

## **Licenciatura em Engenharia Informática**

### **CMU: 3º Ano 1º Semestre** *SmartHome*



Trabalho realizado por:

Lúcia Pereira – 8200278

Tiago Pacheco – 8200421

Rui Coelho - 8190352

## Índice

Introdução .....	3
Estudo do domínio .....	3
Tools Utilizadas.....	4
Desenvolvimento de Mockups .....	4
Elaboração dos Requisitos Funcionais.....	7
Desenvolvimento da aplicação em Android Studio .....	9
Observações.....	13
Considerações Finais.....	13

Figura 1: Mockups com Sign Up e Sign In em light e dark mode .....	5
Figura 2:Mockups com Devices e Select a Device em light e dark mode .....	5
Figura 3:Mockups com Select o Wi-Fi Network e List os Devices em light e dark mode.....	6
Figura 4:Mockups com Criar Groups e List Groups em light e dark mode .....	6

## INTRODUÇÃO

No âmbito da Unidade Curricular de Computação Móvel e Ubíqua foi-nos proposta o desenvolvimento de uma aplicação em Android Studio, com recurso a linguagem de programação Kotlin.

Mediante a apresentação dos temas propostos pelo regente da Unidade Curricular, a equipa de trabalho, constituída por 3 elementos, optou pelo desenvolvimento da aplicação cujo tema é SmartHome.

Esta aplicação passa como o próprio nome indica, por um controle de forma automatizada de um conjunto de dispositivos que normalmente podemos encontrar dentro de uma casa. É considerada inteligente pois esse controle deixa de ser manual, como por exemplo ligar uma luz num interruptor e passa a ser controlado a partir do nosso Smartphone com o sistema operativo Android, onde essa aplicação está instalada.

Com isto, pretendemos desenvolver no IDE do Android Studio uma aplicação que reproduza algo similar e com isto aplicar um conjunto de conhecimentos que foram obtidos ao longo do semestre, nesta disciplina.

## ESTUDO DO DOMÍNIO

Para o desenvolvimento desta aplicação foi necessário fazer uma pesquisa aprofundada sobre esta temática.

Serviram de base algumas aplicações existentes no mercado, nomeadamente a aplicação da *Shelly*, a *Home* desenvolvida pela google e outras semelhantes.

No estudo do seu domínio percebemos que existe sempre a necessidade de:

1. Criar conta do utilizador, com email e password;
2. Fazer uma conexão à rede de internet disponível no local onde os dispositivos se encontram para os poder controlar;
3. Existe a necessidade de associar um dispositivo, de diferentes tipos, tais como uma lâmpada, uma tomada, ou uma persiana.
4. Ao associar um dispositivo podemos renomear para melhor o identificar e a partir daqui passa a existir na aplicação;
5. Neste fluxo, a seguir a associar um dispositivo, por exemplo uma lâmpada, ao estabelecermos conectividade pela rede, podemos verificar o seu estado e controlar se a queremos ligada ou desligada;

6. Outra das funcionalidades presente neste tipo de aplicações é a possibilidade de criar divisórias, que são basicamente os diferentes compartimentos que existem numa casa e associar em cada divisória os respetivos dispositivos, ou seja, no compartimento sala podemos ter duas tomadas associadas, uma lâmpada e uma persiana e quando queremos ligar ou desligar ou subir/descer no caso da persiana podemos fazê-lo sabendo que estamos a manipular os dispositivos associados aquela divisória da casa;
7. Uma funcionalidade também explorada nestas aplicações é o estabelecimento de rotinas onde existe a possibilidade de agendar para determinado dia e hora a mudança de estado de determinado dispositivo.

