

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
COORDENAÇÃO DE LICENCIATURA EM INFORMÁTICA
DESENVOLVIMENTO DE SISTEMAS PARA INTERNET E DIPOSITIVOS MOVEIS**

RAFAEL DE AMARAL

SISTEMA WEB PARA PEDIDOS DE DELIVERY

TRABALHO DE CONCLUSÃO DE CURSO

FRANCISCO BELTRÃO

2014

RAFAEL DE AMARAL

SISTEMA WEB PARA PEDIDOS DE DELIVERY

Monografia apresentada como requisito parcial à obtenção do título de Especialista, da Coordenação de licenciatura em informática, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Me. Gustavo Yuji Sato

FRANCISCO BELTRÃO

2014



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Francisco Beltrão
Diretoria de Pesquisa e Pós-Graduação
Curso de Especialização em Desenvolvimento de Sistemas para Internet
e Dispositivos Móveis

TERMO DE APROVAÇÃO

Dia 20 do mês de novembro de 2014 às: 17 horas, na sala COLIN-Anfiteatro do Campus Francisco Beltrão, realizou-se a apresentação pública da monografia pelo estudante Rafael de Amaral, intitulada “Sistema Web para pedidos de delivery”. Finalizada a apresentação e arguição, a Banca Examinadora declarou aprovada a monografia do estudante, como requisito parcial para obtenção do título de Especialização em Desenvolvimento e Sistemas para Internet e Dispositivo Móveis.

Professor Me. Gustavo Yuji Sato – UTFPR
(Orientador)

Professor Dr. Rafael Wild – UTFPR
(Convidado)

Professor Me. Marcos Mincov Tenorio – UTFPR
(Convidado)

Professor Dr. Ademir Roberto Freddo – UTFPR
(Coordenação)

A folha de aprovação com as rubricas encontram-se disponíveis na Diretoria de Pesquisa e Pós-Graduação, UTFPR, Francisco Beltrão.

AGRADECIMENTOS

Ao professor orientador Prof. Me. Gustavo Yuji Sato pelo apoio e dedicação no período de orientação, onde aceitou me orientar neste trabalho de minha importância.

A minha família que sempre me apoio e incentivou a me dedicar aos estudos, sendo o passo de todo o conhecimento que adquiri.

Aos meus amigos e colegas que me apoiaram a continuar meus estudos, de forma a entender as prioridades do momento.

Por fim a minha namorada Rafaela que sempre esteve ao meu lado, incentivando, apoiando e respeitando o tempo que tive disponível para a realização deste trabalho.

RESUMO

Nos dias de hoje, diversas áreas exploram o uso da tecnologia em suas atividades, seja pela praticidade ou agilidade ou até mesmo pela flexibilidade que proporciona para os usuários que a usufruem. Entregas de lanches crescem cada vez mais com a abertura de novos estabelecimentos. Tendo como objetivo gerenciar isso, o presente trabalho irá propor a ideia de um site que facilitará o monitoramento de pedidos e até mesmo o controle das entregas, permitindo assim, que os clientes do estabelecimento realizem seus pedidos no momento desejado, sem necessidade de ligações para solicitar o mesmo, ou seja, o site permitirá que o pedido seja realizado através da internet sem nenhuma intervenção do funcionário do estabelecimento.

Palavras-Chave: Venda. Pedido. Entrega. Estabelecimento.

ABSTRACT

Today, several areas explore the technology to use in their activities, either by convenience or agility or even by flexibility for users to enjoy. Fast food delivery grow each more time with the opening of new establishments. In order to manage this, the paper will propose the idea of a site that will facilitate the monitoring of applications and even the control of deliveries, enabling customers to the establishment realize their orders at the desired time, without need to calls them, that is, the site will allow the application to be realized through the internet without any intervention of the official establishment.

Keywords: Sale. Request. Delivery. Establishment.

LISTA DE FIGURAS

Figura 1 - URL comum	18
Figura 2 - URL pelo PrettyFaces	18
Figura 3 – Funcionalidades do sistema por usuário.	18
Figura 4 - Site iFood	21
Figura 5 - Site Papa Johns	22
Figura 6 - Modelo Diagrama Caso de uso.....	24
Figura 7 - Exemplo Diagrama de Classe.....	26
Figura 8 - Exemplo Diagrama ER.....	27
Figura 9 – Digrama de caso de uso do sistema	39
Figura 10 - Diagrama de classe de análise do sistema	51
Figura 11 - Diagrama ER do sistema	52
Figura 12 - Interface acesso usuário Master	54
Figura 13 - Interface pesquisa de cardápio	54
Figura 14 - Interface cadastro de cardápio.....	55
Figura 15 - Interface de pesquisa de estabelecimento.....	55
Figura 16 - Interface cadastro de estabelecimento	56
Figura 17 - Interface imagem estabelecimento	56
Figura 18 - Interface perfil acesso administrador	57
Figura 19 - Interface Abertura/Fechamento de estabelecimento.....	57
Figura 20 - Interface pesquisa de usuários	58
Figura 21 - Interface manutenção de usuário.....	58
Figura 22 - Interface de pesquisa de ingredientes	59
Figura 23 - Interface de manutenção de ingredientes	59
Figura 24 - Interface de pesquisa de produtos	60
Figura 25 - Interface de manutenção de produto aba produto	61
Figura 26 - Interface de manutenção de produto aba opções.....	62
Figura 27 - Interface de manutenção de produto aba ingredientes	63
Figura 28 - Interface de manutenção de produto aba acrescentar.....	64
Figura 29 - Interface perfil de acesso funcionário e entregador	64
Figura 30 - Interface de acompanhamento de pedido pelo estabelecimento	65

Figura 31 - Interface de visualização de pedido na listagem pelo estabelecimento ..	66
Figura 32 - Interface cadastro usuário cliente	67
Figura 33 - Interface perfil de acesso cliente.....	67
Figura 34 - Interface de listagem de estabelecimento por cardápio	68
Figura 35 - Interface de listagem de produto por estabelecimento.....	68
Figura 36 - Interface de visualização de produto.....	69
Figura 37 - Interface de carrinho de compra passo pedido	70
Figura 38 - Interface de carrinho de compra passo finalizar pedido.....	71
Figura 39 - Interface de carrinho de compra passo concluído	71
Figura 40 - Interface de acompanhamento de pedido pelo cliente.....	72
Figura 41 - Interface de visualização de pedido na listagem pelo cliente.....	72
Figura 42 - Interface de visualização de pedido	73

LISTAGEM DE QUADROS

Quadro 1 – Estrutura do trabalho	14
Quadro 2 - Modelo Requisito Funcional	23
Quadro 3 - Modelo Requisitos Gerais	23
Quadro 4 - Modelo Caso de Uso	25
Quadro 5 - Requisito Funcional autenticar usuário	29
Quadro 6 - Requisito Funcional Gerenciar Estabelecimento.....	30
Quadro 7 - Requisito Funcional Definir Usuário	31
Quadro 8 - Requisito Funcional Definir Cardápio	32
Quadro 9 - Requisito Funcional Definir Ingrediente.....	32
Quadro 10 - Requisito Funcional Promover Produto.....	33
Quadro 11 - Requisito Funcional Controlar Funcionamento Estabelecimento	34
Quadro 12 - Requisito Funcional Realizar Pedido.....	35
Quadro 13 - Requisito Funcional Acompanhar Pedido	37
Quadro 14 - Requisito Funcional Visualizar Pedido	38
Quadro 15 - Requisito Funcional Alterar Pedido	38
Quadro 16 - Requisitos Gerais do Sistema	38
Quadro 17 - Caso de uso UC1 - Autenticar Usuário	40
Quadro 18 - Caso de uso UC2 - Fluxo Básico	41
Quadro 19 - Caso de uso UC3 - Gerenciar Estabelecimento.....	42
Quadro 20 - Caso de uso UC4 - Definir Usuário	42
Quadro 21 - Caso de uso - Definir Cardápio	43
Quadro 22 - Caso de uso UC6 - Definir Ingrediente.....	43
Quadro 23 - Caso de uso UC7 - Promover Produto.....	44
Quadro 24 - Caso de uso UC8 - Controlar Funcionamento Estabelecimento	44
Quadro 25 - Caso de uso UC9 - Realizar Pedido.....	48
Quadro 26 - Caso de uso UC10 - Acompanhar Pedido	49
Quadro 27 - Caso de uso UC11 - Visualizar Pedido	50
Quadro 28 - Caso de uso UC12 - Alterar Pedido	51

LISTA DE SIGLAS

JSF	Java Server Faces
API	Application Programming Interface
JPA	Java Persistence API
EJB	Enterprise JavaBeans
MVC	Model-View-Controller
JDBC	Java Database Connectivity
ORM	Object-Relational Mapping
HTML	HyperText Markup Language
XML	eXtensible Markup Language
URL	Uniform Resource Locator
UML	Unified Modeling Language
PK	Primary Key
UK	Unique Key
FK	Foreign Key

SUMÁRIO

1 INTRODUÇÃO	12
1.1 PROBLEMA	12
1.2 JUSTIFICATIVA	13
1.3 OBJETIVOS	13
1.3.1 Objetivo Geral	13
1.3.2 Objetivos Específicos	13
1.4 METODOLOGIA.....	14
1.5 ESTRUTURA DO TRABALHO.....	14
2 FUNDAMENTAÇÃO TEÓRICA	15
2.1 TECNOLOGIAS	15
2.1.1 EJB.....	15
2.1.2 JPA e Hibernate	16
2.1.3 Java Server Faces e Primefaces.....	17
2.1.4 PrettyFaces	17
2.2 SISTEMA PROPOSTO	18
2.2.1 Restrições do sistema	19
2.2.2 Premissas.....	20
2.2.3 Diferencial de mercado.....	20
2.3 SISTEMAS SIMILARES	20
2.3.1 iFood	20
2.3.2 Papa Johns	21
2.4 METODOLOGIA DE DESENVOLVIMENTO.....	22
2.4.1 Requisitos de Software.....	23
2.4.2 Casos de Uso.....	24
2.4.3 Diagrama de Classes	25
2.4.4 Diagrama ER.....	26
3 DESENVOLVIMENTO	28
3.1 REQUISITOS DE SOFTWARE	28
3.1.1 Requisitos Funcionais	28
3.1.2 Requisitos Gerais	38

3.2 CASOS DE USO	39
3.3 DIAGRAMA DE CLASSE DE ANÁLISE	51
3.4 DIAGRAMA ER	51
4 RESULTADOS.....	53
4.1 APRESENTAÇÃO SISTEMA	53
4.2 USUÁRIO MASTER DO SISTEMA	53
4.3 USUÁRIOS DO ESTABELECIMENTO	57
4.4 USUÁRIO CLIENTE	66
5 CONCLUSÃO	74
5.1 PROBLEMAS ENCONTRADOS	74
5.2 TRABALHOS FUTUROS	75
REFERÊNCIAS.....	76

1 INTRODUÇÃO

O presente capítulo tem como finalidade apresentar o problema que levou o desenvolvimento deste trabalho, a justificativa, os objetivos a serem alcançados, a metodologia utilizada e a estrutura do trabalho.

1.1 PROBLEMA

Com o passar dos anos a tecnologia está mais presente na vida da população mundial, seja ela de qualquer classe social, junto a esse avanço, cresce o acesso à *Internet* e as oportunidades de negócio de *software*.

Muitas pessoas procuram fazer uma programação para sair da rotina do seu dia a dia, uma delas é a opção de adquirir um alimento pronto e diferente. Com isso, empresas de *fast food* (comida rápida) são bastante buscadas por essas pessoas, dentre eles se destacam os serviços de *delivery*. Esses estabelecimentos oferecem o serviço de entrega com um diferencial para as pessoas que não querem se locomover e ter seus pedidos entregues em casa. Pensando neste ponto, foi percebido a necessidade de desenvolvimento de um site (sistema) de vendas para *delivery*, por ser uma área com investimento crescente e possui um mercado extremamente potencial.

O sistema que será desenvolvido tem por finalidade principal realizar o monitoramento dos pedidos e das entregas solicitadas. Hoje não há um sistema computacional específico que realiza este tipo de controle de forma ágil para gerenciar os pedidos possibilitando sua montagem. Normalmente nos estabelecimentos, esse controle ocorre mediante uma solicitação ao atendente, com o sistema esse controle vai ser via *Web*.

Hoje muitos são os casos de reclamações com problemas nas entregas e nas reservas solicitadas, ocorridos devido à dificuldade de organizar a grande demanda de pedidos. Dessa forma, ocorre o acúmulo de trabalho, perdendo-se assim o controle sobre o tempo na entrega dos produtos.

1.2 JUSTIFICATIVA

O sistema proposto disponibilizará a realização de pedidos, possuindo um ambiente totalmente on-line e de fácil utilização, devido ao público alvo possuir pouco conhecimento técnico em informática. Com serviços diferenciais de solicitações e entregas, o sistema será uma forma de atrativo de um público ainda restrito.

O principal objetivo do sistema é aproximar os consumidores dos estabelecimentos e permitir que os mesmos conquistem novos clientes, além de servir os consumidores com um produto diferencial no mercado com serviços inovadores. Lembrando que o objetivo do produto não é substituir as ligações ou o contato dos clientes com o estabelecimento, mas disponibilizar novos e inovadores serviços aos atuais e novos clientes.

Com o amadurecimento na utilização da ferramenta os estabelecimentos reduzirão custos fixos de telefonemas e atendimento, devido a seus clientes antigos que optarem por usar o site para fazer os pedidos e também para acompanhar os mesmos. Porém se deve destacar que a ferramenta não visa substituir profissionais da área, mas sim, ser uma ferramenta de apoio para os funcionários, assim o índice de descontentamentos com as entregas ou serviços prestados pela empresa poderão ser reduzidos.

1.3 OBJETIVOS

1.3.1 Objetivo Geral

Como objetivo geral será desenvolver um sistema Web para pedidos de *delivery* com recursos de controle e monitoramento.

1.3.2 Objetivos Específicos

Como objetivos específicos do sistema:

- Estudo teórico e bibliográfico do assunto;
- Levantamento de requisitos para o desenvolvimento;
- Implementação do sistema.

1.4 METODOLOGIA

A metodologia utilizada para a realização deste trabalho foi estudo de caso aplicado à tecnologia. “É um tipo de metodologia de pesquisa qualitativa e/ou quantitativa, entendido como uma categoria de investigação que tem como objeto o estudo de uma unidade de forma aprofundada”. (PRODANOV; FREITAS, 2013, p. 60)

1.5 ESTRUTURA DO TRABALHO

Abaixo (quadro 1) é apresentado como está estruturado o trabalho:

Capítulo	Descrição
Introdução	Apresenta a introdução do trabalho.
Fundamentação teórica	Apresenta o sistema a ser desenvolvido, suas tecnologias utilizadas, sistemas similares e o detalhamento das etapas para sua construção.
Desenvolvimento	Apresenta na prática a aplicação das etapas para a construção do sistema.
Resultados	Apresentação do sistema através de imagens e seus detalhamentos.
Conclusão	Apresenta a conclusão do trabalho, problemas encontrados e trabalhos futuros.

Quadro 1 – Estrutura do trabalho
Fonte: Autoria própria

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta a fundamentação teórica do trabalho, apresentando as tecnologias utilizadas, sistemas similares e metodologia de desenvolvimento utilizada.

2.1 TECNOLOGIAS

2.1.1 EJB

Enterprise JavaBeans ou mais conhecida como EJB é uma arquitetura de componentes que permite o desenvolvimento rápido e simplificados de aplicações *Java EE (Java Enterprise Edition)*. Essa arquitetura possui várias especificações e será apenas discutido sobre as quais foram utilizadas no desenvolvimento do sistema proposto.

Uma aplicação EJB necessita de um EJB Container para poder gerenciar suas execuções. Dentre os servidores de aplicações *Java EE* que possuem um EJB *Container* estão o *JBoss*, *Glassfish* e *TomEE*.

O MVC é um padrão de desenvolvimento de *software* que separa em camadas a arquitetura de uma aplicação, onde o modelo (*model*) contém os dados, o estado e a lógica, a visualização (*view*) fornece uma apresentação do modelo e o controlador (*controller*) lida com as ações (FREEMAN, 2007). Em uma aplicação que utiliza o MVC as regras de negócios são implementadas na camada de modelo e utilizando a arquitetura EJB as regras são implementadas em um componente específico chamado *Session Bean*. Esse componente caracteriza três tipos: *Stateless*, *Stateful* e *Singleton*. Por outro lado, as classes Java que armazenam os dados para ser persistido são chamados de *Entity Beans*.

