🍰 Especificação Ágil - Cupcake App

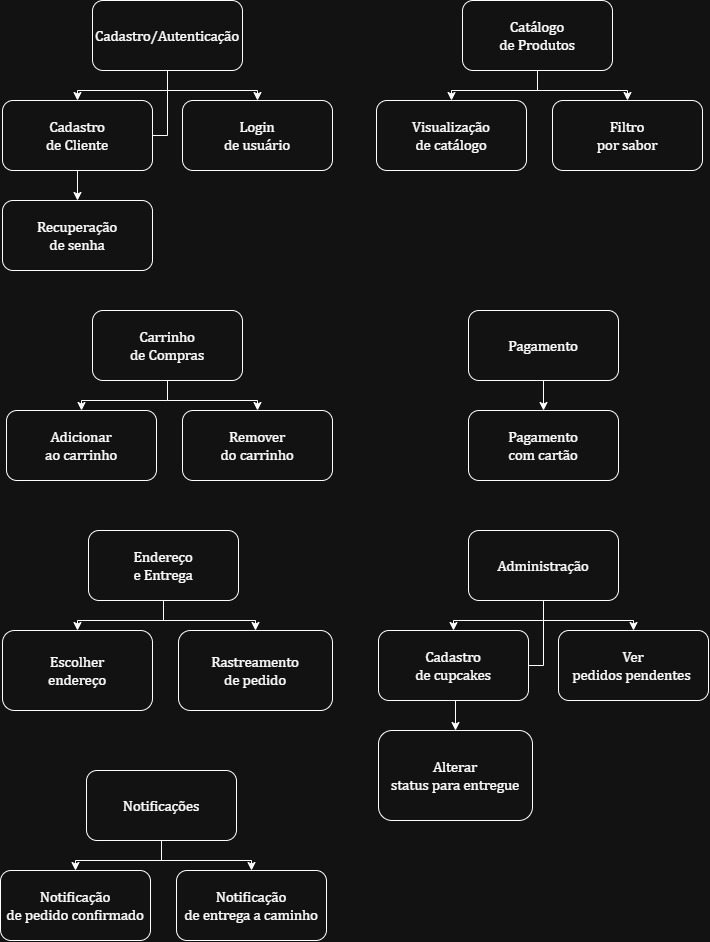
# *Situação-Problema 1*

# 1. Resumo do Projeto

Nome do Produto: Cupcake App  
Objetivo: Criar um aplicativo móvel para uma loja de cupcakes gourmet, oferecendo vitrine virtual de produtos, sistema de pedidos online, pagamentos eletrônicos e acompanhamento de entregas.  
Público-alvo: Clientes finais (consumidores de cupcakes) e administradores da loja (gestores de pedidos e produtos).  
Metodologia: Scrum com entregas incrementais baseadas em histórias de usuário.

# 2. Mapa de Afinidade das Histórias de Usuário

Cadastro/Autenticação: ID 01, 02, 03  
Catálogo de Produtos: ID 04, 05  
Carrinho de Compras: ID 06, 07  
Pagamento: ID 08  
Endereço e Entrega: ID 09, 10  
Administração: ID 11, 12, 13  
Notificações: ID 14, 15



# 3. Backlog de Produto Priorizado

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **História do usuário** | **Estimativa em pontos** | **Prioridade** |
| nº 1 | Como visitante, quero me cadastrar para poder fazer pedidos. | 3 | A |
| nº 2 | Como usuário, quero fazer login para acessar o app. | 2 | A |
| nº 3 | Como usuário, quero recuperar minha senha por e-mail. | 3 | B |
| nº 4 | Como usuário, quero visualizar os cupcakes disponíveis no catálogo. | 3 | A |
| nº 5 | Como usuário, quero filtrar cupcakes por sabor. | 2 | B |
| nº 6 | Como usuário, quero adicionar cupcakes ao carrinho. | 2 | A |
| nº 7 | Como usuário, quero remover cupcakes do carrinho. | 1 | C |
| nº 8 | Como comprador, quero pagar com cartão de crédito pelo app. | 5 | A |
| nº 9 | Como usuário, quero cadastrar/selecionar meu endereço para entrega. | 3 | B |
| nº 10 | Como cliente, quero acompanhar o status da entrega do meu pedido. | 4 | B |
| nº 11 | Como administrador, quero cadastrar novos cupcakes no sistema. | 3 | A |
| nº 12 | Como administrador, quero visualizar pedidos pendentes de entrega. | 2 | A |
| nº 13 | Como administrador, quero alterar o status dos pedidos para "Entregue". | 2 | A |
| nº 14 | Como cliente, quero receber notificação quando o pedido for confirmado. | 2 | C |
| nº 15 | Como cliente, quero ser avisado quando o entregador estiver a caminho. | 2 | C |

# 4. Requisitos e Tarefas por História (Exemplo)

## ID 01 – Cadastro de Cliente

Requisitos Funcionais:

* - Formulário com campos obrigatórios: nome, e-mail, telefone, senha.
* - Verificação de e-mail único.

Requisitos Não Funcionais:

* - Resposta da tela em até 3 segundos.

**Tarefas:**

* - Criar interface de cadastro.
* - Validar dados do formulário.
* - Criar endpoint de API para registro.

## ID 2 – Login de usuário

Requisitos Funcionais:

* - Usuário deve acessar com e-mail e senha.
* - Apenas usuários ativos podem acessar.

Requisitos Não Funcionais:

* - Tempo de autenticação: até 2 segundos.

**Tarefas:**

* - Criar tela de login.
* - Implementar autenticação.
* - Validar senha criptografada.

## ID 3 – Recuperação de senha

Requisitos Funcionais:

* - Sistema deve enviar e-mail com link de redefinição.
* - Link deve expirar em 24 horas.

Requisitos Não Funcionais:

* - Enviar e-mail em até 5 segundos.

**Tarefas:**

* - Criar tela 'Esqueci minha senha'.
* - Implementar envio automático de link.
* - Validar link temporário.

## ID 4 – Visualização de catálogo

Requisitos Funcionais:

* - Listar cupcakes com imagem, nome, preço e descrição.
* - Produtos esgotados não aparecem.

Requisitos Não Funcionais:

* - Interface responsiva.

**Tarefas:**

* - Criar layout do catálogo.
* - Conectar com base de dados.
* - Implementar filtro por disponibilidade.

## ID 5 – Filtro por sabor

Requisitos Funcionais:

* - Cliente pode filtrar por categorias de sabor.
* - Categorias definidas no painel do admin.

Requisitos Não Funcionais:

* - Resposta do filtro em até 2 segundos.

**Tarefas:**

* - Criar botões de filtro.
* - Conectar com backend de categorias.
* - Testar combinações de filtros.

## ID 6 – Adicionar ao carrinho

Requisitos Funcionais:

* - Usuário pode adicionar qualquer cupcake disponível.
* - Quantidade ajustável.

Requisitos Não Funcionais:

* - Persistência local mesmo offline.

**Tarefas:**

* - Criar botão de 'Adicionar ao carrinho'.
* - Atualizar quantidade dinâmica.
* - Sincronizar dados local/servidor.

## ID 7 – Remover do carrinho

Requisitos Funcionais:

* - Cliente pode excluir itens antes da compra.
* - Total deve ser recalculado.

**Tarefas:**

* - Adicionar botão de 'Remover'.
* - Atualizar total automaticamente.
* - Testar remoções múltiplas.

## ID 8 – Pagamento com cartão

Requisitos Funcionais:

* - Sistema deve aceitar dados de cartão.
* - Validar número, CVV, validade.

Requisitos Não Funcionais:

* - Comunicação via HTTPS.
* - Tempo de resposta da API: 3 segundos.

**Tarefas:**

* - Criar tela de pagamento.
* - Integrar com gateway.
* - Implementar tratamento de erros.

## ID 9 – Escolher endereço

Requisitos Funcionais:

* - Permitir cadastrar e salvar endereços.
* - Validação de CEP e região de entrega.

Requisitos Não Funcionais:

* - Tela responsiva com máscara para-CEP.

**Tarefas:**

* - Criar formulário de endereço.
* - Validar formato e área.
* - Salvar endereço no perfil do usuário.

## ID 10 – Rastreamento de pedido

Requisitos Funcionais:

* - Cliente pode acompanhar status: preparando, a caminho, entregue.
* - Admin atualiza status.

**Tarefas:**

* - Criar tela de rastreamento.
* - Atualizar status em tempo real.
* - Exibir ícones visuais de status.

## ID 11 – Cadastro de cupcakes (admin)

Requisitos Funcionais:

* - Admin pode inserir imagem, nome, descrição e preço.
* - Preço deve ser maior que zero.

**Tarefas:**

* - Criar painel de cadastro.
* - Validar campos obrigatórios.
* - Salvar cupcake na base.

## ID 12 – Ver pedidos pendentes (admin)

Requisitos Funcionais:

* - Listar pedidos aguardando entrega.
* - Exibir nome do cliente e hora do pedido.

**Tarefas:**

* - Criar painel de pedidos.
* - Listar com filtros por status.
* - Adicionar busca por cliente.

## ID 13 – Alterar status para entregue (admin)

Requisitos Funcionais:

* - Só pedidos pagos podem ser finalizados.
* - Registrar histórico de alterações.

