



HUST

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY

ONE LOVE. ONE FUTURE.



TRƯỜNG ĐẠI HỌC
BÁCH KHOA HÀ NỘI
HANOI UNIVERSITY
OF SCIENCE AND TECHNOLOGY

LẬP TRÌNH ỨNG DỤNG DI ĐỘNG

Mobile Application Programming

ET4710

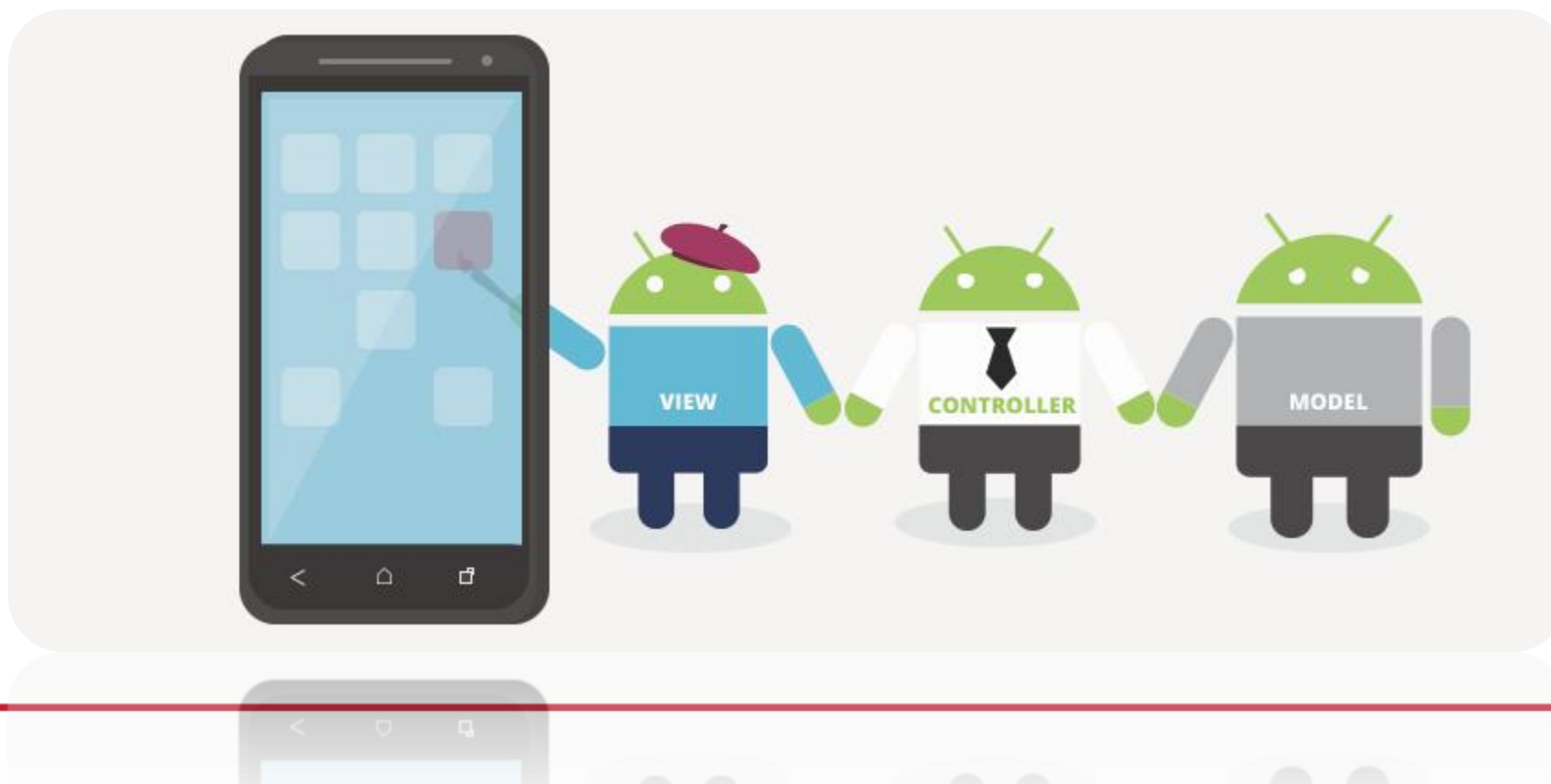
PGS. TS. Đỗ Trọng Tuấn

Viện Điện tử Viễn thông * Đại học Bách Khoa Hà Nội

ONE LOVE. ONE FUTURE.

Chương 2

Mô hình MVC, bảng kê và vòng đời của Activity trong Android (MVC model, Manifest and Activity Lifecycle)



Chương 2

Mô hình MVC, bảng kê và vòng đời của Activity trong Android (MVC model, Manifest and Activity Lifecycle)

2.1. Mô hình thiết kế ứng dụng di động MVC

(Mobile MVC model Design and Implementation)

2.2. Cấu trúc Manifest trong Android (Android Manifest Introduction)

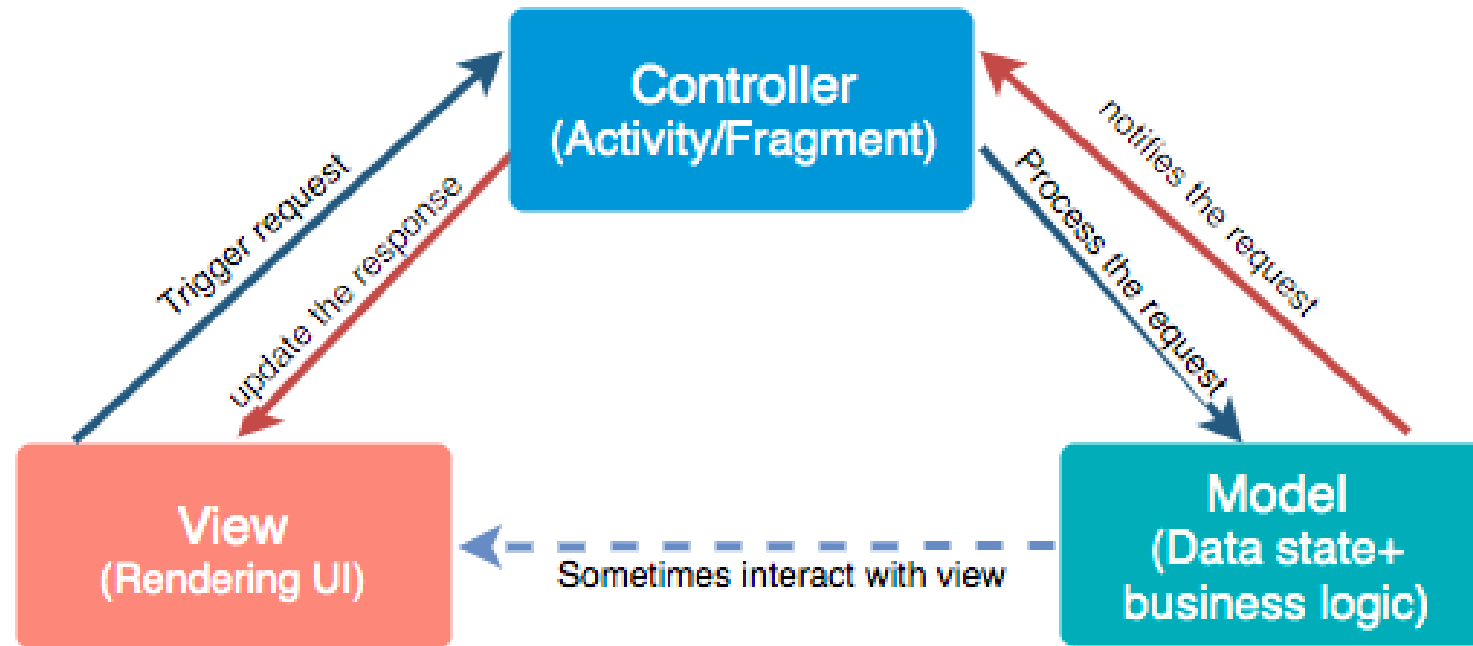
2.3 Vòng đời Activity (Activity Lifecycle)

