CENTRO PAULA SOUZA

FACULDADE DE TECNOLOGIA DE FRANCA

“Dr. THOMAZ NOVELINO”

**TECNOLOGIA EM DESENVOLVIMENTO SOFTWARE MULTIPLATFORMA**

João vitor luquetti silva

Leonardo henrique oliveira cunha

luis pedro dutra carrocini

maria luiza barbosa

vinicius de araujo silva

SITE: diiner’s burgers

FRANCA/SP

2024

1 Introdução

A pandemia da COVID-19 trouxe grandes desafios para o setor de alimentação, impulsionando a necessidade de adaptação e inovação no atendimento ao cliente. Nesse cenário, a criação de uma presença digital, como um site de vendas, tornou-se essencial para garantir a continuidade dos negócios e a satisfação dos clientes. Foi com essa visão que o cliente desejou a criação do site oficial de vendas para sua hamburgueria, Diiners Burgers, que levará o sabor dos seus produtos diretamente até a casa dos seus clientes.

Com a necessidade de distanciamento social e a consequente redução no atendimento presencial, tornou-se evidente a demanda por uma solução que permitisse aos seus clientes acessar o cardápio e fazer pedidos de maneira rápida, segura e conveniente. Além disso, a gestão eficiente dos pedidos e produtos também se mostrou crítica para manter a qualidade do serviço e a satisfação dos seus clientes.

O objetivo é desenvolver uma plataforma online, no caso um site, que atenda tanto às necessidades dos clientes quanto dos funcionários. Para os clientes, o site oferece uma interface amigável para explorar o cardápio, realizar pedidos e acompanhar suas compras. Para os funcionários, o sistema proporciona ferramentas de gerenciamento que facilitam o acompanhamento dos pedidos.

Este projeto é de grande importância para a hamburgueria, pois não só permite a continuidade das operações em tempos desafiadores, mas também abre novas oportunidades de crescimento ao ampliar a base de clientes para além do atendimento presencial. A digitalização do serviço aumenta a eficiência operacional, melhora a experiência do cliente e fortalece a presença da marca no mercado.

Para o desenvolvimento do site, primeiramente realizamos uma entrevista com o cliente, no caso o dono da hamburgueria, conseguindo assim as informações essenciais (requisitos) para o desenvolvimento do projeto. Ele espera que com esse sistema ele possa efetuar as suas vendas de uma forma mais rápida, prática e produtiva, fazendo com que a hamburgueria obtenha mais visibilidade e reconhecimento por qualidade, além de facilitar para os seus usuários a forma de realizar pedidos e encontrar a hamburgueria, caso for de seu desejo se alimentar no local da Diiners Burgers. Posteriormente dividimos o processo de desenvolvimento nas seguintes etapas:

**Análise de Requisitos:** Identificação das necessidades do cliente e dos funcionários através de entrevistas e monitoramento presencial.

**Design da Interface:** Criação de protótipos de baixa para garantir uma experiência de usuário intuitiva e eficiente, utilizando o Figma para isso.

**Desenvolvimento:** Implementação das funcionalidades do site utilizando tecnologias web modernas, como as linguagens SQL, JavaScript, HTML5 e CSS3, além de React e Node JS, garantindo segurança, escalabilidade e performance.

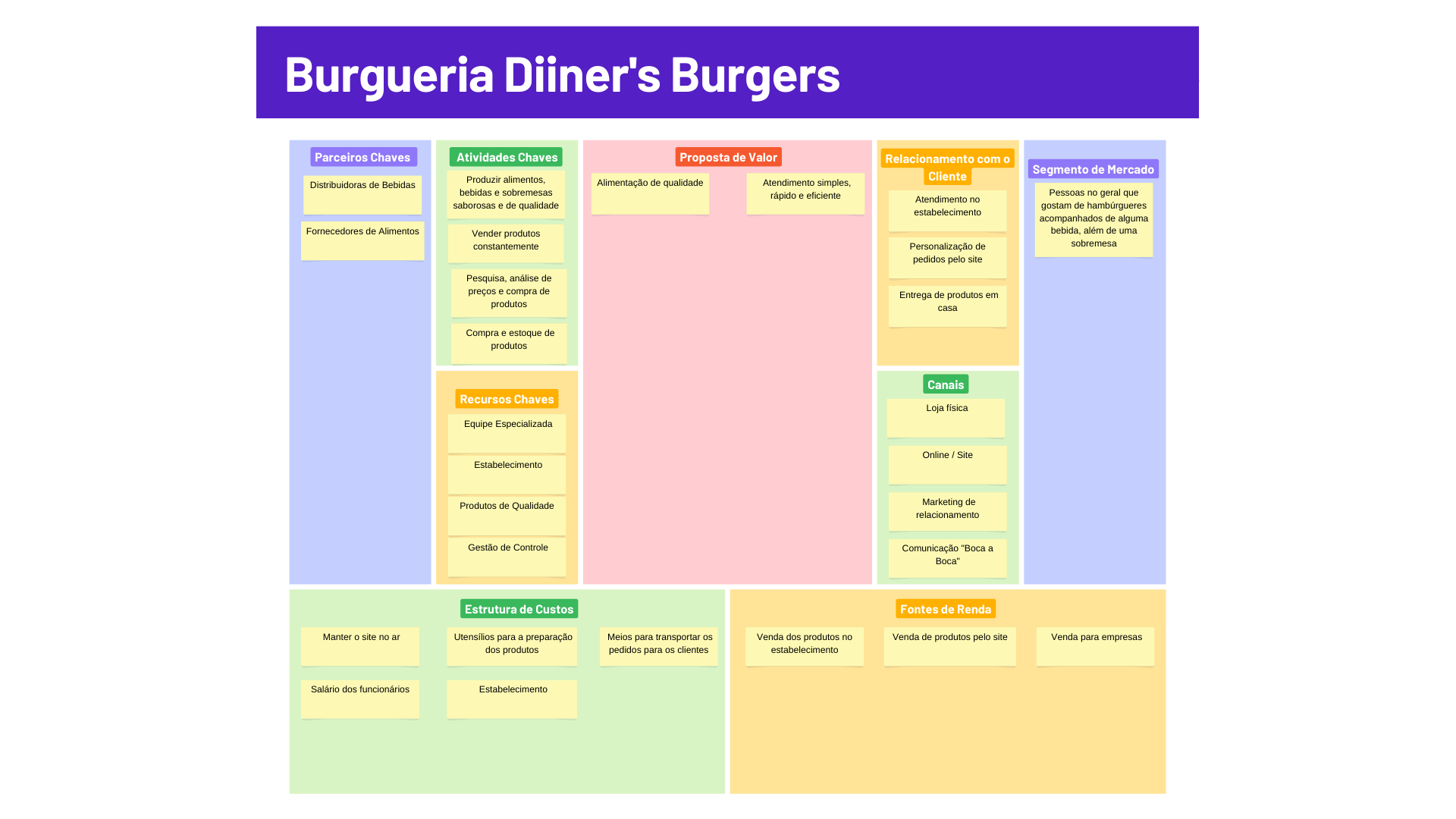
**Testes e Validação:** Realização de testes funcionais e de usabilidade para assegurar que o sistema atende às expectativas dos usuários.