## TOOLS UTILIZADAS

Para o desenvolvimento deste trabalho foram usadas as seguintes ferramentas:

- Moqups App: ferramenta utilizada para a elaboração dos designs mockups a serem usadas na aplicação.
- GitLab: Ferramenta de trabalho colaborativa onde todo o trabalho que foi desenvolvido pelos elementos do grupo convergiu (o qual o regente da disciplina tem acesso);
- Android Studio: IDE de desenvolvimento da aplicação;
- Discord: Canal de comunicação escrita e verbal entre os elementos do grupo, e onde se partilhavam links, prints, code-snippets e onde a equipa reunia para desenvolver o trabalho.
- Visual Studio Code: IDE para a utilização da Api.
- Postman: Pedidos Rest para testar a API.

## DESENVOLVIMENTO DE MOCKUPS

Após o estudo do domínio conseguimos perceber o tipo de funcionalidades a ser desenvolvidas. Contudo para a sua implementação foi necessário desenhar as views que o utilizador irá ver e sobre as quais deverá poder interagir.

Deste modo, e usando o ambiente de desenvolvimento de desenho de mockups da Moqups App, passamos a contruir passo a passo as nossas vistas do user interface.

Foram desenvolvidas várias mockups em modo light ou dark mode como podemos ver de seguida, ou através deste [link](#):

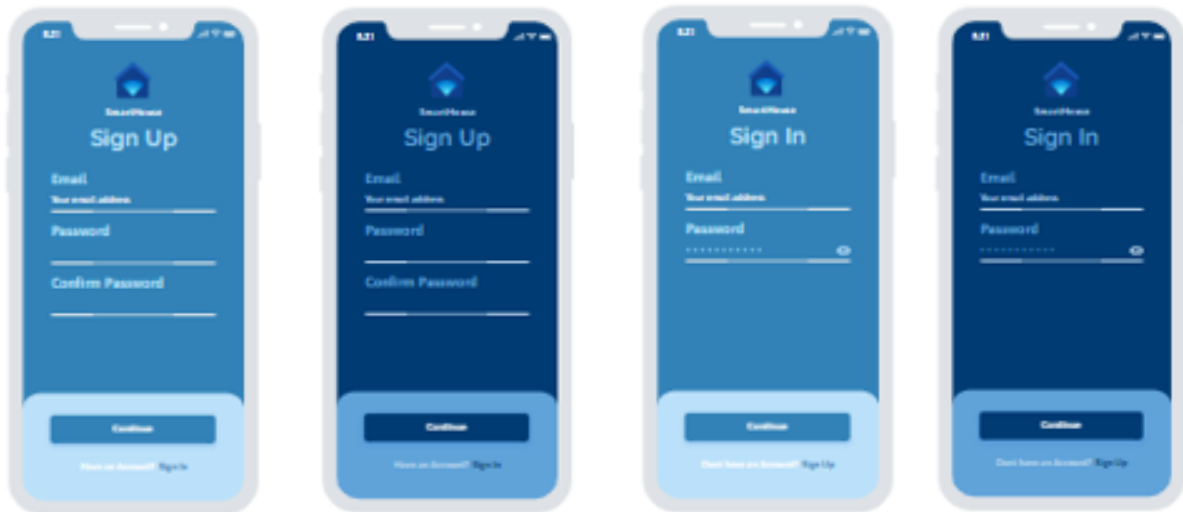


Figura 1: Mockups com Sign Up e Sign In em light e dark mode



Figura 2: Mockups com Devices e Select a Device em light e dark mode



Figura 3:Mockups com Select o Wi-Fi Network e List os Devices em light e dark mode



Figura 4:Mockups com Criar Groups e List Groups em light e dark mode



**Imagem 5: Mockups com Listar Devices de um Group e Routines em light e dark mode**

## ELABORAÇÃO DOS REQUISITOS FUNCIONAIS

A tabela seguinte permite mostrar os requisitos funcionais que esta aplicação deverá ter, que se resumem numa fase embrionária aos seguintes:

Categoria: Criar/Adicionar Utilizador			
ID	Nome	Descrição	Prioridade
[RF_L_001]	Sign Up – Criar conta	Deverá ser possível criar conta de utilizador bastando associar um email e uma password.	Alta
[RF_L_002]	Sign In – Fazer login	Deverá ser possível depois de inserir mail e password fazer o login e entrar na aplicação.	Alta

Categoria: Adicionar Dispositivos			
ID	Nome	Descrição	Prioridade
[RF_D_001]	Add Device – Adicionar dispositivo	Deverá ser possível adicionar um dispositivo, cuja aplicação já disponibiliza, nomeadamente, lâmpada, tomada ou persiana.	Alta
[RF_D_002]	Select Device	Deverá ser possível selecionar dispositivo	Alta
[RF_D_003]	Remove Device	Deverá ser possível remover dispositivo	Alta
[RF_D_004]	List Devices	Deverá ser possível listar todos os dispositivos adicionados	Alta
[RF_D_005]	Update Device	Deverá ser possível editar informações sobre o nome dado aquele dispositivo	Normal
[RF_D_006]	Associar Device a um group/divisória	Deverá ser possível associar um dispositivo a uma divisória da casa (ou grupo).	Baixa

Categoria: Select o Wi-Fi Network			
ID	Nome	Descrição	Prioridade
[RF_WF_001]	Select o Wi-Fi Network	Deverá ser possível associar uma rede wifi disponível e password	Baixa

Categoria: Criar grupos/divisórias			
ID	Nome	Descrição	Prioridade
[RF_G_001]	Add Group - Adicionar grupos	Deverá ser possível associar um ou mais grupos/divisórias da casa	Normal
[RF_G_002]	Remove Group - Remover Grupo	Deverá ser possível remover um grupo/divisória da casa	Normal
[RF_G_003]	Edit Group - Editar Grupo	Deverá ser possível renomear o grupo	Baixa



[RF_G_004]	List Groups - Listar Grupos	Deverá ser possível listar todos os grupos criados	Normal
------------	--------------------------------	---	--------

Categoria: Criar rotinas			
ID	Nome	Descrição	Prioridade
[RF_R_001]	Add Routine – Adicionar rotinas	Deverá ser possível criar um agendamento para ligar ou desligar um determinado dispositivo num dia a uma determinada hora.	Baixa
[RF_R_002]	Edit Rotines – Editar Rotinas	Deverá ser possível para aquele dispositivo reagendar novo horário para ligar/desligar.	Baixa
[RF_R_003]	Remove Rotines – Remover Rotinas	Deverá ser possível remover um agendamento para ligar ou desligar aquele dispositivo naquele dia/hora.	Baixa

## DESENVOLVIMENTO DA APLICAÇÃO EM ANDROID STUDIO

1. Passo: Criação das Views (Mockups) em Jetpack Compose		
Requisitos	Implementado	Observações
[RF_L_001]	Ok	
[RF_L_002]	Ok	
[RF_D_001]	OK	
[RF_D_002]	OK	
[RF_D_003]	Not Ok	Falta adicionar um botão onde seja possível eliminar um device e onde apareça um toast com a informação de eliminado com sucesso.

[RF_D_004]	OK	
[RF_D_005]	Not Ok	Falta adicionar um botão onde seja possível aparecer uma janela para editar informação sobre o nome do device.
[RF_D_006]	Not Ok	
[RF_WF_001]	Not Ok	Esta view apesar de ser necessária numa aplicação real, para este caso a comunicação será feita por uma API Rest disponibilizada pelo docente, logo não faz sentido ser implementada.
[RF_G_001]	Ok	
[RF_G_002]	Not Ok	Falta adicionar um botão onde seja possível eliminar um grupo ou divisória e onde apareça um toast com a informação de eliminado com sucesso.
[RF_G_003]	Not Ok	Falta adicionar um botão onde seja possível editar informações de um grupo ou divisória e onde apareça um toast com a informação de editado com sucesso.
[RF_G_004]	Ok	
[RF_R_001]	Ok	
[RF_R_002]	Not Ok	Prioridade Baixa
[RF_R_003]	Not Ok	Prioridade Baixa

