Luckow e Melo (2010) ressalta que o JSF é bem visto no mercado pelas grandes empresas tais como IBM, Macromedia, Oracle o que possibilita o desenvolvimento de mais ferramentas que trabalhem com JSF, com o JSF o desenvolvimento web fica facilitado pelo fato dele investir em um modo de desenvolvimento mais agradável, onde o desenvolvedor, literalmente, arrasta e solta objetos na montagem de telas web.

O JSF cria códigos em um arquivo xhtml e JSP que permite a integração com uma página WEB, e o JSF é bem diferente de seus antecessores como o Struts, no JSF faz-se o uso de Taglibs, as mesmas são responsáveis por toda a criação da tela, como botões, caixas de texto, caixas de seleção, menus, e o posicionamento dos mesmos. (CORDEIRO, 2014).

Ainda sobre JSF Faria (2013) nos mostra que é uma tecnologia de interfaces gráficas toda baseada em eventos e no padrão de projetos MVC que divide o projeto em três

partes o modelo, a visualização e o controle, onde o modelo é a parte que recebe os comandos dos usuários interpretando e sabendo de que maneira o software deve agir, é no modelo que o aplicativo sabe o que fazer pois é o mesmo que modela o problema a ser resolvido, já o controle é quem interpreta as entradas do mouse e do teclado vindas do usuário e também é a parte que liga o modelo e a visualização, e por último a visualização é a parte gráfica do projeto onde são exibidas todas as informações necessárias para o usuário, o JSF permite o desenvolvedor uma grande agilidade no projeto pois o mesmo não necessita de criar sua interface gráfica toda do zero.

Em uma aplicação WEB utilizando JSF, necessita-se de um banco de dados para o armazenamento de informações fornecidos pelo usuário do sistema que por sua vez possibilita um melhor controle e diversos relatórios para o auxílio em melhorias dentro da organização.

Banco de Dados

Como toda aplicação necessita de um banco de dados, Date (2003) ressalta em seu livro que um banco de dados é um sistema computadorizado que salva dados pertinentes a uma aplicação, sendo os dados organizados por tabelas e estando dispostos ao usuário de modo a gerar informações em um determinado momento com o comando do usuário, dentro do banco de dados existem diversas tabelas que são interligadas por entidades e relacionamentos a fim de gerar informações.

No levantamento de dados para elaboração do banco de dados o mesmo é separado em modelo conceitual, lógico e físico, sendo o conceitual um modelo de fácil entendimento e abstração de dados pelo cliente, é um modelo onde se mostra quais dados vão conter no banco, porém, não mostra como serão armazenados e nem qual sistema de gerenciamento de banco de dados vai ser usado. (HEUSER, 2009).

Já o modelo lógico é definido como um modelo mais avançado que contém, chaves primárias, controle de acesso, tipo de atributos, tamanho máximo, e suas características e relacionamentos podem mudar de acordo com regras e limites impostos pela tecnologia usada, e por último modelo físico é a codificação em SQL que cria o banco de dados, essa codificação pode mudar de acordo com a tecnologia usada para o gerenciamento do banco. (COUGO, 1997).

Persistência de dados

Como em toda aplicação web, desktop ou móvel necessita de um banco de dados, surge a persistência de dados, que segundo Luckow e Melo (2010) vem trazer que toda aplicação necessita de persistência de dados, necessita de um gerenciamento e uma qualidade de dados, para ter tal persistência usa-se o Hibernate que é uma das ferramentas para tratar de persistências de dados que existem no mercado e a mais utilizada hoje.

O Hibernate é uma ferramenta em Java, que auxilia os desenvolvedores em seus projetos, onde a ferramenta elimina o uso excessivo de codificação SQL, tornando o desenvolvimento mais rápido e prático, porém o uso da ferramenta não é aconselhável para qualquer aplicação principalmente em sistemas onde se usa triggers e códigos SQL mais complexos, com o a ferramenta o desenvolvedor mapeia, controla, inseri, deleta, modifica todos os dados pertinentes do banco. (LINHARES, 2012).

Com toda informação ressaltada, percebe-se que um projeto de um sistema de informação necessita de muitos recursos, não sendo só recursos humano ou recursos tecnológicos, necessita em si uma junção dos dois recursos trabalhando de forma unida com o objetivo de alcançar o final do projeto com todos os requisitos atendidos.

