

COOKING DIGITAL

Descrição, especificação, arquitetura e demais tópicos referente a aplicação do Cooking Digital, contemplada com: Página Web; Aplicativo Mobile e compartilhamento do Back End e Banco de Dados.

CONTEXTO



INTRODUÇÃO



Cooking Digital é uma empresa de Software na qual oferece resolução de problemas para restaurantes. O objetivo da empresa é transformar com a tecnologia restaurantes mais inclusivos, acessíveis e modernos.

O objetivo geral deste trabalho é a criação de uma aplicação que ofereça um autoatendimento para restaurantes.

Como objetivos específicos, podemos ressaltar:

- Agilidade, autonomia e modernidade
- Economia com a necessidade de menos garçons
- Acessibilidade para clientes surdos e/ou mudos.

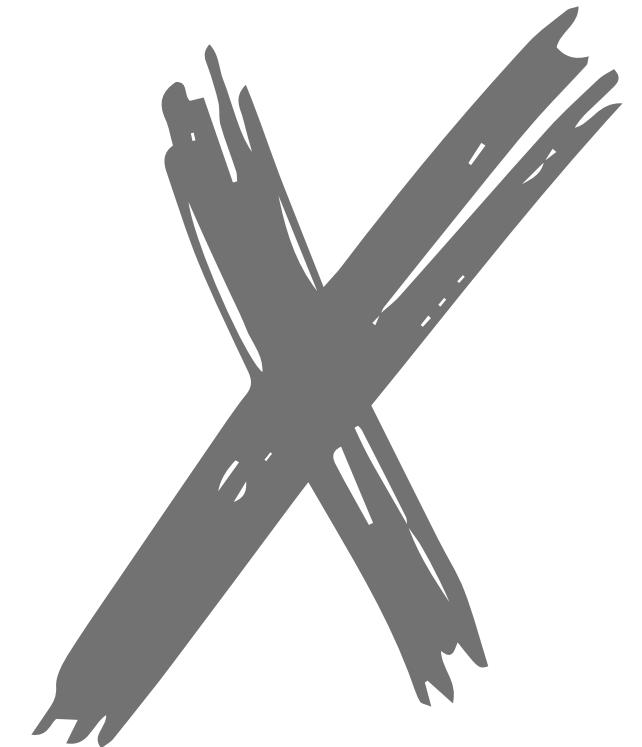


ANÁLISE DA SITUAÇÃO

ATUAL



TEMPO CORRIDO



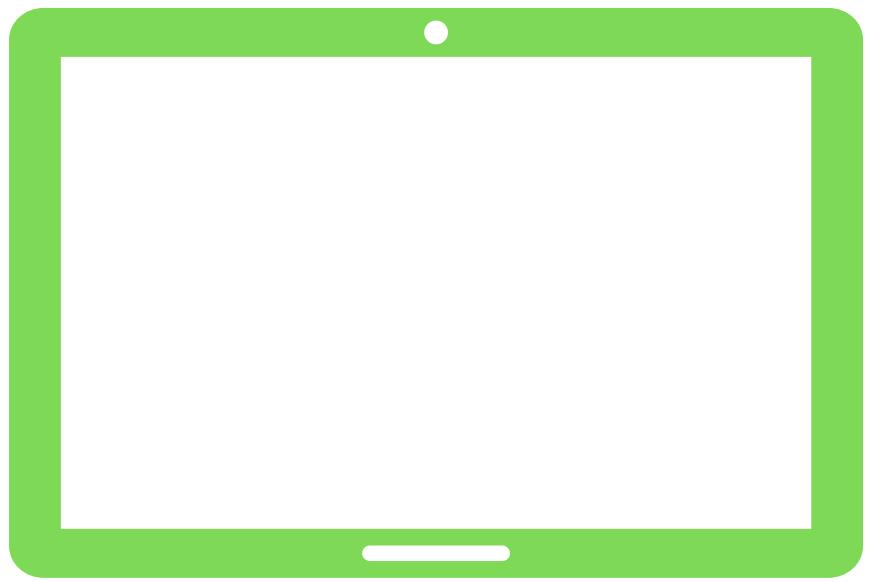
SEM REFEIÇÕES



ESTRESSE

>>>>>

DESCRIÇÃO GERAL DA PROPOSTA



COOKING DIGITAL



REFEIÇÕES



MENOS ESTRESSE

>>>>>

INDICADORES DE DESEMPENHO

RECEITA GERADA

TOTAL DE PEDIDOS

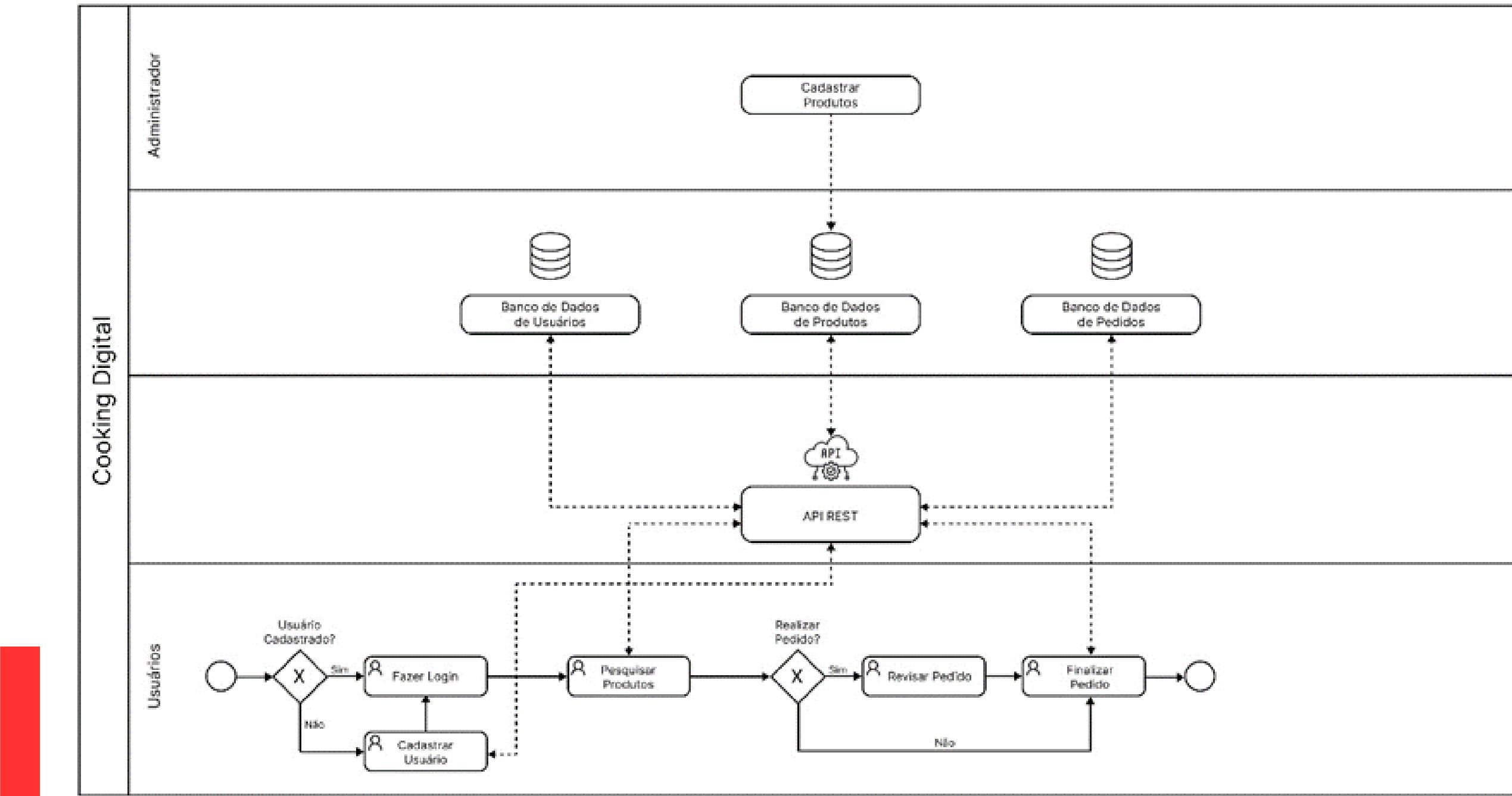
**PEDIDOS POR
PRODUTO**

**TOTAL DE USUÁRIOS
POR GÊNERO**

**MÉDIA DE IDADE
DOS USUÁRIOS POR
GÊNERO**



MODELAGEM DO PROCESSO DE NEGÓCIO



RELACIONAL:

- Banco de Dados Usuário
- SQL Server (A escolha do tipo relacional foi motivada pela consistência e confiabilidade suportada por este modelo)

NÃO RELACIONAL:

- Banco de Dados Produtos e Pedidos.
- MongoDB (A escolha do tipo não relacional foi motivada pela escabilidade suportada por este modelo)



REQUISITOS FUNCIONAIS

ID	Descrição do Requisito	Prioridade
RF-01	A aplicação deverá apresentar na Home Page as informações dos produtos e a navegação para as demais páginas	Média
RF-02	A aplicação deverá permitir ao usuário efetuar login	Alta
RF-03	A aplicação deverá apresentar na página "Meus Pedidos" a funcionalidade de selecionar a quantidade desejada dos produtos	Alta
RF-04	A aplicação deverá permitir na página 'Meus Pedidos' que o usuário selecione mais de um produto	Média
RF-05	A aplicação deverá permitir ao usuário efetuar o cadastro	Alta
RF-06	A aplicação deverá permitir na página de "Meus Pedidos" que o usuário realize o cancelamento de seu pedido em andamento	Alta
RF-07	A aplicação deverá apresentar na página 'Meus Pedidos' as informações e o status dos pedidos realizados pelo usuário	Média
RF-08	A aplicação deverá disponibilizar ao usuário o código do pedido realizado	Alta
RF-09	A aplicação deverá disponibilizar ao usuário as informações do pedido, tais como: produto solicitado, quantidade, preço unitário e valor total	Média
RF-10	A aplicação deverá permitir na página de "Carrinho", que o usuário inclua um pedido no carrinho	Baixa
RF-11	A aplicação deverá permitir na página de "Carrinho", a visualização de todos intens adicionados no carrinho	Média
RF-12	A aplicação deverá permitir na página de "Carrinho", a opção de exclusão de um item do carrinho	Alta
RF-13	A aplicação deverá permitir na página de "Carrinho", a finalização da compra dos itens presentes no carrinho	Alta



REQUISITOS NÃO FUNCIONAIS

ID	Descrição do Requisito	Prioridade
