

ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

Programação Para Dispositivos Móveis I

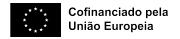
RETROFIT

2024/_25 CTeSP – Desenvolvimento para a Web e Dispositivos Móveis Ricardo Barbosa , rmb@estg.ipp.pt
Carlos Aldeias, cfpa@estg.ipp.pt

Adaptação do conteúdo dos slides de João Ramos <u>irmr@estq.ipp.pt</u> e Fábio Silva <u>fas@estq.ipp.pt</u>







Índice

- Retrofit;
- Configuração;
- Exemplo Retrofit;
- Leitura Adicional.









É um cliente HTTP que abstrai a complexidade na comunicação de uma aplicação Android com serviços REST.

- A biblioteca expõe a API de um servidor em forma de uma interface em Java;
- Permite executar diferentes tipos de pedidos: GET, POST, PUT, DELETE, etc;
- Converte automaticamente a resposta do servidor em objetos Java (Gson Converter);
- Permite executar pedidos assíncronos ou síncronos;
- Possuí boas capacidades de logging (OkHttp 3 Http Logging Interceptor).









Configuração [AndroidManifest.xml]









Configuração [build.gradle (Module)]

```
dependencies {
    implementation 'com.squareup.retrofit2:retrofit:2.11.0'
    implementation 'com.squareup.retrofit2:converter-gson:2.11.0'
    ...
}
```









Exemplo

Vamos utilizar a API da Tour-Pedia como exemplo. A mesma pode ser encontrada na ligação http://tour-pedia.org/api/ e retorna respostas no formato JSON com pontos de interesse de várias cidades europeias.

A documentação para a utilização da API está presente na hiperligação.

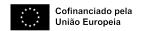
É possível testar algumas queries através do browser ou de uma aplicação externa (ex. Postman).

http://tour-pedia.org/api/getPlaces?location=Amsterdam&category=attraction&name=Science http://tour-pedia.org/api/getPlaces?location=London&name=Restaurant









Configuração [AndroidManifest.xml]

```
<application
    android:usesCleartextTraffic="true"
    ...
</application>
```

A partir das versões 8.0, como a API não é HTTPS é necessário adicionar a sequinte propriedade









Modelo de Dados

O próximo passo é criar o modelo de dados semelhante ao que é retornado pelo servidor:

Devemos criar uma classe Java que representa o modelo de dados do servidor

O Retrofit vai recorrer ao conversor Gson para **automaticamente** converter a resposta JSON do servidor num objeto Java. Esta classe Java deve ser uma copia exata da resposta JSON, isto é, o **nome** e **tipo de dados** das variáveis da classe devem fazer match com o JSON

Mais informação de como é feita a conversão automática: https://github.com/google/gson/blob/master/UserGuide.md









Modelo de Dados

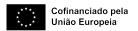
```
"id": 216741,
  "name": "Natural History and Science Museums",
  "address": "London, , United Kingdom",
  "category": "attraction",
  "location": "London",
  "lat": 51.496795820542,
  "lng": -0.17435263598631,
  "details": "http://tour-pedia.org/api/getPlaceDetails?id=216741"
}
```





Exemplo de um resultado de um pedido GET à API (em formato





Modelo de Dados [Attraction.java]

```
"id": 216741,
  "name": "Natural History and Science Museums",
  "address": "London, , United Kingdom",
  "category": "attraction",
  "location": "London",
  "lat": 51.496795820542,
  "lng": -0.17435263598631,
  "details": "http://tour-pedia.org/api/getPlaceDetails?id=216741"
}
```

Podemos ter nomes de variáveis diferentes das variáveis do JSON, contudo devemos especificar o seu correspondente Modelo de Dados para corresponder aos nomes e tipos de dados recebidos

```
public class Attraction {
    private String name;
    private String address;
    private String details;
    aSerializedName("lat")
    private float latitude;
   @SerializedName("lng")
    private float longitude;
```









Interface API [TourDataApi.java (Interface)]

```
Correspondência aos pares chave-
valor presentes na API

@GET("/api/getPlaces")
Call<List<Attraction>> getAttractionList(@Query("location") String location,
```

Retorno utilizado para associação ao modelo de dados

http://tour-pedia.org/api/getPlaces?location=London&category=attraction&name=Science





@Query("category") String category,

aQuery("name") String name);





Instância da API [MainActivity.java]









Execução de um pedido assíncrono [MainActivity.java]









Retrofit Questão #1

Os pedidos do Retrofit são executados na UI Thread?









Retrofit Questão #2

Para que é utilizada a biblioteca GSON no exemplo?









Leitura Adicional

Retrofit:

https://square.github.io/retrofit/











ESCOLA
SUPERIOR
DE TECNOLOGIA
E GESTÃO

Programação Para Dispositivos Móveis I

RETROFIT

2024/_25 CTeSP – Desenvolvimento para a Web e Dispositivos Móveis Ricardo Barbosa , rmb@estg.ipp.pt
Carlos Aldeias, cfpa@estg.ipp.pt

Adaptação do conteúdo dos slides de João Ramos <u>irmr@estq.ipp.pt</u> e Fábio Silva <u>fas@estq.ipp.pt</u>





