



Q) Based on your understanding, identify a recent business trend that has influenced the Android platform. Explain how this trend impacts android app developer and business in mobile app industry.

→ One recent business trend that has influenced the android platform is the rise of AI powered apps. These apps are improving the user experience.

- Personalization : AI can be used to personalize the user experience by recommending content, features and products which are relevant to user interest.

Automation :- AI can be used to automate the tasks, such as scheduling appointments, managing finances.

→ Here are some specific ways in which the trend of AI powered mobile Apps is impacting Android app developers and business in mobile app industry

→ For Android Developers:-

- AI Powered mobile apps offer a new way to create innovative and engaging apps. For ex AI can be used to create apps that can understand and respond to natural language, generate realistic images and videos.

- Developers who want to create AI-powered apps need to have expertise in AI and machine learning. This can be challenging for those who do not have expertise.
 - There is a growing number of AI-powered mobile apps available on the Android platform.
- ⇒ For business in the mobile app industry:
- AI-powered apps offer business a new way to reach and engage with customers. Business can use it to provide customers with personalized recommendations, automate customer support tasks, and provide real-time assistance.
 - Business that want to develop AI-powered apps need to invest in resources such as computing power and AI expertise. This can be expensive and time-consuming investment.
 - There is a growing number of businesses that are developing AI-powered mobile apps. This means that business needs to find ways to differentiate their apps from competitors.

What is the purpose of an inflator layout in android app development, and how does it fit into the Architecture of Android layout.

- The purpose of an inflator of layout in android development is to create a new view object from an xml file that defines a layout. It is a class that can take an xml file as input and build the view object from it.
- It fits into the architecture of android layout or create custom ones without writing java code for each components.
- An inflator or layout can be used in different scenario, such as:
- When creating a custom view class that extends an existing view class, such as
- When creating a custom view class that extends an existing view class, such as TextView or ImageView and inflating a layout xml file in constructor.
- When creating a fragment class that displays a part of the user interface, and inflating a layout xml file in onCreateView method.

→ An Inflator layout can be obtained from sources such as:-

- The getSystemService method of the Context class, which returns an instance of LayoutInflater that can be used to inflate any resources. ~~within the scope of method.~~
- The getMenuInflater method of the Activity class, which returns ~~an~~ instances of LayoutInflater that can be used to inflate any resources within the scope of method.
- An inflator of layout can also be customized by adding a LayoutInflator.Factory or LayoutInflator.Factory2 Interface, which allow developers to create their own View or modify existing ones during Inflation process.

3] Explain the concept of custom dialog in Android applications. Provide example to illustrate its use.

- A Custom DialogBox is type of dialog that allows you to create a custom layout or appearance for your dialog. You can use a Custom DialogBox to display any kind of content that you want such as Images, text, button, or other views.

→ It is useful when you want to provide option or info to the user that a standard dialog can offer.

Custom-dialog.xml

```
<?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">
```

< TextView

```
    android:id="@+id/label"
```

```
    android:layout_width="match-parent"
```

```
    android:layout_height="wrap-content"/>
```

< Button

```
    android:id="@+id/dialog"
```

```
    android:layout_width="match-parent"
```

```
    android:layout_height="wrap-content"
```

```
    android:text="OK"/>
```

</LinearLayout>

CustomDialog.kt.

```
import android.app.Dialog.
```

```
import android.content.Context
```

```
import android.os.Bundle
```

```
import android.view.View
```

```
import android.widget.EditText
```

Class Custom Dialog Context:
{
override fun onCreate(savedInstanceState: Bundle?)
{
super.onCreate(savedInstanceState)
setContentView(R.layout.dialog)
setTitle("Custom Dialog Title")
val text = findViewById<EditText>(R.id.dialog_text).text
dismiss()
}}
mainActivity.kt

```
import android.os.Bundle  
import android.view.View  
import android.widget.Button  
import androidx.appcompat.app.AppCompatActivity
```

```
Class MainActivity : AppCompatActivity {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)

        val button1: Button = findViewById(R.id.button)
        button1.setOnClickListener {
            val customDialog = CustomDialog()
            customDialog.show()
        }
    }
}
```

How do activities, Services, and android manifest file work together to make an android app? Can you describe their main roles and provides basic Example of how they cooperate to design a mobile app?

Activities, Services and android manifest file are essential components of Android app. They work together to define app's structure, behaviour and requirements.

① Activities :-

Activities represent the user interface and the screen of your android app.

Activities handle user interaction such as button clicks, input forms and displaying information.

② Services :

- Services are background components that perform tasks without a user interface.

- They are used for long-running operation such as playing music, downloading files etc.

- Service can continue running even when app is not in the foreground

③ Android Manifest file:-

- it is configuration file that provides essential information about your app to the android.
- it defines components of your app including activities and services with their properties and permissions.
- it also declares app's entry point and required permissions.

Ex let's say you are creating music player.

① Activities:-

- You would have activities for various screens, like the main screen, showing music library, a screen for playing music and so on.
- Each activity handles the user interaction and displays the corresponding screen.