**Tarefas:**

* - Criar botão de 'Marcar como entregue'.
* - Validar status atual do pedido.
* - Registrar data/hora da entrega.

## ID 14 – Notificação de pedido confirmado

Requisitos Funcionais:

* - Enviar push quando pagamento for confirmado.

Tarefas:

* - Integrar com serviço de push.
* - Criar template da notificação.
* - Testar envio no app.

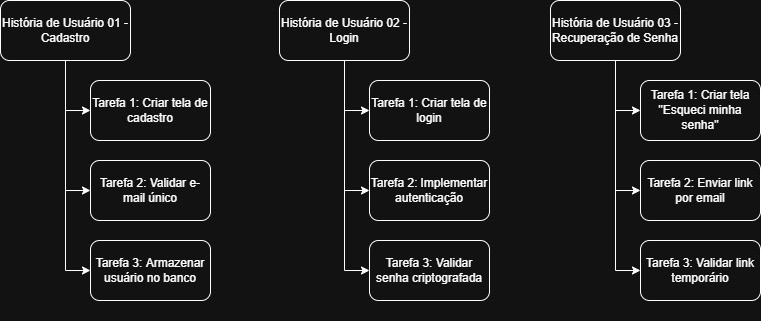
## ID 15 – Notificação de entrega a caminho

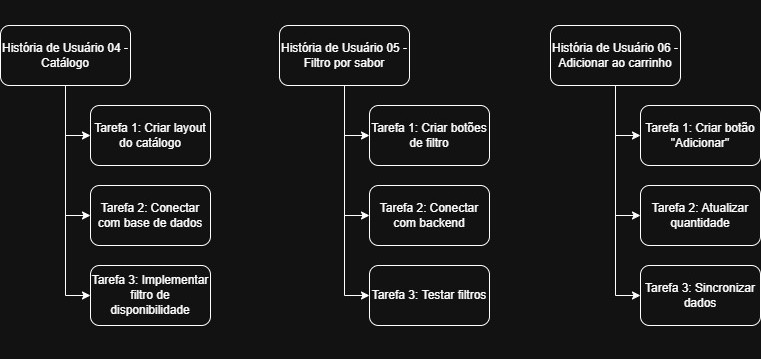
Requisitos Funcionais:

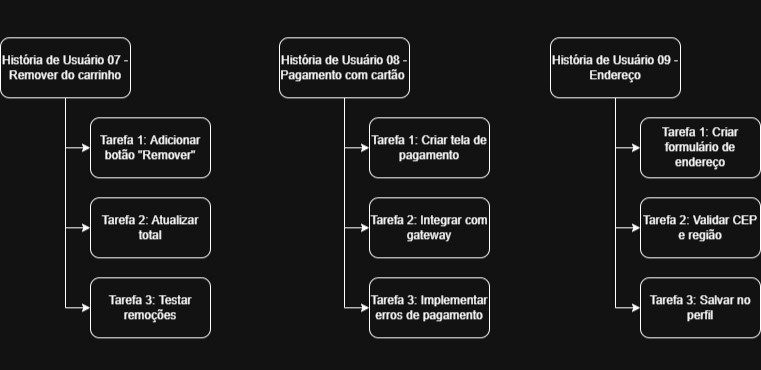
* - Enviar push quando status for 'a caminho'.

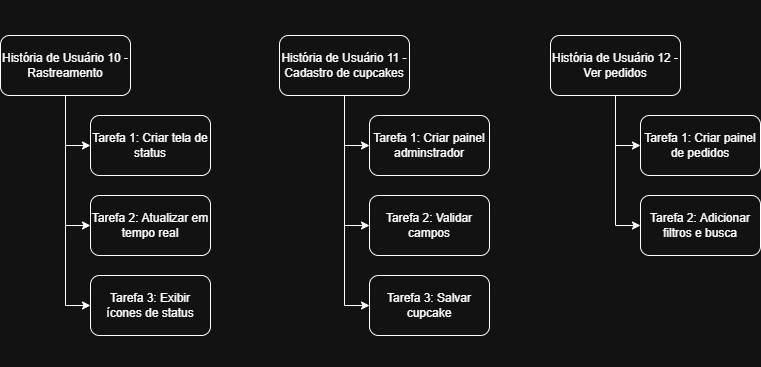
**Tarefas:**

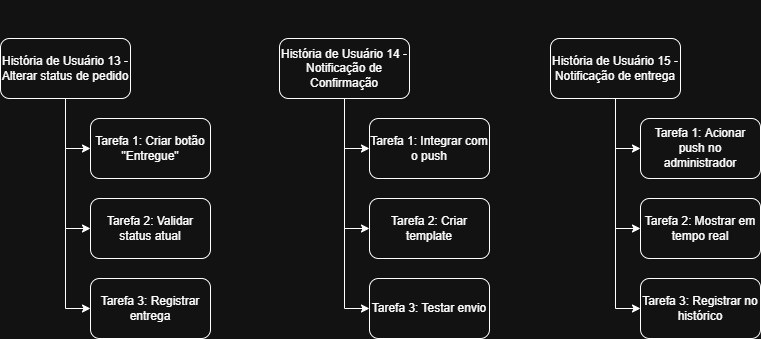
* - Acionar notificação via painel do admin.
* - Exibir em tempo real no app.
* - Registrar notificação no histórico.





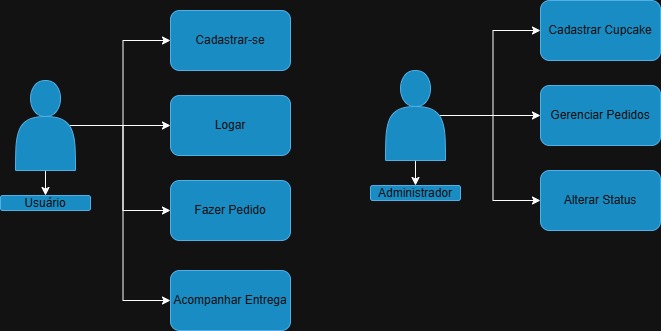






# *Situação-Problema 2*

# 5. Diagrama de caso de uso geral



Atores:

* Usuário
* Administrador

Casos de uso principais:

* Cadastrar-se
* Logar
* Visualizar Catálogo
* Fazer Pedido
* Acompanhar Entrega
* Cadastrar Cupcake
* Gerenciar Pedidos
* Alterar Status

# 6. Casos de uso expandidos

# 6.1 Caso de Uso: Fazer Pedido

**Nome**: Fazer Pedido.

**Identificador**: #CU01.

**Descrição**: Permitir que um cliente autenticado selecione cupcakes, adicione ao carrinho, escolha o endereço e finalize a compra via pagamento eletrônico.

**Pré-condições:**

* O cliente deve estar cadastrado e logado no sistema.
* O sistema deve estar online.

**Pós-condições:**

* O pedido é registrado com status "aguardando preparo".
* O cliente recebe confirmação do pedido.

**Caso básico de ação:**

1. O caso de uso começa quando o cliente acessa o sistema e deseja realizar uma compra;
2. O cliente acessa o catálogo de cupcakes por meio da TELA001 - Vitrine de Produtos;
3. O cliente seleciona um ou mais cupcakes e os adiciona ao carrinho;
4. O cliente acessa a TELA002 - Carrinho de Compras, onde pode revisar e ajustar os itens;
5. O cliente escolhe o endereço de entrega usando a TELA003 - Endereço;
6. O sistema valida o endereço conforme a RN001 - Validar área de entrega;
7. O cliente escolhe o método de pagamento (cartão ou PIX) por meio da TELA004 - Pagamento;
8. O sistema processa o pagamento de acordo com a RN002 - Processar pagamento eletrônico;
9. O sistema registra o pedido e o associa ao cliente e ao endereço;
10. O sistema envia uma notificação via push confirmando a compra (TELA005 - Confirmação);
11. O caso de uso termina.

**Curso Alternativo Alfa:** Falha no pagamento

* Alfa7: O sistema detecta falha no pagamento e exibe mensagem de erro.
* Alfa8: O cliente é instruído a tentar outro método de pagamento.
* Alfa9: O caso de uso retorna para a etapa 7.

**Curso Alternativo Beta:** Cliente cancela o pedido

* Beta5: O cliente acessa o carrinho mas decide não continuar.
* Beta6: O caso de uso termina sem registro de pedido.

**Curso Alternativo Delta:** Endereço fora da área de entrega

* Delta6: O sistema detecta endereço inválido.
* Delta7: O sistema informa o cliente e sugere endereços alternativos salvos.
* Delta8: O cliente retorna à TELA003 para modificar o endereço.
* Delta9: O caso de uso retorna para a etapa 5.

# 6.2 Caso de Uso: Gerenciar Pedido

**Nome:** Gerenciar Pedidos

**Identificador:** #CU02

**Descrição:** Permitir que o administrador visualize e lientse o status dos pedidos feitos pelos lients.

**Pré-condições:**

* O administrador deve estar logado no sistema.
* Haver pedidos registrados.

**Pós-condições:**

* O status dos pedidos é atualizado.
* O cliente recebe uma notificação sobre a alteração.

**Caso básico de ação:**

1. O caso de uso começa quando o administrador acessa o painel administrativo (TELA010 - Painel Admin);
2. O sistema exibe a lista de pedidos em aberto;
3. O administrador seleciona um pedido para atualização;
4. O administrador altera o status para "em preparo", "a caminho" ou "entregue";
5. O sistema salva a alteração e envia uma notificação ao cliente (RN005 - Atualizar status e notificar);
6. O caso de uso termina.