**Lançamento e Monitoramento:** Disponibilização do site ao público, com monitoramento contínuo e suporte técnico para garantir seu funcionamento ideal.

**2 Viabilidade do Projeto**

A viabilidade do projeto é crucial para garantir que o investimento de recursos, tempo e tecnologia resulte em um produto que traga benefícios reais tanto para a hamburgueria quanto para os clientes. A proposta de valor do sistema está em sua capacidade de digitalizar o processo de vendas, oferecendo uma experiência prática e eficiente para os usuários, ao mesmo tempo em que automatiza e otimiza o gerenciamento de pedidos e estoque para a equipe interna. Isso permite que a Diiners Burgers não apenas mantenha a operação em períodos desafiadores, mas também expanda seu alcance de mercado e fortaleça seu relacionamento com os clientes.

2.1 Canvas de Negócio (*Business Model Canvas -* BMC)



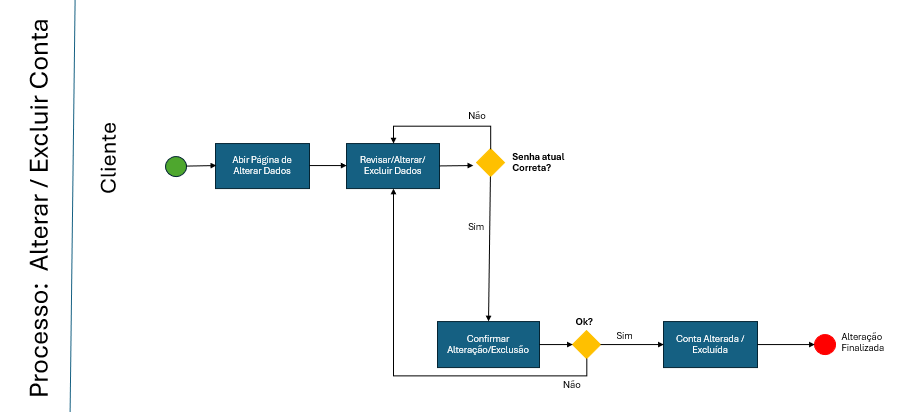
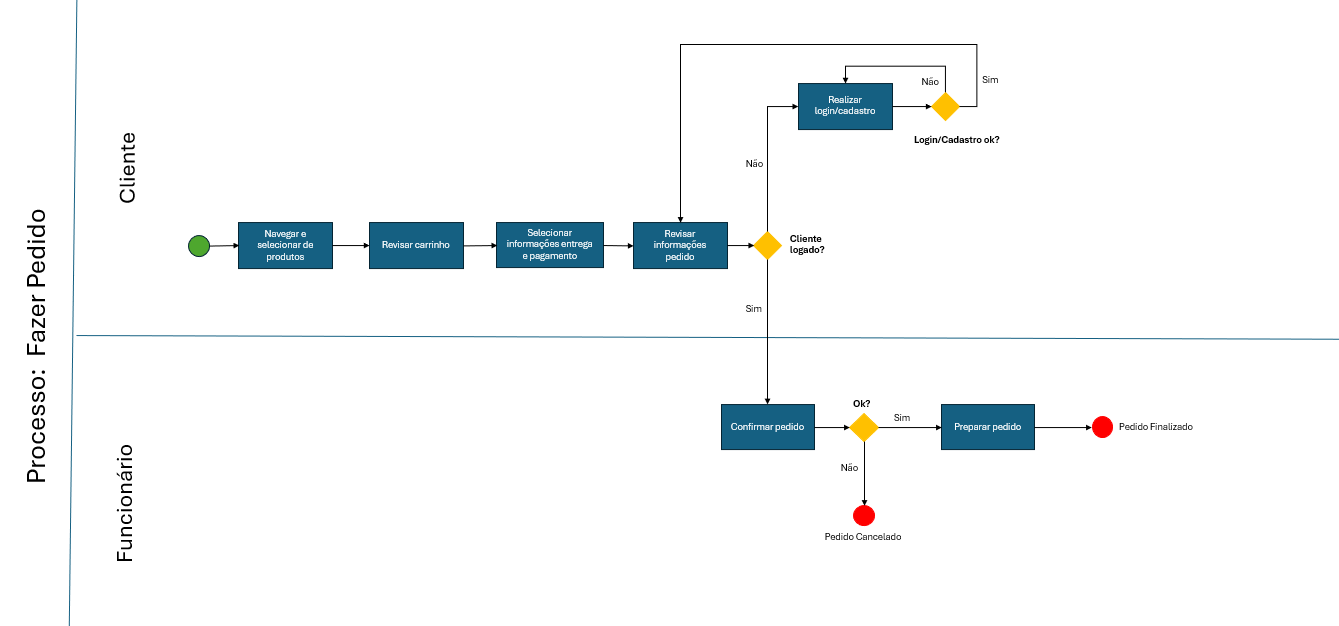
**3 Levantamento de Requisitos**

O levantamento de requisitos é uma etapa fundamental no desenvolvimento de qualquer sistema, pois garante que as necessidades reais do cliente e dos usuários sejam compreendidas de maneira clara e precisa. Permitindo que a equipe de desenvolvimento identifique as funcionalidades essenciais, as expectativas e os desafios específicos do negócio, permitindo que o projeto seja direcionado para atender às expectativas de forma eficaz.