RNF-01	A aplicação deverá ser publicada em um ambiente acessível publicamente na Internet (Repl.it, GitHub Pages, Heroku)	Alta
RNF-02	A aplicação deverá ter bom nível de contraste entre os elementos da tela em conformidade	Alta
RNF-03	A aplicação deverá ser compatível com os principais sistemas operacionais mobile e web do mercado (Android, IOS, Google Chrome, Firefox, Microsoft Edge)	Média
RNF-04	Os bancos de dados serão estruturados utilizando MongoDB	Alta
RNF-05	O sistema deve ser responsivo para dispositivo móvel	Alta
RNF-06	O sistema deve processar as requisições do usuário em no máximo 3 segundos	Alta



METODOLOGIA

- OKR
- Trello
- Metodologia Ágil



CONTROLE DE VERSÃO

A ferramenta de controle de versão adotada no projeto foi o Git, sendo que o Github foi utilizado para hospedagem do repositório.



CONTRIBUIÇÃO TOTAL

CAMILA JARDIM

Testes de Sistemas, Unidade e Integração
⌚ 15 de mai - 4 de jun

Atualização trello com status das contribuições dos membros do time no Github, além dos comentários adicionais sobre as responsabilidades/atribuições de cada membro do time. (Postar no github)
⌚ 15 de mai - 4 de jun

Protótipo interativo das telas com as funcionalidades que fazem parte do sistema (wireframes)
⌚ 15 de mai - 4 de jun

Registro de testes
⌚ 15 de mai - 4 de jun

Projeto de interfaces
⌚ 15 de mai - 4 de jun

Testes de unidade elaborados e documentados.
⌚ 24 de abr - 14 de mai

Definir a visão geral da interação do usuário pelas telas do sistema e protótipo interativo das telas com as funcionalidades que fazem parte do sistema (wireframes)
⌚ 24 de abr - 14 de mai

Modelar processos de negócio no padrão BPMN
⌚ 24 de abr - 14 de mai

Descrever processos e suas respectivas atividades
⌚ 24 de abr - 14 de mai

Especificação do Problema
⌚ 1 de fev - 19 de mar

Controle de mudanças
⌚ 24 de abr - 14 de mai

Modelagem de processos
⌚ 24 de abr - 14 de mai

API - Produtos
⌚ 20 de mar - 23 de abr

Banco de Dados - Cadastro, Usuários e Produtos
⌚ 20 de mar - 23 de abr

API - Cadastro e Usuários
⌚ 20 de mar - 23 de abr

Documento de Contexto
⌚ 1 de fev - 19 de mar

Implementação da API e testes de unidade
⌚ 20 de mar - 23 de abr

Registro de testes
⌚ 24 de abr - 14 de mai

Documentação dos perfis de usuários, suas necessidades e Personas.
⌚ 1 de fev - 19 de mar

RAFAEL MAUTONE

FrontEnd Mobile
⌚ 15 de mai - 4 de jun

Documentação da arquitetura da aplicação distribuída
⌚ 20 de mar - 23 de abr

Front-End - WEB
⌚ 24 de abr - 14 de mai

Projeto de interfaces
⌚ 24 de abr - 14 de mai

Documentação dos requisitos funcionais e não funcionais do projeto de acordo com o contexto levantado
⌚ 1 de fev - 19 de mar

Realização da Apresentação Final
⌚ 5 de jun - 19 de jun

DIEGO ROCHA

Gerenciamento do processo de desenvolvimento de software
⌚ 15 de mai - 4 de jun

Elicitar e categorizar requisitos funcionais e não funcionais derivados dos processos organizacionais
⌚ 15 de mai - 4 de jun

Documentação com justificativa e avaliação do modelo de dados
NoSQL para o contexto da aplicação
⌚ 20 de mar - 23 de abr

Aspectos de usabilidade modelados no Diagrama de Fluxo (fluxograma) para representar as interações entre os usuários e o sistema, e aplicá-las no Wireframe interativo
⌚ 24 de abr - 14 de mai

