

Licenciatura em Sistemas Multimédia

**Ferramentas Multimédia II**

**Quizz Jogo de Perguntas**

**Daniel Guedes / Bruno Faria**

Índice

Conteúdo

[Índice 2](#_Toc72441960)

[1. Android App desenvolvida 3](#_Toc72441961)

[2. Desenvolvimento Android App 4](#_Toc72441962)

[2.1 - Layout || Activity 4](#_Toc72441963)

[2.1.1 - MainWindow 4](#_Toc72441964)

[2.1.2 - Pontuação 5](#_Toc72441965)

[2.1.3 - Login 6](#_Toc72441966)

[2.1.4 - Registar 7](#_Toc72441967)

[2.1.5 - Perguntas 8](#_Toc72441968)

[2.1.6 - Resultado 9](#_Toc72441969)

[2.2 - Código JAVA Public Classes 10](#_Toc72441970)

[2.2.1 - Main 10](#_Toc72441971)

[2.2.2 - Login 11](#_Toc72441972)

[2.2.3 - Registar 13](#_Toc72441973)

[2.2.3.1 - Utilizador 15](#_Toc72441974)

[2.2.4 - Ranking 15](#_Toc72441975)

[2.2.5 - Perguntas 17](#_Toc72441976)

[2.2.5.1 - Pergunta 21](#_Toc72441977)

[2.2.6 - Resultado 21](#_Toc72441978)

[3. Mockup 23](#_Toc72441979)

# Android App desenvolvida

O desenvolvimento desta Android App é realizado com o objetivo de exemplificar um QUIZZ - jogo de perguntas e respostas. Um jogo mental no qual os jogadores tentam responder corretamente a questões que lhes são colocadas.

O jogo é pontuado e os vencedores são os que atingem o maior número de pontos.

Quizz generalista com questões sobre cultura geral com formato de escolha múltipla nas respostas.

No fim das respostas é apresentado o resultado de respostas certas versus erradas, sendo apresentado noutra janela um ranking global.

Para poder jogar o utilizador deve registar-se e fazer o login com a sua conta.

# Desenvolvimento Android App

## 2.1 - Layout || Activity

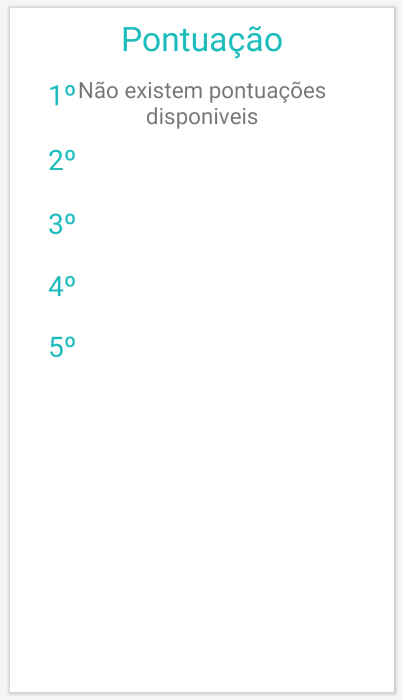
2.1.1 - MainWindow

O programa arranca na página de Mainwindow com as opções de começar e consulta da pontuação dos participantes.



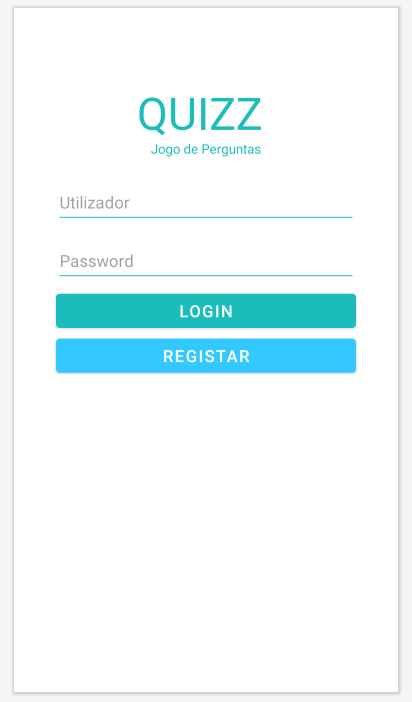
2.1.2 - Pontuação

Selecionado o botão da pontuação seremos remetidos para o Ranking dos melhores participantes.



