

2022 / 2023

# MySmartHome

Licenciatura em Engenharia Informática



**Identificação do Grupo:**

- Grupo 13

**Nomes dos Alunos:**

- Luís Oliveira – Nº 8190370;
- Rafael Costa – Nº 8180246;
- Teresa Peixoto – Nº 8190334.

## Índice

Agradecimentos.....	3
1. Introdução .....	4
1.1 Âmbito do Projeto.....	4
1.2 Propósito do Projeto.....	4
1.3 Ferramentas Utilizadas .....	5
1.4 Estrutura do Documento .....	5
2. Planeamento.....	6
2.1 Identificação dos Requisitos .....	6
2.2 Mockups.....	10
2.3 Estrutura de Dados .....	15
2.4 Estrutura do Retrofit.....	15
3. Implementação.....	16
3.1 Estrutura do Projeto .....	16
3.2 Controlo de Versões.....	16
3.3 Definição de Layouts.....	17
3.4 Identificação dos mecanismos utilizados .....	18
3.4.1 Intents .....	18
3.4.2 Notificações .....	21
3.4.3 Navegação .....	22
3.4.4 Google Maps.....	23
3.4.5 Componentes .....	28
3.4.6 Room e Retrofit .....	28
3.5 Enumeração dos problemas encontrados .....	29
3.5 Objetivos Futuros.....	30
3.7 Resultados Obtidos .....	30
4. Conclusão.....	31
5. Referências Bibliográficas e Web .....	32
6. Anexos .....	33
6.1 Mockups e Gitlab .....	33
6.2 Apresentação dos Layouts.....	33

6.2.1 Login .....	33
6.2.2 Criar Conta.....	34
6.2.3 Escolher Tipo de Casa (Criar / Associar).....	35
6.2.4 Nova Casa .....	36
6.2.5 Nova Divisão .....	37
6.2.6 Dispositivos Não Conectados .....	38
6.2.8 Novo Dispositivo.....	39
6.2.9 Dispositivos Conectados.....	40
6.2.10 Home .....	41
6.2.11 Perfil .....	42
6.2.12 Pedidos de Adesão .....	43
6.2.13 Configurações Pessoais .....	44
6.2.14 Definições de cada Dispositivo.....	45
6.2.15 Detalhes da Divisão .....	46
6.2.15 Membros .....	47
6.2.16 Convidar Membro .....	48
6.2.17 Consumos .....	49
6.2.18 Assistente Pessoal .....	50
6.2.19 Ajuda .....	51
6.2.20 Sobre .....	52
6.2.21 Definições .....	53
6.2.22 Associar Casa .....	54
6.2.23 Localização .....	55

## Agradecimentos

O grupo gostaria de enaltecer o professor responsável por lecionar tanto as aulas Teóricas como Práticas, Fábio Silva, pela forma como tem interagido adequadamente com os alunos, ao longo do presente semestre.

Isto não só acontece quando os alunos têm dúvidas, mas também na forma prática e clara que este transmite o conhecimento necessário durante as aulas.

Isto reflete sem dúvida na aprendizagem dos alunos e, consequentemente, nos resultados que estes obtêm na unidade curricular.

## 1. Introdução

### 1.1 Âmbito do Projeto

O presente tema do trabalho prático foi escolhido pelo grupo, na unidade curricular de Computação Móvel e Ubíqua e incide precisamente na conceção de uma aplicação *Android* capaz de gerir uma *Smart Home*.

Atualmente e, dado que a tecnologia está a desenvolver-se à velocidade da luz, as *Smart Home's* estão cada vez mais a ter procura, tanto em Portugal como ao redor do mundo.

Esta procura não só é feita pelos amantes da tecnologia de último grito, como também por pessoas que pretendem possuir casas mais inteligentes e sustentáveis que permitam o monitoramento dos mais variados dispositivos que a casa tem.

Esta decisão promove não só a sustentabilidade, como também outros aspetos bastante importantes a ter em conta, como a segurança, o bem-estar dos habitantes, a otimização de consumos, redução de desperdícios, personalização pessoal de configurações de dispositivos, entre muitas outras vantagens.

### 1.2 Propósito do Projeto

A conceção deste projeto envolve os mecanismos abordados nas aulas sobre *Android*, desde a criação de *Layouts* para cada ecrã pertencente à aplicação, até ao armazenamento de dados e hospedagem da própria aplicação num servidor de comunicação em tempo real.

No entanto, e para que naturalmente isso seja possível, os principais objetivos desta aplicação passam essencialmente pelos seguintes pontos:

- Garantir que os *Layouts* dos Ecrãs se foquem em maximizar funcionalidades e minimizar a complexidade dos mesmos;
- Recorrer ao uso de *Intents* que permitam comunicação em tempo real com serviços e/ou outro tipo de aplicações internas do *Android*, para facilitar a gestão da *Smart Home*;
- Capturar valores gerados pelos sensores dos dispositivos, bem como analisar e processar a respetiva informação gerada;
- Armazenar a informação recolhida e garantir o acesso à mesma em tempo real.

### 1.3 Ferramentas Utilizadas

Neste projeto, o grupo decidiu usar o *Gitlab* da *ESTG*, para garantir o controlo de versões, armazenar o código produzido, bem como facilitar a organização da nossa aplicação para que todos tenham acesso, em tempo real, ao mesmo.

Para a implementação do código, foi usado o *Android Studio*, previamente indicado pelo docente da unidade curricular, recorrendo à linguagem de programação *Kotlin*.

Também foi fornecido pelo docente da unidade curricular *API's* que permitiram ajudar e ter acesso a informação útil para o desenvolvimento do projeto.

### 1.4 Estrutura do Documento

Capítulo	Descrição
<b>1 - Introdução</b>	Apresentação do âmbito do projeto, bem como recursos e ferramentas usadas na conceção do projeto.
<b>2 - Planeamento</b>	Descrição do planeamento do projeto desde o levantamento de requisitos, passando pelo design dos Mockups, até à identificação de funcionalidades.
<b>3 - Implementação</b>	Explicação da estrutura adotada no projeto, detalhe dos mecanismos usados no Android, levantamento de problemas e de soluções.
<b>4 - Conclusão</b>	Apreciação crítica dos elementos do grupo.
<b>5 - Referências Bibliográficas</b>	Fontes onde foi procurada informação.
<b>6 - Anexos</b>	Informação Adicional (links e Layouts)

## 2. Planeamento

### 2.1 Identificação dos Requisitos

<b>ID: RF-1</b>	<b>Nome: Gestão de utilizadores</b>	<b>Prioridade: Alta</b>
-----------------	-------------------------------------	-------------------------

**Descrição:** O sistema deverá permitir operações CRUD de utilizadores.

**Verificações:** O sistema deverá verificar se:

- Para Inserção (Create):
  - Todos os campos obrigatórios do formulário estão preenchidos.
  - Todos os dados estão no formato esperado.
  - O email não existe na base de dados.
  - A password e a password de confirmação coincidem.
- Para as restantes:
  - Existência de utilizador

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a operação.

**Restrições:** Sem restrições.

<b>ID: RF-2</b>	<b>Nome: Identificação da casa</b>	<b>Prioridade: Alta</b>
-----------------	------------------------------------	-------------------------

**Descrição:** O sistema, aquando da primeira criação de conta associada a uma casa, fornece um Qr-code associado à identificação da casa, que permite a ligação à mesma aquando do scan. O Qr-code estará disponível no utilizador Administrador.

**Verificações:** O sistema deverá verificar se:

- O Qr-code existe

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado

<b>ID: RF-3</b>	<b>Nome: Gestão de permissões</b>	<b>Prioridade: Média</b>
-----------------	-----------------------------------	--------------------------

**Descrição:** O Administrador pode gerir as permissões das outras contas. O tipo de permissões existentes são: Administrador, Residente (Adulto / Infantil), Visitante.

**Verificações:** O sistema deverá verificar se:

- Permissões do utilizador

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado

<b>ID: RF-4</b>	<b>Nome: Gestão de dispositivos</b>	<b>Prioridade: Alta</b>
-----------------	-------------------------------------	-------------------------

**Descrição:** O sistema permite operações CRUD de dispositivos.

**Verificações:** O sistema deverá verificar se:

- Dispositivo já existe
- Dispositivo é compatível com o sistema

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado

<b>ID: RF-5</b>	<b>Nome: Visão geral da casa</b>	<b>Prioridade: Alta</b>
-----------------	----------------------------------	-------------------------

**Descrição:** O sistema permite agrupar dispositivos por categorias / divisões.

**Verificações:** O sistema deverá verificar se:

- Dispositivo conectados
- Divisão está criada

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado

<b>ID: RF-6</b>	<b>Nome: Receber e analisar informação pelos sensores</b>	<b>Prioridade: Média</b>
-----------------	---	--------------------------

**Descrição:** O sistema recebe e analisa informação fornecida pelos sensores, apresentando aos utilizadores a informação processada ( mapa da casa/ temperaturas / consumos/etc)

**Verificações:** O sistema deverá verificar se:

- Sensores para recolha de informação estão disponíveis

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e existência de sensores



<b>ID: RF-7</b>	<b>Nome: Configurações pré-definidas e automatizações</b>	<b>Prioridade: Média</b>
-----------------	---	--------------------------

**Descrição:** O sistema apresenta configurações segundo fatores exteriores como dia/noite/frio/calor/hora.

**Verificações:** O sistema deverá verificar se:

- Sensores para recolha de informação estão disponíveis
- Hora

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e existência de sensores

<b>ID: RF-8</b>	<b>Nome : Configurações pessoais</b>	<b>Prioridade: Baixa</b>
-----------------	--------------------------------------	--------------------------

**Descrição:** O sistema permite associar configurações pessoais de dispositivos a cada conta.

**Verificações:** O sistema deverá verificar se:

- Existem dispositivos conectados

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e dispositivos associados

<b>ID: RF-9</b>	<b>Nome: Notificações</b>	<b>Prioridade: Baixa</b>
-----------------	---------------------------	--------------------------

**Descrição:** O sistema notifica os utilizadores sobre vários tipos de alertas, sejam de perigo, consumo e outras informações

**Verificações:** O sistema deverá verificar se:

- Estado do casa

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e dispositivos associados

<b>ID: RF-10</b>	<b>Nome : Idioma segundo a localização</b>	<b>Prioridade: Alta</b>
------------------	--	-------------------------

**Descrição:** O sistema assume o idioma consoante a localização do utilizador

**Verificações:** O sistema deverá verificar se:

- Localização está ligada

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado

<b>ID: RF-11</b>	<b>Nome:</b> <b>Localização dos residentes menores</b>	<b>Prioridade: Baixa</b>
------------------	--	--------------------------

**Descrição:** O sistema permite localizar no mapa utilizadores

**Verificações:** O sistema deverá verificar se:

- Tipo de utilizador a localizar

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e apenas utilizador infantil pode ser localizado

<b>ID: RF-12</b>	<b>Nome:</b> <b>Assistente pessoal</b>	<b>Prioridade: Baixa</b>
------------------	--	--------------------------

**Descrição:** O sistema reconhece comandos de voz

**Verificações:** O sistema deverá verificar se:

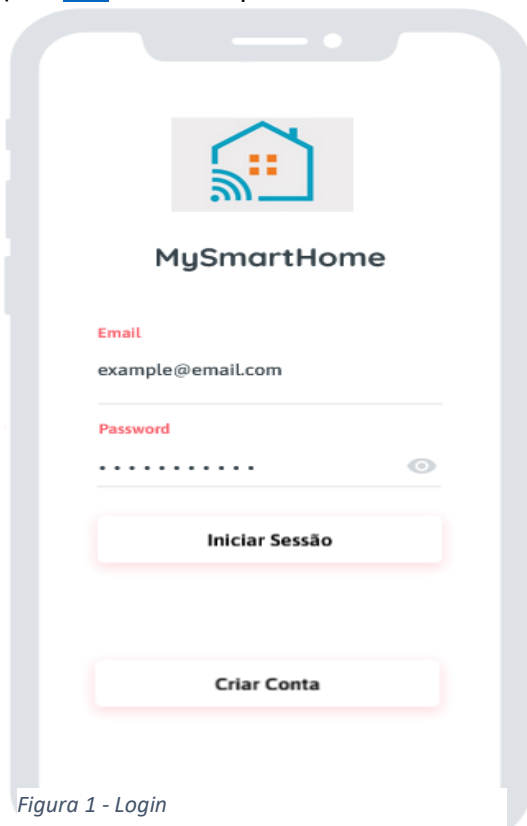
- Autorização para utilizar microfone está aprovada

No caso de algum dos pontos acima referidos não se verifique, o sistema deverá emitir um alerta e não permitir a criação da conta.

**Restrições:** Utilizador autenticado e comandos adicionados

## 2.2 Mockups

(Ver [link](#) com a sequência das mesmas mockups)



**MySmartHome**

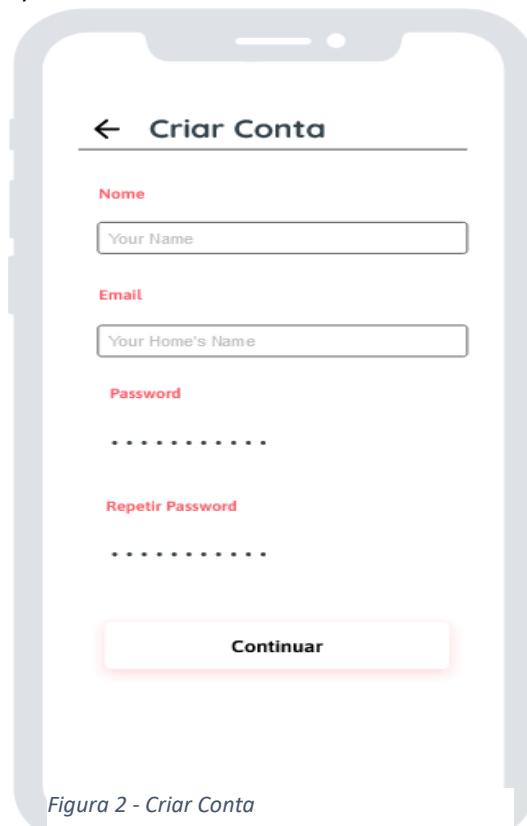
Email  
example@email.com

Password  
.....

