

**CENTRO PAULA SOUZA  
FACULDADE DE TECNOLOGIA DE FRANCA  
“Dr. THOMAZ NOVELINO”**

**TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

**DANIEL LINCON BENDER DOS SANTOS  
LUANA DO NASCIMENTO FERREIRA**

**XSISTEMA:  
UM SOFTWARE PARA APOIAR O PEDIDO DE LANCHES**

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Franca - “Dr. Thomaz Novelino”, como parte dos requisitos obrigatórios para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Orientador: Prof. Dr. Daniel Facciolo Pires

**FRANCA/SP**

**2020**

Ficha catalográfica

F237s     Ferreira, Luana do Nascimento e Santo, Daniel Lincon  
Bender dos  
            XSISTEMA: um software para apoiar o pedido de  
lanches / Luana do Nascimento Ferreira e Daniel Lincon  
Bender dos Santo  
/ [ s.n], 2020

30 f.; 30 cm; il

Trabalho de Graduação (Curso Superior de Análise e  
Desenvolvimento de Sistemas) Fatec - Faculdade de  
Tecnologia "Dr. Thomaz Novelino".

Orientador: Prof. Dr. Daniel Facciolo Pires

1. Consumidor. 2.Equidade. 3.Lanchonete. 4.Preço  
justo. 5. Remover ingrediente. I. Autor. II. Título.

CDD – 005.1

**XSISTEMA:**  
UM SOFTWARE PARA APOIAR O PEDIDO DE LANCHES

**Daniel Lincon Bender dos Santos**

**Luana do Nascimento Ferreira**

### **Resumo**

Um software que traz benefícios para usuários de Fast-Food. É comum pessoas terem que tirar um ou mais ingredientes de um pedido já pronto por questão de saúde ou por não gostarem de alguns ingredientes. Entretanto, na maior parte dos casos que isso ocorre o cliente não recebe nenhum desconto ou bonificação por essa retirada. O objetivo principal deste trabalho é desenvolver um sistema que possa melhorar a experiência dos consumidores, com a retirada de um ou mais ingredientes o sistema calcula o preço do ingrediente removido e dado um bônus para o cliente. Bônus esse que pode ser utilizado futuramente na adição de ingrediente. Tudo será registrado em seu histórico de pedido, logo após se cadastrar e logar no sistema o usuário verá um conjunto de lojas selecionando uma delas ele verá o menu da loja com todos os produtos e ao selecionar um produto e exibido seus ingredientes com seus preços a quantidade de cada ingrediente pode ser alteradas de modo a agradar o cliente. Por ser um sistema com grande diferencial os donos de redes fast-foods podem usá-lo para garantir a fidelidade de seus clientes. Para o desenvolvimento deste projeto foi usado inicialmente as seguintes ferramentas: No backend o PHP, no front-end foi utilizado HTML, CSS, JavaScript. E o banco de dados MySQL. Foram realizadas pesquisas em sites e bibliografias, o levantamento de requisitos e regras de negócios foram feitos com base em uma pesquisa manual com perguntas aos frequentadores de fast-foods, foi utilizado também métodos ensinados pela faculdade para a construção de diagramas, modelagens e afins.

**Palavras-chave:** Consumidor, equidade, lanchonete, preço justo, remover ingrediente.

**Abstract**

*System that brings benefits for people who eat in Fast-Food establishments. The main goal of this System is to make a better experience for the Consumers with health limitations, such as: Lactose Intolerance, food colouring or ingredients that prevent a customer to have a much more pleasant experience when consuming a Burger and trying to adjust a fair price without harming the Fast-Food Company, because some people end up having to remove one or more ingredients by health condition or just dislike a specified component, will remove them anyway and that differentiation can make a customer lose the interest on going into these places. The system aims at maintain the equity on attending the consumer who is the base to any company. In a possible solution a customer would be removing one or more ingredients, the system calculates the price of the ingredients removed and gives a bonus to the customer, according to the score of the customer is growing, the customer has the possibility to use these scores to increase some ingredient that is in the menu, as a customer desires. Everything will be registered in your profile linked to the company.* For the development of the project, the following PHP tools were devised as a base together with HTML, CSS, JavaScript and MySQL together with searches on websites and bibliographies. For the development of this project, the following tools were initially used: In the backend PHP in the front-end HTML, CSS, JavaScript were used. And the MySQL database. Surveys were carried out on websites and bibliographies, the survey of requirements and business rules were made based on a manual survey with questions to the fast-food goers, and methods taught by the faculty for the construction of diagrams, models and the like were also used.

**Keywords:** *Customer, equity, fair price, remove ingredients, snackbar.*

**1 Introdução**

As redes de fast-food e lanchonetes, atualmente, apresentam algumas limitações quanto as opções do cardápio. Como é muito mostrado para se manter em um padrão, muitas redes não podem customizar seus cardápios, dessa forma as opções ficam engessadas e não atingem toda a comunidade. Já existem opções vegetarianas e até mesmo veganas.

O que esse sistema se dispõe a fazer é atingir mais as diferentes preferências nutricionais, como pessoas intolerantes a lactose, glúten e outros ingredientes, eliminando-o de seu lanche e dando um bônus de volta para o cliente, pois muitas pessoas acham injusto pagar o mesmo sem comer todos os ingredientes do lanche.

Dessa forma o cliente paga o valor total, mas ele ganha um bônus por retirar algo de seu lanche. Assim sendo, o cliente poderá sempre voltar e ficar satisfeito, e conforme ele for acumulando pontos, ele pode acrescentar outros ingredientes, ou até mesmo trocar por bônus, mas isso fica a critério da empresa que utilizar o sistema. Sabemos que cada pessoa é única, e devemos tratá-las como tal, adicionando opções para ter mais diversificação de sabores, e também pensando em cada cliente e abraçando cada tal com suas limitações e peculiaridades.

Pensando nisso criamos o sistema, com tecnologias já conceituadas e estáveis no mercado, focado a princípio web, porém o sistema é responsivo, o que ajuda para os clientes que só dispõem de um celular ou tablet. Dessa forma uma pessoa pode acessar de um notebook, computador, celular ou tablet, utilizando apenas um código, com isso espera-se trazer mais satisfação para o consumidor e mantê-lo motivado e contente comprando no estabelecimento desejado.

Para o desenvolvimento deste projeto foi usado inicialmente as seguintes ferramentas: No backend o PHP, no front-end foi utilizado HTML, CSS, JavaScript. E o banco de dados MySQL. Foram realizadas pesquisas em sites e bibliografias, o levantamento de requisitos e regras de negócios foram feitos com base em uma pesquisa manual com perguntas aos frequentadores de fast-foods, foi utilizado também métodos ensinados pela faculdade para a construção de diagramas, modelagens e afins.