No decorrer do desenvolvimento desse projeto será utilizado documentos para total controle do mesmo, que se encontraram em apêndice, entre esses documentos encontra-se, termo de abertura do projeto, termo de encerramento, casos de uso, modelo conceitual e lógico e físico do banco de dados, telas do software.

CONCLUSÕES

O presente projeto trata-se de um sistema para auxiliar o gerenciamento de pedidos nas empresas de fast-food, onde o mesmo conta com cadastros diversos junto com a emissão de um relatório de entrega, salvamento de dados referentes as entregas e emissão de relatórios de entregas por data.

Dentro do desenvolvimento do projeto foi levantado objetivos para serem descritos tais como conceituação de sistemas de informação e projeto de sistemas, logística de pedidos e automação, ferramentas necessárias para o desenvolvimento, todos os objetivos foram levantados e exposto o motivo de sua utilidade, sendo que os objetivos levantados foram suficientes para a realização do projeto, não necessitando de outros objetivos.

Junto aos objetivos se têm as hipóteses do projeto, sendo uma hipótese A favorável à conclusão do projeto, atendendo a todos os requisitos e objetivos, e uma hipótese B, negativa, que por algum fator o projeto não pode ser terminado.

Diante disto o presente se confirma com a hipótese A que se encontra no projeto, onde o mesmo foi realizado com sucesso, atendendo todas as expectativas e objetivos.

O projeto realizado supre as necessidades descritas no começo, e o problema a ser solucionado foi resolvido, tendo em vista que os pedidos realizados a partir do Delivery System oferecem um melhor controle de clientes e pedidos.

Este trabalho foi só o começo, pelo fato de toda empresa necessitar de automação em seus processos, o processo de pedidos foi o primeiro, abrindo um leque de oportunidades para demais automações em processos que ainda não o possuem, fica aberto esse trabalho a possíveis atualizações como novas implementações e melhorias nos processos existentes.

Este trabalho mostra que cada etapa de uma empresa pode ser auxiliada, de algum modo, por uma tecnologia junto com uma boa logística, podendo assim, agradar mais o cliente e ter um retorno muito mais favorável a organização.

REFERÊNCIAS

BALBÉ, Marília. **Levantamento de Requisitos: Você sabe o que é?** Disponível em <<https://www.profissionaisti.com.br/2011/06/levantamento-de-requisitos-voce-sabe-o-que-e/>>. Acesso em 17 de setembro de 2017.

BALLOU, Ronald H. **Logística Empresarial**. 1 ed. São Paulo: Atlas, 2009.

BARNES, David J., KOLLING, Michael. **Programação Orientada a Objetos com Java**. 4 ed. São Paulo: Pearson Prentice Hall, 2009.

BOWERSOX, Donald J. et al. **Gestão logística da cadeia de suprimentos**. 1 ed. Porto Alegre: AMGH Editora, 2014.

CAMPOS, Luiz Fernando Rodrigues, BRASIL, Caroline V. de Machado. **Logística, Teia de Relações**. 1 ed. Curitiba: Ibpex, 2007.

COUGO, Paulo Sérgio. **Introdução ao Hibernate 3**. Disponível em <http://www.guj.com.br/content/articles/hibernate/intruducacao_hibernate3_guj.pdf>. Acesso em 15 de novembro de 2017.

CRUZ, Fábio. **Scrum e PMBOK unidos no Gerenciamento de Projetos**. Rio de Janeiro: Brasport, 2013.