Documentação da Arquitetura Distribuída
⌚ 1 de fev - 19 de mar

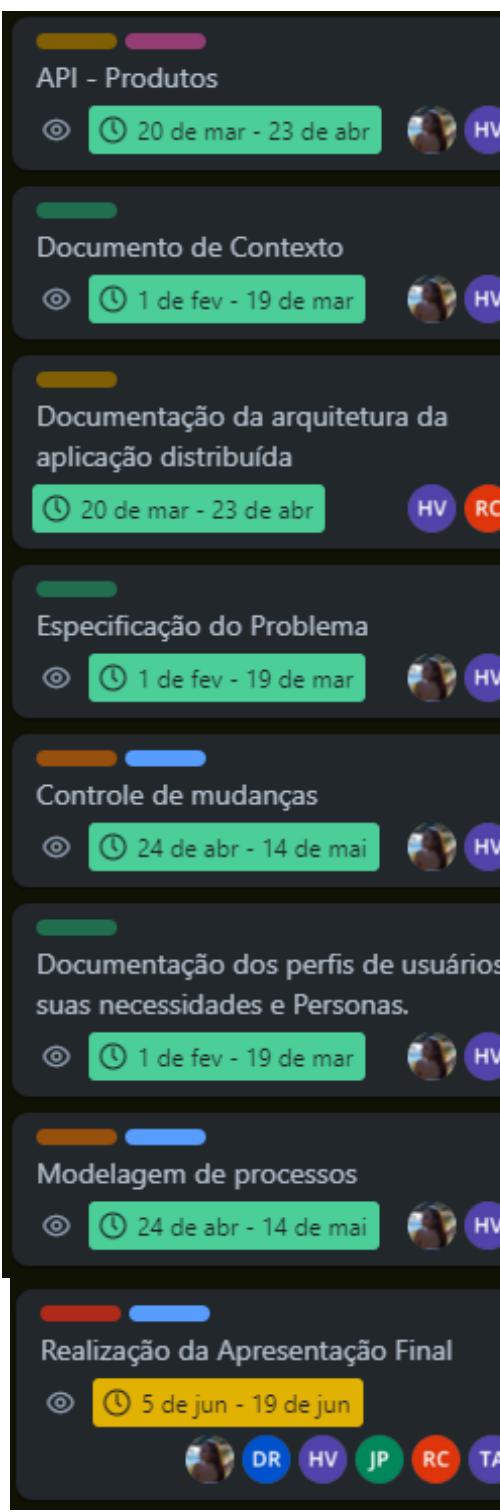
Realização da Apresentação Final
⌚ 5 de jun - 19 de jun

CONTRIBUIÇÃO TOTAL

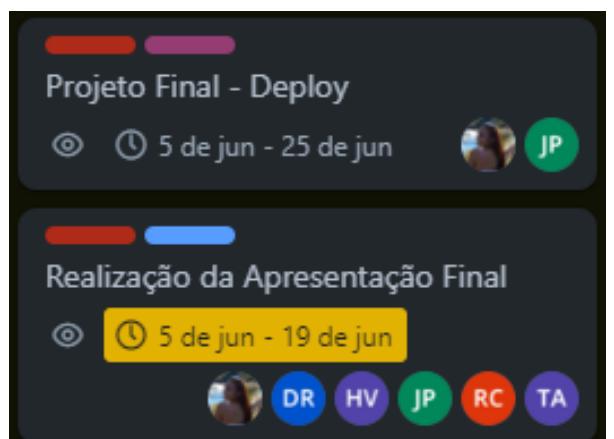
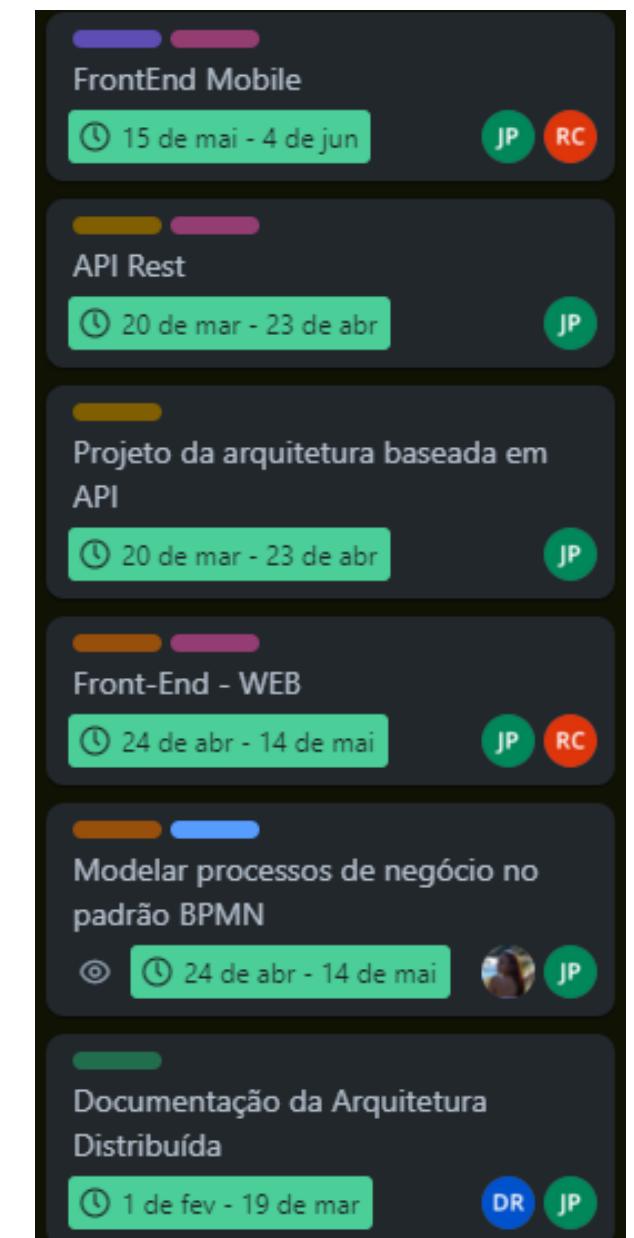
TARCÍSIO ALMEIDA



HUGO VINICIUS



JAQUELINE POLETO



WIREFRAME - WEB



photo

Início

Meus Pedidos



Registre-se gratuitamente.

Nome

Sobrenome

Endereço de E-mail

Criar senha

[Já possui uma conta?](#)

[Registrar](#)



photo

Início

Meus Pedidos



Login

Endereço de E-mail

Senha

[Entrar](#)

WIREFRAME - WEB

photo Início Meus Pedidos Pesquisador Buscar SAIR

Lanches



Descrição do produto

Adicionar



Descrição do produto

Adicionar



Descrição do produto

Adicionar

Bebidas



Descrição do produto

Adicionar



Descrição do produto

Adicionar



Descrição do produto

Adicionar

photo Início Meus Pedidos 

Meus Pedidos

#	Data	Fila	Status	
104245576	14/03/2023 19:57:23	4	Em andamento	<button>Cancelar</button>
416943670	13/02/2023 21:34:56	8	Concluído	
14956023	25/09/2022 14:28:33	2	Concluído	