2.4 Gỡ lỗi trong phát triển di động (Mobile Development Debugging)

2.5 Các phiên bản SDK và sự tương thích (SDK versions and their compatibility)

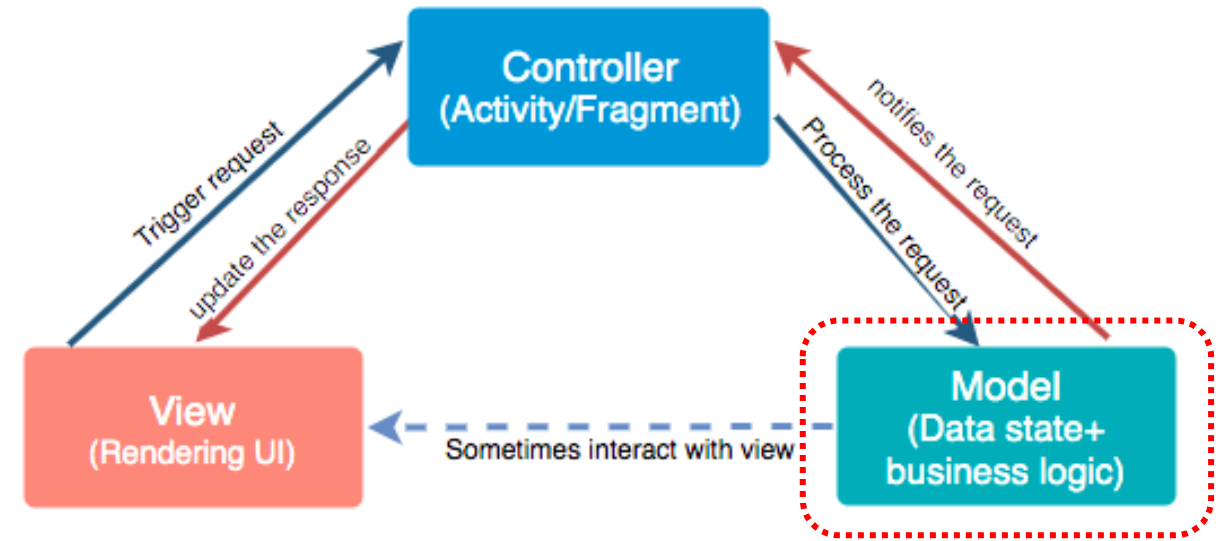
2.1. Mô hình thiết kế ứng dụng di động MVC

- **Android Architecture** is very important for **organized applications** with some set of rules, describe proper functionalities implementing with proper protocols.
- **Model View Controller (MVC)** is one of the most common **architectural patterns** in software.



2.1. Mô hình thiết kế ứng dụng di động MVC

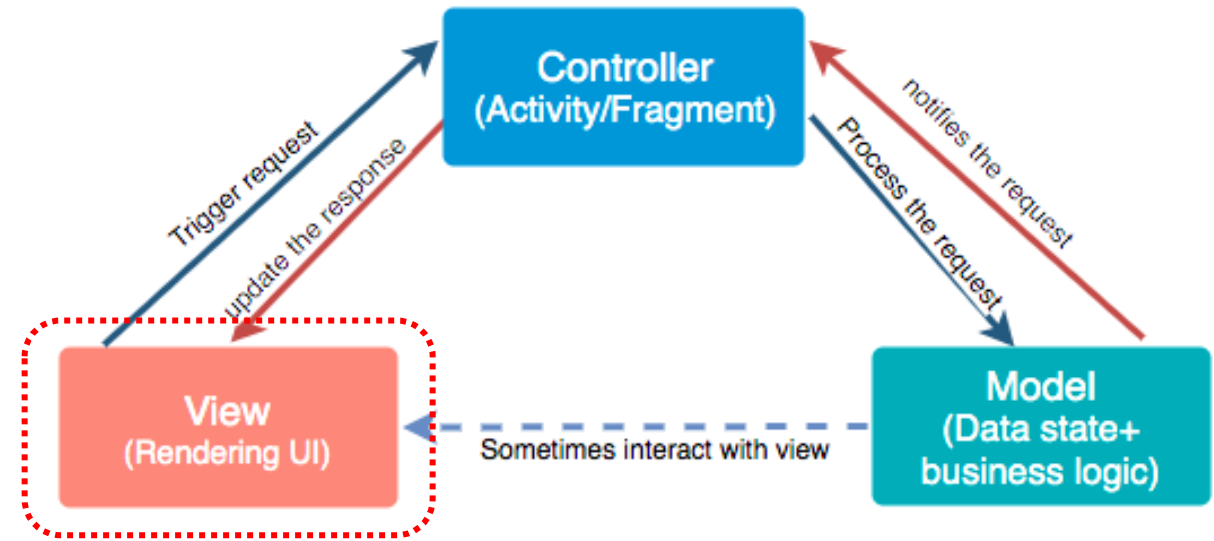
- A model object holds the application's data and “business logic.”
- Model objects have no knowledge of the UI; their sole purpose is holding and managing data.