Stateless Session Bean caracteriza-se por não armazenar estado conversacional. Quando acessado este tipo de EJB, caso exista uma instância criada que esteja disponível o EJB *Container* atende a chamada pela mesma, senão é criada uma nova instância a qual realizará a chamada. A definição desse *Session Bean* é feita através da anotação `@Stateless` que vai acima da classe *Java*.

Stateful Session Bean caracteriza-se por manter esta convencional. Nesse tipo de componente, é criada uma nova instância para cada cliente, depois de criada a instância toda vez o que o cliente faz uma chamada a mesma é atendida por esta instância. Quando uma instância criada para determinado cliente fica ociosa por um determinado tempo, a mesma é destruída, sendo que quando o cliente fizer uma nova chamada uma nova instância será criada. A definição desse *Session Bean* é feita através da anotação `@Stateful`.

Singleton Session Bean caracteriza-se pelo compartilhamento de dados da aplicação, este tipo de EJB está disponível na versão 3.1 ou superior. Quando uma chamada é feita a este *Session Bean* caso exista uma instância criada à mesma atende a chamada, senão é criada a instância. A definição desse *Session Bean* é feita através da anotação `@Singleton`.

Para os *Session Bean* pode ser usada a anotação `@Local` ou `@Remote` acima da *interface* ou classe *Java*. A anotação `@Local` define que o acesso ao *Session Bean* vai ser local, apenas por ser acessada pela aplicação internamente, já a anotação `@Remote` define que o acesso pode ser remoto, aplicações que estejam fora do mesmo servidor de aplicação podem acessar o *Session Bean*. Na versão 3.1 ou superior do EJB, quando o acesso a um *Session Bean* é local, não é mais necessário definir uma interface *Java* nem utilizar a anotação `@Local` (K19, 2013).

2.1.2 JPA e Hibernate

O JDBC é uma grande API escrita em *Java*, até os dias de hoje se utiliza muito. Com o passar do tempo foi visto o quanto é trabalhoso escrever comandos SQL e popular objeto para persistir ou buscar dados utilizando o JDBC. Com isso surgiu o JPA, segundo COELHO (2013), é uma ferramenta muito poderoso, que reduz o tempo de desenvolvimento e torna a implementação de banco de dados mais fácil, sem a necessidade de criação de queries de banco de dados e sem a necessidade de popular os objetos para persistir dados.

O JPA é um *framework* ORM que faz o mapeamento das classes *Java* para as tabelas do banco de dados. Para COELHO (2013) é nada mais que um conjunto de regras, normas e interface que definem um comportamento. Esse comportamento

necessita da implementação para que possa ser utilizado, um exemplo é o *Hibernate* utilizado na implementação do sistema proposto.

O *Hibernate* cuida do mapeamento de objeto relacional através de classes Java para tabelas de banco de dados que podem ser feitas através do XML ou através de anotações nas classes Java.

2.1.3 Java Server Faces e Primefaces

O *Java Server Faces* é um framework para desenvolvimento de interface de usuário para aplicações *Java Web*. Tem como característica simplificar o desenvolvimento devido a ser baseado em componentes. O JSF permite ao desenvolvedor criar os componentes visuais que serão exibidos no formato HTML, além de poder utilizar os componentes e eventos que o mesmo possui.

O JSF é baseado no padrão MVC, onde os componentes das páginas se comunicam com a classe Java através dos *Managed Beans*, os quais podem ser definidos através do arquivo *faces-config* ou pela anotação `@ManagedBean` na classe Java. Os principais escopos dos *Managed Beans* são:

- *RequestScoped*: Escopo padrão. A cada requisição um novo objeto do *Managed Bean* será criado;
- *ViewScoped*: Escopo da página. Enquanto o usuário estiver na mesma página, o *Managed Bean* é mantido.
- *SessionScoped*: Escopo de sessão. Enquanto a sessão com o servidor não expirar, o mesmo objeto do *Managed Bean* atenderá o mesmo cliente.

Primefaces é uma suíte de componente leve de código aberto para JSF, com mais de 100 ricos conjuntos de componentes.

2.1.4 PrettyFaces

PrettyFaces é uma biblioteca Java de código aberto para reescrever URL com suporte ao JSF. Em um projeto JSF, devido ao padrão MVC, as páginas são

estruturadas por pastas tornando muitas vezes o caminho longo e não bonito das URLs, com o PrettyFaces é possível à criação de URLs bonitas.

Exemplo aplicação JSF comum:

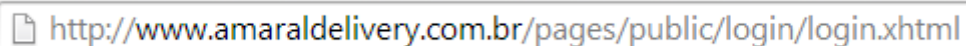


Figura 1 - URL comum
Fonte: Autoria própria

Exemplo aplicação JSF com PrettyFaces:

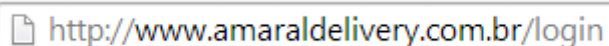


Figura 2 - URL pelo PrettyFaces
Fonte: Autoria própria

2.2 SISTEMA PROPOSTO

As funcionalidades do sistema são mostradas na figura 3, onde são apresentadas por usuário. Essas funcionalidades são detalhadas ao decorrer do trabalho.

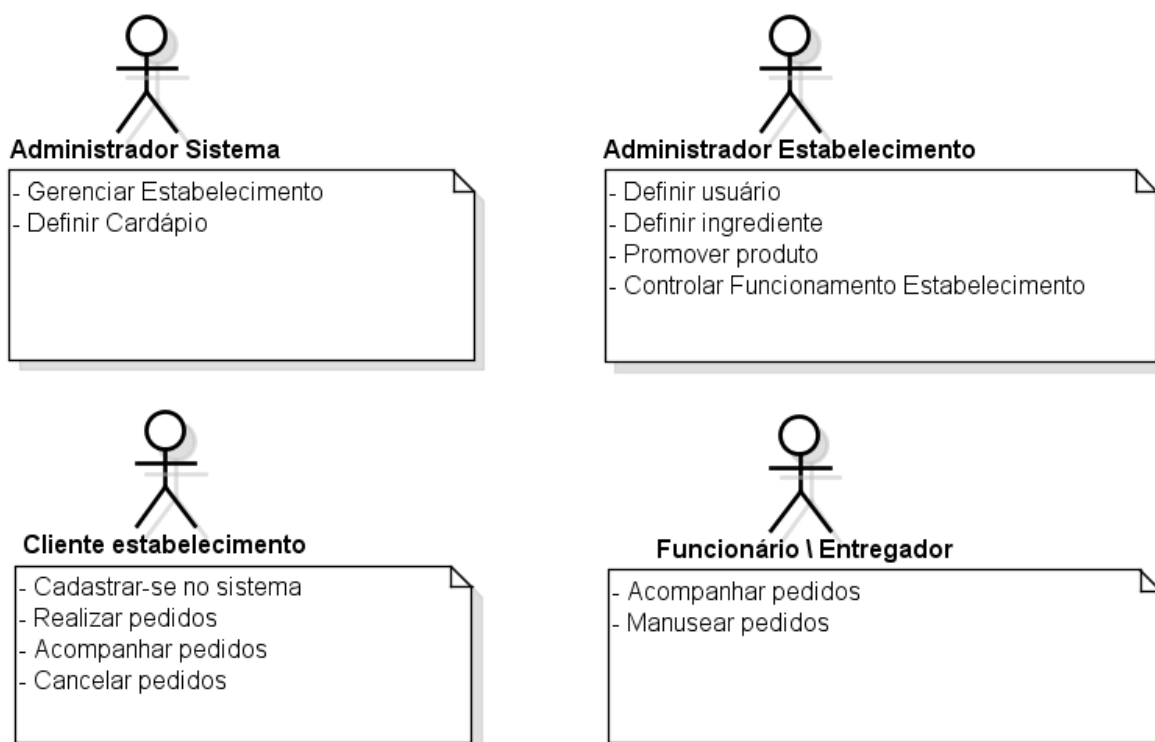


Figura 3 – Funcionalidades do sistema por usuário.
Fonte: Autoria própria

Conforme apresentado o sistema será totalmente *Web*, onde os estabelecimentos que tiverem interesse de usar do sistema disponibilizando seus produtos deverão entrar em contato pelo *e-mail* que será disponibilizado no site. O administrador do sistema será responsável por buscar as informações dos estabelecimentos para disponibilizar acesso aos mesmos.

Disponibilizado acesso o administrador do estabelecimento em questão vai poder realizar a manutenção de ingredientes, gerenciar usuários, controlar o funcionamento do estabelecimento e gerenciar os produtos. A manutenção de usuário vai permitir ao administrador incluir usuário administrador, funcionário e entregador. No gerenciamento de produtos o administrador vai poder realizar a manutenção de produtos informando os ingredientes cadastrados que compõem ou poderão compor o produto e referenciar o produto a um cardápio disponibilizado pelo administrador do sistema. Para controle do funcionamento do estabelecimento o administrador vai poder abrir ou fechar o estabelecimento quando bem entender.

Usuários que desejam realizar um pedido no site poderão navegar pelo cardápio selecionando estabelecimento(s) e produto(s). É possível finalizar pedido somente se estiver cadastrado no sistema. O cadastro de usuário no sistema poderá ser feito pelo formulário de acesso ou pelo carrinho de compra (itens do pedido), informando os dados necessários.

A visualização de pedidos por estabelecimento mostrará todos os pedidos realizados para o usuário funcionário, enquanto para o usuário entregador aparecerá apenas os pedidos que devem ou foram entregues. Já a visualização por usuário mostrará os pedidos que o usuário realizou.

No capítulo 3 e 4 deste trabalho é possível conhecer mais sobre o sistema, onde contém a análise do sistema e sua apresentação de resultados.

2.2.1 Restrições do sistema

- O sistema não contemplará emissão de notas ou cupons fiscais;
- Funcionalidades do sistema não serão projetadas para dispositivos *mobile*.

2.2.2 Premissas

- O estabelecimento deve possuir um *link* de *internet*;
- O estabelecimento deve possuir um cadastro no sistema.

2.2.3 Diferencial de mercado

O diferencial do sistema é a listagem de produtos por cardápio e estabelecimento, onde os usuários buscam diretamente suas necessidades, por exemplo, uma pessoa que está querendo uma *pizza*, ela acessa o cardápio “*Pizza*”, o sistema busca todos os estabelecimentos que possuam o cardápio em seus produtos, após escolher o estabelecimento são listadas as pizzas cadastradas pelo estabelecimento. A ideia é atender diretamente na necessidade e não apresentar os produtos de todos os estabelecimentos, evitando descontentamento de estabelecimento por mostrar produtos de concorrentes lado a lado.

2.3 SISTEMAS SIMILARES

2.3.1 iFood

É um dos maiores *sites* de pedido *online* com foco em restaurantes, *pizzarias* e lanchonetes. O site é um caso de sucesso, com cada vez mais clientes novos e de outros estados vem utilizando o site. Pelo iFood é possível verificar os restaurantes abertos e fechados podendo navegar pelo cardápio de cada um. O cardápio é montado pelo próprio restaurante onde é possível filtrar pelo mesmo.



Restaurantes abertos (23)



Figura 4 - Site iFood
Fonte: IFOOD, 2014

2.3.2 Papa Johns

É uma franquia de restaurantes mais conhecido por suas pizzas, atualmente a rede está presente em mais de 30 países. A franquia possui um site para pedido online onde é possível informar o endereço trazendo a franquia atendida pelo mesmo. Depois de buscado a franquia é listado o cardápio onde é possível selecionar o mesmo.

Inicio · Nosotros · Servicio al Cliente · Ventas Corporativas y Eventos · Contáctenos

Menú Promociones Restaurantes

Pide delivery online aquí

Ingresar / Registrarse

Donde te encuentras en Lima? Ve sólo los restaurantes que entregan a tu casa.

Tu ciudad: Lima

Tu distrito: Tu distrito

Tu dirección: e.j. Av. Conquistadores 591

Ver restaurantes

Escoge entre 17 restaurantes en Lima

Filtrar resultados

Ordenar por: Relevancia | Calificación | Pedido mínimo | Coste de envío

Buscar

Mostrar restaurantes

- Pago en Línea disponible
- Ofertas disponibles
- Retiro en el local
- Acepta cupón descuento
- Delivery
- Delivery gratis
- El delivery más rápido
- Promos Mundialistas
- Saludable

Tipo de Comida

- Pizza (17)

Papa Johns (Bellavista)

Lima, Bellavista, Mall Aventura Plaza - Av Colonial 3800

4 Reviews

Pizza

Abierto: 12:00 - 23:00

Envío gratis

Pedido mínimo S/. 19,90

Ver menú >

Papa Johns (Benavides 20)

Lima, Surquillo, Av. Benavides 2019, Urb. La Aurora, Miraflores

50 Reviews

Pizza

Abierto: 12:00 - 23:00

Envío gratis

Pedido mínimo S/. 19,90

Ver menú >

Figura 5 - Site Papa Johns
Fonte: PAPAJOHNS, 2014

2.4 METODOLOGIA DE DESENVOLVIMENTO

A metodologia utilizada para a especificação do sistema proposto é uma metodologia orientada a objetos. Segundo DEITEL (2010) para onde quer que se olhe no mundo real se vê objetos como exemplo pessoas, animais, carros, computadores e assim por diante, assim o que define um objeto é que ele tem atributos (tamanho, forma, cor e peso) e todos exibem comportamentos (por exemplo, o carro anda, freia e desvia).

UML é o esquema de representação gráfica mais utilizada para modelar sistemas orientados a objetos, como seu conceito já diz Linguagem de Modelagem Unificada. A UML não é uma linguagem de programação, mas sim, uma notação utilizada por métodos para demonstrar seus projetos. A metodologia UML foi

utilizada para que o projeto seja documentado, analisado e também visualizado posteriormente de forma eficiente.

Baseando-se nas metodologias anteriormente citadas, abaixo é apresentando os modelos para os seguintes artefatos: requisitos de software, os casos de uso (junto de seu diagrama), o diagrama de classes, digrama ER e logo após a prototipação de telas.

2.4.1 Requisitos de Software

Os requisitos foram definidos de forma simples e exemplificados, não seguido nenhum padrão já definido ou documentado, apenas exemplos utilizados em projetos finais. Abaixo segue o modelo utilizado para os requisitos funcionais (quadro 1):

Fx – Nome do Requisito	
Descrição:	
Requisitos Não Funcionais Associados	
NFx.x	

Quadro 2 - Modelo Requisito Funcional
Fonte: Autoria própria

Os requisitos funcionais devem contemplar com um título demonstrando a funcionalidade, uma descrição e abaixo os requisitos não funcionais associados ao requisito não funcional. Requisito funcional define uma função de um *software* ou um componente em específico, já o requisito não funcional definem as propriedades e restrições do sistema relacionado ao requisito.

Para os requisitos gerais também será seguido nenhum padrão definido, mas com diferença dos requisitos funcionais, o modelo terá o nome do requisito geral e um detalhamento do mesmo. Abaixo segue o modelo (quadro 2):

Nome	Restrição
Gx	

Quadro 3 - Modelo Requisitos Gerais
Fonte: Autoria própria

2.4.2 Casos de Uso

Os casos de uso são utilizados para expressar e documentar os comportamentos pretendidos para as funções do sistema, eles referem-se aos serviços ou funções que podem ser utilizados por usuários do sistema, como cadastrar um cliente ou emitir um relatório. Em geral pode-se associar um caso de uso a uma funcionalidade do sistema (GUEDES, 2009).