**Curso Alternativo Alfa:** Erro ao alterar status

* **Alfa4:** O sistema apresenta erro de conexão**.**
* **Alfa5:** O administrador é instruído a tentar novamente**.**
* **Alfa6:** O caso de uso retorna para a etapa 3.

**Curso Alternativo Beta: Pedido já entregue**

* **Beta3:** O administrador tenta alterar um pedido com status "entregue".
* **Beta4:** O sistema informa que não é possível reverter o status final.
* **Beta5:** O caso de uso termina.

# 6.3 Caso de Uso: Cadastrar-se

**Nome:** Cadastrar-se

**Identificador:** #CU03

**Descrição:** Permitir que um novo usuário crie uma conta no aplicativo com seus dados pessoais.

**Pré-condições:**

* O usuário não deve ter uma conta já registrada com o mesmo e-mail.

**Pós-condições:**

* O usuário tem uma conta ativa e pode logar no sistema.

**Caso básico de ação:**

1. O usuário acessa a TELA006 - Tela de Cadastro;
2. Preenche os campos obrigatórios: nome, e-mail, telefone, senha;
3. O sistema valida o e-mail de acordo com RN010 - E-mail único;
4. O sistema salva os dados e cria a conta;
5. O sistema exibe uma mensagem de sucesso;
6. O caso de uso termina.

**Curso Alternativo Alfa: E-mail já existente**

* **Alfa3:** O sistema detecta e-mail duplicado;
* **Alfa4:** O sistema informa que o usuário já está registrado;
* **Alfa5:** O caso de uso termina.

**Curso Alternativo Beta: Dados inválidos**

* **Beta3:** O sistema detecta campos obrigatórios vazios;
* **Beta4:** O sistema alerta e destaca os erros;
* **Beta5:** O caso de uso retorna à etapa 2.

# 6.4 Caso de Uso: Visualizar Catálogo

**Nome:** Visualizar Catálogo

**Identificador:** #CU04

**Descrição:** Permitir que o cliente veja a lista de cupcakes disponíveis para compra.

**Pré-condições:**

* O sistema deve estar disponível com conexão ativa.

**Pós-condições:**

* O cliente visualiza os cupcakes ativos em estoque.

**Caso básico de ação:**

1. O usuário acessa a TELA001 - Vitrine de Produtos;
2. O sistema carrega todos os cupcakes disponíveis;
3. Os produtos são apresentados com imagem, nome, descrição e preço;
4. O usuário pode navegar e clicar em produtos para ver mais detalhes;
5. O caso de uso termina.

**Curso Alternativo Alfa: Nenhum produto disponível**

* **Alfa3:** O sistema detecta que não há cupcakes disponíveis;
* **Alfa4:** Exibe uma mensagem de "Catálogo indisponível";
* **Alfa5:** O caso de uso termina.

# 6.5 Caso de Uso: Adicionar ao Carrinho

**Nome:** Adicionar ao Carrinho

**Identificador:** #CU05

**Descrição:** Permitir que o cliente adicione cupcakes ao carrinho de compras.

**Pré-condições:**

* O usuário deve estar logado.

**Pós-condições:**

* O produto é adicionado ao carrinho.

**Caso básico de ação:**

1. O cliente visualiza um cupcake na TELA001;
2. Clica em "Adicionar ao carrinho";
3. O sistema verifica disponibilidade do produto (RN020 - Verificar estoque);
4. O item é adicionado com a quantidade desejada;
5. O sistema atualiza o total do carrinho;
6. O caso de uso termina.

**Curso Alternativo Alfa: Produto esgotado**

* **Alfa3:** O sistema informa que o produto está indisponível;
* **Alfa4:** O botão de adicionar é desativado;
* **Alfa5:** O caso de uso termina.

# 6.6 Caso de Uso: Remover do Carrinho

**Nome:** Remover do Carrinho

**Identificador:** #CU06

**Descrição:** Permitir que o cliente remova cupcakes do carrinho antes da finalização do pedido.

**Pré-condições:**

* O cliente deve estar com itens adicionados ao carrinho.

**Pós-condições:**

* Os itens selecionados são removidos e o total é atualizado.

**Caso básico de ação:**

1. O cliente acessa a TELA002 - Carrinho de Compras;
2. O cliente clica em "Remover" no item desejado;
3. O sistema atualiza o carrinho excluindo o item;
4. O total do pedido é recalculado;
5. O sistema exibe o carrinho atualizado;
6. O caso de uso termina.

**Curso Alternativo Alfa: Carrinho vazio**

* **Alfa3:** O carrinho fica sem itens após a remoção;
* **Alfa4:** O sistema exibe mensagem "Carrinho vazio";
* **Alfa5:** O caso de uso termina.

# 6.7 Caso de Uso: Realizar Pagamento com Cartão

**Nome:** Realizar Pagamento com Cartão

**Identificador:** #CU07

**Descrição:** Permitir que o cliente pague seu pedido com cartão de crédito de forma segura.

**Pré-condições:**

* O cliente deve ter itens no carrinho e endereço de entrega válido.

**Pós-condições:**

* O pagamento é aprovado e o pedido é confirmado.

**Caso básico de ação:**

1. O cliente acessa a TELA004 - Pagamento;
2. Informa os dados do cartão: número, validade, CVV;
3. O sistema valida os dados com a API do provedor de pagamento (RN003 - Validar transação);
4. O sistema aprova a compra e registra o pagamento;
5. O cliente é redirecionado para a confirmação do pedido;
6. O caso de uso termina.

**Curso Alternativo Alfa: Cartão recusado**

* **Alfa3:** O provedor recusa a transação;
* **Alfa4:** O sistema exibe mensagem de falha e solicita nova tentativa;
* **Alfa5:** O caso de uso retorna à etapa 2.

# 6.8 Caso de Uso: Acompanhar Entrega

**Nome:** Acompanhar Entrega

**Identificador:** #CU08

**Descrição:** Permitir que o cliente acompanhe o status da entrega do pedido em tempo real.

**Pré-condições:**

* O pedido deve estar em andamento.

**Pós-condições:**

* O cliente visualiza o status atualizado.

**Caso básico de ação:**

1. O cliente acessa a TELA007 - Rastreamento;
2. O sistema exibe o status atual do pedido (em preparo, a caminho, entregue);
3. O status é atualizado automaticamente conforme ações do entregador ou administrador;
4. O cliente recebe notificações push a cada atualização;
5. O caso de uso termina.

**Curso Alternativo Alfa: Pedido ainda não enviado**

* **Alfa3:** O sistema informa que o pedido ainda está em preparo;
* **Alfa4:** O cliente é orientado a aguardar;
* **Alfa5:** O caso de uso termina.

# 6.9 Caso de Uso: Cadastrar Cupcake (Admin)

**Nome:** Cadastrar Cupcake (Admin)

**Identificador:** #CU09

**Descrição:** Permitir que o administrador registre um novo cupcake no catálogo com nome, descrição, preço e imagem.

**Pré-condições:**

* O administrador deve estar autenticado.

**Pós-condições:**

* O cupcake é salvo no sistema e exibido na vitrine.

**Caso básico de ação:**

1. O administrador acessa a TELA011 - Cadastro de Produto;
2. Preenche nome, descrição, valor e anexa imagem do cupcake;
3. O sistema valida os campos obrigatórios;
4. O produto é salvo na base de dados e aparece no catálogo;
5. O caso de uso termina.

**Curso Alternativo Alfa: Dados incompletos**

* **Alfa3:** O sistema detecta campos em branco;
* **Alfa4:** Exibe mensagem de erro e impede o envio;
* **Alfa5:** caso de uso retorna à etapa 2.

# 6.10 Caso de Uso: Visualizar Pedidos (Admin)

**Nome:** Visualizar Pedidos (Admin)

**Identificador:** #CU10

**Descrição:** Permitir que o administrador visualize todos os pedidos feitos pelos clientes.

**Pré-condições:**

* O administrador deve estar autenticado no sistema.

**Pós-condições:**

* O administrador acessa a lista de pedidos com os detalhes relevantes.

**Caso básico de ação:**

1. O administrador acessa a TELA012 - Lista de Pedidos;
2. O sistema exibe os pedidos organizados por status e data;
3. O administrador pode clicar em um pedido para visualizar detalhes;
4. O sistema mostra os itens do pedido, dados do cliente e endereço;
5. O caso de uso termina.

**Curso Alternativo Alfa: Nenhum pedido disponível**

* **Alfa3:** O sistema informa que não há pedidos no momento;
* **Alfa4:** O administrador é redirecionado para o painel principal;
* **Alfa5:** O caso de uso termina.

# 6.11 Caso de Uso: Alterar Status do Pedido (Admin)

**Nome:** Alterar Status do Pedido (Admin)

**Identificador:** #CU11

**Descrição:** Permitir que o administrador altere o status de um pedido, conforme seu andamento.

**Pré-condições:**

* O pedido deve estar registrado e não finalizado.

**Pós-condições:**

* O status do pedido é atualizado e registrado no sistema.