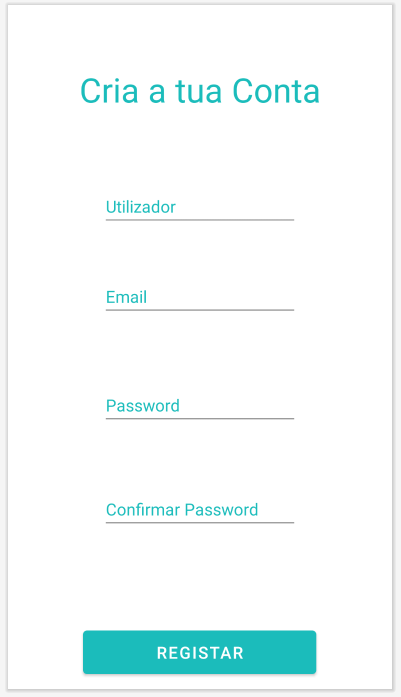
2.1.3 - Login

Selecionada a opção começar, a activity apresentada é a do login onde será necessário realizar o registo do participante registando os dados para autenticação.



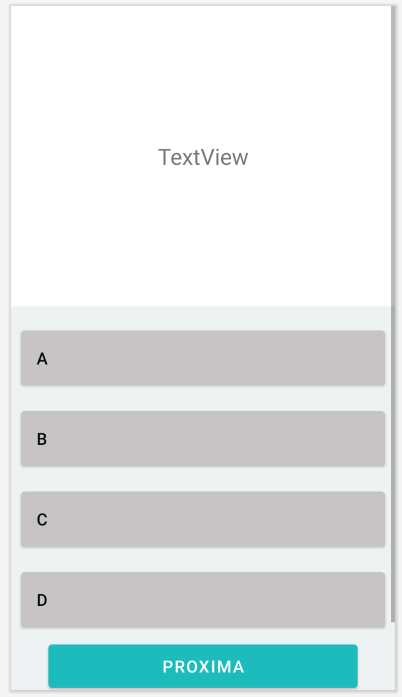
2.1.4 - Registar

Selecionada a opção registar é disponibilizado um formulário de inscrição / registo.



2.1.5 - Perguntas

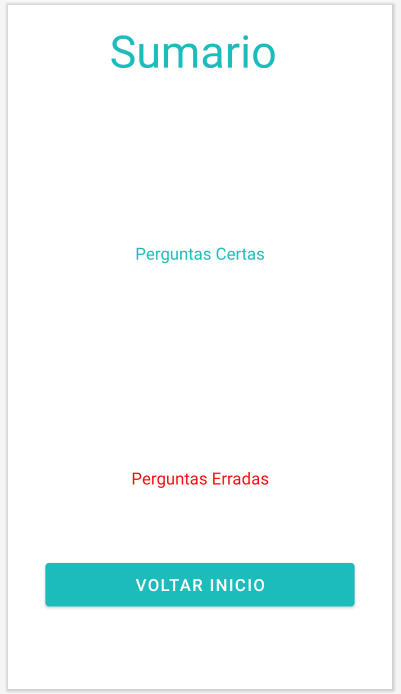
Nesta activity será onde serão apresentadas as perguntas com as respetivas 4 opções de respostas. O participante deve escolher uma das respostas e esta ficará sinalizada a outra cor. Ao fazer próximo será remetido para outra pergunta.



2.1.6 - Resultado

No fim do ciclo de perguntas será apresentado resultado com o cálculo das perguntas certas e erradas.

Estará disponível um botão para voltar ao início.