- DATE, Christopher J. **Introdução a sistemas de bancos de dados**. 8 ed. Rio de Janeiro: Elsevier, 2004.
- DEITEL, Paul, DEITEL, Harvey. **Java Como Programar**. 8 ed. São Paulo: Pearson Prentice Hall, 2010.
- DEMO, Pedro. **Pesquisa e construção de conhecimento**. Rio de Janeiro: Tempo Brasileiro, 1996.
- EGYPTO, CÂNDIDO. **Introdução à Programação**. João Pessoa: ASPER, 2004.
- GIL, Antônio Carlos. **Como Elaborar Projetos de Pesquisa**. 5 ed. São Paulo: Atlas, 2010.
- GOMES, Carlos Francisco Simões. RIBEIRO, Priscilla Cristina Cabral. **Gestão da Cadeia de Suprimentos Integrada à Tecnologia da Informação**. São Paulo: Pioneira Thomson Learning, 2004.
- GORTARDO, Reginaldo. **Linguagem de Programação I**. 1 ed. Rio de Janeiro: Seses, 2015.
- HEUSER, Carlos Alberto. **Projeto de banco de dados: Volume 4 da Série Livros didáticos informática UFRGS**. 6ª ed. Porto Alegre: Bookman Editora, 2009.
- KERZNER, Harold. **Gestão de Projetos: As Melhores Práticas**. 3 ed. São Paulo. Bookman, 2016.
- KUHN, Giovane Roslindo; PAMPLONA, Vitor Fernando. **Apresentando XP. Encante seus clientes com Extreme Programming**. Blumenal, São Paulo. Universidade Regional de Blumenal, 2004.
- LUCKOW, Décio Heinzemann, MELO, Alexandre Altair de. **Programação Java Para Web**. 1 ed. São Paulo: Novatec Editora, 2010
- MENEZES, Luís Cesar de Moura. **Gestão de Projetos**. 3 ed. São Paulo: Atlas, 2009.
- PRESSMAN, Roger S. **Engenharia de Software**. 1 ed. São Paulo: Pearson Makron Books, 1995.
- PRESSMAN, Roger S. MAXIM, Bruce R. **Engenharia de Software. Uma Abordagem Profissional**. 8 ed. São Paulo: AMGH Editora LTD, 2016.
- PRODANOV, Cleber Cristiano; DE FREITAS, Ernani Cesar. **Metodologia do Trabalho Científico: Métodos e Técnicas da Pesquisa e do Trabalho Acadêmico-2ª Edição**. Rio Grande do Sul, Editora Feevale, 2013.
- QUITERIO. Ana Paula. **Análise de Requisitos**. Disponível em < <http://www.infoescola.com/engenharia-de-software/analise-de-requisitos/> >. Acesso em 17 de setembro de 2017.

RIBEIRO, Marcos Antônio. **Automação Industrial**. 4 ed. Salvador: Naval, 2001.

ROSÁRIO, João Maurício. **Automação Industrial**. 1ª ed. São Paulo: Baraúna, 2009.

SOARES, Michel dos Santos. **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software**. INFOCOMP, 2004.

SOMMERVILLE, Ian. **Engenharia de Software**. 9 ed. São Paulo: Pearson Prentice Hall, 2011.

SOUZA, Cesar Alexandre de. **Uso organizacional da tecnologia de informação: um estudo sobre a avaliação do grau de informatização de empresas industriais paulistas**. 2004. Tese de Doutorado. Universidade de São Paulo.

SUTHERLAND, Jeff. **Scrum. A arte de fazer o dobro do trabalho na metade do tempo**. 2 ed. São Paulo: Leya, 2016.

