Android Assignment [Research]

a. What is Android? Who created it? What are Android Apps?

Android is a mobile operating system based on a modified version of the Linux kernel and other open source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance, with the main contributor and commercial marketer being Google.

Android Inc. was founded in Palo Alto, California, in October 2003 by Andy Rubin, Rich Miner, Nick Sears, and Chris White. Rubin described the Android project as "tremendous potential in developing smarter mobile devices that are more aware of its owner's location and preferences". The early intentions of the company were to develop an advanced operating system for digital cameras, and this was the basis of its pitch to investors in April 2004. The company then decided that the market for cameras was not large enough for its goals, and by five months later it had diverted its efforts and was pitching Android as a handset operating system that would rival Symbian and Microsoft Windows Mobile.

An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS.

b. What is the software used in the development of Android Apps?

There are many softwares which are used in the development of Android Apps. Major among them are listed below:

1. Android Studio

Android Studio is the official IDE for Android and contains everything you need to build an Android app. Completely designed for Android, it accelerates your development process and helps you build quality applications for every device. It is also known as the official development environment for Google's Android operating system. It has countless features such as visual layout, APK analyzer, intelligent code editor, fast simulator, flexible build system, and a real-time analyzer.

2. Android SDK

The Android SDK is a development kit for Android. It includes a complete set of development and debugging tools.

The Android SDK is composed of modular packages that you can download separately using the Android SDK Manager. The current version is 26.1.1 and the toolkit includes:

SDK tool
SDK platform tools
SDK documentation
Android SDK resources
SDK example
Google API
Android support

3. B4A

Basic4android is a simple but powerful Android application development environment. Its language is similar to the Visual Basic language and supports more objects. Basic4android compiles code into Android native apps without the support of other runtime environments. This tool is developed solely for Android applications and does not support other platforms.

4. Fabric

Fabric is a tool that helps your mobile app development team build better apps, learn about your users in real time, and grow your business based on analytical data. From development to launch and beyond, Fabric provides a complete, real-time app performance and health status for everyone on your development team. It also gives developers data about user activity and where users are most likely to interact with the application.

c. Which are the languages commonly used in the development of android apps? Which language does InstiApp use?

Languages Used:

1. Java

Java is the official language for Android App Development and consequently, it is the most used language as well. Many of the apps in the Play Store are built with Java and it is also the most supported language by Google. In addition to all this, Java has a great online community for support in case of any problems.

2. Kotlin

Kotlin is a cross-platform programming language that may be used as an alternative to Java for Android App Development. It has also been introduced as a secondary ?official? Java language in 2017. Kotlin can inter-operate with Java and it runs on the Java Virtual Machine.

3. C++

C++ can be used for Android App Development using the Android Native Development Kit(NDK). However, an app cannot be created totally using C++ and the NDK is used to implement parts of the app in C++ native code. This helps in using C++ code libraries for the app as required.

4. C#

C# is quite similar to Java and so it is ideal for Android App Development. Like Java, C# also implements garbage collection so there are fewer chances of memory leaks. And C# also has a cleaner and simpler syntax than Java which makes coding with it comparatively easier.

5. Python

Python can be used for Android App Development even though Android doesn?t support native Python development. This can be done using various tools that convert the Python apps into Android Packages that can run on Android devices.

6. HTML, CSS, JavaScript

Android apps can be created using HTML, CSS, and JavaScript using the Adobe PhoneGap framework that is powered by Apache Cordova. The PhoneGap framework basically allows the usage of web development skills to create hybrid apps that are shown through "WebView" but are packaged like an app.

InstiApp is a WnCC initiative to create a unified platform for everything that happens inside IIT Bombay.

The project has three sections:

1. The API Backend of the app, which is a RESTful API in Django **(Language Used - Python).**

2. A Progressive Web Application (PWA), written in Angular using **HTML, CSS, Javascript, TypeScript.** It involves a lot of async programming, cache handling, performance optimization etc.

3. A native Android App : **Languages used - Java and XML.**

d. What is the activity cycle of a basic Android application?