Iniciar Sessão

Criar Conta

Figura 1 - Login



← Criar Conta

Nome  
Your Name

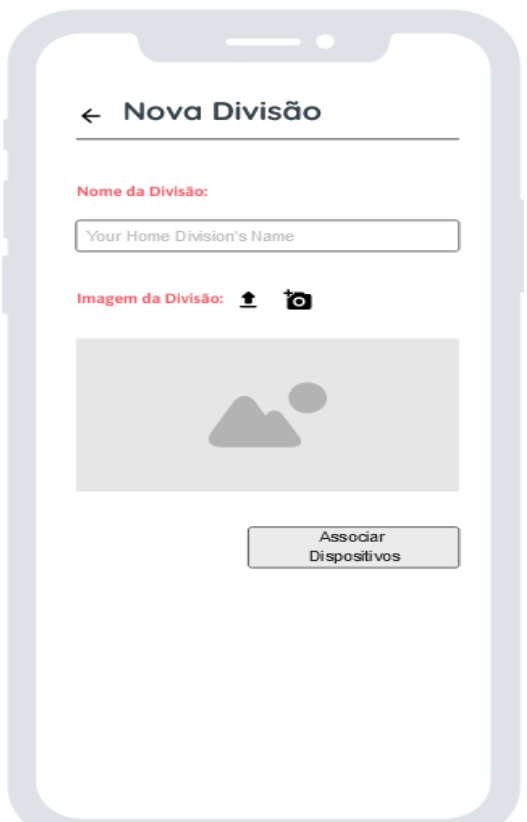
Email  
Your Home's Name

Password  
.....

Repetir Password  
.....


Continuar

Figura 2 - Criar Conta



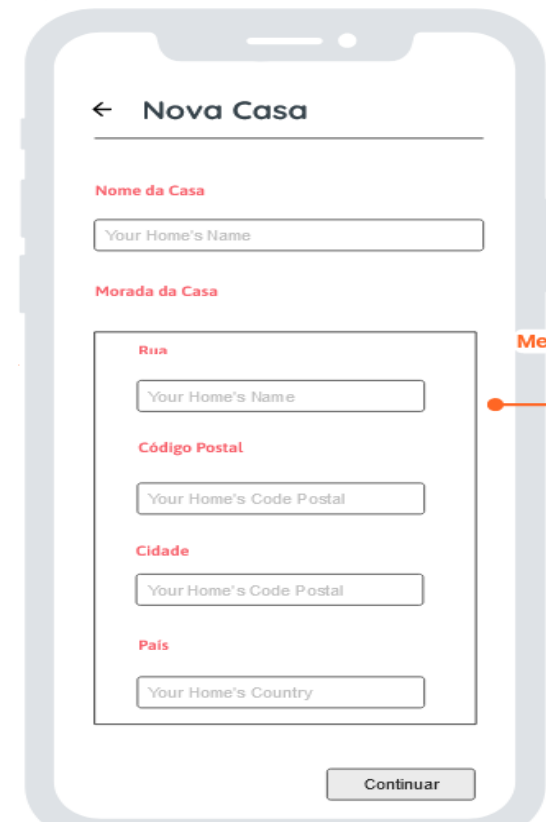
← Nova Divisão

Nome da Divisão:  
Your Home Division's Name

Imagem da Divisão: 

Associar Dispositivos

Figura 3 - Adicionar Divisão



← Nova Casa

Nome da Casa  
Your Home's Name

Morada da Casa

Rua  
Your Home's Name

Código Postal  
Your Home's Code Postal

Cidade  
Your Home's Code Postal

País  
Your Home's Country

Continuar

Figura 4 - Adicionar Casa

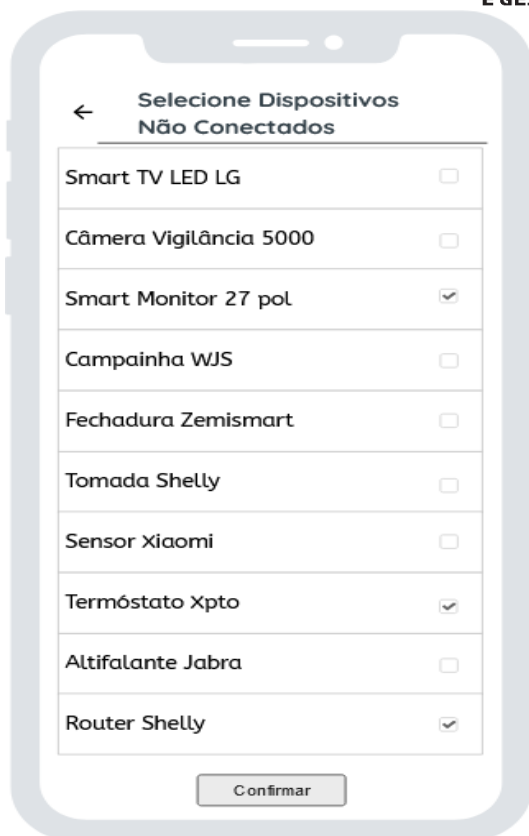


Figura 5 - Selecionar Dispositivo a Conectar



Figura 6 – Ecrã Principal da Aplicação

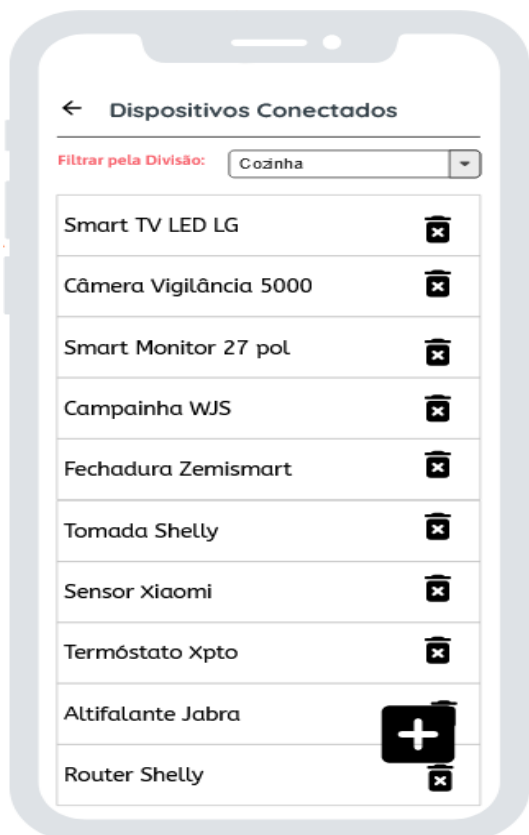


Figura 7 - Lista de Dispositivos Conectados

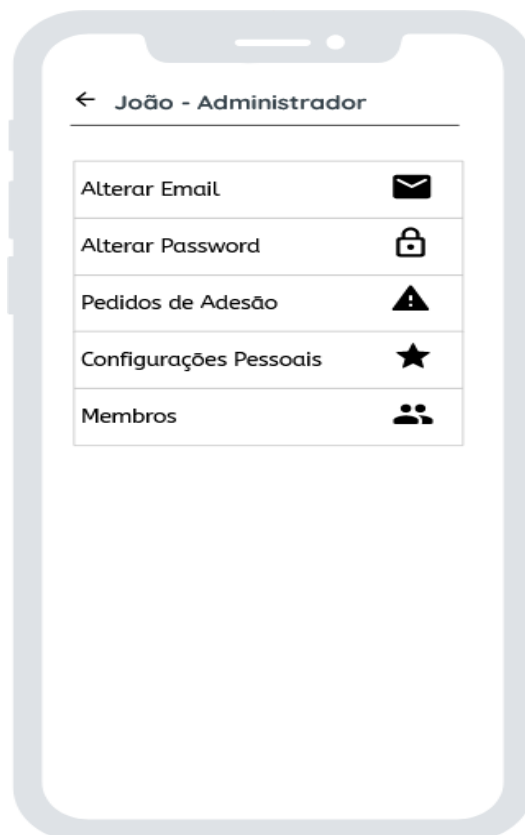


Figura 8 - Perfil do Utilizador

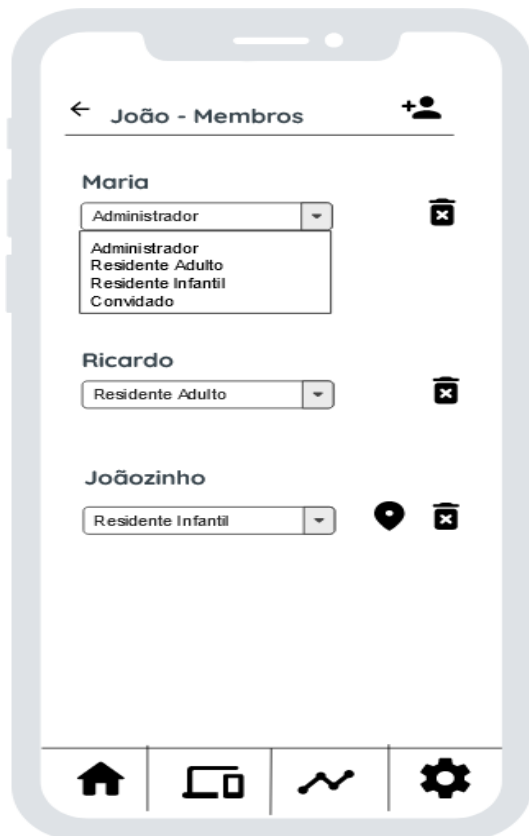


Figura 9 - Atribuição de Permissões

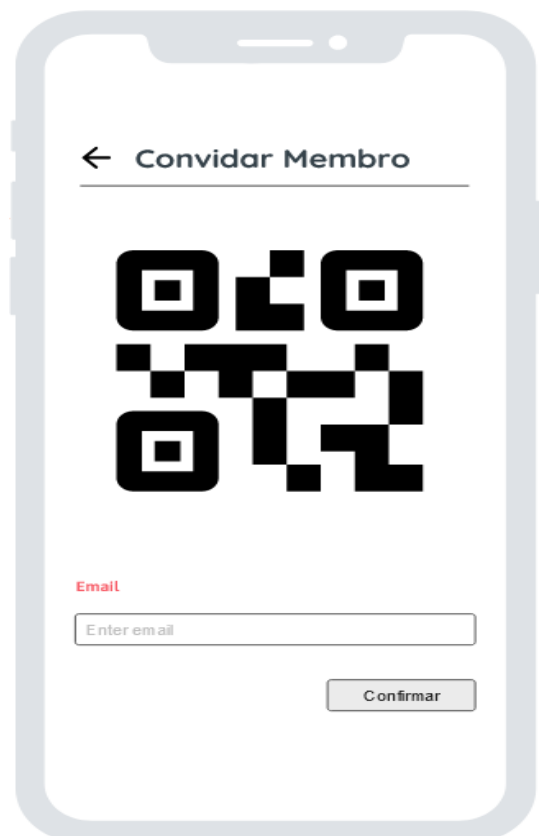


Figura 10 - QR Code para aceder à casa

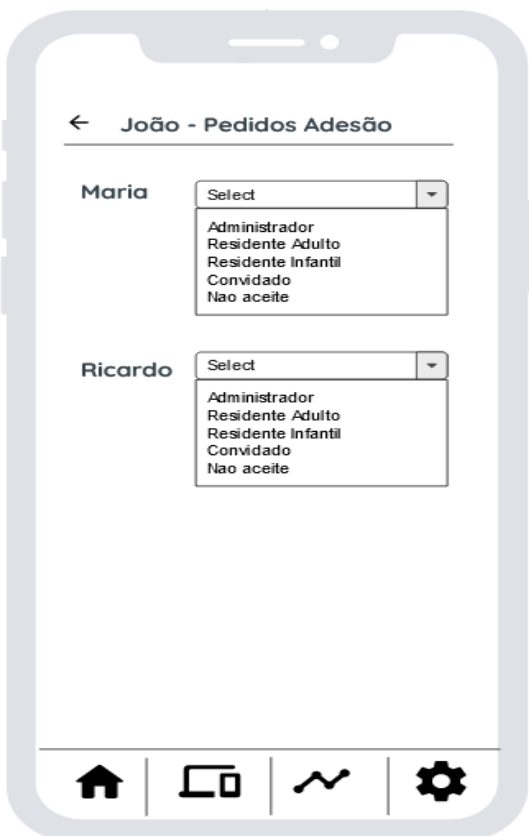


Figura 11 - Lista de Pedidos de Adesão (Administrador)