- Model classes are typically designed to model the things your app is concerned with, such as a user, a product in a store, a photo on a server, a television show – or a true-false question.
- In Android applications, model classes are generally custom classes you create. All of the model objects in your application compose its model layer.

2.1. Mô hình thiết kế ứng dụng di động MVC

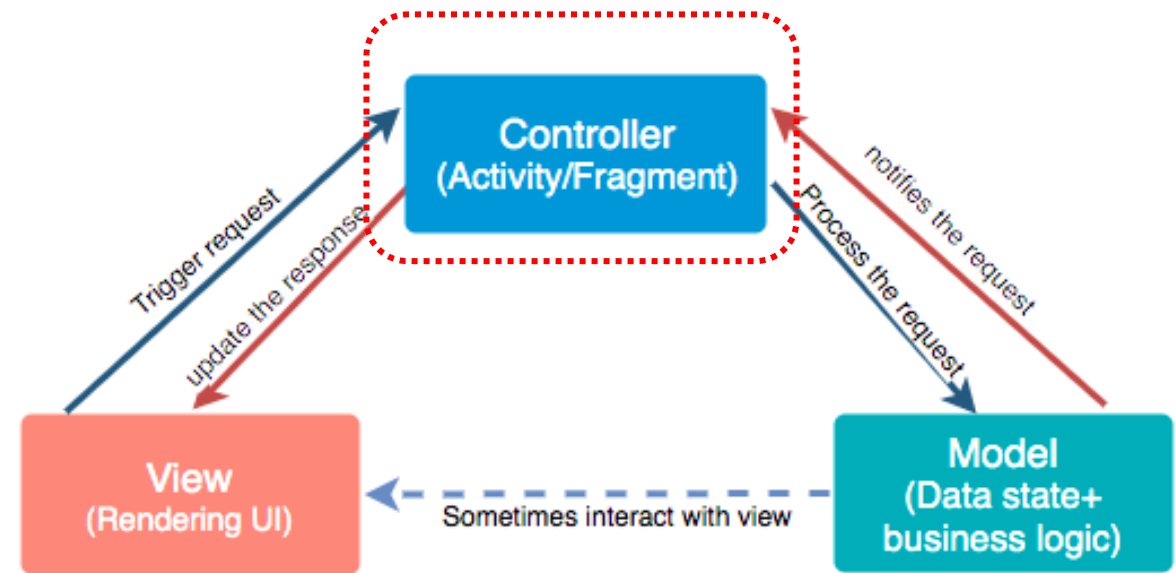
- View objects know how to draw themselves on the screen and how to respond to user input, like touches.
- A simple rule of thumb is that if you can see it on screen, then it is a view.



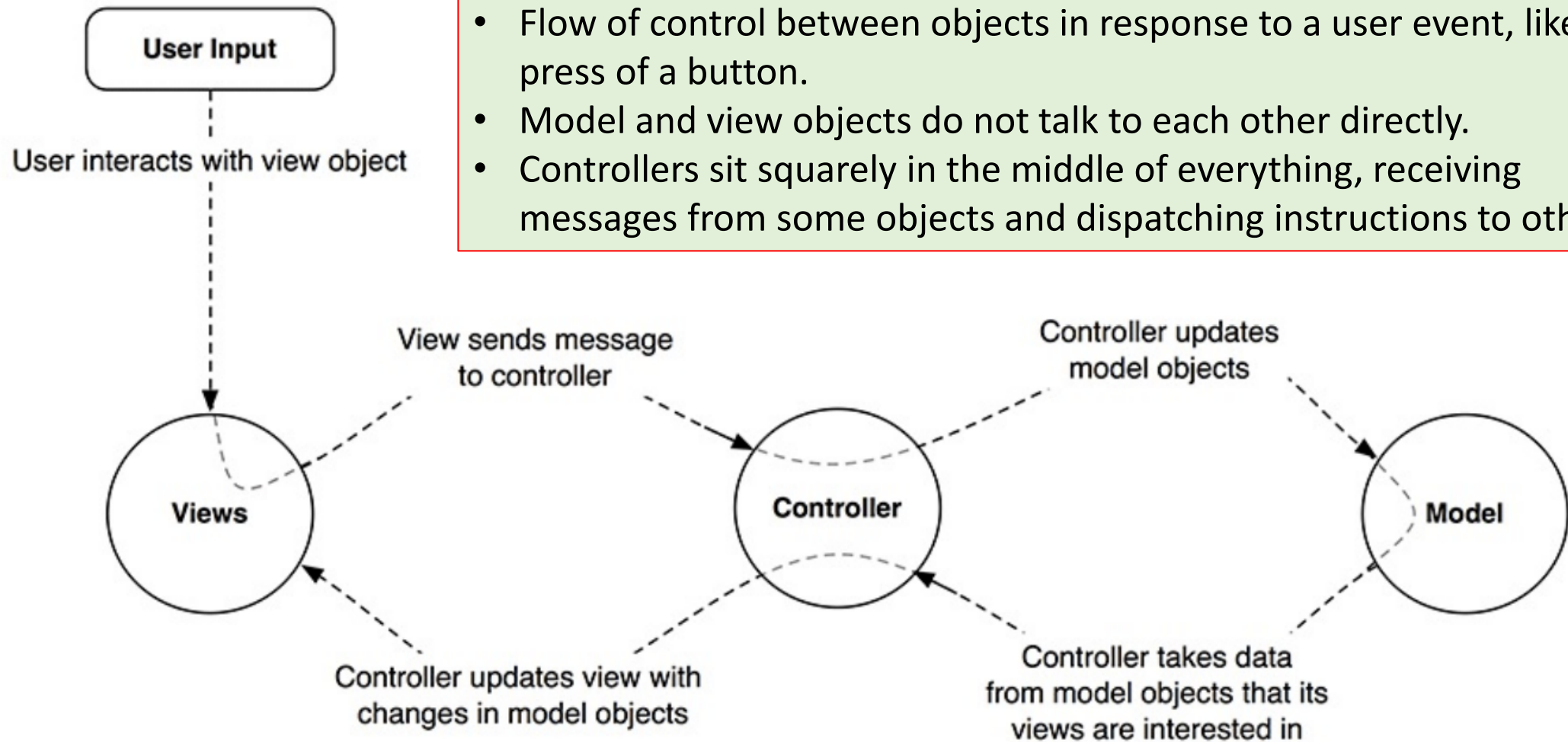
- Android provides a wealth of configurable view classes.
- You can also create custom view classes.
- An application's view objects make up its view layer.

2.1. Mô hình thiết kế ứng dụng di động MVC

- Controller objects tie the view and model objects together. They contain “**application logic**.”
- Controllers are designed to respond to various events triggered by view objects and to manage the flow of data to and from model objects and the view layer.
- In Android, a controller is typically a subclass of Activity, Fragment, or Service.



