

Curso Técnico Superior Profissional de programação de sistemas de informação

Acesso Móvel a Sistemas de Informação



**Trabalho Realizado por:**

Carlos Eduardo Afonso Vinagre – nº2180687

Rui Jorge Miguel Penetra - nº 2180680

Conteúdo

[Introdução 4](#_Toc61899856)

[Contextualização da aplicação 4](#_Toc61899857)

[Publico alvo 4](#_Toc61899858)

[Requisitos finais implementados 4](#_Toc61899859)

[Motivação para o desenvolvimento do projeto 5](#_Toc61899860)

[Desenvolvimento 5](#_Toc61899861)

[Funcionalidades da aplicação 5](#_Toc61899862)

[Metodologia, procedimentos, regras 5](#_Toc61899863)

[Tecnologias usadas e ou pesquisadas 6](#_Toc61899864)

[Estrutura da base de dados Local 6](#_Toc61899865)

[Descrição dos problemas/dificuldades 7](#_Toc61899866)

[Justificação das soluções implementadas 7](#_Toc61899867)

[Identificação das funcionalidades propostas, mas não implementadas 7](#_Toc61899868)

[URL utilizado pela Aplicação 8](#_Toc61899869)

[O que pode fazer online e o que pode fazer offline 10](#_Toc61899870)

[Online 10](#_Toc61899871)

[Offline 10](#_Toc61899872)

[Manual de utilização 10](#_Toc61899873)

[Registo 10](#_Toc61899874)

[Login 11](#_Toc61899875)

[Atualizar Perfil 11](#_Toc61899876)

[Contactos 12](#_Toc61899877)

[Produto 13](#_Toc61899878)

[Configuração de IP para a API 15](#_Toc61899879)

[Logout 15](#_Toc61899880)

[Criar 1 Pedido como empregado de mesa 16](#_Toc61899881)

[Consultar 1 Pedido 17](#_Toc61899882)

[Eliminar 1 pedido 18](#_Toc61899883)

[Cliente criar 1 takeaway 19](#_Toc61899884)

[Fazer update de 1 Produto no Pedido Restaurante/Takeaway 21](#_Toc61899885)

[Eliminar 1 produto no pedido Restaurante/Takeaway 21](#_Toc61899886)

[Conclusão 22](#_Toc61899887)

[Descrição dos resultados obtidos 22](#_Toc61899888)

[Problemas por resolver 22](#_Toc61899889)

[Previsão para o futuro da aplicação 22](#_Toc61899890)

# Introdução

## Contextualização da aplicação

A nossa aplicação foi desenvolvida com o intuito de disponibilizar a qualquer restaurante uma solução mais económica e segura, tanto no processo de registo e consulta de pedidos feitos no restaurante por parte do empregado de mesa como também os pedidos takeaway por parte do cliente, em conjunto a possibilidade de poder consultar ementas do restaurante sem que seja necessário dirigir ao restaurante.

## Publico alvo

O publico alvo desta aplicação são os restaurantes como também os próprios clientes, trazendo assim uma solução mais eficaz na realização de pedidos de refeições no restaurante por parte do empregado de mesa como também pedidos takeaway por parte do cliente. Para além disso também é possível consultar ementas pela aplicação sendo que após consultar uma primeira vez online, é possível consultar as mesmas offline.

## Requisitos finais implementados

* O projeto usa a Shared Preference para armazenar dados como:
  + Endereço IP (Para efeitos de desenvolvimento e apresentação);
  + Username, password, token de acesso e o estado da checkbox relembrar;
  + O nome de utilizador (Nome + Apelido), cargo (empregado de mesa, cliente) e o numero de registo (ID) ;
* Possibilidade de qualquer cliente se registar podendo depois atualizar os seus próprios dados pessoais e de autenticação (exceto a password).
* Possibilidade de realizar pedidos tanto takeaway por parte do cliente como pedidos por parte do empregado de mesa no restaurante.
* Possibilidade de consultar ementas online e após essa consulta uma primeira vez online, será possível consultar as mesmas ementas offline sendo que os dados das ementas ficaram guardados localmente numa base de dados.
* Opção de contactar o restaurante tanto via telefónica como por via email.

