

Curso Técnico Superior Profissional de programação de sistemas de informação

Acesso Móvel a Sistemas de Informação



Trabalho Realizado por:

Carlos Eduardo Afonso Vinagre – 2180687

Rui Jorge Miguel Penetra -2180680

Conteúdo

[Introdução 4](#_Toc61856676)

[Contextualização da aplicação 4](#_Toc61856677)

[Publico alvo 4](#_Toc61856678)

[Requisitos finais implementados 4](#_Toc61856679)

[Motivação para o desenvolvimento do projeto 4](#_Toc61856680)

[Desenvolvimento 5](#_Toc61856681)

[Funcionalidades da aplicação 5](#_Toc61856682)

[Metodologia, procedimentos, regras 5](#_Toc61856683)

[Tecnologias usadas e ou pesquisadas 6](#_Toc61856684)

[Estrutura da base de dados Local 6](#_Toc61856685)

[Descrição dos problemas/dificuldades 6](#_Toc61856686)

[Justificação das soluções implementadas 6](#_Toc61856687)

[Identificação das funcionalidades propostas, mas não implementadas 6](#_Toc61856688)

[URL utilizado pela Aplicação 7](#_Toc61856689)

[O que pode fazer online e o que pode fazer offline 7](#_Toc61856690)

[Online 7](#_Toc61856691)

[Offline 7](#_Toc61856692)

[Manual De utilização 8](#_Toc61856693)

[Registo 8](#_Toc61856694)

[Login 8](#_Toc61856695)

[Atualizar Perfil 9](#_Toc61856696)

[Contactos 9](#_Toc61856697)

[Produto 12](#_Toc61856698)

[Configuração de IP para a API 14](#_Toc61856699)

[Logout 14](#_Toc61856700)

[Criar 1 Pedido como empregado de mesa 15](#_Toc61856701)

[Consultar 1 Pedido 16](#_Toc61856702)

[Eleminar 1 pedido 17](#_Toc61856703)

[Cliente criar 1 takeaway 18](#_Toc61856704)

[Fazer update de 1 Produto no Pedido Restaurante/Takeaway 20](#_Toc61856705)

[Eleminar 1 produto no pedido Restaurante/Takeaway 20](#_Toc61856706)

[Conclusão 21](#_Toc61856707)

[Descrição dos resultados obtidos 21](#_Toc61856708)

[Problemas por resolver 21](#_Toc61856709)

[Previsão para o futuro da aplicação 21](#_Toc61856710)

# Introdução

## Contextualização da aplicação

A nossa aplicação consiste numa solução para os restaurantes facilitarem a consulta e a criação de pedidos na parte do funcionário, na parte do cliente eles podem fazer a criação de pedidos takeaway, consultar os produtos que o restaurante tem e contactar o restaurante por chamada ou através de 1 email.

## Publico alvo

O publico alvo desta aplicação é os empregados de mesa que utilizam esta aplicação para consultar e registar os pedidos dos clientes, e os clientes que também podem utilizar a aplicação para consultar que refeições existem no restaurante ou fazer pedidos de takeaway também têm a opção de telefonar para o restaurante ou mandar email para esclarecerem alguma dúvida ou para marcar 1 reserva.

## Requisitos finais implementados

O projeto usa a Shared Preference para armazenar o nome de utilizador, password, token, cargo do utilizador,IP,género,nome completo,o id e é guardado o estado da checkbox relembrar.

A aplicação utiliza o CRUD:

* Criar Pedidos,Perfis
* Visualizar Pedidos
* Update Perfil,Pedidos
* Remover Pedidos

Os produtos depois de serem carregados pela primeira vez através da internet são guardados na base de dados local.

## Motivação para o desenvolvimento do projeto

O que nos motivou para desenvolver o projeto foi arranjar 1 forma mais fácil de registar pedidos tanto da parte do funcionário como da parte do cliente e visualizar os mesmos, como também dispõe de 1 secção onde é possível visualizar as ementas, reduzindo assim o “desperdício” de papel visto que os pedidos são feitos através da aplicação e as ementas em vez de ser necessário escrever em papel são visíveis através da aplicação.