Figura 12 - Vista de uma divisão

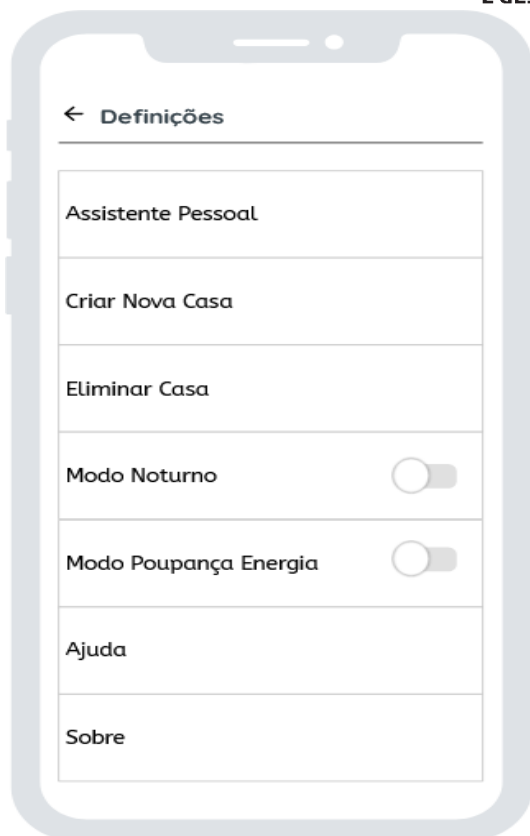


Figura 13 - Definições

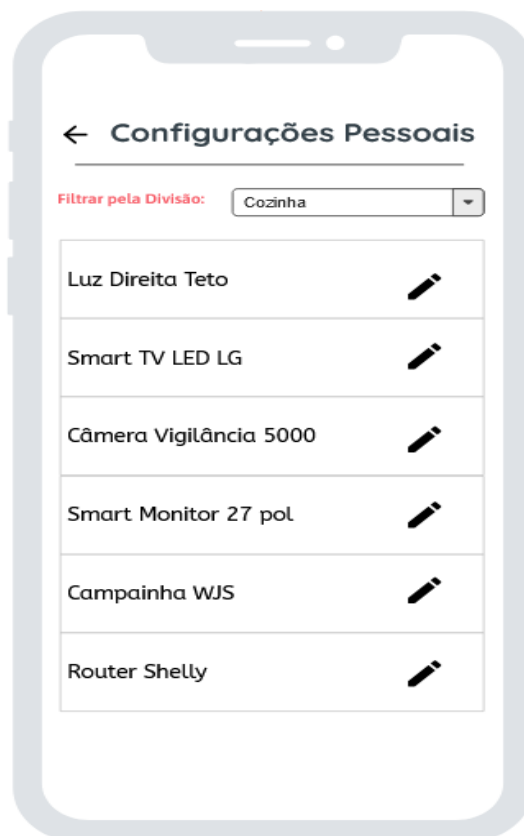


Figura 14 - Configurações pessoais do utilizador

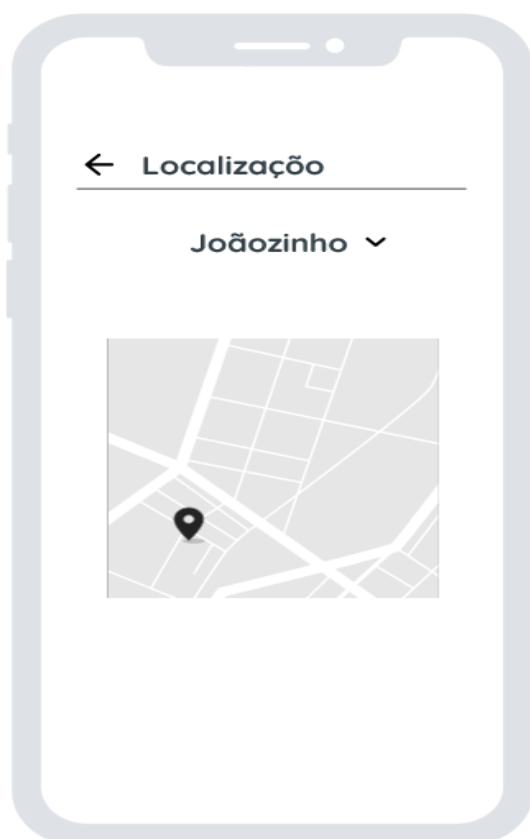


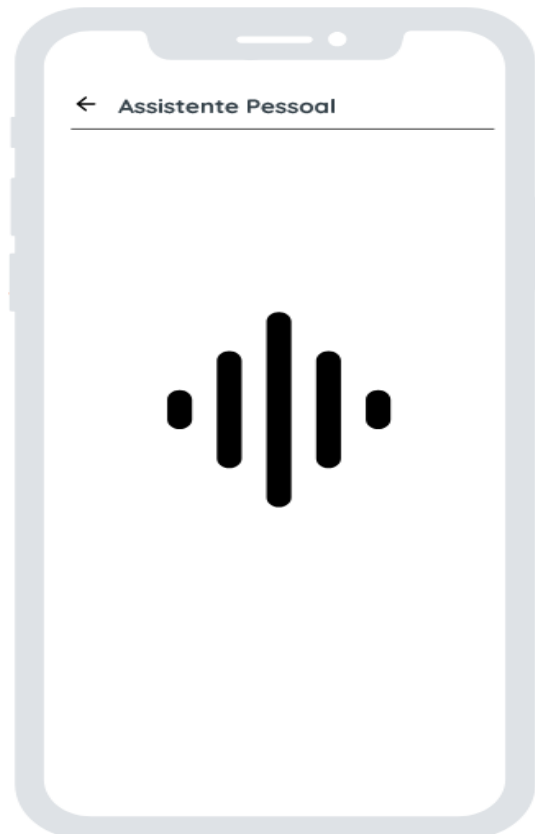
Figura 15 – Localização de Residente Infantil



Figura 16 - Configuração de um dispositivo



*Figura 17 - View com vídeo com tutorial*



*Figura 18 – Assistente Pessoal*

## 2.3 Estrutura de Dados

O seguinte diagrama ilustra a forma como os dados ficaram representados:

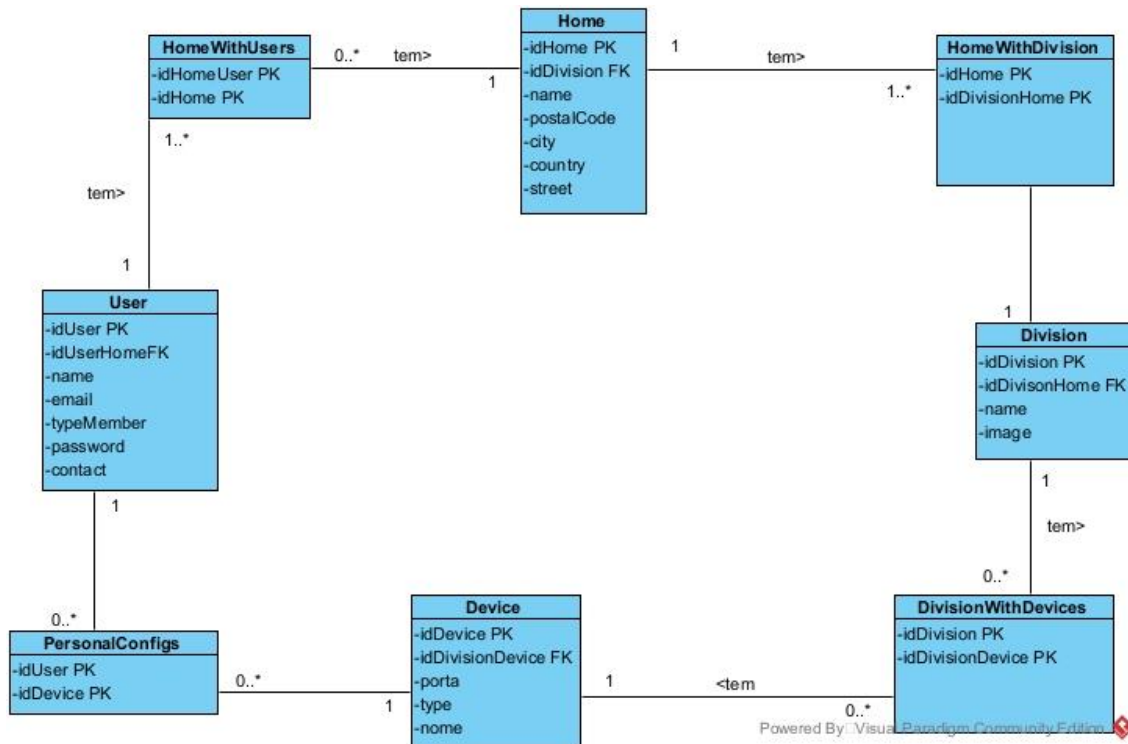


Figura 19 - Diagrama de Classes

## 2.4 Estrutura do Retrofit

O grupo de trabalho, neste momento, apenas está a receber informação relativa a lâmpadas, recebendo os seguintes campos:

- ssid
- isOn
- mode
- temperature
- overTemperature



### 3. Implementação

#### 3.1 Estrutura do Projeto

Durante a fase inicial de implementação do projeto, o grupo decidiu adotar uma estrutura universal, aconselhada pelo docente da unidade curricular, durante as aulas práticas, que consiste em dividir as classes / ficheiros por *Packages*.

Isto, no sentido de simplificar ao máximo todo o código implementado, bem como de transmitir uma boa organização do projeto. Tendo isso em conta, o grupo decidiu adotar a seguinte estrutura:

- **Database:** Trata-se de um package que contém todas as entidades, queries e respetivas operações à Base de Dados, mais precisamente ao **Room**.
- **Retrofit:** Consiste num package responsável por conter todas as chamadas à API, bem como a estruturação dos dados fornecidos pela mesma.
- **Enums:** Package que dispõe das enumerações usadas no projeto.
- **UI:** Caracteriza-se como sendo um package que tem todos os componentes usados no projeto que facilitam a perceção do código e podem ser reutilizados quando necessário. Contém também **Screens** que são as classes que contêm o aspeto visual da aplicação.

#### 3.2 Controlo de Versões

Desde o planeamento inicial do projeto, o grupo decidiu adotar um sistema que permitisse garantir o controlo de versões, tirando partido do conhecimento que fora instruído ao longo dos anos pelos docentes de várias unidades curriculares.

Tendo isso em conta, o grupo decidiu pôr em prática o mais conhecido sistema de controlo de versões denominado de **GITLAB**.

Este é responsável por facilitar o armazenamento do código implementado, bem como garantir que vários membros possam trabalhar ao mesmo tempo, garantindo que, caso haja conflitos, estes possam ser resolvidos, e assim, evitar perdas de dados.

Nesse sentido, cada individuo sempre que necessitou de implementar novas funcionalidades, criou uma **Branch** pessoal, procedeu às alterações necessárias e assim que tivesse terminado de fazer **Commits**, fazia **Merge** para a Branch Master, isto para garantir que na Branch **Master** o código esteja sempre atualizado.

Segue-se um pequeno excerto do grupo no Gitlab:

Name	Last commit	Last update
.idea	New Division Camera and FileUpload	2 weeks ago
app	Components Final	14 hours ago
gradle/wrapper	Estrutura do projeto	1 month ago
.gitignore	Estrutura do projeto	1 month ago
README.md	add README	1 month ago
build.gradle	Estrutura do projeto	1 month ago

Figura 20- Página do grupo no Gitlab

O grupo fez questão de no início do projeto convidar o Docente para ter acesso a esta página do Gitlab, no entanto, e caso não consiga ter acesso por algum motivo, abaixo encontra-se o link do repositório.

**Link do Repositório:** [https://gitlab.estg.ipp.pt/cmu\\_grupo131/mysmarthome](https://gitlab.estg.ipp.pt/cmu_grupo131/mysmarthome)

### 3.3 Definição de Layouts

Durante a fase de planeamento do projeto, o grupo decidiu que deveria ter em consideração dar prioridade à fácil utilização da aplicação.

Nesse sentido, a decisão tomada foi evitar que, o utilizador tivesse de dar vários cliques para aceder a um determinado ponto da aplicação.

Com isso em mente, foi possível criar ecrãs que tivessem apenas a informação necessária e garantir que o utilizador pudesse navegar sem dificuldades entre os vários ecrãs da aplicação.

Outro aspeto a ter em consideração tanto para facilitar o uso da mesma, como para garantir que o utilizador pudesse usar a aplicação em qualquer dispositivo, o grupo decidiu aplicar o conhecimento passado durante as aulas, e tornar a aplicação disponível para Ecrãs Pequenos, Médios e Grandes.

Mas, o grupo não ficou restringido apenas a isso, optou também por garantir que a aplicação fosse utilizada para qualquer orientação de ecrã (horizontal e vertical).

No entanto, e por motivos de não estender em demasia o tamanho do presente relatório, apenas estarão representados os layouts de cada screen, para ecrã pequeno (PIXEL 2), fornecido como exemplo da primeira aula prática de CMU, na orientação vertical.

O grupo gostaria também de salientar que para já apenas se está a considerar existir apenas um tipo de permissão: **Administrador**.

Para consultar os layouts da *MySmartHome*, deve aceder ao capítulo: *ANEXOS*.

### 3.4 Identificação dos mecanismos utilizados

#### 3.4.1 Intents

- Câmara

No sentido de possibilitar ao utilizador da aplicação tirar uma foto com a câmara do seu dispositivo e poder realizar uma determinada ação com a mesma, como por exemplo, guardá-la na base de dados, foi necessário recorrer a Jetpack Activity Contracts.

Esta é uma forma muito útil para obter ficheiros de qualquer tipo, e funciona quase como uma espécie de *intent*, e que serve para permitir que o utilizador faça pedido ao sistema para que possa ligar a sua câmara para, por exemplo, tirar uma fotografia.

Um exemplo onde esse mecanismo foi utilizado, foi no ecrã de Nova Divisão.

Abaixo, segue um excerto de como fazer pedido ao sistema, para abrir a funcionalidade da câmara:

```
val launcher =  
    rememberLauncherForActivityResult(ActivityResultContracts.TakePicturePreview()) {  
        result.value = it  
    }
```

Para executar este pedido é necessário fazer launch do mesmo: *launcher.launch()*

No entanto, como esta operação envolve a câmara, é necessário garantir que as permissões foram aceites pelo utilizador. Para isso no ficheiro *AndroidManifest* tiveram de ser atribuídas permissões para o efeito.

```
<uses-permission android:name="android.permission.CAMERA"
    android:required="true"
    android:requiredFeature="true"/>
```

```
<uses-feature android:name="android.hardware.camera.any"
    android:required="true" />
```

```
(...)
```

```
<action android:name="android.media.action.IMAGE_CAPTURE" />
```

```
<category android:name="android.intent.category.LAUNCHER" />
```

Por outro lado, e para dar upload de ficheiros, como por exemplo de uma imagem, recorreu-se ao mesmo procedimento, mas neste caso interessa retornar o URI do ficheiro, onde este está armazenado no sistema:

```
val imgPicker = rememberLauncherForActivityResult(
    contract = ActivityResultContracts.GetContent(),
    onResult = { uri ->
        hasImg = uri != null
        imgUri = uri
    })
```

Para este último caso, o grupo fez uso da biblioteca aconselhada pelo docente da UC, denominada de *Coil*, que permite dar *display* de uma imagem assincronamente, ou seja, quase instantaneamente. Para isso foi necessário atualizar o ficheiro *build.gradle* com a dependência necessária para o efeito.

- SMS

Relativamente ao ecrã de Membros, o grupo achou necessidade de ter uma funcionalidade que permitisse enviar algum tipo de mensagem de texto a um membro da casa, para os mais variados motivos, tais como por exemplo para alertar/avisar o membro da casa de alguma situação inesperada.

Para isso recorreu-se ao uso de *intents* de mensagens de texto, para compor uma determinada mensagem, para o contacto telefónico de um determinado membro.

O que acontece, é que o utilizador preenche o campo da mensagem que pretende enviar, e é encaminhado para a aplicação de mensagens do sistema, onde só deve clicar em “Enviar”.

Para isso foi necessário adicionar a permissão de enviar mensagem de texto no ficheiro de strings: `<action android:name="android.intent.action.SEND" />`

- Call

Também no ecrã de Membros, foi necessário dispor da funcionalidade de Telefonar a uma determinada pessoa pertencente à casa. Para esse efeito, a *intent* de fazer um telefonema foi útil neste contexto.

No entanto, para além de serem necessárias permissões do utilizador:

```
<uses-permission android:name="android.permission.CALL_PHONE" />
```

```
<uses-permission android:name="android.permission.INTERNET" />
```

Foi necessário fazer parse do número de telefone, para executar a *intent*, e garantir que o utilizador deixa fazer este tipo de ação (ou seja, se este dá permissões de aceder à atividade de contactos).

- Email

No ecrã de Convidar Membro, existe a possibilidade de enviar um email, para um ou vários destinatários no sentido de convidar estes para a casa.

Além disso estes, podem anexar um ficheiro como o *QR Code*, gerado pela casa, para que os convidados tenham acesso à mesma.

Nesse sentido, o grupo fez questão de dar uso a *intents* mais uma vez para enviar email. Ou seja, fez-se uso da aplicação de mensagens do sistema para o efeito.

No entanto, para que fosse possível anexar ficheiros (upload) de qualquer tipo foi necessário fazer uma modificação relativamente a ficheiros do tipo imagem apenas:

```
val filePicker = rememberLauncherForActivityResult(  
    contract = ActivityResultContracts.GetContent(),  
    onResult = { uri ->  
        hasFile = uri != null  
        fileUri = uri  
    })
```

(...)

```
filePicker.launch(*/*)
```

- Calendar

O uso da agenda foi um extra que o grupo decidiu que a aplicação tivesse.