WIREFRAME - WEB

photo

Início Meus Pedidos

Carrinho

Produto	Qtdd	Preço
 Hamburger - Rio de Janeiro	- 1 +	1 x 22,90

[Limpar carrinho](#) [Finalizar pedido](#)

photo

Início Meus Pedidos

Carrinho

Carrinho Vazio

[+ Adicionar itens](#)

photo

Início Meus Pedidos

Carrinho

Produto	Qtdd	Preço
 Hamburger - Rio de Janeiro	- 1 +	1 x 22,90

[Cancelar pedido](#)
02:59

photo

Início Meus Pedidos

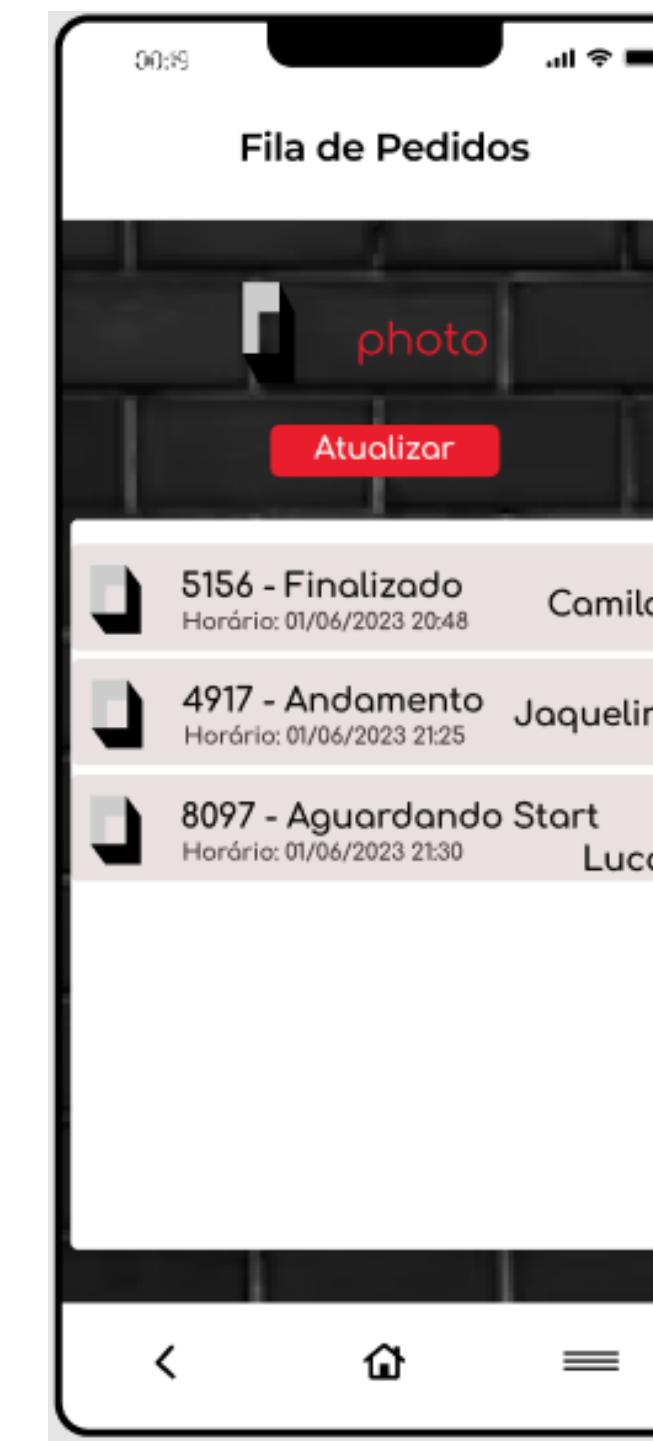
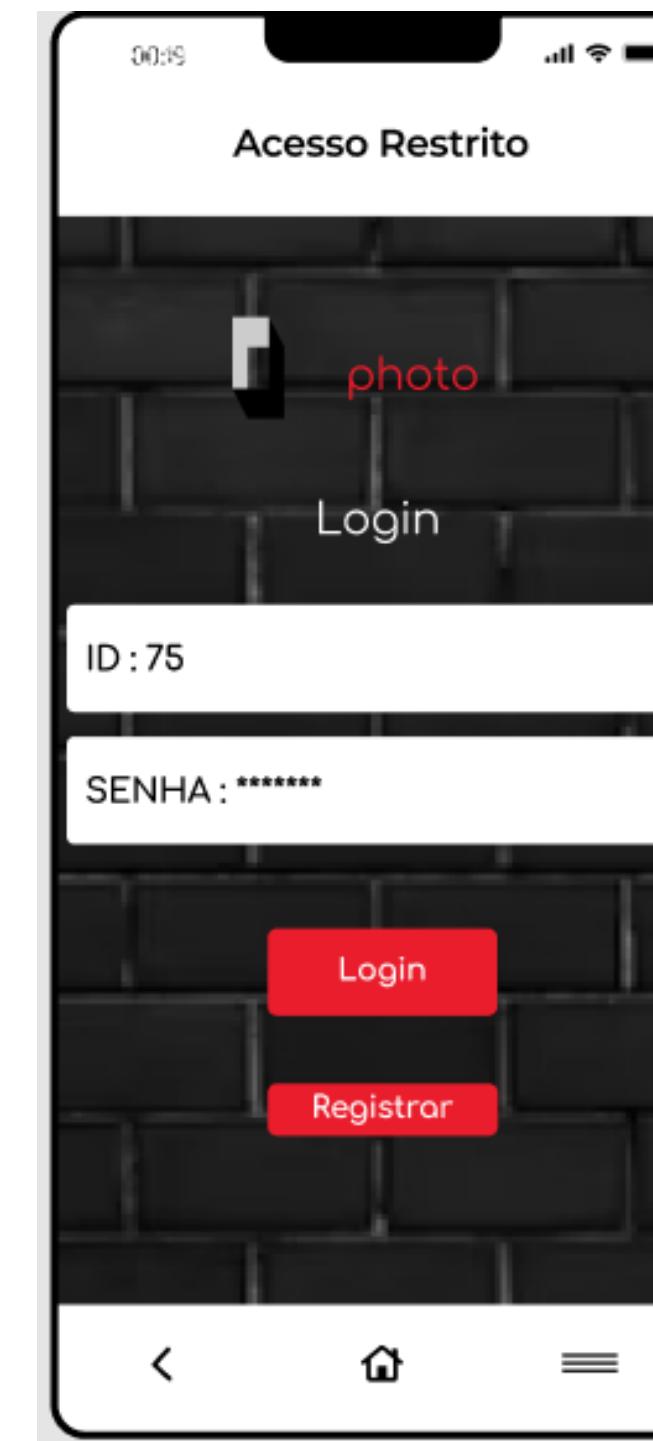
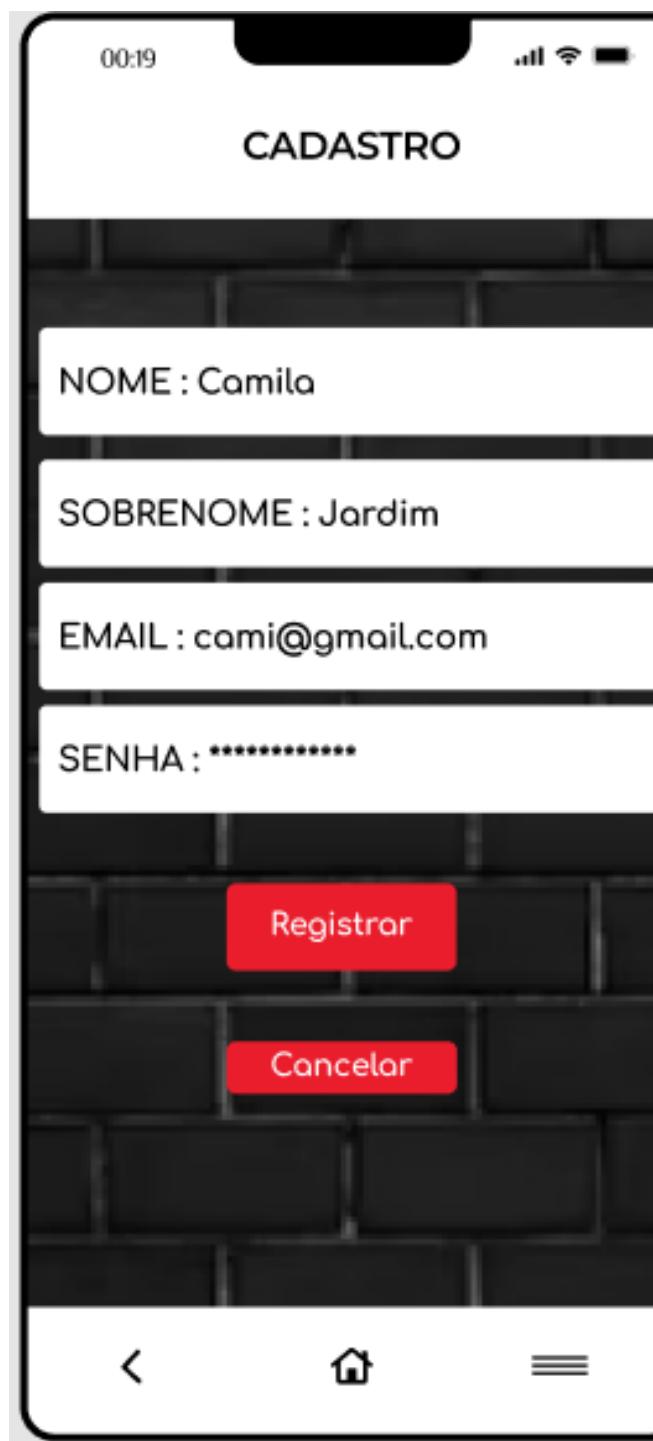
Carrinho