2.1. Mô hình thiết kế ứng dụng di động MVC



MVC flow with user input

2.1. Mô hình thiết kế ứng dụng di động MVC

- An application can accumulate features until it is too complicated to understand
- Separating code into classes helps you design and understand the application as a whole
- You can think in terms of classes instead of individual variables and methods

Benefits of MVC

- Separating classes into Model, View, and Controller Layers helps you design and understand an application;
- You can think in terms of layers instead of individual classes.

2.1. Mô hình thiết kế ứng dụng di động MVC

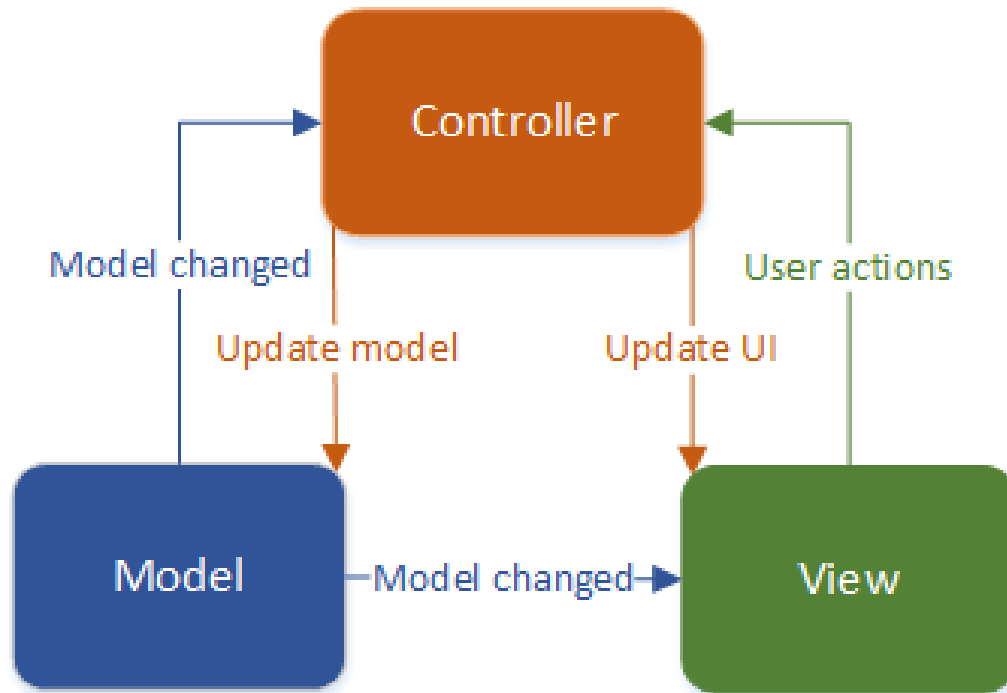
- Proper MVC implementation has the following characteristics:
 - ✓ Readable and maintainable code
 - ✓ Decoupled responsibilities
 - ✓ Modular design
 - ✓ Testable code
 - ✓ Code which is fun to work with
- Writing our code in a class/activity/ fragment in an unorganized manner then facing the problems:
 - ✓ The number of lines of code will increase that it will become complex to understand.
 - ✓ It decreases readability and increases the number of bugs. Thus, it is difficult to test and reduces the quality of the product.

Clean Code

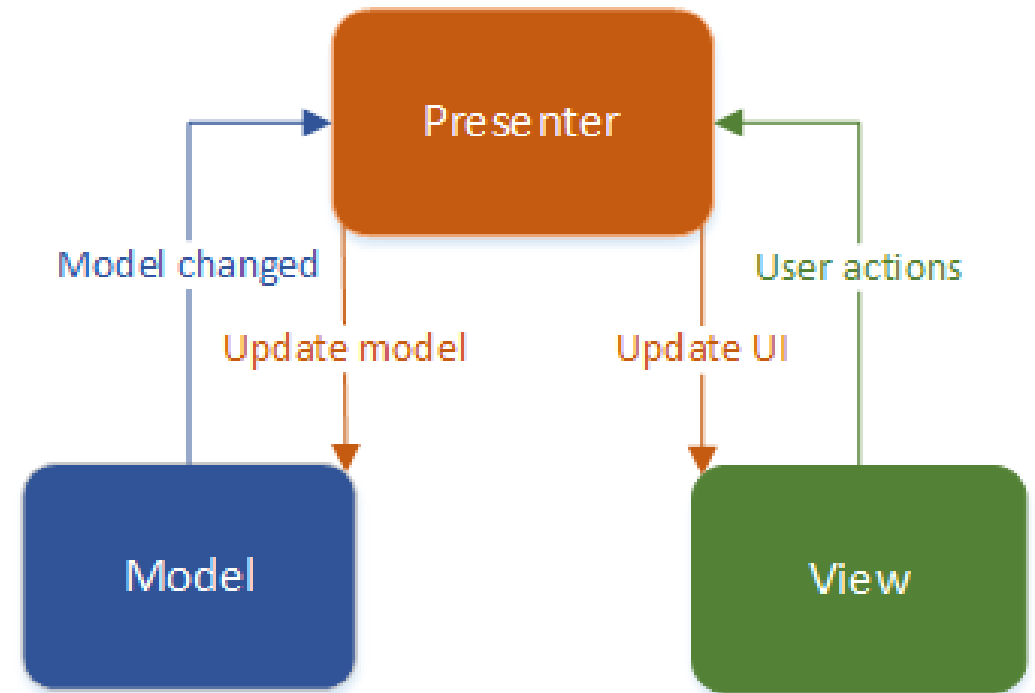
2.1. Mô hình thiết kế ứng dụng di động MVC