## 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 como a visualização das ementas, com a vantagem de poder apos a primeira utilização online, poder consulta-las oflline.

# 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 mesmo está dividido em 9 Sprints cada Sprint com a duração de 2 semanas, e dividimos as tarefas de maneira a que os Story Points se enquadrassem no esforço previsto para cada uma das tarefas.

|  |  |
| --- | --- |
| **Carlos Vinagre** | **Rui Penetra** |
| * Criação Listeners * Criação de Registar * Atualização de ícones e imagens * Adicionar permissões ao manifest * Implementação shared preference * Remover produtos ao pedido * Atualizar produtos ao pedido * Limpeza de código * Relatório Android | * Adicionar produtos ao pedido * Criação de modelos * Criação do login * Atualizar Perfil * Atualizar Pedidos * Apagar Pedido * Criação da base de dados local * Limpeza de código * 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 | Integer Primary Key |
| NOME | Text Not Null |
| INGREDIENTES | Text Not Null |
| PRECO | Text Not Null |
| CATEGORIA | Integer Not Null |

## Descrição dos problemas/dificuldades

No decorrer do desenvolvimento do projeto foram surgindo algumas dificuldades, que acabaram por dificultar o desenvolvimento do mesmo, sendo necessário despender mais tempo em algumas tarefas que acabou por influenciar no início de outras. Dificuldades essas que começaram a surgir no decorrer da criação de novas atividades/fragmentos em que começou a surgir conflitos entre a identificação (id) de cada componente visual de cada actividade/fragmento visto que estariam com identificações iguais. Também surgiu algumas dificuldades na utilização dos listeners numa primeira faze as listas produtos e pedidos após a criação/remoção/atualização de registos não atualizavam.

## Justificação das soluções implementadas

No início do desenvolvimento do projeto definimos que após o cliente se registar necessitaria de após de concluir o seu registo com sucesso, de fazer login em vez de se registar e automaticamente fazer o login. Inicialmente foi definido que iria existir um menu diferente consoante fosse funcionário ou cliente, contudo optamos por utilizar o mesmo menu para os dois, visto que de momento teriam funcionalidades muito semelhantes, onde apenas o que os diferenciam é que o cliente faz um pedido takeaway e o funcionário (empregado de mesa ) faz um pedido a ser realizado no restaurante. Na parte dos pedidos foi determinado que seria possível apagar pedidos com produtos associados (Pedidos Produto), sendo que futuramente só iria ser possível apagar caso o pedido esteja em estado de processo.

## Identificação das funcionalidades propostas, mas não implementadas

No início do projeto foram definidas algumas propostas que acabaram por não ser implementadas como foi o caso: de um menu diferente caso fosse cliente, menu esse (Bottom navigation), também foi mencionado que o cliente poderia selecionar uma hora a buscar o pedido, no entanto acabou por disponibilizar o estado para que o cliente tenha a precessão em que fase se encontra o seu pedido, fases essas, em processo, em preparação, concluído.

## URL utilizado pela Aplicação

* **Autenticação**

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/auth

* **Registo utilizador**

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/auth/registar

* **Perfil**

Consultar:

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/perfil?access-token= **{ token user}**

Atualizar:

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/perfil/ **{ id user } /**atualizar?access-token= **{ token user }**

* **Produtos**

Consultar Todos:

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/produto

Consultar por categoria:

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/produto/categoria/**{ ID categoria }**

* **Pedido**

Consultar todos os pedidos

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedido?access-token= **{ token user }**

Criar pedidos

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedido/criar?access-token= **{ token user }**

Apagar pedidos

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedido/**{ id pedido } /**apagar?access-token= **{ token user }**