**3.1 Elicitação e especificação dos Requisitos**

Um questionário foi desenvolvido para levantar os requisitos que o cliente considerava essenciais para a criação do sistema, depois desse levantamento foi realizada uma pesquisa de campo, analisando pessoalmente na hamburgueria se os requisitos levantados no questionário realmente seriam uteis ou se vinha a faltar algum outro. Ele deseja que a hamburgueria ganhe mais visibilidade e reconhecimento pela sua qualidade, além de tornar mais fácil para os usuários realizarem pedidos e localizarem a hamburgueria.

**3.2. BPMN**



**3.3 Requisitos Funcionais**

**Quadro 1** – Requisitos Funcionais do sistema

|  |  |  |
| --- | --- | --- |
| **RF001** - Cadastro de Clientes | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: O sistema deve permitir que o usuário se cadastre, para o seu cadastro devem ser informados os seguintes dados: Nome, e-mail (tendo uma validação para ver se o mesmo já está cadastrado, pois não pode haver dois iguais no banco de dados), telefone e senha (é bom ter um segundo campo para confirmar a senha). | | |

|  |  |  |
| --- | --- | --- |
| **RF002** - Feedback do cliente | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  ( ) Alta  ( ) Média  (X) Baixa |
| **Descrição**: O sistema deve mostrar, na página inicial, as avaliações dos clientes mostrando quem avaliou e os seus feedbacks. Devem ser mostradas três avaliações no mínimo. | | |

|  |  |  |
| --- | --- | --- |
| **RF003** – Localização da Hamburgueria | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: O sistema deve mostrar, na página inicial, a localização da hamburgueria, por meio de um mapa em que o cliente possa localizar a hamburgueria facilmente. Sendo sua localização: (Av. Maj. Nicácio, 1930 - Centro, Franca - SP, 14400-850.). | | |

|  |  |  |
| --- | --- | --- |
| **RF004** – Sobre a Hamburgueria | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: O sistema deve mostrar uma breve apresentação da hamburgueria, contando um pouco da sua história, mostrando algumas imagens do estabelecimento e algumas de suas premiações. | | |

|  |  |  |
| --- | --- | --- |
| **RF005** – Realizar Pedido | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  (X) Altíssima  ( ) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: O sistema deve ter uma área onde seja apresentado os produtos vendidos pela Diiners Burgers, separados por categorias (Lanches, Sobremesas, Bebidas), contendo seu nome, preço, uma breve descrição e uma foto. Dando a opção do cliente adicioná-los ao carrinho, que poderá ser exibido ao clicar em seu símbolo, nele deve ser mostrado a quantidade de cada produto desejável (permitindo que ela seja alterada), uma observação para cada produto (caso o cliente deseje alterar/adicionar alguma coisa no produto, além do preço da compra (calculado de acordo com a quantidade de produtos), taxa de entrega e o total a pagar. Ao final deverá haver um botão para finalizar a compra, quando ele for pressionado deverá ser cadastrado o pedido no banco de dados, caso o cliente já esteja logado/cadastrado na sua conta. | | |

|  |  |  |
| --- | --- | --- |
| **RF006** – Alterar Dados de Clientes | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  (X) Alta  ( ) Média  ( ) Baixa |
| **Descrição**: O sistema deve permitir que o cliente possa alterar os seus dados, sendo os mesmos informados durante o seu cadastro. | | |

|  |  |  |
| --- | --- | --- |
| **RF007** – Excluir conta | Categoria:  ( ) Oculto  (X)Evidente | Prioridade:  ( ) Altíssima  ( ) Alta  (X) Média  ( ) Baixa |
| **Descrição**: O sistema deve permitir que o cliente possa excluir a sua conta, sendo todos os seus dados e dados ligados ao dele, sejam removidos do banco de dados. | | |

**3.4 Requisitos Não Funcionais**

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **RNF001** - Login de Clientes | O sistema deve permitir que o usuário/cliente entre em sua área, sendo necessário que ele informe o e-mail e a senha utilizados no cadastro. | Tipo: Segurança | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF002** - Tempo de resposta do site | O sistema deve ter um tempo de resposta, para cada função que ele executar, de no máximo 5 segundos. | Tipo: Desempenho | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF003** - Tratamento de dados sensíveis | O sistema deve criptografar os dados sensíveis ao serem armazenados no banco de dados. | Tipo: Segurança | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF004** - Interface intuitiva | O sistema deve apresentar uma interface bonita, tendo uma identidade visual, além de ser intuitiva, facilitando a usabilidade para o usuário. | Tipo: Usabilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF005** - Interface responsiva | O sistema deve ser responsivo, podendo ser exibido em diferentes telas, como celulares, tablets e monitores de Pc’s. Sendo que ao ser exibida nas diferentes telas ele não deve apresentar bugs visuais. | Tipo: Usabilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF006** – Interação com Banco de Dados | O sistema deve ter uma conexão e interagir com o banco de dados. Armazenando, alterando, excluindo e exibindo os dados quando necessário. | Tipo: Compatibilidade | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |
| **RNF007** – Tipo do Sistema | O sistema deve ser um site. Para o seu desenvolvimento, deverão ser utilizadas as linguagens HTML5, CSS3, Javascript, além de Node JS e React. Poderão ser utilizados alguns frameworks cabíveis. | Tipo: Arquitetura | ( ) Desejável  (X) Obrigatório | (X) Permanente  ( ) Transitório |