Isto no sentido, de ser possível agendar eventos interinos da própria casa, como todos os membros decidirem agendar a ida a um evento público como assistir a um filme no cinema, entre outros.

Para isso, recorreu-se à *intent* de agendar evento na aplicação de calendário do sistema.

Foi necessário declarar um *intent-filter*, para que isso fosse possível, (consultar código no repositório do gitlab).

- Alarm Clock

O uso do alarme, foi precisamente um extra como o uso da agenda referida acima.

Tratou-se de uma utilidade extra para o utilizador marcar um alarme no sistema, para acordar ou para qualquer outro efeito.

Foram necessárias permissões, que constam no ficheiro *AndroidManifest*, e declaradas *intent-filter* para que o uso do alarme fosse possível.

- Location on Map

Esta foi outra *intent* extra adotada pelo grupo do projeto para situar a própria casa no mapa. É uma *intent* útil para definir percursos, entre outras necessidades.

Para isso foi necessário atualizar o *intent-filter* para que isso fosse possível.

### 3.4.2 Notificações

Atualmente em quase todas as aplicações android, o uso de notificações é constantemente utilizado pelos programadores.

Isto porque permitem que os utilizadores recebam alertas, avisos e que estes possam tomar decisões ou que possam redirecionar-se para outra parte da aplicação apenas com um click de um botão numa notificação.

A MySmartHome dispõe de notificações de alerta, para que os utilizadores acedam a links externos, por exemplo. Estas notificações podem também ser úteis, no sentido de avisar um utilizador que algo aconteceu.

Portanto, para criar uma notificação foi necessário, acima de tudo, ter um canal específico para que um grupo restrito de utilizadores tenham acesso à notificação emitida.

Esta pode adotar várias estruturas, como ter um título, uma descrição, um ícone, uma prioridade (que deve ser preferencialmente Default), tal como constam as boas práticas, pode ser removida automaticamente e pode ter até 3 botões que realizam um determinado tipo de operação. Tais operações podem fazer uso de intents, ou apenas de reencaminhamento para outras partes do sistema ou da aplicação.

#### Excerto da Notificação de Receber Boas-Vindas após ter feito Login

```
fun notification_logged_in() {  
    val CHANNEL_ID = "007"  
    // Create an explicit intent for an Activity in your app  
    val intent = Intent(Intent.ACTION_VIEW,  
        Uri.parse("https://www.worten.pt/smart-home-e-redes/smart-home/tudo-sobre-smart-home"))  
    val pendingIntent: PendingIntent = PendingIntent.getActivity(context: this, requestCode: 0, intent, flags: 0)  
    var builder = NotificationCompat.Builder(context: this, CHANNEL_ID)  
        .setSmallIcon(R.drawable.ic_launcher_background)  
        .setContentTitle("Início de Sessão")  
        .setStyle(  
            NotificationCompat.BigTextStyle()  
                .bigText("Seja Bem-Vindo(a)! \nAcabou de iniciar sessão na aplicação MySmartHome! \n" +  
                    "Disfrute ao máximo da gestão da sua casa inteligente!")  
        )  
        .setPriority(NotificationCompat.PRIORITY_DEFAULT)  
        .setContentIntent(pendingIntent)  
        .setAutoCancel(true)  
        .addAction(R.drawable.ic_launcher_foreground, title: "Adquirir Dispositivos", pendingIntent)  
}
```

Figura 3- Notificação de Boas-Vindas

#### 3.4.3 Navegação

Para que fosse possível navegar entre ecrãs diferentes da aplicação, foi necessário ter algum tipo de navegação, quer seja através do click de botões, quer seja pelo uso da Topbar, da Bottombar, etc.

Ou seja, foi necessário declarar todos os ecrãs num ficheiro que serviu como componente de navegação.

Quando a aplicação é executada, na *MainActivity*, este componente é primeiramente invocado para que esteja disponível a navegar de ecrã para ecrã.

Para tal ser possível foi necessário definir uma *NavHost*, declarando como página inicial a página de Login, e dentro estão declarados todos os ecrãs existentes e a respetiva rota usada para aceder a cada um, tal como demonstra a figura abaixo representada:

```

@Composable
fun Navigation(mainActivity: MainActivity) {

    val navController = rememberNavController()

    NavHost(navController = navController, startDestination = "LoginScreen") { this: NavGraphBuilder
        composable( route: "LoginScreen") { it: NavBackStackEntry
            LoginScreen( navController = navController)
        }
        composable( route: "HomePageScreen") { it: NavBackStackEntry
            HomePageScreen(mainActivity, navController= navController)
        }
        composable( route: "ConnectedDevicesScreen") { it: NavBackStackEntry
            ConnectedDevicesScreen( navController= navController)
        }
        composable( route: "ProfileScreen") { it: NavBackStackEntry
            ProfileScreen( navController= navController)
        }
    }
}

```

Figura 4- Componente de Navegação

Ou seja, cada ecrã recebe como parâmetro sempre, a `navController` que permite receber a instância usada para fazer `navigate` para o ecrã que se pretende ir.

Por exemplo num click de um botão de um determinado ecrã:

```
navController.navigate("NewAccountScreen")
```

#### 3.4.4 Google Maps

Na presente aplicação, o grupo decidiu inicialmente que um sistema de localização seria extremamente importante, para que membros da casa, que fossem crianças, ou até mesmo no caso de membros incapacitados (doença, deficiência, entre outros) pudessem ser localizados através do *Google Maps*.

Isto é uma mais-valia da *MySmartHome*, porque possibilita o controlo e a supervisão deste tipo de membros da casa, que têm menos capacidade de noção dos riscos inerentes ou até mesmo fraca capacidade de mobilidade pelos motivos acima referidos.

Tendo isso em conta, foi necessário adotar uma *library* do *Google Maps*, que permite ter acesso aos mapas, acesso a percursos, distâncias e outros mecanismos importantes para garantir que este tipo de membros, fiquem seguros mesmo à distância.

Ou seja, para que esta funcionalidade fosse suportada o grupo teve de primeiramente atualizar o ficheiro *build.gradle*, adicionando as seguintes dependências:

```
implementation 'com.google.maps.android:maps-compose:1.0.0'
implementation 'com.google.android.gms:play-services-maps:18.0.2'
```

Depois, no browser, deve-se inserir o seguinte url:

[console.cloud.google.com](https://console.cloud.google.com)



Após isso, deve-se criar um novo projeto:

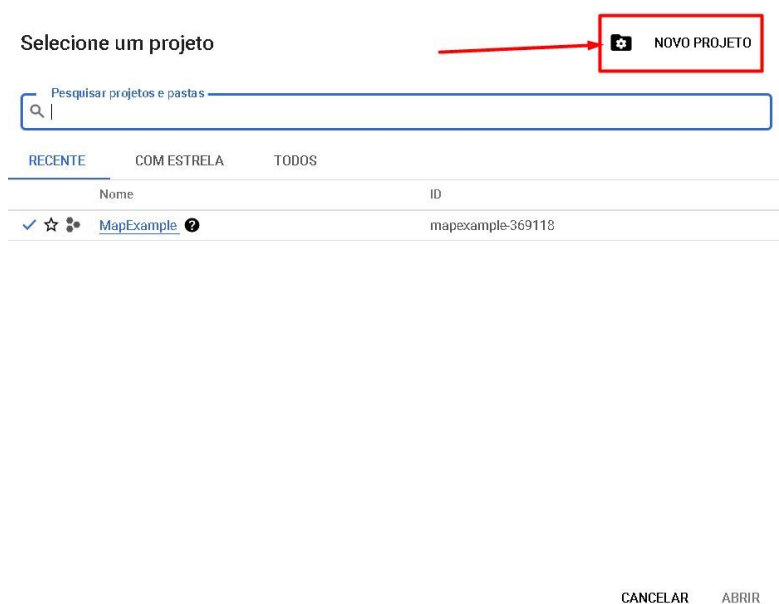


Figura 5- Criar novo projeto Google Cloud

Atribuir um nome ao mesmo:

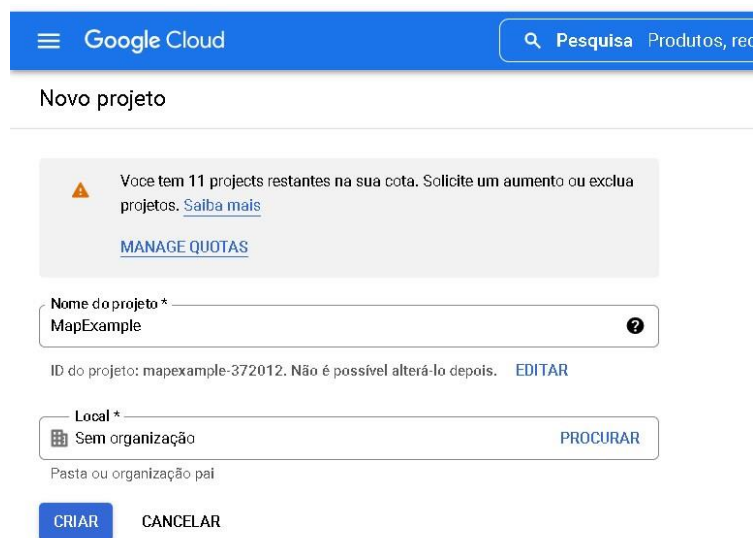


Figura 6- Nome do Projeto (Google Cloud)

Seguidamente, deve-se aceder ao menu lateral à opção APIs e Serviços Ativados:

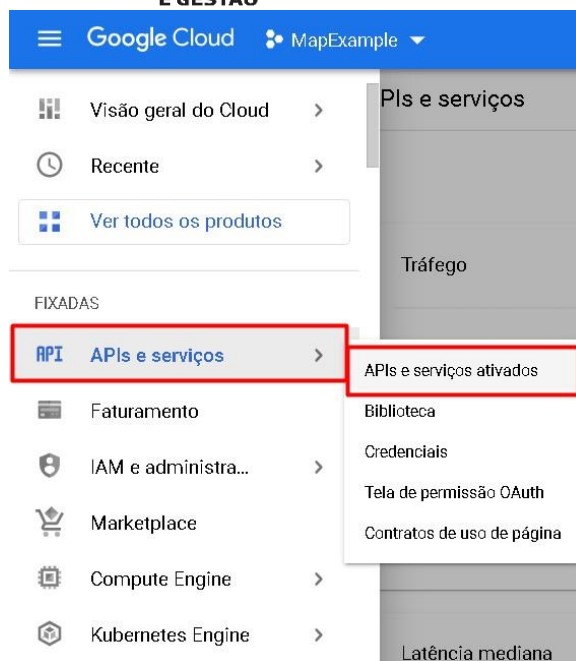


Figura 7- APIs e Serviços Ativados (Google Cloud)

Selecionar Ativar APIs e Serviços:

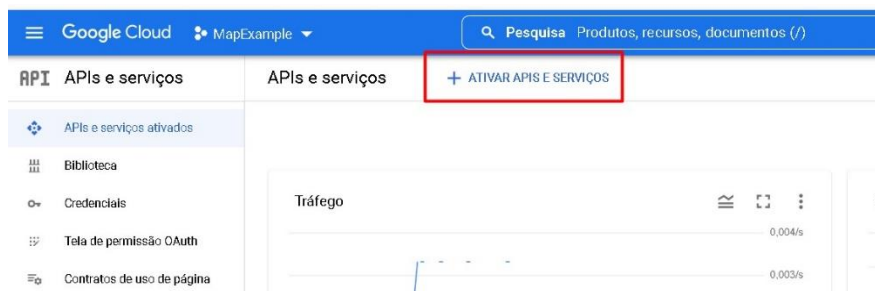


Figura 8- Ativar APIs e Serviços (Google Cloud)

Na caixa de pesquisa deve-se inserir Maps Android e selecionar a opção:

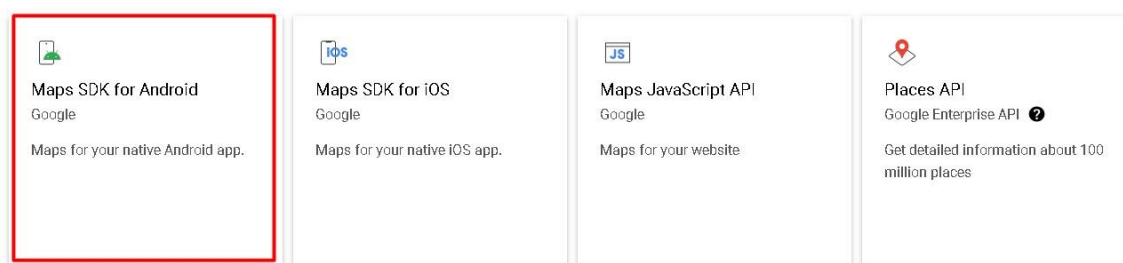


Figura 9- Maps Android (Google Cloud)

Ativar o Serviço, clicando no botão para o efeito:

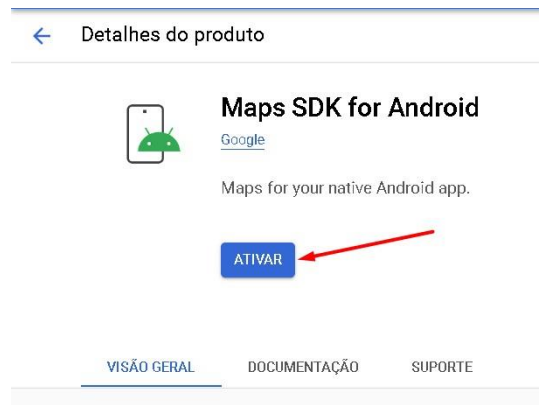


Figura 10- Ativar Serviço (Google Cloud)

Aceder à opção Credenciais:

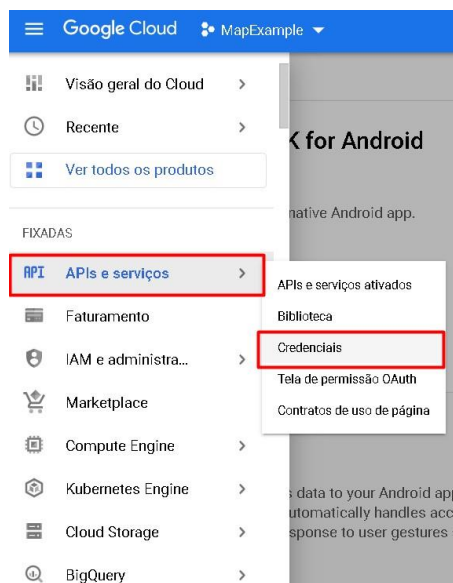


Figura 11- Credenciais (Google Cloud)

Criar Nova Credencial, selecionar Chave de API:

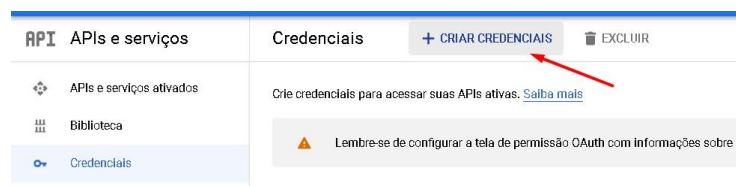




Figura 12- Nova Credencial (Google Cloud)

Exibir chave para poder copiar a mesma para o ficheiro de strings do projeto:

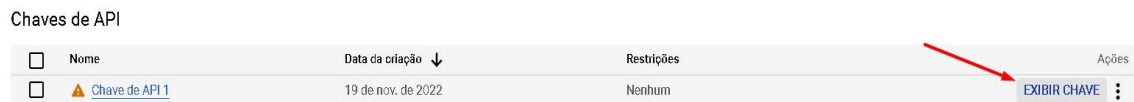


Figura 13- Exibir Chave (Google Cloud)

Selecionar opção de copiar:

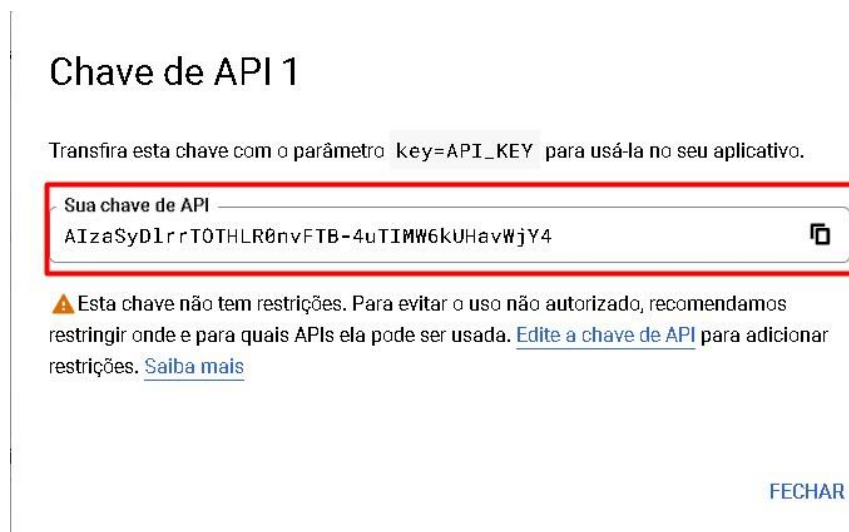


Figura 14- Copiar Chave (Google Cloud)

Por último deve-se atualizar o ficheiro de strings em res->values->strings, com a chave gerada:

```
<string name="googleMapsKey">AIzaSyD1rrT0THLR0nvFTB-4uTIMW6kUHavWjY4</string>
```

Figura 15- Ficheiro de Strings com a Chave do Google Cloud

### 3.4.5 Componentes

O uso de componentes é extremamente crucial numa aplicação mobile e web, pois facilita a reutilização de objetos visuais, permite uma melhor organização do código e facilita que, no caso de serem necessárias alterações num determinado objeto, seja possível alterar apenas um componente e não ter de editar várias classes que contenham esse objeto.

Nesse sentido, foi necessário criar alguns componentes que resumiram bastante não só as linhas de código utilizadas, como também a complexidade do projeto.

E por isso, o grupo decidiu que deveriam existir os seguintes componentes:

- Bottombar
- Topbar
- Textfields
- Buttons
- List's Rows
- CollapsibleExpandedMenus
- Texto
- Popups (apenas 1 tipo)
- Display Images

### 3.4.6 Room e Retrofit

No sentido de armazenar informação relativa à aplicação localmente, foi necessário recorrer ao uso do Room.

Foram criados:

<b>Entidades</b>	Home (com classe embutida Address) , Division, User, Devices, PersonalConfigs
<b>Relações</b>	DivisionWithDevices, HomeWithDivision, HomeWithUsers, DeviceWithUsers, UserWithDevices

Para esta entrega apenas apresentamos Login e Registo como exemplo.

Em relação às operações que fizemos à base de dados:

- DivisionDao: permite operações CRUD relativas a divisões
- HomeDAO: permite inserir, obter e eliminar uma casa, não sendo possível editar a mesma. Isto porque uma casa não vai alterar a sua morada, no caso de os utilizadores mudarem de casa/morada, criam uma nova casa.
- PersonalConfigDAO: permite operações CRUD relativas a configurações pessoais.
- UserDao: permite operações CRUD relativas a utilizadores.
- DeviceDAO: permite operações CRUD relativas a dispositivos.

É de salientar, que no caso de um dispositivo não estar associado a uma divisão, ou de um utilizador não estar associado a uma casa, o id da respetiva relação entre os dois tem valor 0.

Relativamente ao Retrofit, de momento, o grupo apenas está a obter dados de lâmpadas provenientes da API, sendo que para já outro tipo de operações ainda não foram implementadas.

### 3.5 Enumeração dos problemas encontrados

- Room – Definição das Relações entre Entidades necessárias

A identificação das Entidades foi (e ainda neste momento da entrega é) uma dificuldade encontrada. O grupo propõe a existência de configurações favoritas para cada utilizador relativamente a cada dispositivo. As relações entre Utilizador, Dispositivo e Configurações Pessoais, nesta fase, ainda não são claras a 100%.

Com a melhor implementação do Retrofit e trabalho no mesmo, o grupo acredita melhorar esta componente.

- Componentes – Reutilização de Popups e Textfields

Um dos maiores problemas que o grupo se deparou durante a implementação do projeto, foi na criação de componentes para Popups e Textfields.

Isto porque os Textfields envolvem inputs que estão constantemente a ser atualizados e houveram algumas dificuldades em retornar o valor inserido pelo utilizador para a classe onde o componente é invocado.

Isto para fazer verificações e condições quanto ao valor inserido em cada textfield.

No entanto, o grupo conseguiu ultrapassar essa dificuldade, retornando `textfield.text`, retornando o valor inserido sob a forma de String.

Por outro lado, outra dificuldade sentida relativa ao uso de componentes, foi a reutilização de popups, principalmente os que servem para inserir dados em textfields. Por esse motivo, o grupo apenas conseguiu criar um componente de popup que pudesse ser reutilizado, apenas com texto de aviso/alerta.

Desse modo, o grupo ainda não conseguiu, tal como pretendia ultrapassar essa barreira.

- Google Play Services – Representação de Mapas em alguns Emuladores

O grupo, como era de esperar, deparou-se com vários obstáculos e dificuldades durante a implementação do projeto.

Uma dessas dificuldades foi descobrir a solução para o problema que aparecia relativamente ao suporte para Mapas. Isto porque em alguns emuladores, como por

exemplo para dispositivos de médio tamanho e de grande tamanho, como por exemplo para tablets, o mapa não aparecia no ecrã, dado que ocorria um erro de Google Play Services.

Para isso e após várias pesquisas, veio a descobrir-se que poderia ser por causa desses dispositivos não possuírem o ícone de Google Play Store na criação dos respetivos emuladores.

E para isso seria necessário alterar o ficheiro **config.ini** situado no explorador de ficheiros no caminho:

**C:\Users\<MY\_USER>\.android\avd\<EMULATOR\_NAME>**

Essas alterações no ficheiro **config.ini**, deveriam ser as seguintes:

*PlayStore.enabled = true*

*image.sysdir.1=system-images\android-27\google\_apis\_playstore\x86\*

*tag.display=Google Play*

*tag.id=google\_apis\_playstore*

Contudo, após algum tempo o grupo verificou que apenas mudando a API usada pelo Android Studio de 30 (R), para 33 (Tiramisu), o problema seria resolvido e deixaria de ser necessário alterar o ficheiro de configurações dos respetivos emuladores.

### 3.5 Objetivos Futuros

O grupo pretende futuramente:

- obter os dados dos diferentes tipos de dispositivos suportados pela API
- implementar o assistente pessoal
- associação de casa através de QR code
- botão de logout
- vídeo de tutorial da aplicação
- permissões
- serviços em background
- autenticação

### 3.7 Resultados Obtidos

## 4. Conclusão



## 5. Referências Bibliográficas e Web

Slides da Unidade Curricular de CMU;

Biblioteca do Google Maps:

<https://developers.google.com/maps/documentation/android-sdk/maps-compose>

Intents em Android:

<https://developer.android.com/guide/components/intents-common>

Room:

<https://developer.android.com/training/data-storage/room>

Youtube (alguns tutoriais)

<https://www.youtube.com/@PhilippLackner>

## 6. Anexos

### 6.1 Mockups e Gitlab

[https://gitlab.estg.ipp.pt/cmu\\_grupo131/mysmarthome](https://gitlab.estg.ipp.pt/cmu_grupo131/mysmarthome)

<https://app.mogups.com/Gtqi0eHa5plb2u68Kk9B1tjrdT73V4zH/edit/page/a880590a1>

### 6.2 Apresentação dos Layouts

#### 6.2.1 Login

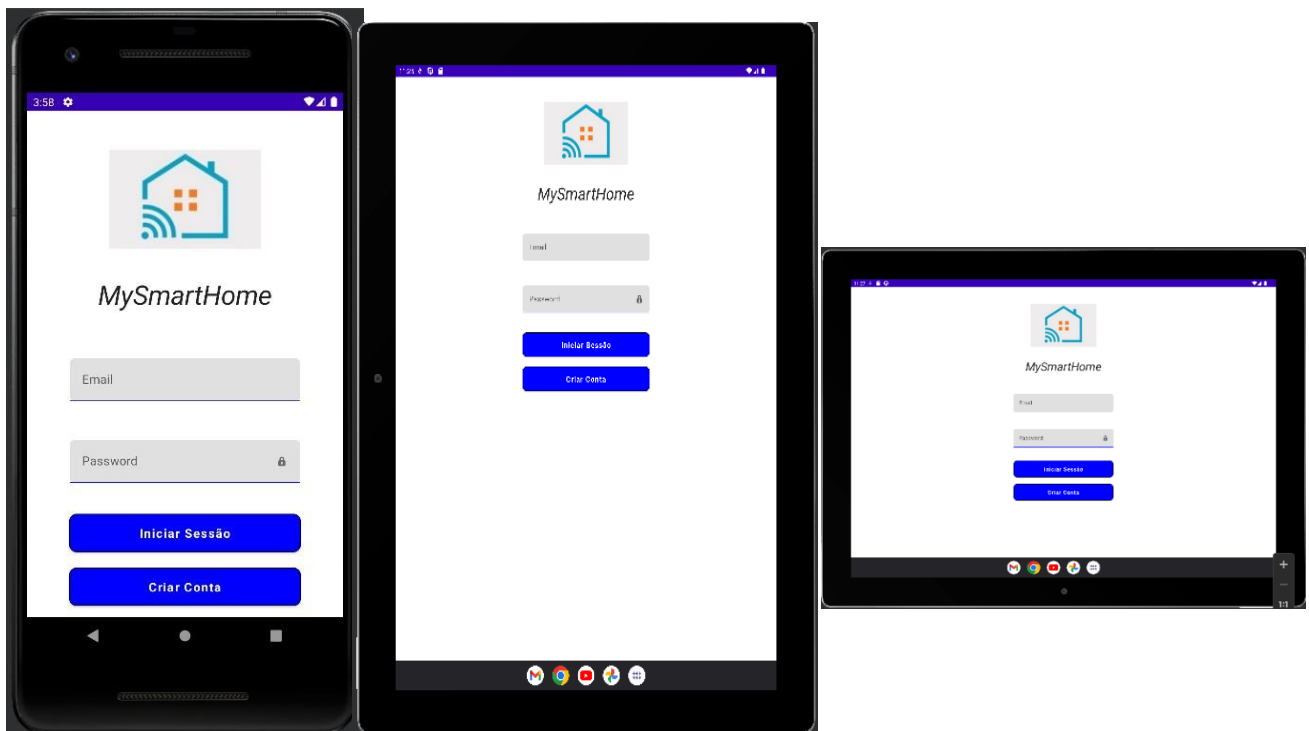


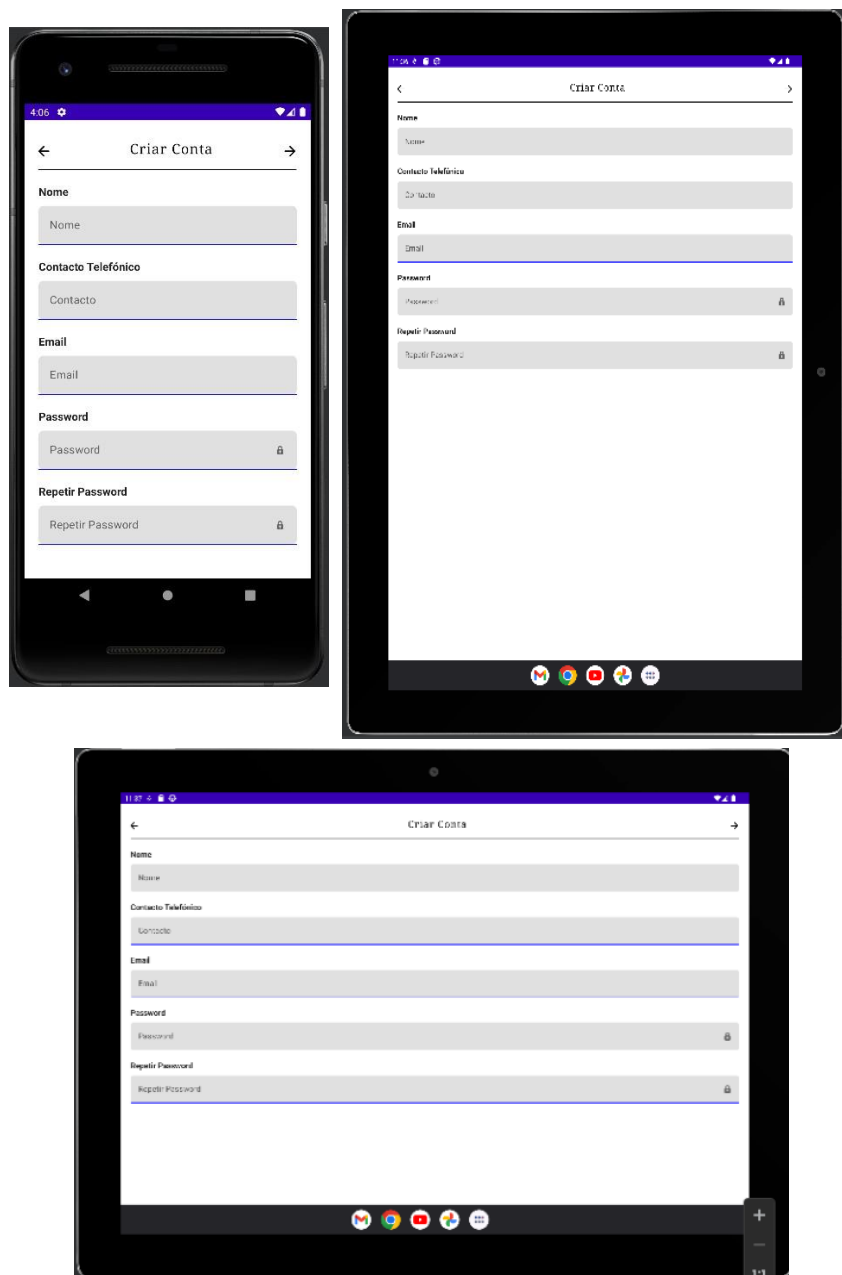
Figura 21- Página de Login (smartphone, tablet vertical, tablet horizontal)

#### Descrição:

Ecrã onde é possível **Iniciar Sessão**, para usufruir de todas as funcionalidades da aplicação, isto se já tiver uma conta previamente criada;

**Criar Conta**, caso ainda não tenha previamente criado uma.

## 6.2.2 Criar Conta



*Figura 22- Página de Criar Conta (smartphone, tablet vertical, tablet horizontal)*

### **Descrição:**

Ecrã onde é possível **Criar Conta**, apenas aplica-se a utilizadores que pretendam usar a aplicação e que ainda não possuem uma conta no sistema.

