**Lab 11: Restful API**

**Xây dựng ứng dụng tương tác Restful API trong Android**

**Yêu cầu:**

1. Dùng Mockapi.io để quản lý cơ sở dữ liệu và cung cấp API
2. Ứng dụng call API thực hiện các phương thức Get, Post, Put, Delete đáp ứng yêu cầu CRUD
3. Hình sau đây minh họa trường hợp Create thông tin và call API lưu trữ trên database trong Mockapi
4. Thực hiện lần lượt các chức năng Xem, Sửa, Xóa trên giao diện ứng dụng

Graphical user interface, application

Description automatically generated Graphical user interface, text, application

Description automatically generated

**Hướng dẫn:**

1. Tạo dự án mới có tên: FeedbackManagementSystem
   * Thêm dependencies

A close-up of a sign

Description automatically generated

* + Chọn Sync Now

1. Tạo Model
   * Tạo package **model** 
     + Tại Model tạo Class Trainee

A close up of words

Description automatically generated

* + - Tạo Geter, setter cho class Trainee

1. Tạo Interface cung cấp các anotation cho từng phương thức GET, POST, PUT, DELETE
   * Tạo mới model api
   * Tại api tạo interface TraineeService

Text

Description automatically generated

1. Tạo thể hiện của Retrofit để gửi Request đến API

Graphical user interface, text, application, email

Description automatically generated

1. Tại API Tạo Class TraineeRepository

Graphical user interface, text, application

Description automatically generated

1. Thêm quyền truy cập Internet cho ứng dụng trong Manifest.xml
2. Tương tác với API
   * Thiết kế layout cho MainActivity.xml

Graphical user interface, application

Description automatically generated

* + Code MainActivity.java

Text

Description automatically generated

Graphical user interface, text, application

Description automatically generated

1. Truy cập vào link **mockapi.io** thực hiện các bước sau
2. Đăng ký tài khoản mới hoặc login bằng tài khoản google
3. Tạo một project mới có tên Demoapp
4. Chọn New Resource tạo các thuộc tính cho bảng Nhanvien như sau:

Table

Description automatically generated

1. Copy link API trên web như sau:

Graphical user interface, text, chat or text message

Description automatically generated



1. **Tự làm: Các chức năng Xem, Sửa Xóa**

Gợi ý code:Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

* Xóa

Graphical user interface, text, application, email

Description automatically generated

### 🛠 ****1. Cấu trúc thư mục dự án****

pgsql

Sao chépChỉnh sửa

com.example.api

│── api

│ │── TraineeRepository.java

│ └── TraineeService.java

│── model

│ └── Trainee.java

│── MainActivity.java

│── layout

│ │── activity\_main.xml

│ └── trainee\_item.xml

## ****📌 1. TraineeRepository.java****

💡 **Mô tả**: Class này chịu trách nhiệm tạo Retrofit instance để kết nối với MockAPI.

java

Sao chépChỉnh sửa

package com.example.api.api;

import retrofit2.Retrofit;import retrofit2.converter.gson.GsonConverterFactory;

public class TraineeRepository {

private static final String BASE\_URL = "https://67d114fb825945773eb2f4ab.mockapi.io/v1/"; // Thay bằng URL của bạn

private TraineeService traineeService;

// Constructor khởi tạo Retrofit

public TraineeRepository() {

Retrofit retrofit = new Retrofit.Builder()

.baseUrl(BASE\_URL) // Thiết lập URL API

.addConverterFactory(GsonConverterFactory.create()) // Chuyển đổi JSON thành Java Object

.build();

traineeService = retrofit.create(TraineeService.class);

}

// Getter để lấy instance của TraineeService

public TraineeService getTraineeService() {

return traineeService;

}

}

## ****📌 2. TraineeService.java****

💡 **Mô tả**: Interface định nghĩa các API call đến MockAPI.

java

Sao chépChỉnh sửa

package com.example.api.api;

import com.example.api.model.Trainee;

import java.util.List;

import retrofit2.Call;import retrofit2.http.Body;import retrofit2.http.DELETE;import retrofit2.http.GET;import retrofit2.http.POST;import retrofit2.http.PUT;import retrofit2.http.Path;

public interface TraineeService {

// Lấy danh sách tất cả nhân viên

@GET("nhanvien")

Call<List<Trainee>> getAllTrainees();

// Tạo mới một nhân viên

@POST("nhanvien")

Call<Trainee> createTrainee(@Body Trainee trainee);

// Cập nhật thông tin nhân viên theo ID

@PUT("nhanvien/{id}")

Call<Trainee> updateTrainee(@Path("id") String id, @Body Trainee trainee);

// Xóa nhân viên theo ID

@DELETE("nhanvien/{id}")

Call<Void> deleteTrainee(@Path("id") String id);

}

## ****📌 3. Trainee.java****

💡 **Mô tả**: Class model đại diện cho dữ liệu nhân viên.

java

Sao chépChỉnh sửa

package com.example.api.model;

public class Trainee {

private String id;

private String name;

private String email;

private String phone;

private String gender;

// Constructor không tham số (dùng cho Gson)

public Trainee() {}

// Constructor có tham số

public Trainee(String name, String email, String phone, String gender) {

this.name = name;

this.email = email;

this.phone = phone;

this.gender = gender;

}

// Getter và Setter

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPhone() {

return phone;

}

public void setPhone(String phone) {

this.phone = phone;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

}

## ****📌 4. MainActivity.java****

💡 **Mô tả**: Class chính điều khiển giao diện và gọi API.