O diagrama de caso de uso procura por meio de uma simples linguagem, possibilitar a compreensão o comportamento do sistema por qualquer pessoa, através de uma perspectiva do usuário. As setas representam as interações ou relacionamentos entre atores que fazem parte do diagrama, entre os atores e os casos de usos ou os relacionamentos entre um caso de uso e outro. Os relacionamentos entre os casos de usos podem ser inclusão (*include*) que indica obrigatoriedade, ou seja, que o caso também será executado ou extensão (*extends*) utilizada para descrever os cenários opcionais. Abaixo segue exemplo de um caso de uso (figura 6):

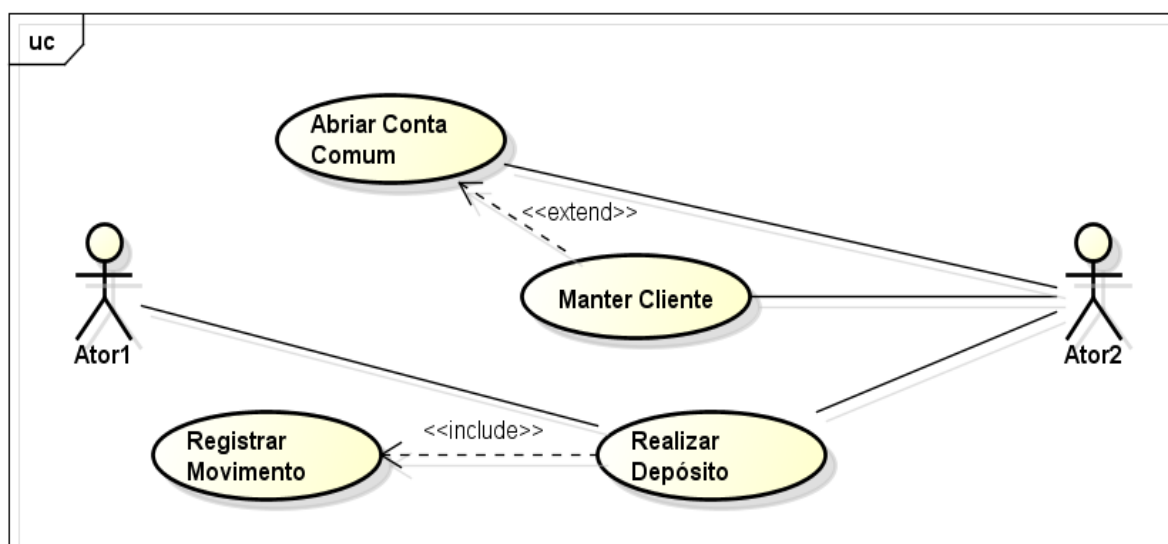


Figura 6 - Modelo Diagrama Caso de uso
Fonte: Autoria própria

O modelo diagrama caso de uso (figura 6) mostrar um simples exemplo de conta bancária onde o Ator1 pode realizar depósitos e o Ator2 pode abrir conta, manter cliente ou realizar depósitos. No caso de uso manter cliente depois de realizado o caso de uso o ator pode abrir conta comum do mesmo não sendo necessária essa operação. Já no caso de uso realizar depósitos após executado o

caso de uso será necessário executar o caso de uso registrar movimento. Para o caso de uso realizar depósito podem ser feitos tanto pelo Ator1 quanto o Ator2.

A documentação de um caso de uso costuma descrever por meio de uma simples linguagem a descrição e uso em linhas gerais do caso de uso, que atores interagem, quais as etapas a serem executadas, parâmetros a serem fornecidos e o tratamento de exceções. O formato de documentação foi usado de exemplo por projetos finais de acordo com a característica do próprio diagrama, onde o formato de um caso de uso é bastante flexível. Abaixo segue o modelo utilizado (quadro 3):

Caso de Uso: UCx – Nome do Caso de Uso	
Descrição	
Ator(es)	
Pré-condição	
Pós-condição	
Requisitos correlacionados	
Fluxo Principal	
Fluxo Alternativo	
Tratamento de Exceções	

Quadro 4 - Modelo Caso de Uso
Fonte: Autoria própria

2.4.3 Diagrama de Classes

O diagrama de classes representa a estrutura do sistema, recorrendo ao conceito de classe e suas relações. Seu principal enfoque está em permitir a visualização das classes a ser utilizada pelo sistema com seus respectivos atributos e métodos. O diagrama representa uma visão estática de como as classes estão organizadas, procurando em como definir sua estrutura lógica (GUEDES, 2009).

Basicamente, o diagrama de classe é composto por suas classes e pelo relacionamento entre classes caso exista. Os símbolos de menos (-) e mais (+) na frente dos atributos representam a visibilidade dos mesmos onde menos pode ser

vista por outras classes e mais não pode ser visto por outras classes. Abaixo segue exemplo de diagrama de classe onde é mostrado o nome da classe, os atributos da classe e os métodos da classe:

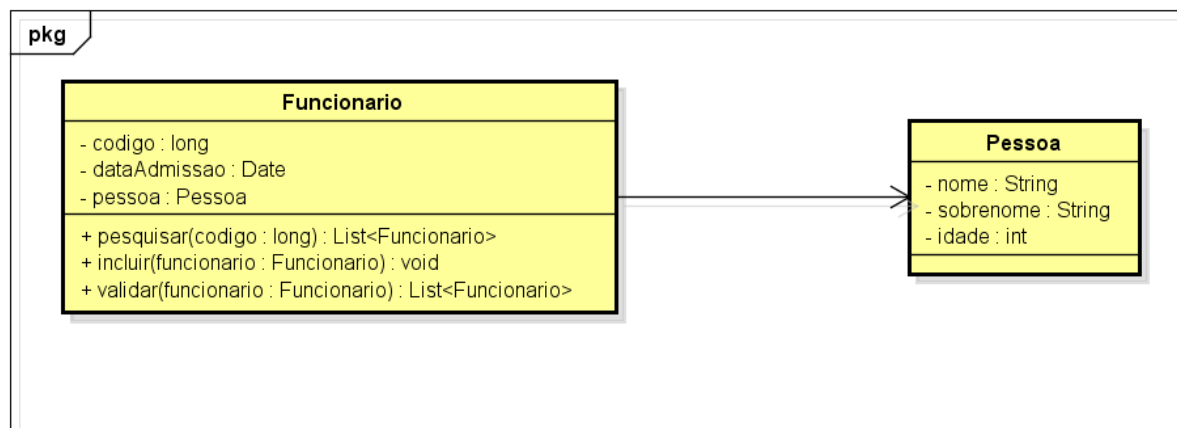


Figura 7 - Exemplo Diagrama de Classe
Fonte: Autoria própria

O exemplo de diagrama de classe (figura 7) mostrar a classe Funcionario e Pessoa. Na classe Funcionario é mostrado o nome da classe, depois os atributos privados codigo que é do tipo long, dataAdmissao do tipo Date e pessoa que é do tipo Pessoa (outra classe), logo abaixo dos atributos são mostrados os métodos públicos pesquisar que retorna uma lista da própria classe, incluir que não possui retorno e o método privado validar que retorna um boolean. O método pesquisar espera um long como parâmetro já os métodos incluir e validar espera um Funcionario como parâmetro.

2.4.4 Diagrama ER

Com o diagrama ER (Entidade Relacionamento) é possível representar de forma objetiva e bem estruturada, todos os elementos abstraídos de análise de sistemas. O digrama de ER é composto por entidades e o relacionamento entre elas caso exista. Cada entidade possui seus atributos os quais podem ser únicos representados pela sigla UK, chave primária representada pela sigla PK ou pode ser chave estrangeira representada pela sigla FK. Atribuindo cada entidade definida atributos pertinentes ao sistema, desta forma, pode-se definir conceitualmente que representação os elementos armazenará dados, que por sua vez, são representados pelos atributos. Abaixo segue exemplo de um diagrama ER (figura 8):

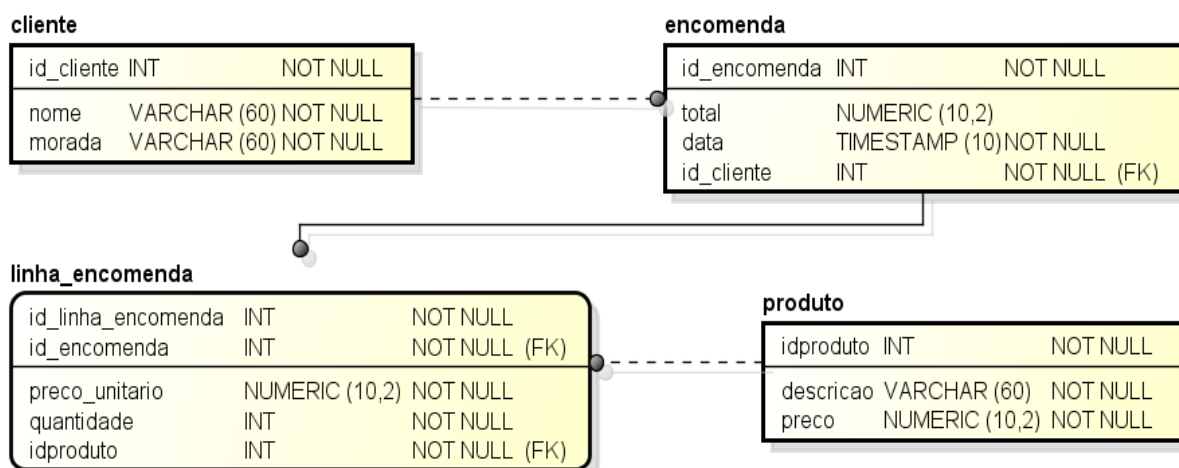


Figura 8 - Exemplo Diagrama ER
Fonte: Autoria própria

O exemplo diagrama ER (figura 8) mostra as entidades de relacionamento necessárias para uma encomenda, onde é mostrado o nome da entidade ou tabela, a chave primária da tabela, logo abaixo da chave primária os campos da tabela com seu tipo. Também para as tabelas que possuam chave estrangeira é mostrado o nome da chave estrangeira e o seu relacionamento com outras tabelas.

3 DESENVOLVIMENTO

Este capítulo tem como objetivo aplicar a metodologia de desenvolvimento apresentada. Dentre o UML são apresentados os requisitos de *software* que vão compor a aplicação, os casos de usos com suas descrições, o diagrama de classes de análise e o digrama entidade relacionamento.

3.1 REQUISITOS DE SOFTWARE

A seguir, são apresentados os requisitos funcionais do sistema:

- F1 – Autenticar Usuário
- F2 – Gerenciar Estabelecimento
- F3 – Definir Usuário
- F4 – Definir Cardápio
- F5 – Definir Ingrediente
- F6 – Promover Produto
- F7 – Controlar Funcionamento Estabelecimento
- F8 – Realizar Pedido
- F9 – Acompanhar Pedido
- F10 – Visualizar Pedido
- F11 – Alterar Pedido

3.1.1 Requisitos Funcionais

F1 – Autenticar Usuário

Descrição: Os usuários podem realizar autenticação através de um usuário e senha previamente cadastrados no sistema. A autenticação só deve ser realizada se o usuário e a senha estiverem cadastrados no banco de dados.

Requisitos Não funcionais Associados

NF1.1	Autenticação deve ser realizada via banco de dados.
NF1.2	A senha deve ser armazenada criptografada em banco de dados.
NF1.3	Os dados para autenticação devem contemplar as seguintes

	<p>informações:</p> <ul style="list-style-type: none"> • E-mail (obrigatório); • Senha (obrigatório).
--	---

Quadro 5 - Requisito Funcional autenticar usuário
Fonte: Autoria própria

F2 – Gerenciar Estabelecimento

Descrição: O sistema deve permitir o gerenciamento de estabelecimento permitindo a inclusão ou alteração. Os estabelecimentos são clientes do sistema.

Requisitos Não Funcionais Associados

NF2.1	<p>Os filtros de pesquisa devem contemplar as seguintes informações:</p> <ul style="list-style-type: none"> • E-mail; • CNPJ.
NF2.2	<p>Os dados do estabelecimento devem contemplar as seguintes informações:</p> <ul style="list-style-type: none"> • Nome (obrigatório); • CNPJ; • Fone (obrigatório); • E-mail (obrigatório); • Taxa de entrega; • Atendimento; • CEP (obrigatório); • UF (obrigatório); • Endereço (obrigatório); • Nº (obrigatório); • Bairro (obrigatório); • Cidade (obrigatório); • Bloqueado; • Alterar senha (Apenas na edição).
NF2.3	<p>Os dados da listagem da pesquisa devem contemplar as seguintes informações:</p> <ul style="list-style-type: none"> • Nome; • Fone;

	<ul style="list-style-type: none"> • Bloqueado.
NF2.4	Um estabelecimento pode ser bloqueado, não listando o mesmo nos cardápios do catálogo <i>Web</i> .
NF2.5	As informações do estabelecimento só podem ser gerenciadas pelo usuário máster do sistema.

Quadro 6 - Requisito Funcional Gerenciar Estabelecimento
Fonte: Autoria própria

F3 – Definir Usuário

Descrição: O sistema deve permitir incluir, alterar e bloquear os usuários que poderão acessar o sistema e realizar ações no mesmo conforme as permissões do seu perfil.

Requisitos Não Funcionais Associados

NF3.1	Os filtros de pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome.
NF3.2	Os dados do usuário devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome (obrigatório); • E-mail; • Função; • Bloqueado.
NF3.3	Os dados da listagem da pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome; • Função; • Bloqueado.
NF3.4	Ao incluir um estabelecimento pelo usuário máster, automaticamente o sistema inclui um usuário administrador através das informações nome e e-mail do estabelecimento.
NF3.5	Para usuário incluído ou marcado a opção “Alterar senha” pelo administrador, ao gravar o mesmo o sistema envia e-mail para alterar sua senha através do e-mail cadastrado.
NF3.6	O sistema deve permitir a inclusão de usuário para realizar pedido com as informações:

	<ul style="list-style-type: none"> • E-mail (obrigatório); • Senha (obrigatório); • Nome (obrigatório); • CPF (obrigatório); • Data Nascimento (obrigatório); • Sexo (obrigatório); • Fone (obrigatório); • Celular; • CEP (obrigatório); • Endereço (obrigatório); • Número (obrigatório); • Complemento; • Bairro (obrigatório); • Cidade (obrigatório); • Estado (obrigatório); • Referência.
NF3.7	O sistema deve permitir que todos os usuários possam recuperar sua senha informando o e-mail em seu cadastro para que um e-mail possa ser enviado.
NF3.8	O usuário bloqueado ou o estabelecimento relacionado ao usuário estando bloqueado, não será possível a autenticação do mesmo no sistema.
NF3.9	O sistema deve permitir que todos os usuários possam alterar sua senha.
NF3.10	O sistema deve permitir ao usuário cliente poder alterar seus dados (menos seu e-mail).

Quadro 7 - Requisito Funcional Definir Usuário
Fonte: Autoria própria

F4 – Definir Cardápio

Descrição: O sistema deve permitir incluir, alterar e bloquear os cardápios.

Requisitos Não Funcionais Associados

NF4.1	Os filtros de pesquisa devem contemplar as seguintes informações:
-------	---

	<ul style="list-style-type: none"> • Nome.
NF4.2	Os dados da listagem da pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome; • Bloqueado.
NF4.3	Os dados do cardápio devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome (obrigatório); • Bloqueado.
NF4.4	O cardápio pode ser bloqueado. Não estando bloqueado o cardápio será apresentado no catálogo Web.

Quadro 8 - Requisito Funcional Definir Cardápio
Fonte: Autoria própria

F5 – Definir Ingrediente

Descrição: O sistema deve permitir incluir, alterar e bloquear os ingredientes que serão utilizados nos produtos.

Requisitos Não Funcionais Associados

NF5.1	Os filtros de pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome.
NF5.2	Os dados da listagem da pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome; • Bloqueado.
NF5.3	Os dados do ingrediente devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome (obrigatório); • Bloqueado;
NF5.4	O Ingrediente pode ser bloqueado. Não estando bloqueado o ingrediente será apresentado na manutenção de produtos.

Quadro 9 - Requisito Funcional Definir Ingrediente
Fonte: Autoria própria

F6 – Promover Produto

Descrição: O sistema deve permitir incluir, alterar e bloquear os produtos que serão utilizados para realização de pedidos.

Requisitos Não Funcionais Associados	
NF6.1	Os filtros de pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome.
NF6.2	Os dados da listagem da pesquisa devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome; • Cardápio; • Valor; • Bloqueado.
NF6.3	Os dados do produto devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome (obrigatório); • Cardápio (Nome) (obrigatório); • Valor (obrigatório); • Bloqueado; • Imagem; • Opções (Opção, Acréscimo, Bloqueado); • Ingredientes (Ingrediente, Retirar); • Acrescentar (Ingrediente, Valor);
NF6.4	O produto pode ser bloqueado. Não estando bloqueado o produto será apresentado na seleção do cardápio referente ao estabelecimento.
NF6.5	O dado opção do produto pode ser bloqueado e alterado o nome e o valor de acréscimo.
NF6.6	O dado ingrediente do produto pode ser excluído.

Quadro 10 - Requisito Funcional Promover Produto

Fonte: Autoria própria

F7 – Controlar Funcionamento Estabelecimento

Descrição: O sistema deve permitir ao estabelecimento controlar seu funcionamento.