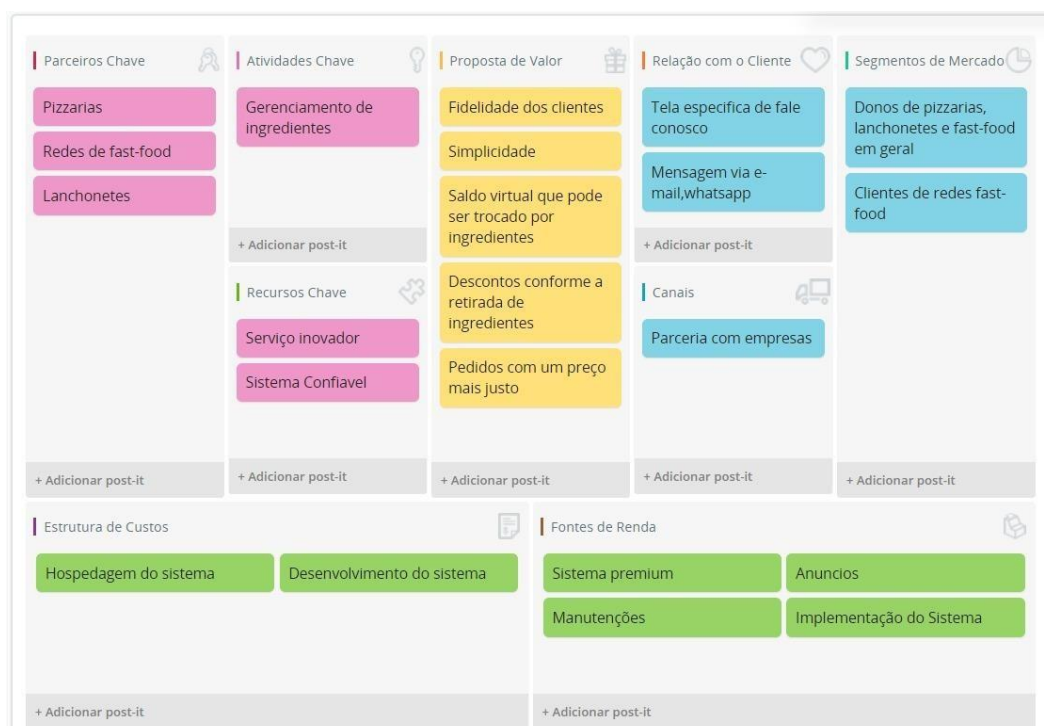
## **2 Viabilidade do projeto**

O Canvas é um esquema visual que possibilita as pessoas criarem e visualizem melhor seus esquemas de negócio com base em nove pilares criados pelo Suíço Alexander Osterwalder (PAIM, 2013).

Como é mostrado a seguir na figura 1 foi criado o canvas baseado no sistema desenvolvido, os parceiros chave foram escolhidos com base na área de mercado

em que pretendemos atuar. Nas atividades chaves é proposto o que pretende ser feito, recursos chaves é a parte de diferenciais, a proposta de valor é aquilo que é oferecido, para manter contanto com os clientes se tem as relações com o cliente, os canais de divulgação serão os próprios estabelecimentos de fast-foods, a segmentação de mercado abrange não somente os clientes de fast-foods como também os donos desses estabelecimentos, a estrutura de custo e renda são baseados em fatos hipotéticos, que podem vir a acontecer com a continuação deste sistema.

**Figura 1: Canvas**



**Fonte:** Autor

A imagem acima e o canvas do projeto onde mostramos nossos parceiros chaves que são as engrenagens que o sistema tem como base, a atividade chave que é o que o sistemas se propoe a fazer, a proposta de valor que são os diferenciais e as vantagens do sistemas juntamente com as demais informações .

### 3 Levantamento de Requisitos

#### 3.1 Elicitação de Requisitos

Neste processo, foi utilizado a escolha da etnografia (técnica de observação), visitando e frequentando vários estabelecimentos alimentícios, desde as grandes redes de fast-food à médios e pequenos pontos de lanchonetes, foi percebido uma limitação ao sistema dessas entidades quanto ao seu cardápio configurado de forma linear. Na forma padrão de estabelecimentos, caso o cliente opte por retirar alguns ingredientes, ele pagará o mesmo preço que um lanche/Hambúrguer com todos os ingredientes fazendo com que não seja justo.

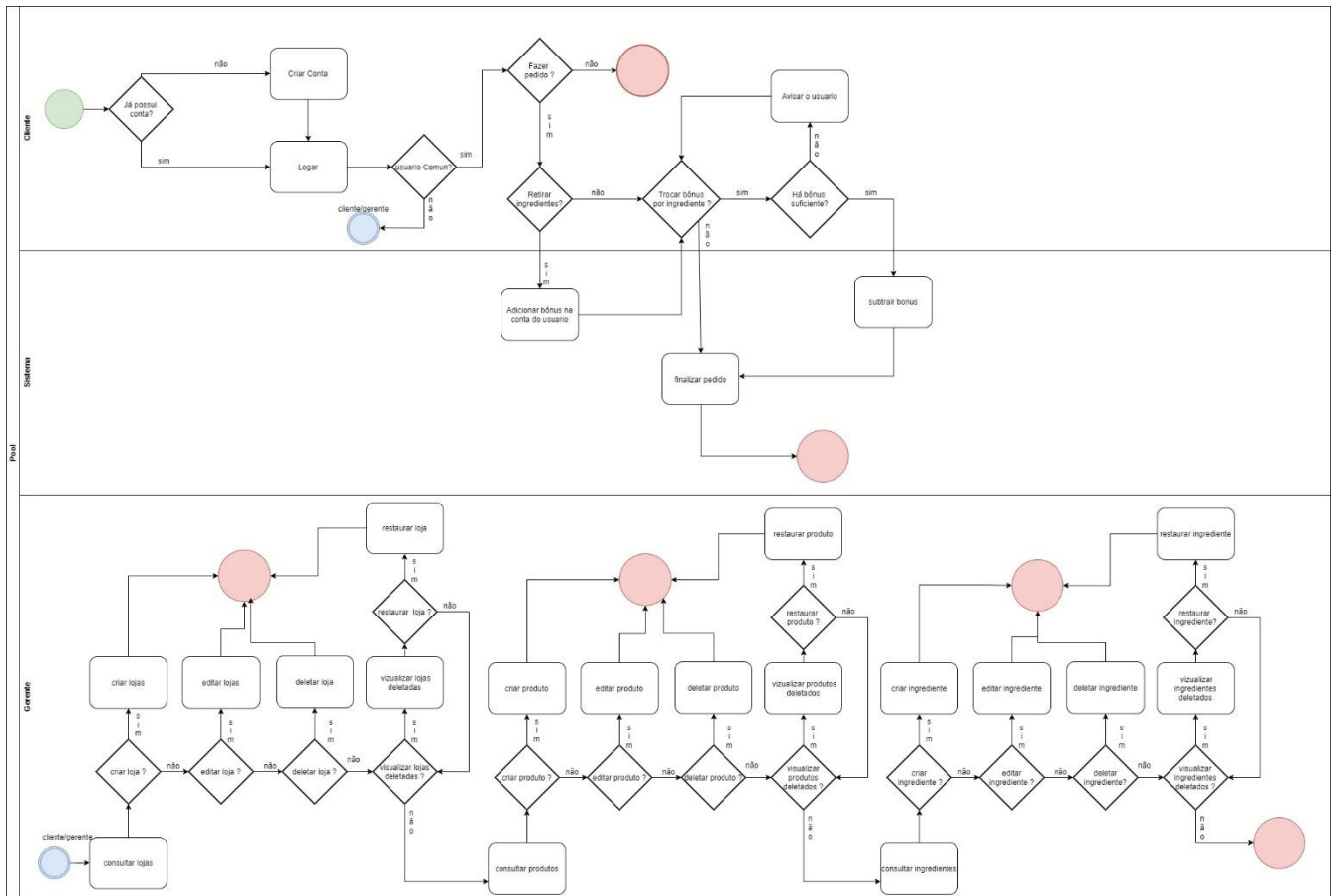
Logo após observar esse funcionamento, foi pensado algumas sugestões para alterar e fazer com que a entidade seja mais flexível com o consumidor. A conclusão foi criar um mecanismo que faça com que os clientes não percam dinheiro e o preço do lanche não seja diminuído, criando assim um formato de pontos, ganhos por cada ingrediente removido, o cliente juntará esses pontos e poderá trocar por outros ingredientes em compras futuras.

### 3.2 BPMN- Business Process Model and Notation

BPMN é uma representação gráfica onde são utilizados símbolos de modo a facilitar o entendimento dos processos ou seja é um modo mais simples de expressar os processos de forma clara através de elementos gráficos de notação (RODRIGUES, 2014).

Como é mostrado a seguir na figura 2 foi montado o BPMN do sistema utilizando o draw.io como ferramenta, a Intenção do BPMN é mostrar os processos e funcionalidades do sistema de forma gráfica afim de fazer com que as pessoas entendam o com mais facilidade.