2. Passo: Implementação do Room		
Requisitos	Implementado	Observações
[RF_L_001]	Ok	
[RF_L_002]	Ok	
[RF_D_001]	Not Ok	
[RF_D_002]	Not Ok	
[RF_D_003]	Not Ok	

[RF_D_004]	Not Ok	
[RF_D_005]	Not Ok	
[RF_D_006]	Not Ok	
[RF_WF_001]	Not Ok	N/A
[RF_G_001]	Not Ok	
[RF_G_002]	Not Ok	
[RF_G_003]	Not Ok	
[RF_G_004]	Not Ok	
[RF_R_001]	Not Ok	
[RF_R_002]	Not Ok	Prioridade Baixa
[RF_R_003]	Not Ok	Prioridade Baixa

3. Passo: Implementação da Navegação entre Páginas		
Requisitos	Implementado	Observações
[RF_L_001]	Ok	
[RF_L_002]	Ok	
[RF_D_001]	Ok	
[RF_D_002]	Ok	
[RF_D_003]	Not Ok	
[RF_D_004]	Ok	
[RF_D_005]	Not Ok	
[RF_D_006]	Not Ok	

[RF_WF_001]	Not Ok	N/A
[RF_G_001]	Ok	
[RF_G_002]	Not Ok	
[RF_G_003]	Not Ok	
[RF_G_004]	Ok	
[RF_R_001]	Ok	
[RF_R_002]	Not Ok	Prioridade Baixa
[RF_R_003]	Not Ok	Prioridade Baixa

1. Passo: Implementação do Retrofit		
Requisitos	Implementado	Observações
[RF_L_001]	Not Ok	
[RF_L_002]	Not Ok	
[RF_D_001]	Ok	
[RF_D_002]	Ok	
[RF_D_003]	Not Ok	
[RF_D_004]	Ok	
[RF_D_005]	Not Ok	
[RF_D_006]	Not Ok	
[RF_WF_001]	Not Ok	N/A
[RF_G_001]	Not Ok	
[RF_G_002]	Not Ok	
[RF_G_003]	Not Ok	
[RF_G_004]	Not Ok	

[RF_R_001]	Not Ok	
[RF_R_002]	Not Ok	Prioridade Baixa
[RF_R_003]	Not Ok	Prioridade Baixa

## OBSERVAÇÕES

A aplicação só é funcional se estiver pelo menos um dispositivo alocado na API.

De acordo com o que foi desenvolvido nesta aplicação, verificamos que existiu alguns constrangimentos sobretudo na implementação do *Retrofit*, tal como apenas ser possível comunicar com um dispositivo (uma porta) devido ao comportamento da API disponibilizada.

Na categoria de rotinas e no menu do utilizador as páginas não estão implementadas logo os botões no bottom bar não estão funcionais.

Se o utilizador não iniciar sessão os botões de navegação (bottom bar) estão inativos, propositadamente.

A password é encriptada quando guardada na base de dados.

É importante salientar que no decorrer da elaboração do projeto, foi feita grande parte da documentação do código implementado até ao momento, sendo mais fácil a sua interpretar/compreensão.

## CONSIDERAÇÕES FINAIS

Após a realização desta *milestone* verificamos que a utilização do *jetpack compose* facilita o desenvolvimento da aplicação comparada com as tecnologias anteriormente lecionadas.

De acordo com o mencionado anteriormente, podemos constatar que algumas funcionalidades foram implementadas e outras ainda serão alvo implementação na segunda *milestone*. No entanto, esta primeira *milestone*, consideramos que foi bem-sucedida na medida em que conseguimos realizar os objetivos propostos para esta parte do trabalho, nomeadamente, as *views* com *jetpack compose*, *navigation*, *room* e *retrofit*.