CANCELADO

Produto	Qtdd	Preço
 Hamburger - Rio de Janeiro	- 1 +	1 x 22,90

[Realizar um novo pedido](#)

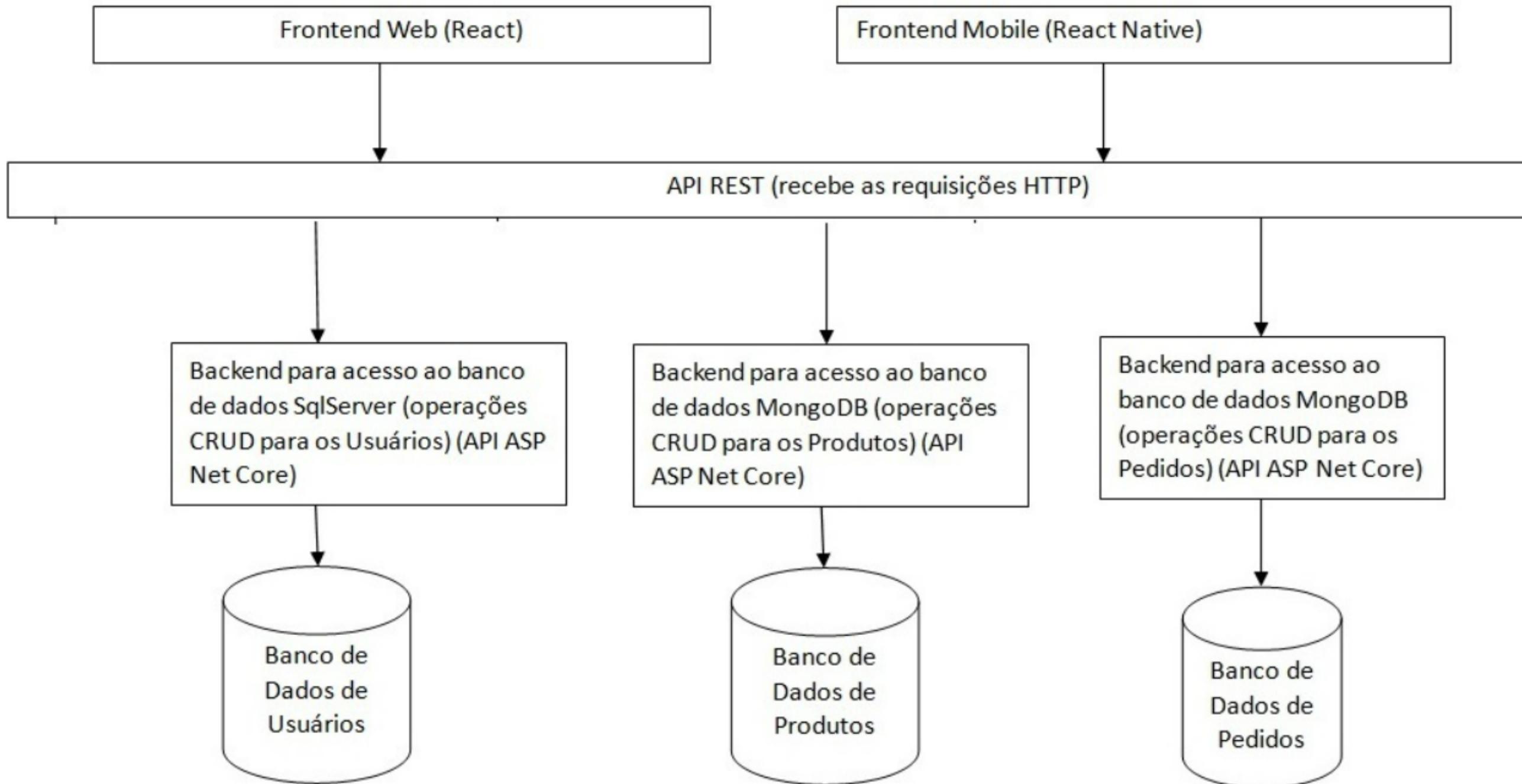
WIREFRAME - MOBILE



ARQUITETURA DA SOLUÇÃO



ARQUITETURA DE SOFTWARE DISTRIBUÍDO



ARQUITETURA DA SOLUÇÃO



MICROSERVIÇOS

Definição: Este estilo arquitetural coloca prioridade no desacoplamento entre serviços através da definição de domínios de aplicação que são independentes de outros, em termos de código fonte e esquemas de banco de dados.