### 6.2.3 Escolher Tipo de Casa (Criar / Associar)



*Figura 23- Página de Escolher Tipo de Casa*

#### **Descrição:**

Ecrã onde é possível escolher entre duas opções: **Criar Nova Casa**, ou **Associar uma Casa** já existente através de um QR Code.

Note-se que para já apenas é possível Criar Nova Casa, isto porque ainda não se implementou a funcionalidade de associar casa por QR Code até ao momento.

Em vez disso o grupo decidiu criar um screen para substituir essa funcionalidade até a mesma estar pronta.

## 6.2.4 Nova Casa



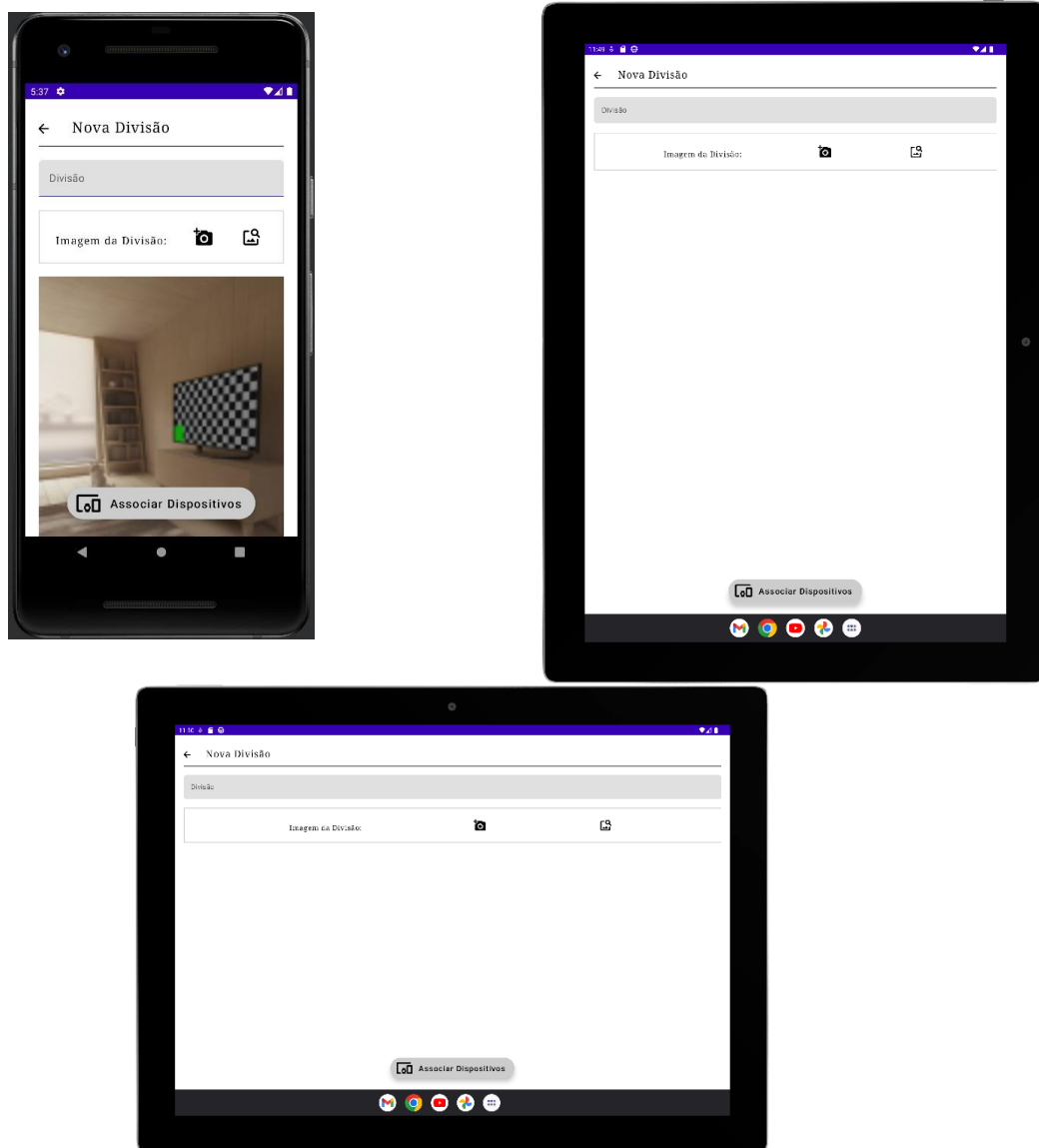
Figura 24- Página de Nova Casa (smartphone, tablet vertical, tablet horizontal)

### Descrição:

Ecrã onde é possível **Criar Nova Casa**, caso a mesma ainda não exista no sistema.

No futuro ao realizar este procedimento, deverá ser possível gerar um QR Code, que depois servirá para convidar membros novos para a Casa, bem como para Associar Casa.

### 6.2.5 Nova Divisão



*Figura 25- Página de Nova Divisão(smartphone, tablet vertical, tablet horizontal)*

#### **Descrição:**

Ecrã onde é possível **Criar Nova Divisão**, caso a mesma ainda não exista no sistema.

De notar que em frente ao título “Imagem da Divisão”, existem dois ícones, onde é possível tirar uma foto com a câmara do dispositivo e esta é mostrada em baixo (1º ícone).

É possível também fazer upload de uma imagem que já exista nos ficheiros pessoais do dispositivo, e esta é mostrada em baixo, tal como a outra opção (2º ícone).

## 6.2.6 Dispositivos Não Conectados

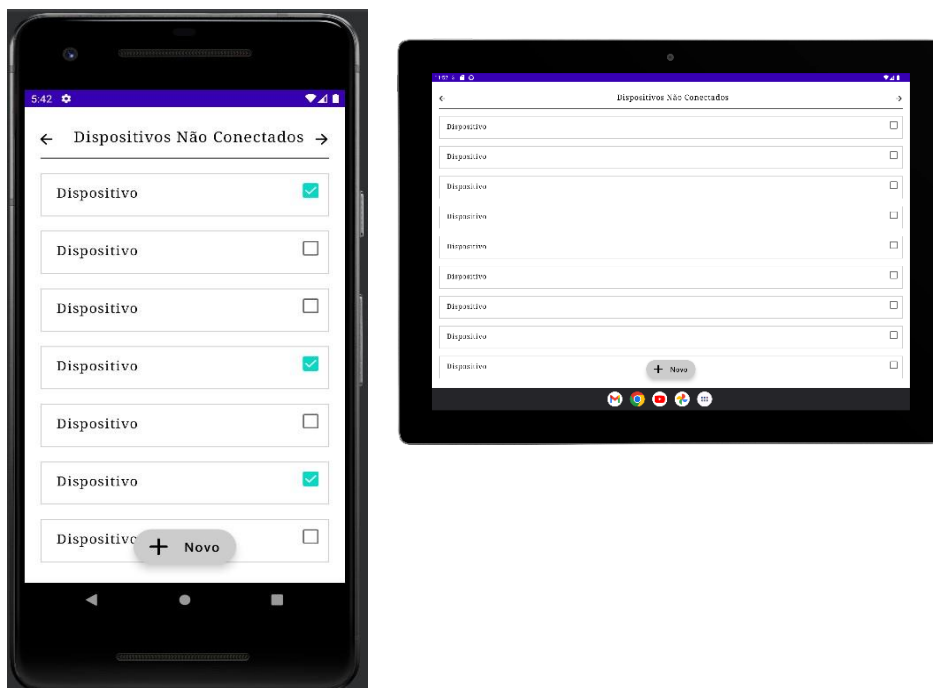


Figura 26- Página de Dispositivos Não Conectados (smartphone, tablet horizontal)

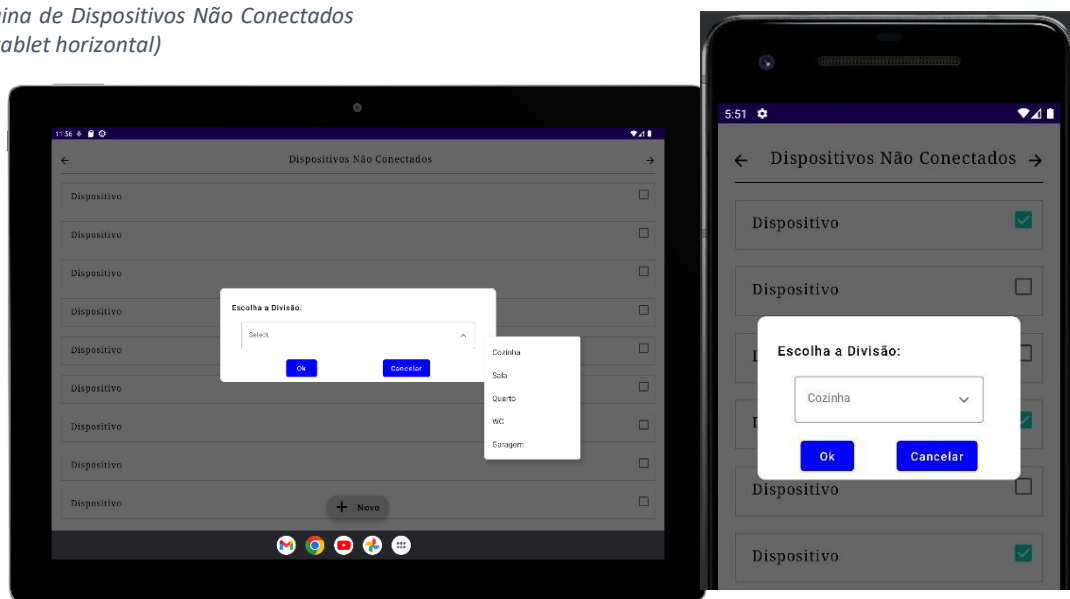


Figura 27- Popup para Escolher Divisão (tablet horizontal, smartphone)

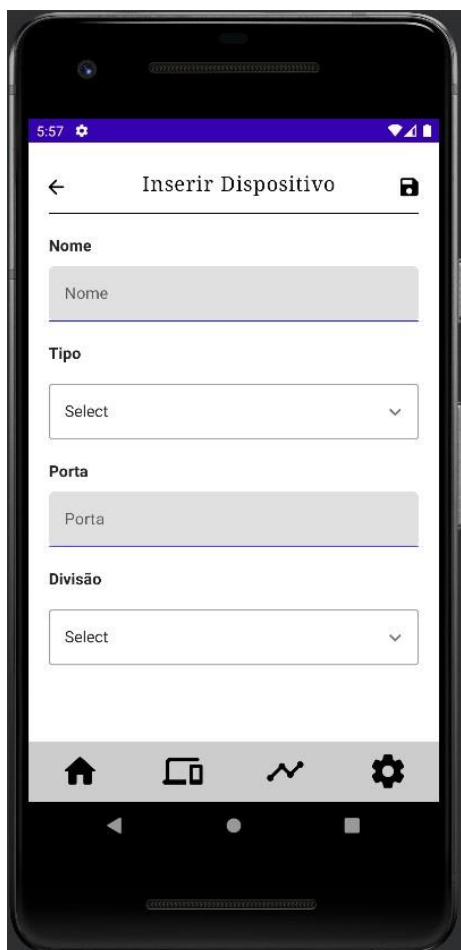
### Descrição:

Ecrã onde é possível seleccionar entre todos os dispositivos não conectados, aqueles que pretendemos adicionar a uma respetiva divisão.

Na segunda imagem é apresentado o popup que serve para escolher a divisão à qual pretende-se associar os dispositivos escolhidos (clizando no ícone com seta para a direita da topbar).

No entanto, e caso o dispositivo não conste na lista apresentada por qualquer razão, é possível adicionar um novo dispositivo, clicando no botão “Novo”.

### 6.2.8 Novo Dispositivo



The image shows a smartphone screen displaying a form titled "Inserir Dispositivo". The form has four input fields: "Nome" (text), "Tipo" (dropdown menu), "Porta" (text), and "Divisão" (dropdown menu). The status bar at the top shows the time 5:57 and various icons. The bottom navigation bar contains icons for home, a folder, a line graph, and settings.

*Figura 28- Página de Criar Dispositivo*

**Descrição:**

Ecrã onde é possível criar um novo dispositivo, caso o mesmo ainda não conste inserido no sistema.



## 6.2.9 Dispositivos Conectados



Figura 29- Página de Dispositivos Conectados

### Descrição:

Ecrã onde é possível eliminar dispositivos já existentes e conectados ao sistema.

No sentido de filtrar dispositivos por divisão, deve-se escolher uma opção na select box frente ao título “Filtrar pela Divisão”.

Caso o dispositivo não esteja aqui presente deve ser possível adicioná-lo, no botão “Adicionar”, que leva o utilizador para “Dispositivos Não Conectados”.

## 6.2.10 Home



Figura 30- Página de Home



Figura 31- Menu de Utilidades

**Descrição:**

Ecrã pós login, onde mostra em grelha todas as divisões que existem já criadas até ao momento. De notar que para já cada divisão tem como cor de fundo uma cor aleatória, deverá ser possível atribuir como imagem de fundo a imagem escolhida ao criar divisão.

