



Nome: Daniel marques mussinhate

RGM: 23194243i

****Índice****

1. **Introdução**

1.1 Objetivo do Trabalho

2. **Requisitos e Especificações**

2.1 Identificação do Projeto

2.2 Descrição do Aplicativo Web Mobile

2.3 Critérios de Aceitação

2.4 Regras de Negócio

2.5 Requisitos Não Funcionais

2.6 Prioridades e Pontos de História

3. **Histórias do Usuário**

3.1 História do Usuário 1 - Compra de Cupcakes Gourmet Online

3.2 História do Usuário 2 - Pagamento via Aplicativo ou Site

3.3 História do Usuário 3 - Escolha de Cupcakes Gourmet

3.4 História do Usuário 4 - Consulta de Valores dos Cupcakes

3.5 História do Usuário 5 - Atendimento Online

3.6 História do Usuário 6 - Opção de Cupcakes Gourmet Diet

3.7 História do Usuário 7 - Adição de Itens no Cupcake Gourmet

3.8 História do Usuário 8 - Retirada de Pedido no Local

3.9 História do Usuário 9 - Atendimento Adaptado a Necessidades Especiais

3.10 História do Usuário 10 - Cashback para Desconto de Fidelidade

3.11 História do Usuário 11 - Cadastro de Cliente com Login

3.12 História do Usuário 12 - Análise do Perfil do Cliente

3.13 História do Usuário 13 - Informações de Contato no Cadastro

- 3.14 História do Usuário 14 - Oferta de Diversas Opções de Cupcakes
- 3.15 História do Usuário 15 - Pagamentos via Cartão de Crédito ou Pix
- 3.16 História do Usuário 16 - Cashback para Clientes com Fidelidade
- 3.17 História do Usuário 17 - Recebimento de Avaliações no Aplicativo

4. **Análise**

- 4.1 Descrição do Caso de Uso: Compra de Cupcakes Gourmet Online
- 4.2 Definição de Classes de Análise de Candidatos
- 4.3 Filtragem de Classes de Análise
- 4.4 Identificação e Descrição de Responsabilidades e Relações entre Classes
- 4.5 Estudo da Interação entre Objetos de Classe
- 4.6 Identificação de Atributos para as Classes

5. **Backlog do Produto**

- 5.1 Backlog do Produto – Prioridades

6. ** Backend e frontend

. 7**Projeto Físico do Banco de Dados**

- 7.1 *Criação do Banco de Dados "CupcakeGourmet"*
- 7.2 Criação da Tabela "Cliente"
- 7.3 Criação da Tabela "Cupcake"
- 7.4 Criação da Tabela "Pedido"
- 7.5 Criação da Tabela "DetalhesPedido"
- 7.6 Criação da Tabela "ProgramaFidelidade"
- 7.7 Criação da Tabela "ComunicacaoCliente"



ID	23194243
Título:	Aplicativo web mobile
Requerente:	cupcakes gourmet
Ação	Desenvolver Aplicativo web mobile
Comentários:	
Critérios de aceitação: CA#1 recursos disponíveis...	
Regras de negócio: RN #1...	
Requisito não funcional: RN F#1 Serviço acessível pelo Aplicativo WEB Mobile. ...	
Prioridade: <input type="checkbox"/> A <input type="checkbox"/> B <input type="checkbox"/> C	
Pontos de história:	34



ID	História do usuário	Estimativa em pontos	Prioridade
1	Como cliente eu quero comprar cupcakes gourmet via aplicativo ou site para não precisar sair de casa	34	A
2	Como cliente eu quero efetuar o pagamento via aplicativo ou site para agilizar meu tempo	5	A
3	Como cliente eu quero escolher o cupcakes gourmet de minha preferência via aplicativo ou site para atender minhas necessidades	2	A
5	Como cliente eu quero saber os valores dos cupcakes gourmet para que eu possa escolher a melhor opção	5	A
6	Como cliente eu quero um canal de atendimento online para atender minhas dúvidas caso necessário.	5	A
7	Como cliente eu quero opção de cupcakes gourmet diet para atender meus critérios de saúde	1	B
8	Como cliente eu quero adicionar itens no cupcake gourmet para completar meu pedido.	1	B
9	Como cliente eu quero retirar meu pedido onde estiver para que eu não precise me locomover.	2	A