**Caso básico de ação:**

1. O administrador acessa a TELA012 - Lista de Pedidos;
2. Seleciona o pedido desejado;
3. Escolhe o novo status: "Em preparo", "A caminho" ou "Entregue";
4. O sistema registra a alteração e atualiza a interface do cliente;
5. O sistema envia uma notificação ao cliente sobre a atualização;
6. O caso de uso termina.

**Curso Alternativo Alfa: Status inválido**

* **Alfa3:** O sistema não reconhece o novo status selecionado;
* **Alfa4:** A alteração é bloqueada e o administrador é notificado;
* **Alfa5:** O caso de uso retorna à etapa 3.

# 6.12 Caso de Uso: Recuperar Senha

**Nome:** Recuperar Senha

**Identificador:** #CU12

**Descrição:** Permitir que o cliente redefina sua senha por meio de link enviado por e-mail.

**Pré-condições:**

* O cliente deve ter um e-mail cadastrado no sistema.

**Pós-condições:**

* A senha é atualizada e o cliente pode acessar o sistema.

**Caso básico de ação:**

1. O cliente acessa a TELA008 - Esqueci minha senha;
2. Informa o e-mail de cadastro;
3. O sistema envia um link de redefinição de senha ao e-mail fornecido;
4. O cliente acessa o link e digita uma nova senha;
5. O sistema salva a nova senha e exibe confirmação;
6. O caso de uso termina.

**Curso Alternativo Alfa: E-mail não encontrado**

* **Alfa3:** O sistema informa que o e-mail não está cadastrado;
* **Alfa4:** O cliente é orientado a verificar ou se cadastrar;
* **Alfa5:** O caso de uso termina.

# 6.13 Caso de Uso: Filtrar Produtos por Sabor

**Nome:** Filtrar Produtos por Sabor

**Identificador:** #CU13

**Descrição:** Permitir que o cliente aplique filtros por tipo de sabor ao navegar no catálogo.

**Pré-condições:**

* O catálogo deve conter produtos cadastrados com categorias de sabor.

**Pós-condições:**

* O sistema exibe apenas os produtos que se encaixam no filtro.

**Caso básico de ação:**

1. O cliente acessa a TELA001 - Vitrine de Produtos;
2. Clica no botão de filtros e escolhe a categoria (ex: chocolate, morango);
3. O sistema aplica o filtro e exibe os resultados correspondentes;
4. O cliente navega nos produtos filtrados;
5. O caso de uso termina.

**Curso Alternativo Alfa: Nenhum resultado encontrado**

* **Alfa3:** O sistema exibe mensagem "Nenhum produto encontrado para esse sabor";
* **Alfa4:** O cliente pode remover o filtro para ver todos os produtos;
* **Alfa5:** O caso de uso termina.

# 6.14 Caso de Uso: Notificação de Confirmação de Pedido

**Nome:** Notificação de Confirmação de Pedido

**Identificador:** #CU14

**Descrição:** Permitir que o sistema envie uma notificação ao cliente após confirmação do pagamento.

**Pré-condições:**

* O pagamento deve ser aprovado.

**Pós-condições:**

* O cliente é notificado com os dados do pedido.

**Caso básico de ação:**

1. sistema detecta pagamento aprovado;
2. Gera notificação com número e resumo do pedido;
3. Envia a notificação via push no aplicativo;
4. cliente visualiza confirmação na TELA005 - Confirmação;
5. caso de uso termina.

**Curso Alternativo Alfa: Erro no envio de notificação**

* **Alfa3:** O sistema falha ao comunicar a notificação;
* **Alfa4:** A mensagem é registrada no histórico do usuário como fallback;
* **Alfa5:** O caso de uso termina.

# 6.15 Caso de Uso: Notificação de Entrega a Caminho

**Nome:** Notificação de Entrega a Caminho

**Identificador:** #CU15

**Descrição:** Permitir que o sistema notifique o cliente quando o pedido estiver em rota de entrega.

**Pré-condições:**

* O status do pedido deve ter sido atualizado para "A caminho".

**Pós-condições:**

* O cliente é alertado sobre a entrega iminente.

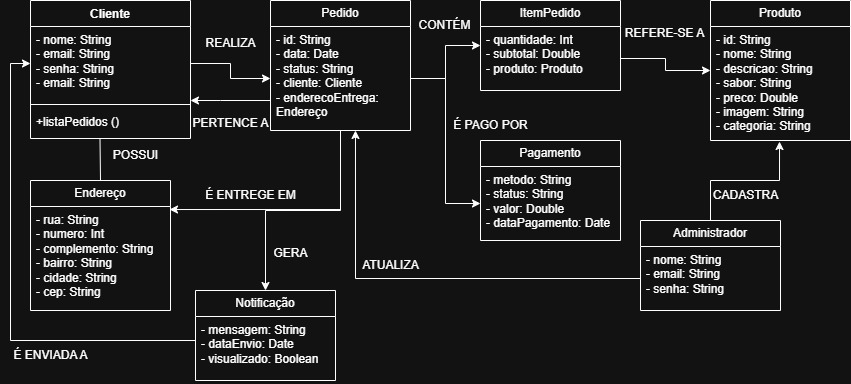
**Caso básico de ação:**

1. O administrador altera o status do pedido para "A caminho";
2. O sistema gera e envia notificação push ao cliente;
3. O cliente recebe e visualiza o alerta;
4. O sistema atualiza o rastreamento na TELA007 - Rastreamento;
5. O caso de uso termina.

**Curso Alternativo Alfa: Cliente offline**

* **Alfa3:** O cliente está sem conexão e não recebe push;
* **Alfa4:** O sistema registra o alerta para envio posterior;
* **Alfa5:** O caso de uso termina.

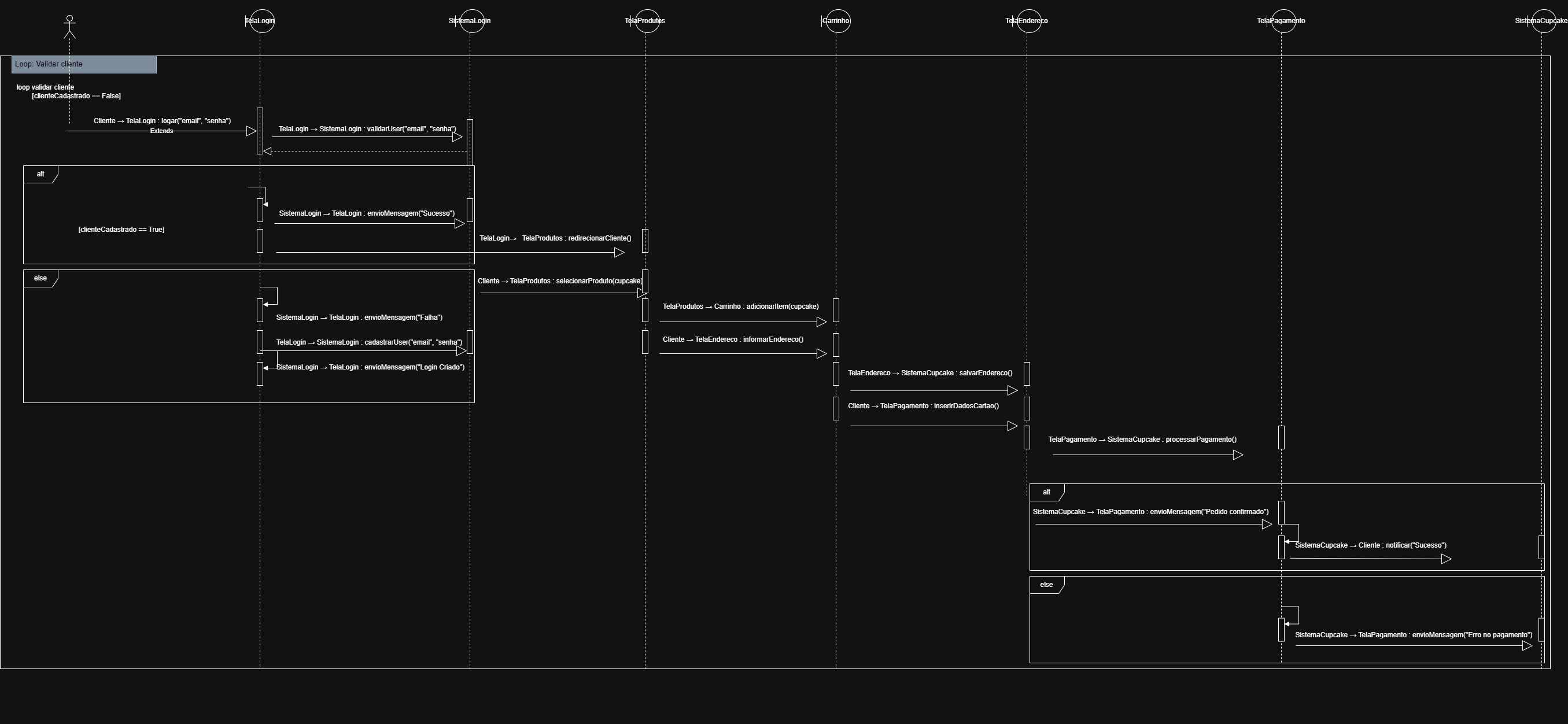
# 7. Diagrama de classes



Entidades principais (relacionamentos em UML):

* Usuário (id, nome, email, senha)
* Produto (id, nome, descrição, preço, estoque)
* Pedido (id, usuário, status, data)
* ItemPedido (pedido\_id, produto\_id, quantidade)
* Endereço (id, usuário, rua, bairro, cidade, cep)
* Pagamento (id, pedido\_id, método, status)

# 8. Diagrama de sequência



Fluxo: Cliente Faz Pedido

Cliente → Catálogo → Carrinho → Endereço → Pagamento → Confirmação

1. Cliente escolhe cupcake no catálogo.
2. Adiciona ao carrinho.
3. Informa endereço de entrega.
4. Escolhe método de pagamento.
5. Sistema valida e registra pedido.
6. Envia confirmação.

