

BATCH :- SB-S

## Assignment-1

1. Based on your Understanding, identify a recent business trend that has influenced the android platform. Explain how this trend impacts Android app developers and business in the mobile industry.

→ One recent business trend that has influenced the android platform is the rise of subscription-based services. This trend is being driven by number of factors, including the increasing popularity of streaming services like netflix and disney+, the growing demand for mobile-first content and the convenience of recurring payments.

Here some specific ways in which the Subscription-based trend is impacting Android App developers and businesses:-

→ Increased competition:- As more and more businesses offer subscription-based services, developers and businesses are facing increased competition for users attention and spending. This means that they need to develop innovative and engaging apps that offer great value to users.

→ New Opportunities:- Subscription-based services also offer new opportunities for developers and businesses to monetize their apps. For examples, developers can offer subscription-based access to exclusive



Content or features. or businesses can use subscription-based services to provide their customers with ongoing support services.

New challenges:- However, Subscription-based services also create new challenges for developers and businesses for example. they need to develop and maintain recurring billing systems, and they need to provide high level of customer support to ensure that users are satisfied with their service.

2. What is purpose of an inflater or layout in Android development, and how does it fit into the architecture of android layouts?

→ The purpose of an inflater is of the Android layout in Android development is to create a view hierarchy from an XML layout file. The view hierarchy is a tree of objects that represent the UI of an Android app.

The Inflater is used in a variety of places in an Android app. For an example, it is used to create the root view of an activity or fragment, and to inflate layout files for dialogs and menus.

// Get the Inflater service.

```
Layout Inflater inflater = (LayoutInflater) getSystemService(Context.LAYOUT_INFLATER_SERVICE);
```

// Inflate the layout file

```
View rootView = inflater.inflate(R.layout.main, Activity.this);
```

// Set the root view of the Activity  
Set ContentView (rootView)



3. Explain the concept of a custom DialogBox in Android applications. provide example to illustrate its use.

→ In android applications, a custom DialogBox is a pop-up window that overlays the current activity and is often used to interact with the user, gather input or display information.

Overview of custom DialogBox:

Purpose:- Custom-dialogs are used when you want to present information, receive user input or perform actions within a self-contained, isolated UI element that temporarily interrupts.

Components:- A Custom dialog typically consists of various UI elements like buttons, textviews, images or input fields, tailored to the specific interaction you want to facilitate.

Customization:- Developers can design the dialog's appearance, layout, and behavior according to their app's branding or specific requirements. This customization allows for creativity in design and functionality.

Simple example of creating and using a Custom dialog in Android.



Fun Custom Dialog Q2

val customDialog = Dialog(this)

customDialog setContentView(R.layout.custom\_dialog)

val messageTextView = customDialog.findViewById<TextView>(R.id.message)

val okButton = customDialog.findViewById<Button>(R.id.ok)

messageTextView.text = "This is a custom dialog"

okButton.setOnClickListener {

customDialog.dismiss()

customDialog.show()

use case of custom dialog Box: login, confirmation  
Dialog, Setting, Informational pop-up  
media playback controls

4. How do activities, services and the Android manifest work together to make an android app? Can you describe the main roles and provide a basic example of how they cooperate to design a mobile app?

### 1. Activities:-

• Role:- Activities represent individual screens or UI components in an Android app. They manage the user interface and user interactions.

### 2. Services:-

• Role:- Services are background components that perform long running operations or handle tasks that don't have a user interface. They can run even if the app UI is not visible.

### 3. Android manifest file:-

Rule:- The android manifest.xml file is like the app blueprint. It declares the app components and defines how they interact with the android system and other components.



Ex In android manifest.xml, you specify which activities are part of your app, their launch modes, permissions and services declarations. This file acts as a blueprint for android system to understand your app's structure and behavior.

```
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        startServiceButton.setOnClickListener {
            val serviceIntent = Intent(this, NotificationService::class.java)
            startService(serviceIntent)
        }
    }
}
```

```
class NotificationService : IntentService("NotificationService") {
    override fun onHandleIntent(intent: Intent?) {
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
            val name = "my channel"
            val notificationManager = getSystemService(NotificationManager::class.java)
            notificationManager.createNotificationChannel(name)
            val builder = NotificationCompat.Builder(this, name)
            builder.setSmallIcon(R.drawable.ic_launcher_foreground)
            builder.setContentText("This is notification from Service")
        }
    }
}
```



5. How does the Android manifest file impact the development of an android application? Provide an example to demonstrate its significance.

The android manifest file is a crucial component in the development of an android application. It serves several important purposes, and its content significantly impacts the android system interacts with and manages your App.

Significance of the Android manifest file:

- App Configuration
- Component Declaration - permissions
- Intent filters
- App lifecycle.

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myApp">
    <Application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:supportRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name="mainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN">
                <category android:name="android.intent.category.launcher"/>
            </intent-filter>
        </activity>
        <activity android:name="secondActivity">
            -- Declare additional activities here
        </activity>
    </Application>
    <user-permission android:name="android.permission"
```



--- Declare require permissions here ---

<application>

<manifest>

What is the role of resource in android development? Discuss the various types of resources and their significance in creating well-structured application. provide example to clarify your points

Resources play a fundamental role in Android development by providing a structured way to manage assets, values, layouts and other elements used in your app.

## 1. Layout Resources.

- type xml files in the 'res/layout' directory

Significance: Define the structure and appearance of the app's user interface

Ex

<Button

android:id="@id/myButton"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Click me"/>

## 2. Drawable Resources.

type: Image and drawables assets in the 'res/drawable' directory.

significance: Store graphics, icons and images used in your app

Ex: ic\_launcher.png is the app's launcher icon



### 3. String resource:-

Type: String defined in xml files under 'res/values'.  
Significance: Store text string, making it easier to provide translations and maintain consistency.

ex res/values/string.xml

```
<string name="app_name">my app</string>  
<string name="welcome_message">welcome to  
my app</string>
```

### 4. Color resource:-

Type: Color defined in xml files under 'res/values'.  
Significance: Store color values, ensuring consistency in the app design.

ex res/values/colors.xml defines color resources.

```
<color name="primary_color">#007ACC</color>  
<color name="accent_color">#FFA500</color>
```

### 7. How does an Android Service contribute to the functionality of the mobile application? Describe the process of developing an Android Service.

→ Contributions of android services:

1. Background Processing: Services allow apps to perform tasks in the background without blocking the user interface.

2. Long-running operations: Services are ideal for handling operations that require more time to complete, such as playing music.

3. Inter-Component Communication: Services enable components like activities, broadcast receivers and other services to communicate with each other efficiently.

4. Foreground Services: Android services can run in the foreground, even when the app isn't in the foreground. This is useful for features that require ongoing user interaction.



1. Define the services class:- create a new java or other kotlin class that extends the 'service' class. overrides like onCreate(), onDestroy(), onStartCommand() to define the behavior of your service.
2. Configure service in manifest. Declare your service in `android:manifest` file to inform the android system about its existence and configuration. `<service android:name="myService">`