Generally, the activities in a general android application will go through different stages in their life cycle. In android, Activity class have 7 callback methods like **onCreate(), onStart(), onPause(), onRestart(), onResume(), onStop() and onDestroy()** to describe how the activity will behave at different stages.

In an android application, activities are generally implemented as a **subclass of Activity class** like as shown below.

public class MainActivity extends Activity { }

In android, an activity goes through a series of states during its lifetime. By using callback methods we can get the activity transitions between the states.

Android system initiates its program within an Activity starting with a call on **onCreate() callback method**. There is a sequence of callback methods that start up an activity and a sequence of callback methods that tear down an activity.

Whenever the user trying to leave an activity like switching from one app to another app, the system will use callback methods to dismantle the activity completely or partially to resume the activity from where the user left off.

Based on our requirements we can implement the activity in the android app using the callback method and it?s not necessary to use all callback methods in each android application.

Following is the pictorial representation of the Android Activity Life cycle which shows how Activity will behave in different stages using callback methods:

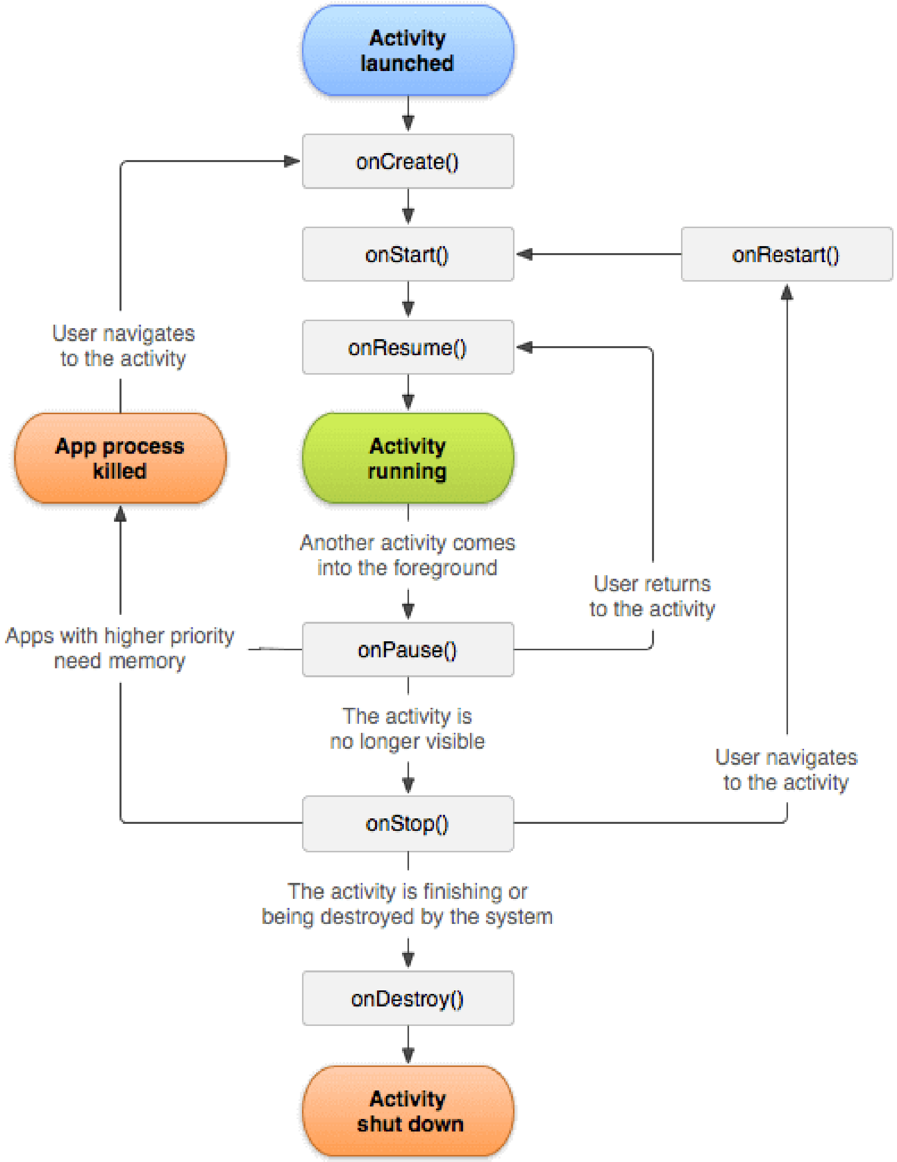
Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

Figure 1: Android Activity Cycle

e. What are five different UI elements in an android app?