## 2.2 - Código JAVA Public Classes

2.2.1 - Main

**public class** Main **extends** AppCompatActivity {  
 Button **obtbcomecar**;  
 Button **obtbpontacao**;  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 System.***out***.println(**"MainActivity - Iniciou"**);  
  
  
 **obtbcomecar** = (Button)findViewById(R.id.***buttonComecar***);  
 **obtbcomecar**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 accessScreen(Perguntas.**class**);  
 }  
 });  
  
 **obtbpontacao** = (Button)findViewById(R.id.***buttonPontuacao***);  
 **obtbpontacao**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 accessScreen(Ranking.**class**);  
 }  
 });  
 }  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className)  
 {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

2.2.2 - Login

**public class** Login **extends** AppCompatActivity {  
 ArrayList<Utilizador> **utilizadores**;  
 Button **obtbCriaRegisto**;  
 Button **obtbEntrar**;  
 EditText **username**;  
 EditText **password**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_login***);  
 System.***out***.println(**"Registo - Iniciou"**);  
  
 carregarDados();  
  
 *// inicia os butoes* **obtbCriaRegisto** = (Button) findViewById(R.id.***buttonRegistar***);  
 **obtbEntrar** = (Button) findViewById(R.id.***buttonLogin***);  
 **obtbEntrar** = (Button)findViewById(R.id.***buttonLogin***);  
  
 **obtbEntrar**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 **username** = findViewById(R.id.***utilizadorText***);  
 **password** = findViewById(R.id.***passwordText***);  
 **if**(podeFazerLogin()){  
 guardarInicioSessao();  
 accessScreen(Main.**class**);  
 } **else** {  
 *// mostrar toast* Toast.*makeText*(Login.**this**, **"Utilizador ou password erradas, por favor verifique novamente."**, Toast.***LENGTH\_SHORT***).show();  
 }  
 }  
 });  
  
 **obtbCriaRegisto** = (Button)findViewById(R.id.***buttonRegistar***);  
 **obtbCriaRegisto**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 accessScreen(Registar.**class**);  
 }  
 });  
 }  
  
 *// verifica se pode fazer o login c* **private boolean** podeFazerLogin() {  
 **for** (Utilizador user : **utilizadores**) {  
 *// no caso do utilizador e password corresponderem* **if** (user.**utilizador**.equals(**username**.getText().toString()) && user.**password**.equals(**password**.getText().toString())) {  
 **return true**;  
 }  
 }  
 *// no caso do utilizador e password nao corresponderem* **return false**;  
 }  
  
  
 */// carrega os dados dos utilizadores e guarda em utilizadores* **private void** carregarDados() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 Gson gson = **new** Gson();  
 String json = sharedPreferences.getString(**"utilizadores"**, **null**);  
 Type type = **new** TypeToken<ArrayList<Utilizador>>() {}.getType();  
 **utilizadores** = gson.fromJson(json, type);  
 **if** (**utilizadores** == **null**) {  
 **utilizadores** = **new** ArrayList<>();  
 *// adiciona utilizador por defeito* **utilizadores**.add(**new** Utilizador(**"admin"**,**"admin"**,**"admin@quiz.com"**,100));  
 **utilizadores**.add(**new** Utilizador(**"xavier"**,**"xavier"**,**"xavier@quiz.com"**,150));  
 **utilizadores**.add(**new** Utilizador(**"matias"**,**"matias"**,**"matias@quiz.com"**,200));  
 **utilizadores**.add(**new** Utilizador(**"roberto"**,**"roberto"**,**"roberto@quiz.com"**,250));  
 **utilizadores**.add(**new** Utilizador(**"zacarias"**,**"zacarias"**,**"zacarias@quiz.com"**,300));  
 }  
 }  
  
 *// Guarda o inicio de sessao para saber qual o utilizador logado* **private void** guardarInicioSessao() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 SharedPreferences.Editor editor = sharedPreferences.edit();  
 editor.putString(**"sessao"**, **username**.getText().toString());  
 editor.apply();  
 }  
  
 **public void** onbuttonRegistarClick(View view) {  
 **if** (view.getId() == R.id.***buttonRegistar***) {  
 accessScreen(Registar.**class**);  
 } **else if** (view.getId() == R.id.***buttonLogin***) {  
 accessScreen(Perguntas.**class**);  
  
 }  
 }  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className)  
 {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

2.2.3 - Registar

**public class** Registar **extends** AppCompatActivity {  
 ArrayList<Utilizador> **utilizadores**;  
 EditText **user**;  
 EditText **email**;  
 EditText **password**;  
 EditText **confirmarPassword**;  
 Button **registarBtn**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_criar\_conta***);  
 System.***out***.println(**"criarConta - Iniciou"**);  
  
 carregarDados();  
  