10	Como cliente eu quero se tiver necessidade especiais atendimento adaptado para atender clientes com necessidades especiais.	1	A
11	Como cliente eu quero cachback para ter desconto fidelidade nas próximas compras.	1	A
12	Como proprietário eu quero cadastro do cliente com login para que eu possa identificá-lo.	8	A
13	Como proprietário eu quero uma análise do perfil do cliente para maior segurança na venda.	5	A
14	Como proprietário eu quero que o cliente possua telefone e email para efetuado o cadastro.	2	A
15	Como proprietário eu quero oferecer diversas opções de cupcakes para meu cliente ter diversidades	2	C
16	Como proprietário eu quero pagamentos via cartão de crédito ou pix para que eu possa liberar a compra.	5	A
17	Como proprietário eu quero oferecer cachback para clientes com fidelidade.	1	B
18	Como proprietário eu quero receber avaliações no aplicativo para ter feedback do cliente.	1	A

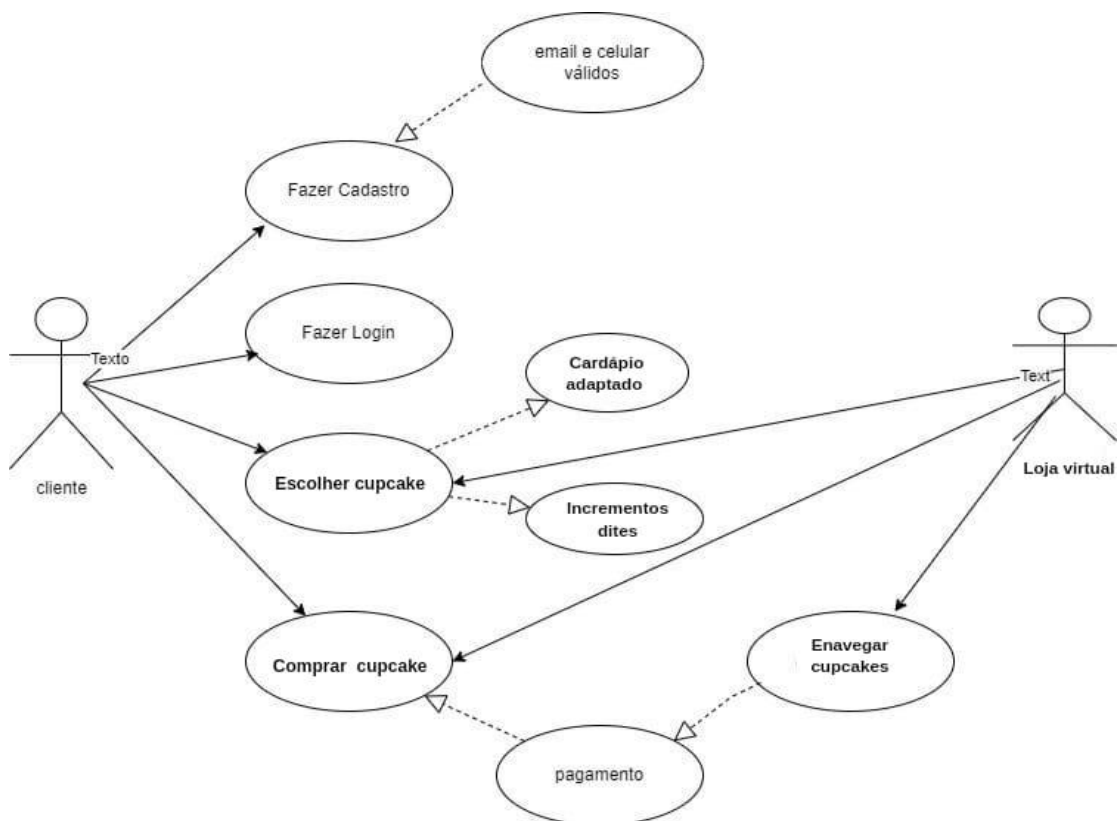


19	Como desenvolvedor eu quero desenvolver sites e aplicativos responsivos para atender as normas padrões.	8	A
20	Como desenvolvedor eu quero desenvolver mockup do site e aplicativo para que a empresa entenda o design a ser desenvolvido.	5	A
21	Como desenvolvedor eu quero construir um protótipo para que o cliente veja como irá funcionar a aplicação.	8	A
22	Como desenvolvedor eu quero fornecer mapas conceituais e navegacionais para maior entendimento de todos os envolvidos.	11	A
23	Como desenvolvedor eu quero controle ágil de projeto para que haja alto compromisso.	8	A
24	Como desenvolvedor eu quero usar linguagem “no code” utilizando Appypie (web, android e IOS) para diminuir tempo de entrega e custo	2	B