Requisitos Não Funcionais Associados

NF7.1	Deve ser apresentada a opção ao administrador do estabelecimento de acordo com seu funcionamento. O estabelecimento estando
-------	---

	fechado deve apresentar a opção para abrir o estabelecimento, senão deve ser apresentada a opção para fechar o estabelecimento.
--	---

Quadro 11 - Requisito Funcional Controlar Funcionamento Estabelecimento
Fonte: Autoria própria

F8 – Realizar Pedido	
Descrição: O sistema deve permitir ao usuário cliente realizar pedidos.	
Requisitos Não Funcionais Associados	
NF8.1	Os dados dos estabelecimentos na listagem pelo cardápio devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Imagem; • Nome; • Fone; • Atendimento; • Aberto ou fechado.
NF8.2	Os dados dos produtos na listagem pelo cardápio e estabelecimento devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Imagem; • Nome; • Opções com seus valores (se possuir).
NF8.3	Os dados do produto na visualização devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Nome; • Imagem; • Quantidade; • Opções (se possuir); • Ingredientes (se possuir); • Acrescentar (se possuir).
NF8.4	Selecionado um cardápio o mesmo é apresentado, senão é apresentada ao ator a informação selecione um cardápio.
NF8.5	Selecionado um estabelecimento são mostrados os dados do mesmo logo abaixo do cardápio selecionado contemplando as seguintes informações:

	<ul style="list-style-type: none"> • Nome; • Imagem; • Endereço (rua, número rua, bairro, CEP, cidade, UF); • Fone; • Atendimento.
NF8.6	<p>Nos passos “Pedido” e “Finalizar pedido” os dados do estabelecimento devem contemplar com as seguintes informações:</p> <ul style="list-style-type: none"> • Nome; • Imagem; • Endereço (rua, número rua, bairro, CEP, cidade, UF); • Fone; <p>Logo abaixo, devem ser listados os dados dos produtos com as seguintes informações:</p> <ul style="list-style-type: none"> • Produto (nome do produto com a opção selecionada caso possua, ingredientes a retirar no produto caso possua, ingredientes acrescentar no produto caso possua); • Quantidade; • Valor unidade; • Total.
NF8.7	<p>Os dados do endereço no passo “Finalizar pedido” devem contemplar as seguintes informações:</p> <ul style="list-style-type: none"> • CEP; • Endereço; • Número; • Complemento; • Bairro; • Cidade; • Estado; • Referência.

Quadro 12 - Requisito Funcional Realizar Pedido
Fonte: Autoria própria

Descrição: O sistema deve permitir os usuários que possuem acesso, acompanhar os pedidos realizados.	
Requisitos Não Funcionais Associados	
NF9.1	<p>As opções de busca dos pedidos para o estabelecimento para o usuário funcionário deve contemplar com as seguintes informações:</p> <ul style="list-style-type: none"> • Pedido do dia; • Pedidos em aberto; • Pedidos em preparo; • Pedidos aguardando; • Pedidos a entregar; • Pedidos entregues; • Pedidos pendurados; • Pedidos cancelados; • Pedidos por número; • Pedidos por data; • Pedidos por cliente; • Todos os pedidos.
NF9.2	<p>As opções de busca dos pedidos para o estabelecimento para o usuário entregador deve contemplar com as seguintes informações:</p> <ul style="list-style-type: none"> • Pedidos aguardando; • Pedidos a entregar; • Pedidos entregues.
9.3 NF	<p>As opções de busca dos pedidos para o usuário cliente deve contemplar com as seguintes informações:</p> <ul style="list-style-type: none"> • Últimos pedidos; • Pedidos em aberto; • Pedidos entregues; • Pedidos por número; • Pedidos por data; • Todos os pedidos.
NF9.4	A opção “Pedidos aguardando” para o usuário entregador deve

	apresentar os pedidos já preparados que precisam ser entregues.
NF9.5	A opção “Pedidos cancelados” deve apresentar os pedidos cancelados na data atual do sistema.
NF9.6	A opção “Pedidos por número” deve apresentar o filtro de pesquisa “Pedido” (obrigatório).
NF9.7	A opção “Pedidos por data” deve apresentar os filtros de pesquisa data: <ul style="list-style-type: none"> • Inicial (obrigatório); • Final (obrigatório).
NF9.8	A opção “Pedidos por cliente” deve apresentar os filtros de pesquisa: <ul style="list-style-type: none"> • E-mail; • CPF.
NF9.9	A opção “Últimos pedidos” deve apresentar os 10 últimos pedidos realizados pelo usuário cliente.
NF9.10	Os dados da listagem da opção de busca dos pedidos devem contemplar as seguintes informações: <ul style="list-style-type: none"> • Pedido; • Total; • Data; • Status; • Entregar.

Quadro 13 - Requisito Funcional Acompanhar Pedido
Fonte: Autoria própria

F10 – Visualizar Pedido

Descrição: O sistema deve permitir aos usuários que possuem acesso, visualizar os pedidos.

Requisitos Não Funcionais Associados

NF10.1	Os dados de visualização dos itens do pedido devem contemplar com as seguintes informações: <ul style="list-style-type: none"> • Produto (nome do produto com a opção selecionada caso possua, ingredientes a retirar no produto caso possua, ingredientes acrescentar no produto caso possua);
--------	--

	<ul style="list-style-type: none"> • Quantidade; • Valor; • Total.
--	---

Quadro 14 - Requisito Funcional Visualizar Pedido
Fonte: Autoria própria

F11 – Alterar Pedido

Descrição: O sistema deve permitir a alteração do status do pedido.

Requisitos Não Funcionais Associados

NF 1.1	<p>A hierarquia de avanço do pedido é apresentada da seguinte forma:</p> <ol style="list-style-type: none"> 1. Aberto; 2. Em preparo; 3. Aguardando; 4. Saiu para entrega (se necessário entregar); 5. Entregue.
--------	---

Quadro 15 - Requisito Funcional Alterar Pedido
Fonte: Autoria própria

3.1.2 Requisitos Gerais

Nome	Restrição
G1	O sistema deve funcionar em ambiente <i>Web</i> .
G2	O sistema deve ser compatível com o <i>Java 7</i> .
G3	O sistema deve rodar em um banco de dados <i>PostgreSQL 9.1</i> ou Superior.
G4	O sistema deve rodar em um servidor de aplicação <i>JBoss Application Server 7</i> .
G5	A <i>interface</i> do sistema deve ser desenvolvida em xHTML validado conforme especificações da W3C.
G6	O <i>design</i> da <i>interface</i> deve ser desenvolvido com o uso de CSS (Folhas de Estilo em Cascata) em um arquivo separado do xHTML.
G7	O sistema deve ser compatível com o <i>Internet Explorer 8</i> ou superior, <i>Mozilla Firefox 4.0</i> ou Superior e o <i>Google Chrome</i> .

Quadro 16 - Requisitos Gerais do Sistema
Fonte: Autoria própria

3.2 CASOS DE USO

A figura 9 apresenta o caso de uso do sistema relacionado aos requisitos funcionais levantados.

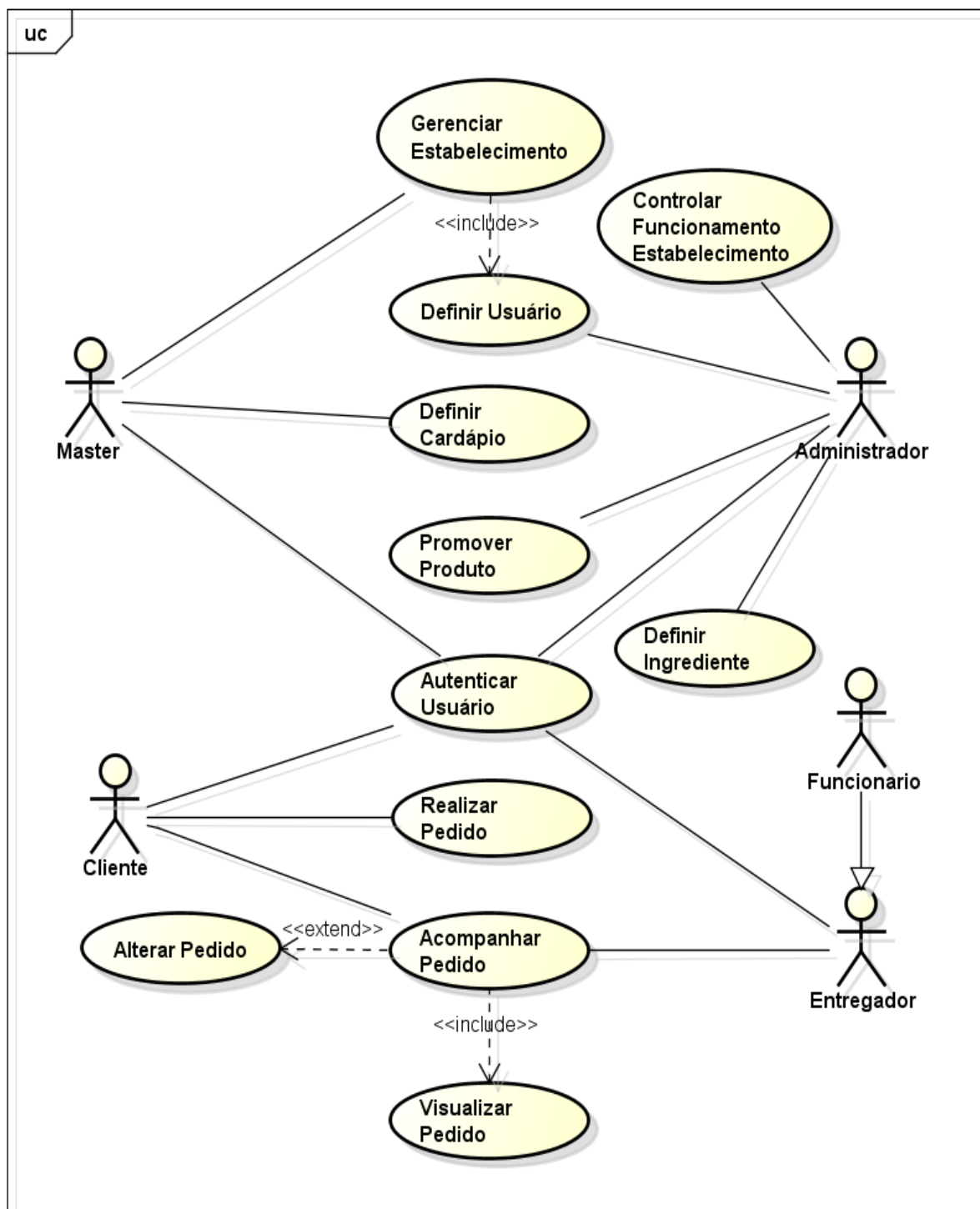


Figura 9 – Digrama de caso de uso do sistema

Fonte: Autoria própria

Caso de Uso: UC1 – Autenticar Usuário	
Descrição	Este caso de uso tem por objetivo permitir a autenticação dos usuários no sistema.
Ator(es)	Master, Administrador, Cliente, Entregar, Funcionário.
Pré-condição	Ter usuário incluso em banco de dados do sistema.
Pós-condição	Usuário autenticado com sucesso.
Requisitos correlacionados	F1, NF1.1, NF1.2, NF1.3
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Solicita ao sistema a funcionalidade desejada; 2. [Sistema] Apresenta um formulário com os dados de autenticação; 3. [Ator] Informa e submete seus dados ao sistema; 4. [Sistema] Realiza a autenticação do usuário. [E1.1]; 	
Tratamento de exceções	
[E1.1 - Usuário ou senha não encontrado]	
<ol style="list-style-type: none"> 1. [Sistema] O sistema apresenta uma mensagem informando ao ator que o usuário ou senha estão inválidos. <p>Retorna ao passo 2 do fluxo principal.</p>	

Quadro 17 - Caso de uso UC1 - Autenticar Usuário

Fonte: Autoria própria

Caso de Uso: UC2 – Fluxo Básico	
Descrição	Este caso de uso tem por objetivo permitir a manutenção de um determinado registro.
Ator(es)	Varia de acordo com a manutenção do registro.
Pré-condição	Varia de acordo com a manutenção do registro.
Pós-condição	Varia de acordo com a manutenção do registro.
Requisitos correlacionados	Varia de acordo com a manutenção do registro.
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Solicita ao sistema a funcionalidade desejada; 2. [Sistema] Apresenta um formulário de pesquisa e a opção “Inserir”; 3. [Ator] Informa os filtros de pesquisa desejados e aciona a opção “Pesquisar”; 	

<p>[A2.1];</p> <p>4. [Sistema] Apresenta os registros encontrados em uma listagem a partir dos filtros informados. [E2.1]; [A2.2];</p>
Fluxo Alternativo
<p>[A2.1 – Incluir novo registro]</p> <ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Inserir”; 2. [Sistema] Apresenta um formulário em uma interface modal para a inclusão de um novo registro; 3. [Ator] Informa os dados necessários e solicita a inclusão do registro; 4. [Sistema] Realiza a inclusão do registro em banco de dados; [E2.2]; <p>Retorna ao passo 4 do fluxo principal.</p> <p>[A2.2 – Alterar registro existente]</p> <ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Editar” a partir dos resultados da pesquisa; 2. [Sistema] Apresenta um formulário em uma interface modal para a edição dos dados do registro selecionado; 3. [Ator] Edita os dados desejados e solicita a alteração do registro; 4. [Sistema] Realiza a edição do registro em banco de dados. [E2.2]; <p>Retorna ao passo 4 do fluxo principal.</p>
Tratamento de exceções
<p>[E2.1 – Nenhum registro encontrado]</p> <ol style="list-style-type: none"> 1. [Sistema] Apresenta na listagem de pesquisa a informação que nenhum registro foi encontrado. <p>Fim do caso de uso.</p> <p>[E2.2 – Dados obrigatórios ou inválidos]</p> <ol style="list-style-type: none"> 1. [Sistema] Destaca os dados obrigatórios não informados e os dados informados inválidos. <p>Retorna ao passo 2 do fluxo alternativo [A2.1] [A2.2].</p>

Quadro 18 - Caso de uso UC2 - Fluxo Básico
 Fonte: Autoria própria

Caso de Uso: UC3 – Gerenciar Estabelecimento

Descrição	Este caso de uso tem por objetivo permitir o gerenciamento de estabelecimentos do sistema.
Ator(es)	Master.

Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Estabelecimento disponível ou bloqueado no sistema.
Requisitos correlacionados	F2, NF2.1, NF2.2, NF2.3, NF2.4, NF2.5
Fluxo Principal	
Respeitar o fluxo principal do caso de uso [UC2] .	
Fluxo Alternativo	
Respeitar o fluxo alternativo do caso de uso [UC2] .	
Tratamento de exceções	
Respeitar o tratamento de exceções do caso de uso [UC2] .	

Quadro 19 - Caso de uso UC3 - Gerenciar Estabelecimento
Fonte: Autoria própria

Caso de Uso: UC4 – Definir Usuário	
Descrição	Este caso de uso tem por objetivo permitir a manutenção de usuários por estabelecimento.
Ator(es)	Administrador.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Usuário disponível ou bloqueado no sistema.
Requisitos correlacionados	F3, NF3.1, NF3.2, NF3.3, NF3.4, NF3.5, NF3.6, NF3.7, NF3.8
Fluxo Principal	
Respeitar o fluxo principal do caso de uso [UC2] .	
Fluxo Alternativo	
Respeitar o fluxo alternativo do caso de uso [UC2] .	
Tratamento de exceções	
Respeitar o tratamento de exceções do caso de uso [UC2] .	

Quadro 20 - Caso de uso UC4 - Definir Usuário
Fonte: Autoria própria

Caso de Uso: UC5 – Definir Cardápio	
Descrição	Este caso de uso tem por objetivo permitir a manutenção de cardápios.
Ator(es)	Master.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.

Pós-condição	Cardápio disponível ou bloqueado no sistema.
Requisitos correlacionados	F4, NF4.1, NF4.2, NF4.3, NF4.4
Fluxo Principal	
Respeitar o fluxo principal do caso de uso [UC2] .	
Fluxo Alternativo	
Respeitar o fluxo alternativo do caso de uso [UC2] .	
Tratamento de exceções	
Respeitar o tratamento de exceções do caso de uso [UC2] .	

Quadro 21 - Caso de uso - Definir Cardápio

Fonte: Autoria própria

Caso de Uso: UC6 – Definir Ingrediente

Descrição	Este caso de uso tem por objetivo permitir a manutenção de ingredientes.
Ator(es)	Administrador.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Ingrediente disponível ou bloqueado no sistema.
Requisitos correlacionados	F5, NF5.1
Fluxo Principal	
Respeitar o fluxo principal do caso de uso [UC2] .	
Fluxo Alternativo	
Respeitar o fluxo alternativo do caso de uso [UC2] .	
Tratamento de exceções	
Respeitar o tratamento de exceções do caso de uso [UC2] .	