 **user** = findViewById(R.id.***utilizadorText***);  
 **email** = findViewById(R.id.***emailText***);  
 **password** = findViewById(R.id.***passwordText***);  
 **confirmarPassword** = findViewById(R.id.***confirmPasswordText***);  
  
 **registarBtn** = (Button) findViewById(R.id.***buttonRegistar***);  
 **registarBtn**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
*// System.out.println(username.getText());* **user** = findViewById(R.id.***utilizadorText***);  
 **email** = findViewById(R.id.***emailText***);  
  
 **if**(!**confirmarPassword**.getText().toString().equals(**password**.getText().toString())){  
 Toast.*makeText*(Registar.**this**, **"As passwords nao combinam"**, Toast.***LENGTH\_SHORT***).show();  
 **return**;  
 }  
 **if**(podeAdicionar()){  
 adicionarUser();  
 Toast.*makeText*(Registar.**this**, **"O utilizador adicionado"**, Toast.***LENGTH\_SHORT***).show();  
 } **else** {  
 *// mostrar toast* Toast.*makeText*(Registar.**this**, **"O utilizador ou email ja existe, por favor verifique novamente."**, Toast.***LENGTH\_SHORT***).show();  
 }  
  
 }  
 });  
 }  
  
 *// verifica se pode adicionar a nova conta* **private boolean** podeAdicionar() {  
 **for** (Utilizador user : **utilizadores**) {  
 *// no caso do utilizador e email corresponderem* **if** (user.**utilizador**.equals(**this**.**user**.getText().toString()) && user.**email**.equals(**email**.getText().toString())) {  
 **return false**;  
 }  
 }  
 *// no caso do utilizador e email nao corresponderem* **return true**;  
 }  
  
 */// Adiciona o utilizador na localstorage e redireciona para o login* **private void** adicionarUser() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 SharedPreferences.Editor editor = sharedPreferences.edit();  
 String usernameTxt = **user**.getText().toString();  
 String passwordTxt = **password**.getText().toString();  
 String emailTxt = **email**.getText().toString();  
  
 **utilizadores**.add(**new** Utilizador( usernameTxt,passwordTxt,emailTxt,0));  
 Gson gson = **new** Gson();  
 String json = gson.toJson(**utilizadores**);  
 editor.putString(**"utilizadores"**, json);  
 editor.apply();  
 accessScreen(Login.**class**);  
 }  
  
 **private void** carregarDados() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 Gson gson = **new** Gson();  
 String json = sharedPreferences.getString(**"utilizadores"**, **null**);  
 Type type = **new** TypeToken<ArrayList<Utilizador>>() {}.getType();  
 **utilizadores** = gson.fromJson(json, type);  
 **if** (**utilizadores** == **null**) {  
 **utilizadores** = **new** ArrayList<>();  
 *// adiciona utilizador por defeito* **utilizadores**.add(**new** Utilizador(**"admin"**,**"admin"**,**"admin@quiz.com"**,100));  
 **utilizadores**.add(**new** Utilizador(**"xavier"**,**"xavier"**,**"xavier@quiz.com"**,150));  
 **utilizadores**.add(**new** Utilizador(**"matias"**,**"matias"**,**"matias@quiz.com"**,200));  
 **utilizadores**.add(**new** Utilizador(**"roberto"**,**"roberto"**,**"roberto@quiz.com"**,250));  
 **utilizadores**.add(**new** Utilizador(**"zacarias"**,**"zacarias"**,**"zacarias@quiz.com"**,300));  
 }  
 }  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className)  
 {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

#### 2.2.3.1 - Utilizador

**public class** Utilizador {  
 **public** String **id**;  
 **public** String **utilizador**;  
 **public** String **password**;  
 **public** String **email**;  
 **public int pontuacao**;  
  
 **public** Utilizador( String utilizador, String password, String email, **int** pontuacao) {  
 **this**.**id** = UUID.*randomUUID*().toString();  
 **this**.**utilizador** = utilizador;  
 **this**.**password** = password;  
 **this**.**email** = email;  
 **this**.**pontuacao** = pontuacao;  
 }  
}