# Desenvolvimento

## Funcionalidades da aplicação

* Fazer login
* Fazer Registo de utilizador
* Fazer Pedidos para comer no Restaurante
* Fazer Pedidos Takeaway
* Atualizar perfil
* Ver ementa
* Fazer Telefonemas para o restaurante
* Mandar Email para o restaurante
* Ver Horário do restaurante

## Metodologia, procedimentos, regras

Para a criação deste projeto foi usada 1 metodologia ágil em que usamos o jira para organizarmos as tarefas o jira consiste em 9 Sprints cada Sprint com a duração de 2 semanas, e dividimos as tarefas de maneira a que os Story Points tivesse mais ou menos equilibrados para cada 1 de nós.

**Carlos Vinagre Rui Penetra**

Criação Listeners Adicionar produtos ao pedido

Criação de Registar Criação de modelos

Atualização de ícones e imagens Criação do login

Adicionar permissões ao manifest Atualizar Perfil

Implementação shared preference Atualizar Pedidos Produto

Remover produtos ao pedido Apagar Pedido

Atualizar produtos ao pedido Criação da base de dados local

Relatório Android Relatório Android

## Tecnologias usadas e ou pesquisadas

As tecnologias usadas foi o android studio que usamos para programar em java, utilizamos o emulador que vem incorporado no mesmo e para controlar a versão guardada no GitHub nos usamos o GitKraken para gerir as versões.

## Estrutura da base de dados Local

**Tabela Produtos**

|  |  |
| --- | --- |
| ID\_PRODUTO | Integer Primary Key |
| NOME\_PRODUTO | **Text Not Null** |
| INGREDIENTES\_PRODUTO | **Text Not Null** |
| PRECO\_PRODUTO | **Text Not Null** |
| CATEGORIA\_PRODUTO | **Integer Not Null** |

## Descrição dos problemas/dificuldades

O perfil demora um pouco carregar.

Para mostrar os produtos enquanto esta com o login feito tivemos que repetir 1 atividade.

## Justificação das soluções implementadas

Temos que criar 1 atividade a mais para mostrar os produtos quando a pessoa esta com o log in feito, no futuro teríamos que reaproveitar a atividade quando o cliente não esta com o login feito para solucionar o problema.

## Identificação das funcionalidades propostas, mas não implementadas

* Bottom navigation menu para o cliente
* Na proposta de projeto mencionamos que o cliente tinha que selecionar 1 hora para ir buscar o pedido, no entanto decidimos apenas disponibilizar o estado para o cliente saber em que estado se encontra o seu pedido ou seja assim que o cozinheiro recebe o pedido e assim que ele tenha disponibilidade ele passa o pedido para “em preparação” e começa a preparar o mesmo.

## URL utilizado pela Aplicação

http://[IP Adress]/GestorRestauranteAPI/API/web/v1/auth

* **O Url acima indicado serve para Autenticação e Registo**

http://[IP Adress]/GestorRestauranteAPI/API/web/v1/produto

* **O Url acima indicado serve para ver a ementa**

http://[IP Adress]/GestorRestauranteAPI/API/web/v1/pedido

* **O Url acima indicado serve para ver a lista de pedidos**

http://[IP Adress]/GestorRestauranteAPI/API/web/v1/pedidoproduto

* **O Url acima indicado serve para ver os produtos dentro do pedido**

http://[IP Adress] /GestorRestauranteAPI/API/web/v1/perfil

* **O Url acima indicado serve para ver o perfil e atualizar**

## O que pode fazer online e o que pode fazer offline

### Online

A aplicação online pode fazer pedidos, pode atualizar perfil, pode fazer login.

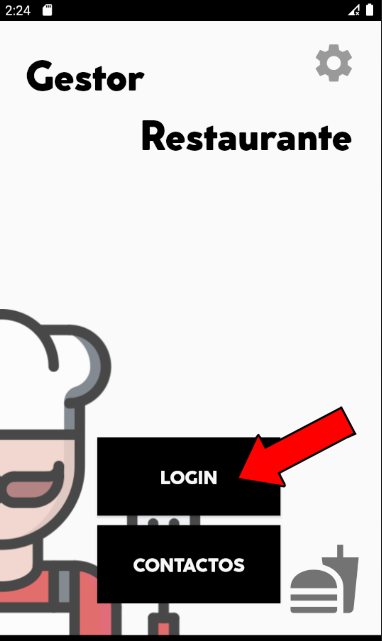
### Offline

A aplicação offline pode ver os pratos após ter carregado os mesmos da primeira vez que se ligou a rede, pode fazer telefonemas no botão contacto, pode fazer logout e tem 1 hiperligação que redireciona o utilizador para a aplicação de mandar emails ex. (Gmail,Outlook).