Quadro 22 - Caso de uso UC6 - Definir Ingrediente

Fonte: Autoria própria

Caso de Uso: UC7 – Promover Produto

Descrição	Este caso de uso tem por objetivo permitir a manutenção de produtos.
Ator(es)	Administrador.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Produto disponível ou bloqueado no sistema.
Requisitos correlacionados	F6, NF6.1, NF6.2, NF6.3, NF6.4, NF6.5
Fluxo Principal	

Respeitar o fluxo principal do caso de uso [UC2] .
Fluxo Alternativo
Respeitar o fluxo alternativo do caso de uso [UC2] .
Tratamento de exceções
Respeitar o tratamento de exceções do caso de uso [UC2] .

Quadro 23 - Caso de uso UC7 - Promover Produto
Fonte: Autoria própria

Caso de Uso: UC8 – Controlar Funcionamento Estabelecimento	
Descrição	Este caso de uso tem por objetivo permitir ao estabelecimento controlar seu funcionamento para que os clientes possam realizar seus pedidos.
Ator(es)	Administrador.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Estabelecimento aberto ou fechado no sistema.
Requisitos correlacionados	F7, NF7.1
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Solicita ao sistema a funcionalidade desejada; 2. [Sistema] Apresenta a opção “Abrir estabelecimento”; [A8.1]; 3. [Ator] Aciona a opção “Abrir estabelecimento”; 4. [Sistema] Realiza a abertura do estabelecimento no sistema. 	
Fluxo Alternativo	
[A8.1 – Fechar estabelecimento]	
<ol style="list-style-type: none"> 1. [Sistema] O Estabelecimento estando aberto apresenta a opção “Fechar estabelecimento”; 2. [Ator] Aciona a opção “Fechar estabelecimento”; 3. [Sistema] Realiza o fechamento do estabelecimento no sistema; Fim do caso de uso;	
Tratamento de exceções	

Quadro 24 - Caso de uso UC8 - Controlar Funcionamento Estabelecimento
Fonte: Autoria própria

Caso de Uso: UC9 – Realizar Pedido	
Descrição	Este caso de uso tem por objetivo permitir a realização de um pedido.
Ator(es)	Cliente.
Pré-condição	Ter ao menos um produto incluso em banco de dados e não bloqueado.
Pós-condição	Pedido realizado com sucesso.
Requisitos correlacionados	F8, NF8.1, NF8.2, NF8.3, NF8.4, NF8.5, NF8.6, NF8.7
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Seleciona um cardápio do catálogo Web; 2. [Sistema] Apresenta os estabelecimentos que possui produto para o cardápio selecionado; [A9.1]; [E9.1] 3. [Ator] Seleciona o estabelecimento desejado; 4. [Sistema] Apresenta os produtos de acordo com cardápio e estabelecimento selecionado; [A9.2]; [E9.1]; 5. [Ator] Aciona a opção “Pedir” dos produtos listados; 6. [Sistema] Apresenta um formulário em uma interface modal para a visualização do produto com a opção para adicionar ao pedido, cancelar visualização, informar quantidade, selecionar opção do produto caso possua, desmarcar ingredientes a remover do produto caso possua e selecionar ingredientes a acrescentar caso possua; 7. [Ator] Aciona a opção “Adicionar”; [A9.6]; [A9.7]; [A9.8]; [A9.9]; [A9.10]; 8. [Sistema] Adiciona o produto ao pedido; [E9.2]; [E9.3]; <p style="text-align: center;">Retorna ao passo 4 deste fluxo.</p>	
Fluxo Alternativo	

[A9.1 – Estabelecimento selecionado]

Estando com estabelecimento selecionado apresenta o passo 4 do fluxo principal.

[A9.2 – Finalizar pedido]

1. [Ator] Aciona a opção “Carrinho de compras”;
2. [Sistema] Apresenta a listagem dos produtos adicionados ao pedido por estabelecimento com seus passos: Pedido (Selecionado), Identificação, Finalizar pedido e Concluído; **[E9.1]**;
3. [Ator] Aciona a opção “Prosseguir” no passo “Pedido”; **[A9.3]**; **[A9.11]**; **[A9.12]**;
4. [Sistema] Apresenta o passo “Identificação”; **[A9.4]**;
5. Executa o caso de uso **[UC1]**;
6. [Sistema] Apresenta o passo “Finalizar Pedido”;
7. [Sistema] Apresenta o resumo do pedido com seus produtos listados por estabelecimento com a opção “Finalizar” e a opção “Alterar endereço” caso não acionado a opção “Buscar Pedido” para todos os estabelecimentos;
8. [Ator] Aciona a opção “Finalizar”; **[A9.5]**;
9. [Sistema] Realiza a gravação do pedido e endereço de entrega caso não acionado a opção “Buscar Pedido”, por estabelecimento em banco de dados; **[E9.2]**; **[E9.3]**;
10. [Sistema] Apresenta o passo “Concluído”;
11. [Sistema] Apresenta o(s) número(s) do(s) pedido(s) gerado(s) ao ator;
12. Fim do caso de uso.

[A9.3 – Buscar pedido]

1. [Ator] Aciona a opção “Buscar pedido” presente ao final dos produtos do estabelecimento;
2. [Sistema] Se marcado a opção “Buscar pedido” diminui o valor da taxa de entrega senão aumenta o valor da taxa de entrega ao total do pedido do estabelecimento;

Retorna ao passo 2 do fluxo alternativo **[A9.2]**.

[A9.4 – Usuário autenticado]

Estando o usuário autenticado apresenta o passo 6 do fluxo alternativo **[A9.2]**.

[A9.5 – Alterar endereço]

1. [Ator] Aciona a opção “Alterar endereço” presente ao final do resumo;
2. [Sistema] Habilita os dados de endereço para alteração;
3. [Ator] Altera os dados desejados;

Retorna ao passo 7 do fluxo alternativo **[A9.2]**.

[A9.6 – Cancelar visualização]

1. [Ator] Aciona a opção “Cancelar”;
2. [Sistema] Fecha a interface de visualização do produto;

Retorna ao passo 4 do fluxo principal.

[A9.7 – Informar quantidade]

1. [Ator] Informa a quantidade desejada do produto;

Retorna ao passo 6 do fluxo principal.

[A9.8 – Opção do produto]

1. [Ator] Seleciona a opção desejada referente ao produto;

Retorna ao passo 6 do fluxo principal.

[A9.9 – Remover ingredientes]

1. [Ator] Desmarca os ingredientes desejados que estejam disponíveis para remoção referente ao produto;

Retorna ao passo 6 do fluxo principal.

[A9.10 – Acrescentar ingredientes]

1. [Ator] Seleciona os ingredientes desejados para acrescentar ao produto;

Retorna ao passo 6 do fluxo principal.

[A9.11 – Alterar quantidade]

1. [Ator] Aciona a opção “Editar” presente nos produtos da listagem;
2. [Sistema] Habilita quantidade do produto selecionado para alteração;
3. [Ator] Informa a quantidade desejada para o produto e solicita alteração;
4. [Sistema] Atualiza total do pedido para o estabelecimento;

Retorna ao passo 2 do fluxo alternativo **[A9.2]**.

[A9.12 – Remover produto]

1. [Ator] aciona a opção “Excluir” presente nos produtos da listagem;
2. [Sistema] Remove o produto selecionado da listagem;
3. [Sistema] Atualiza total do pedido para o estabelecimento;

Retorna ao passo 2 do fluxo alternativo [A9.2] .
Tratamento de exceções
<p>[E9.1 – Nenhum registro encontrado]</p> <p>1. [Sistema] Apresenta na listagem a informação que nenhum registro foi encontrado;</p> <p>Não permite prosseguir com o próximo passo do fluxo em questão.</p> <p>[E9.2 – Dados obrigatórios ou inválidos]</p> <p>1. [Sistema] Destaca os dados obrigatórios não informados e os dados informados inválidos;</p> <p>Retorna ao passo 6 do fluxo principal ou ao passo 7 do fluxo alternativo [A9.2].</p> <p>[E9.3 – Estabelecimento fechado]</p> <p>1. [Sistema] Apresenta mensagem ao ator que o estabelecimento está fechado, não podendo realizar o pedido para o mesmo;</p> <p>Retorna ao passo 6 do fluxo principal ou ao passo 7 do fluxo alternativo [A9.2].</p>

Quadro 25 - Caso de uso UC9 - Realizar Pedido
 Fonte: Autoria própria

Caso de Uso: UC10 – Acompanhar Pedido	
Descrição	Este caso de uso tem por objetivo permitir a acompanhar os pedidos realizados.
Ator(es)	Cliente, Entregador, Funcionário.
Pré-condição	Ter executado o caso de uso [UC1] com sucesso.
Pós-condição	Pedido acompanhado.
Requisitos correlacionados	F9, NF9.1, NF9.2, NF9.3, NF9.4, NF9.5, NF9.6, NF9.7, NF9.8, NF9.9, NF9.10
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Solicita ao sistema a funcionalidade desejada; 2. [Sistema] Apresenta um formulário com as opções de busca de acordo com usuário autenticado; 3. [Ator] Aciona uma opção sem filtros de busca; [A10.1]; 4. [Sistema] Apresenta os pedidos encontrados em uma listagem a partir da opção selecionada. [E10.1]; [A10.2]; [A10.3]; 	
Fluxo Alternativo	

<p>[A10.1 – Opção com filtro de busca]</p> <ol style="list-style-type: none"> 1. [Ator] Aciona uma opção com filtros de busca; 2. [Sistema] Apresenta um formulário com os filtros de busca de acordo com a opção selecionada; 3. [Ator] Informa os filtros necessários e solicita a pesquisa; 4. [Sistema] Realiza a busca pelos filtros informados; [E10.2]; <p>Retorna ao passo 4 do fluxo principal.</p> <p>[A10.2 – Visualizar pedido]</p> <ol style="list-style-type: none"> 1. Executa o caso de uso [UC11]. <p>[A10.3 – Alterar status pedido]</p> <ol style="list-style-type: none"> 1. Executa o caso de uso [UC12].
Tratamento de exceções
<p>[E10.1 – Nenhum registro encontrado]</p> <ol style="list-style-type: none"> 1. [Sistema] Apresenta na listagem a informação que nenhum registro foi encontrado. <p>[E10.2 – Dados obrigatórios ou inválidos]</p> <ol style="list-style-type: none"> 1. [Sistema] Destaca os dados obrigatórios não informados e os dados informados inválidos. <p>Retorna ao passo 2 do fluxo alternativo [A10.1].</p>

Quadro 26 - Caso de uso UC10 - Acompanhar Pedido
Fonte: Autoria própria

Caso de Uso: UC11 – Visualizar Pedido	
Descrição	Este caso de uso tem por objetivo permitir a visualização do pedido.
Ator(es)	Cliente, Entregador, Funcionário.
Pré-condição	Ter executado o caso de uso [UC10] com sucesso.
Pós-condição	Pedido visualizado.
Requisitos correlacionados	F10, NF10.1
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Visualizar” presente nos pedidos listados; 2. [Sistema] Apresenta um formulário em uma interface modal para a visualização dos produtos do pedido com a opção “Fechar”; 3. [Ator] Aciona a opção “Fechar”; 	

4. [Sistema] Fecha a interface de visualização dos produtos.
Fluxo Alternativo
Tratamento de exceções

Quadro 27 - Caso de uso UC11 - Visualizar Pedido
Fonte: Autoria própria

Caso de Uso: UC12 – Alterar Pedido	
Descrição	Este caso de uso tem por objetivo permitir a realização de um pedido.
Ator(es)	Cliente, Entregador, Funcionário.
Pré-condição	Ter executado o caso de uso [UC10] com sucesso.
Pós-condição	Pedido alterado com sucesso.
Requisitos correlacionados	F11, NF11.1
Fluxo Principal	
<ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Avançar” presente nos pedidos listados; [A12.1]; [A12.2]; [A12.3]; 2. [Sistema] Altera o status do pedido para o próximo nível de hierarquia de avanço em banco de dados. 	
Fluxo Alternativo	
[A12.1 – Cancelar pedido]	
<ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Cancelar” presente nos pedidos listados; 2. [Sistema] Altera o status do pedido para “Cancelado” em banco de dados; Fim caso de uso.	
[A12.2 – Pendurar pedido]	
<ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Pendurar” presente nos pedidos listados; 2. [Sistema] Altera o status do pedido para “Pendurado” em banco de dados; Fim caso de uso.	
[A12.3 – Despendurar pedido]	
<ol style="list-style-type: none"> 1. [Ator] Aciona a opção “Despendurar” presente nos pedidos listados; 2. [Sistema] Altera o status do pedido para “Aberto” em banco de dados; Fim caso de uso.	
Tratamento de exceções	

Quadro 28 - Caso de uso UC12 - Alterar Pedido
Fonte: Autoria própria

3.3 DIAGRAMA DE CLASSE DE ANÁLISE

Para apresentar as classes de entidades (Entity Beans) do sistema foi utilizado o diagrama de classe de análise (figura 10).

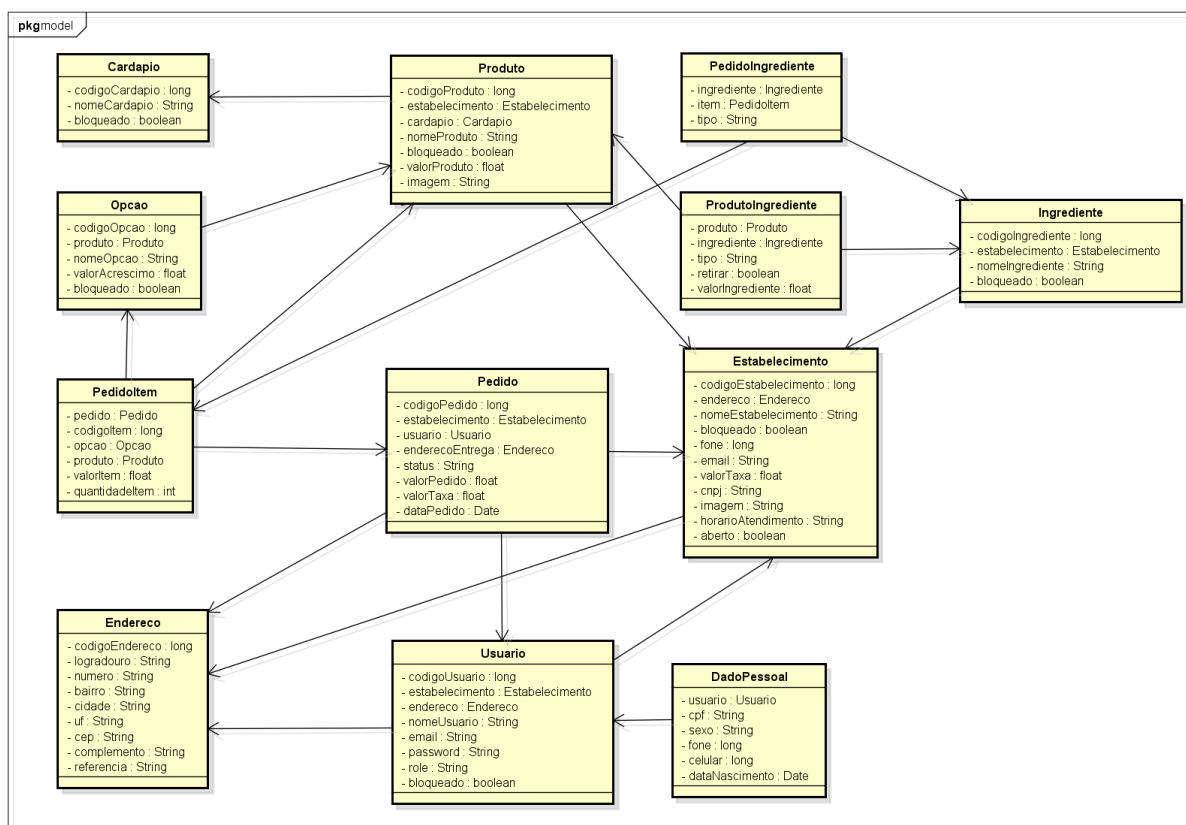


Figura 10 - Diagrama de classe de análise do sistema
Fonte: Autoria própria

3.4 DIAGRAMA ER

Uma estrutura bem definida do banco de dados do sistema pode ser visto pelo diagrama entidade relacionamento (figura 11).

