



Departamento de Engenharia Informática e de Sistemas
Instituto Superior de Engenharia de Coimbra
Licenciatura em Engenharia Informática
Arquiteturas Móveis 2021/2022

Relatório Técnico

Reversi

André Rodrigues 2019139754

Nuno Honório 2019126457

Samuel Tavares 2019126468

Índice

| | |
|---------------------------------|---|
| 1. Introdução..... | 3 |
| 2. Estrutura do projeto..... | 4 |
| 2.1. Atividades..... | 4 |
| 2.1.1. MainActivity | 4 |
| 2.1.2. PerfilActivity | 5 |
| 2.1.3. FotografiaActivity | 5 |
| 2.1.4. ModoJogoActivity..... | 6 |
| 2.1.5. JogoActivity | 6 |
| 2.1.6. OpcoesActivity..... | 6 |
| 2.1.7. ScoresActivity | 6 |
| 2.2. Lógica..... | 6 |

1. Introdução

Este projeto pretendia-se desenvolver um jogo para android, o jogo de tabuleiro Reversi.

Para isso, foi usado o IDE Android Studio e desenvolvido em Kotlin e XML, Kotlin para a lógica da aplicação e XML para as interfaces criadas. Foram escolhidas estas linguagens pois foram lecionadas ao longo do semestre.

Numa fase inicial é necessário realizar login na conta Google ou criar uma conta email na base de dados. Ao criar a conta pela primeira vez é pedido para preencher os dados respetivos ao mesmo, e ainda tirar a fotografia para o seu perfil.

O jogo tem três modos. O modo 1, sendo o mais básico, permite dois jogadores jogarem no mesmo dispositivo. O modo 2, é igualmente um jogador contra outro, mas desta vez um jogador tem de criar um servidor para o outro jogador ligar-se como cliente através do ip gerado. No modo 3 é criado um servidor e dois utilizadores conectam-se ao mesmo servidor, ou seja, será um jogo 1v1v1.

Depois da escolha de modo, e conexão caso necessário, começa o jogo e o resultado do jogo é armazenado no Firebase da Google. Também toda a informação do perfil é guardada, desde a imagem ao email e password, permitindo no futuro realizar login na conta já criada.

O jogo decorre até as peças de um dos jogadores deixarem de existir ou o tabuleiro estiver completamente cheio de peças. Quando alguém ganha é apresentado um pop up com o nome dele e se pretende ou não iniciar um novo jogo.

Além disso, são guardados os 5 melhores scores de todas as partidas jogadas no próprio dispositivo.

2. Estrutura do projeto

O nosso projeto está dividido em várias atividades de forma a facilitar a organização de código e interpretação do mesmo. Cada atividade tem um ou mais ficheiros xml atribuídos, ficheiros os quais onde é editada a estrutura e design de cada página da aplicação desde cor de Fundo a botões.

2.1. Atividades

Encontram-se no package “atividades” e utilizam os fragmentos presentes no package “fragmentos”. Foram criados de modo a tornar a aplicação o mais simples e eficaz possível.

Além disso, é usada uma classe Utils, criada com funções que são usadas em várias atividades.

2.1.1. MainActivity

Esta atividade é a atividade inicial do jogo. É composta por botões que iniciam novas atividades e por um botão que encerra.

Para o seu design tem atribuído os xml: activity_modos_jogo e outro para a versão land da aplicação quando o utilizador inclina o telemóvel e tem a rotação permitida.

Para passar às próximas atividades são criados intents, que iniciam essa mesma atividade.

Estes intents são chamados quando cada botão é clicado, como por exemplo, carregando na imagem de perfil inicia o intent da PerfilActivity, abrindo a interface dessa atividade.

```
fun onNovoJogo(view: android.view.View){

    if(FirebaseAuth.getInstance().currentUser == null) {
        Toast.makeText( context this, text "É necessário iniciar sessão para poder jogar!", Toast.LENGTH_SHORT).show()

        val intent = Intent( packageContext this, PerfilActivity::class.java)
        startActivity(intent)

    }else{

        val intent = Intent( packageContext this, ModoJogoActivity::class.java)
        startActivity(intent)
    }
}

fun onOpcoes(view: android.view.View){
    val intent = Intent( packageContext this, OpcoesActivity::class.java)
    startActivity(intent)
}

fun onPontuacoes(view: android.view.View){
    val intent = Intent( packageContext this, ScoresActivity::class.java)
    startActivity(intent)
}

fun onPerfil(view: android.view.View){
    val intent = Intent( packageContext this, PerfilActivity::class.java)
    startActivity(intent)
}

override fun onBackPressed() {
    super.onBackPressed()
    finishAffinity(); // Close all activites
    exitProcess( status 0); // Releasing resources
}

fun onSair(view: android.view.View){
    finishAffinity(); // Close all activites
    exitProcess( status 0); // Releasing resources
}
```

Para isto acontecer é necessário “chamar” as atividades no AndroidManifest:

```
<activity
    android:name=".atividades.FotografiaActivity"
    android:configChanges="orientation"
    android:exported="true" />
<activity
    android:name=".atividades.PerfilActivity"
    android:configChanges="orientation"
    android:exported="true" />
<activity
    android:name=".atividades.JogoActivity"
    android:configChanges="orientation"
    android:exported="true" />
<activity
    android:name=".atividades.ModoJogoActivity"
    android:configChanges="orientation"
    android:exported="true" />
<activity
    android:name=".atividades.OpcoesActivity"
    android:configChanges="orientation"
    android:exported="true" />
<activity
    android:name=".atividades.ScoresActivity"
    android:configChanges="orientation"
    android:exported="true"/>
<activity
    android:name=".atividades.MainActivity"
    android:configChanges="orientation"
    android:exported="true">
```

2.1.2. PerfilActivity

Esta atividade é chamada quando o utilizador carrega na imagem de perfil e se encontra na atividade “MainActivity” ou carrega no botão “Novo Jogo” sem realizar login primeiramente.