② Services:-

- You'd use a service to play music in the background.

→ a

- When the user selects a song and "Play", the main activity can start a playing service.

③ Android manifest file:-

- In the manifest file, you declare all your activities and services, specifying their properties.
- You declare the main activity, which is the entry point of your app.

Ex:-

```
<manifest xmlns:android="http://schemas.  
          android.com/apk/res/android"  
    package=".com.example.musicplayer">  
  
    <application>  
        <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=".musicplayActivity"/>  
        <service android:name=".MusicplayerService"/>  
  
        <!-- Permission -->  
  
        <uses-permission android:name="android.  
                           permission.READ_EXTERNAL_STORAGE">  
        <uses-permission android:name="android.permission.  
                           INTERNET"/>  
  
    </application>  
    </manifest>
```

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

→ the android manifest file plays a crucial role in the development of an android application as it serves as a blue print of the Android system to understand and manage your application.

① Component declaration:-

- the manifest file declares all the components of your app, including activities, services, broadcast receivers and content providers.

② Entry point:-

- it specifies the main activity the Android System should launch when app is opened. this is starting point of app.

③ Permissions :-

- you declare required permission in manifest file. this is crucial for security and access to device resources.

④ Intent filters :-

- you define intent filters for services and broadcast receivers. these filters specify how your component respond to intents.

⑤ Application metadata :-

these help user to identify your app.

⑥ Hardware and software Features:-

- You can specify the hardware and software features your app requires ensuring its run on compatible devices.

```
<manifest xmlns:android = "http://schemas.
    android.com/apk/res/android"
    package = "com.example.weatherapp">
```

```
<application
```

```
    android:icon = "@drawable/ic_launcher"
    android:label = "@string/app-name"
    android:theme = "@style/appTheme">
```

```
<activity
```

```
    android:name = ".MainActivity"
    android:label = "@string/app-name">
```

```
<intent-filter>
```

```
    <action android:name = "android.intent.action
        MAIN"/>
```

```
    <category android:name = "android.intent.category
        LAUNCHER"/>
```

```
</intent-filter>
```

```
</activity>
```

```
<service
```

```
    android:name = ".WeatherUpdateService"
    android:exported = "false"/>
```

```
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-feature android:name="android.hardware.CAMERA" />
```

Q-8] What is the role of resources in Android development? Discuss the various types of Resources and their significance in creating Structured applications. Provide example to clarify your points.

→ Resources in android development are essential assets and data that are external to your application code. They play crucial role in a well structured application. Provide Example to your Points.

→ Resources in android development are used to separate content from code, customize appearance, behaviour and content.

① Layout Resources:

Type:- xml file in 'res/layout' directory

Significance:- Layout Resources define the structure and appearance of user interface components such as Views and View Groups. They help to separate UI and functionality.

activity_main.xml

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content">
```

<Button

```
    android:id="@+id/button"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="click me"/>
```

</LinearLayout>

② Drawable Resources :-

Type => image file, xml drawable, assets.

Significance : used for displaying images, icon and background in your app

Ex An image resource serves as app launcher icon.

③ String Resources :-

Type => string value.

Significance :- It source text content, including labels, error, message and UI.

Ex A string resource define app's name

Ex <resources>
<string name="app_name">my App</resources>

④ Color Resources

Type :- Color Values.

- Significance :- it define colour palettes for your app's UI Elements.

Ex <resources>

<color name="primary_color"> #
</resources>

⑤ Style Resources :-

Type = XML files

Significance :- it define reusable themes and styles for UI components. They help maintain consistent design throughout the app.

Ex <resources>

<style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">
<item name="colorPrimary"> @color/purple </item>

<item name="android:windowBackground"> @drawable/background </item>

</style>

<resources>

How does an android Service contribute to the functionality of mobile applications? Describe the process of developing android Service.

An Android Service is a component that can perform a long running operation in background without a user interface. It can enhance the functionality of mobile application by allowing it to handle tasks that are not directly related to user interaction. Such as downloading files, playing music. A service can also communicate with other components of application or even with other application using Inter Process Communication.

The process of developing Android service involves several steps:-

① Create a Service class:-

Start by creating a Java or Kotlin class that extends the 'Service' class. You will need to override methods like 'onCreate', 'onStartCommand()', and ' onBind()' as per your service requirements.

② Define the Service in manifest.

* Declare your service in the app's AndroidManifest.xml file. This step is essential to register the service with android's system.

```
<service>  
    android:name = "your Service"  
    android:Enabled = "true"  
    android:Exported = "false">  
</service>
```

- Start or bind service from another component of your application, such as activity or broadcast receiver. Use startService() method to start service.
- You can use bindService() method to bind service that provides a client-server interface that provides a client-server interface runs till another component bound to it.
- Stop or unbind the service when no longer needed. Use stopService() and unbindService() to stop that started service. We can also use stopSelf() method to stop service within itself.

Ans
6/10/23