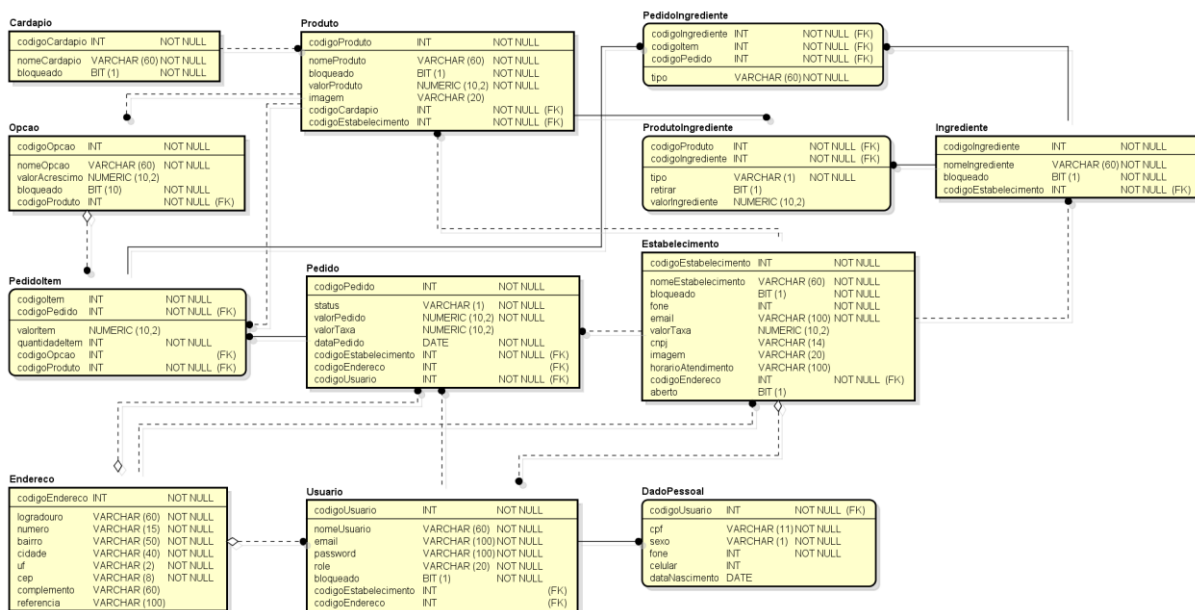


Figura 11 - Diagrama ER do sistema
Fonte: Autoria própria

4 RESULTADOS

Este capítulo apresenta os resultados do desenvolvimento do site através dos objetivos alcançados após o estudo teórico e bibliográfico realizado nos capítulos 1 e 2 que mostrou o problema, a ideia e a metodologia. No levantamento de requisitos necessários para a construção do sistema e a aplicação da metodologia, apresentados no capítulo 3.

Na construção do sistema foi utilizado o EJB *Stateless Session Bean* do tipo local para conter toda a regra de negócio para cada *Entity Bean*. No carrinho de compras do sistema não foi utilizado o *Stateful Session Bean* e sim um objeto de sessão. Como implementação do JPA foi utilizado o *hibernate*. As páginas *Web* foram organizadas por pastas sendo utilizado o *PrettyFaces* para tornar as URLs mais amigáveis.

Para o desenvolvimento do sistema foi utilizado à versão 7 do *Java*. A interface de usuário foi utilizada a versão 2.2 do JSF junto com a versão 5.0 do *Primefaces*. Para o EJB foi utilizado à versão 3.1. Foi utilizada a versão 4.0.1 do *hibernate*. JPA foi utilizado à versão 2.0. Por fim, foi utilizada a versão 2.0.12 do *PrettyFaces*.

4.1 APRESENTAÇÃO SISTEMA

Ao acessar o sistema através um *browser* é apresentado à interface principal do sistema onde o usuário pode navegar pelo menu do cardápio que lista os cardápios cadastrados no sistema. Também é possível solicitar a interface de acesso através do menu clicando sobre a imagem usuário.

4.2 USUÁRIO MASTER DO SISTEMA

No sistema existe apenas um usuário Master, que é responsável para manutenção dos cardápios e dos estabelecimentos do sistema ilustrado na figura 12.

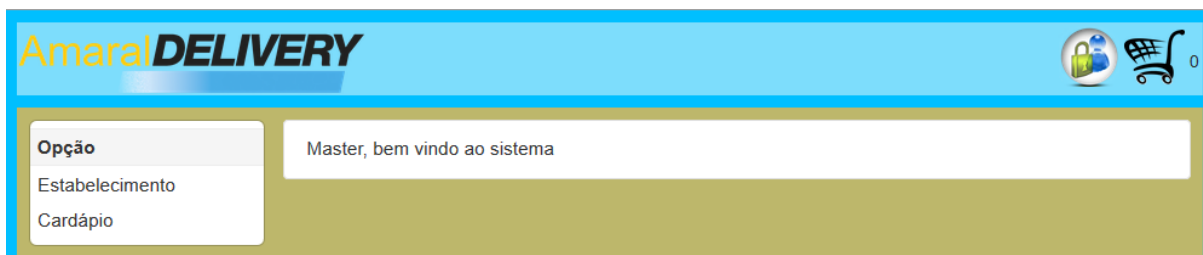
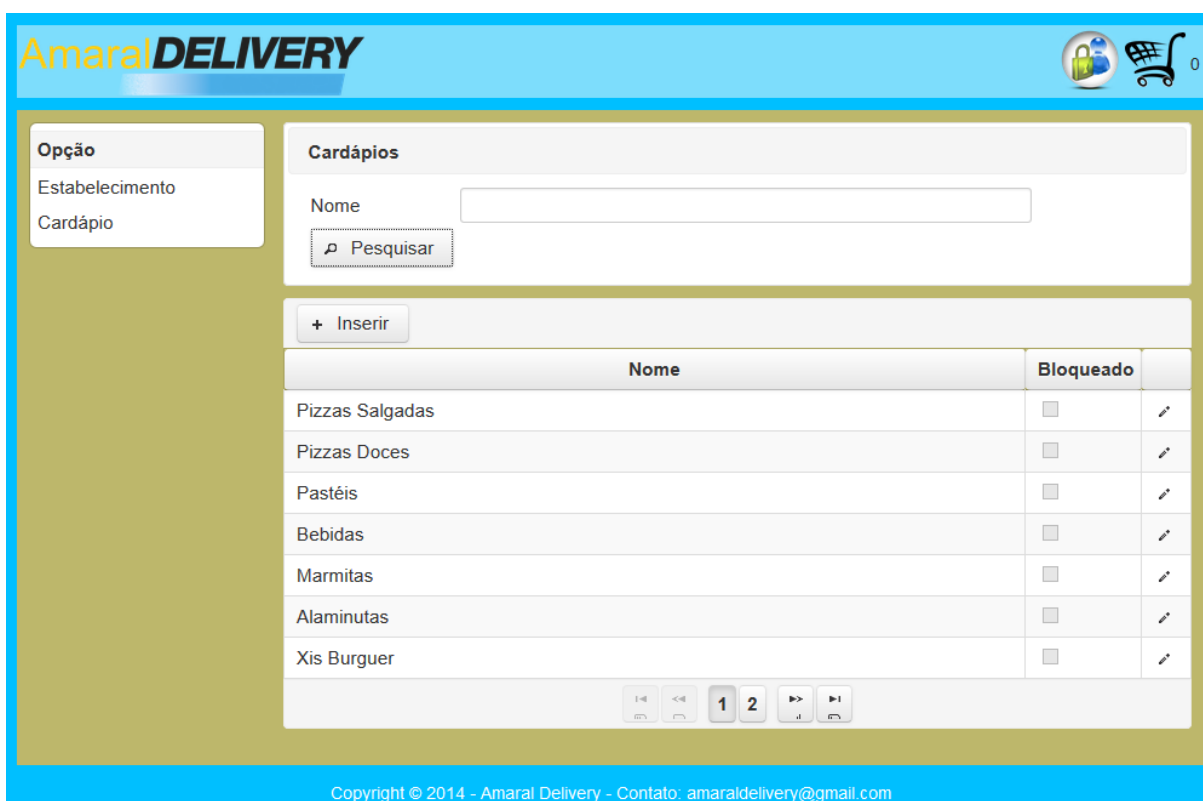


Figura 12 - Interface acesso usuário Master
Fonte: Autoria própria

Na figura 13 mostra as informações referentes à pesquisa de cardápio do sistema, onde é possível pesquisar, incluir e alterar cardápio.



Copyright © 2014 - Amaral Delivery - Contato: amaraldelivery@gmail.com

Figura 13 - Interface pesquisa de cardápio
Fonte: Autoria própria

Na figura 14 mostra os dados referentes à manutenção de cardápios do sistema.

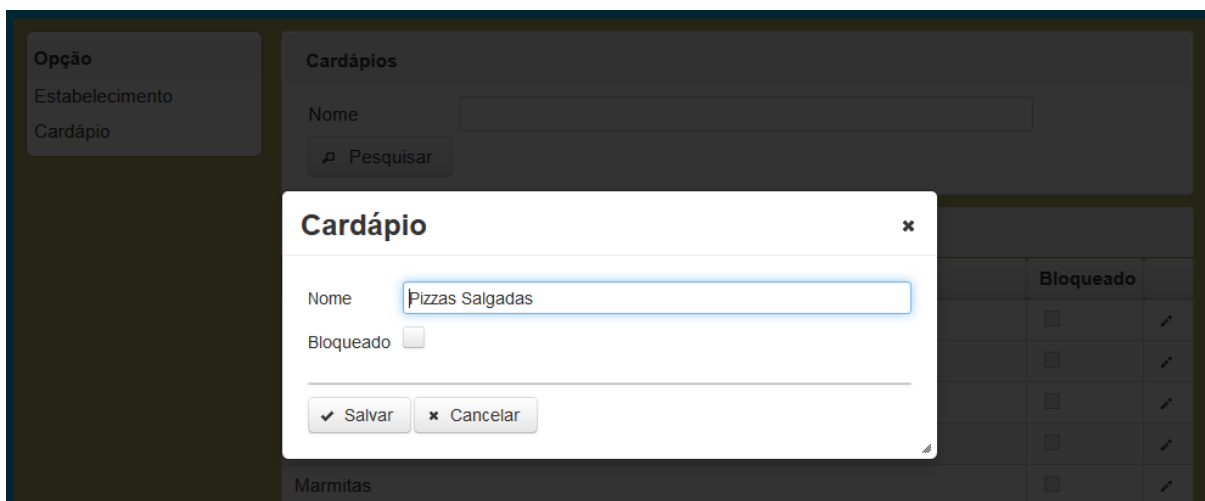


Figura 14 - Interface cadastro de cardápio

Fonte: Autoria própria

Na figura 15 mostra as informações da pesquisa de estabelecimento, onde é possível pesquisar, incluir e alterar estabelecimento. Passando com o mouse sobre o nome do estabelecimento é apresentado à imagem cadastrado do mesmo.

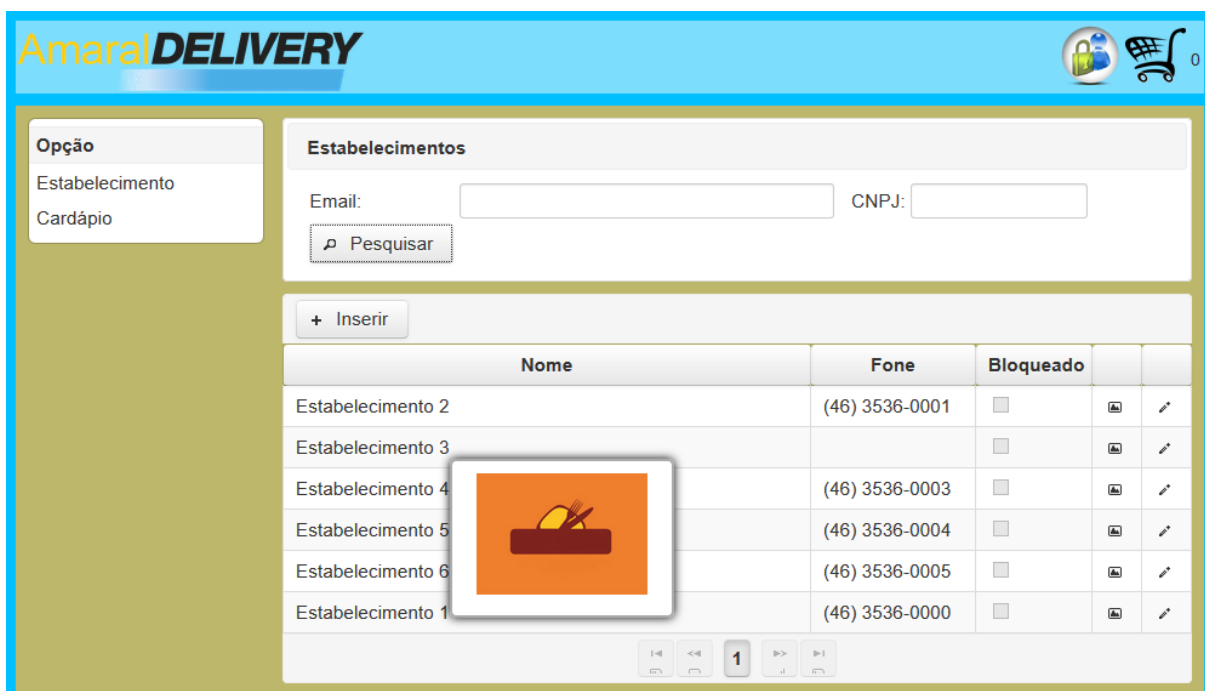


Figura 15 - Interface de pesquisa de estabelecimento

Fonte: Autoria própria

Na manutenção de estabelecimento ao inserir um novo registro, o sistema cria um usuário com perfil administrador, através do nome e e-mail do estabelecimento, enviando um e-mail para que o usuário possa alterar sua senha. O mesmo e-mail é enviado quando o campo "Alterar Senha" é marcado e clicado no

botão salvar. Os dados do estabelecimento são mostrados na figura 16, sendo que o campo “Alterar Senha” é apresentado apenas na alteração.

Copyright © 2014 - Amaral Delivery - Contato: amaraldelivery@gmail.com

Figura 16 - Interface cadastro de estabelecimento
Fonte: Autoria própria

A imagem do estabelecimento, a qual será apresentada no cardápio, pode ser incluída ou alterada através da fotografia listada na pesquisa de estabelecimento (figura 15). É possível informar a imagem e enviar para o sistema conforme figura 17.

1

Figura 17 - Interface imagem estabelecimento
Fonte: Autoria própria

4.3 USUÁRIOS DO ESTABELECIMENTO

O usuário administrador é responsável pela administração do estabelecimento em questão no sistema. Este usuário é responsável pela manutenção de ingredientes, produto, usuários e o controle de funcionamento do estabelecimento conforme mostrado na figura 18.

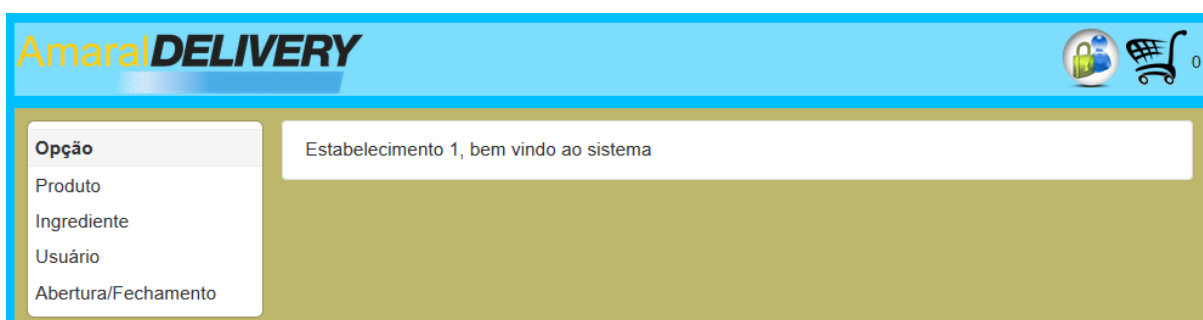


Figura 18 - Interface perfil acesso administrador
Fonte: Autoria própria

Na figura 19 apresenta a interface para controle do estabelecimento pelo usuário administrador. O sistema verifica se o estabelecimento está aberto ou fechado mostrando a opção para abrir ou fechar o estabelecimento.