**3.5 Regras de Negócio**

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

|  |
| --- |
| **RN001 - Vendas de produtos** |
| **Descrição**: Só deverão ser vendidos produtos que estiverem no cardápio, pode haver pedidos com produtos com alterações de ingredientes e preferencias, porém os produtos a serem alterados devem estar no cardápio. E no caso de entrega, os produtos só serão entregues caso o cliente tenha efetuado o pagamento antes, o pedido pode sair para a entrega antes de ser realizado o pagamento, porém o cliente só o terá em mão caso já tenha pagado. Caso o cliente desejar se alimentar no local físico, o pagamento pode ser feito depois. Não é permitido “pendurar a conta”. |
| **RN002 - Variedade de opções de pedidos** |
| **Descrição**: Devem ser oferecidos uma ampla variedade de produtos para atender a diferentes preferências do cliente. |
| **RN003 - Atendimento ao cliente** |
| **Descrição**: Deve-se garantir um excelente atendimento ao cliente, garantindo que os clientes sejam recebidos calorosamente, seus pedidos sejam tratados com precisão e quaisquer problemas sejam resolvidos prontamente. |
| **RN004 - Velocidade de serviço** |
| **Descrição**: Os tempos de espera devem ser curtos e eficientes para garantir uma experiência de refeição rápida e conveniente. |
| **RN005 - Marketing e inovação** |
| **Descrição**: Devem ser implementadas estratégias de marketing eficazes para atrair e reter clientes, como promoções. E estar aberto a novas tendências, inovando-se para manter a relevância e atratividade no mercado. |
| **RN006 - Feedback do cliente** |
| **Descrição**: Estar aberto ao feedback dos clientes e usar essa informação para melhorar continuamente os produtos e serviços oferecidos, além de usa-los como méritos quando o feedback for satsifatório. |

**3.6 Casos de Uso**

**Quadro 4** – Use Case

|  |  |
| --- | --- |
| **Caso de Uso – Cadastrar Clientes** | |
| **ID** | UC 001 |
| **Descrição** | Este caso de uso tem por objetivo cadastrar o cliente. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Email a ser cadastrado não estar cadastrado. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página de cadastrar-se. 2. O sistema carrega o formulário de cadastro. 3. O usuário informa os dados válidos: Nome, Email, Telefone, Senha e Senha de confirmação. 4. O sistema efetua as validações e cadastra o cliente no banco de dados. 5. O sistema entra na página desejável. |
| **Pós-condição** | O usuário é cadastrado no banco de dados e pode acessar as páginas restritas para usuários logados. |
| **Cenário Alternativo** | 4a – O usuário informa algum dado não válido.  4a.1 O sistema retorna uma mensagem informando qual é o dado inválido. |

|  |  |
| --- | --- |
| **Caso de Uso – Logar** | |
| **ID** | UC 002 |
| **Descrição** | Este caso de uso tem por objetivo o usuário entrar em páginas restritas somente a usuários cadastrados e logados. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário já estar cadastrado. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página de login.  2. O sistema carrega o formulário de login.  3.O usuário informa os dados válidos: Email e Senha.  4. O sistema efetua as validações.  5. O sistema entra na página home. |
| **Pós-condição** | O usuário pode acessar as páginas restritas a usuários cadastrados e logados. |
| **Cenário Alternativo** | 4a – O usuário informa algum dado não válido.  4a.1 O sistema retorna uma mensagem informando qual é o dado inválido.  4b – O usuário extrapola o limite de tentativas de senha.  4b.1 – O sistema bloqueia o usuário naquela máquina por 30 minutos. |

|  |  |
| --- | --- |
| **Caso de Uso – Atualizar Dados Cadastro Cliente** | |
| **ID** | UC 003 |
| **Descrição** | Este caso de uso tem por objetivo o usuário alterar os seus dados informados no cadastro. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário já estar cadastrado e logado. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página de alterar dados pessoais. 2. O sistema carrega o formulário com os dados informados no cadastro. 3. O usuário informa os dados válidos: Nome, Email, Telefone, Senha Antiga e Nova Senha. 4. O sistema efetua as validações. 5. O sistema altera os dados no banco de dados e recarrega a página. |
| **Pós-condição** | O usuário tem os seus dados alterados no banco de dados. |
| **Cenário Alternativo** | 4a – O usuário informa algum dado não válido.  4a.1 O sistema retorna uma mensagem informando qual é o dado inválido. |

|  |  |
| --- | --- |
| **Caso de Uso – Realizar Pedidos** | |
| **ID** | UC 004 |
| **Descrição** | Este caso de uso tem por objetivo o usuário realizar um pedido. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário já estar cadastrado e logado. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página de fazer pedido. 2. O sistema carrega o menu com todos os produtos cadastrados que posem ser inclusos no pedido, separados por categoria. 3. O usuário seleciona os produtos e adiciona-os no carrinho. 4. O usuário clica no carrinho, e escolhe se vai ser entrega ou não. 5. O sistema realiza a validação para ver se os produtos existem em estoque. E se for entregar, se o endereço informado (CEP) é valido. 6. O sistema cadastra o pedido no banco de dados. |
| **Pós-condição** | O pedido é cadastrado e enviado ao funcionário com o status em aberto. |
| **Cenário Alternativo** | 1a – O usuário não está logado.  1a.1 – O sistema redireciona o usuário para a página de login.  4a – O usuário seleciona um produto que não tem no estoque ou informa um endereço incorretamente.  4a.1 O sistema retorna uma mensagem informando qual o erro. |

|  |  |
| --- | --- |
| **Caso de Uso – Visualizar Pedidos** | |
| **ID** | UC 005 |
| **Descrição** | Este caso de uso tem por objetivo o usuário visualizar os pedidos feitos. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário já estar cadastrado/logado e ter um pedido(s) cadastrados. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na sua página home, onde serão exibidos os pedidos feitos. 2. O sistema exibe todos os pedidos feitos com os dados: Número do pedido, tipo de pagamento, método de envio, valor total e status. |
| **Pós-condição** | O usuário consegue visualizar os status dos pedidos. |
| **Cenário Alternativo** | 2a – Não há nenhum pedido solicitado.  2a.1 – O sistema exibe uma mensagem de que não há pedidos cadastrados. |