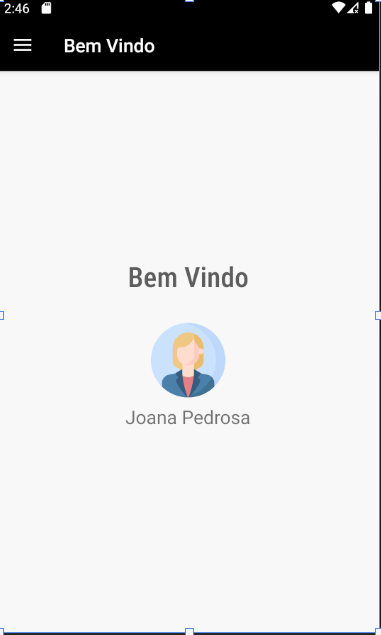
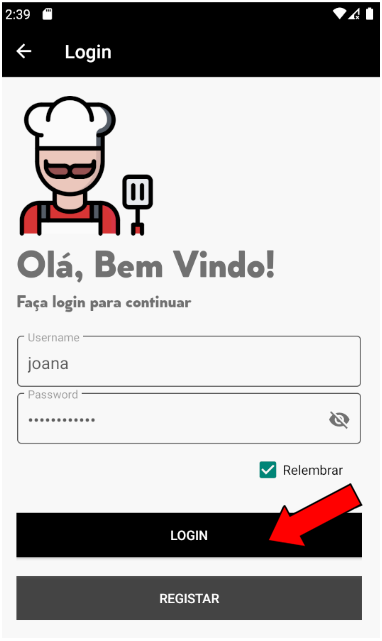
## Manual De utilização

### Registo

Uma imagem com texto

Descrição gerada automaticamente

### Login

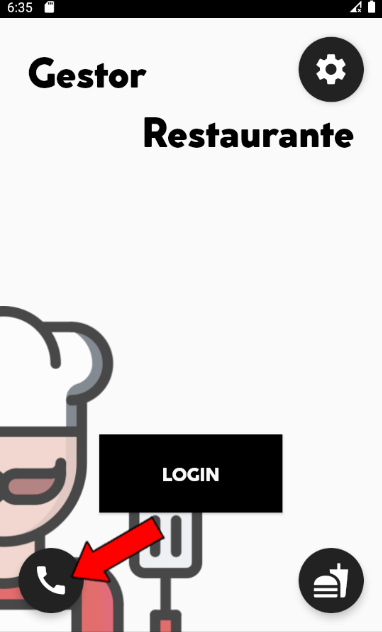


### Uma imagem com texto Descrição gerada automaticamenteAtualizar Perfil

### Contactos

#### Opção 1

#### Opcao 2

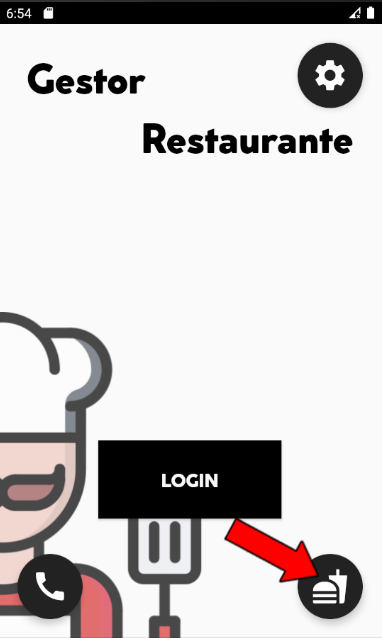
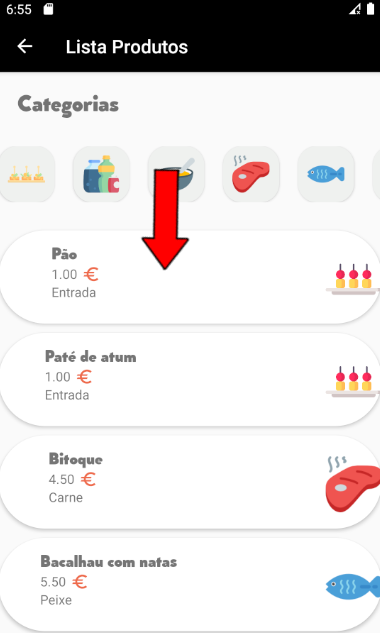