We have a different type of UI controls available in android to implement the user interface for our android applications.

Following are the commonly used UI or input controls in android applications:

- TextView
- EditText
- AutoCompleteTextView
- Button
- ImageButton
- ToggleButton
- CheckBox
- RadioButton
- RadioGroup
- ProgressBar
- Spinner
- TimePicker
- DatePicker
- SeekBar
- AlertDialog
- Switch
- RatingBar

1. Android TextView

In android, TextView is a user interface control that is used to display the text to the user.

2. Android EditText

In android, EditText is a user interface control which is used to allow the user to enter or modify the text.

3. Android AutoCompleteTextView

In android, AutoCompleteTextView is an editable text view which is used to show the list of suggestions based on the user typing text. The list of suggestions will be shown as a dropdown menu from which the user can choose an item to replace the content of the textbox.

4. Android Button

In android, Button is a user interface control that is used to perform an action when the user clicks or tap on it.

5. Android Image Button

In android, Image Button is a user interface control that is used to display a button with an image to perform an action when the user clicks or tap on it.

Generally, the Image button in android looks similar as regular Button and perform the actions same as regular button but only difference is for image button we will add an image instead of text.

6. Android Toggle Button

In android, Toggle Button is a user interface control that is used to display ON (Checked) or OFF (Unchecked) states as a button with a light indicator.

7. Android CheckBox

In android, Checkbox is a two-states button that can be either checked or unchecked.

8. Android Radio Button

In android, Radio Button is a two-states button that can be either checked or unchecked and it cannot be unchecked once it is checked.

9. Android Radio Group

In android, Radio Group is used to group one or more radio buttons into separate groups based on our requirements.

In case if we group radio buttons using the radio group, at a time only one item can be selected from the group of radio buttons.

10. Android Progressbar

In android, ProgressBar is a user interface control which is used to indicate the progress of an operation.

11. Android Spinner

In android, Spinner is a drop-down list which allows a user to select one value from the list.

12. Android Timepicker

In android, TimePicker is a widget for selecting the time of day, either in 24-hour or AM/PM mode.

13. Android DataPicker

In android, DatePicker is a widget for selecting a date.

f. What are some of the salient features of those languages(part c)? How similar are they to C++?

1. Java

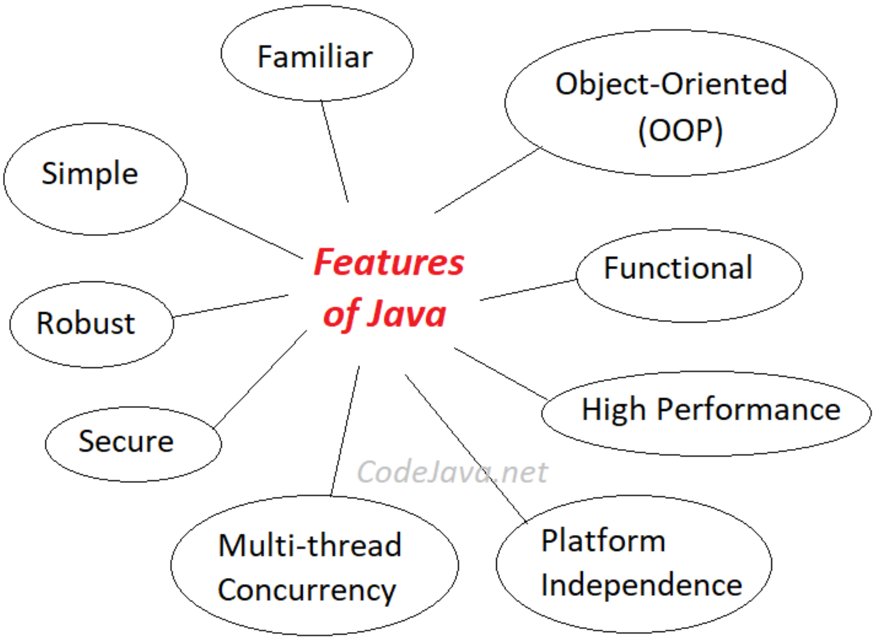

Figure 2: Salient Features of Java

Similarities between JAVA and C++ :