# *Situação-Problema 3*

# 9. Wireframes

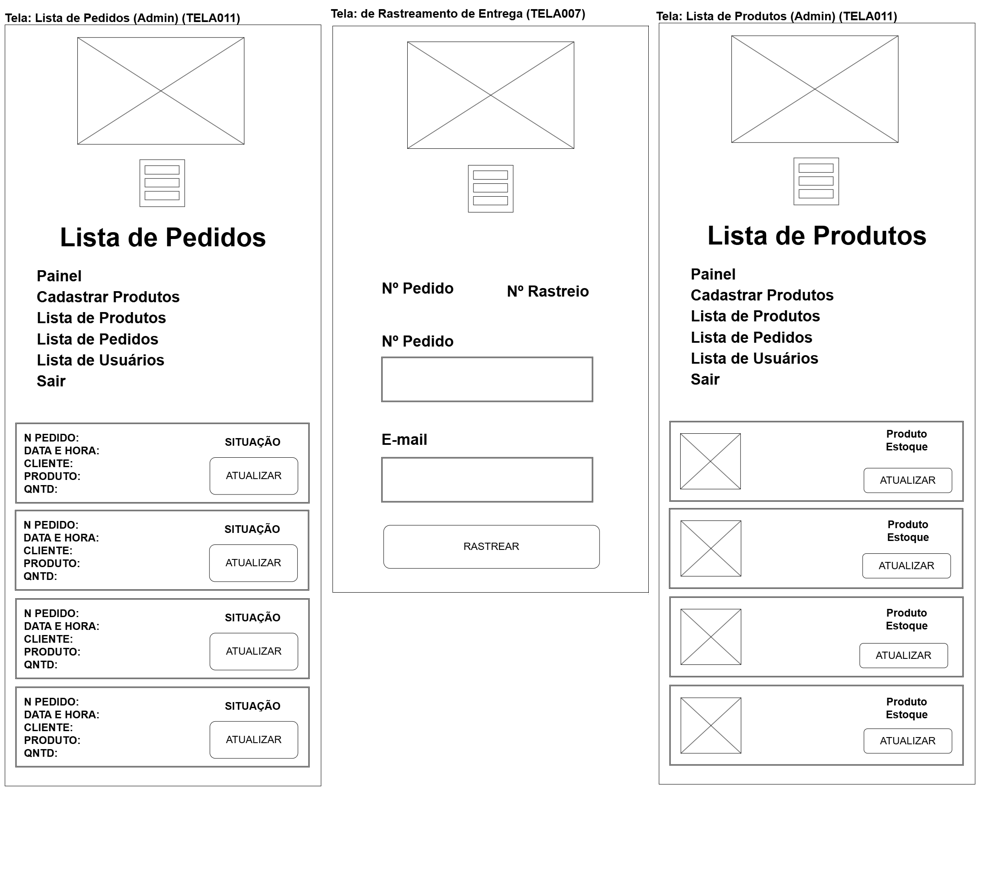
# Diagrama O conteúdo gerado por IA pode estar incorreto.



Diagrama

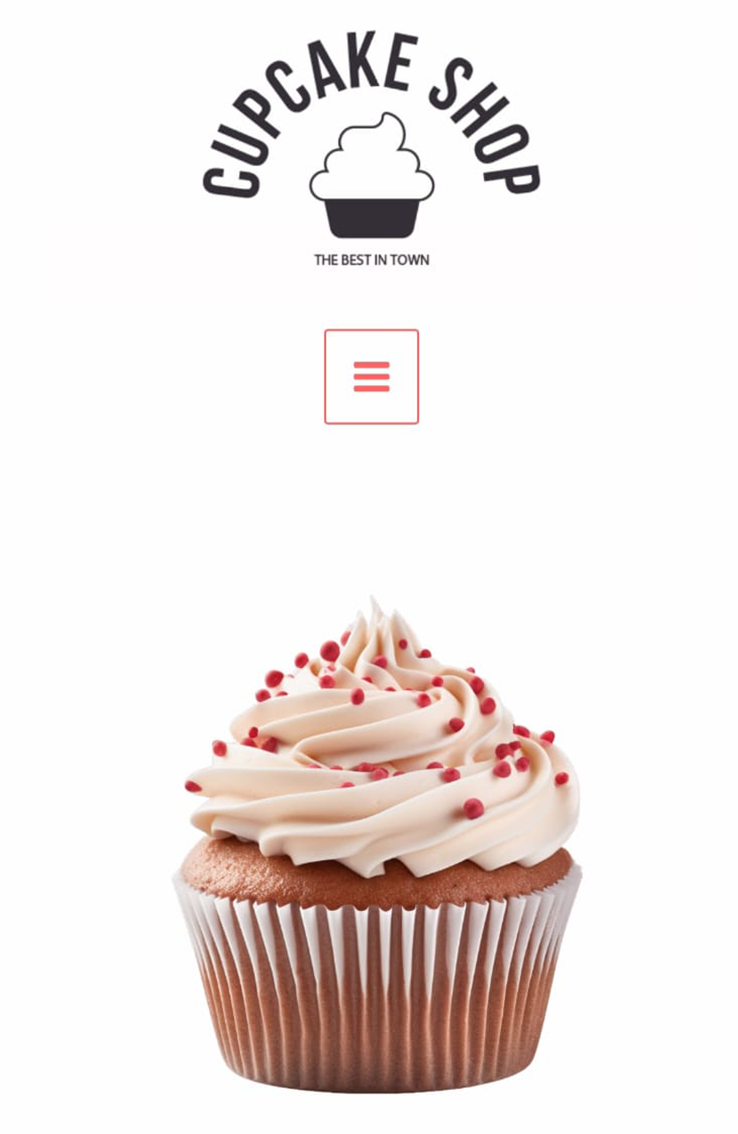
O conteúdo gerado por IA pode estar incorreto.