##### Telefonar para o restaurante e mandar email

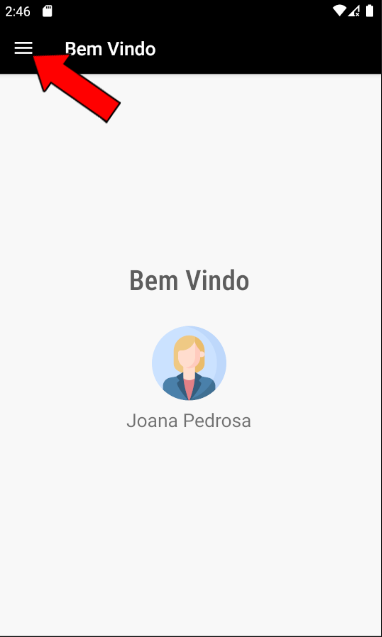
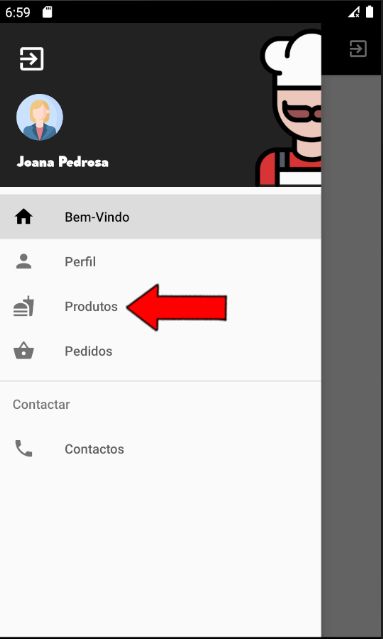