* **Pedido Produto**

Consultar todos pedidos produto de um determinado pedido

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedidoproduto/all/**{ id pedido }**?access-token= **{ token user }**

Criar um determinado pedido produto

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedidoproduto/criar?access-token= **{ token user }**

Atualizar um determinado pedido produto

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedidoproduto/ **{ id pedido produto } /**atualizar?access-token= **{ token user }**

Apagar um determinado pedido produto

http:// **{ IP Adress }** /GestorRestauranteAPI/API/web/v1/pedidoproduto/ **{ id pedido produto } /**remover?access-token= **{ token user }**

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

### Online

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

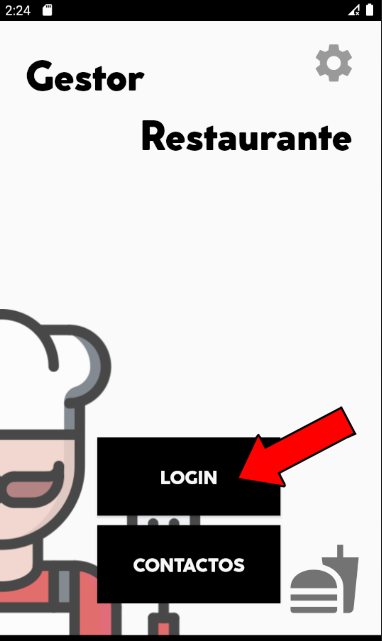
### Offline

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

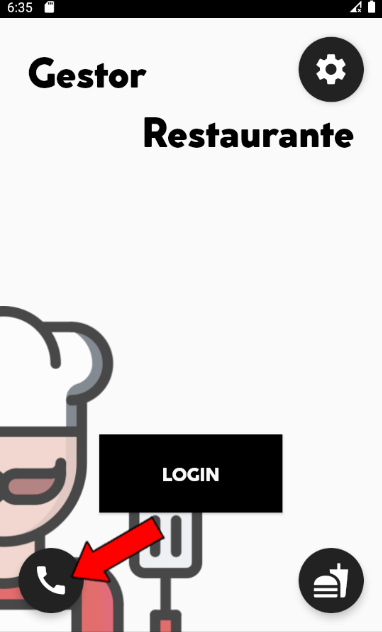
### LoginUma imagem com texto Descrição gerada automaticamente

### Atualizar Perfil

### Contactos

#### Opção 1

#### Opcao 2

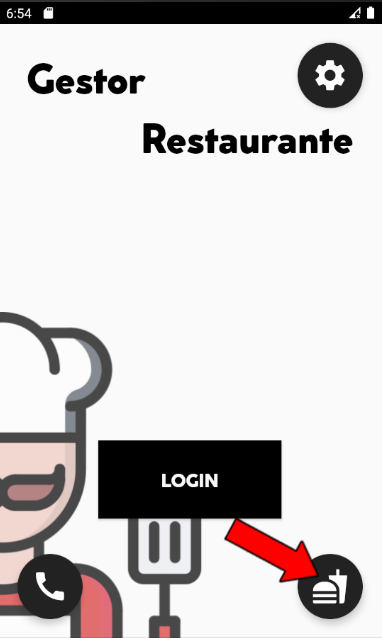
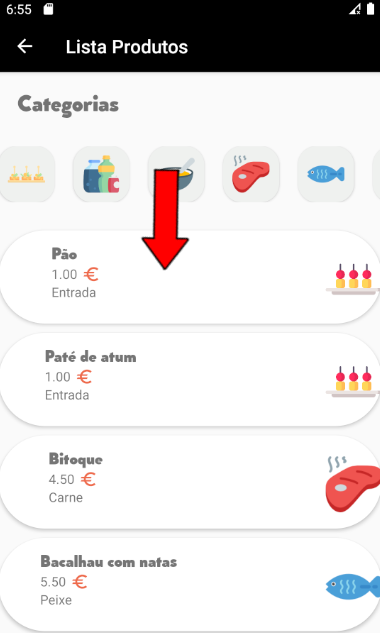