2.2.4 - Ranking

**public class** Ranking **extends** AppCompatActivity {  
 **private** List<Utilizador> **utilizadores** = **new** ArrayList<>();  
 ;  
 TextView **top1**, **top2**, **top3**, **top4**, **top5**;  
 TextView **labelTop1**, **labelTop2**, **labelTop3**, **labelTop4**, **labelTop5**, **labelVazio**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_ranking***);  
 System.***out***.println(**"resultado - Iniciou"**);  
  
 *// inicia as textviews* **top1** = (TextView) findViewById(R.id.***textView1Lugar***);  
 **top2** = (TextView) findViewById(R.id.***textView2Lugar***);  
 **top3** = (TextView) findViewById(R.id.***textView3Lugar***);  
 **top4** = (TextView) findViewById(R.id.***textView4Lugar***);  
 **top5** = (TextView) findViewById(R.id.***textView5Lugar***);  
 **labelTop1** = (TextView) findViewById(R.id.***labelLugar1***);  
 **labelTop2** = (TextView) findViewById(R.id.***labelLugar2***);  
 **labelTop3** = (TextView) findViewById(R.id.***labelLugar3***);  
 **labelTop4** = (TextView) findViewById(R.id.***labelLugar4***);  
 **labelTop5** = (TextView) findViewById(R.id.***labelLugar5***);  
  
 **labelVazio** = (TextView) findViewById(R.id.***textVazio***);  
 **labelVazio**.setVisibility(View.***INVISIBLE***);  
  
 carregarDados();  
 ordenarPontuacoes();  
 carregarRankings();  
 }  
  
 @SuppressLint(**"SetTextI18n"**)  
 */// Carrega a tabela de rankings e esconde as labels* **private void** carregarRankings() {  
 **if** (**utilizadores** == **null**) {  
 **labelTop1**.setVisibility(View.***INVISIBLE***);  
 **labelTop2**.setVisibility(View.***INVISIBLE***);  
 **labelTop3**.setVisibility(View.***INVISIBLE***);  
 **labelTop4**.setVisibility(View.***INVISIBLE***);  
 **labelTop5**.setVisibility(View.***INVISIBLE***);  
 **labelVazio**.setVisibility(View.***VISIBLE***);  
 **return**;  
 }  
 **top1**.setText(**utilizadores**.get(0).**utilizador** + **" - "** + **utilizadores**.get(0).**pontuacao**);  
 **top2**.setText(**utilizadores**.get(1).**utilizador** + **" - "** + **utilizadores**.get(1).**pontuacao**);  
 **top3**.setText(**utilizadores**.get(2).**utilizador** + **" - "** + **utilizadores**.get(2).**pontuacao**);  
 **top4**.setText(**utilizadores**.get(3).**utilizador** + **" - "** + **utilizadores**.get(3).**pontuacao**);  
 **top5**.setText(**utilizadores**.get(4).**utilizador** + **" - "** + **utilizadores**.get(4).**pontuacao**);  
 }  
  
 */// ordena da pontuacao maior para a mais pequena* **private void** ordenarPontuacoes() {  
 **if** (**utilizadores** == **null**) **return**;  
 Collections.*sort*(**utilizadores**, **new** Comparator<Utilizador>() {  
 **public int** compare(Utilizador u1, Utilizador u2) {  
 **if** (u1.**pontuacao** == u2.**pontuacao**)  
 **return** 0;  
 **return** u1.**pontuacao** > u2.**pontuacao** ? -1 : 1;  
 }  
 });  
 }  
  
 */// carrega todos os utilizadores* **private void** carregarDados() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 Gson gson = **new** Gson();  
 String json = sharedPreferences.getString(**"utilizadores"**, **null**);  
 Type type = **new** TypeToken<ArrayList<Utilizador>>() {  
 }.getType();  
 **utilizadores** = gson.fromJson(json, type);  
 }  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className) {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

2.2.5 - Perguntas

**public class** Perguntas **extends** AppCompatActivity {  
 ArrayList<Pergunta> **perguntas**;  
 **int indiceAtual** = 0;  
 Pergunta **perguntaAtual**;  
 Button **btnProxima**;  
 **int acertadas** = 0;  
 **int erradas** = 0;  
 Button **bR1**;  
 Button **bR2**;  
 Button **bR3**;  
 Button **bR4**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_perguntas***);  
 System.***out***.println(**"Perguntas - Iniciou"**);  
  