1. Análise.



****A. Proveja a descrição do caso de uso:****



****Caso de Uso:**** Compra de Cupcakes Gourmet Online

****Descrição:**** Este caso descreve como um cliente usa o sistema "Cupcake Gourmet" para comprar cupcakes gourmet online através da aplicação web mobile.

****Pré-condições:**** O cliente está conectado ao sistema.

****Pós-condições:**** O cliente realiza uma compra ou não.

****Requisitos Não-Funcionais:**** Serviço acessível pelo Aplicativo WEB Mobile.

****Fluxo Básico de Eventos:****

1. O caso de uso começa quando o operador do cliente acessa a página ou aplicativo de compra de cupcakes.
2. O sistema fornece ao cliente as datas e horários de entrega dos cupcakes, além de campos para preenchimento do número do cliente.
3. O cliente anota datas e horários do pedido, bem como seu número de fidelização, caso o possua.
4. O sistema pergunta ao usuário que tipo de cupcake ele deseja.
5. O sistema exibirá todas as opções de tipos de cupcakes disponíveis no aplicativo selecionado no dia e horário escolhido, mostrando o preço básico na compra, que pode ser reduzido de acordo com o perfil de fidelização do cliente.
6. O cliente escolhe o tipo de cupcake a ser comprado.
7. O sistema solicita ao cliente os dados de identificação (nome completo, telefone, e-mail, bandeira do cartão de crédito, data de validade, etc.), utilizando informações de perfil existentes, se disponíveis.
8. O cliente fornece as informações de identificação solicitadas.
9. O sistema apresenta informações sobre opções de incrementos, dietas e saudáveis com os respectivos valores, pedindo ao cliente para aceitar ou rejeitar cada oferta.
10. O cliente anuncia suas escolhas.
11. O sistema pede a confirmação da compra.



12. O cliente aceita a compra.

13. O sistema apresenta uma página com informações sobre o pedido (tipo de cupcake, datas, horários e locais, valor total da compra) e sua confirmação.

****Pontos Importantes:****

- A descrição é independente de design e tecnologia.
- As informações do cliente podem ser pré-preenchidas usando dados do perfil existente.
- As interações são centradas na aplicação web mobile.

****B. Definir classes de análise de candidatos:****

1. Cliente
2. Cupcake
3. Pedido
4. Programa de Fidelidade
5. SMS
6. E-mail

****C. Filtrar as classes de análise:****

- Pode-se reduzir o número de classes candidatas dependendo das necessidades específicas do sistema. No entanto, as classes listadas acima abordam os principais elementos envolvidos no processo de compra de cupcakes online.

****D. Identificar e Descrever as Responsabilidades de Cada Classe e Possíveis Relações Entre Elas:****

1. ****Cliente:****



- Responsabilidades: Fornecer informações de identificação, fazer escolhas de compra.

- Relações: Associação com Pedido, Participação no Programa de Fidelidade.

2. **Cupcake.**

- Responsabilidades: Representar os diferentes tipos de cupcakes disponíveis.

- Relações: Associado a Pedido, Pode pertencer a uma Categoria.

3. **Pedido.**

- Responsabilidades: Armazenar informações sobre a compra.

- Relações: Associado a Cliente, Contém Cupcakes.

4. **Programa de Fidelidade.**

- Responsabilidades: Gerenciar informações de fidelização do cliente.

- Relações: Associado a Cliente.

5. **SMS e E-mail.**

- Responsabilidades: Enviar confirmações de compra.

- Relações: Associados a Cliente.

E. Estudar a Interação Entre Objetos de Classe.

- O Cliente interage com a Classe Cupcake ao fazer escolhas.

- O Pedido é criado para armazenar informações sobre a compra do Cliente.