java

Sao chépChỉnh sửa

package com.example.api;

import android.os.Bundle;import android.view.View;import android.widget.ArrayAdapter;import android.widget.Button;import android.widget.EditText;import android.widget.ListView;import android.widget.TextView;import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.example.api.api.TraineeRepository;import com.example.api.model.Trainee;

import java.util.ArrayList;import java.util.List;

import retrofit2.Call;import retrofit2.Callback;import retrofit2.Response;

public class MainActivity extends AppCompatActivity {

private EditText etName, etEmail, etPhone, etGender;

private ListView listViewTrainees;

private TraineeRepository traineeRepository;

private ArrayAdapter<Trainee> adapter;

private List<Trainee> traineeList = new ArrayList<>();

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

etName = findViewById(R.id.et\_name);

etEmail = findViewById(R.id.et\_email);

etPhone = findViewById(R.id.et\_phone);

etGender = findViewById(R.id.et\_gender);

listViewTrainees = findViewById(R.id.list\_view\_trainees);

Button btnRefresh = findViewById(R.id.btn\_refresh);

traineeRepository = new TraineeRepository();

// Bấm nút để tải danh sách nhân viên

btnRefresh.setOnClickListener(v -> refreshList());

adapter = new ArrayAdapter<>(this, android.R.layout.simple\_list\_item\_1, traineeList);

listViewTrainees.setAdapter(adapter);

}

// Phương thức lưu nhân viên mới

public void saveTrainee(View view) {

String name = etName.getText().toString();

String email = etEmail.getText().toString();

String phone = etPhone.getText().toString();

String gender = etGender.getText().toString();

Trainee trainee = new Trainee(name, email, phone, gender);

traineeRepository.getTraineeService().createTrainee(trainee)

.enqueue(new Callback<Trainee>() {

@Override

public void onResponse(Call<Trainee> call, Response<Trainee> response) {

if (response.body() != null) {

traineeList.add(response.body());

adapter.notifyDataSetChanged();

Toast.makeText(MainActivity.this, "Lưu thành công", Toast.LENGTH\_SHORT).show();

}

}

@Override

public void onFailure(Call<Trainee> call, Throwable t) {

Toast.makeText(MainActivity.this, "Lỗi khi lưu", Toast.LENGTH\_SHORT).show();

}

});

}

// Tải danh sách nhân viên từ API

public void refreshList() {

traineeRepository.getTraineeService().getAllTrainees()

.enqueue(new Callback<List<Trainee>>() {

@Override

public void onResponse(Call<List<Trainee>> call, Response<List<Trainee>> response) {

if (response.body() != null) {

traineeList.clear();

traineeList.addAll(response.body());

adapter.notifyDataSetChanged();

}

}

@Override

public void onFailure(Call<List<Trainee>> call, Throwable t) {

Toast.makeText(MainActivity.this, "Lỗi tải danh sách", Toast.LENGTH\_SHORT).show();

}

});

}

}

## 📌 ****1.**** activity\_main.xml

💡 **Mô tả**: Đây là giao diện chính của ứng dụng, bao gồm:

* Các ô nhập dữ liệu: EditText để nhập **Tên**, **Email**, **Số điện thoại**, **Giới tính**.
* Nút **Lưu nhân viên** (Button).
* Nút **Tải danh sách** (Button).
* Danh sách nhân viên (ListView).

xml

Sao chépChỉnh sửa

<?xml version="1.0" encoding="utf-8"?><LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:padding="16dp">

<!-- Tiêu đề -->

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Quản lý Nhân viên"

android:textSize="22sp"

android:textStyle="bold"

android:layout\_gravity="center"

android:paddingBottom="16dp" />

<!-- Nhập Tên -->

<EditText

android:id="@+id/et\_name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Nhập tên nhân viên" />

<!-- Nhập Email -->

<EditText

android:id="@+id/et\_email"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Nhập email" />

<!-- Nhập Số điện thoại -->

<EditText

android:id="@+id/et\_phone"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Nhập số điện thoại" />

<!-- Nhập Giới tính -->

<EditText

android:id="@+id/et\_gender"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:hint="Nhập giới tính (Nam/Nữ)" />

<!-- Nút Lưu nhân viên -->

<Button

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Lưu nhân viên"

android:onClick="saveTrainee"

android:backgroundTint="@color/teal\_700"

android:textColor="@android:color/white"

android:layout\_marginTop="10dp" />

<!-- Nút Tải danh sách -->

<Button

android:id="@+id/btn\_refresh"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Tải danh sách nhân viên"

android:backgroundTint="@color/purple\_500"

android:textColor="@android:color/white"

android:layout\_marginTop="10dp" />

<!-- Danh sách nhân viên -->

<ListView

android:id="@+id/list\_view\_trainees"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:divider="@android:color/darker\_gray"

android:dividerHeight="1dp"

android:paddingTop="10dp" />

</LinearLayout>

## 📌 ****2.**** trainee\_item.xml

💡 **Mô tả**: Đây là layout hiển thị từng nhân viên trong danh sách ListView.

xml

Sao chépChỉnh sửa

<?xml version="1.0" encoding="utf-8"?><LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:padding="12dp"

android:background="@android:color/white"

android:layout\_marginBottom="8dp"

android:elevation="2dp">

<!-- Tên nhân viên -->

<TextView

android:id="@+id/tv\_name"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Tên nhân viên"

android:textSize="18sp"

android:textStyle="bold"

android:textColor="@android:color/black" />

<!-- Email -->

<TextView

android:id="@+id/tv\_email"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Email"

android:textSize="14sp"

android:textColor="@android:color/darker\_gray" />

<!-- Số điện thoại -->

<TextView

android:id="@+id/tv\_phone"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Số điện thoại"

android:textSize="14sp"

android:textColor="@android:color/darker\_gray" />

<!-- Giới tính -->

<TextView

android:id="@+id/tv\_gender"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Giới tính"

android:textSize="14sp"

android:textColor="@android:color/darker\_gray" />

</LinearLayout>