 *// Inicializa todos os botoes* **btnProxima** = (Button) findViewById(R.id.***butaoProximo***);  
 **bR1** = (Button) findViewById(R.id.***buttonRespostaA***);  
 **bR2** = (Button) findViewById(R.id.***buttonRespostaB***);  
 **bR3** = (Button) findViewById(R.id.***buttonRespostaC***);  
 **bR4** = (Button) findViewById(R.id.***buttonRespostaD***);  
  
 carregarPerguntas();  
 *// Inicia o questionatio no indice atual* **perguntaAtual** = **perguntas**.get(**indiceAtual**);  
  
 carregarPerguntaNaView();  
  
 **bR1**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 respostaEscolhida(**"A"**);  
 }  
 });  
  
 **bR2**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 respostaEscolhida(**"B"**);  
 }  
 });  
  
 **bR3**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 respostaEscolhida(**"C"**);  
 }  
 });  
  
 **bR4**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 respostaEscolhida(**"D"**);  
 }  
 });  
  
 */// esconde o butao proxima* **btnProxima**.setVisibility(View.***INVISIBLE***);  
  
 **btnProxima**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 mudarPergunta();  
 *// esconde o butao proxima* **btnProxima**.setVisibility(View.***INVISIBLE***);  
 *// ativa os butoes de respostas* ativarButoesDeResposta(**true**);  
 }  
 });  
 limparCoresDosButoes();  
 }  
  
  
 **private void** respostaEscolhida(String respostaSelecionada) {  
 limparCoresDosButoes();  
 **switch** (respostaSelecionada) {  
 **case "A"**:  
 **bR1**.setBackgroundColor(Color.***BLUE***);  
 **if** (**perguntaAtual**.**respostaCorreta**.equals(**bR1**.getText().toString())) {  
 **bR1**.setBackgroundColor(Color.***GREEN***);  
 **acertadas** += 1;  
 } **else** {  
 **bR1**.setBackgroundColor(Color.***RED***);  
 **erradas** += 1;  
 }  
 **break**;  
 **case "B"**:  
 **bR2**.setBackgroundColor(Color.***BLUE***);  
 **if** (**perguntaAtual**.**respostaCorreta**.equals(**bR2**.getText().toString())) {  
 **bR2**.setBackgroundColor(Color.***GREEN***);  
 **acertadas** += 1;  
  
 } **else** {  
 **bR2**.setBackgroundColor(Color.***RED***);  
 **erradas** += 1;  
 }  
 **break**;  
 **case "C"**:  
 **bR3**.setBackgroundColor(Color.***BLUE***);  
 **if** (**perguntaAtual**.**respostaCorreta**.equals(**bR3**.getText().toString())) {  
 **bR3**.setBackgroundColor(Color.***GREEN***);  
 **acertadas** += 1;  
  
 } **else** {  
 **bR3**.setBackgroundColor(Color.***RED***);  
 **erradas** += 1;  
 }  
 **break**;  
 **case "D"**:  
 **bR4**.setBackgroundColor(Color.***BLUE***);  
 **if** (**perguntaAtual**.**respostaCorreta**.equals(**bR4**.getText().toString())) {  
 **bR4**.setBackgroundColor(Color.***GREEN***);  
 **acertadas** += 1;  
 } **else** {  
 **bR4**.setBackgroundColor(Color.***RED***);  
 **erradas** += 1;  
 }  
 **break**;  
 **default**:  
 System.***out***.println(**"Resposta Invalida"**);  
 }  
 ativarButoesDeResposta(**false**);  
 }  
  
 */// desativa os botoes de resposta e mostra o butao proxima* **private void** ativarButoesDeResposta(**boolean** opcao) {  
 **bR1**.setEnabled(opcao);  
 **bR2**.setEnabled(opcao);  
 **bR3**.setEnabled(opcao);  
 **bR4**.setEnabled(opcao);  
 **btnProxima**.setVisibility(!opcao ? View.***VISIBLE*** : View.***INVISIBLE***);  
 }  
  