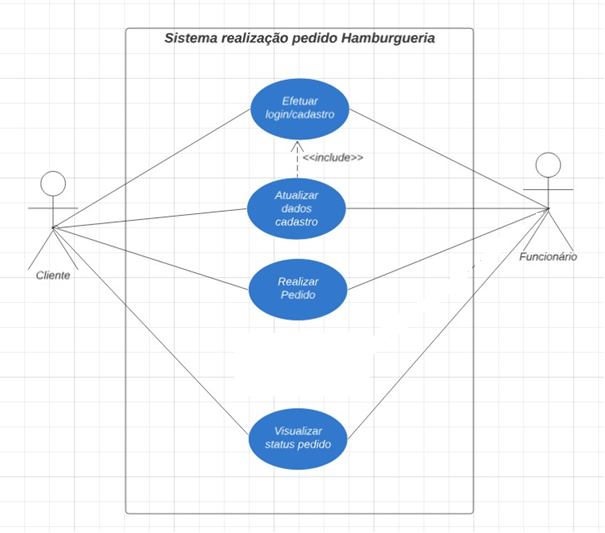
|  |  |
| --- | --- |
| **Caso de Uso – Excluir Conta** | |
| **ID** | UC 006 |
| **Descrição** | Este caso de uso tem por objetivo o usuário excluir a sua conta do site. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário já estar cadastrado/logado. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página de dados do cliente. 2. O sistema exibe todos os dados do cliente e a opção de excluir a conta. 3. O usuário insere a senha atual correta e clica em excluir a conta. 4. O sistema exclui todos os dados do cliente no banco de dados. |
| **Pós-condição** | O usuário exclui a sua conta, pagando todos os seus dados e os dados ligados ao seu do banco de dados. |
| **Cenário Alternativo** | 3a – O usuário não insere a senha correta.  3a.1 – O sistema emite uma mensagem de senha incorreta. |

|  |  |
| --- | --- |
| **Caso de Uso – Informativo** | |
| **ID** | UC 007 |
| **Descrição** | Este caso de uso tem por objetivo o usuário conhecer a hamburgueria. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário estar com o site aberto. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página que conta sobre a hamburgueria.  2. O sistema exibe as informações sobre a hamburgueria. |
| **Pós-condição** | O usuário fica informado a respeito da hamburgueria. |
| **Cenário Alternativo** |  |

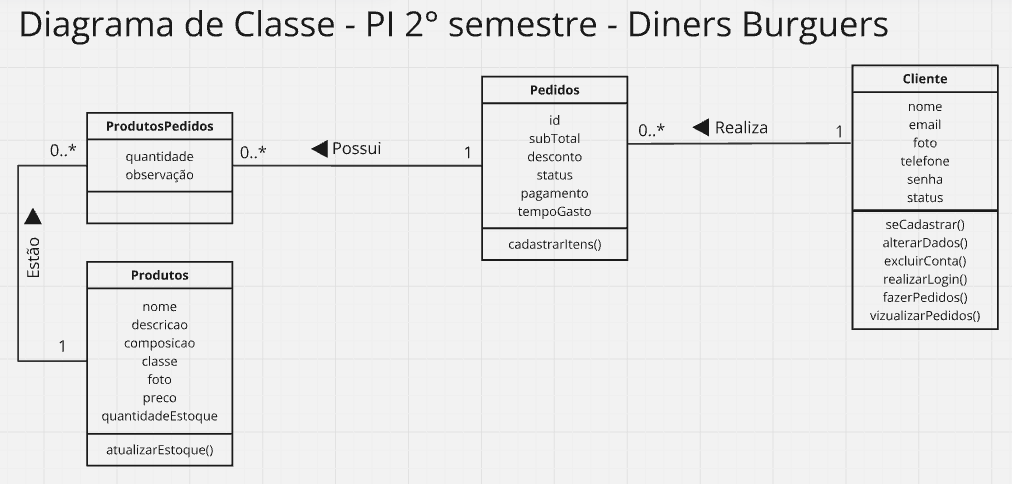
|  |  |
| --- | --- |
| **Caso de Uso – Localização** | |
| **ID** | UC 008 |
| **Descrição** | Este caso de uso tem por objetivo o usuário conhecer a localização da hamburgueria. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário estar com o site aberto. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página inicial do site.  2. O sistema exibe a localização da hamburgueria  3. O usuário visualiza a localização, podendo ver se está ao seu alcance. |
| **Pós-condição** | O usuário fica informado a respeito da localização da hamburgueria. |
| **Cenário Alternativo** | 2a – Há uma falha com a API de localização.  2a.1 – O sitema não consegue exibir a localização. |

|  |  |
| --- | --- |
| **Caso de Uso – Avaliações** | |
| **ID** | UC 009 |
| **Descrição** | Este caso de uso tem por objetivo o usuário visualizar as avaliações de outros usuáriosa. |
| **Ator Primário** | Cliente |
| **Pré-condição** | Usuário estar com o site aberto. |
| **Cenário Principal** | 1. O use case inicia quando o usuário entra na página inicial do site.  2. O sistema exibe as avaliações de outros clientes.  3. O usuário visualiza as avaliações de outros clientes. |
| **Pós-condição** | O usuário fica informado a respeito da qualidade da hamburgueria de acordo com as suas avaliações. |
| **Cenário Alternativo** |  |