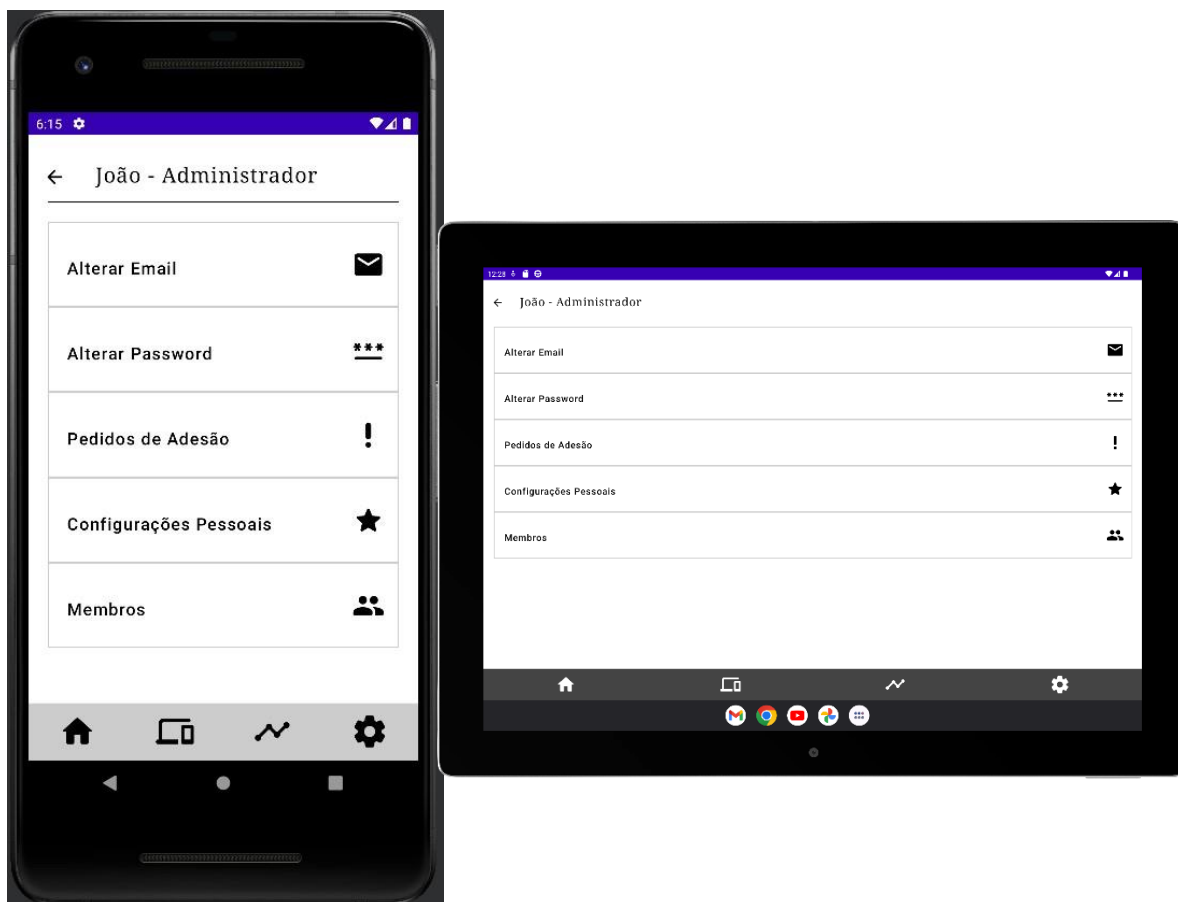
Caso o utilizador altere a casa, alterando a select box, deverá ser possível mudar-se para a casa selecionada. Útil para utilizadores com várias casas em sua posse.

Na opção de menu (canto superior esquerdo), existe um menu de utilidades onde deve ser possível dar uso a algumas utilidades, recorrendo ao uso de intents.

O Botão Nova, leva o utilizador a criar nova divisão.

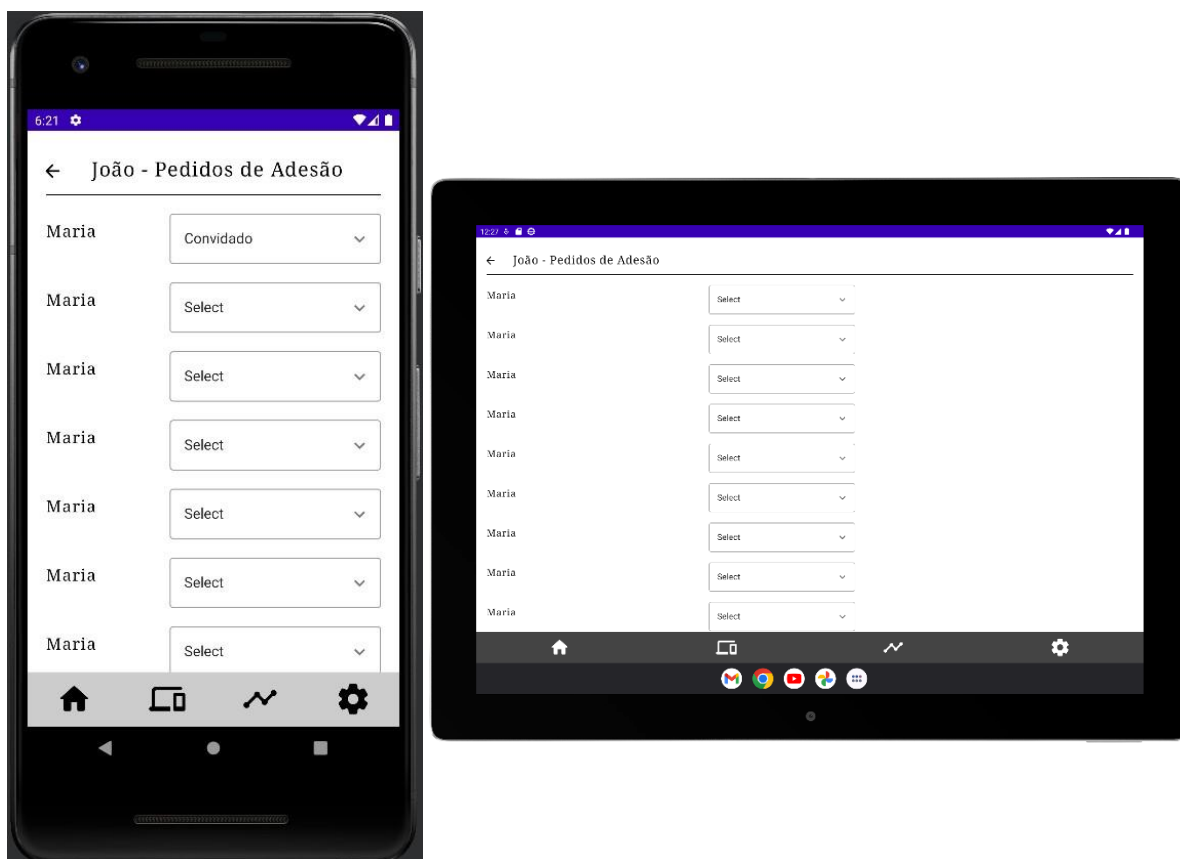
Quanto à Bottombar, o primeiro ícone serve para navegar até aos Dispositivos Conectados, o 2º ícone serve para navegar até ao perfil, o 3º para o ecrã de consumos e o último para Definições.

## 6.2.11 Perfil

*Figura 32- Página de Perfil***Descrição:**

No Perfil é possível alterar alguns dados pessoais, bem como outro tipo de funcionalidades consoante tipo de permissão. Neste caso para administrador (único tipo validado até ao momento), deve aceitar ou não pedidos de adesão e escolher o seu cargo, visualizar as suas configurações pessoais relativas a dispositivos e aceitar ou não membros, bem como escolher o seu respetivo cargo.

### 6.2.12 Pedidos de Adesão

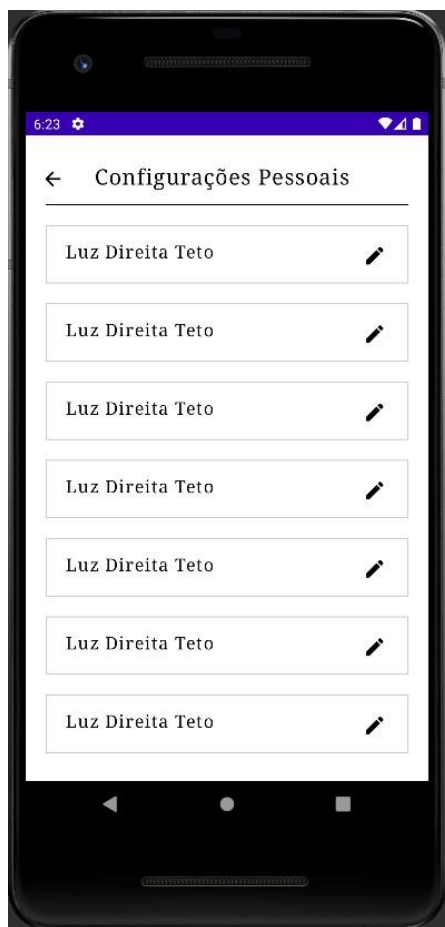


*Figura 33- Página de Pedidos de Adesão(smartphone, tablet horizontal)*

#### **Descrição:**

Relativamente a Pedidos de Adesão, deverá ser possível escolher o tipo de cargo de cada candidato, ou escolher Não Aceite, caso seja algum intruso.

### 6.2.13 Configurações Pessoais



*Figura 34- Página de Configurações Pessoais*

**Descrição:**

Este ecrã demonstra a lista de dispositivos que o utilizador marcou como favoritos relativos à sua preferência.

Cada dispositivo aqui listado é possível ser alterado clicando em cada botão, onde o utilizador é encaminhado para o dispositivo que marcou como favorito.

## 6.2.14 Definições de cada Dispositivo

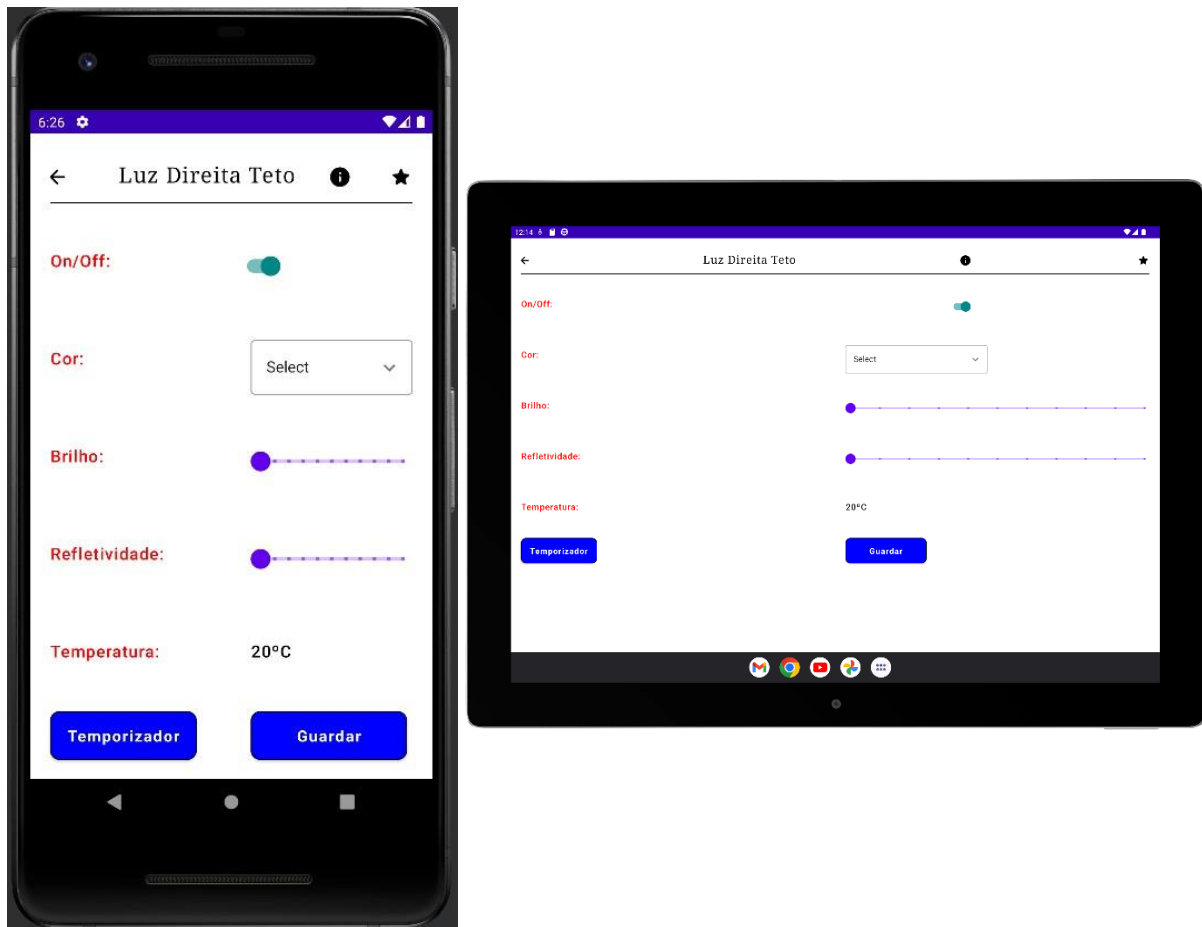


Figura 35- Página de Definições de Dispositivo (smartphone, tablet horizontal)

### Descrição:

Este ecrã no futuro desdobrar-se-á para todos os tipos de dispositivos: Luzes, Estores e Tomadas. Mas para já apenas se considerou Luzes.

Portanto, neste ecrã é possível configurar o presente dispositivo, podendo neste caso agendar temporizador, onde clicando no botão de Info da topbar, é mostrada a informação do temporizador.

Na topbar é possível marcar o dispositivo como favorito clicando no ícone da estrela, para associar o mesmo a configuração pessoal.