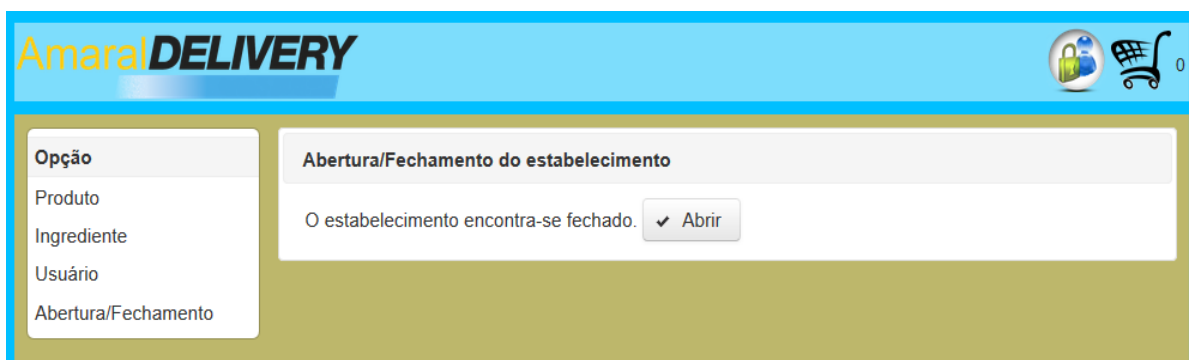


Figura 19 - Interface Abertura/Fechamento de estabelecimento
Fonte: Autoria própria

A figura 20 mostra a interface de pesquisa de usuários onde é possível pesquisar, alterar ou inserir usuário.



Figura 20 - Interface pesquisa de usuários
Fonte: Autoria própria

Na figura 21 mostrar os dados da manutenção de usuário referente ao estabelecimento.

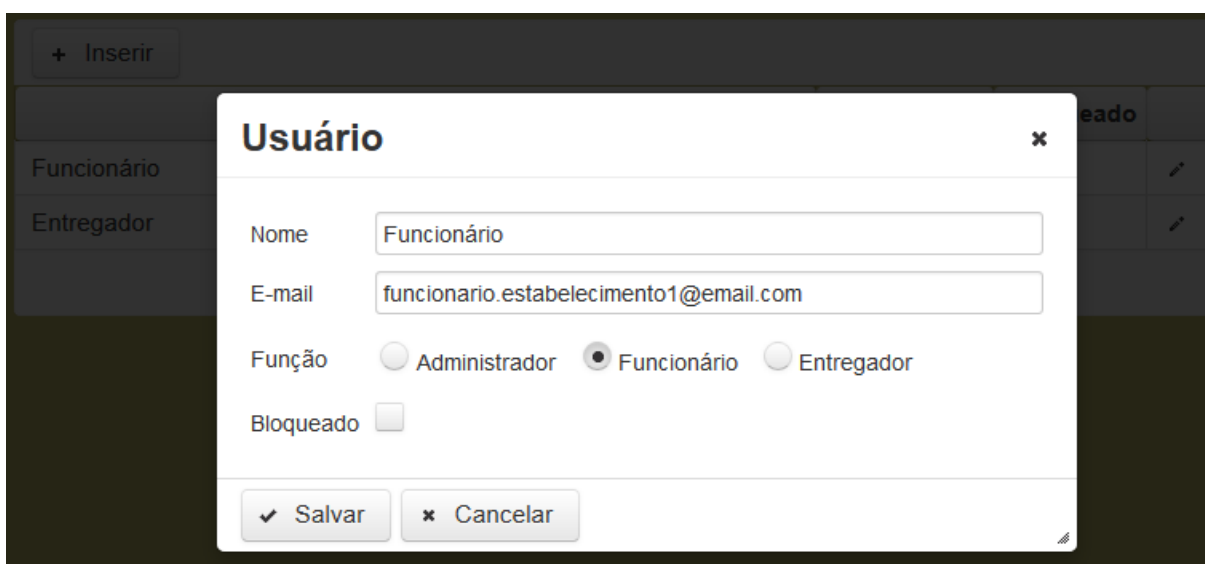


Figura 21 - Interface manutenção de usuário
Fonte: Autoria própria

Na figura 22 mostra as informações referentes à pesquisa de ingredientes, onde é possível pesquisar, incluir e alterar ingrediente.



Figura 22 - Interface de pesquisa de ingredientes
Fonte: Autoria própria

Na figura 23 mostra os dados da manutenção de ingrediente referente ao estabelecimento.

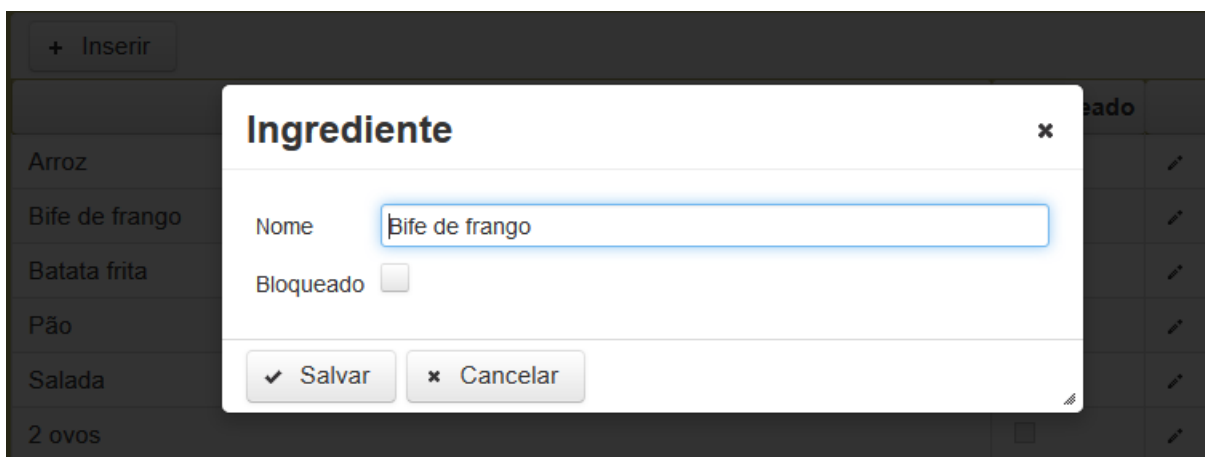


Figura 23 - Interface de manutenção de ingredientes
Fonte: Autoria própria

Na figura 24 são mostrados os dados referentes à pesquisa de produtos, onde é possível pesquisar, incluir ou alterar produto. Passando o mouse é apresentada a imagem do produto.

The screenshot shows the 'Amaral DELIVERY' web application interface. At the top, there is a blue header with the logo and a shopping cart icon. Below the header, there is a search bar with the text 'Produtos' and a 'Pesquisar' button. To the left, there is a sidebar with the title 'Opção' and a list of options: 'Produto', 'Ingrediente', 'Usuário', and 'Abertura/Fechamento'. Below the search bar, there is a '+ Inserir' button. The main content area displays a table of products with the following columns: 'Nome', 'Cardápio', 'Valor', and 'Bloqueado'. The table contains the following data:

Nome	Cardápio	Valor	Bloqueado
Pepsi	Bebidas	R\$ 5,50	<input type="checkbox"/>
Guaraná antarctica	Bebidas	R\$ 5,00	<input type="checkbox"/>
Calabresa	Pizzas Salgadas	R\$ 25,00	<input type="checkbox"/>
Estrogono	Pizzas Salgadas	R\$ 26,00	<input type="checkbox"/>
Filé catupi	Pizzas Salgadas	R\$ 25,00	<input type="checkbox"/>
Dois amor	Pizzas Doces	R\$ 26,00	<input type="checkbox"/>
Sonho de	Pizzas Doces	R\$ 22,00	<input type="checkbox"/>

An image of a pizza is overlaid on the table, showing a close-up of a pizza with various toppings.

Figura 24 - Interface de pesquisa de produtos

Fonte: Autoria própria

Na manutenção de produto as informações são organizadas por aba conforme ilustra a figura 25, mostrando as informações da aba principal. Para cadastrar a imagem para o produto é necessário acionar a opção “Imagem”, logo após acionar a opção “Enviar”, ao salvar o produto a imagem é salva relacionando ao produto em questão.

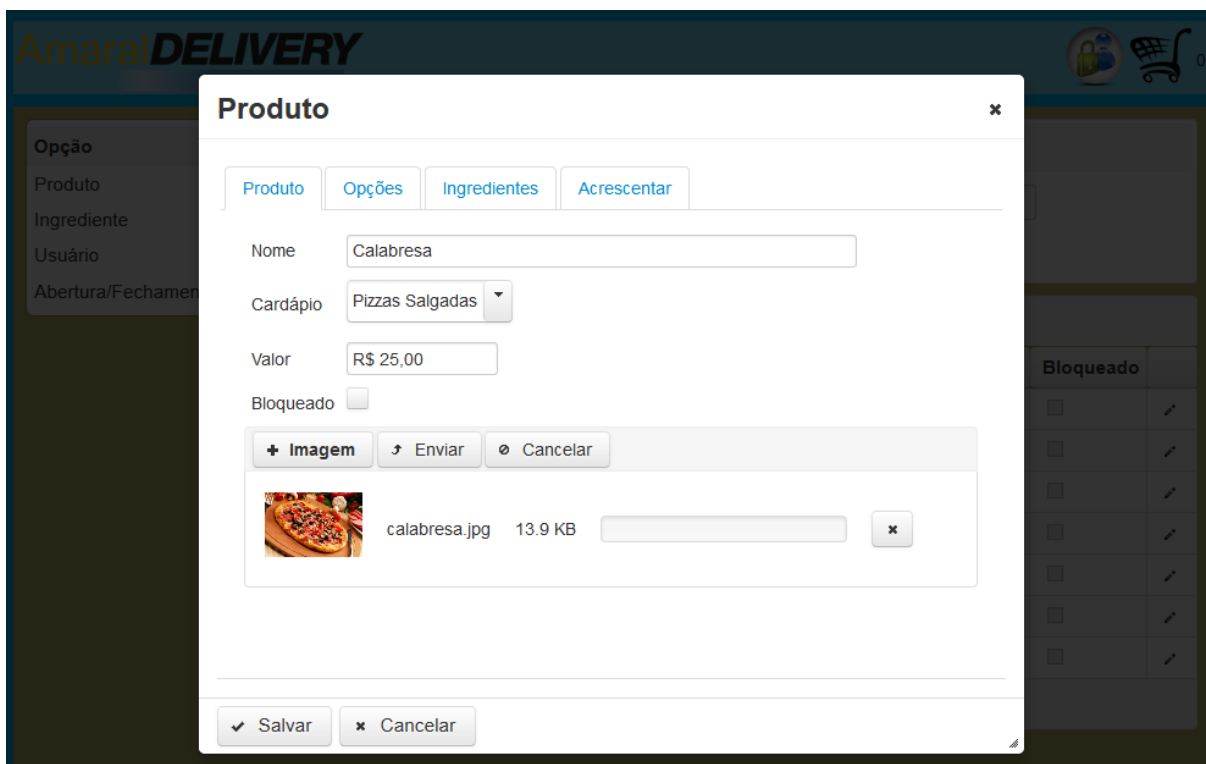


Figura 25 - Interface de manutenção de produto aba produto
Fonte: Autoria própria

Na figura 26 são mostrados os dados da aba opções da manutenção de produto. Nesta aba é possível adicionar ou alterar opções.

The screenshot shows a window titled "Produto" with a close button (X) in the top right corner. Below the title bar are four tabs: "Produto", "Opções", "Ingredientes", and "Acrecentar". The "Opções" tab is active. Below the tabs is a form for adding a new option, with fields for "Opção" (empty), "Acréscimo" (R\$ 0,00), and a "+" button. Below the form is a table with the following data:

Opção	Acréscimo	Bloqueado	
Normal	R\$ 0,00	<input type="checkbox"/>	
Mini	-R\$ 3,50	<input type="checkbox"/>	

At the bottom of the window are two buttons: "✓ Salvar" and "✕ Cancelar".

Figura 26 - Interface de manutenção de produto aba opções
Fonte: Autoria própria

Na figura 27 mostra os dados referentes à aba ingrediente da manutenção de produto. Nesta aba é possível incluir ou remover ingrediente referente ao produto.

Produto ×

Produto
Opções
Ingredientes
A acrescentar

Ingrediente Filé Retirar +

Ingrediente	Retirar	
Presunto	<input type="checkbox"/>	🗑
Pão	<input type="checkbox"/>	🗑
Maionese	<input type="checkbox"/>	🗑
Alface	<input type="checkbox"/>	🗑
Coração de galinha	<input type="checkbox"/>	🗑
Milho	<input checked="" type="checkbox"/>	🗑
Ervilha	<input checked="" type="checkbox"/>	🗑
Tomate	<input checked="" type="checkbox"/>	🗑

✓ Salvar
✕ Cancelar

Figura 27 - Interface de manutenção de produto aba ingredientes
 Fonte: Autoria própria

Na figura 28 mostra os dados referentes à aba acrescentar da manutenção de produto. Nesta aba é possível incluir ou remover ingredientes que podem ser acrescentados no produto.

Produto ×

Produto
Opções
Ingredientes
A acrescentar

Ingrediente Valor

Ingrediente	Valor	
Ovo	R\$ 3,00	
Queijo parmesão	R\$ 2,00	
Queijo provolone	R\$ 2,00	

Figura 28 - Interface de manutenção de produto aba acrescentar
 Fonte: Autoria própria

Nas abas ingredientes e acrescentar, os ingredientes não podem ser repetidos, devem estar em uma aba ou outra.

Outro usuário do estabelecimento é o funcionário que é responsável por fazer o controle dos pedidos conforme ilustrado na figura 29. Também seguindo o mesmo controle, o usuário entregador é responsável pelas entregas, podendo ser vistos os pedidos conforme apresentado nos requisitos.

Amaral DELIVERY 0

Funcionário Estabelecimento 1, bem vindo ao sistema

Figura 29 - Interface perfil de acesso funcionário e entregador
 Fonte: Autoria própria

Na figura 30 mostra as informações referentes aos pedidos realizados pertencentes ao estabelecimento, onde é possível acompanhar os pedidos pelas opções listadas, podendo fazer os encaminhamentos necessários.

Opção		Pedidos do dia	Pedidos em aberto	Pedidos em preparo	Pedidos aguardando		
Pedido		Pedidos a entregar	Pedidos entregues	Pedidos pendurados	Pedidos cancelados		
		Pedidos por número	Pedidos por data	Pedidos por cliente	Todos os pedidos		
	Pedido	Total	Data	Status	Entregar	φ	
🔍	12	R\$ 17,00	13/09/2014	Aberto	Sim	⊖ ⊕	μ
🔍	11	R\$ 37,00	02/09/2014	Aberto	Sim	⊖ ⊕	μ
🔍	10	R\$ 38,50	29/08/2014	Aberto	Sim	⊖ ⊕	μ
🔍	9	R\$ 25,00	29/08/2014	Em preparo	Não	⊖ ⊕	μ
🔍	8	R\$ 13,00	29/08/2014	Aberto	Sim	⊖ ⊕	μ
🔍	7	R\$ 9,00	29/08/2014	Entregue	Não		μ
🔍	6	R\$ 15,00	29/08/2014	Saiu para entrega	Sim	🔒 ⊖ ⊕	μ
🔍	5	R\$ 17,50	29/08/2014	Aberto	Sim	⊖ ⊕	μ
🔍	4	R\$ 28,50	29/08/2014	Cancelado	Sim		μ
🔍	3	R\$ 23,00	29/08/2014	Saiu para entrega	Sim	🔒 ⊖ ⊕	μ

Figura 30 - Interface de acompanhamento de pedido pelo estabelecimento
Fonte: Autoria própria

É possível visualizar a informação cliente e endereço de entrega referente ao pedido, acionando a opção de expansão pela imagem ao início das informações do pedido na listagem conforme ilustrado na figura 31.

