



**Escola Superior
de Tecnologia
e Gestão**

Politécnico de Coimbra

Instruções de compilação e execução

EzShop

Autores:

Francisco Mesquita – 2018056868

Luís Chaves – 2018056407

Data: 2021/07/16

Índice

Informação geral.....	5
1.1. Visão geral do Sistema	5
1.2. Requisitos para utilizar o sistema.....	5
2. Backend	6
3. Aplicação Android	7
4. Model Maker.....	8

Informação geral

1.1. Visão geral do Sistema

A EzShop é uma aplicação de listas de compras com a funcionalidade de registo de produtos através de classificação de imagens. Desta forma por um lado temos uma lista de compras que pode ser partilhada por várias pessoas e por outro temos a adição de produtos usando reconhecimento de imagens.

1.2. Requisitos para utilizar o sistema

Para executar a aplicação é necessário ter um dispositivo Android com versão superior ou igual à 5.0 (Android Lollipop) e ter conexão à internet.

Para executar o *backend* é necessária conexão à internet para descarregar a gradle e suas dependências. Uma base de dados MySQL, de preferência a última versão.

Na geração do modelo, é necessário o Python 3.7 e conexão à internet para descarregar diversas dependências.

2. Backend

Para persistência de dados no *backend* é necessária a instalação de um SGBD, recomendamos o uso do MySQL 8.

Deverá ser criada uma base de dados com o nome desejado, e um utilizador que irá aceder à mesma.

```
root@Projeto:~# mysql -u root -p
```

```
mysql> CREATE USER 'projeto'@'localhost' IDENTIFIED WITH  
mysql_native_password BY 'projeto'; // Utilize uma password segura
```

```
mysql> CREATE DATABASE projeto_dev;
```

```
mysql> GRANT ALL ON projeto.* TO 'projeto'@'%';
```

```
mysql> FLUSH PRIVILEGES;
```

A ligação do Ktor ao MySQL está configurada no ficheiro `src/main/kotlin/pt/ipc/estgoh/francisco_luis/plugins/DBConfig.kt`.

A configuração anterior é utilizada no seguinte *snippet*.

```
val dataSource = HikariDataSource().apply {  
    username = "projeto"  
    password = "projeto"  
    driverClassName = "com.mysql.cj.jdbc.Driver"  
    jdbcUrl = "jdbc:mysql://localhost:3306/projeto"  
}
```

Para iniciar o servidor web, basta correr o seguinte comando a partir do diretório *root* do *backend*.

```
root@Projeto:~/backend# ./gradlew run
```

Opcionalmente pode incluir as *flags* **--stacktrace** e **--debug**.

As tabelas serão automaticamente geradas ao correr o *backend* pela primeira vez. Deverá ainda executar manualmente o *script* de inserção das categorias e itens padrão.

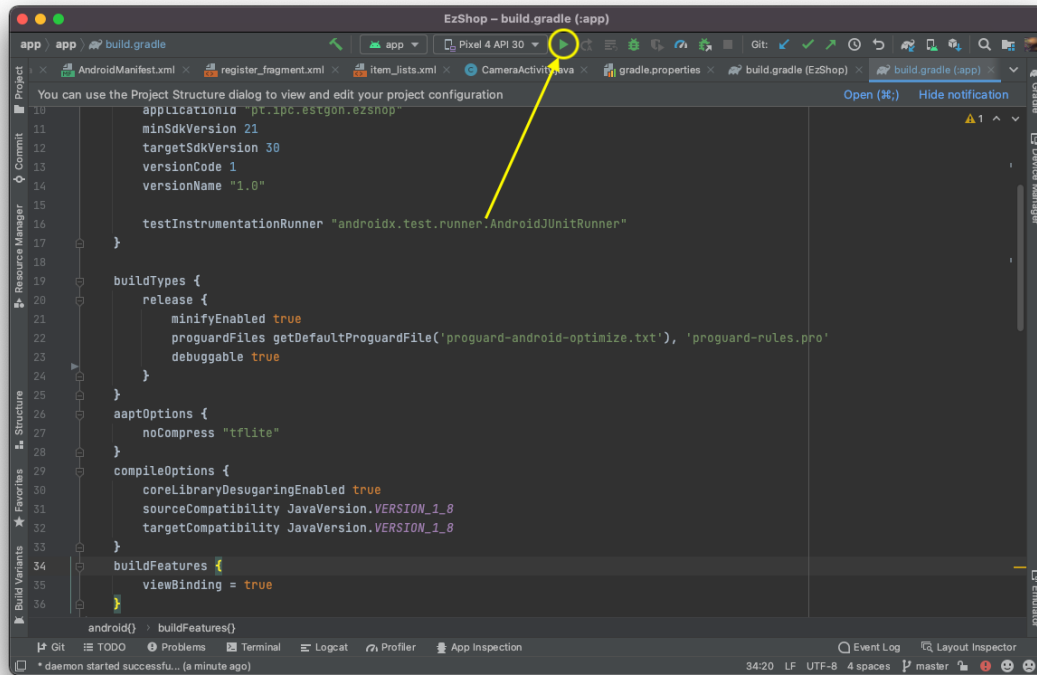
```
root@Projeto:~/backend# mysql -u projeto -p projeto < inserts.sql
```

O servidor irá correr no porto 8080.

3. Aplicação Android

A aplicação foi desenvolvida para o sistema Android. Para executar o código, deverá então utilizar o IDE Android Studio. Disponibilizámos também um *.apk ready to install* junto com o projeto.

O IDE irá descarregar a gradle, as suas dependências e indexar o projeto. Após estas tarefas, deverá clicar no botão *run* para executar o projeto.



Deverá alterar no ficheiro `app/src/main/java/pt/ipc/estgoh/ezshop/data/api/API.java` o IP ou domínio do servidor onde está localizada a base de dados.

```
private static final String BASE_URL = "http://IP DO SERVIDOR:8080/";
```

Nota: caso utilize o *localhost*, deverá inserir o IP local da máquina **e não 127.0.0.1**;

4. Model Maker

Para gerar o modelo deverá possuir o Python 3.7 e descarregar as seguintes dependências com o **pip**:

tensorflow

tf-lite-model-maker

firebase-admin

No ficheiro *main.py*, defina a localização das imagens na seguinte variável:

```
data_dir = "./food_photos"
```

Seguidamente, execute o *script* para gerar o modelo.

```
python main.py
```

Após finalizar a tarefa, o modelo será gerado e guardado na diretoria *models*. É também carregado para o servidor Firebase, ficando disponível na aplicação android