Figura 2: BPMN



Fonte: Autor

A imagem acima mostra o BPMN do sistema que é uma forma mais simbólica de demonstrar as funcionalidades que estão sendo propostas

### 3.3 Requisitos Funcionais

Segundo Cysneiros (2001, p.22), “Os requisitos funcionais são requisitos que expressam funções ou serviços que um software deve ou pode ser capaz de executar ou fornecer”. Ou seja nos requisitos funcionais é mostrado as funcionalidades exigidas do sistema. Abaixo estão todos os requisitos funcionais do sistema, com suas numerações, títulos, categoria, Prioridade e Descrições

Quadro 1 – Requisitos Funcionais do sistema

<p><b>RF 001 – Administrar lojas</b></p>	<p>Categoria:                  ( ) Oculto                  (X) Evidente</p>	<p>Prioridade:                  (X) Altíssima                  ( ) Alta                  ( ) Média                  ( ) Baixa</p>
<p><b>Descrição:</b> O sistema deve permitir que o usuário crie, edite, delete e recupere lojas</p>		



<b>RF 002 – Administrar produto</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição</b> O sistema deve permitir que o usuário crie, edite, delete e recupere produtos.		
<b>RF 003 – Administrar ingredientes</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário crie, edite, delete e recupere ingredientes.		
<b>RF 004 – Cadastrar cliente</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir o cadastro do cliente, com nome, e-mail e senha		
<b>RF 005 – Visualizar Produtos</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário visualize os produtos de cada loja		
<b>RF 006 – Visualizar histórico de pedidos</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário possa visualizar todo o seu histórico de pedidos, deixando claro para comprovar caso ocorra algum erro.		
<b>RF 007 – Visualizar Bônus</b>	Categoria: <input type="checkbox"/> Oculto <input checked="" type="checkbox"/> Evidente	Prioridade: <input checked="" type="checkbox"/> Altíssima <input type="checkbox"/> Alta <input type="checkbox"/> Média <input type="checkbox"/> Baixa
<b>Descrição:</b> O sistema deve permitir que o usuário visualize os bônus adquiridos em compras de produtos.		

### 3.4 Requisitos Não Funcionais

Segundo Cysneiros (2001, p.22) “Os requisitos não funcionais são requisitos que declaram restrições, ou atributos de qualidade para um software e/ou para o processo de desenvolvimento deste sistema. Segurança, precisão, usabilidade, performance e manutenibilidade são exemplos de requisitos não funcionais”. Ou seja nos requisitos não funcionais é onde se exhibe as restrições e exigências do sistema. Abaixo estão todos os requisitos não funcionais do sistema, com suas numerações, títulos, Descrições, Tipo, Categoria, e sua Durabilidade dentro do sistema

**Quadro 2** – Requisitos Não Funcionais do sistema

<b>RNF001-</b> Acesso	O sistema deverá verificar se o e-mail já existe	Tipo Segurança	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório
<b>RNF002-</b> Plataforma	O sistema deverá ser acessado pela web através de computadores e celulares.	Tipo: Desempenho	( ) Desejável (X) Obrigatório	(X) Permanente ( ) Transitório

**Fonte:** Autor

Matrizes de Rastreabilidade entre Requisitos Funcionais e Requisitos Não Funcionais:

	RF001	RF002	RF003	RF004	RF005	RF006	RF007
RN001				X			
RN002	X	X	X	X	X	X	X

### 3.5 Regras de Negócio

**Quadro 3** – Regras de Negócio do sistema.

<b>RN 001 – Quem utiliza</b>
Apenas lanchonetes/serviços Fast-food administra o sistema, e usuários comuns podem usá-lo
<b>RN 002 – Faixa etária permitida</b>
Pessoas menores de idade podem se cadastrar
<b>RN 003 – Consulta do sistema</b>
apenas usuários com cadastro podem utilizar as funcionalidades do sistema.

<b>RN 004 – Transferência de informação</b>
Usuário deve possuir uma conexão de internet para poder consultar todos os dados do sistema.

### Matrizes de Rastreabilidade entre Requisitos Funcionais e Regras de Negócio

	RF 001	RF 002	RF 003	RF 004	RF 005	RF 006	RF 007
RN 001	X	X	X	X			
RN 002				X	X	X	X
RN 003	X	X	X		X	X	X
RN 004	X	X	X	X	X	X	X

### 3.6 Casos de Uso

O Casos de Uso auxilia no levantamento dos requisitos funcionais do sistema descrevendo suas funcionalidades (VIEIRA, 2015).

A seguir é exibido os índices e atores de todos os casos de uso do sistema com seus Nome, Descrições, Atores primários, Pré-condição (se existir), Cenário principal, Pós-condição (se existir).

#### Índice de casos de uso:

- UC 001: Cadastrar usuário
- UC 002: Cadastrar loja
- UC 003: Cadastrar produto

- UC 004: Cadastrar ingrediente
- UC 005: Visualizar saldo
- UC 006: Visualizar lojas
- UC 007: Visualizar produtos
- UC 008: Visualizar ingredientes
- UC 009: Finalizar pedido
- UC 010: Visualizar histórico de pedidos

#### Indicação dos atores do sistema:

- Cliente: é o usuário que representa consumidor final
- Dono: é o ator que administra as lojas
- Sistema: é o ator responsável pelas funcionalidades do sistema

**Quadro 4 – Use Case Cadastrar Usuários**

<b>Caso de Uso –: Cadastrar usuário</b>	
<b>ID</b>	UC 001
<b>Descrição</b>	Este caso de uso tem por objetivo cadastrar produto
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Estar cadastrado
<b>Cenário Principal</b>	1. O use case inicia quando o usuário seleciona a opção cadastra-se 2. O sistema carrega o formulário de cadastro. 3. O sistema informa se houve sucesso na criação..
<b>Pós-condição</b>	Para ativar o login
<b>Cenário Alternativo</b>	2a – O usuário informa um e-mail que já existe no sistema 3a.1 O sistema retorna uma mensagem informando que o e-mail já existe.
<b>Caso de Uso – Cadastrar loja</b>	
<b>ID</b>	UC 002
<b>Descrição</b>	Este caso de uso tem por objetivo cadastrar loja e adiministra-las ou seja criar editar deletar e restaurar de modo que melhor atenda as necessidades do usuario
<b>Ator Primário</b>	Dono
<b>Pré-condição</b>	Estar cadastrado como administrado
<b>Cenário Principal</b>	1. O use case inicia quando o usuário seleciona alguma opção de administrar uma loja. 2. O sistema carrega o devido formulário. 3. O sistema informa se houve sucesso mudança.
<b>Pós-condição</b>	Para ativar produtos
<b>Caso de Uso – Cadastrar produtos</b>	
<b>ID</b>	UC 003
<b>Descrição</b>	Este caso de uso tem por objetivo cadastrar produtos e adiministra-los ou seja criar editar deletar e restaurar de modo que melhor atenda as necessidades do usuario
<b>Ator Primário</b>	Dono
<b>Pré-condição</b>	Ter uma loja criada

<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona alguma opção de administrar um produto.</li> <li>2. O sistema carrega o devido formulário.</li> <li>3. O sistema informa se houve sucesso mudança.</li> </ol>
<b>Pós-condição</b>	Para ativar ingredientes
<b>Caso de Uso – Cadastrar ingrediente</b>	
<b>ID</b>	UC 004
<b>Descrição</b>	Este caso de uso tem por objetivo cadastrar ingredientes e adiministra-los ou seja criar editar deletar e restaurar de modo que melhor atenda as necessidades do usuario
<b>Ator Primário</b>	Dono
<b>Pré-condição</b>	Ter um produto criado
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona alguma opção de administrar um ingrediente.</li> <li>2. O sistema carrega o devido formulário.</li> <li>3. O sistema informa se houve sucesso mudança.</li> </ol>
<b>Pós-condição</b>	
<b>Caso de Uso –Visualizar saldo</b>	
<b>ID</b>	UC 005
<b>Descrição</b>	Este caso de uso tem por objetivo visualizar o saldo
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Estar logado
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona a opção de logar.</li> <li>2. O sistema carrega as informações de saldo.</li> <li>3. O sistema mostra exibe o saldo.</li> </ol>
<b>Caso de Uso –Visualizar lojas</b>	
<b>ID</b>	UC 006
<b>Descrição</b>	Este caso de uso tem por objetivo visualizar as lojas
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Estar logado
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona a opção de logar.</li> <li>2. O sistema mostra as informações das lojas.</li> </ol>
<b>Pós-condição</b>	Visualizar produtos
<b>Caso de Uso –Visualizar produtos</b>	
<b>ID</b>	UC 007