**3.7. Diagrama de Caso de Uso**

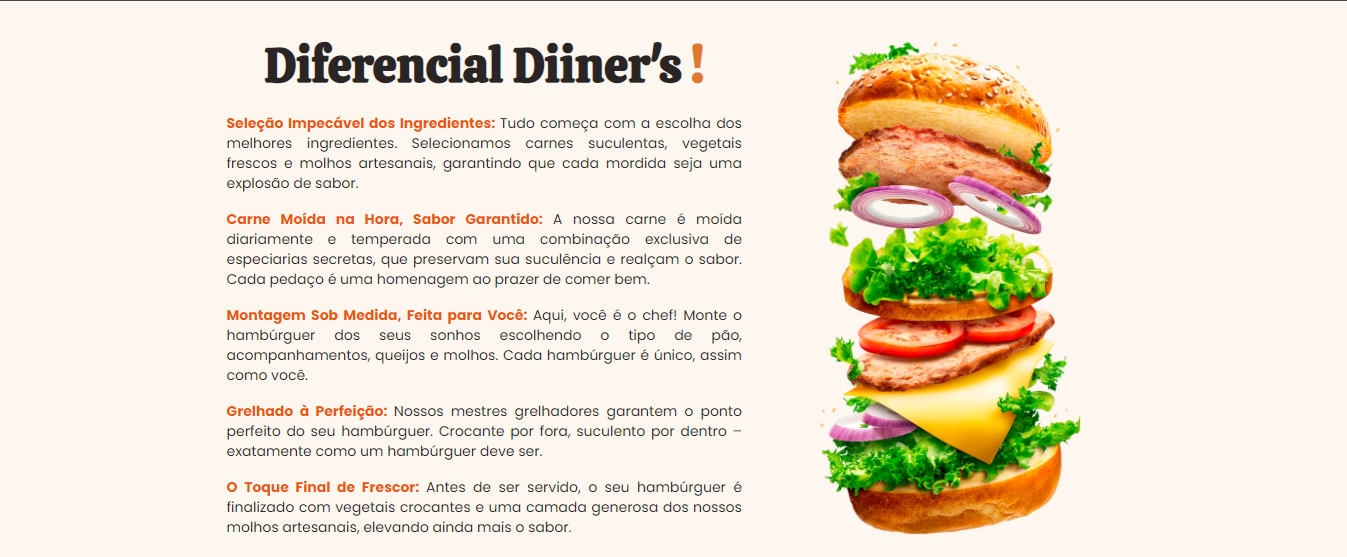


**3.8. Diagrama de Classes**



**4.0 – Protótipo da aplicação**

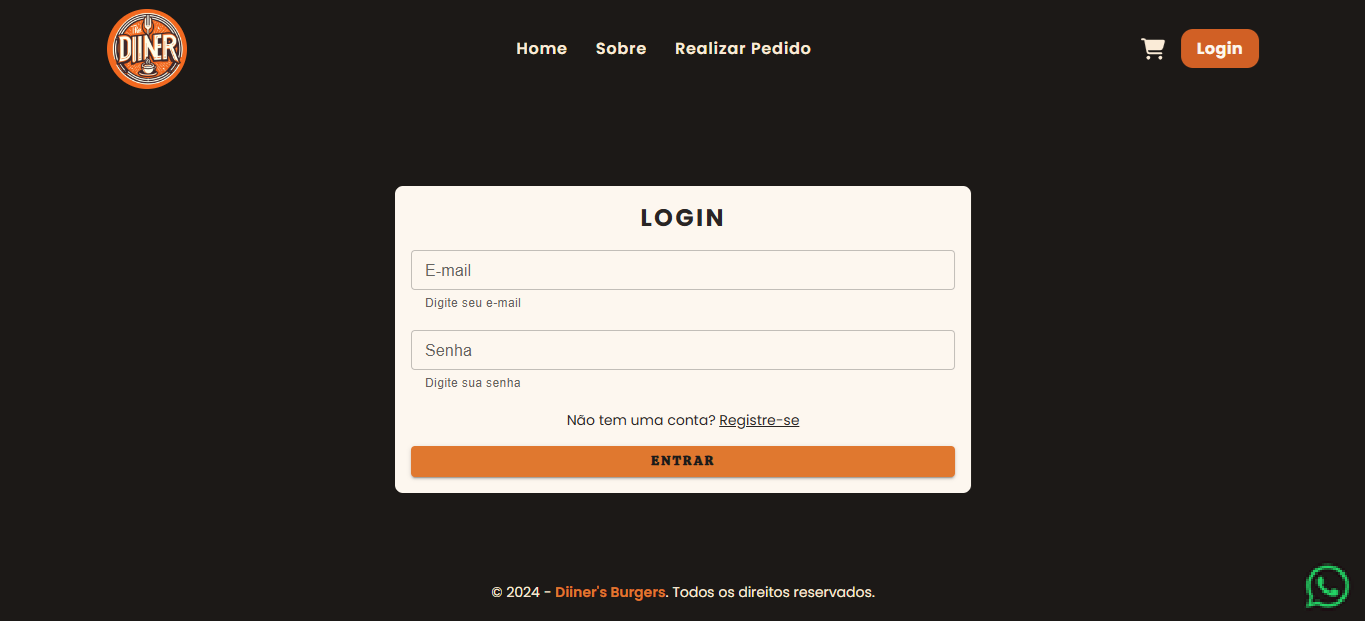