É a atividade dedicada ao perfil e tudo que envolve o mesmo.

Quando o utilizador abre o perfil pela primeira vez pede para ele registar email ou login. Existem duas opções de login, por email, criado no registo, ou por Google.

Após realizar login pela primeira vez é pedido os dados ao utilizador, desde username, nacionalidade e foto de perfil. Só quando todos estes campos são preenchidos é o perfil é guardado.

A está a ser salva no Firebase Storage e a restante informação relativa sobre no Firebase Database.

Após realizar login é apresentado ao utilizador o seu perfil, criado no registo. Além disso, apresenta um botão que permite editar a sua informação pessoal e a fotografia.

Para realizar tudo o que era destinado a esta atividade foram criados 3 fragmentos para o Perfil (“RegistrarLoginFragment”, “RegistoPerfilFragment” e “VisualizarPerfilFragment”), com o objetivo de ir alterando a interface consoante o estado de registo/login.

2.1.3. FotografiaActivity

Esta atividade é chamada quando o utilizador clica no botão tirar fotografia disponível na “PerfilActivity”.

Tem como objetivo permitir ao utilizador tirar uma foto para adicionar ao seu perfil.

Esta interface é composta por 3 botões, o botão tirar fotografia, o botão rodar câmara e um botão oculto confirmar fotografia.

O botão confirmar fotografia só é revelado quando o utilizador tira a primeira fotografia.

2.1.4. ModoJogoActivity

Esta atividade é chamada quando o utilizador pressiona o botão “Novo Jogo” e tem como objetivo permitir ao utilizador escolher o modo que pretende jogar.

Assim sendo esta interface é composta por 4 botões: 2 Jogadores Local, 2 Jogadores Online, 3 Jogadores Online e Voltar ao Menu.

Quando o modo 2 Jogadores Online ou 3 Jogadores Online é escolhido abre um fragmento com dois botões, os botões Modo Servidor ou Modo Cliente.

2.1.5. JogoActivity

Esta atividade é chamada após o modo de jogo ter sido escolhido, ou ter sido sucedida a conexão dos clientes ao servidor nos modos online.

Esta actividade trata da gestão geral do jogo, jogadas, atualização do tabuleiro, scores, etc.

Além disso, é a atividade responsável por guardar os melhores scores na Firebase Database.

A interface é composta por vários fragmentos e dois botões. O fragmento dos Scores e Tabuleiro depende do modo de jogo, variando entre apresentar as peças conquistadas de 2 jogadores ou 3 jogadores e um tabuleiro de 8x8 ou 10x10. Os botões existentes são responsáveis pelas jogadas especiais e ficam ocultos após o jogador utilizar a jogada especial.

Quando um jogador não tem mais jogadas possíveis ou o jogo termina aparecem Dialogs, o primeiro para trocar de jogador e o segundo para iniciar um novo jogo ou sair da aplicação.

2.1.6. OpcoesActivity

Esta atividade é chamada quando o utilizador clica no icon das definições, na MainActivity.

Possibilita ao utilizador seleccionar várias opções que alteram a interface da aplicação e o jogo em si. Algumas funções como: Alterar o tema de cores, Ativar ou Desativar as ajudas das Jogadas Possíveis e ver a página informativa Sobre.

2.1.7. ScoresActivity

Esta atividade é chamada quando o utilizador clica no botão Pontuações que se encontra na MainActivity.

Esta interface é composta por uma lista e um botão voltar atrás.

Tem como objetivo apresentar os 5 melhores resultados obtidos no dispositivo em questão.

2.2. Lógica

A lógica necessária ao desenvolvimento do nosso projeto foi toda organizada no package “modelo”.

Quando falamos em lógica foram criados os seguintes ficheiros:

Reversi – Classe que contém toda a lógica do jogo. É responsável pelas validações e todo o tipo de jogadas.

Constantes – Classe que contém todas as constantes fixas usadas no Reversi.

3. Funcionalidades

3.1. Funcionalidades implementadas

- Atribuição das peças aos jogadores e às 4 células iniciais
- Verificações de posições irregulares escolhidas, inclusive peças especiais
- Mostrar as jogadas possíveis opcionalmente
- Alterar entre jogadores após as jogadas realizadas
- Atribuição do jogador inicial random
- Inicialização do tabuleiro 8x8
- Peça bomba e Peça Troca
- Verificações de existência de peça especial
- Troca de jogador quando não existem jogadas possíveis
- Atualização do score após jogada
- Verificação de fim de jogo e jogador vencedor
- Criar conta email e guardar informação do perfil na base de dados – Firebase database
- Tirar fotografia e guardar online – Firebase storage
- Permitir iniciar sessão com email registado anteriormente na base de dados
- Alterar dados pessoais e fotografia do perfil
- Aplicação totalmente landscape expeto PerfilActivity
- Permitir iniciar sessão via Google
- “Sobre nós”
- 5 melhores pontuações realizadas localmente
- Modo 2 Jogadores Local
 - ❖ Jogo totalmente funcional
- Modo 2 Jogadores Online
 - ❖ Jogo funcional online
- Modo 3 Jogadores Online – incompleto
 - ❖ Jogo totalmente funcional localmente

3.2. Funcionalidades não implementadas

- Landscape não funcional no Perfil Activity
- Modo 3 Jogadores Online
 - Não efetua ligação aos 3 clientes