- **Both C++ and Java supports Object Oriented Programming:** OOPs is a modular approach, which allows the data to be applied within stipulated program area, It also provides the re-usability feature to develop productive logic, which means to give more emphasis on data. It support classes and objects. OOPs features include:

  Inheritance: process by which objects of one class can link and share some common properties of objects from another class.

Polymorphism: Allows us to perform a single action in different ways. It is the process of using a function for more than one purpose.

Abstraction: It is the act of representing essential features without including the background details.

Encapsulation: Wrapping up of data and functions into a single unit.

- **They have similar syntax:**

  Comments Syntax are identical:
  Both the single and multiple line comments are written as //?. and /* ?. */ respectively.

  The loops (like while, for etc.) and conditional statements (like if-else, switch etc.) are similar.

  Both have same arithmetic and relational operators:
  Arithmetic operators such as +, -, *, / Relational operators such as ¿, ¡, =, != (not equal to)The loops (like while, for etc.) and conditional statements (like if-else, switch etc.) are similar:

- **Execution of both the C++ and Java programs starts from the main function:**
  It is the entry point of the execution of the program. However, the function declaration is different, but the name is same.

- **They have same primitive data types:**
  These include datatypes like int, float, char, double etc. with some differences like the Boolean data type is called boolean in Java but it is called bool in C++. Many of their keywords are same:
  Example: break, continue, char, double, new, public, private, return, static etc.

- **Both have multi threading support:**
  Both allow executing multiple threads(sub-processes) simultaneously to achieve multitasking.

2. Kotlin

Kotlin provides you some amazing and powerful features over Java. That?s why developers love Kotlin. There are many tech giants who use Kotlin before Google announces the official support for it.
Following are main features of Kotlin:

- Concise code
- Null safety
- Expressive code
- Modern features

- Interoperability with Java
- JavaScript transpilation

1. Concise Code:

Developers always love clean and concise code. Less code takes less time to write, less time to read, and also makes easy to maintain it.

Kotlin has really concise code compare to java. It drastically reduces the amount of boilerplate code. So compare to java you will write less amount of code in Kotlin, which increases the speed of your development. Rough estimates indicate that Kotlin requires 40 percent less line of codes compare to java.

2. Null Safety:

More than 70 percent android app crashes due to null pointer exception of java. Java does not provide any ways of protection against the null. The developer has to handle it on his/her own.

But Kotlin handles nulls really nicely. In Kotlin you cannot get null pointer exception implicitly. Kotlin compiler by default doesn?t allow any types to have a value of null at compile time. So if you use Kotlin then it will reduce your app crashes and gives better user experience to your user.

3. Expressive code:

Expressive code means the code that can be easily understandable by every programmer.

Kotlin is really expressive. All data types, built-in functions, keywords, etc. are really very expressive and easily understandable. Kotlin is designed in such a way so that even a person who never codes in kotlin can easily understand its code.

4. Modern features:

As kotlin is a fourth generation language, it introduces and combined a lot of new features like Lambda functions, smart casts, null safety, operator overloading, etc. Which will help you in increasing your productivity? Kotlin is influenced by six major languages and it inherits a lot of best features of them.

5. Interoperability with java:

Kotlin is 100 percent interoperable with java. You can easily call Kotlin code from Java and Java code from Kotlin. This makes adoption much easier and lower-risk. There is also an automated Java-to-Kotlin converter built into the IDE that simplifies migration of existing code. This makes easy to work on an existing Android project. You can keep your old java file as it is and create new features in the new kotlin file.