**Página Inicial:**



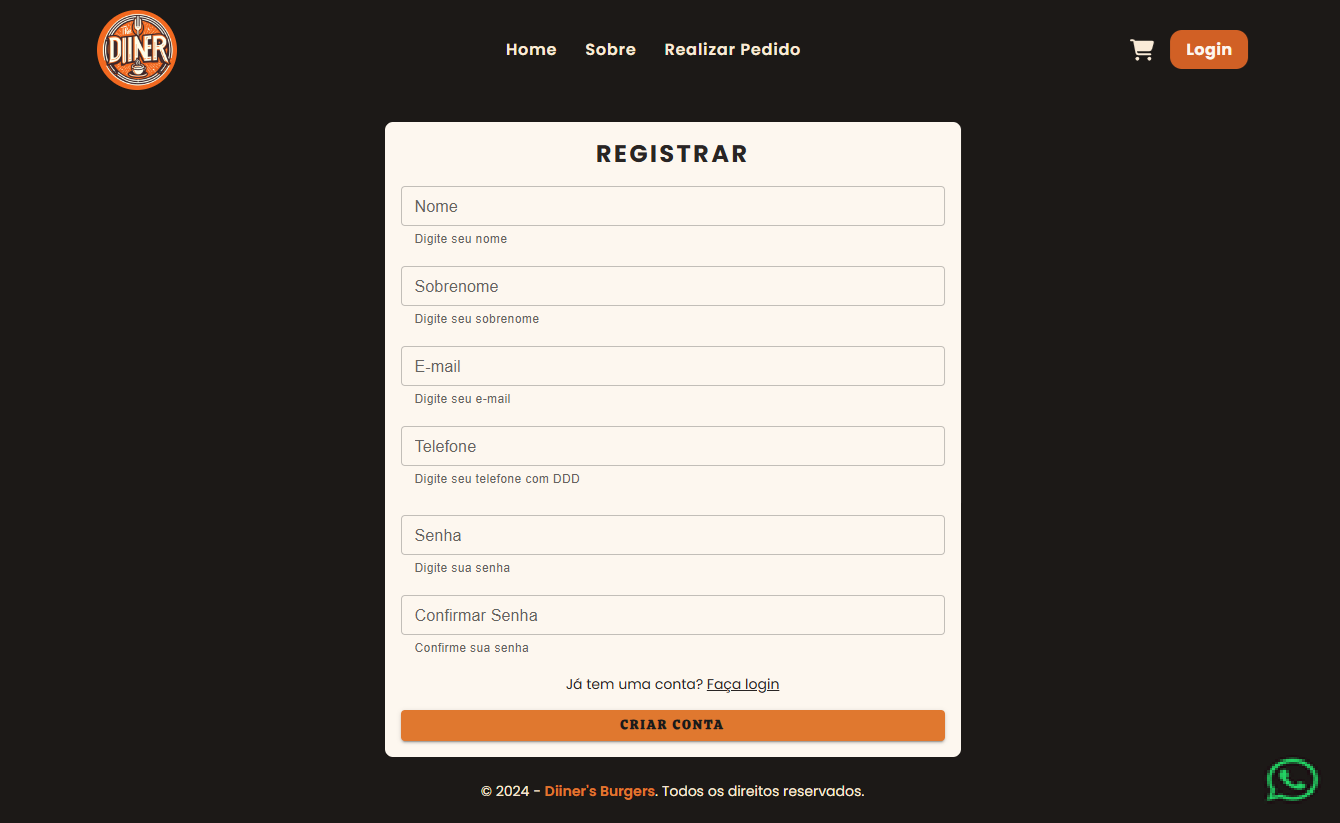
**Página Sobre:**



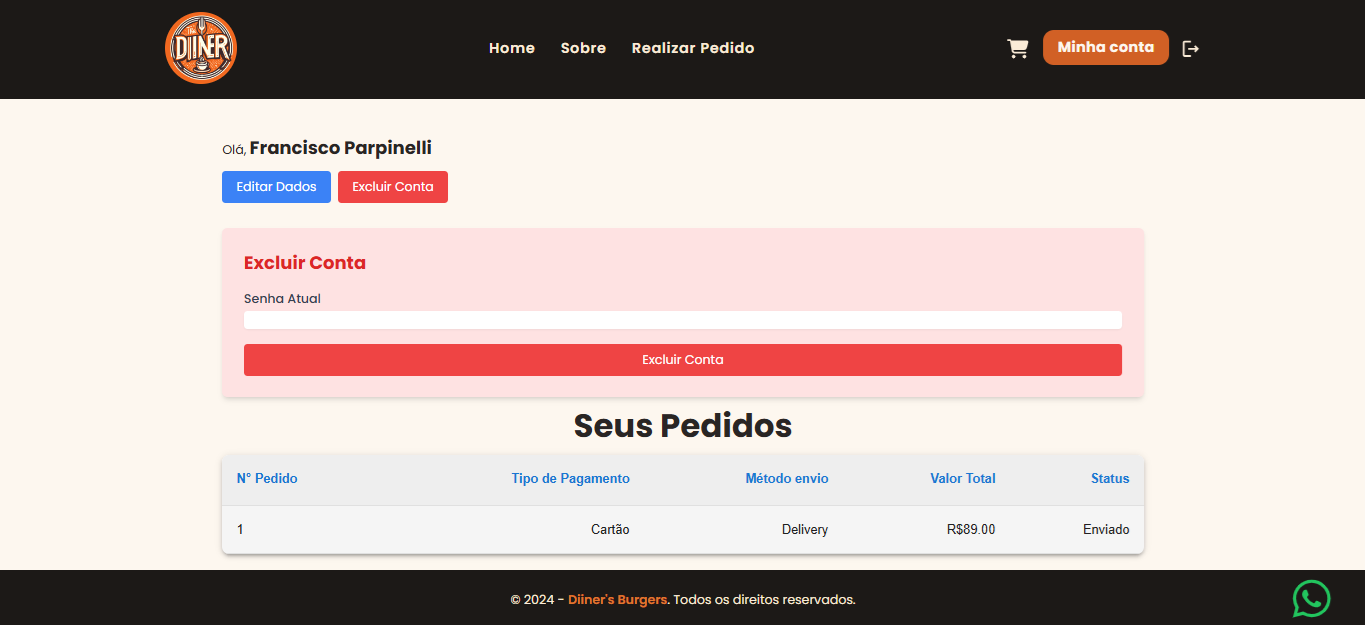