##### Telefonar para o restaurante e mandar email

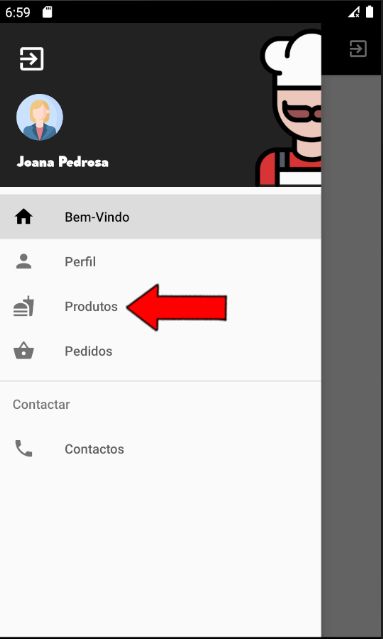
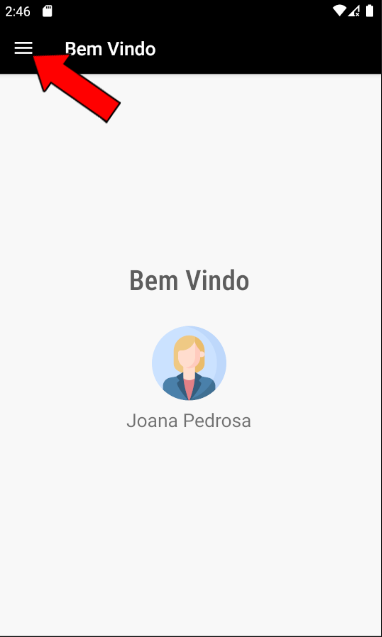
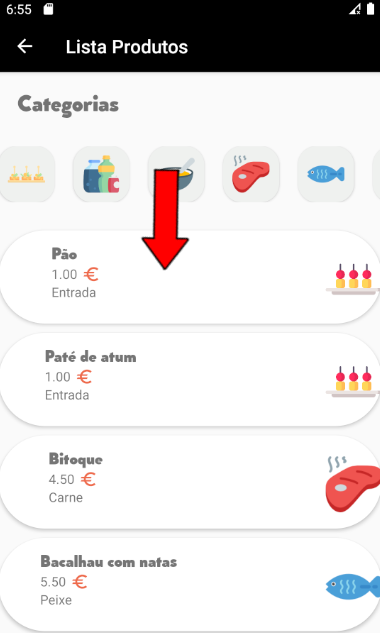


### Produto

#### Opção 1

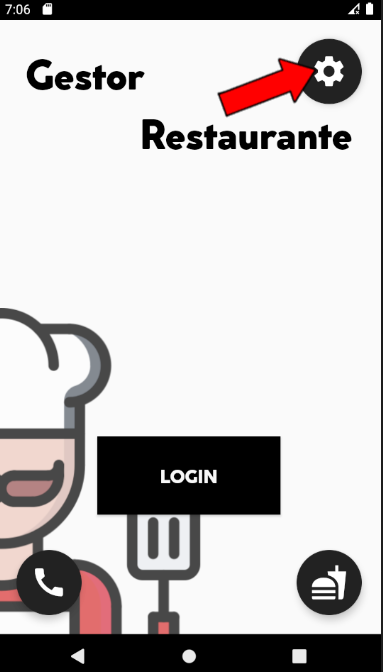
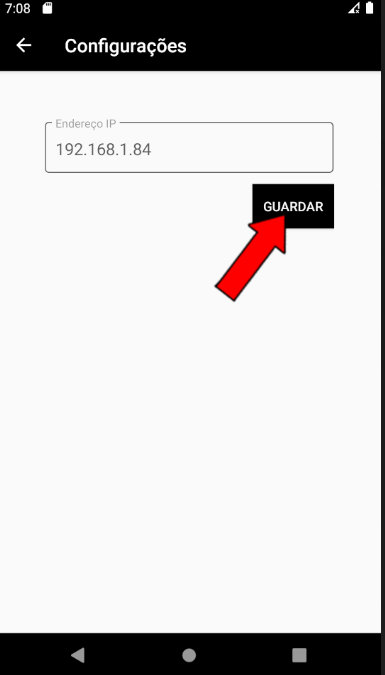