6. JavaScript transpilation:

You can transcompile your kotlin code to JavaScript code. That way even your kotlin code can run on browsers. When you choose the JavaScript target, any Kotlin code that is part of the project as well as the standard library that ships with Kotlin is transpiled to JavaScript. However, this excludes the JDK and any JVM or Java framework or library used. Any file that is not Kotlin will be ignored during compilation.

Android Assignment [THINK]

There are number of UI controls provided by Android that allow us to build the graphical user interface for our app.

1. TextView: This control is used to display text to the user.

2. EditText: EditText is a predefined subclass of TextView that includes rich editing capabilities.

3. AutoCompleteTextView: The AutoCompleteTextView is a view that is similar to EditText, except that it shows a list of completion suggestions automatically while the user is typing.

4. Button A push-button that can be pressed, or clicked, by the user to perform an action.

5. ImageButton: An ImageButton is an AbsoluteLayout which enables you to specify the exact location of its children. This shows a button with an image (instead of text) that can be pressed or clicked by the user.

6. CheckBox: An on/off switch that can be toggled by the user. You should use check box when presenting users with a group of selectable options that are not mutually exclusive.

7. ToggleButton: An on/off button with a light indicator.

8. RadioButton: The RadioButton has two states: either checked or unchecked.

9. RadioGroup: A RadioGroup is used to group together one or more RadioButtons.

10. ProgressBar: The ProgressBar view provides visual feedback about some ongoing tasks, such as when you are performing a task in the background.

11. Spinner: A drop-down list that allows users to select one value from a set.

12. TimePicker: The TimePicker view enables users to select a time of the day, in either 24-hour mode or AM/PM mode.

13. DatePicker: The DatePicker view enables users to select a date of the day.

A schematic diagram of the screen:

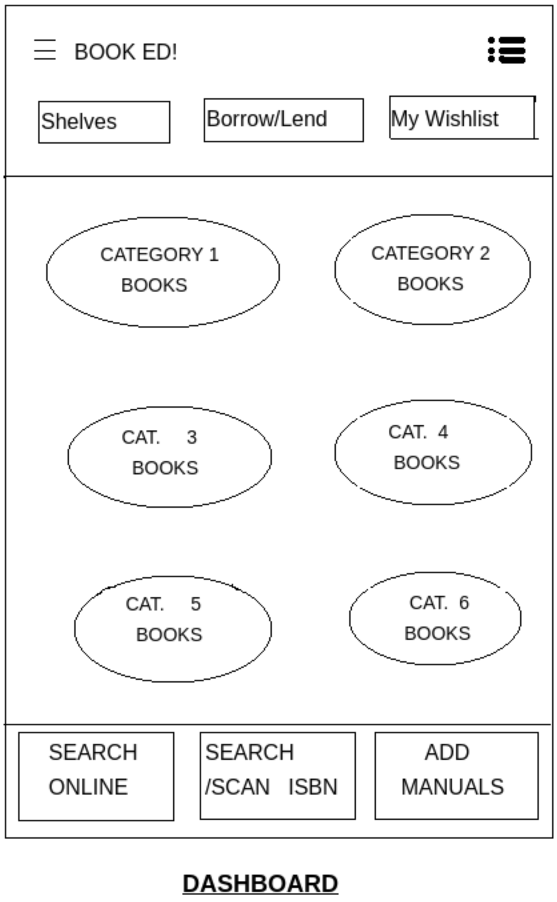

Figure 3: A Schematic Diagram Of The Screen.