<b>Descrição</b>	Este caso de uso tem por objetivo visualizar os produtos
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Selecionar uma loja
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona a opção de ver produtos de uma loja.</li> <li>2. O sistema exibe todos os produtos daquela loja</li> </ol>

<b>Pós-condição</b>	Visualizar ingredientes
<b>Caso de Uso –Visualizar ingredientes</b>	
<b>ID</b>	UC 008
<b>Descrição</b>	Este caso de uso tem por objetivo visualizar os ingredientes
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Selecionar um produto
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona a opção de ver os ingredientes de um produto.</li> <li>2. O sistema exibe todos os ingredientes menos o pão.</li> <li>3. O sistema permite que o usuário altere ou não as quantidades dos ingredientes.</li> </ol>
<b>Pós-condição</b>	Finalizar pedido
<b>Caso de Uso –Finalizar pedido</b>	
<b>ID</b>	UC 009
<b>Descrição</b>	Este caso de uso tem por objetivo finalizar um pedido
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Selecionar Finalizar pedido
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>1. O use case inicia quando o usuário seleciona a opção de finalizar pedido.</li> <li>2. O sistema carrega as informações sobre quanto que deu o pedido e caso tenha dado mais que o padrão ele pergunta para o usuário se ele deseja utilizar o bônus.</li> <li>3. O sistema adiciona o pedido ao histórico do cliente e atualiza seu saldo.</li> </ol>
<b>Cenário Alternativo</b>	<ol style="list-style-type: none"> <li>2a – O usuário seleciona usar saldo mais não tem saldo suficiente</li> <li>3a.1 O sistema retorna uma mensagem informando o saldo não é suficiente.</li> </ol>
<b>Caso de Uso –Visualizar histórico de pedidos</b>	

<b>ID</b>	UC 010
<b>Descrição</b>	Este caso de uso tem por objetivo visualizar o histórico de pedidos
<b>Ator Primário</b>	Cliente
<b>Pré-condição</b>	Estar logado
<b>Cenário Principal</b>	<ol style="list-style-type: none"> <li>O use case inicia quando o usuário seleciona a opção de visualizar histórico de pedidos.</li> <li>O sistema exibe as informações sobre todas as compras do usuário.</li> </ol>

**Fonte:** Autor

### 3.7 Diagrama de Caso de Uso

Para fazer o Diagrama de Caso de Uso foi usado o site da Lucidchart e segundo eles “O objetivo do diagrama de caso de uso em UML é demonstrar as diferentes maneiras que o usuário pode interagir com um sistema” (Lucidchart 2020). Ou seja são usados desenhos bem simplificados para facilitar a explicação e o entendimento do projeto. Como podemos ver na imagem abaixo o diagrama de caso de uso do nosso sistema.

Figura 3: Diagrama de caso de uso



**Fonte:** Autor

A imagen a cima mostra o Diagrama de caso de uso que tem como objetivo mostrar as diferentes maneiras do usuario interagir com o sistemas, de um lado vemos as opção dos clientes e do outro as do gerente/dono do estabelecimento.

### 3.8 Diagrama de Classes

Segundo o site Lucidchart:

“O diagrama de classes foi criado como um modelo padronizado para descrever uma abordagem de programação orientada ao objeto. Como as classes são os componentes básicos dos objetos, diagramas de classes são os componentes básicos do diagrama. Os diversos componentes em um diagrama de classes podem representar as classes que serão realmente programadas, os principais objetos ou as interações entre classes e objetos.

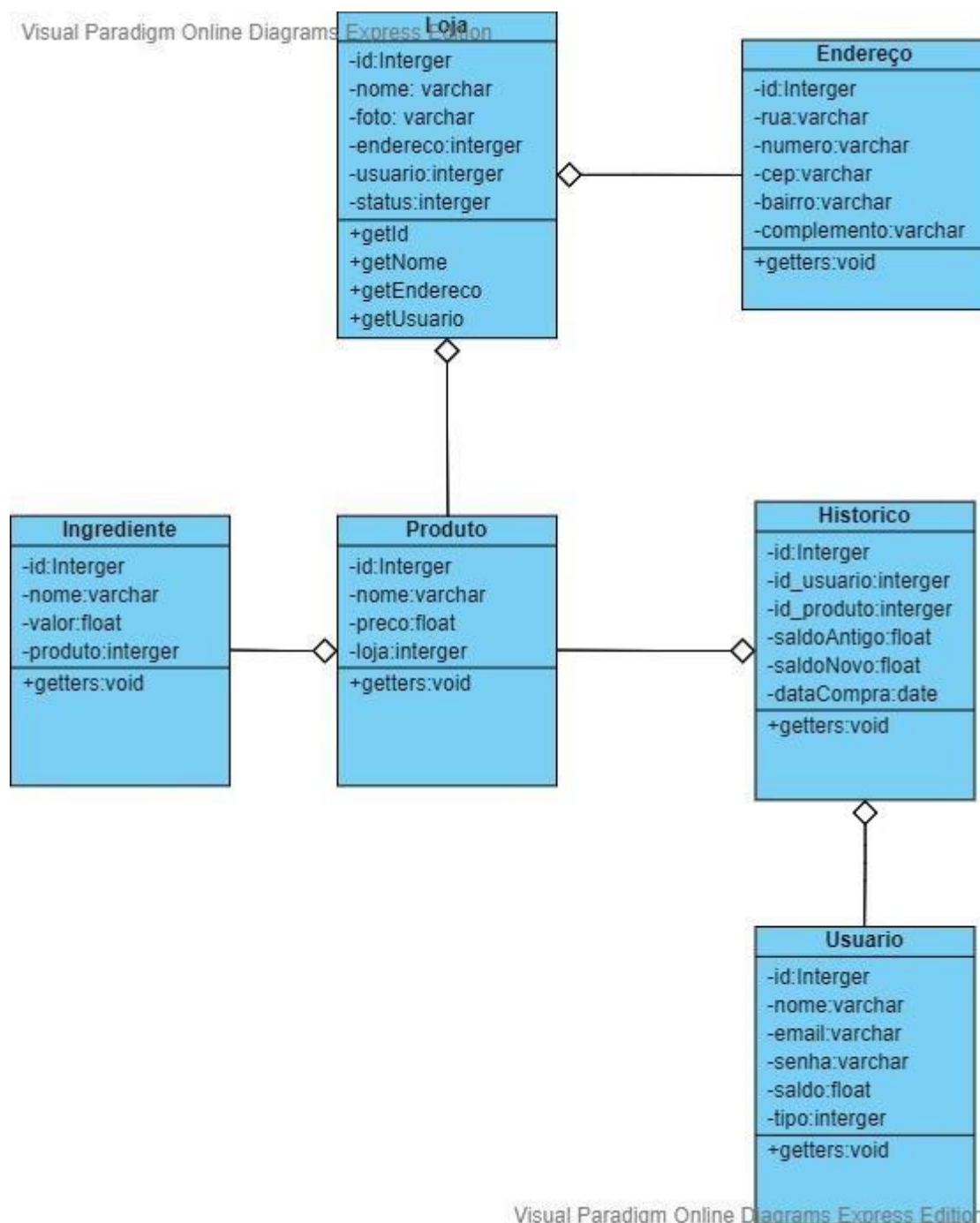
A forma de classe em si consiste em um retângulo com três linhas. A linha superior contém o nome da classe, a linha do meio, os atributos da classe e a linha inferior expressa os métodos ou operações que a classe pode utilizar. Classes e subclasses são agrupadas juntas para mostrar a relação estática entre cada objeto”. (Lucidchart 2020).