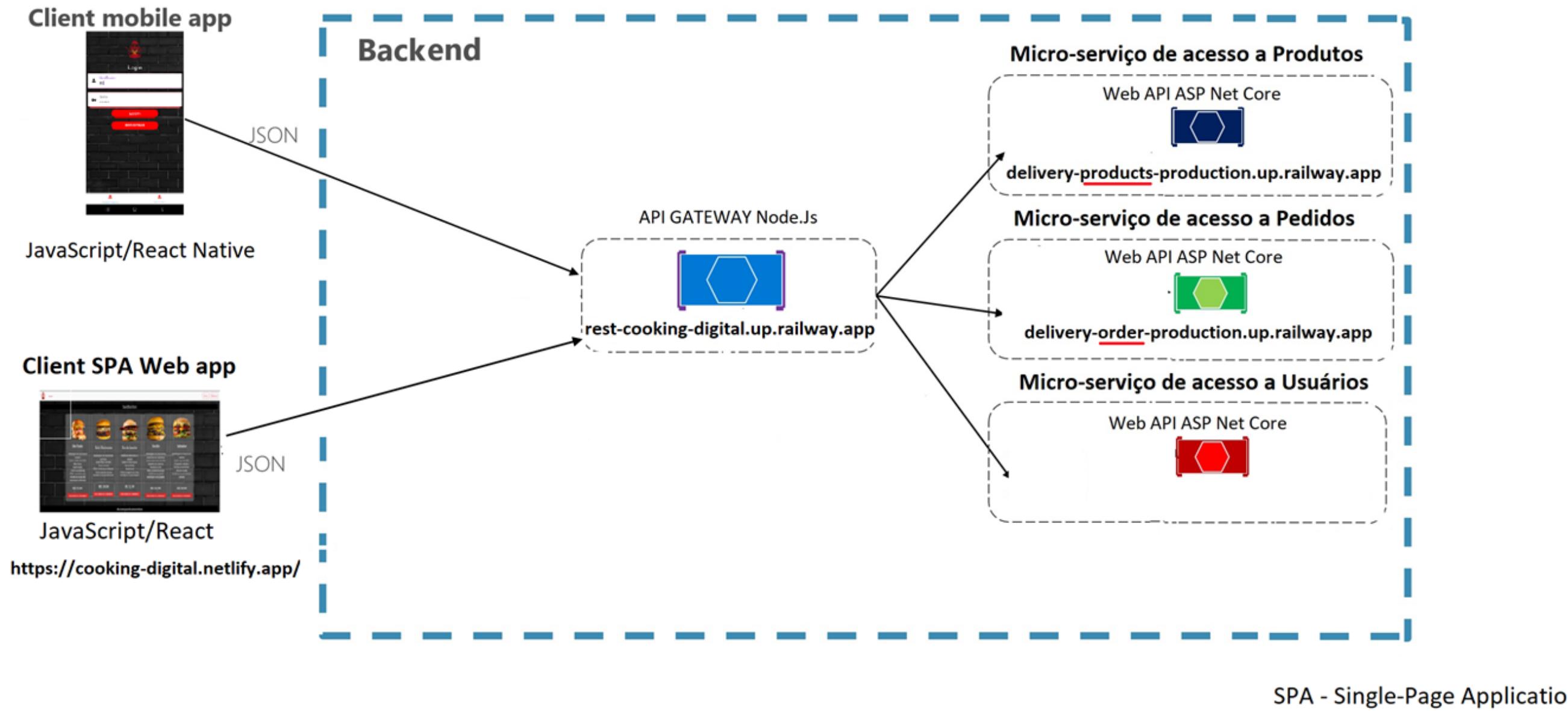
VANTAGENS:

1. Escalabilidade: Projetado para ser escalável, permitindo que o sistema seja expandido facilmente para lidar com um grande número de usuários.
2. Bancos de dados distribuídos: bancos de dados espalhados por vários servidores que armazenam informações de pedidos, cardápios, usuários e outros dados relevantes.



ARQUITETURA DA SOLUÇÃO

API Gateway service



ARQUITETURA DA SOLUÇÃO



VANTAGENS DO GATEWAY DE API'S:

- Conhecida como API Orquestradora, a função dela é centralizar todas as chamadas para terminais de microsserviços em um único serviço.
- Nesse cenário os aplicativos se conectam a um único ponto de extremidade, o Gateway de API, que é configurado para encaminhar solicitações a microsserviços individuais.



ARQUITETURA DA SOLUÇÃO



VANTAGENS DO GATEWAY DE API'S:

- Facilidade na manutenção uma vez que a camada de API's do backend é isolada da camada de clientes do Frontend.
- Segurança: Possibilidade de publicar apenas a API orquestradora, mantendo as demais API's do backend protegidas em uma rede local.



DOCUMENTAÇÃO DAS API'S

 API REST Cooking Digital 1.0.0 OAS3

API de acesso aos backends do projeto Cooking Digital.

Para as rotas autenticadas, faça o login e informe o token no botão Authorize do lado direito dessa tela.

[Contact the developer](#)

Servers Authorize 

Usuarios

POST /usuario/login Login de usuário

GET /usuario Listagem de usuários

POST /usuario Criação de usuário

GET /usuario/{id} Pesquisa de usuário

DELETE /usuario/{id} Exclusão de usuário

PUT /usuario/{id} Atualização de usuário

Produtos

Pedidos

ARQUITETURA DA SOLUÇÃO



DIAGRAMA DE CASOS DE USO

Identificação dos usuários do sistema:

- Administrador
- Usuário de manutenção no site - Usuário responsável pelas atualizações cadastrais
- Clientes do restaurante
- Atendente do restaurante

Casos de uso:

- 1.O administrador do sistema cadastra o Usuário de manutenção responsável pelo cadastro de produtos
- 2.O Usuário de manutenção cadasra os produtos disponíveis para venda
- 3.O Usuário de manutenção atualiza os produtos disponíveis para venda
- 4.O Usuário de manutenção exclui produtos disponíveis para venda
- 5.O cliente cadasra o seu próprio usuário
- 6.O cliente cadasra o seu pedido
- 7.O sistema avisa sobre o novo pedido cadastrado ao Atendente
- 8.O Atendente coloca o pedido no Status "em Andamento"
- 9.O cliente poderá cancelar um pedido desde que ele não tenha sido iniciado, ou seja, o pedido não poderá ser cancelado se estiver no status "em Andamento"
- 10.O Atendente finaliza o pedido