- O Programa de Fidelidade interage com o Cliente para gerenciar seu status de fidelidade.

- SMS e E-mail interagem com o Cliente para enviar confirmações.



****F. Identificar Possíveis Atributos para as Classes:****

1. ****Cliente:****

- Nome
- Número de Telefone
- Endereço de E-mail
- Número de Fidelização

2. ****Cupcake:****

- Tipo
- Preço
- Informações Nutricionais

3. ****Pedido:****

- Data e Hora do Pedido
- Itens do Pedido
- Valor Total

4. ****Programa de Fidelidade:****

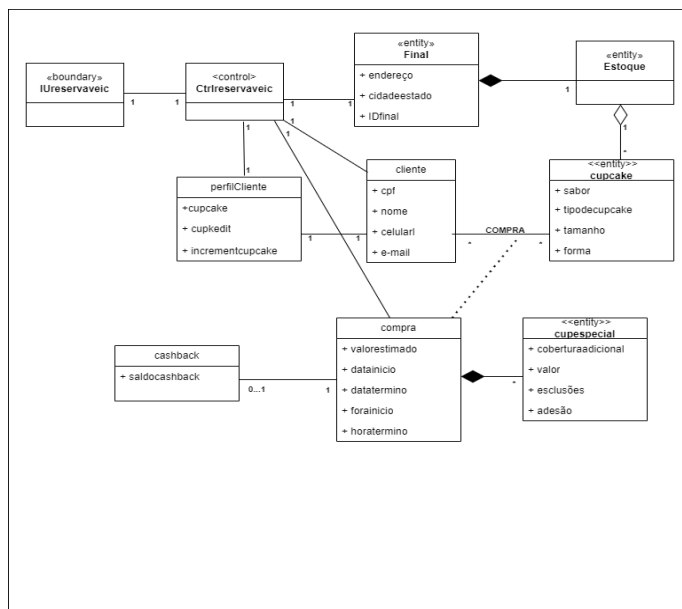
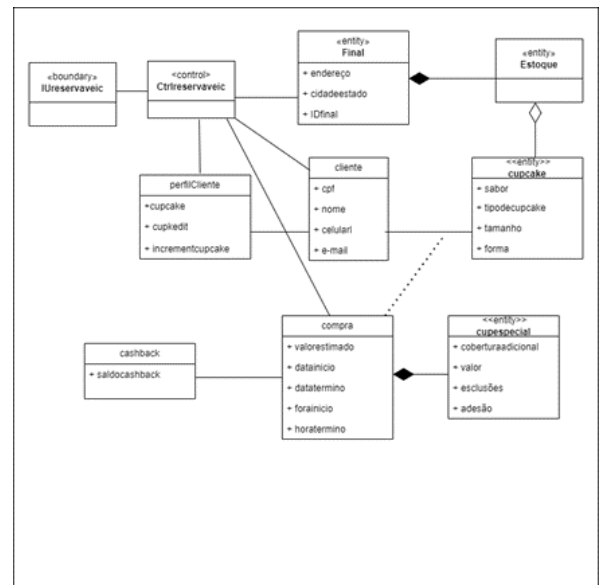
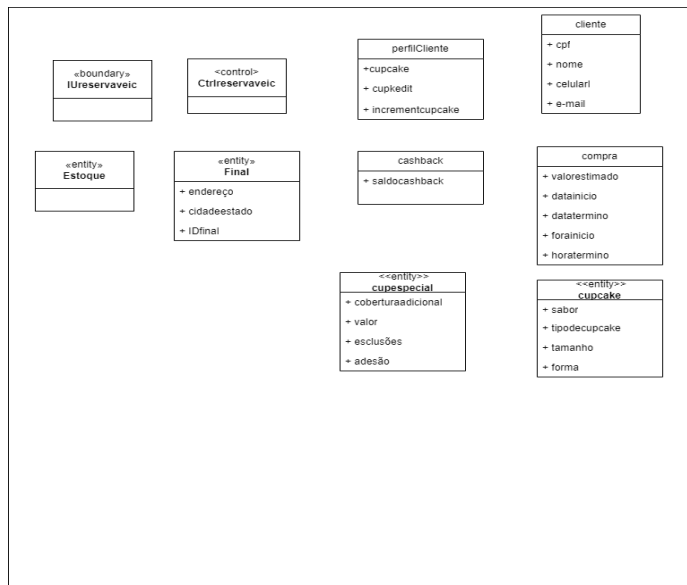
- Status de Fidelidade
- Benefícios