VS

MVC

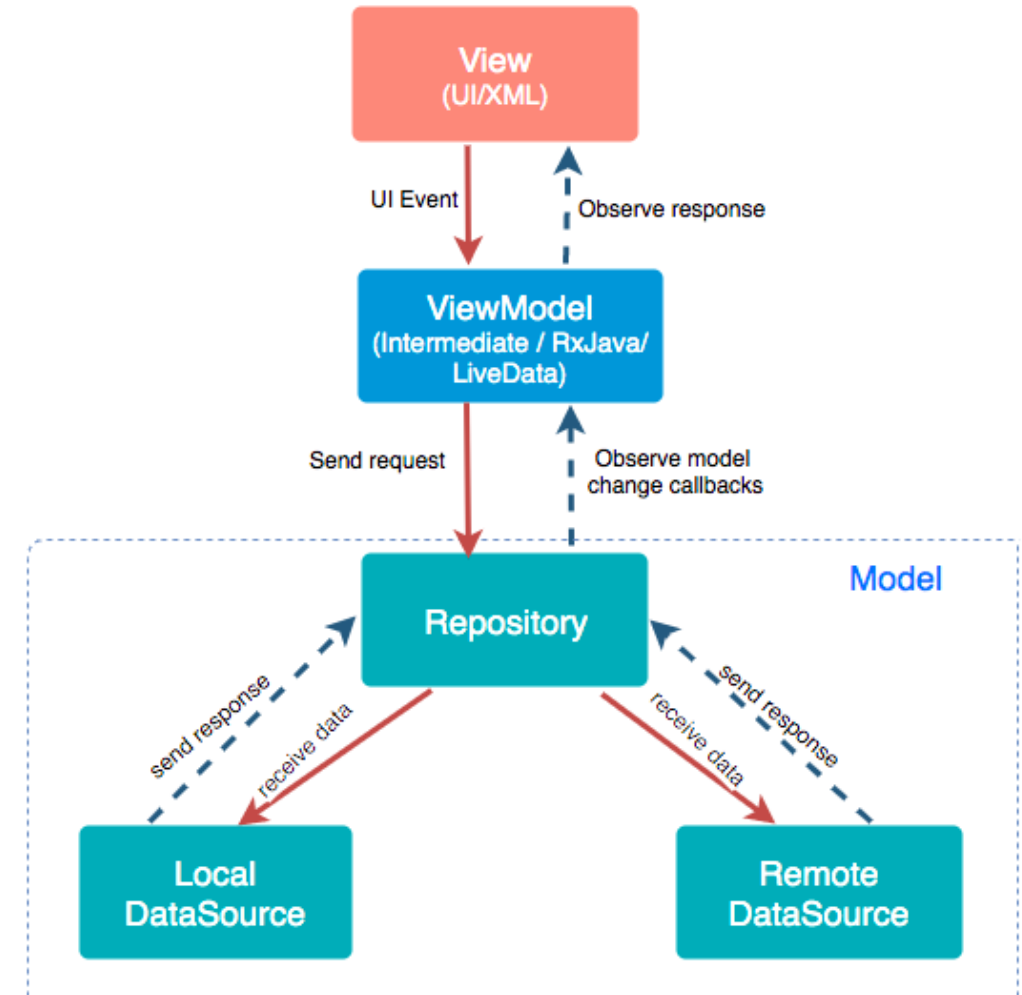
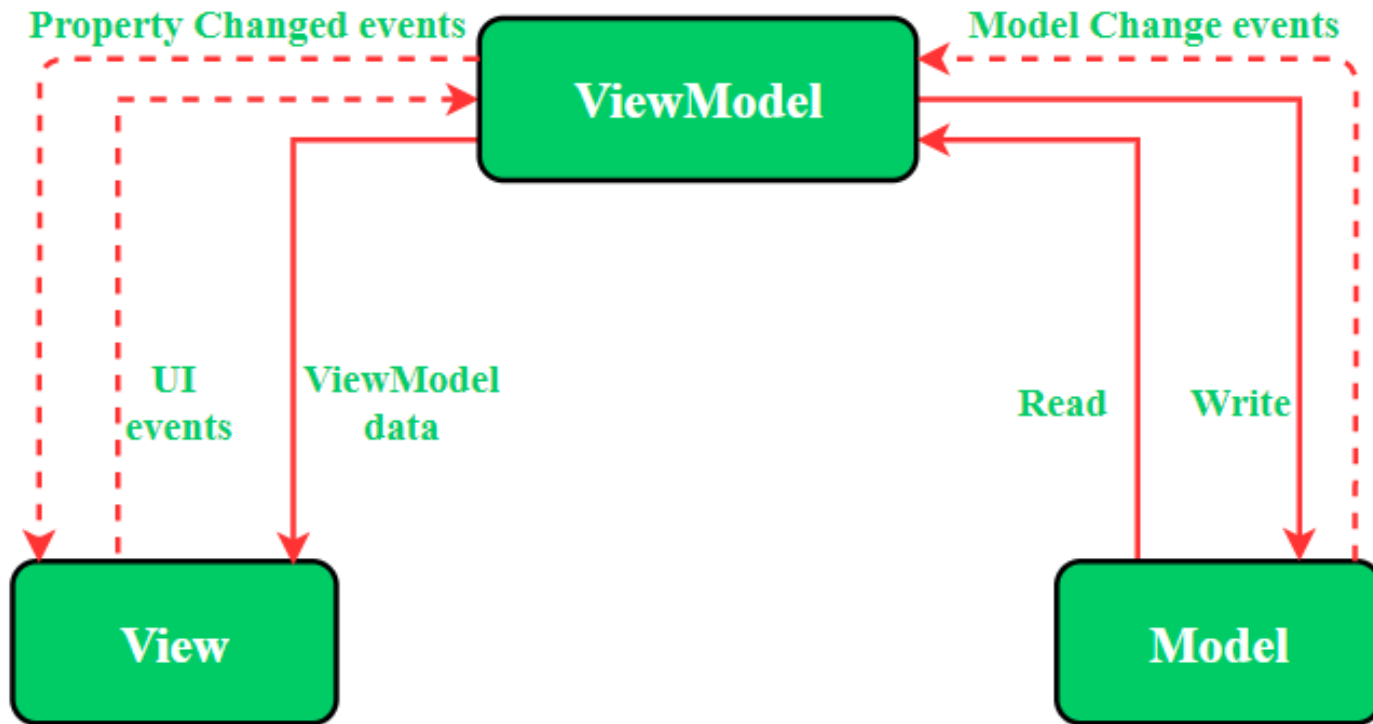