Nesse processo foi utilizado o site: Visual Paradigm Online Diagrams. É comum nos diagramas de classe ter setas ligando um filho em seu pai, isso chamamos de herança. No caso não foi utilizado herança em nosso software.

Na figura abaixo podemos ver a representação do diagrama de classe.



Figura 4:Diagrama de Classes



**Fonte:** Autor

Diagrama de classes é uma representação estática utilizada na área da programação para descrever a estrutura de um sistema, apresentando suas classes, atributos, operações e as relações entre os objetos.

Uma classe descreve como certos tipos de objetos se parecem do ponto de vista da programação, pois quando definimos uma classe precisamos definir duas coisas:

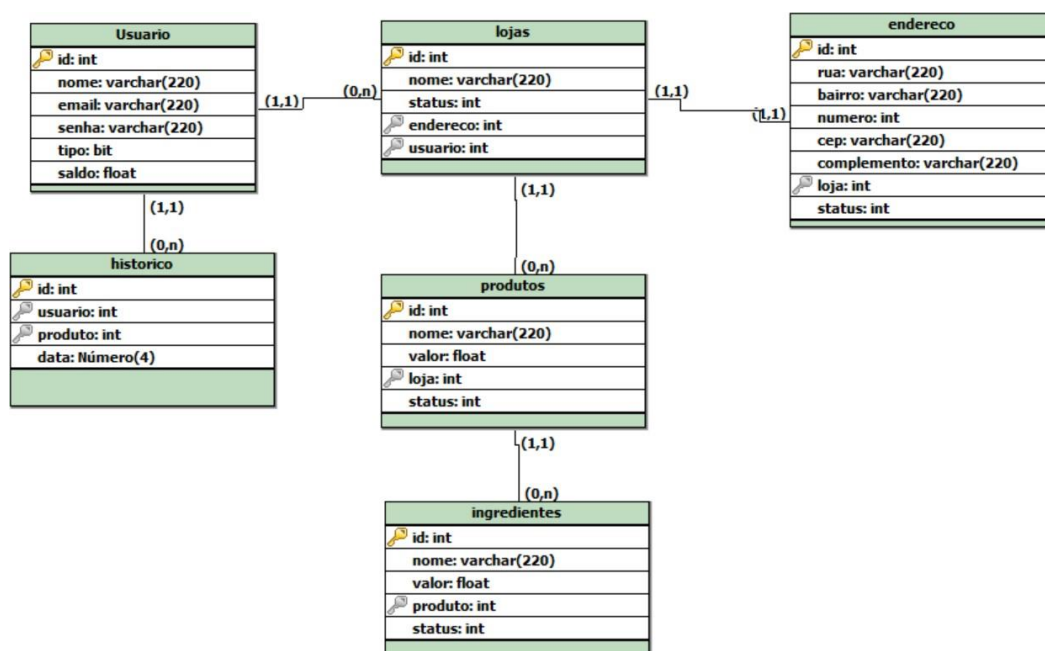
Propriedades - Informações específicas relacionadas a uma classe de objeto. São as características dos objetos que as classes representam. Ex Cor, altura, tamanho, largura, etc...

Métodos: São ações que os objetos de uma classe podem realizar. Ex: Latir , correr , sentar , comer, etc.

### 3.9 Diagrama Entidade-Relacionamento

O Diagrama de Entidade e relacionamento são normalmente usados para facilitar a comunicação entre os integrantes da equipe (JOEL, 2014). Ou seja, são as bases a serem seguidas por aqueles que vão fazer desenvolvimento da base de dados. A figura 5 abaixo temos a representação do diagrama entidade-relacionamento do xsistema.

Figura 5:Diagrama de Entidade Relacionamento



Fonte: Autor

A imagem acima mostra o diagrama de entidade relacionamento que tem por objetivo exemplificar como ficou o banco de dados

## 4 Ferramentas e Métodos ou Desenvolvimento

### 4.1 Ferramentas

Segundo CISNEIRO (2009).

” uma das melhores habilidades do PHP é lidar com bancos de dados de uma forma fácil. Hoje em dia os sistemas para Web estão cada vez mais usufruindo das capacidades magníficas que os bancos de dados podem oferecer.

O MySQL é um banco de dados simples, rápido e eficiente para se trabalhar com sistemas Web.”

Scudero (2017, p 1). tem uma definição bem sucinta das tecnologias HTML, CSS e Javascript como podemos ver a seguir. “O HTML é o responsável pelo esqueleto dos sites, o CSS pela aparência e o JavaScript pelas habilidades adicionais (comportamento).”

Segundo Magno (2018, p 1).,

” O Bootstrap te ajuda a ter o ponto de partida, como o próprio nome diz, a partir dele, você sobrescreve estilos e faz diferentes combinações para dar a seu sistema um estilo único. Se sua necessidade for usar como protótipo, você já tem um kit para construir seu front-end pronto para uso.”

## 4.2 Métodos ou Desenvolvimento

Para o desenvolvimento deste projeto foi usado inicialmente PHP como base juntamente com HTML, CSS, JavaScript e MySQL, primeiramente foi criado um repositório no GitHub logo em seguida foram montados vários protótipos de telas que se aprimoraram cada vez mais, com uma noção já estabelecida foi montado a modelagem do banco de dados mysql e em seguida a criação do mesmo, ao longo do tempo foi determinado que seria usado o framework Bootstrap para a facilitação na montagem das telas, e que as senhas dos usuários seriam criptografadas antes de serem salvas no banco de dados assim gerando uma segurança melhor para os usuários.

Abaixo na figura 6, vemos uma parte do código onde é implementado a criptografia hash na senha do usuário antes de salvá-la na base de dados afim de proteger os dados dos usuários do sistema usar criptografia e uma das considerações mais básicas de segurança que deve ser feita afinal segurança nunca pouca

Figura 6:Código fonte

```
// as variáveis nome, email e senha recebem os dados digitados na página anterior
$nome = filter_input(INPUT_POST, 'nome', FILTER_SANITIZE_EMAIL);

$email = filter_input(INPUT_POST, 'email', FILTER_SANITIZE_EMAIL);

$senha = filter_input(INPUT_POST, 'senha', FILTER_SANITIZE_STRING);
// senha e criptografadas
$senha = password_hash($senha,PASSWORD_DEFAULT );
```

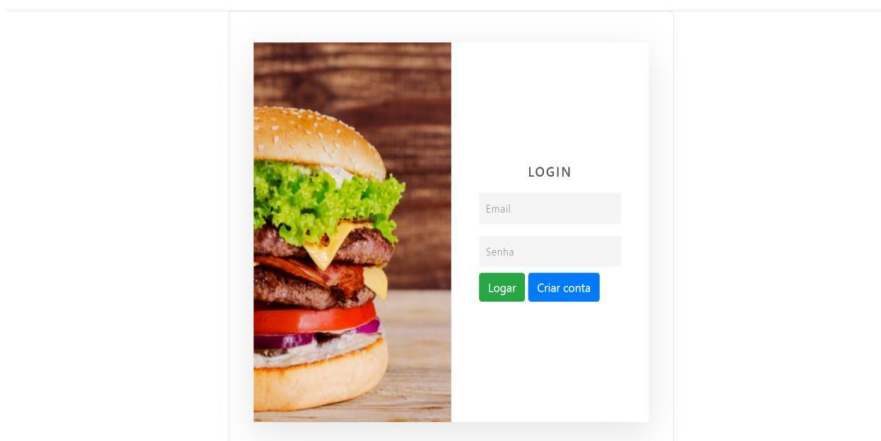
**Fonte:** Autor

A imagem acima mostra um trecho do código fonte do sistema

## 5 Resultados e Discussão

Ao entrar no sistema o usuário se depara com a figura 7.