### Produto

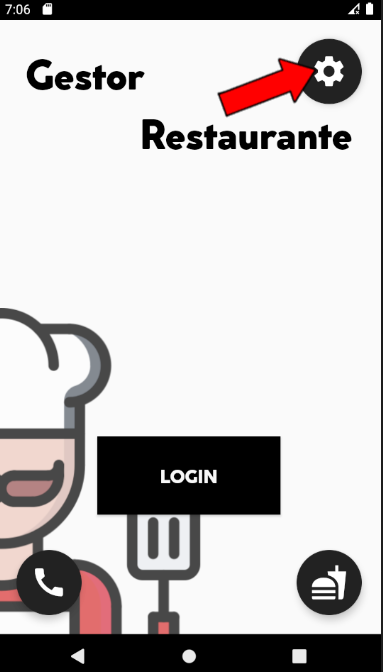
#### Opção 1



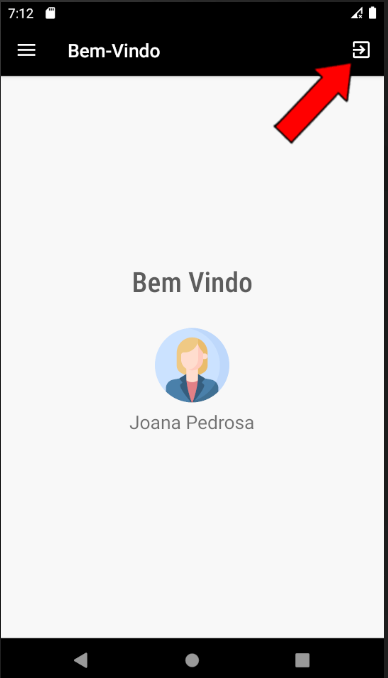
#### Opção 2



### Configuração de IP para a API

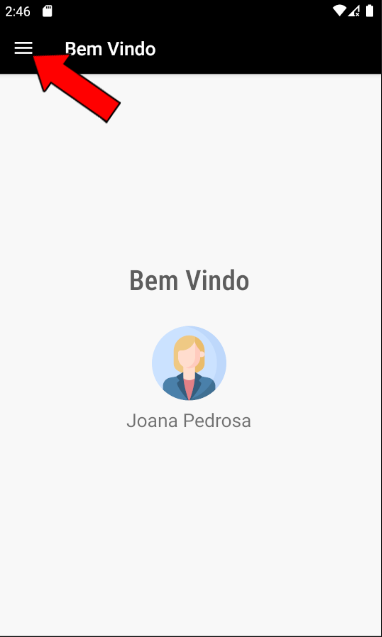
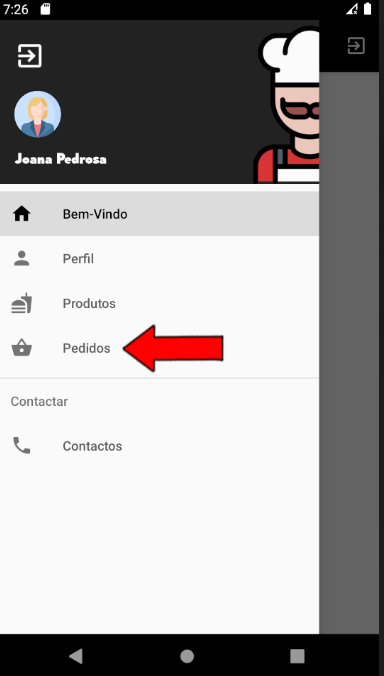


### Logout



### Criar 1 Pedido como empregado de mesa

Uma imagem com texto

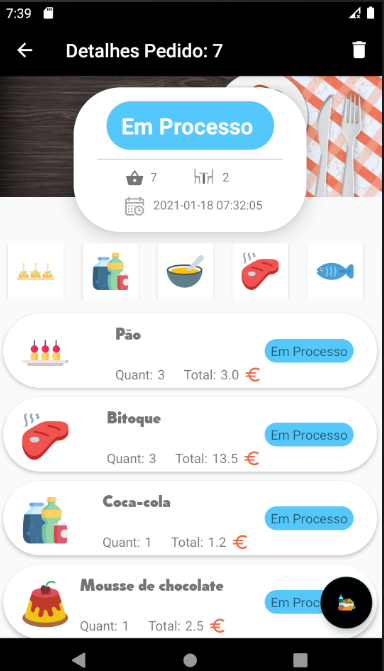
Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

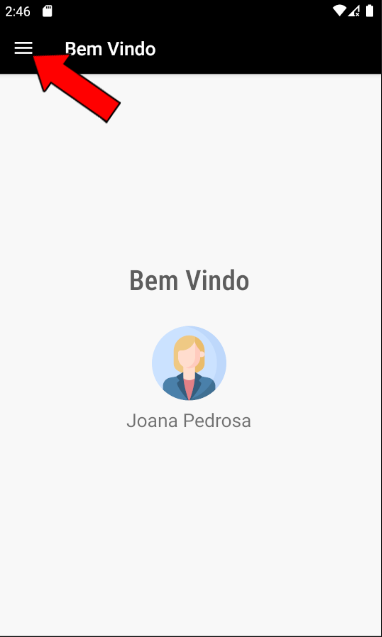
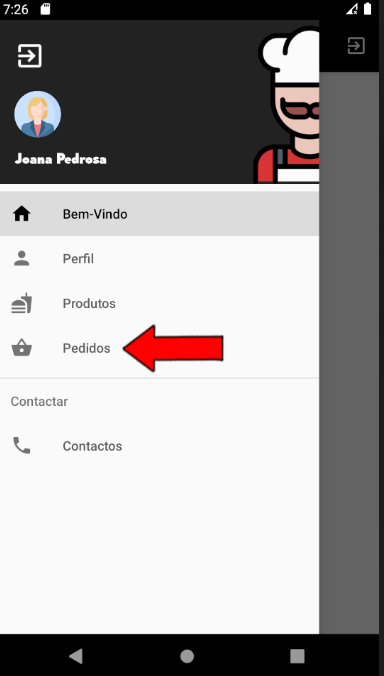
Uma imagem com texto