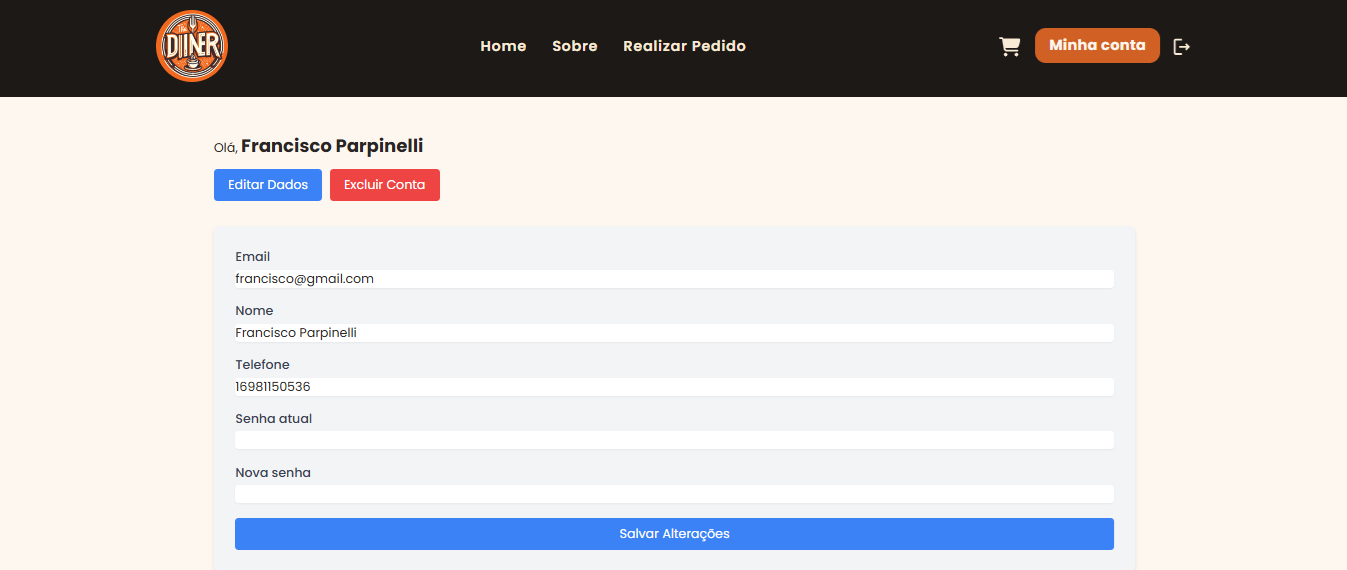
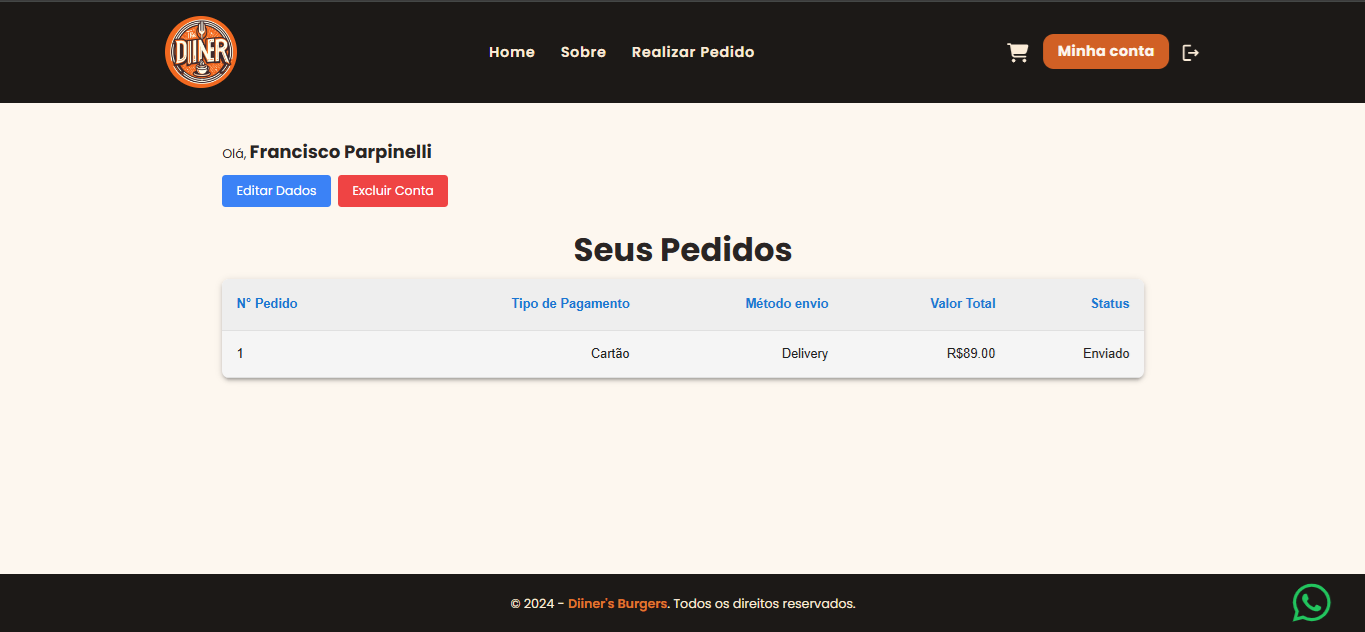
**Página Login:**



**Página Cadastrar-se:**



**Página Home Cliente:**



**Página Fazer Pedido:**