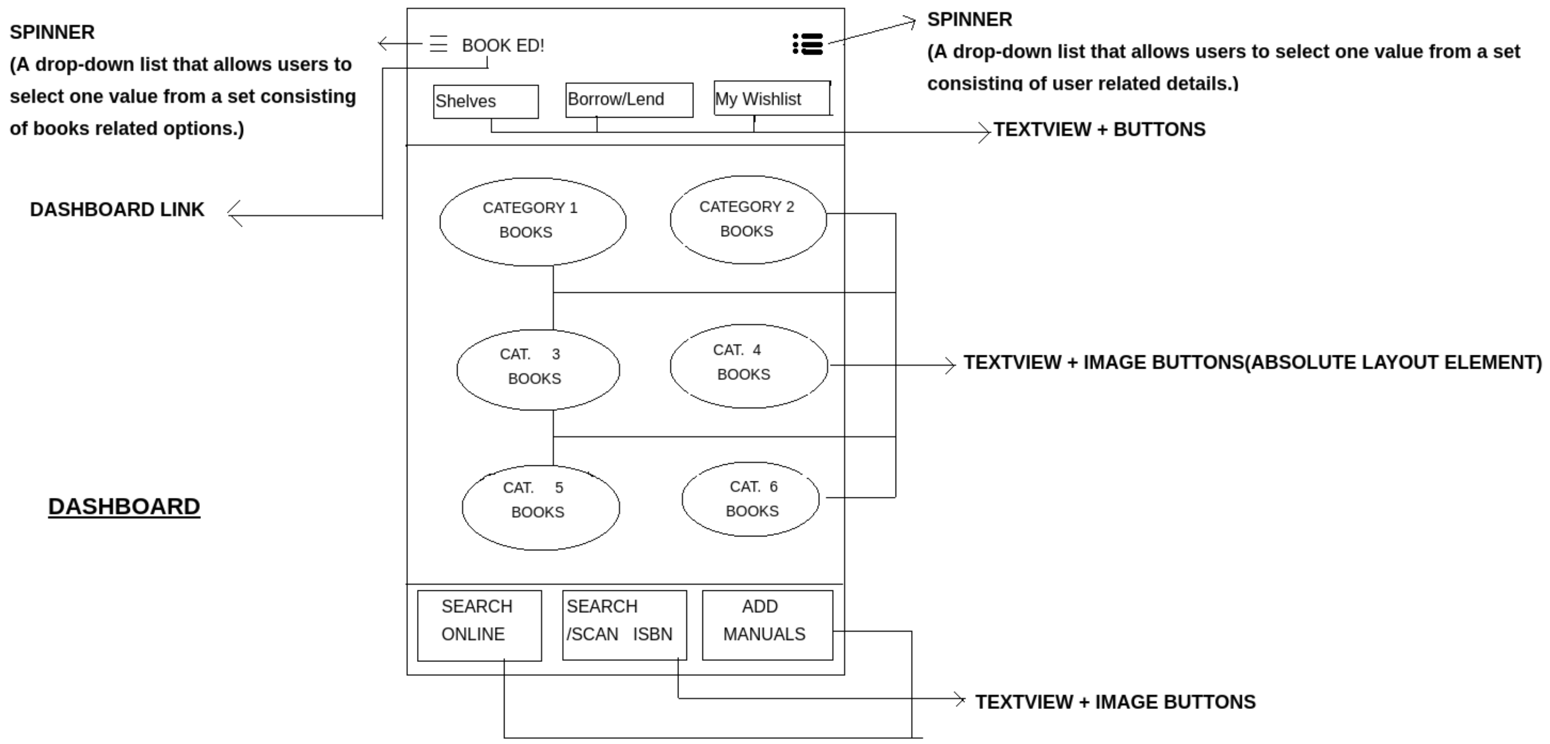
SPINNER
(A drop-down list that allows users to select one value from a set consisting of books related options.)
DASHBOARD LINK
DASHBOARD
BOOK ED!
Shelves
Borrow/Lend
My Wishlist
SPINNER
(A drop-down list that allows users to select one value from a set consisting of user related details.)
TEXTVIEW + BUTTONS
CATEGORY 1 BOOKS
CATEGORY 2 BOOKS
CAT. 3 BOOKS
CAT. 4 BOOKS
CAT. 5 BOOKS
CAT. 6 BOOKS
TEXTVIEW + IMAGE BUTTONS(ABSOLUTE LAYOUT ELEMENT)
SEARCH ONLINE
SEARCH /SCAN ISBN
ADD MANUALS
TEXTVIEW + IMAGE BUTTONS

Figure 4: Various Elements In My Design.