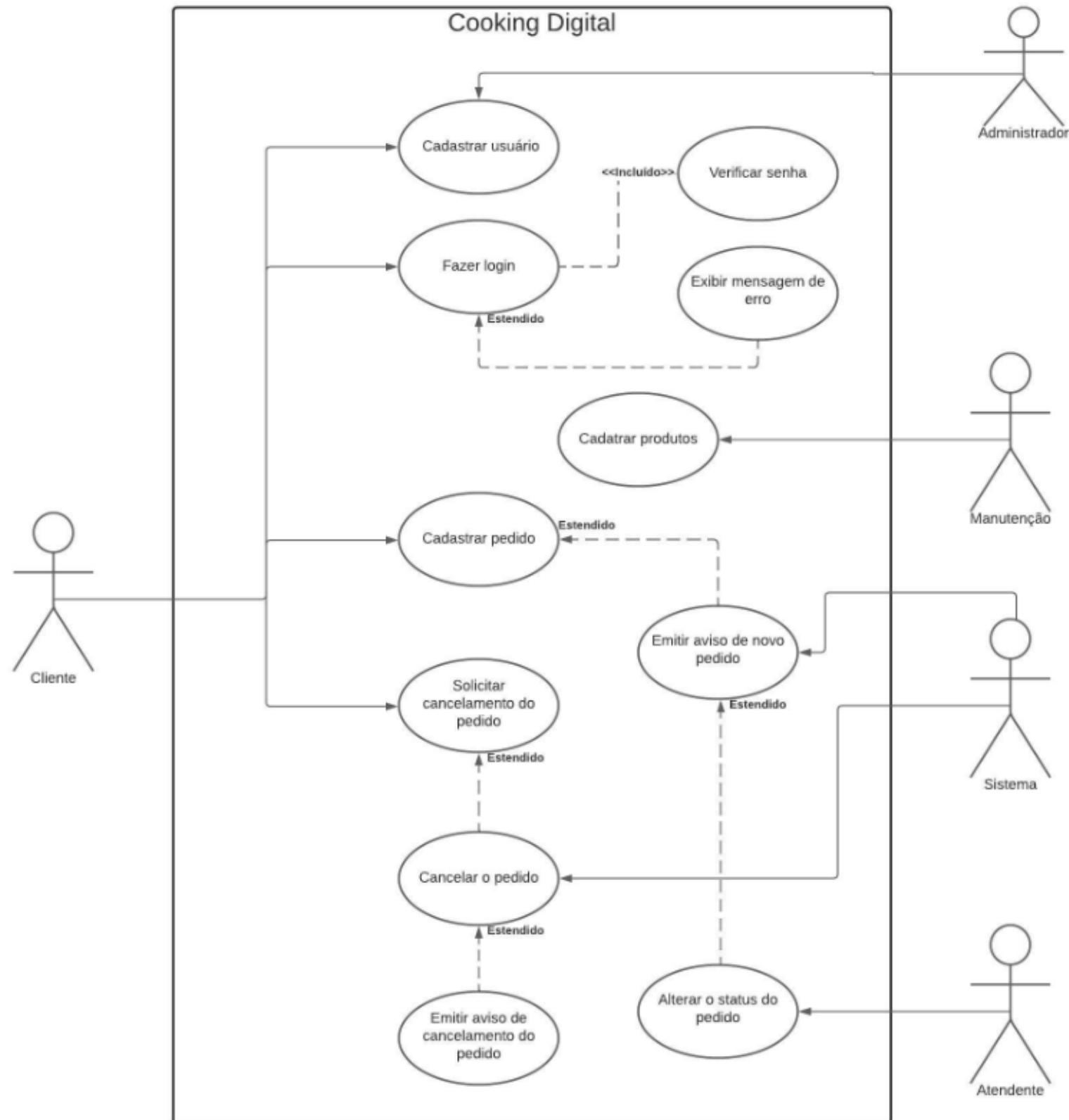
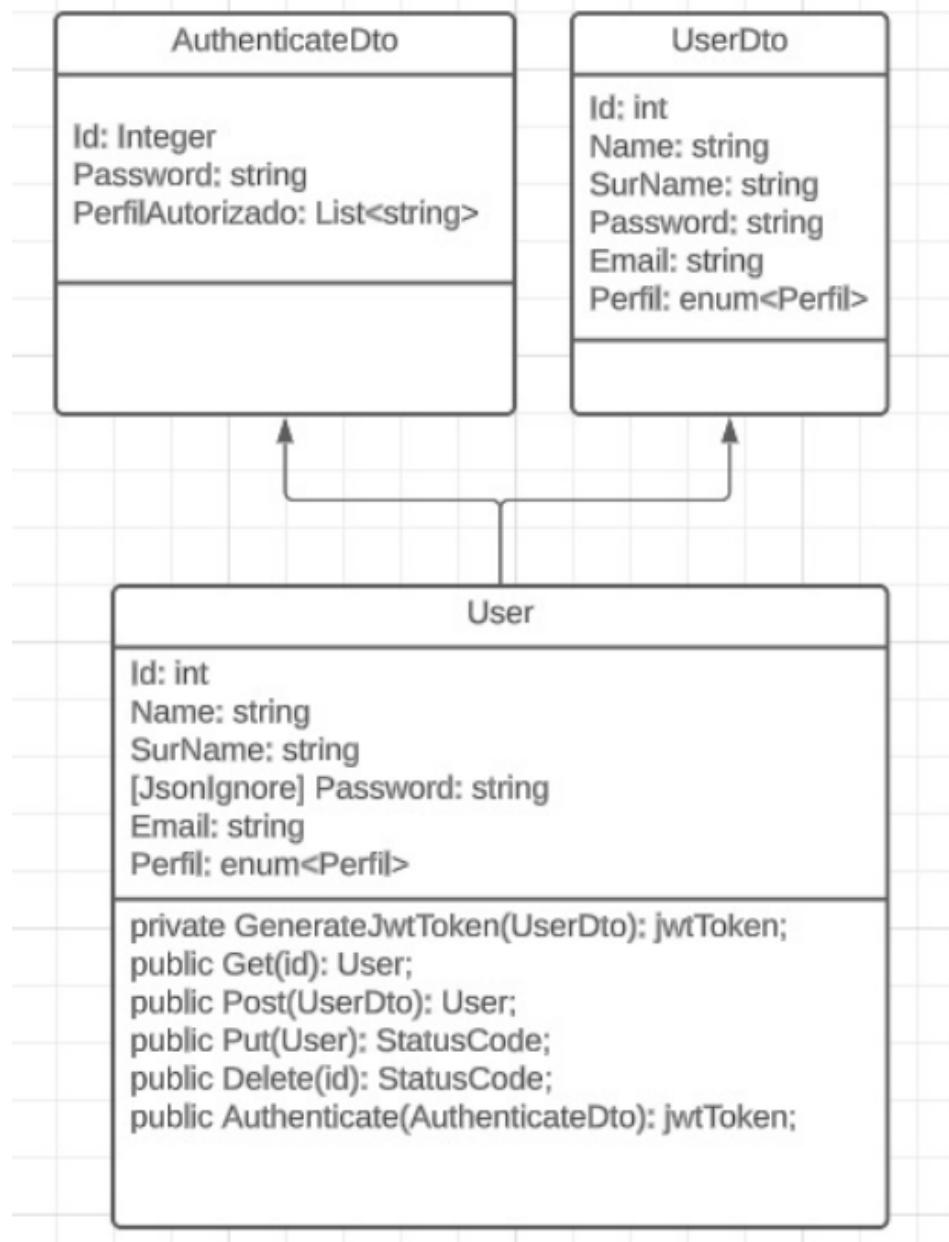