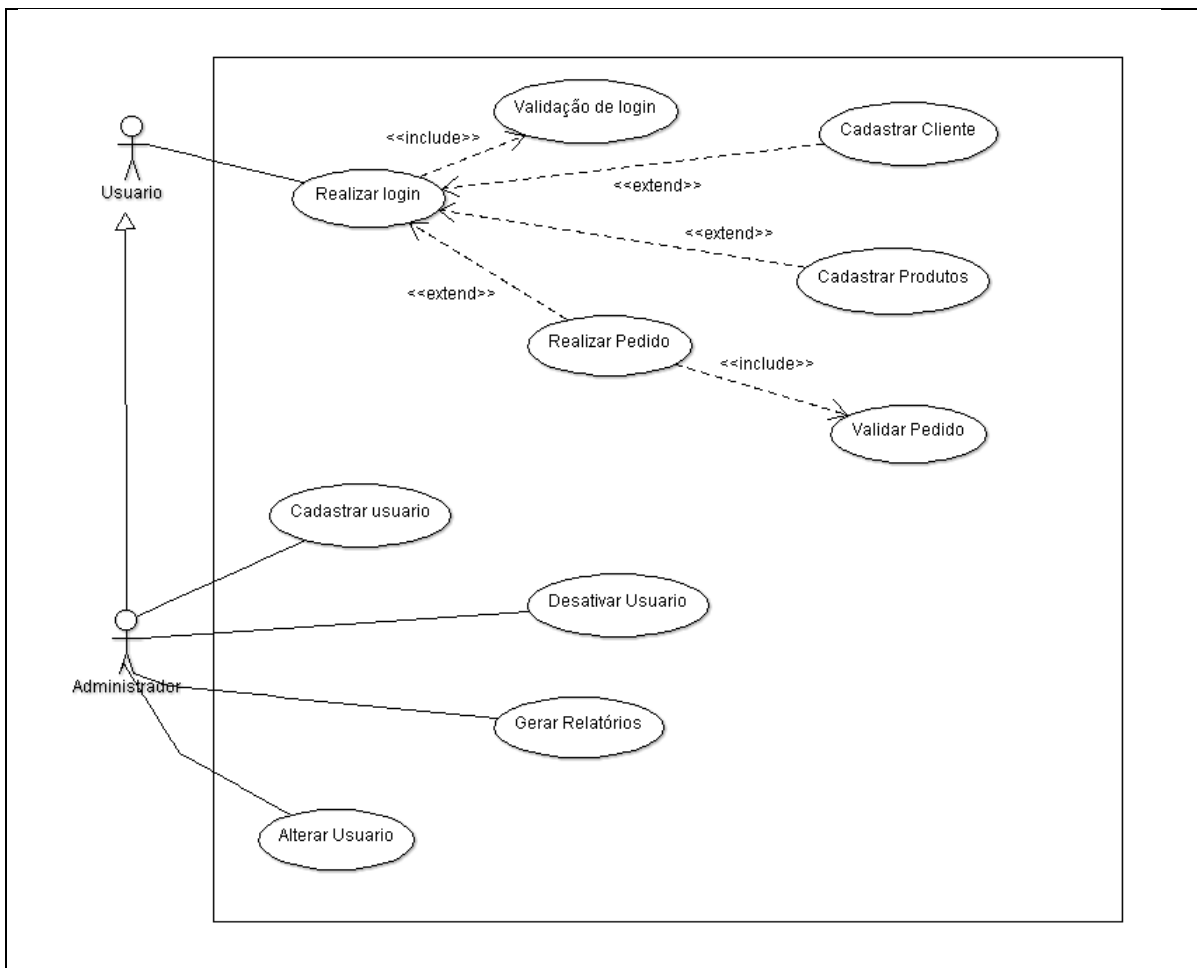
TELES, Vinícius Manhães. **Extreme Programming. Aprenda como encantar seus usuários desenvolvendo software com agilidade e alta qualidade**. 2 ed. São Paulo: Novatec Editora, 2017.

TURBAN, Efrain, RAINER JR., R. Kelly, POTTER, Richard E. **Introdução a Sistemas de Informação. Uma Abordagem Gerencial**. 1 ed. Rio de Janeiro: Elsevier, 2007.

BALTZAN, Paige; PHILLIPS, Amy. **Sistemas de informação**. São Paulo: AMGH Editora, 2012.