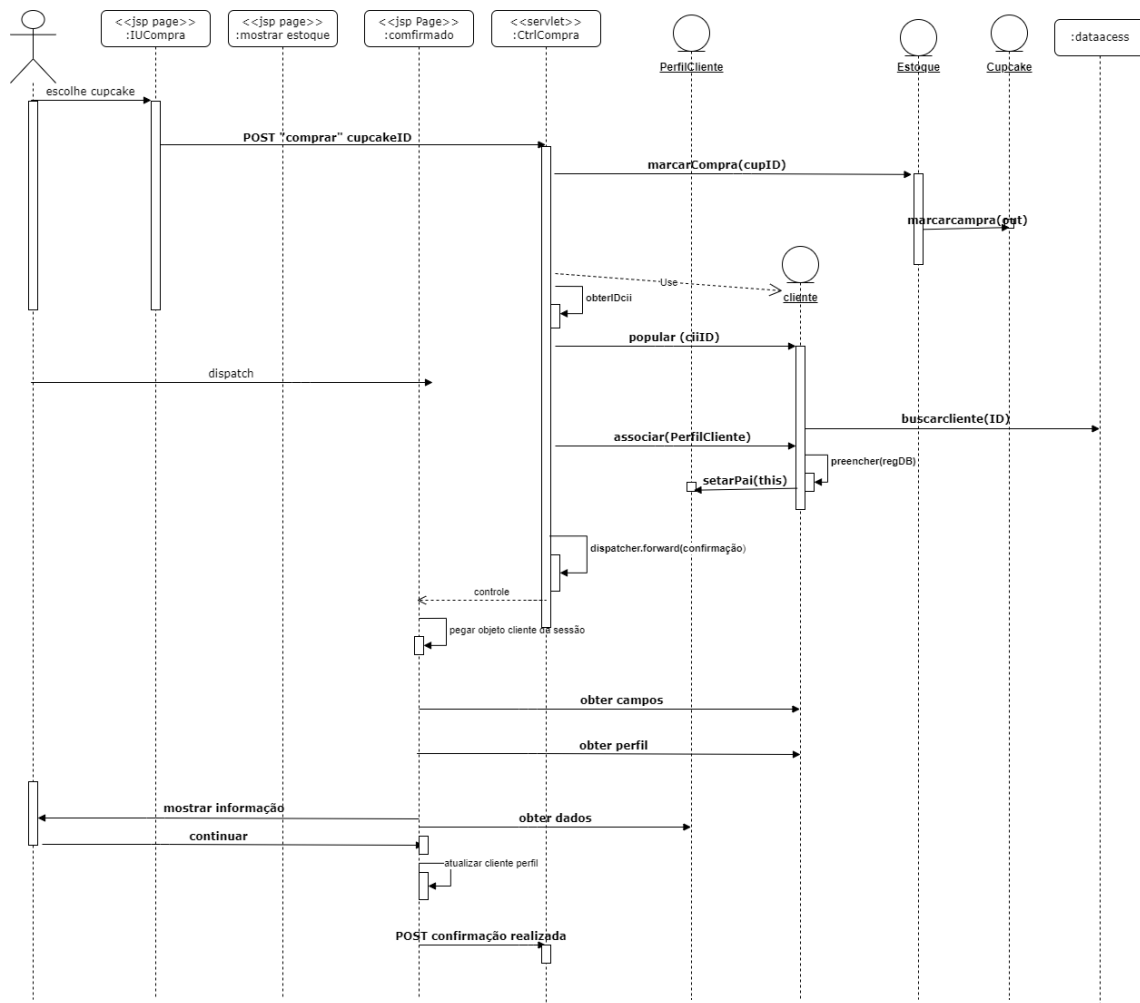
5. ****SMS e E-mail:****

- Mensagem de Confirmação

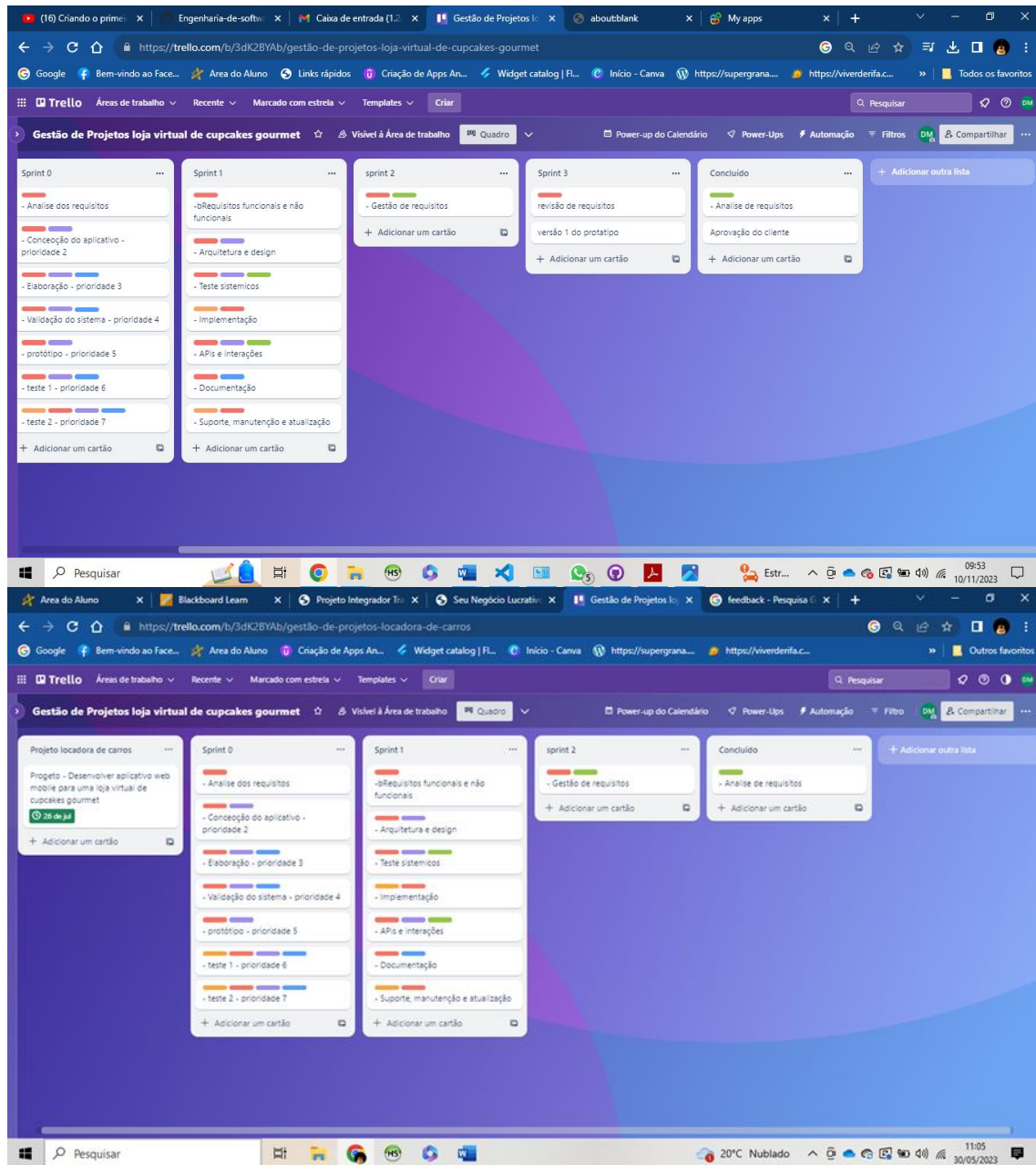
Essas são análises iniciais e podem ser refinadas com base em requisitos específicos e considerações de implementação.