MVP



2.1. Mô hình thiết kế ứng dụng di động MVC

MVVM (Model View ViewModel) Architecture Pattern



2.1. Mô hình thiết kế ứng dụng di động MVC

Practices

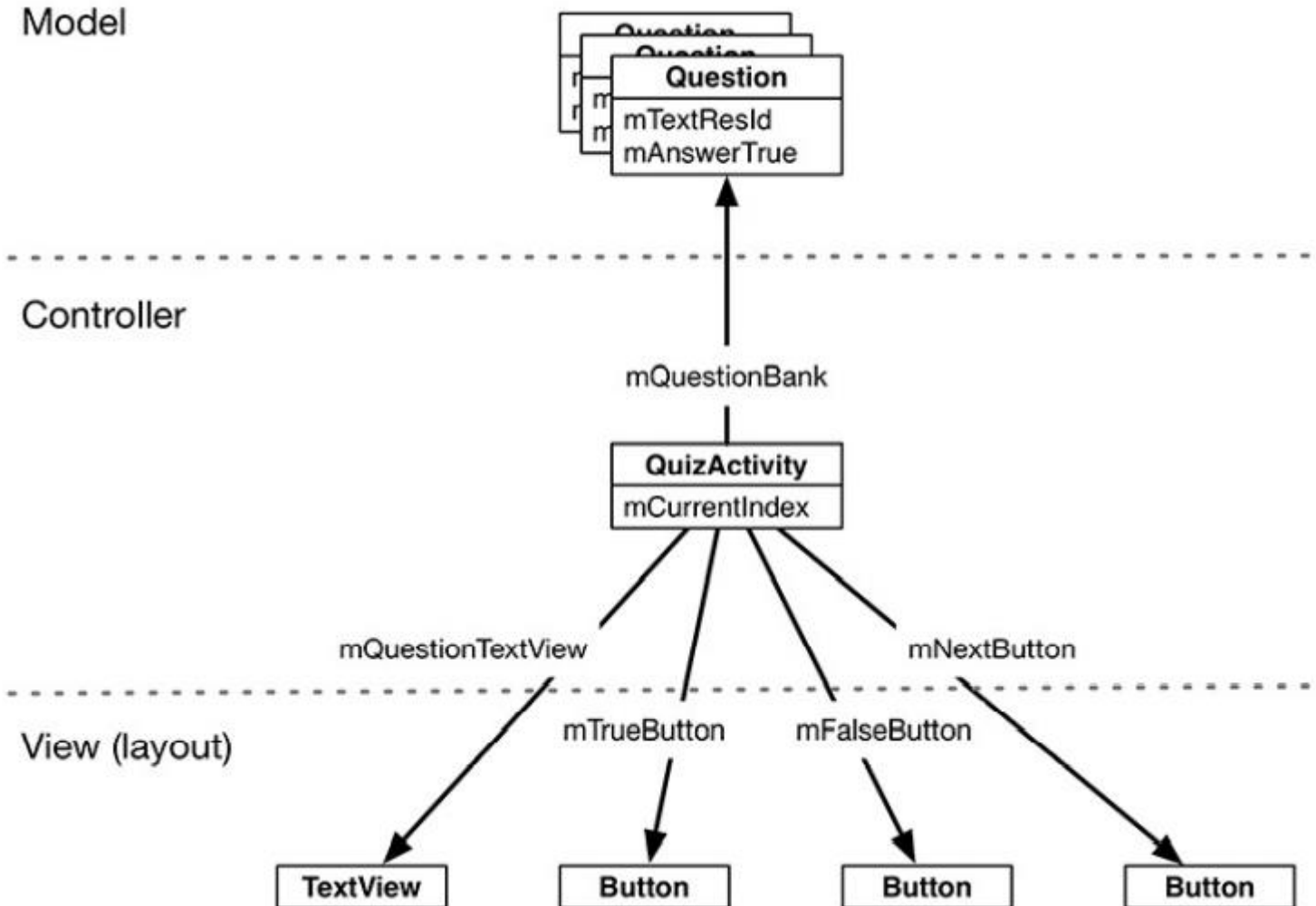
Upgrade GeoQuiz App to present more than one question!



2.1. Mô hình thiết kế ứng dụng di động MVC

Epx.

Object Diagram for GeoQuiz



2.2. Manifest trong Android

- The **Android Manifest** is an **XML file** which contains important **metadata** about the Android app.
- **Android Manifest** includes information about: (1) Package name; (2) Activity names; (3) Main Activity (the entry point to the app); (4) Android version support; (5) Hardware features support; (6) Permissions, and (7) Other configurations...
- Every app project must have an **AndroidManifest.xml** file (*with precisely that name*) at the root of the project source set.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapp"
    android:versionCode="1"
    android:versionName="1.0" >
    ...
</manifest>
```



2.2. Manifest trong Android

- **Android Manifest** includes information about: (1) Package name; (2) Activity names; (3) Main Activity (the entry point to the app); (4) Android version support; (5) Hardware features support; (6) Permissions, and (7) Other configurations...

```
<manifest ... >
    <application ... >
        <activity android:name="com.example.myapp.MainActivity" ... >
            </activity>
        </application>
    </manifest>
```

```
<manifest ... >
    <uses-permission android:name="android.permission.SEND_SMS" />
    ...
</manifest>
```

2.2. Manifest trong Android

- **Android Manifest** includes information about: (1) Package name; (2) Activity names; (3) Main Activity (the entry point to the app); (4) Android version support; (5) Hardware features support; (6) Permissions, and (7) Other configurations...

```
android {  
    defaultConfig {  
        applicationId 'com.example.myapp'  
  
        // Defines the minimum API level required to run the app.  
        minSdkVersion 15  
  
        // Specifies the API level used to test the app.  
        targetSdkVersion 28  
  
        ...  
    }  
}
```

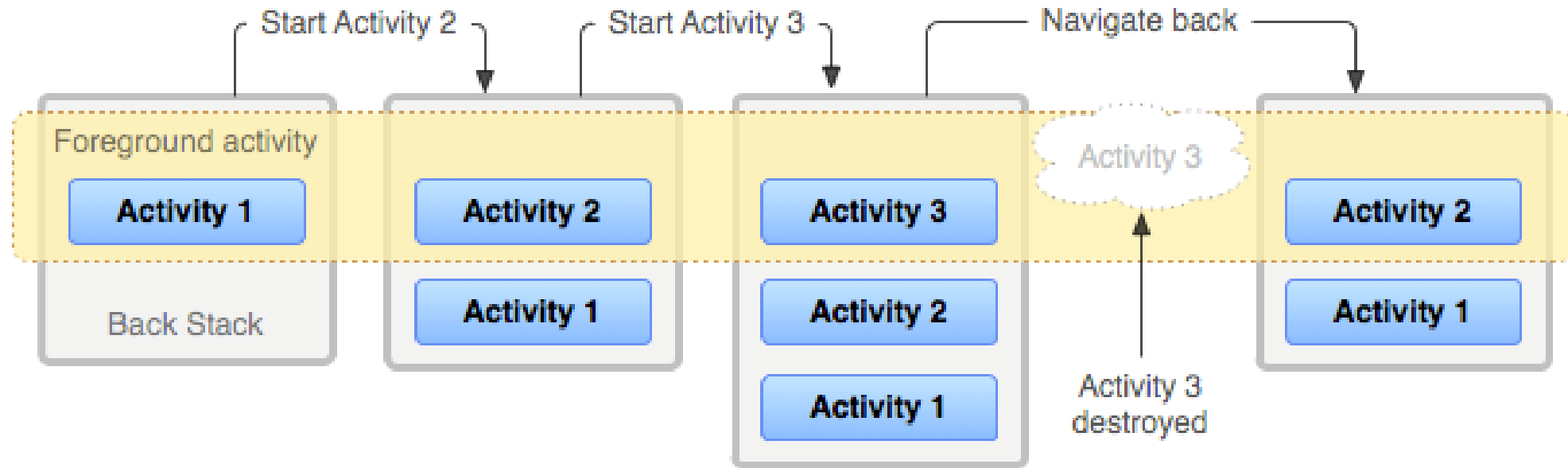
2.2. Manifest trong Android

- **Android Manifest** includes information about: (1) Package name; (2) Activity names; (3) Main Activity (the entry point to the app); (4) Android version support; (5) Hardware features support; (6) Permissions, and (7) Other configurations...