#### Opção 2

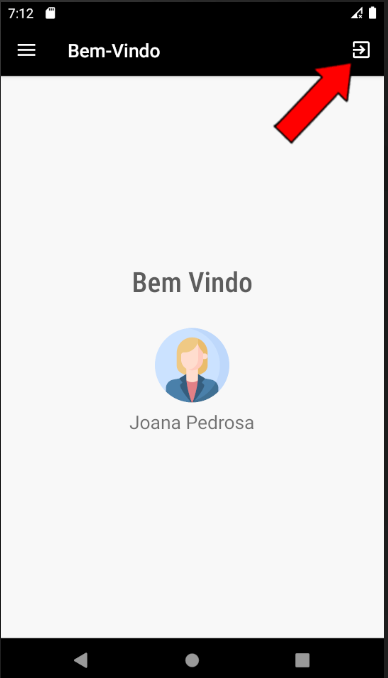
**



### Configuração de IP para a API

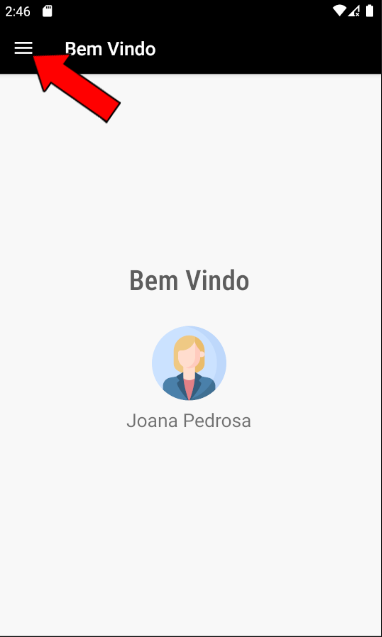
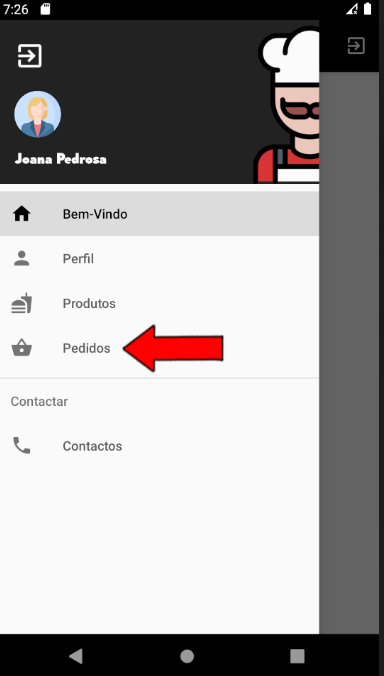