Diagrama de classes



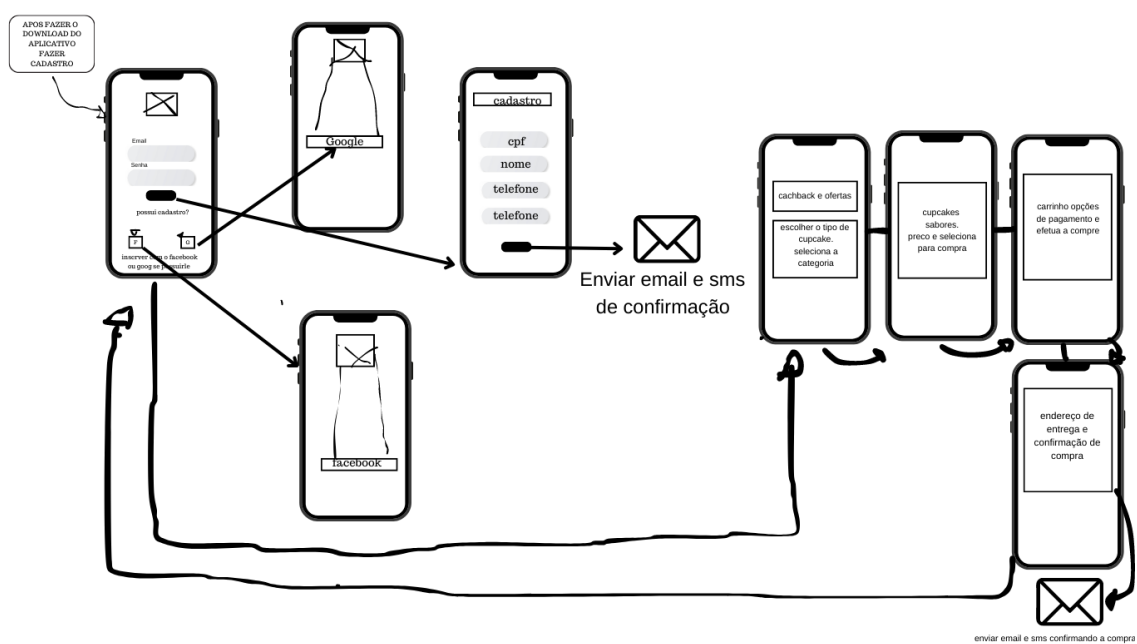


Backlog do produto



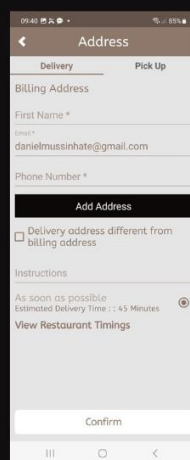
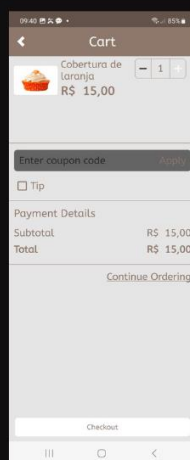
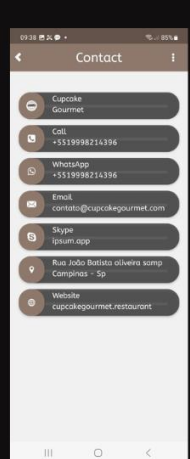
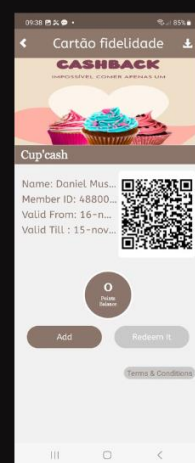
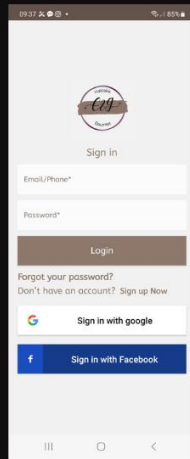
The image displays two screenshots of a Trello project board titled "Gestão de Projetos loja virtual de cupcakes gourmet". The board is organized into columns representing different stages of the project: Sprint 0, Sprint 1, Sprint 2, Sprint 3, and Concluído. Each column contains a list of tasks with progress bars and priority levels. The first screenshot shows the board with tasks for Sprints 0 through 3. The second screenshot shows the board with tasks for Sprints 0 through 2, and a new column for "Projeto locadora de carros" is added on the left. The tasks are as follows:

Sprint	Task	Priority
Sprint 0	Análise dos requisitos	2
Sprint 0	Concepção do aplicativo	2
Sprint 0	Elaboração	3
Sprint 0	Validação do sistema	4
Sprint 0	protótipo	5
Sprint 0	teste 1	6
Sprint 0	teste 2	7
Sprint 1	Requisitos funcionais e não funcionais	2
Sprint 1	Arquitetura e design	2
Sprint 1	Teste sistemáticos	3
Sprint 1	Implementação	4
Sprint 1	APIs e interações	5
Sprint 1	Documentação	6
Sprint 1	Suporte, manutenção e atualização	7
Sprint 2	Gestão de requisitos	2
Sprint 3	revisão de requisitos	2
Sprint 3	versão 1 do protótipo	2
Concluído	Análise de requisitos	2
Concluído	Aprovação do cliente	2





Cruzeiro do Sul Virtual
Educação a distância



Backend:

Otimização de Processos:

Implementar no backend automações e otimizações de processos.

Desenvolvimento de Habilidades da Equipe:

Treinamentos específicos para tecnologias e ferramentas usadas no backend.

Gestão de Recursos:

Análise de recursos e alocação eficiente dos mesmos no backend.

Coleta e Análise de Dados:

Implementação de ferramentas de análise de dados no backend.

Frontend:

Aprimoramento da Experiência do Usuário:

Atualizações na interface do usuário e interações no frontend.

Desenvolvimento de Habilidades da Equipe:

Treinamentos específicos para linguagens e frameworks frontend.

Coleta e Análise de Dados:

Exibição de dados e feedback no frontend para os usuários.

projeto físico do banco



-- Criação do Banco de Dados

CREATE DATABASE IF NOT EXISTS CupcakeGourmet;

-- Seleção do Banco de Dados

USE CupcakeGourmet;

-- Criação da Tabela Cliente

```
CREATE TABLE IF NOT EXISTS Cliente (  
    ClienteID INT AUTO_INCREMENT PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Telefone VARCHAR(15),  
    Email VARCHAR(100),  
    FidelidadeStatus BOOLEAN DEFAULT FALSE  
);
```

-- Criação da Tabela Cupcake

```
CREATE TABLE IF NOT EXISTS Cupcake (  
    CupcakeID INT AUTO_INCREMENT PRIMARY KEY,  
    Nome VARCHAR(100) NOT NULL,  
    Tipo VARCHAR(50),  
    Preco DECIMAL(8, 2) NOT NULL,  
    InformacoesNutricionais TEXT  
);
```

-- Criação da Tabela Pedido

```
CREATE TABLE IF NOT EXISTS Pedido (  
    PedidoID INT AUTO_INCREMENT PRIMARY KEY,
```



```
ClienteID INT,  
DataHoraPedido TIMESTAMP DEFAULT CURRENT_TIMESTAMP,  
ValorTotal DECIMAL(10, 2) NOT NULL,  
FOREIGN KEY (ClienteID) REFERENCES Cliente(ClienteID)  
);
```

-- Criação da Tabela DetalhesPedido para relacionamento muitos-para-muitos entre Pedido e Cupcake

```
CREATE TABLE IF NOT EXISTS DetalhesPedido (  
    DetalhesPedidoID INT AUTO_INCREMENT PRIMARY KEY,  
    PedidoID INT,  
    CupcakeID INT,  
    Quantidade INT NOT NULL,  
    FOREIGN KEY (PedidoID) REFERENCES Pedido(PedidoID),  
    FOREIGN KEY (CupcakeID) REFERENCES Cupcake(CupcakeID)  
);
```

-- Criação da Tabela ProgramaFidelidade

```
CREATE TABLE IF NOT EXISTS ProgramaFidelidade (  
    ProgramaFidelidadeID INT AUTO_INCREMENT PRIMARY KEY,  
    ClienteID INT,  
    PontosFidelidade INT DEFAULT 0,  
    Beneficios VARCHAR(255),  
    FOREIGN KEY (ClienteID) REFERENCES Cliente(ClienteID)  
);
```

-- Criação da Tabela ComunicacaoCliente para armazenar informações de SMS e E-mail



```
CREATE TABLE IF NOT EXISTS ComunicacaoCliente (  
    ComunicacaoID INT AUTO_INCREMENT PRIMARY KEY,  
    ClienteID INT,  
    TipoComunicacao ENUM('SMS', 'E-mail') NOT NULL,  
    Mensagem TEXT NOT NULL,  
    FOREIGN KEY (ClienteID) REFERENCES Cliente(ClienteID)  
);
```