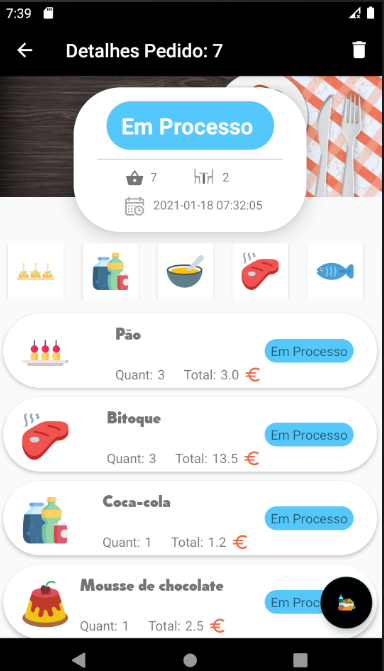
Descrição gerada automaticamente

Uma imagem com texto

Descrição gerada automaticamente

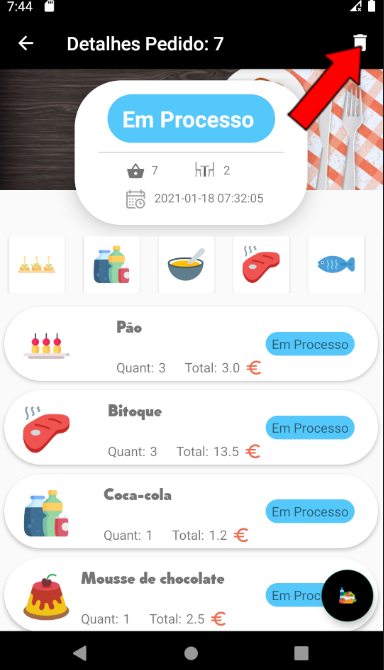
### Uma imagem com texto Descrição gerada automaticamenteConsultar 1 Pedido





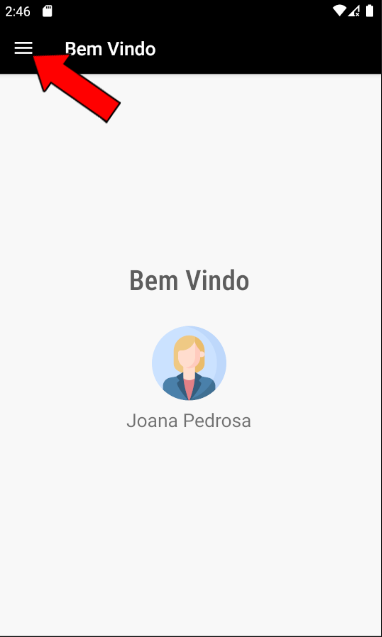
### Uma imagem com texto Descrição gerada automaticamenteEleminar 1 pedido

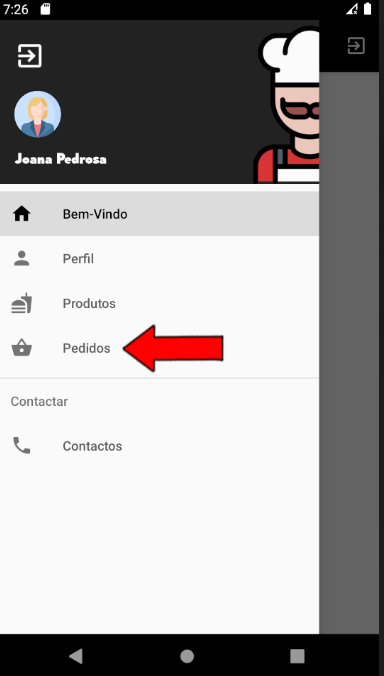
Uma imagem com texto

Descrição gerada automaticamente

### Cliente criar 1 takeaway

Uma imagem com texto

Descrição gerada automaticamente

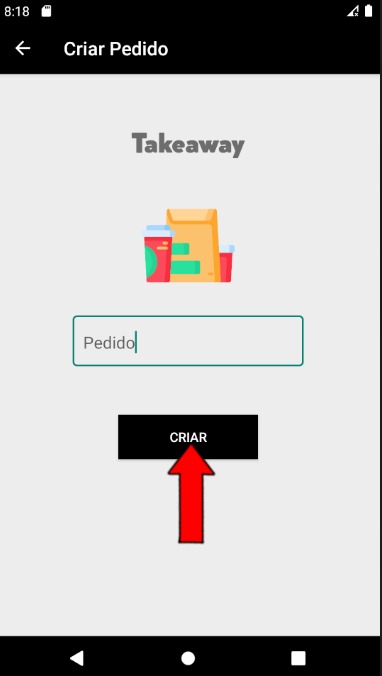


Uma imagem com texto

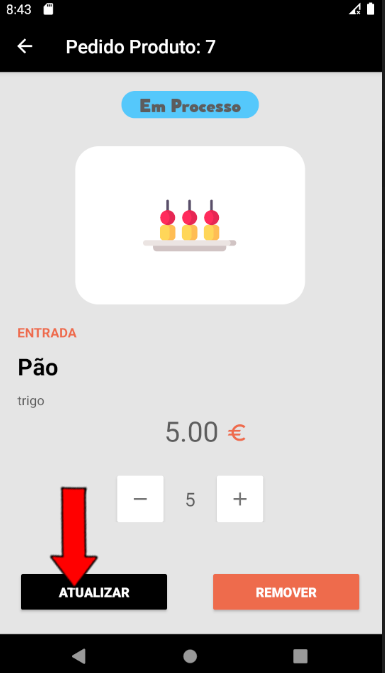
Descrição gerada automaticamenteUma imagem com texto