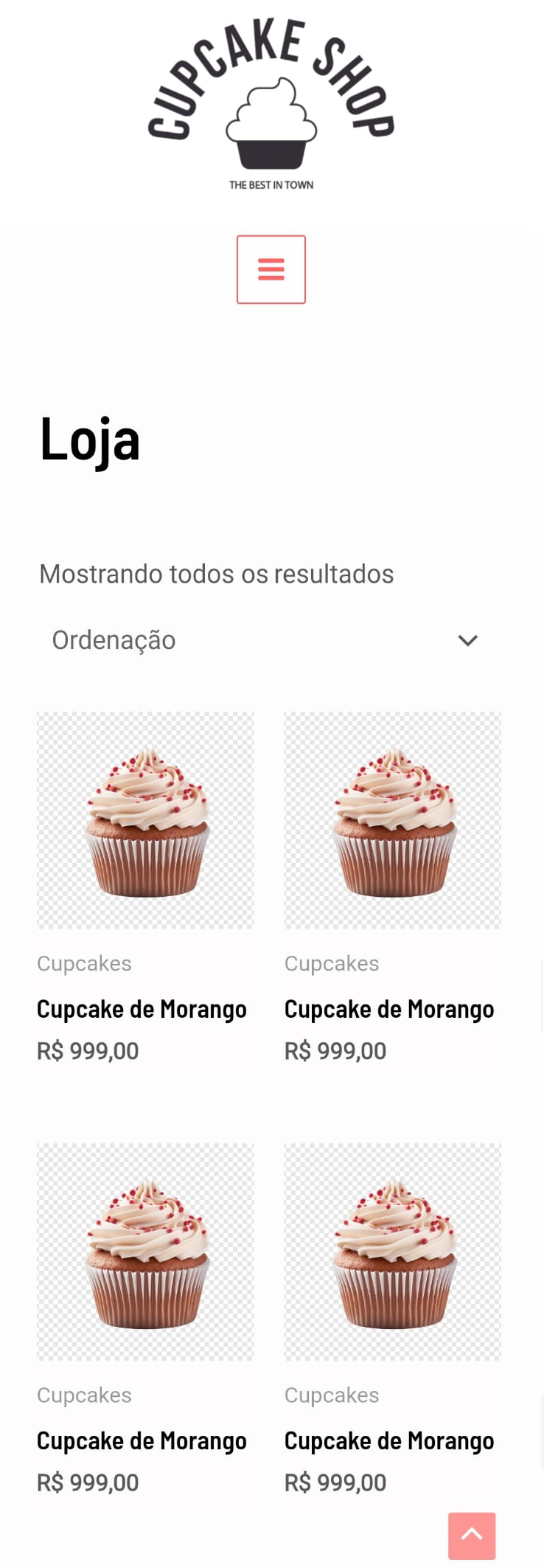


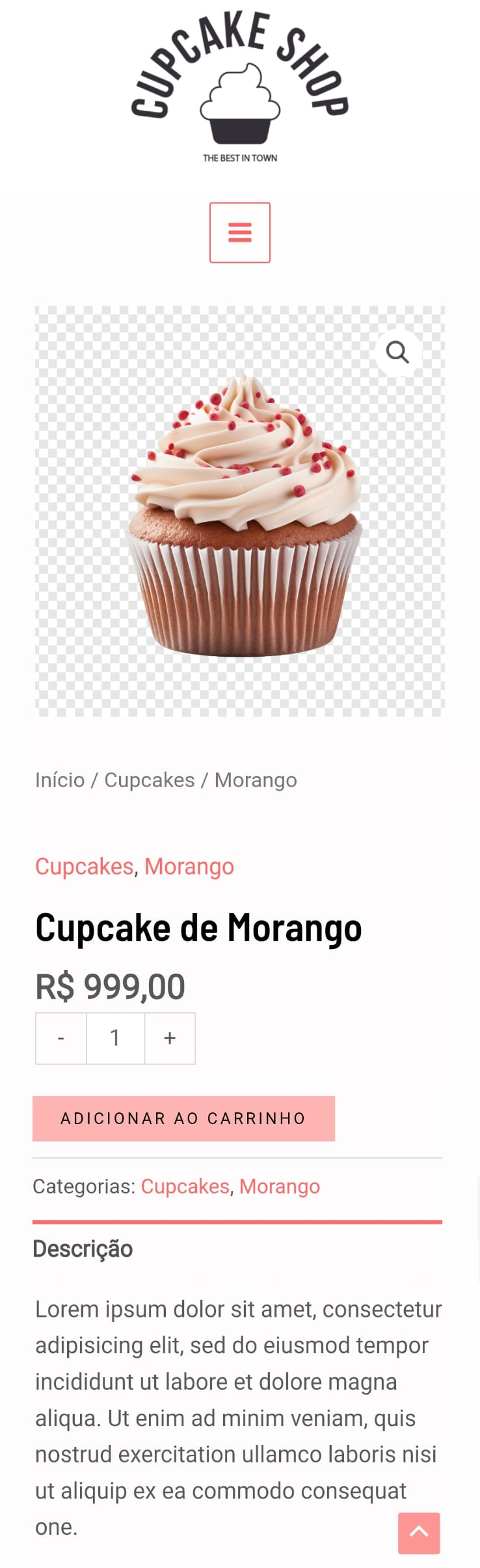


# 10. Mockups

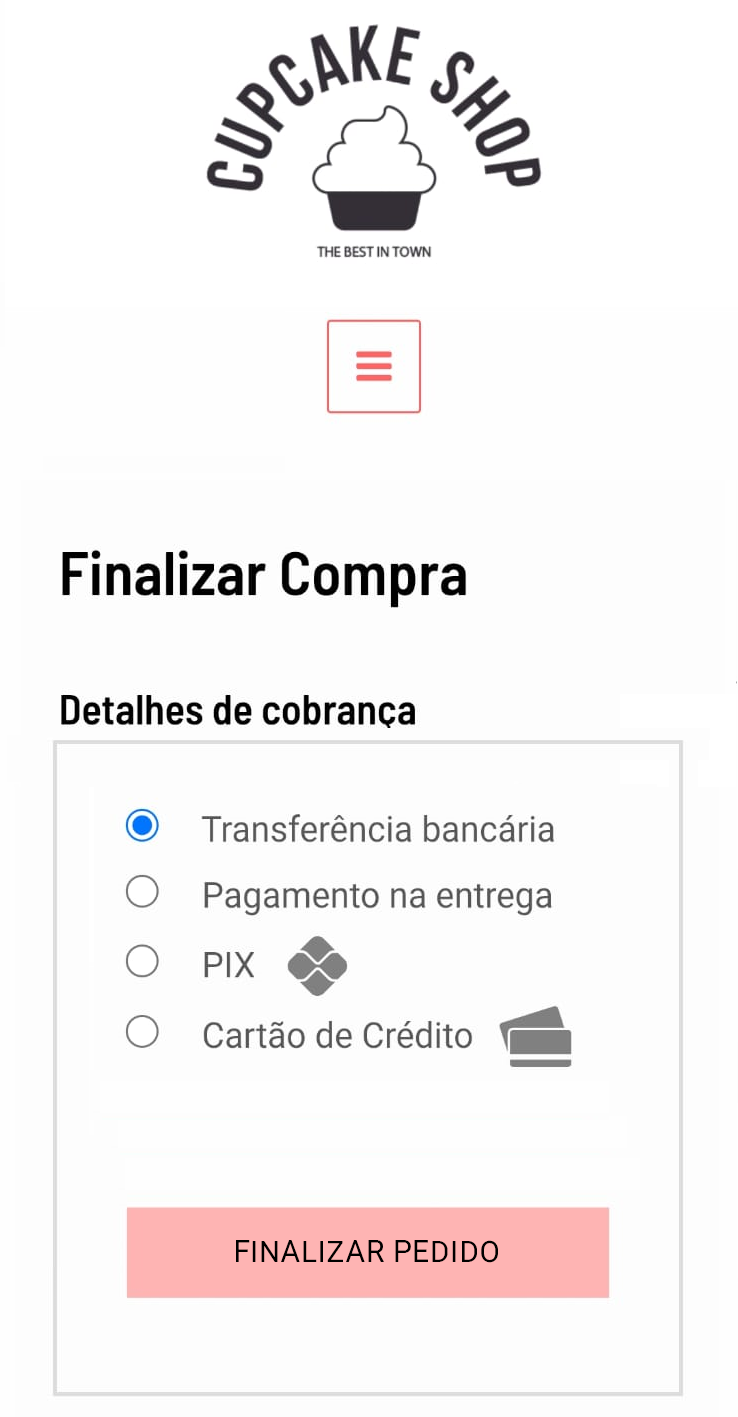
1. Tela de Boas-Vindas 2. Tela de Login (TELA006)