## 6.2.15 Detalhes da Divisão

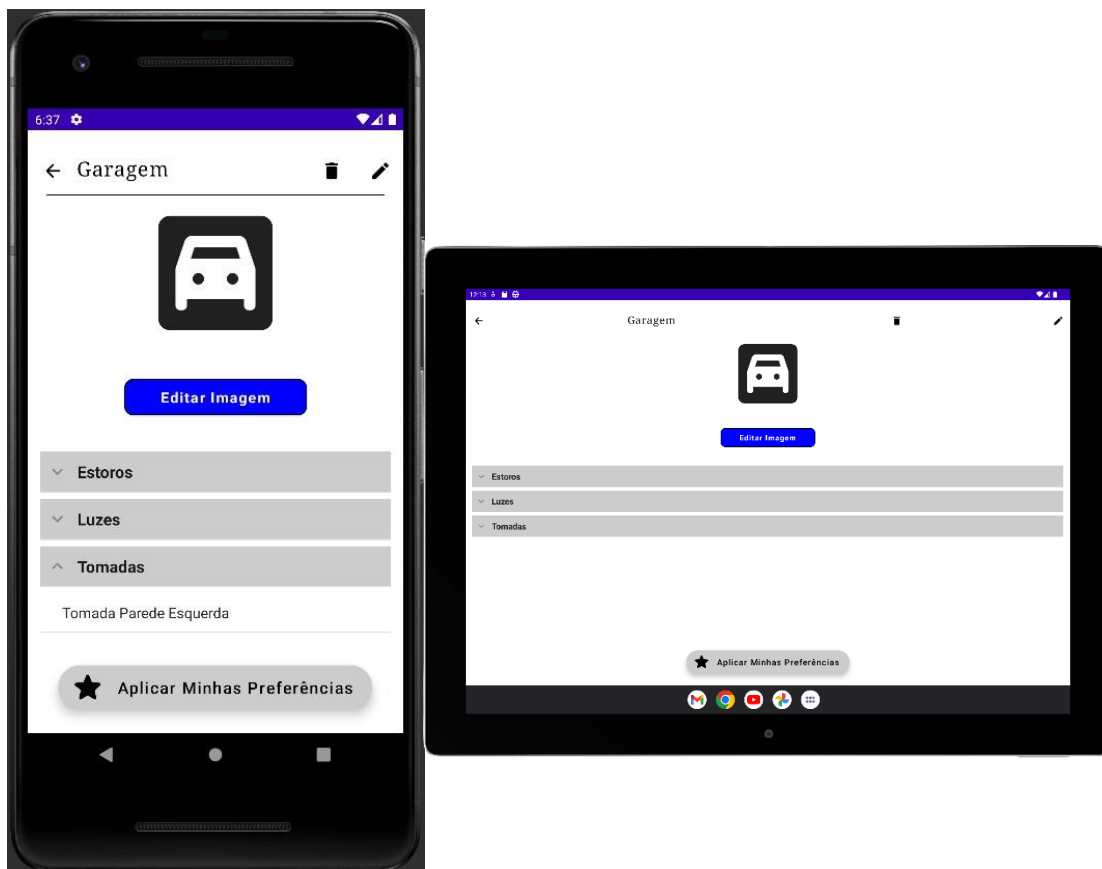


Figura 36- Página de Detalhes da Divisão (smartphone, tablet horizontal)

### Descrição:

Este ecrã permite alterar/ remover uma dada divisão da casa. Além disso é possível também aceder aos dispositivos contidos numa dada divisão. Estes estão categorizados por 3 tipos: Estores, Luzes, Tomadas.

## 6.2.15 Membros



Figura 37- Página de Membros



Figura 38- Popup (Escolhendo Residente Infantil)

### Descrição:

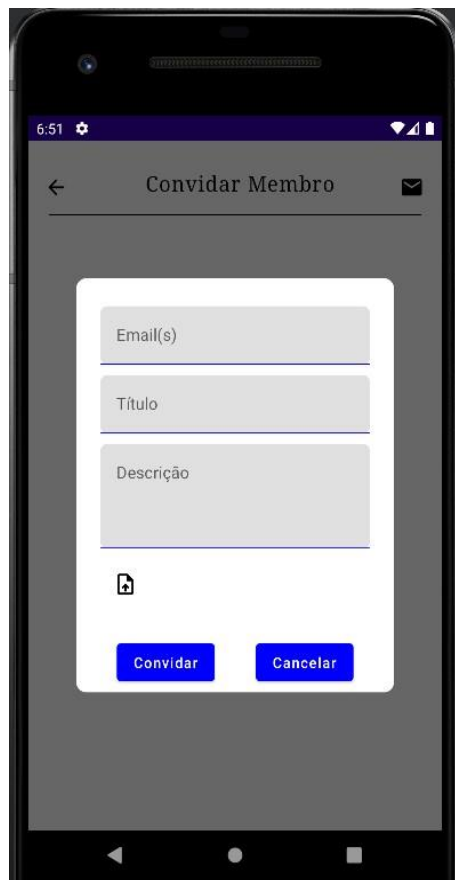
Este ecrã permite listar os membros que existem numa determinada casa. Caso seja pretendido adicionar novo membro, é possível convidá-lo, clicando no botão do canto superior direito.

Por outro lado, é possível alterar o cargo de cada membro no ícone de editar, telefonar, enviar uma mensagem, ou até mesmo eliminá-lo da casa.

Caso o cargo escolhido de uma determinada pessoa, seja Residente Infantil, aparecerá um botão que quando clicado pode levar ao ecrã de Localização.



## 6.2.16 Convidar Membro

*Figura 39- Página de Convidar Membro**Figura 40 - Popup de Convidar Membro por Email***Descrição:**

Para convidar membro, é possível o membro apontar a câmara do seu dispositivo ao QR Code aqui gerado, e este associa-se à casa ou então, pode-se convidá-lo por email ou por emails (separando por vírgulas), mandando como anexo o QR Code para o membro se poder juntar.

## 6.2.17 Consumos

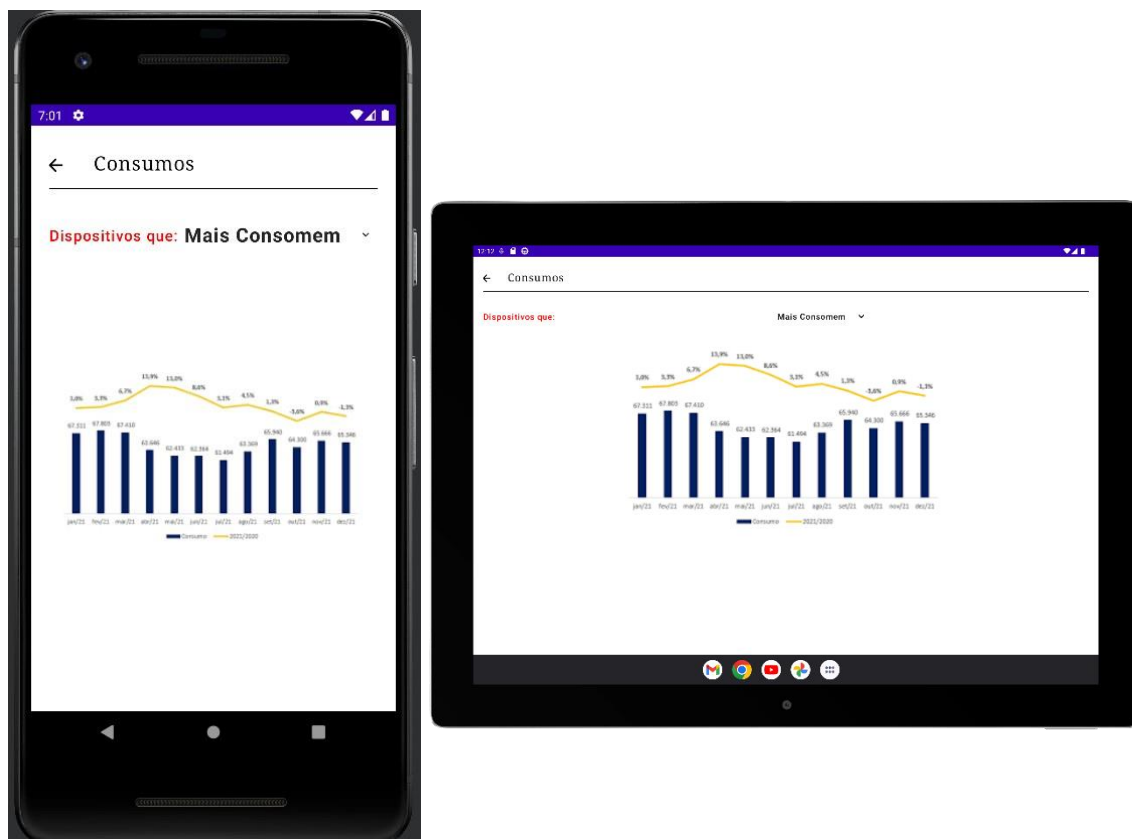
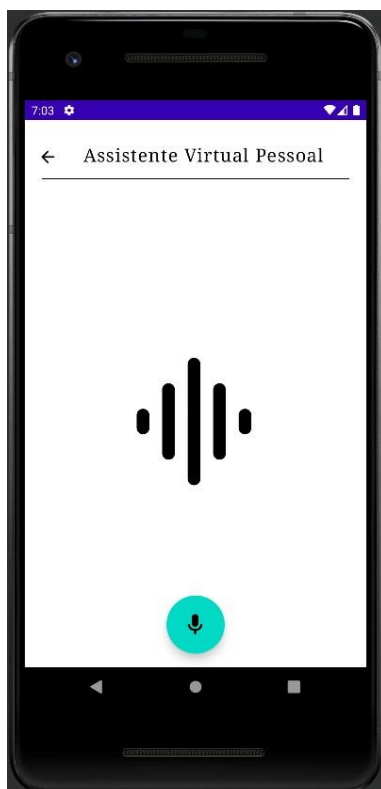


Figura 41- Página de Consumos(smartphone, tablet horizontal)

### Descrição:

Ecrã que apresenta sob a forma de gráficos, os dispositivos que mais/menos consomem ou mesmo todos, dependendo da opção seleccionada.

## 6.2.18 Assistente Pessoal

*Figura 42- Página de Assistente Pessoal***Descrição:**

Ecrã que retrata o assistente pessoal, que servirá como recetor de comandos de voz para executar determinadas operações na aplicação.

## 6.2.19 Ajuda

*Figura 43- Página de Ajuda***Descrição:**

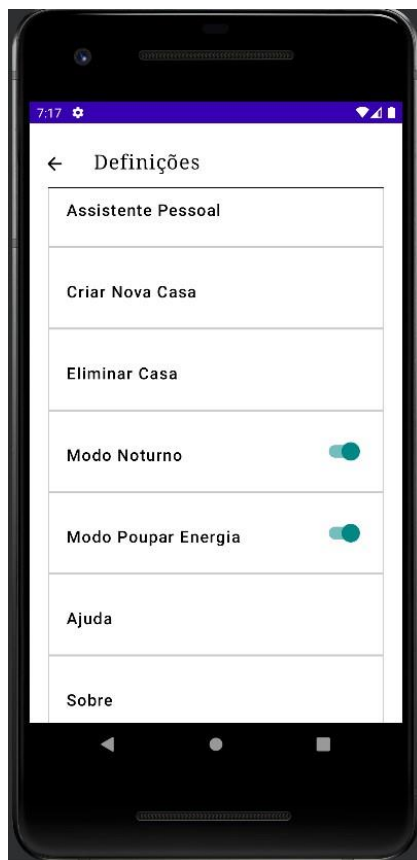
O presente ecrã no futuro conterá um vídeo explicativo de como a aplicação funciona na sua generalidade, é uma espécie de tutorial.

## 6.2.20 Sobre

*Figura 44- Página de Sobre***Descrição:**

O presente ecrã expõe qual o propósito da aplicação MySmartHome, bem como um pequeno resumo dos objetivos da mesma. Mostra também quem foram os seus desenvolvedores e criadores.

### 6.2.21 Definições



*Figura 45- Página de Definições*

**Descrição:**

O presente ecrã demonstra as definições da aplicação, onde é possível ter acesso a outros tipos de funcionalidades adicionais. Possui também dois modos: Modo Noturno e Poupança de Energia, que deverão alterar as configurações dos dispositivos existentes para que reduzam o seu consumo/adaptem-se à noite, entre outros.

## 6.2.22 Associar Casa

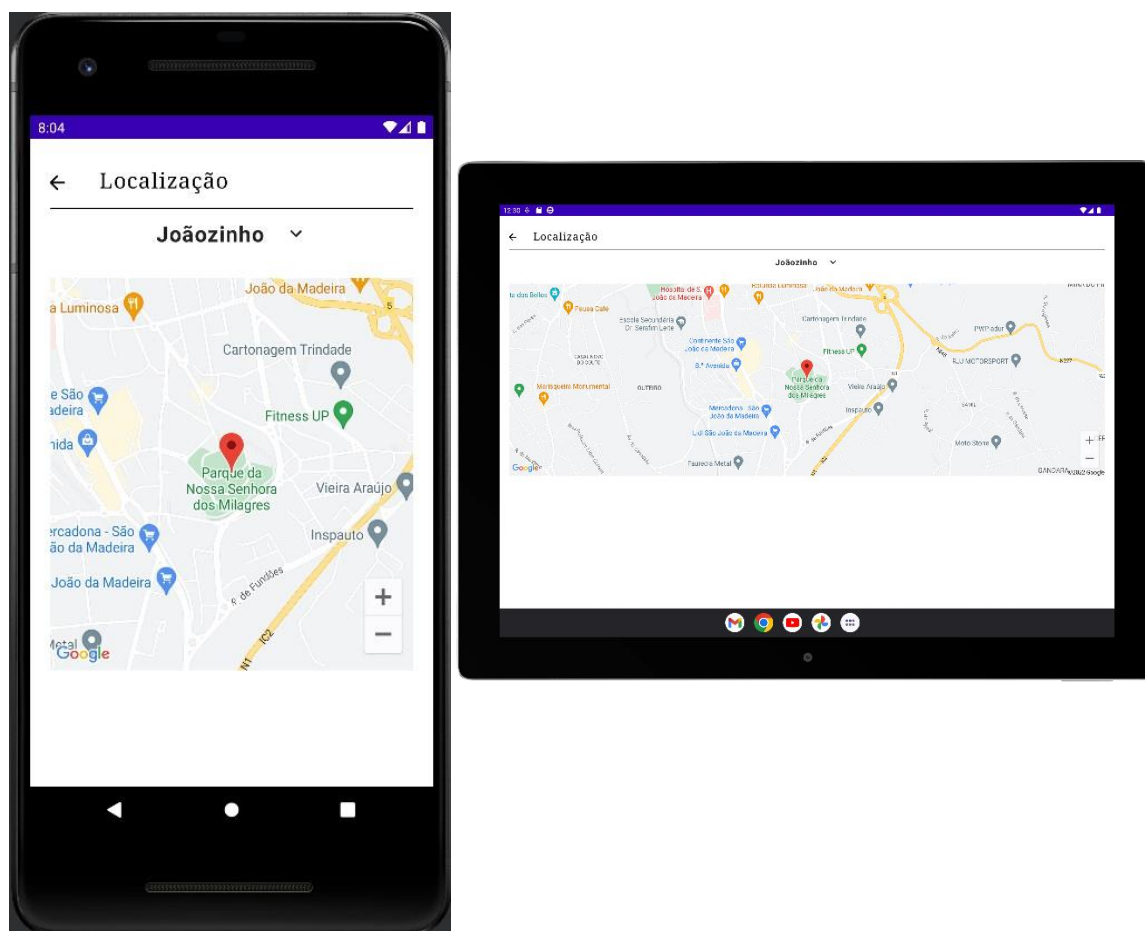


*Figura 46- Página de Associar Casa*

### **Descrição:**

Este ecrã é uma mera apresentação, sendo que no futuro deverá apresentar abrir a câmara e permitir que o utilizador possa reconhecer o QR Code através da câmara do seu dispositivo.

## 6.2.23 Localização



*Figura 47- Página de Localização (smartphone, tablet horizontal)*

### **Descrição:**

Este ecrã mostra a localização atual de um membro infantil / incapacitado, sendo este membro da casa.