APÊNDICE A – Diagrama de caso de uso

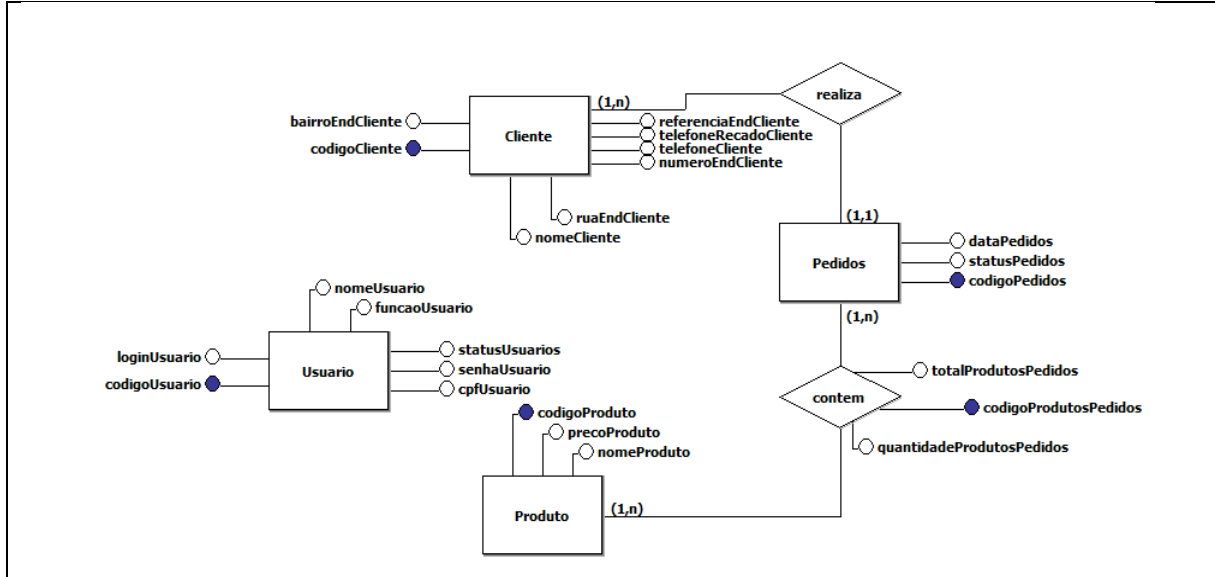
Figura 1 - Diagrama de caso de uso



Fonte: Elaborado pelo autor

APÊNDICE B – Modelo conceitual de dados

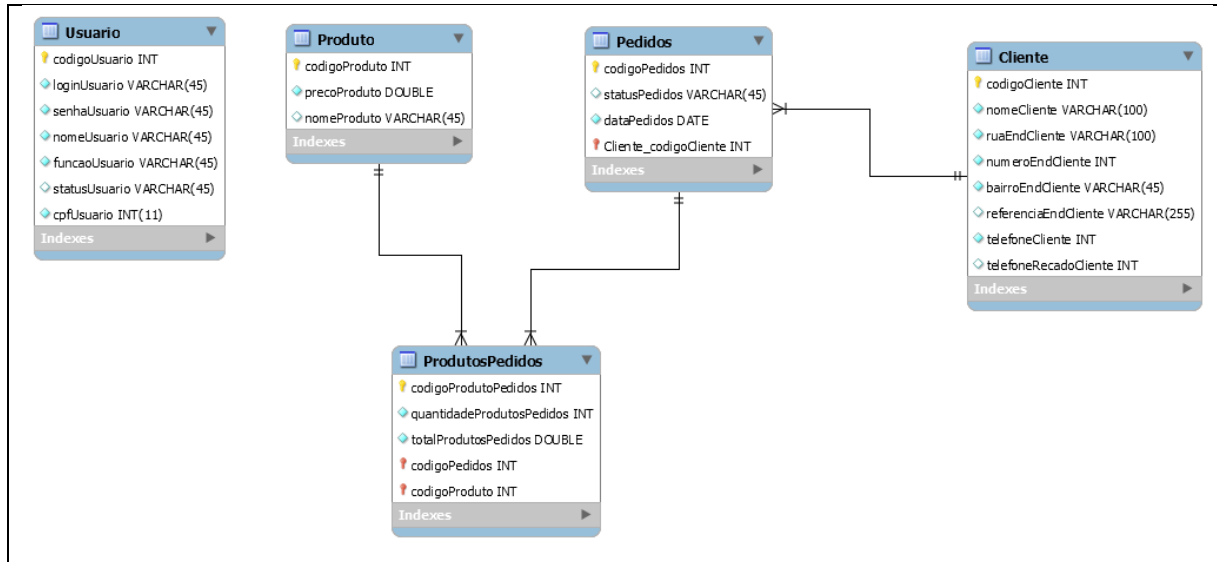
Figura 2 - Modelo conceitual de dados



Fonte: Elaborada pelo autor

APÊNDICE C – Modelo lógico de dados

Figura 3 - Modelo lógico de dados



Fonte: Elaborada pelo autor



APÊNDICE D – Código SQL gerado de Hibernate

Hibernate:

```
create table Cliente (  
    codigoCliente integer not null,  
    bairroEndCliente varchar(255),  
    nomeCliente varchar(255),  
    numeroEndCliente varchar(255),  
    referenciaEndCliente varchar(255),  
    ruaEndCliente varchar(255),  
    telefoneCliente integer,  
    telefoneRecadoCliente integer,  
    primary key (codigoCliente)  
    ) engine=MyISAM
```

Hibernate:

```
create table hibernate_sequence (  
    next_val bigint  
    ) engine=MyISAM
```

Hibernate:

```
insert into hibernate_sequence values ( 1 )
```

Hibernate:

```
insert into hibernate_sequence values ( 1 )
```

Hibernate:

```
insert into hibernate_sequence values ( 1 )
```

Hibernate:

```
insert into hibernate_sequence values ( 1 )
```

Hibernate:

```
insert into hibernate_sequence values ( 1 )
```

Hibernate:

```
create table Pedidos (  
    codigoPedidos integer not null,  
    dataPedidos date,  
    statusPedidos varchar(255),  
    codigoClientes integer,  
    primary key (codigoPedidos)  
    ) engine=MyISAM
```