```
<manifest ... >
    <uses-feature android:name="android.hardware.sensor.compass"
        android:required="true" />
    ...
</manifest>
```

```
<intent-filter ... >
    <action android:name="android.intent.action.EDIT" />
    <action android:name="android.intent.action.INSERT" />
    <action android:name="android.intent.action.DELETE" />
    ...
</intent-filter>
```

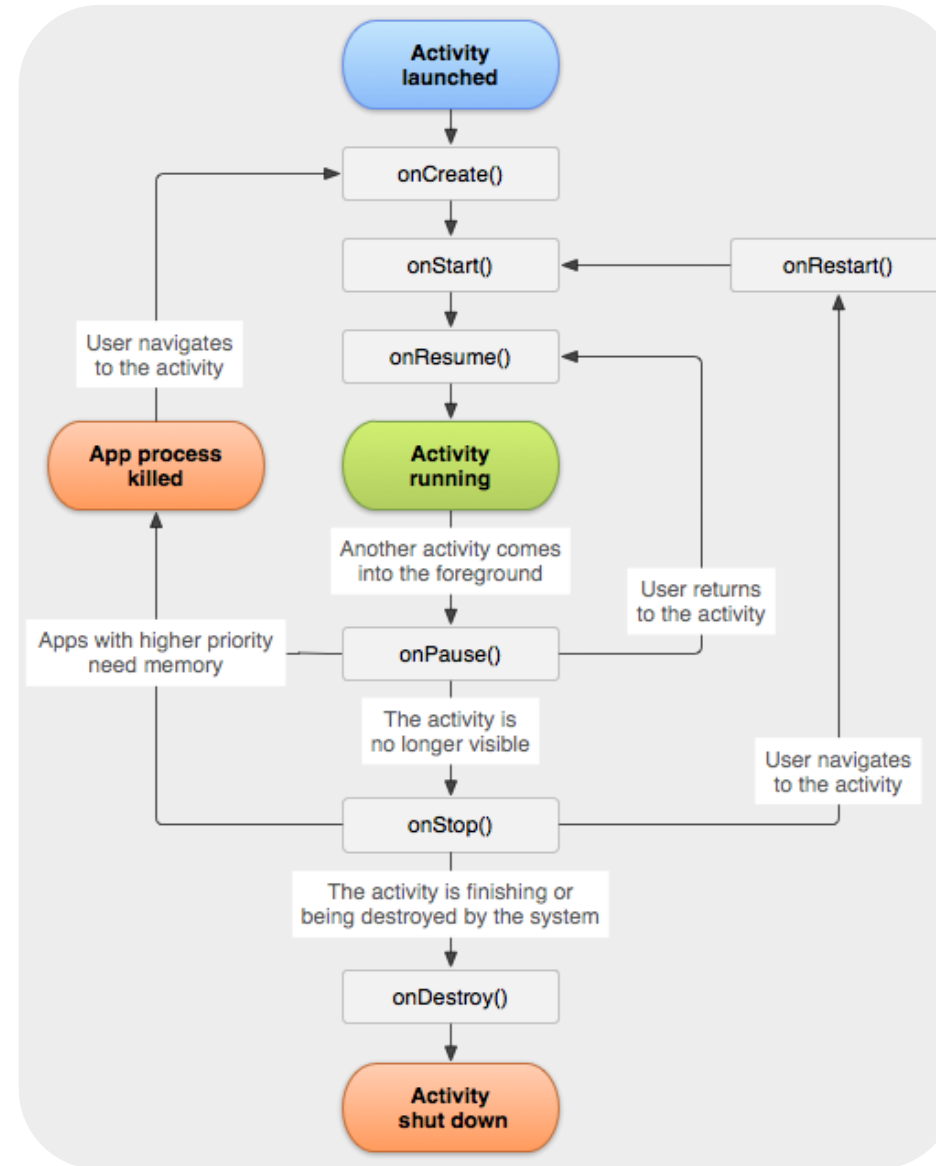
2.3. Vòng đời Activity (**Activity Lifecycle**)



- The **Activity** class is a crucial component of an Android app.
- Generally, one activity implements **one screen** in an app.
- Most apps contain **multiple screens**, which means they comprise **multiple activities**.
- **Main activity** is the **first screen** to appear when the user launches the app.
- Each activity can then **start another activity** in order to **perform different actions**.

2.3. Vòng đời Activity (**Activity Lifecycle**)

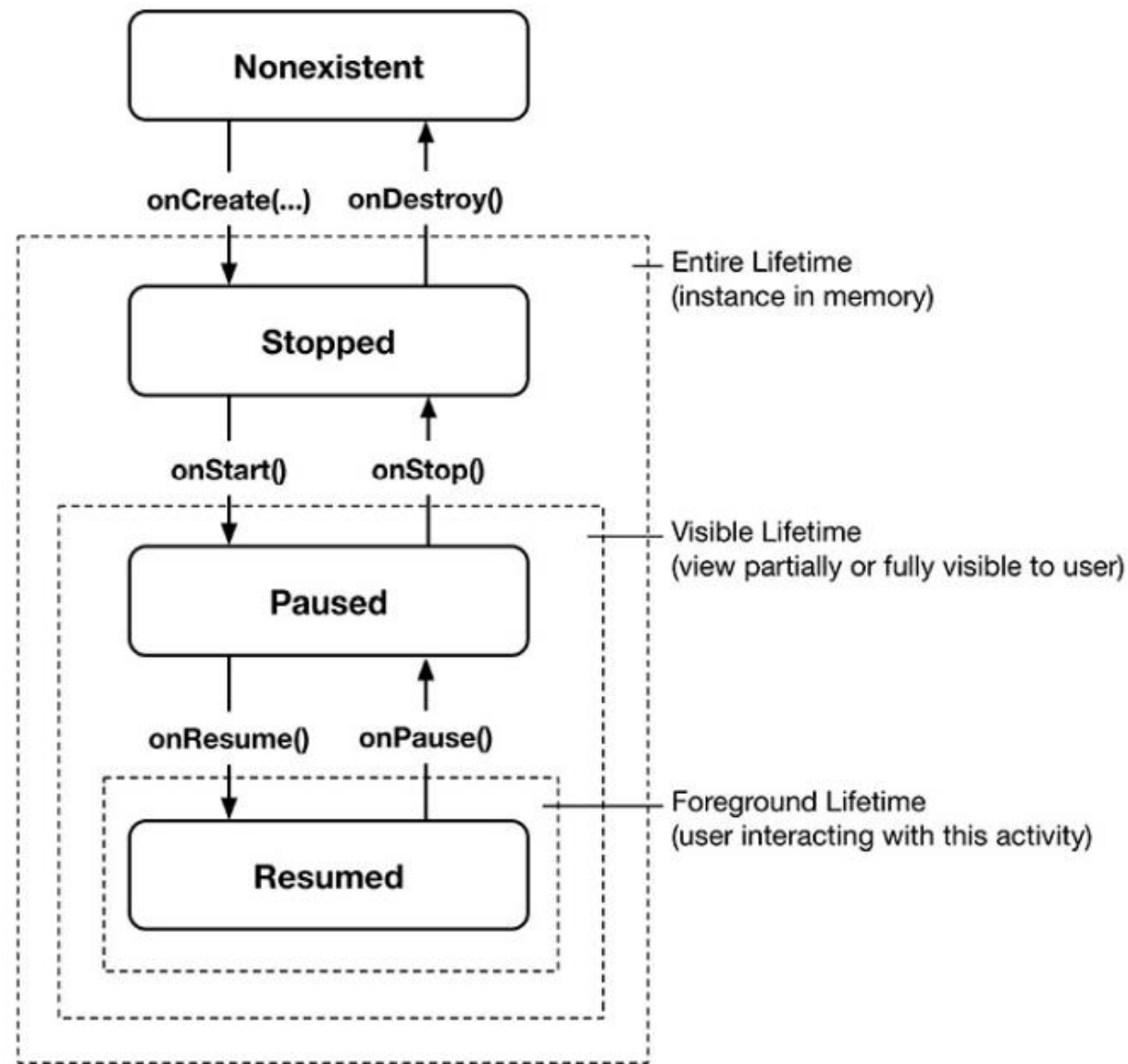
Over the course of its **lifetime**, an **activity** goes through a **number of states**.