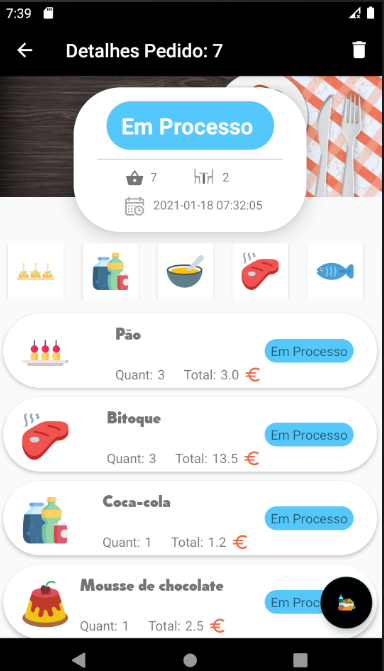
### Logout



### Criar 1 Pedido como empregado de mesa

Uma imagem com texto

Descrição gerada automaticamente



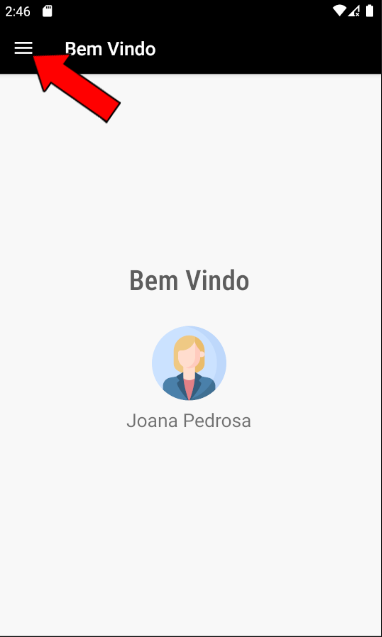
Uma imagem com texto

Descrição gerada automaticamente

Uma imagem com texto

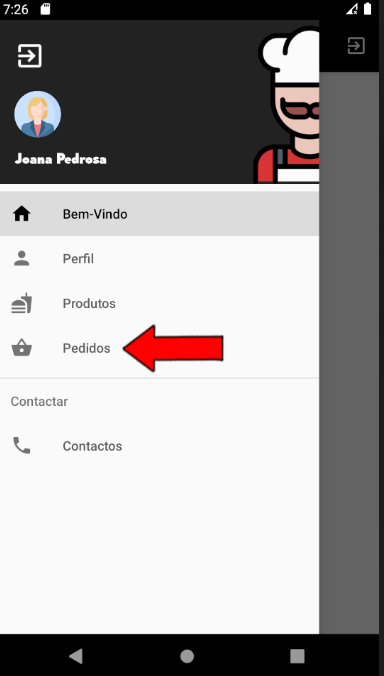
Descrição gerada automaticamenteUma imagem com texto