Opção	Pedidos do dia	Pedidos em aberto	Pedidos em preparo	Pedidos aguardando			
Pedido	Pedidos a entregar	Pedidos entregues	Pedidos pendurados	Pedidos cancelados			
	Pedidos por número	Pedidos por data	Pedidos por cliente	Todos os pedidos			
	Pedido	Total	Data	Status	Entregar	☺	
🔍	12	R\$ 17,00	13/09/2014	Aberto	Sim	☺ ☹	⌂
🔍	11	R\$ 37,00	02/09/2014	Aberto	Sim	☺ ☹	⌂
🔍	10	R\$ 38,50	29/08/2014	Aberto	Sim	☺ ☹	⌂
Cliente Cadastro 02 cadastro02@email.com (46) 3536-8900		Endereço de entrega Teste, 123 Teste 85660-000 Dois Vizinhos - PR Perto Parque exposições					
🔍	9	R\$ 25,00	29/08/2014	Em preparo	Não	☺ ☹	⌂
🔍	8	R\$ 13,00	29/08/2014	Aberto	Sim	☺ ☹	⌂

Figura 31 - Interface de visualização de pedido na listagem pelo estabelecimento
Fonte: Autoria própria

4.4 USUÁRIO CLIENTE

Pela interface de acesso informando o e-mail e acionando a opção “Cadastrar” o sistema valida se o *e-mail* está disponível para cadastrar, direcionando para a interface de cadastro de usuário conforme ilustra a figura 32. No formulário é possível informar os dados de acesso, dados pessoais e dos dados de entrega. O usuário deste cadastro é exclusivamente para realizar pedidos a qualquer estabelecimento cadastrado no sistema.

Figura 32 - Interface cadastro usuário cliente
Fonte: Autoria própria

Os usuários cliente podem acompanhar o andamento dos pedidos realizados, alterar dados e alterar senha conforme ilustrado na figura 33. Este usuário também pode realizar pedidos para qualquer estabelecimento.

Figura 33 - Interface perfil de acesso cliente
Fonte: Autoria própria

Ao selecionar um cardápio sem ter sido selecionado um estabelecimento o sistema apresenta os estabelecimentos que possuem produtos referentes ao cardápio selecionado. As informações do estabelecimento são mostradas na figura 34.



Figura 34 - Interface de listagem de estabelecimento por cardápio
Fonte: Autoria própria

Selecionando um estabelecimento ou cardápio (já selecionado o estabelecimento) são apresentados os produtos do estabelecimento e cardápio selecionados. As informações são apresentadas pela figura 35.

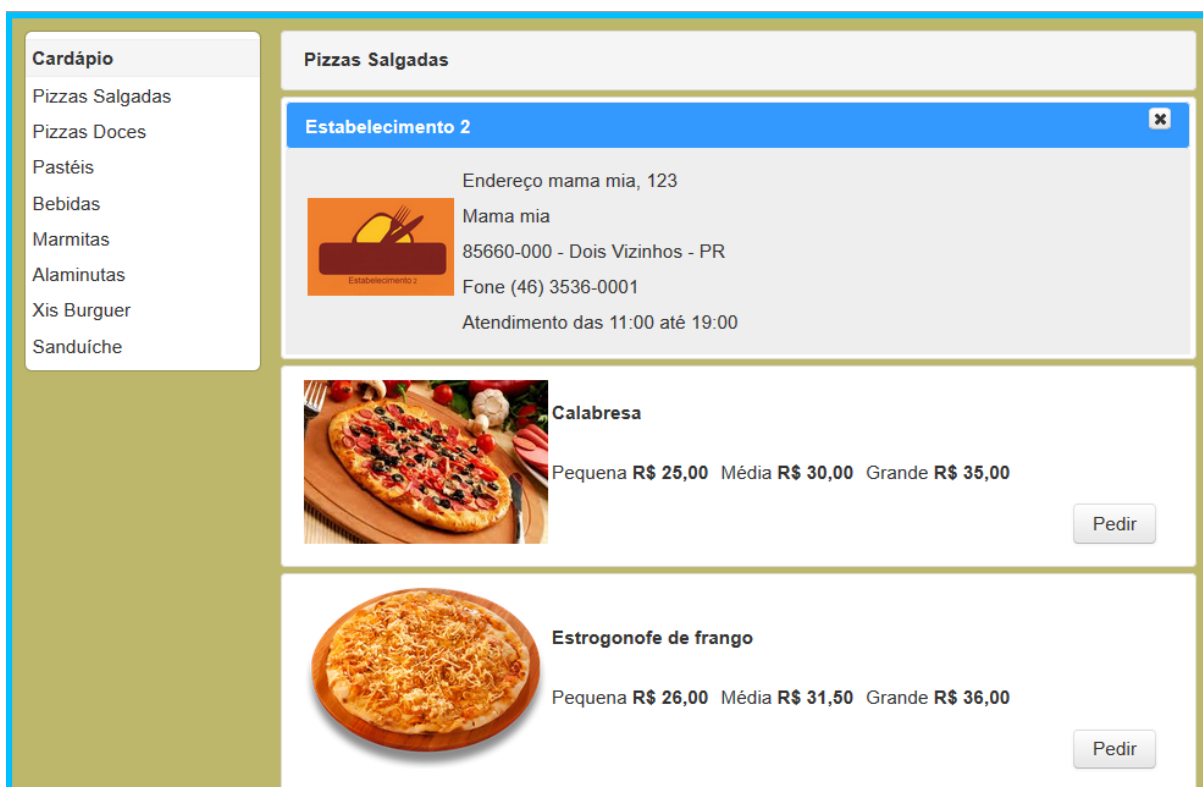


Figura 35 - Interface de listagem de produto por estabelecimento
Fonte: Autoria própria

Acionando a opção “Pedir” na listagem de produtos (figura 35) é apresentado à visualização de produto com as opções referentes ao produto conforme mostra a figura 36.

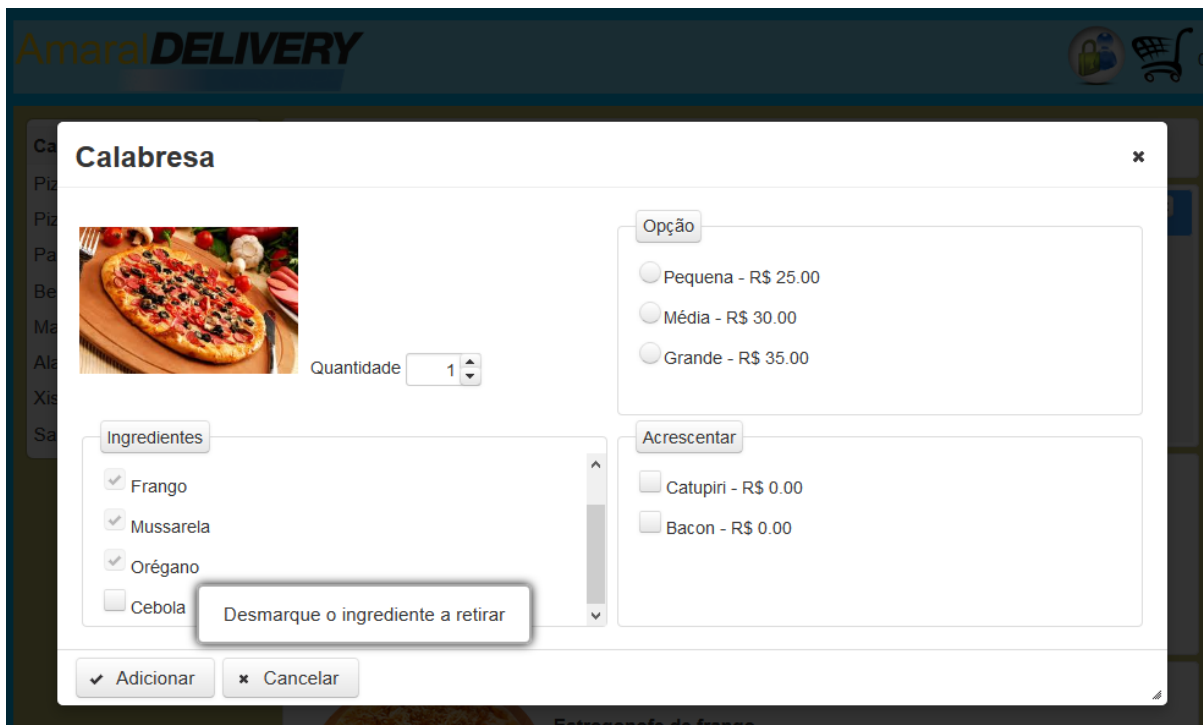


Figura 36 - Interface de visualização de produto
Fonte: Autoria própria

Acionando o menu “Produtos” pela imagem carrinho são apresentados os produtos selecionados, por estabelecimento conforme mostra as informações na figura 37.




Figura 37 - Interface de carrinho de compra passo pedido

Fonte: Autoria própria

Acionando a opção “Proseguir” do carrinho de compra (figura 37) e o usuário não estar autenticado no sistema é apresentado o passo para identificação com a opção de informar o usuário já cadastrado ou realizar o cadastro (figura 32) no sistema.

O passo finalizar compra apresenta um resumo do pedido a ser realizado, podendo o usuário alterar os dados de entrega caso não for buscar o pedido no estabelecimento como ilustra a figura 38.

Estabelecimento 2


 Endereço mama mia, 123
 Mama mia
 85660-000 - Dois Vizinhos - PR
 Fone: (46) 3536-0001

Produto	Quantidade	Unidade	Total
Calabresa (Média)			
(-) Retirar -> Cebola	1	R\$ 30,00	R\$ 30,00
(+) Acrescentar -> Bacon			
Taxa de entrega: R\$ 3,50			R\$ 33,50

Endereço entrega

Alterar endereço

CEP:

Endereço:

Número:

Complemento:

Bairro:


Cidade:


Estado:

Referência:

Figura 38 - Interface de carrinho de compra passo finalizar pedido
 Fonte: Autoria própria

Após acionado a opção “Finalizar” (figura 38) o sistema apresenta número do pedido ou pedidos conforme figura 39.

Amara! **DELIVERY**



Recebemos seu(s) pedido(s).
 Você pode acompanhar o andamento do(s) pedido(s) através do site.

Número do seu pedido é 13

Figura 39 - Interface de carrinho de compra passo concluído
 Fonte: Autoria própria

No menu de acesso o usuário estando autenticado, o sistema disponibiliza a opção “Meus pedidos” onde é possível acompanhar os pedidos realizados de acordo com as opções listadas conforme figura 40.

Amara DELIVERY

Minha conta
Meus pedidos
Alterar dados
Alterar senha

Últimos pedidos | Pedidos em aberto | Pedidos entregues | Pedidos por número
Pedidos por data | Todos os pedidos

	Pedido	Total	Data	Status	Entregar	φ	
▶	10	R\$ 33,50	29/08/2014	Aberto	Sim	⊖	⌂
	9	R\$ 25,00	29/08/2014	Em preparo	Não	⊖	⌂
▶	8	R\$ 8,00	29/08/2014	Aberto	Sim	⊖	⌂
	7	R\$ 9,00	29/08/2014	Entregue	Não		⌂
▶	6	R\$ 10,00	29/08/2014	Saiu para entrega	Sim		⌂
▶	5	R\$ 12,50	29/08/2014	Aberto	Sim	⊖	⌂
▶	4	R\$ 23,50	29/08/2014	Cancelado	Sim		⌂

Estabelecimento
Estabelecimento 1
Av. Rosseti, 206
Bairro PIO X
85660-000 - Dois Vizinhos - PR
Fone (46) 3536-3136

Endereço de entrega
Teste, 123
Teste
85660-000 Dois Vizinhos - PR
Perto Parque exposições

Figura 40 - Interface de acompanhamento de pedido pelo cliente
Fonte: Autoria própria

É possível visualizar a informação estabelecimento e endereço de entrega referente ao pedido, acionando a opção de expansão pela imagem ao início das informações do pedido na listagem conforme ilustrado na figura 41.

Minha conta
Meus pedidos
Alterar dados
Alterar senha

Últimos pedidos | Pedidos em aberto | Pedidos entregues | Pedidos por número
Pedidos por data | Todos os pedidos

	Pedido	Total	Data	Status	Entregar	φ	
▶	10	R\$ 33,50	29/08/2014	Aberto	Sim	⊖	⌂
	9	R\$ 25,00	29/08/2014	Em preparo	Não	⊖	⌂
▶	8	R\$ 8,00	29/08/2014	Aberto	Sim	⊖	⌂
	7	R\$ 9,00	29/08/2014	Entregue	Não		⌂
▶	6	R\$ 10,00	29/08/2014	Saiu para entrega	Sim		⌂

Estabelecimento
Estabelecimento 1
Av. Rosseti, 206
Bairro PIO X
85660-000 - Dois Vizinhos - PR
Fone (46) 3536-3136

Endereço de entrega
Teste, 123
Teste
85660-000 Dois Vizinhos - PR
Perto Parque exposições

Figura 41 - Interface de visualização de pedido na listagem pelo cliente
Fonte: Autoria própria

Na figura 42 mostra as informações da visualização de pedido. A mesma visualização é usada pelos usuários funcionário, entregador e cliente acionando a opção de visualização do pedido (figura 30 e 40).



Produto	Quantidade	Valor	Total
Calabresa (Média)			
(-) Retirar -> Cebola	1	R\$ 30,00	R\$ 30,00
(+) Acrescentar -> Catupiri			
Taxa de entrega: R\$ 3,50	1		R\$ 33,50

Figura 42 - Interface de visualização de pedido
Fonte: Autoria própria

5 CONCLUSÃO

Durante o desenvolvimento desta etapa do trabalho, foi percebida a utilização dos diversos conhecimentos formados ao longo do curso de especialização de desenvolvimento de sistemas para internet e dispositivos moveis. Esta fase proporcionou principalmente a aplicação dos conhecimentos de pesquisa e desenvolvimento de sistemas, e também proporcionou o contato com novas metodologias e tecnologias.

A utilização do UML neste trabalho demonstrou uma demanda de trabalho e tempo grande, mostrando ao final uma melhor visualização de como será o sistema antes de apresentar os resultados.

Com as informações obtidas nas etapas, foi desenvolvido o sistema analisado, onde, depois de implementado, foi concluído que o sistema atingiu o objetivo proposto.

Respeitando a viabilidade de mercado do projeto e buscando o desenvolvimento de um produto inovador para a região, a análise do sistema buscou atender todos os requisitos necessários para que um *delivery* utilize os benefícios e facilidades de uma ferramenta de vendas web. Sendo assim, o *software* busca automatizar e facilitar o maior número possível de processos internos do estabelecimento.

5.1 PROBLEMAS ENCONTRADOS

Durante a implementação do sistema foram encontrados pequenos problemas que com pouco tempo de pesquisa se chegou à solução. Um problema que levou mais tempo foi à configuração do mapeamento do *servlet* no arquivo *web.xml*, o qual impactava no não funcionamento da API *FileUpload*.

5.2 TRABALHOS FUTUROS

O sistema apresentado não será comercializado no presente momento, apenas foi desenvolvido para realização do trabalho. Para que o sistema seja comercializado ou para fim de estudos, tem a necessidade de novas implementações e alterações que possa adequar à realidade das pessoas interessadas.

As principais alterações no sistema é melhorar a interface, controle de busca dos estabelecimentos por cidade ou CEP, cor de fundo para indicar se o estabelecimento está aberto ou fechado nas interfaces para os usuários do estabelecimento e a criação da versão mobile. Além das alterações será necessário fazer uma avaliação do sistema atual pelos estabelecimentos da região.

Com as informações coletadas pode ser feito a criação de relatórios que possam ajudar no apoio a decisões. Informações como qual melhor dia de venda, qual o produto mais vendido, qual a região mais atendida, entre outras.

REFERÊNCIAS

GUEDES, Gilleanes T. A. **UML 2: uma abordagem prática**. 2. ed. São Paulo: Novatec, 2009. 484 p.

LARMAN, Craig. **Utilizando UML e Padrões: Uma introdução à análise e ao projeto orientados a objetos e ao Processo Unificado**. 2. ed. Porto Alegre: Bookman, 2004. 607 p.

DEITEL, H. M., DEITEL, P. J. **Java Como Programar**. 8. ed. Prentice Hall, 2010. 1144 p.

COELHO, Hébert. **JPA eficaz: as melhores práticas de persistência de dados em Java**. Cada do Código, 2013. 167 p.

FREEMAN, Eric, FREEMAN, Elisabeth. **Use a Cabeça Padrões de Projetos**. 2. ed. Altabooks, 2007.

PRODANOV, Cleber Cristiano; FREITAS, Ernani Cesar de. **Metodologia do trabalho científico: métodos e técnicas de pesquisa e do trabalho acadêmico**. 2. ed. Rio Grande do Sul: Feevale, 2013. 276 p.

K19. **Desenvolvimento Web Avançado com JSF2, EJB3.1 e CDI**. São Paulo, 2013, 182 p. Apostila do Curso K20 - Formação Desenvolvedor Java Avançado.

I FOOD. Disponível em: <<https://www.ifood.com.br/lista-restaurantes>>. Acesso em 9. set. 2014.

PAPAJOHNS. Disponível em: <http://www.papajohns.com.pe/pedidos_online.php>. Acesso em 9. set. 2014.