Figura 7:Login

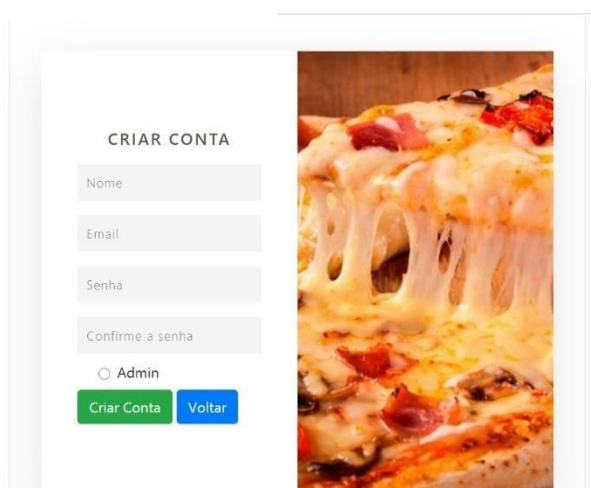


**Fonte:** Autor

A imagem acima apresenta a tela de login onde foi usada uma imagem chamativa da área de fast-food, caso o usuário não tenha conta ainda e possível criá-la através do botão de criar conta, caso o usuário já possua uma conta cadastrada basta ele informar os dados e entrar no sistema

Caso o usuário não tenha uma conta ele se depara com a figura 8.

Figura 8:Cadastro

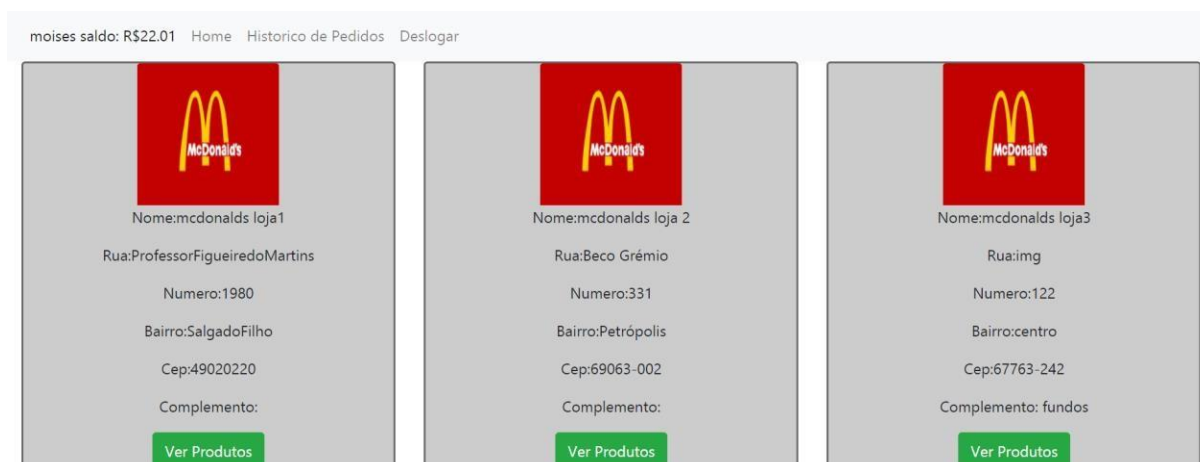


**Fonte:** Autor

A imagem acima mostra que o usuário pode criar uma conta entrando na tela de cadastro, onde informando alguns de seus dados como nome, e-mail, senha e se ele é um administrador de rede fast-food ele já se torna apto a usar o sistema.

A tela seguinte do login para os usuários comuns é a da figura 9.

Figura 9:Home



**Fonte:** Autor

A imagem acima mostra a tela Home onde são exibidas todas as lojas já cadastradas e ativas até o momento, o usuário pode também visualizar em sua tela algumas informações pessoais como nome e seu saldo atual assim como um menu navegável na parte superior da página

Logo após selecionar uma loja o usuário se depara com a figura 10

Figura 10:Produtos



**Fonte:** Autor

A imagem acima mostra o cardápio de produtos dela com informações básicas como nome do produto e seu preço juntamente com um botão para se ter mais informações sobre o produto como seus ingredientes e as quantidades e preços deles.

Após a escolha do produto o usuário visualiza a figura 11

Figura 11:Ingredientes

#	Ingrediente	Valor	Alterar Quantidade
1	alface	R\$ 0.3	1 <input type="text"/>
2	milho	R\$ 0.1	0 <input type="text"/>
3	bacon	R\$ 0.5	2 <input type="text"/>
4	queijo	R\$ 0.5	1 <input type="text"/>
5	salsicha	R\$ 0.99	1 <input type="text"/>

Visualizar Pedido

**Fonte:** Autor

A imagem acima mostra a tela de ingrediente com as informações do mesmos tais quais como Nome Valor e Quantidade, a quantidade dos ingredientes pode ser mudada de acordo com a vontade do cliente todavia o preço do produto pode vir a mudar de acordo com as quantidades dos ingredientes desejada.

A tela de finalizar pedido esta representado na figura 12

Figura 12:Finalizar Pedido

Ingredientes	Quantidade
alface	1
milho	0
bacon	2
queijo	1
salsicha	1

valor total: 20.4  
Usar Bonus ?

Finalizar Pedido

**Fonte:** Autor

A imagem acima mostra a tela de finalizar pedido onde é exibido os ingredientes do produto assim como suas quantidades atuais que podem ou não terem sido alteradas na tela anterior juntamente com o novo preço do produto caso tenha ocorrido alguma mudança nas quantidades dos ingredientes, caso o preço esteja abaixo do indicado originalmente na tela 4 o usuário ganhara o valor da subtração do preço original com o valor do preço atual, caso o novo valor seja maior que o original o sistema perguntara para o usuário se ele deseja utilizar o valor de seu bônus atual para pagar esse diferença se selecionado a opção de usar bônus o sistema verifica se o saldo atual do usuário e o bastante para pagar pelos

ingredientes extras se for ele subtrair esse valor excedente caso contrário ele informa para o usuário que seu saldo não é suficiente.

Na figura 13 temos o historico

Figura 13:Histórico de pedidos

#	Produto	Saldo Anterior	Novo Saldo	Data Compra
1	x-tudo	0	0.5	2020-10-17 00:14:28
2	x-tudo	0.5	0.6	2020-10-17 00:15:03
3	pizza	0.6	0.1	2020-10-17 00:15:10
4	pizza-tudo	0.1	22.1	2020-10-17 00:15:33

**Fonte:** Autor

A imagem acima mostra a tela de Histórico de Pedidos o usuário pode ver algumas informações sobre seus pedidos até o momento como o nome do produto comprado e saldo anterior e posterior a compra juntamente com a data da compra

Se o usuário for um administrador de redes fast-foods em sua tela inicial a da figura 14

Figura 14:Home admin

#	Nome	Foto	Endereco	Produtos	Deletar	Editar
1	bk	16029048375f8a6305dd2ae.jpg	Bairro :SalgadoFilho Rua :ProfessorFigueiredoMartins Numero :1980 Complemento : Cep :49020220	Visualizar Produtos	Deletar	Editar
2	mec	16028938255f8a3801ea350.jpg	Bairro :img Rua :img Numero :1 Complemento : Cep :1	Visualizar Produtos	Deletar	Editar