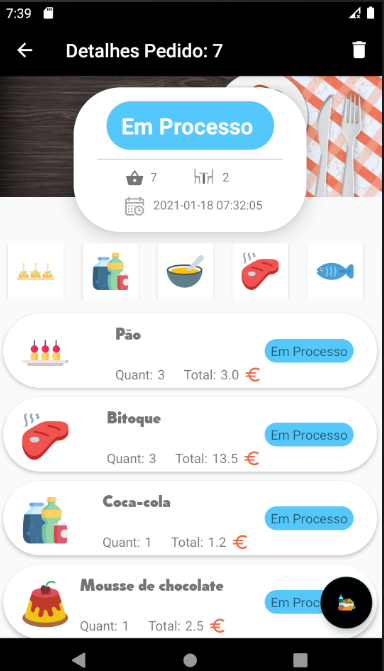
Descrição gerada automaticamente



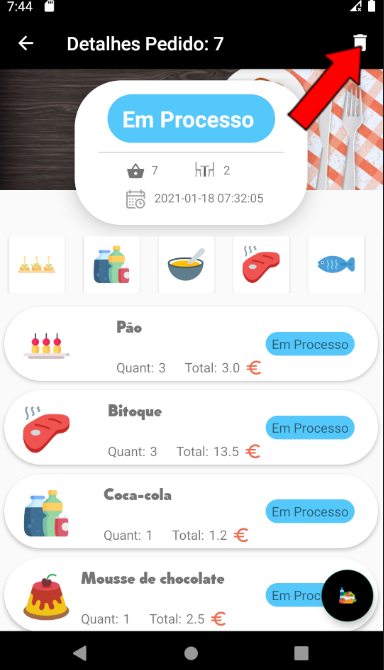
### Consultar 1 Pedido

Uma imagem com texto

Descrição gerada automaticamente



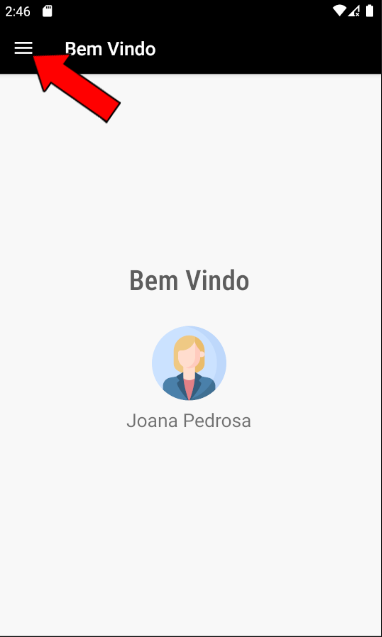
### Uma imagem com texto Descrição gerada automaticamenteEliminar 1 pedido

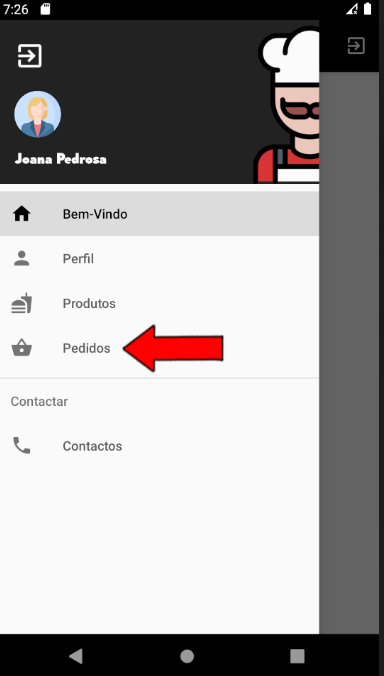
Uma imagem com texto

Descrição gerada automaticamente

### Cliente criar 1 takeaway

Uma imagem com texto

Descrição gerada automaticamente

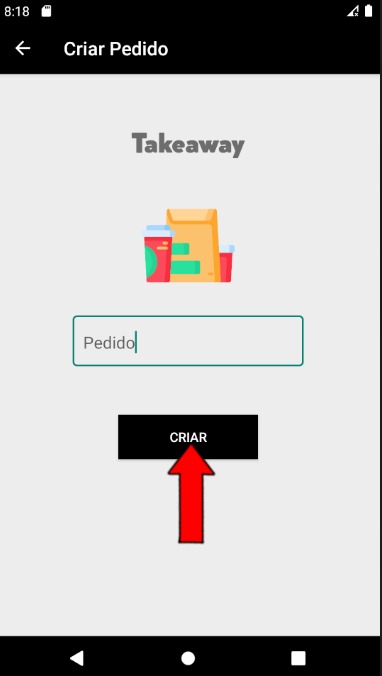


Uma imagem com texto

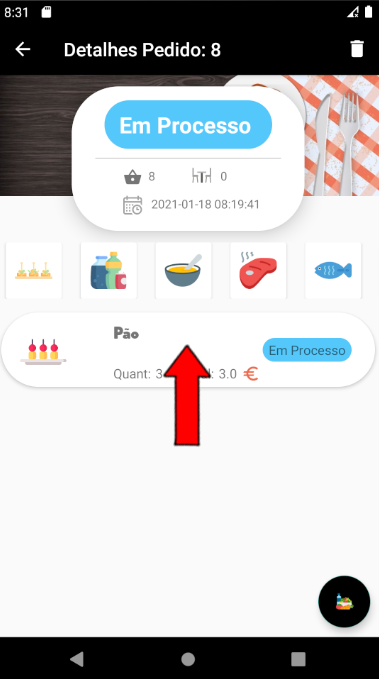
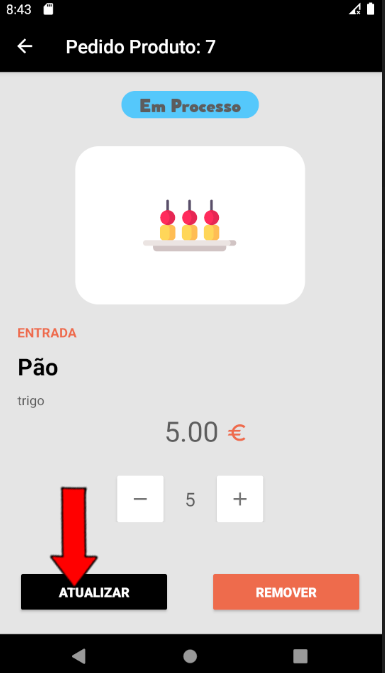
Descrição gerada automaticamenteUma imagem com texto