A **series of callbacks** are used to **handle transitions** between **states**.

2.3. Vòng đời Activity (**Activity Lifecycle**)

- Every instance of **Activity** has a **lifecycle**
- During this **lifecycle**, an activity transitions between **four states**:
 - (1) Resumed
 - (2) Paused
 - (3) Stopped
 - (4) Nonexistent



2.3. Vòng đời Activity (**Activity Lifecycle**)

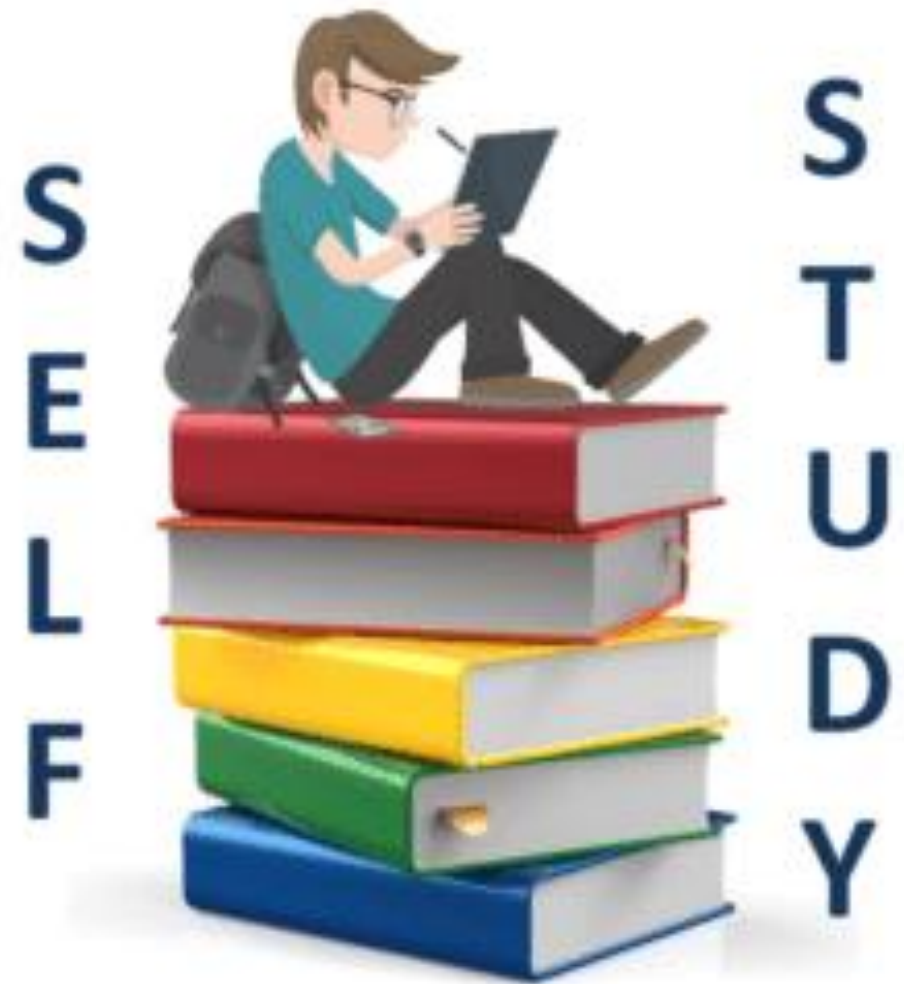
Practices

Upgrade GeoQuiz App to deal with “**rotation problem**” ?



Giới thiệu về phát triển ứng dụng di động

- Gỡ lỗi trong phát triển di động (**Mobile Development Debugging**)
- Các phiên bản SDK và sự tương thích (**SDK versions and their compatibility**)



A large, stylized circular graphic composed of many small dots, arranged in a spiral pattern, filling the left half of the slide.

HUST

THANK YOU !

Lập trình ứng dụng di động

Mobile Application Programming

ET4710

PGS. TS. Đỗ Trọng Tuấn
*Viện Điện tử Viễn thông * Đại học Bách Khoa Hà Nội*

ONE LOVE. ONE FUTURE.