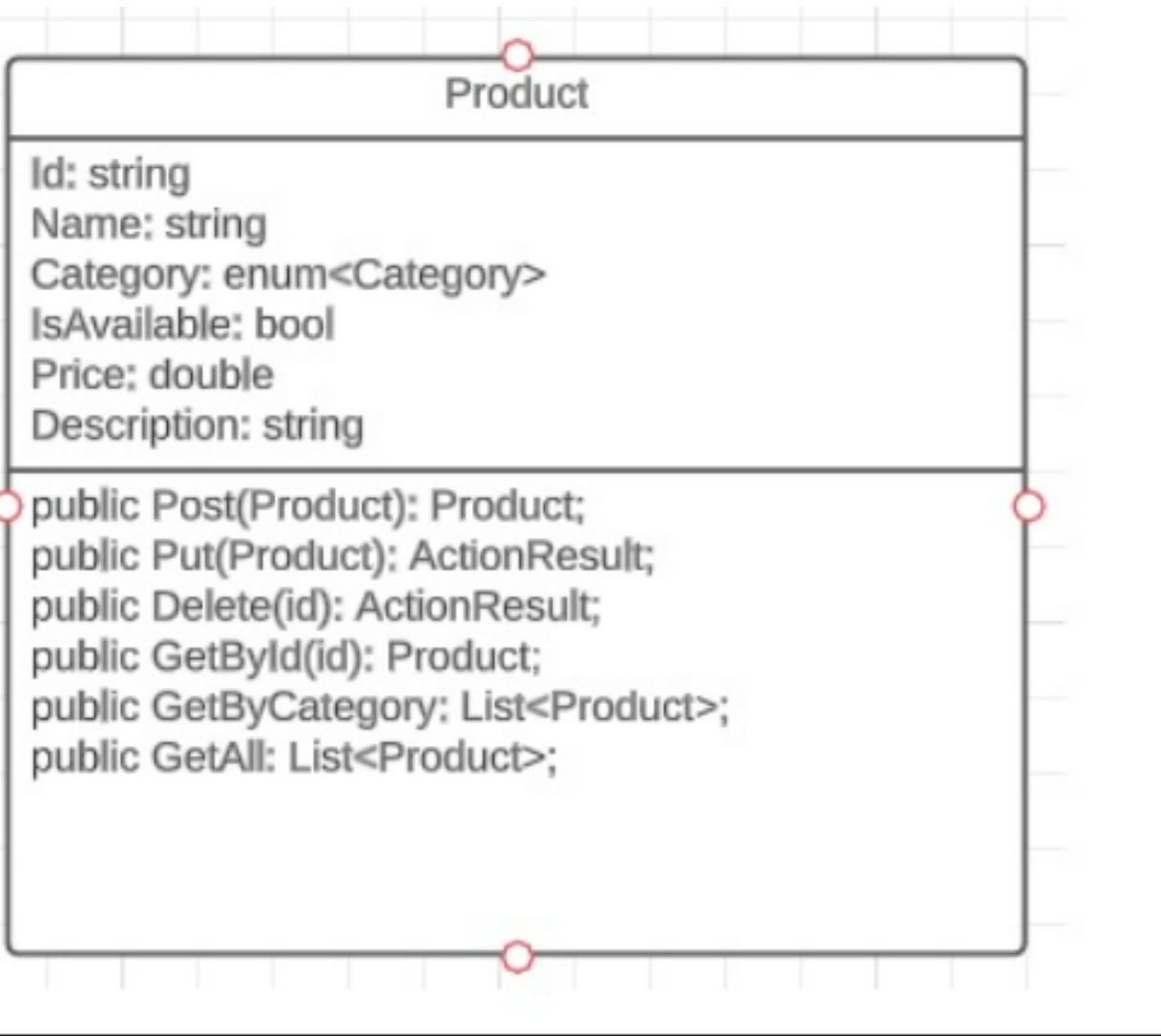
DIAGRAMA DE CLASSES



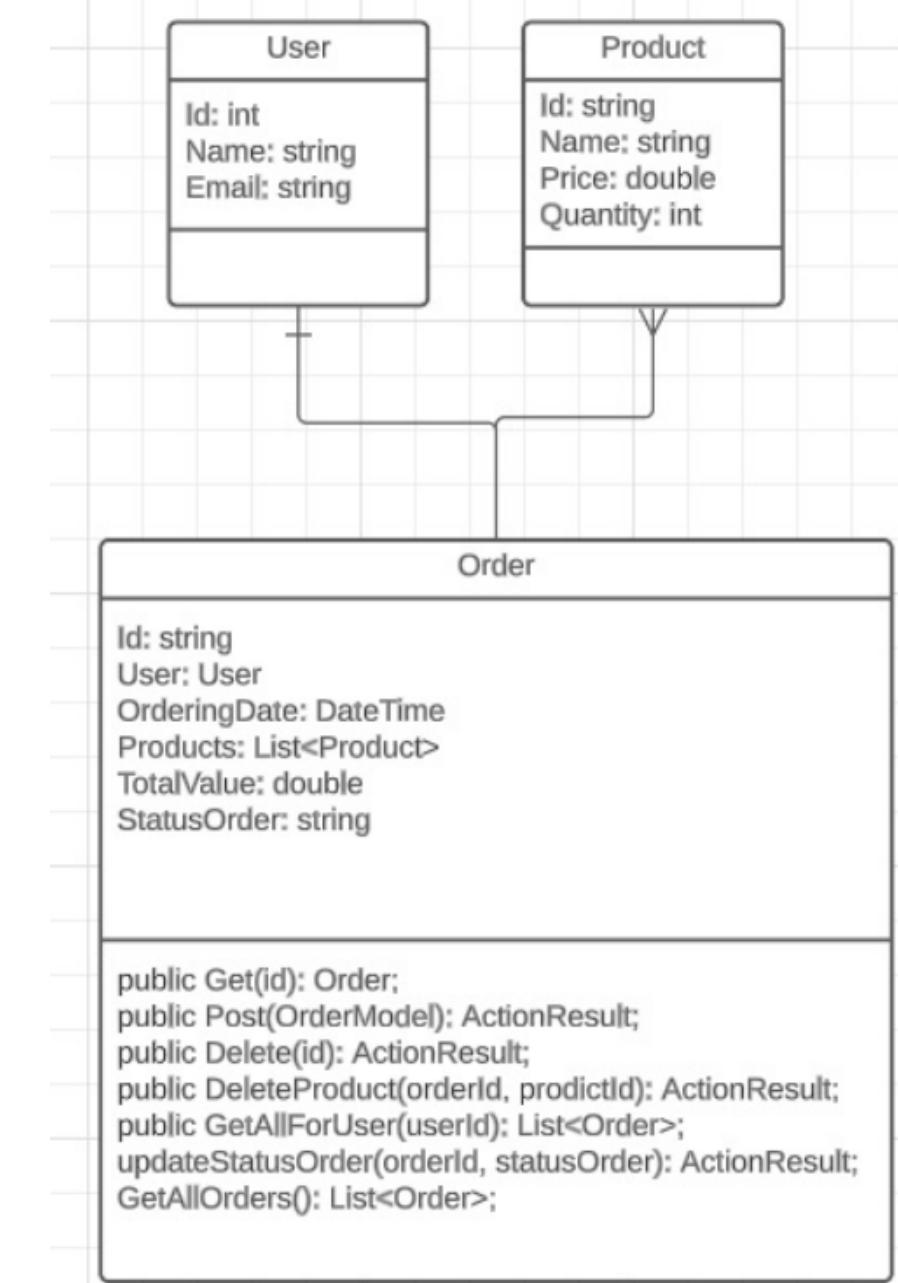
USUÁRIOS



PRODUTOS



PEDIDOS



HOSPEDAGEM



- O banco de dados MongoDB foi hospedado no [railway](#).
Nome do banco de dados: delivery

Coleção: Products e Orders:

MongoDB

Data Connect Logs Metrics Settings

Database: delivery

/ delivery / products / 4000

collections	products	4000
Start collection	Add document	{ "Name": "Belo Horizonte", "Category": "sanduiche", "IsAvailable": true, "Price": 20, "Description": "Hamb\u00frgue bovina grelhado\nQueijo Min derretido\nBacon crocante\nchurrasco defumado\nCebola conservada\nServido em um p\u00e3o brioche" }
products	7752	>
orders	4000	>
	9525	>
	1731	>
	7688	>
	4098	>
	2685	>

MongoDB

Data Connect Logs Metrics Settings

Database: delivery

/ delivery / orders / 526

collections	orders	526
Start collection	Add document	{ >User": { "_id": 25, "Name": "Jaqueline", >Email": "jaq@gmail.com" }, "OrderingDate": "2023-06- 14T00:37:17.385Z", "Products": [{_id": "7752", "Name": "S\u00e3o Paulo", "Quantity": 3, "Price": 20}] }
products	526	>
orders	6521	>
	7240	>
	9084	>

HOSPEDAGEM



- Plataformas de hospedagem:
 - Netlify
 - Railway

- URL do Frontend WEB:

<https://cooking-digital.netlify.app>

- URL para instalação do Frontend Mobile:

<https://expo.dev/artifacts/eas/6iJcjUuwVjiALmU7XWv9Vf.aab>

- URL da API Gateway disponibilizada via ngrok para fins de teste:

<https://7639-2804-14c-5bc6-9f9d-fd3c-ae1b-fd6e-14e1.ngrok-free.app/api-docs/>

APLICAÇÃO

• • •



COOKING DIGITAL