Criar Lojas

Ver Lojas Deletadas

**Fonte:** Autor

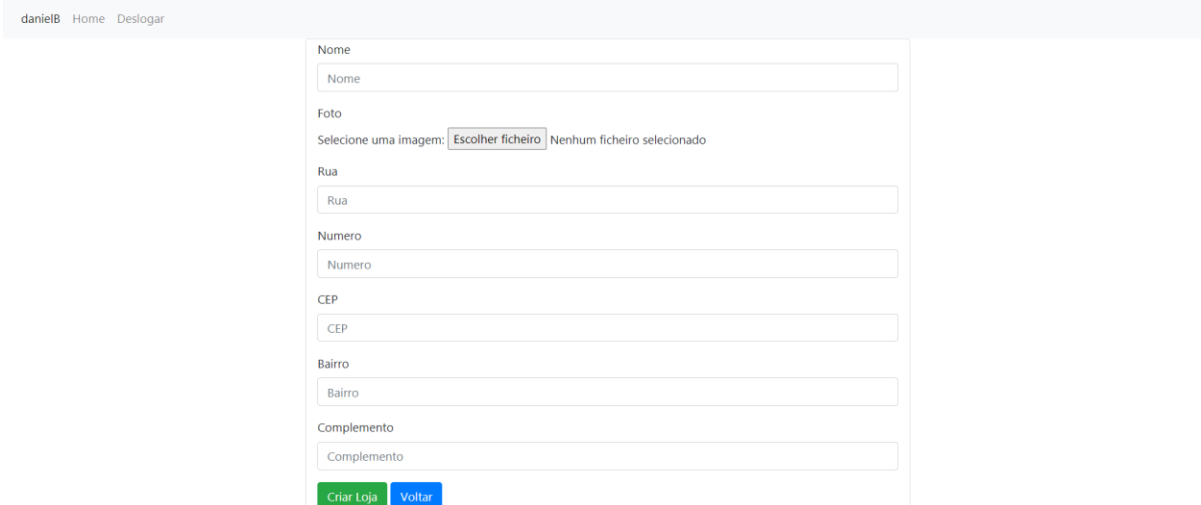
A imagem acima mostra para o administrador que será exibida suas atuais lojas assim como algumas opções como adicionar uma nova loja, editar ou deletar uma loja já existente, visualizar os produtos desta loja, ou visualizar as suas lojas



deletadas até o momento.

Tanto para criar quanto para editar uma loja as informações necessárias são as mesmas da figura15

Figura 15: Criar/Editar lojas



The image shows a web form for creating or editing a store. At the top left, there is a navigation bar with the text "danielB Home Deslogar". The form itself is a white box with a light gray border. It contains the following fields and elements:

- Nome:** A text input field with the placeholder text "Nome".
- Foto:** A section with the text "Selecione uma imagem:" followed by a button labeled "Escolher ficheiro" and the text "Nenhum ficheiro selecionado".
- Rua:** A text input field with the placeholder text "Rua".
- Numero:** A text input field with the placeholder text "Numero".
- CEP:** A text input field with the placeholder text "CEP".
- Bairro:** A text input field with the placeholder text "Bairro".
- Complemento:** A text input field with the placeholder text "Complemento".
- At the bottom of the form, there are two buttons: a green button labeled "Criar Loja" and a blue button labeled "Voltar".

**Fonte:** Autor

A imagem acima mostra o procedimento para criar editar e deletar ,tento apenas um preenchimento com as informações já existentes caso o administrador esteja editando uma loja, as informações aqui fornecidas serão as mesmas que os usuários comuns iram ver em suas telas informações essas que são o nome da loja uma imagem que a represente ( como uma logo ou algo do gênero ) e o endereço da loja.

As telas de restauração seguem o padrão da figura 16

Figura 16: Restauração

#	Nome	Foto	Endereço	Restaurar
1	loja5	16031472195f8e15d385285.jpg	Bairro :2 Rua :test Numero :2 Complemento : Cep :2	Restaurar
2	loja7	16031476015f8e1751ca240.jpg	Bairro :aeroporto3 Rua :ruatal Numero :123 Complemento :fundos Cep :123456789	Restaurar

Voltar

Fonte: Autor

A imagem acima mostrar todas as informações que existiam antes da deleção isso se aplica para (lojas, produtos e ingrediente). A seguir vemos na tela 10 o layout padrão de todas as telas de restauração.

A tela mostrada a seguir e da figura 17.

Figura 17: Produtos loja

#	Nome	Preço	Ingredientes	Deletar	Editar
1	xtudo	20	Visualizar Ingredientes	Deletar Produto	Editar Produto
2	xsalada	15	Visualizar Ingredientes	Deletar Produto	Editar Produto
3	X-egg	12.99	Visualizar Ingredientes	Deletar Produto	Editar Produto
4	a moda da casa	40.99	Visualizar Ingredientes	Deletar Produto	Editar Produto
5	qualquer um	39.95	Visualizar Ingredientes	Deletar Produto	Editar Produto
6	qualquer coisa	55	Visualizar Ingredientes	Deletar Produto	Editar Produto
7	x-queijo	9	Visualizar Ingredientes	Deletar Produto	Editar Produto

Adicionar Produto

Ver Produtos Deletados

Fonte: Autor

A imagem acima de visualização dos produtos e exibida conforme é mostrado na tela 11, ela contém algumas funcionalidades tais quais existem também na tela 8 como criação deleção edição e restauração entretanto neste caso essas funcionalidades se aplicam aos produtos e temos a opção de visualizar os ingredientes destes produtos mostrado a seguir na tela

Tanto para criar quanto para editar um podemos visualizar a figura 18

Figura 18: Criar/Editar Produtos

**Fonte:** Autor

A imagem acima mostra o processo de criação e edição do produto as informações necessárias são as mesmas tanto apenas um preenchimento com as informações já existentes caso o administrador esteja editando um produto, as informações aqui fornecidas serão as mesmas que os usuários comuns iram ver em suas telas informações essas que são o nome do produto e seu preço que deve ser informado usando o “.” como separador da casa decimal.

A tela de visualização dos ingredientes e exibida conforme mostrado na figura 19

Figura 19: Ingredientes loja

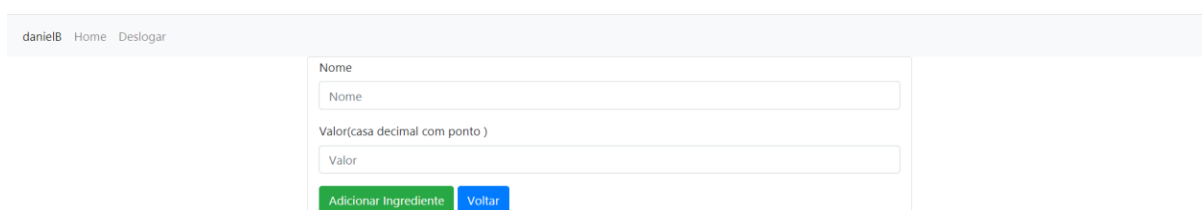
#	Nome	Preço	Deletar	Editar
1	alfece	0.3	Deletar Ingrediente	Editar Ingrediente
2	milho	0.1	Deletar Ingrediente	Editar Ingrediente
3	bacon	0.5	Deletar Ingrediente	Editar Ingrediente
4	queijo	0.5	Deletar Ingrediente	Editar Ingrediente
5	salsicha	0.99	Deletar Ingrediente	Editar Ingrediente

**Fonte:** Autor

A imagem acima mostra algumas informações que serão visualizadas mais tarde pelos usuários comuns informações essas que são o nome e preço de cada ingrediente o administrador pode também adicionar ou remover tanto quantos ingredientes lhe for necessário para representar seu produto

Tanto para criar quanto para editar um ingrediente podemos visualizar na figura

Figura 20: Criar/Editar Ingredientes



The screenshot shows a web interface for managing ingredients. At the top, there is a navigation bar with the text 'danielB Home Deslogar'. Below this, a form is displayed with two input fields. The first field is labeled 'Nome' and the second is labeled 'Valor(casa decimal com ponto )'. At the bottom of the form, there are two buttons: a green button labeled 'Adicionar Ingrediente' and a blue button labeled 'Voltar'.