Hibernate:

```
create table Produto (  
    codigoProduto integer not null,  
    nomeProduto varchar(255),  
    precoProduto double precision,  
    primary key (codigoProduto)  
    ) engine=MyISAM
```

Hibernate:



```
create table ProdutosPedidos (  
  codigoProdutoPedidos integer not null,  
  quantidadeProdutosPedidos integer,  
  totalProdutosPedidos double precision,  
  codigoPedido integer,  
  codigoProduto integer,  
  primary key (codigoProdutoPedidos)  
) engine=MyISAM
```

Hibernate:

```
create table Usuario (  
  codigoUsuario integer not null,  
  cpfUsuario bigint,  
  funcaoUsuario varchar(255),  
  loginUsuario varchar(255),  
  nomeUsuario varchar(255),  
  senhaUsuario varchar(255),  
  statusUsuario varchar(255),  
  primary key (codigoUsuario)  
) engine=MyISAM
```

Hibernate:

```
alter table Pedidos  
  add constraint FKifbsn52oj9c2kv4e8mh816v7l  
  foreign key (codigoClientes)  
  references Cliente (codigoCliente)
```

Hibernate:

```
alter table ProdutosPedidos  
  add constraint FK976yh0fgayx87mmj2e5cjvaoi  
  foreign key (codigoPedido)  
  references Pedidos (codigoPedidos)
```

Hibernate:

```
alter table ProdutosPedidos  
  add constraint FKj9jf8t77g0y2pghh4qmt7mv7y  
  foreign key (codigoProduto)  
  references Produto (codigoProduto)
```

APÊNDICE E – Termo de Abertura do Projeto

Nome do Projeto: Delivery System

Data de Elaboração: 13/02/2017

1 Escopo preliminar do projeto

1.1 Escopo do produto

O presente projeto devera possuir as seguintes funcionalidades: cadastrar usuários com nível de acesso, cadastro de cliente contendo endereço completo, cadastro de produtos com respectivo preço, e realização de pedidos com emissão de relatório de entrega, emissão de relatórios de vendas por data.

1.2 Objetivos, premissas e restrições do projeto

Desenvolver um software para gerenciar os pedidos em empresas de fast-food de forma mais rápida e agradável ao cliente com telas amigáveis e com boa segurança. Para a elaboração do projeto será necessário a coleta de dados que pelos documentos de requisitos funcionais e não funcionais. A plataforma que será utilizada é uma ferramenta feita pela comunidade de desenvolvedores, Eclipse, em linguagem Java junto com o Primefaces, Bootsfaces, JSF, Hibernate, um servidor de aplicação, para execução dos códigos, Tom Cat 9.0. O presente projeto tem data final para entrega, sendo o mesmo dia 24/12/2017, onde após esse dia o presente não será recebido.

2 Equipe preliminar

Esse projeto será todo elaborado e implementado por Gustavo Henrique Borges Rosa.

3 Riscos preliminares

Coleta de informações com falhas;



Não conseguir terminar o projeto dentro do prazo específico.

Falha nos dispositivos físicos usados no desenvolvimento;

Desastres naturais;

4 Aprovação

Data de Aprovação: 13/02/2017

Sérgio Augusto De Souza Moraes

APÊNDICE F – Requisitos Funcionais

Tabela 1 - Requisitos Funcionais

Requisito	Descrição	Requisito não Funcional
RF-01	O usuário realiza <i>login</i>	RNF-01
RF-02	O usuário pode cadastrar produto	RNF-02
RF-03	Só um administrador pode cadastrar usuário	RNF-03, RNF-04 E RNF-05
RF-04	Todo pedido gera um relatório de entrega	RNF-06
RF-05	Usuário realiza pedidos	RNF-07
RF-06	Só um administrador pode gerar relatórios	RNF-08
RF-07	Só um administrador pode desativar um usuário	RNF-09
RF-08	Só um administrador pode alterar dados de um usuário	RNF-10
RF-09	Usuário pode cadastrar cliente	RNF-11