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

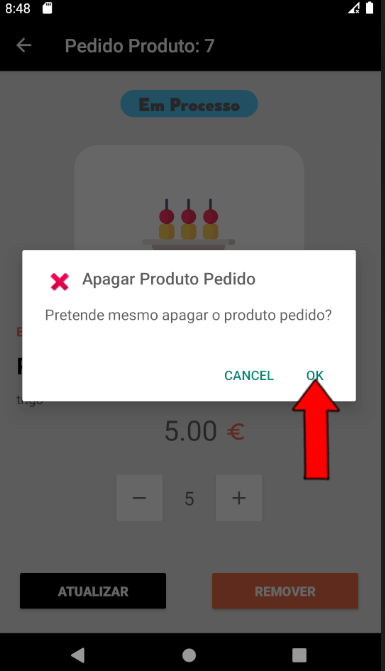
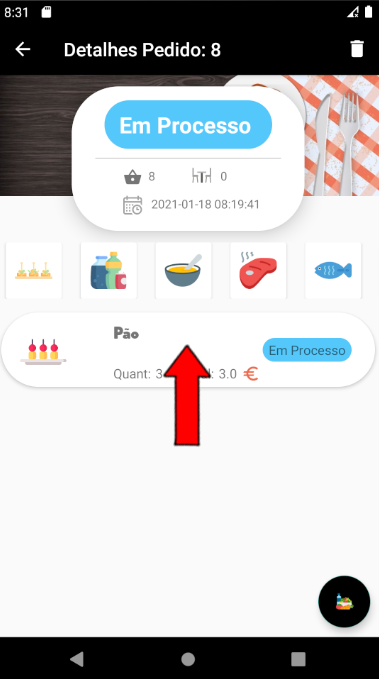
Descrição gerada automaticamenteUma imagem com texto

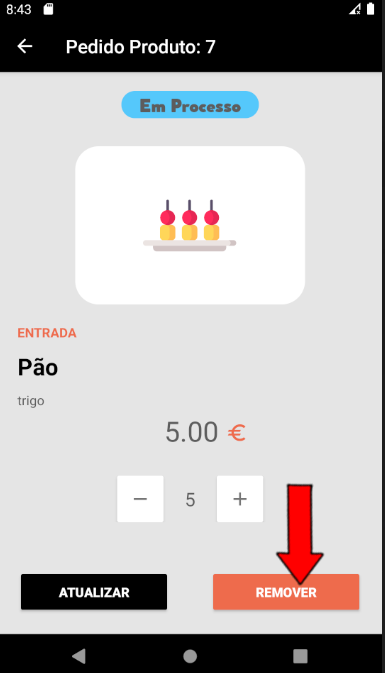
Descrição gerada automaticamente

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



### Eliminar 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, incluindo offline 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 se encontra logada, e para esse efeito e para outros casos a ideia seria reutilizar outras atividades de modo a reduzir actividades reduzindo assim algumas atividades. Também ficou por implementar a filtragem dos pedidos produto por categoria.

## 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.