**Fonte:** Autor

A imagem acima mostra as informações necessárias para criar e editar ingredientes que são as mesmas tanto apenas um preenchimento com as informações já existentes caso o administrador esteja editando um ingrediente, as informações aqui fornecidas serão as mesmas que os usuários comuns irão ver em suas telas informações essas que são o nome do ingrediente e seu preço que deve ser informado usando o “.” como separador da casa decimal.

Nas duas telas a seguir temos um print do css e do html, o css consiste na estilização do site, e o html seria o esqueleto. As tecnologias mostradas na tela abaixo são: Html e css para o front-end e Php para o back-end.

Podemos ver uma parte do css, nesse caso ele está mexendo dentro do h2 que está dentro do form, no caso ele está modificando a palavra login, deixando-a em letras maiúsculas com o uppercase, levemente em negrito com o font-weight, alinhando ao centro e cor da fonte cinza escuro

Figura 21:Css

```
69 section .container .user form h2{
70     font-size: 18px;
71     font-weight: 600;
72     text-transform: uppercase;
73     letter-spacing: 2px;
74     text-align: center;
75     width: 100%;
76     margin-bottom: 10px;
77     color: #555;
78 }
```

**Fonte:** Autor

A imagem acima mostra uma parte do código fonte

Figura 22:Html

```

<div class="container">
  <div class="user login">
    <div class="imgBX"> </div>
    <div class="formBX">
      <form action='./include/processa_login.php' method="POST">
        <h2> Login</h2>

```

**Fonte:** Autor

A imagem acima mostra uma parte do código fonte

Como podemos visualizar na figura 23 abaixo foi utilizado PHP como backend e para o usuário poder se cadastrar verificamos não somente as informações por ele digitadas mais também se em nossa base de dados o e-mail já não está sendo usado. caso contrário o adicionamos

Figura 23:PHP

```

14 // as próximas linhas são responsáveis por fazer a conexão com o bando de dados.
15
16 include_once("../include/conn.php");
17 $consulta = ("SELECT * FROM usuarios WHERE email = '$email' ");
18
19 $result = mysqli_query($conn,$consulta);
20 if(mysqli_num_rows($result) > 0 ){
21   $_SESSION['msg'] = "<p class='centro' style='color:red'> Email já cadastrado </p>";
22   header('location:./add_usuario.php');
23 }
24 else{
25   $senha = password_hash($senha,PASSWORD_DEFAULT );
26   $sql = "INSERT INTO usuarios (nome, email, senha,saldo) VALUES ('$nome', '$email', '$senha','0')";
27   if (mysqli_query($conn, $sql)) {
28     $_SESSION['msg'] = "<p class='centro' style='color:green'> Conta Cadastrada Com Sucesso</p>";
29     header('location:./add_usuario.php');
30   }
31   else {
32     echo "Error: " . $sql . "<br>" . mysqli_error($conn);
33   }
34 }
35 }
36 }
37 ?>

```

**Fonte:** Autor

A imagem acima mostra uma parte do código fonte

## Considerações finais

O objetivo inicial e principal desse projeto era deixar um pedido mais justo ao consumidor, sem fazer a empresa ter grandes prejuízos e o consumidor final ainda ter algum benefício. Como podemos ver, a medida que fomos fazendo o projeto, vimos que atualmente as pessoas estão mais preocupadas com suas saúdes, dessa forma retirar um ingrediente que ela não goste, tenha alguma intolerância, ou simplesmente não o ache saudável, ficou mais fácil e benéfico para todos, pois assim, a loja ainda terá seu lucro e conseguirá manter uma cartela ainda maior e diversificada de clientes.

Conforme o sistema estava sendo feito, vimos que a questão de gerencia de estoque (que era algo que queríamos colocar), não seria viável para esse MVP (*“Minimum Viable Product”* – mínimo produto viável), ficando assim como implementação futura, a ideia inicial também era integrar o sistema, a sistemas já prontos como uma funcionalidade, isso é uma ideia interessante, porém inviável a princípio. Queremos fazer um ERP (*“Enterprise Resource Planning”*, ou sistema de gestão integrado) de comida, o aplicativo futuramente não servirá somente para pedidos de comida, terá toda uma parte administrativa, como controle de estoque, análises de feedback e gráficos com valores ganhos.

Portanto como podemos analisar, o objetivo principal do sistema foi atingido, e como foi dito no parágrafo anterior, temos futuras implementações em mente.

## Referências

CISNEIRO Hugo. **Trabalhando com PHP e MySQL: Uma Introdução**. sd. Disponível em: <<https://www.monografias.com/pt/trabalhos2/trabalhando-php-mysql/trabalhando-php-mysql.shtml#:~:text=livro%20de%20visitas-1.,o%20uso%20em%20p%C3%A1ginas%20Web.&text=Para%20quem%20n%C3%A3o%20conhece%20linguagens,que%20PHP%20%C3%A9%20muito%20f%C3%A1cil>>. Acesso em 29.out.2020.

CYSNEIROS Luiz. **Requisitos Não Funcionais: Da Elicitação ao Modelo Conceitual**. 2001. PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO. rio de janeiro. Disponível em: < <http://www-di.inf.puc-rio.br/~julio/Tese%20-%205.pdf>>. Acesso em:28.out.2020

DEMO PROJECT, **Visual Paradigm Online Diagrams**. sd. Disponível em: < <https://online.visual-paradigm.com/drive/#diagramlist:proj=0&new=ClassDiagram>>. Acesso em: 26.out.2020.

DRAW.IO, **diagrams.net**, sd. Disponível em :< <https://app.diagrams.net>>. Acesso em: 15.out.2020.

JOEL. Modelo Entidade Relacionamento (MER) e Diagrama Entidade-Relacionamento (DER) **Site DevMedia**. 2014. Disponível em: <https://www.devmedia.com.br/modelo-entidade-relacionamento-mer-e-diagrama-entidade-relacionamento-der/14332> >. Acesso em: 29.out.2020.

LUCIDCHART. Diagrama de caso de uso UML: O que é, como fazer e exemplos. **Site Lucidchart**. sd. Disponível em: < <https://online.visual-paradigm.com/app/diagrams/>>. Acesso em: 28.out.2020.

LUCIDCHART. O que é um diagrama de classe UML? **Site Lucidchart**. sd. Disponível em: < <https://www.lucidchart.com/pages/pt/o-que-e-diagrama-de-classe-uml#:~:text=A%20forma%20de%20classe%20em,rela%C3%A7%C3%A3o%20est%C3%A1tica%20entre%20cada%20objeto.>>. Acesso em: 28.out.2020.

MAGNO Alexandre. **7 motivos para (ainda) usar (e continuar) com o Bootstrap**. sd. Disponível em: <<https://alexandremagno.net/2018/05/7-motivos-para-ainda-usar-e-continuar-com-o-bootstrap/>> Acesso em 29.out.2020

PAIM, Flavio. Canvas - Um modelo de negócios inovador **Site administradores**. 2013. Disponível em: < <https://administradores.com.br/artigos/canvas-um-modelo-de-negocios-inovador>>. Acesso em: 28.out.2020.

RODRIGUES, Regina Porque usar a Notação BPMN? **Site administradores**. 2013. Disponível em: < <https://administradores.com.br/artigos/porque-usar-a-notacao-bpmn>>. Acesso em: 28.out.2020.

SCUDEIRO Erick. **5 aplicações que podem ser criadas aprendendo apenas HTML, CSS e JS**. Sd. Disponível em: < <https://becode.com.br/aplicacoes-que-podem-ser-criadas-aprendendo-html-css-e-javascript/> > Acesso em 29.out.2020

SEBRAE, Crie seu modelo de negócios, sd. Disponível em :< <https://sebraecanvas.com/#/dashboard/meus-canvas/660183>>. Acesso em: 25.out.2020.

VIEIRA, Rodrigo. UML — Diagrama de Casos de Uso **Site Operacional ti**. 2015. Disponível em: < <https://medium.com/operacionalti/uml-diagrama-de-casos-de-uso-29f4358ce4d5>>. Acesso em: 28.out.2020.