Descrição gerada automaticamenteUma imagem com texto, captura de ecrã, aparelho

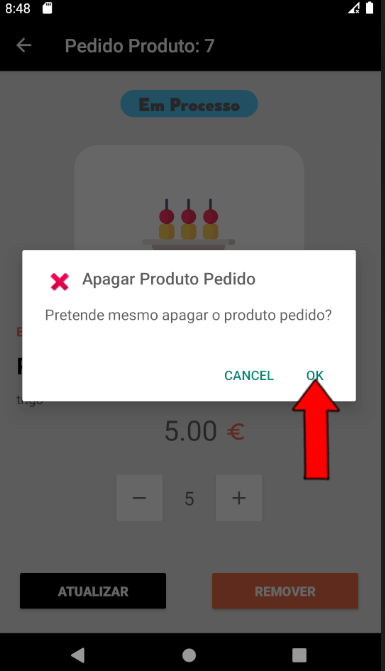
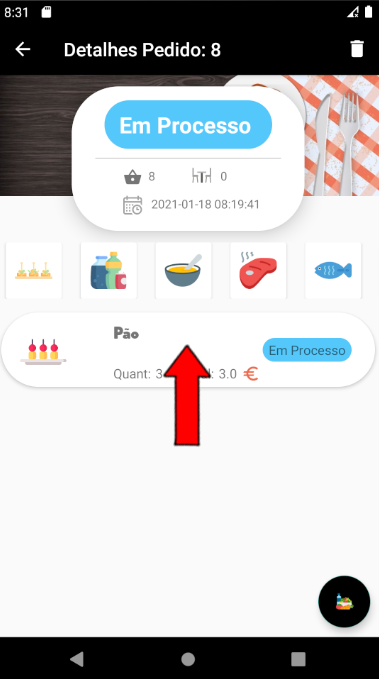
Descrição gerada automaticamenteUma imagem com texto

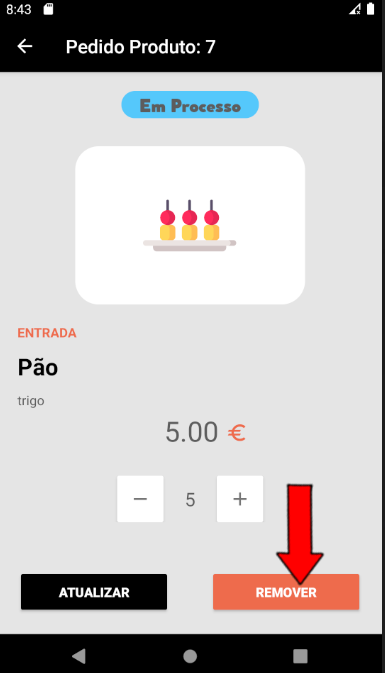
Descrição gerada automaticamente

### Fazer update de 1 Produto no Pedido Restaurante/Takeaway



### Eleminar 1 produto no pedido Restaurante/Takeaway





# Conclusão

## Descrição dos resultados obtidos

Com esta aplicação desenvolvida tornou-se mais fácil registar pedidos tanto para os empregados de mesa como para os clientes que desejam pedir takeaway, também se tornou mais fácil obter informação sobre o restaurante por parte do cliente realçando a possibilidade de consultar ementas e informações sem necessidade de se deslocar ao restaurante.

## Problemas por resolver

No decorrer do desenvolvimento do projeto foi necessário criar 1 atividade a mais para mostrar os produtos quando a pessoa esta com o log in feito.

Falta a implementação do filtras nos pedidos.

## Previsão para o futuro da aplicação

No futuro seria implementado 1 widget do google maps onde poderia se ver a localização do restaurante.

No futuro seria implementado 1 opção para selecionar o horário que o cliente quer ir buscar o pedido como também o funcionário poderia consultar o horário e as próprias faltas.