 */// limpa a cor dos botoes para o default* **private void** limparCoresDosButoes() {  
 **bR1**.setBackgroundColor(Color.***LTGRAY***);  
 **bR2**.setBackgroundColor(Color.***LTGRAY***);  
 **bR3**.setBackgroundColor(Color.***LTGRAY***);  
 **bR4**.setBackgroundColor(Color.***LTGRAY***);  
 }  
  
 */// carrega a pergunta selecionada para a activity* **private void** carregarPerguntaNaView() {  
 TextView questao = findViewById(R.id.***textviewPergunta***);  
 questao.setText(**perguntaAtual**.**questao**);  
 Button a = (Button) findViewById(R.id.***buttonRespostaA***);  
 a.setText(**perguntaAtual**.**resposta1**);  
 Button b = (Button) findViewById(R.id.***buttonRespostaB***);  
 b.setText(**perguntaAtual**.**resposta2**);  
 Button c = (Button) findViewById(R.id.***buttonRespostaC***);  
 c.setText(**perguntaAtual**.**resposta3**);  
 Button d = (Button) findViewById(R.id.***buttonRespostaD***);  
 d.setText(**perguntaAtual**.**resposta4**);  
 }  
  
 */// muda a pergunta para a proxima* **private void** mudarPergunta() {  
 */// incrementa para passar para a proxima pergunta* **indiceAtual** += 1;  
 */// caso o indice da pergunta seja 9 entao envia os resultados para a activity `Resultados`  
 /// 9 porque o index comeca em 0* **if** ((**indiceAtual**) == 9) { finalizar(); **return**;}  
 */// muda a pergunta para o novo index* **perguntaAtual** = **perguntas**.get(**indiceAtual**);  
 carregarPerguntaNaView();  
 limparCoresDosButoes();  
 }  
  
 **private void** finalizar() {  
 Intent intent = **new** Intent(**this**, Resultado.**class**);  
 intent.putExtra(**"totalErradas"**, **erradas** + **""**);  
 intent.putExtra(**"totalCertas"**, **acertadas** + **""**);  
 startActivity(intent);  
 }  
  
 */// Carrega as perguntas e guarda no local storage* **private void** carregarPerguntas() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 Gson gson = **new** Gson();  
 String json = sharedPreferences.getString(**"perguntas"**, **null**);  
 Type type = **new** TypeToken<ArrayList<Pergunta>>() {  
 }.getType();  
 **perguntas** = gson.fromJson(json, type);  
 **if** (**perguntas** == **null**) {  
 **perguntas** = **new** ArrayList<>();  
 *// adiciona 10 perguntas* **perguntas**.add(**new** Pergunta(**"De quem é a famosa frase “Penso, logo existo”?"**, **"Platão"**, **"Galileu Galilei"**, **"Descartes"**, **"Sócrates"**, **"Descartes"**));  
 **perguntas**.add(**new** Pergunta(**"De onde é a invenção do chuveiro elétrico?"**, **"França"**, **"Austrália"**, **"Itália"**, **"Brasil"**, **"Brasil"**));  
 **perguntas**.add(**new** Pergunta(**"Quantas casas decimais tem o número pi?"**, **"Duas"**, **"Centenas"**, **"Infinitas"**, **"Milhares"**, **"Infinitas"**));  
 **perguntas**.add(**new** Pergunta(**"Atualmente, quantos elementos químicos a tabela periódica possui?"**, **"113"**, **"109"**, **"108"**, **"265"**, **"108"**));  
 **perguntas**.add(**new** Pergunta(**"O que a palavra legend significa em português?"**, **"Legenda"**, **"Conto"**, **"Legendário"**, **"Lenda"**, **"Lenda"**));  
 **perguntas**.add(**new** Pergunta(**"Qual o número mínimo de jogadores numa partida de futebol?"**, **"7"**, **"12"**, **"13"**, **"15"**, **"7"**));  
 **perguntas**.add(**new** Pergunta(**"Quanto tempo a luz do Sol demora para chegar à Terra?"**, **"12 minutos"**, **"1 dia"**, **"segundos"**, **"8 minutos"**, **"8 minutos"**));  
 **perguntas**.add(**new** Pergunta(**"Qual a nacionalidade de Che Guevara?"**, **"Cubana"**, **"Peruana"**, **"Panamenha"**, **"Argentina"**, **"Argentina"**));  
 **perguntas**.add(**new** Pergunta(**"Em que período da pré-história o fogo foi descoberto?"**, **"Neolítico"**, **"Paleolítico"**, **"Idade Média"**, **"Jurassico"**, **"Paleolítico"**));  
 **perguntas**.add(**new** Pergunta(**"Em qual local da Ásia o português é língua oficial?"**, **"Índia"**, **"Filipinas"**, **"Moçambique"**, **"Macau"**, **"Macau"**));  
 }  
 }  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className) {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