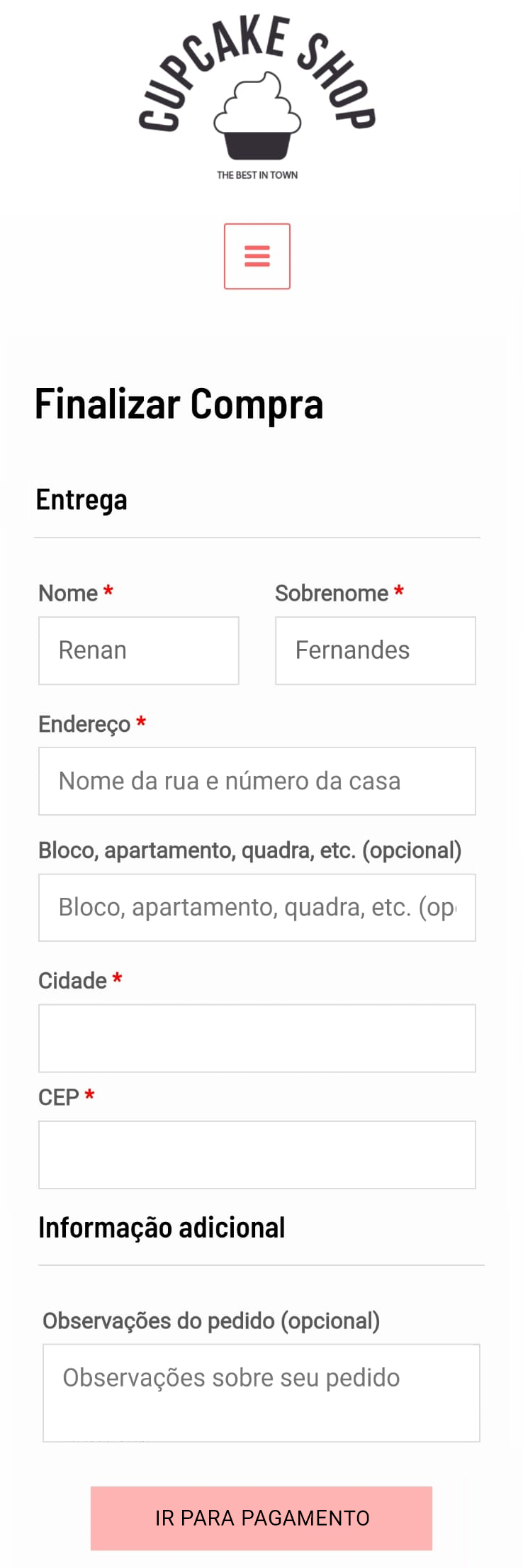


3. Tela: Catálogo de Produtos (Vitrine) (TELA001) . 4. Tela: Detalhe do Produto



5. Tela: Endereço de Entrega (TELA003) 6. Tela: de Pagamento (TELA004)





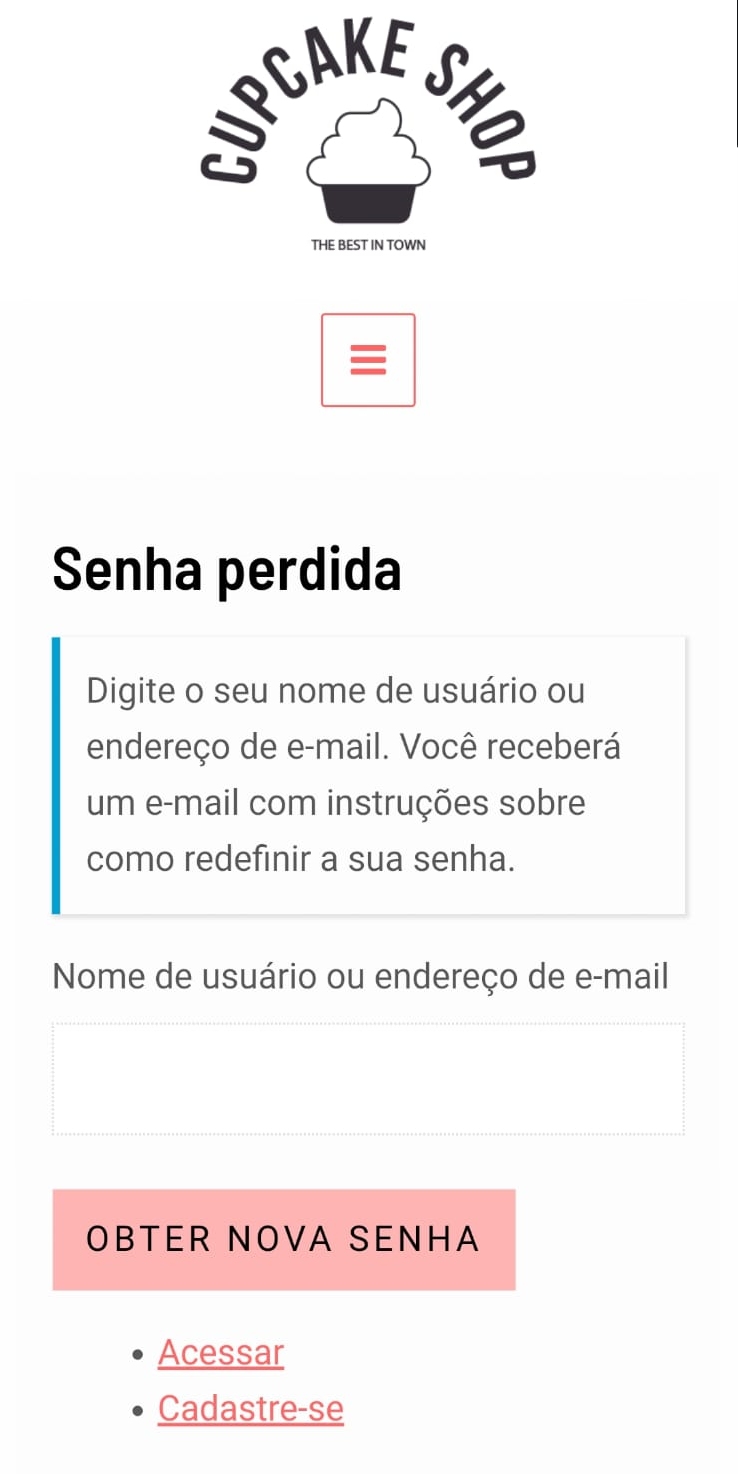
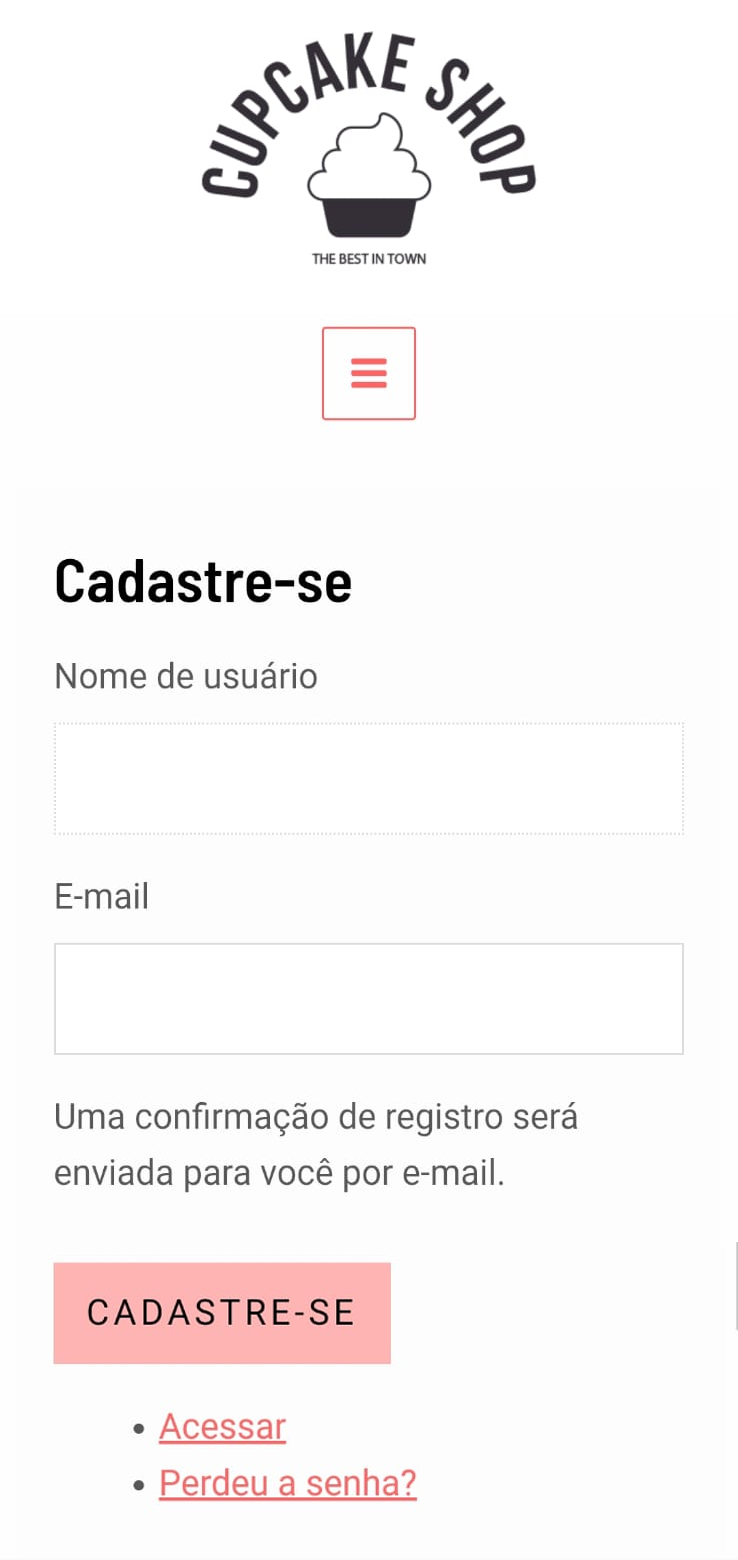
7. Tela: Confirmação do Pedido (TELA005)

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto. 8. Tela: de Rastreamento de Entrega (TELA007)