Fonte: Elaborado Pelo Autor

APÊNDICE G – Requisitos não Funcionais**Tabela 2 - Requisitos não Funcionais**

Requisito	Descrição
RNF-01	LOGIN deve ser autenticado
RNF-02	Não pode se ter dois produtos com o mesmo nome
RNF-03	Usuário deve ter CPF válido
RNF-04	Não pode ter dois Logins iguais no banco de dados
RNF-05	Usuário administrador tem permissões a mais que usuário comum
RNF-06	Relatório de entrega pode ser impresso
RNF-07	Pedidos devem conter <i>status</i> de finalização
RNF-08	Somente relatório de vendas
RNF-09	Usuários não devem ser excluídos, e sim desativados
RNF-10	Usuários comuns não tem acesso a alteração de dados cadastrais
RNF-11	Cliente deve conter endereço completo junto com referência

Fonte: Elaborado Pelo Autor

APÊNDICE H – Telas do Sistema

Figura 4 - Tela de Login

Delivery System

Usuário

Senha

Esqueceu sua senha? Redefinir senha agora

Entrar

Cadastrar

Fonte: Elaborado Pelo Autor

Figura 5 - Tela de Cadastro de Usuário

Início Cliente Produtos Usuários Pedidos Sair

Nome:

CPF:

Função:
ADMINISTRADOR

Login:

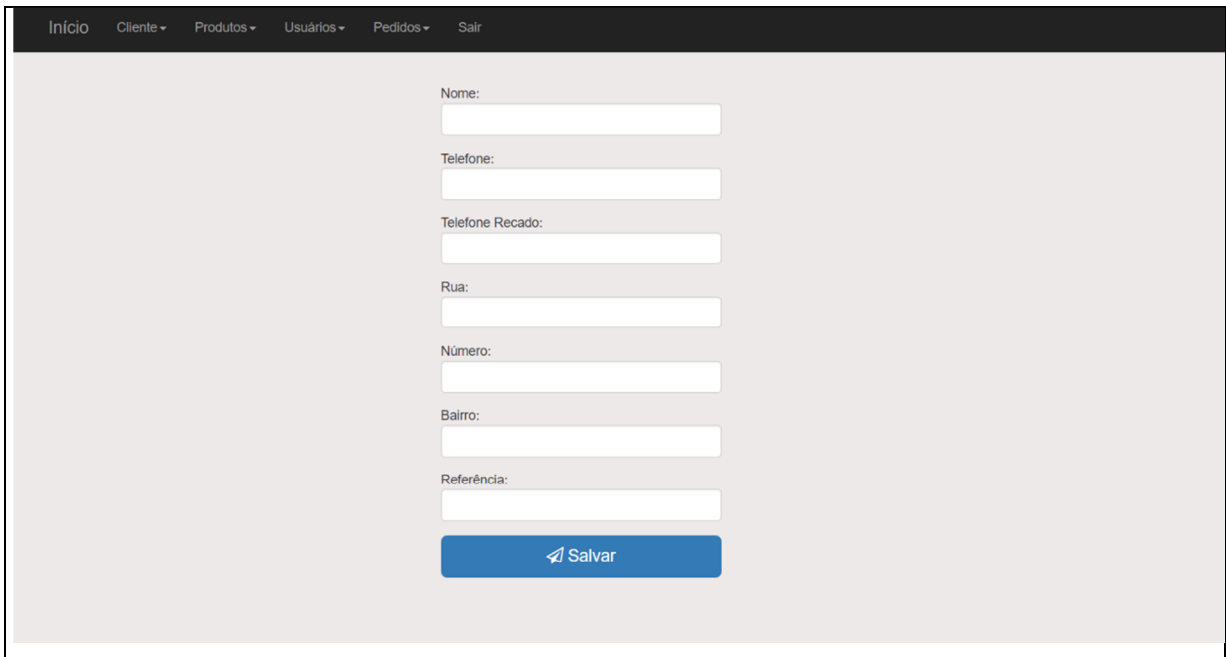
Senha:

Confirmação de Senha:

Salvar

Fonte: Elaborado Pelo Autor

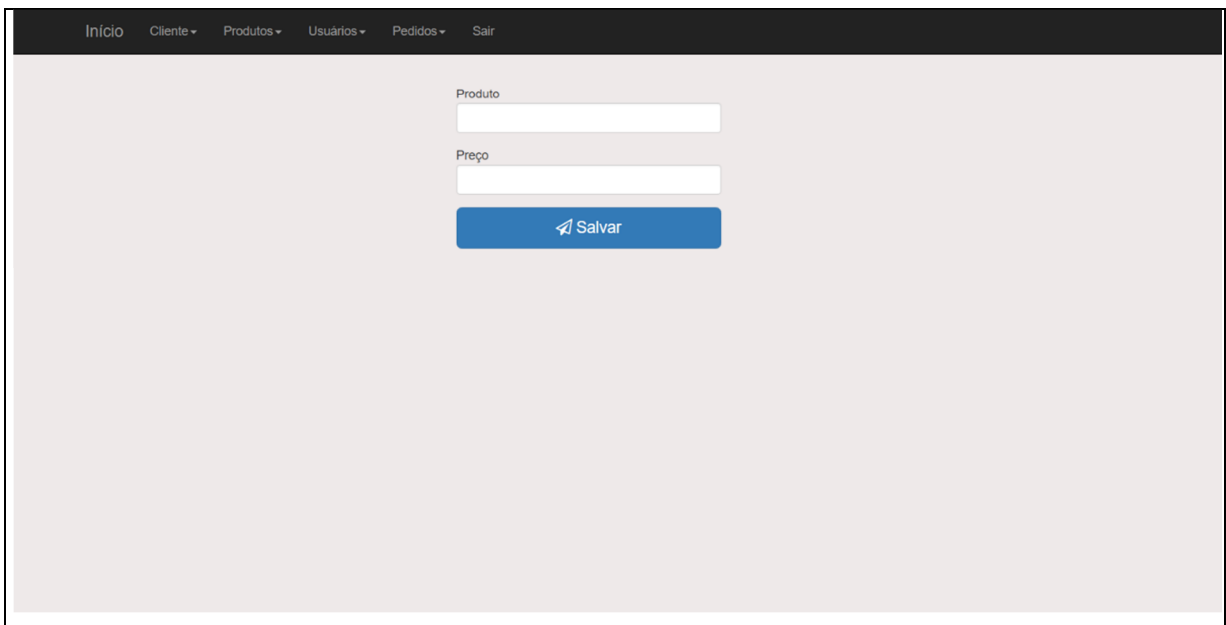
Figura 6 - Tela de Cadastro de Cliente



The screenshot shows a web application interface for client registration. At the top, there is a navigation bar with the following menu items: Início, Cliente, Produtos, Usuários, Pedidos, and Sair. The main content area contains a form with the following fields: Nome, Telefone, Telefone Recado, Rua, Número, Bairro, and Referência. Each field is represented by a white text input box. Below the input boxes is a blue button with a white arrow icon and the text "Salvar".

Fonte: Elaborado Pelo Autor

Figura 7 - Tela de Cadastro de Produtos



The screenshot shows a web application interface for product registration. At the top, there is a navigation bar with the following menu items: Início, Cliente, Produtos, Usuários, Pedidos, and Sair. The main content area contains a form with the following fields: Produto and Preço. Each field is represented by a white text input box. Below the input boxes is a blue button with a white arrow icon and the text "Salvar".

Fonte: Elaborado Pelo Autor

APÊNDICE I – Termo de Enceramento do Projeto

Nome do projeto: *Delivery System*

Período de execução: de 13/02/2017 até 24/11/2017

1 Produtos entregues

Um software com as seguintes especificações: manter usuários, manter pedidos, manter produtos e gerar relatórios.

2 Avaliação do projeto pela equipe

Para os usuários foi de enorme valor, pois melhorou o processo de entrega tanto quanto o gerenciamento, possibilitando uma melhor comunicação e atendimento ao cliente, para mim, como desenvolvedor, foi de enorme ganho de conhecimento e aprofundamento no desenvolvimento web, onde adquiri muito conhecimento sobre as ferramentas que estão atuantes no mercado web, e como é grande o mercado de trabalho web.

3 Aceite do projeto

Confirmando o recebimento da configuração de software descrita neste documento e declaro que esta configuração atende plenamente os objetivos definidos para o projeto. Portanto considero o projeto concluído com sucesso, e assumo a partir desta data, a posse e propriedade do software produzido no projeto. A partir desta data o desenvolvedor fica liberado de quaisquer obrigações relacionadas com este software, incluindo a manutenção do software, e a transferência de conhecimento técnico necessário para operar, administrar e manter todos os produtos listados neste documento.

Paracatu, 24 de novembro de 2017

Sérgio Augusto De Souza Moraes