#### 2.2.5.1 - Pergunta

**public class** Pergunta {  
 **public** String **id**;  
 **public** String **questao**;  
 **public** String **resposta1**;  
 **public** String **resposta2**;  
 **public** String **resposta3**;  
 **public** String **resposta4**;  
 **public** String **respostaCorreta**;  
  
 **public** Pergunta( String questao, String resposta1, String resposta2, String resposta3, String resposta4, String respostaCorreta) {  
 **this**.**id** = UUID.*randomUUID*().toString();  
 **this**.**questao** = questao;  
 **this**.**resposta1** = resposta1;  
 **this**.**resposta2** = resposta2;  
 **this**.**resposta3** = resposta3;  
 **this**.**resposta4** = resposta4;  
 **this**.**respostaCorreta** = respostaCorreta;  
 }  
}

2.2.6 - Resultado

**public class** Resultado **extends** AppCompatActivity {  
 ArrayList<Utilizador> **utilizadores**;  
 String **utilizadorLogado**;  
 **int pontuacao**;  
 Button **obtbvoltarinicio**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_resultado***);  
 System.***out***.println(**"resultado - Iniciou"**);  
  
 carregarDados();  
  
 Bundle b = getIntent().getExtras();  
 String totalErradas = b.getString(**"totalErradas"**);  
 String totalCertas = b.getString(**"totalCertas"**);  
  
 TextView erradas = (TextView) findViewById(R.id.***textViewErradas***);  
 TextView certas = (TextView) findViewById(R.id.***textViewCertas***);  
  
 erradas.setText(totalErradas);  
 certas.setText(totalCertas);  
 **pontuacao** = ((1000 \* Integer.*parseInt*(totalCertas)) / Integer.*parseInt*(totalErradas));  
 salvarResultado();  
  
 **obtbvoltarinicio** = (Button)findViewById(R.id.***buttonVoltar***);  
 **obtbvoltarinicio**.setOnClickListener (**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 accessScreen(Main.**class**);  
 }  
 });  
  
  
 }  
  
 **private void** salvarResultado() {  
 **for** (Utilizador user : **utilizadores**) {  
 *// no caso do utilizador e email corresponderem* **if** (user.**utilizador**.equals(**utilizadorLogado**)) {  
 user.**pontuacao** = **pontuacao**;  
 }  
 }  
 salvarUtilizador();  
 }  
  
 **private void** salvarUtilizador() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 SharedPreferences.Editor editor = sharedPreferences.edit();  
 Gson gson = **new** Gson();  
 String json = gson.toJson(**utilizadores**);  
 editor.putString(**"utilizadores"**, json);  
 editor.apply();  
 }  
  
  
 **private void** carregarDados() {  
 SharedPreferences sharedPreferences = getSharedPreferences(**"quizzy"**, ***MODE\_PRIVATE***);  
 Gson gson = **new** Gson();  
 String json = sharedPreferences.getString(**"utilizadores"**, **null**);  
 Type type = **new** TypeToken<ArrayList<Utilizador>>() {  
 }.getType();  
 **utilizadores** = gson.fromJson(json, type);  
 */// recebe o utilizador logado e guarda* **utilizadorLogado** = sharedPreferences.getString(**"sessao"**, **null**);  
 }  
  
  
  
  
 */// acede a activity passada na [className]* **private void** accessScreen(Class className) {  
 Intent intent = **new** Intent(**this**, className);  
 startActivity(intent);  
 }  
}

# Mockup



Obrigado,

Daniel ||| Bruno