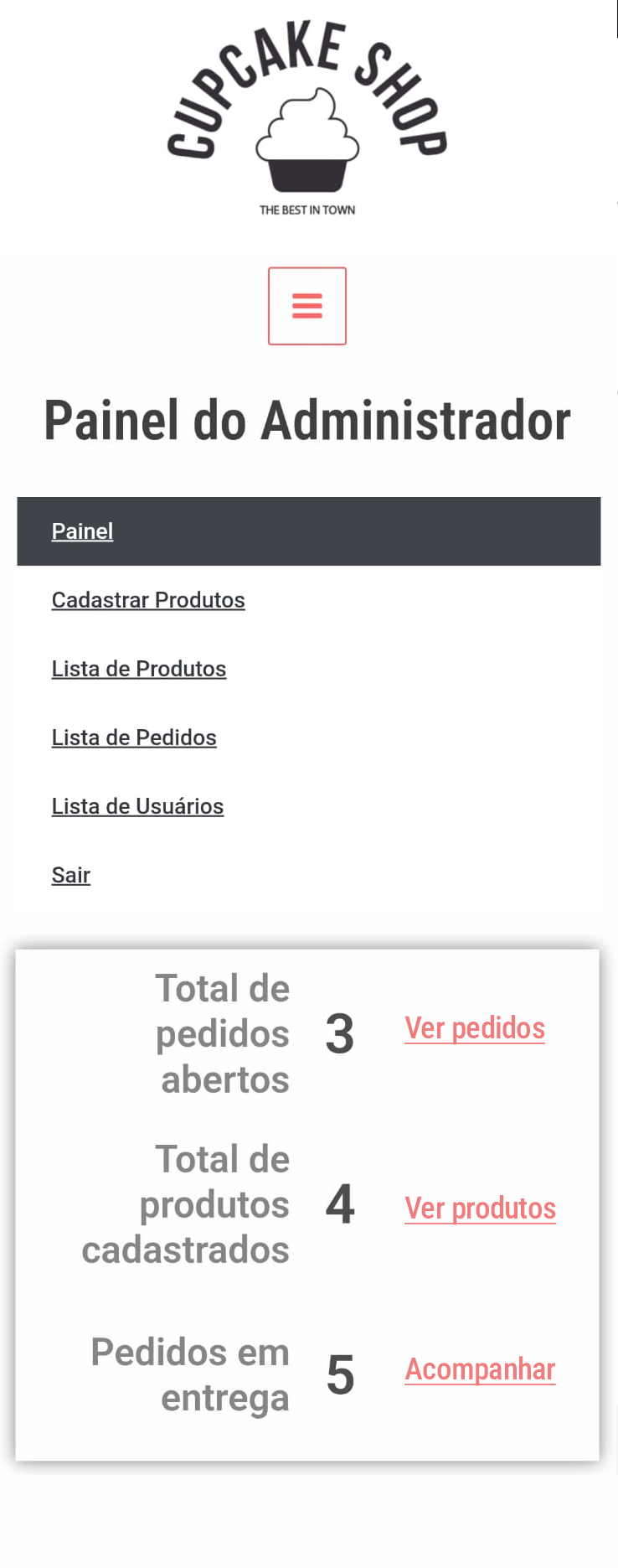
9. Tela: Cadastro de Usuário

 10. Tela: Cadastro de Usuário

11. Tela: Cadastro de Usuário

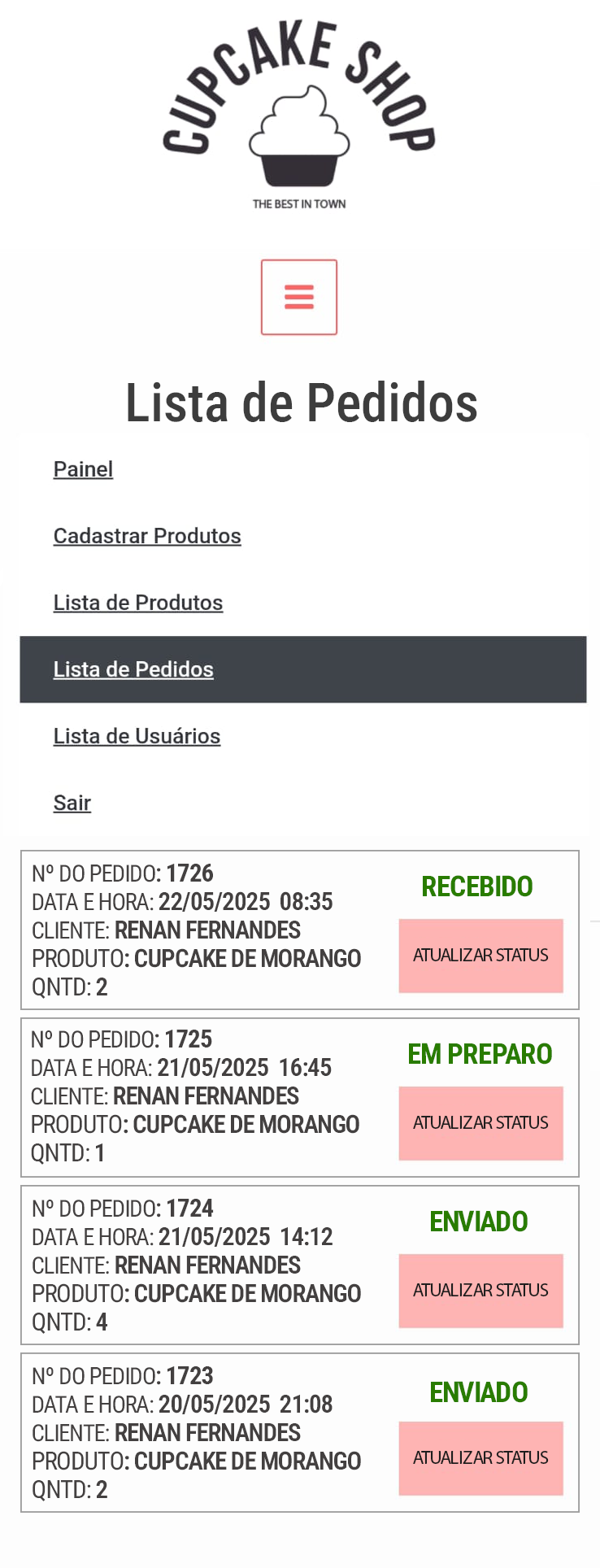


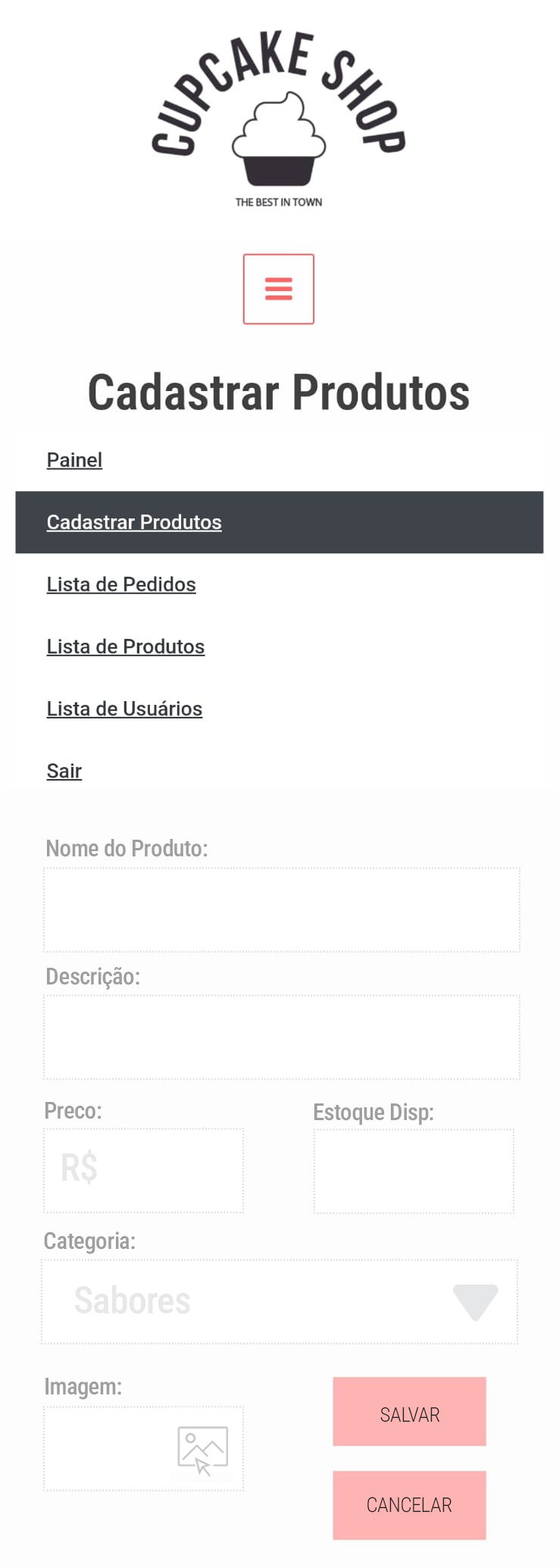
1. Tela: Carrinho de Compras (TELA002)



1. Tela: Carrinho de Compras (TELA002)

14. Tela: Carrinho de Compras (TELA002)

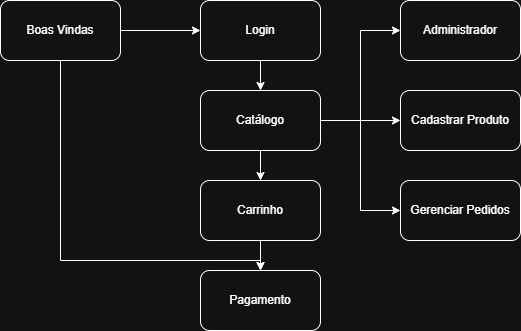




15. Tela: Carrinho de Compras (TELA002)



# 11. Mapa Navegacional



[Boas-vindas]

├─> [Login]

├─> [Cadastro]

└─> [Catálogo]

└─> [Detalhes do Cupcake]

└─> [Carrinho]

└─> [Pagamento]

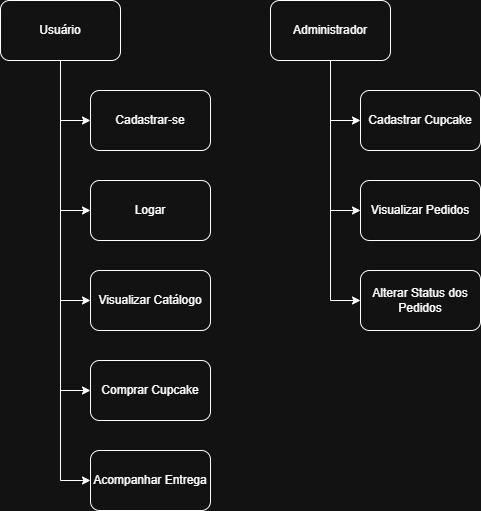
└─> [Rastreamento]

[Admin]

├─> [Cadastrar Produto]

└─> [Gerenciar Pedidos]

# 12. Mapa Conceitual



Usuário:

- Cadastrar-se

- Logar

- Visualizar catálogo

- Comprar cupcake

- Acompanhar entrega

Administrador:

- Cadastrar cupcake

- Visualizar pedidos

- Alterar status dos pedidos

# 13. Conclusão

Este documento descreve a especificação inicial do sistema da loja de cupcakes. Com base nas histórias de usuário, requisitos e priorização, o time pode agora iniciar os sprints ágeis para desenvolver o app de forma incremental.