Assignment - 1

2] 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 app industry

→ As per my knowledge one notable trend in the mobile app industry that has been influencing the android platform that has the rise of Progressive web apps (PWAs) PWAs are web application that offer app like experience directly through web browsers.

Impact on android app developers.

1) Cross Platform Compatibility: PWAs are designed to work seamlessly across various platform and to consider android Developer has to consider creating PWAs alongside traditional android apps to ensure broad accessibility.

2) Enhanced user experience: PWAs aimed to provide a smother and more engaging user experience which set highes expectations for android app developers This encouraged them to focus on improving the quality and performance of their apps to enhance usability.

GHANSHYAM

3) Progressive Enhancement: Developers needs to adopt Progressive enhancement strategies to ensure that android apps remained competitive by offering progress and responsive competition experience.

Impact on bussiness in the mobile app Industry.

1) Cost saving: Bussiness in the mobile app industry development costs by investing in a single PWA that work across multiple platform including Android, rather than building seperate native apps.
2) Competition and engagements: the rise of PWA introduced competitions driving bussiness to innovate their android apps to keep up with evolving user expectations and technology trends.
3) Improved engagements: the focus on delivery app to prioity user engagements and retesing. Ultimately benefit their competition mobile app strategy.

2] What is the purpose of an Inflater of layout in Android development and how does it fit into the architecture of Android layouts.

→ In Android development the purpose of an Inflater specifically a layoutInflater is a create view object from an xml layout resource file it is a crucial component in Android layout architecture beause it bridges the gap between the xml layout definition and the actual view object that are used in the app's user interface.

these hows it fits into the architecture or layout.

1) xml layout files : Developers defines the structure and appearance of uses interface in xml layout files these files describe how elements like TextViews, Buttons, and other widgets are arranged and styled within an activity or fragments.
2) Activity / Fragment : In Android Activities and fragments represents the UI components of an app To populate these UI components with their defined layouts you need to inflate the xml layout files which is where the layout Inflaters comes in.
3) Inflation : The layout Inflater is used to convert an xml layout resource into a corresponding hierarchy of view objects it parses the xml and creates view objects based on the elements and attributes defined in the layout resource
4) Display : once the layout is inflated the resulting view hierarchy is added to the Activity or fragment content view this enables the user interface defined in the xml layout to be displayed on the screen.

→ Examples

```
Protected void onCreate(Bundle savedInstanceState){
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
}
```

GHANSHYAM

the LayoutInflater is a critical component for creating the user interface of android apps by translating xml layout files into actual view objects that can be displayed on the screen making it a fundamental part of Android layout architecture.

3] Explain the concept of an Inflater of layout in android applications. Provide example to illustrate its use.

In Android applications a custom DialogBox is a is often window that overlays the current activity and is often used to interact with the user. gathers input or display information

overview of a custom DialogBox.

- Purpose: Custom dialogs are used when you want to present information, receive user input, or to actions within a self contained isolated UI elements that temporarily interrupts.
- Components: A custom dialog typically consists of various UI elements like buttons text views images or input fields tailored to the specific interactions you want to facilitate.
- Customizations: Developers can design the dialog appearance layout and behavior according to their app branding or specific requirements this customizations allows for creativity in design and functionality.

example:

```
fun showCustomDialog() {
    val customDialog = Dialog(this)
    customDialog.setContentView(R.layout.customDialog_
                                layout)
    val messageTextView = customDialog.findViewById
        <TextView>(R.id.myTextView)
    val okButton = customDialog.findViewById<Button>
                   (R.id.okBtn)
    messageTextView.text = "This is a custom dialog"
    okButton.setOnClickListener {
        customDialog.dismiss()
    }
    customDialog.show()
}
```

→ Use cases of CustomDialogBox : Login / Confirmation Dialog · setting · Information Pop-up · Media playback Controls.

2) How do activities, services and the Android Manifest file work together to make an Android app? Can you describe their main roles and provide a basic example of how they cooperate to design a mobile app?

→ 2) Activities :

Role: Activities represents individual screen or UI components in android app. They manage the user interface and user interactions.

2) Services:

Roles services are background components that perform long running operations or handle tasks that don't interuses they can run even if the app's UI not visible.

3) Android Manifest file:

Role: the Android Manifest file is like the app's blueprint. It decbes the app's components and defines how they interact with the Android system and their components.

Example:

```
class MainActivity:
APP.ComfortActivity () {
    override fun onCreate (saveInstanceState: Bundle!) {
super.onCreate (savedInstanceState)
setContentView (R.layout.activity_main)
startService Button.setOnClick Listener {
    Val ServiceIntent = Intent(this, NotificationService::
                                    class.java)
    startServices (serviceIntent)
    }
}
}
```

→ class NotificationService: IntentServices ("Notification Service") {

override fun onHandleIntent (intent: Intent) {
if (intent != null) {
createNotification()
}
}

```
Private fun createNotificationChannel() {
    val channelID = "my_channel"
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        val name = "my channel"
        val notificationManager.createNotificationChannel(
                                    (channel))
    }
    val Builder = NotificationCompat.Builder
                                    (this, channelID)
        .setSmallIcon(R.drawable.ic_launcher_foreground)
        .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 as an important App's content significant impact how the android system interacts with and manages your App's significance of the Android manifest file:
- App configuration
- Component Declaration
- Permission
- Intent Filters
- app lifecycle.

Examples:

GHANSHYAM

```
<?xml version="1.0" encoding="utf-8" ?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.INTERNET" />

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        // android:icon="@mipmap/ic_launcher"
        android:label="Assignment"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.Assignment"
        tools:targetApi="31" >
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name="SecondActivity" />
        </activity>
    </application>
</manifest>
```

6] What is the role of resources in Android development? Discuss the various types of resources and their significance in creating well-structured applications. Provide example to clarify your points.

→ Resources play a critical role in Android development as they allow you to separate code from assets like layout images strings and more. this separate enhances maintainability localization, and adaptation to different device configuration Lets discuss the various type of resources and significance.

1.) Layout Resources:
Type: XML file (activity_main.xml)
Significances: Define the structure and appearance of user interfaces for activities and fragments.

2) Drawable Resources:
Type: Image files (i.e - launcher.png)
Significance: store images and graphics for use in your app's UI elements such as icons background images and buttons.

3) String Resources:
Type: XML files (strings.xml)
Significances: Store text and string literals used in your app. allowing for easy localization and text management.

GHANSHYAM

4) Colour Resources:
Type: XML Files (colour.xml)
Significance: Define color values used in your app's UI promoting consistent theming and ease of color changes.

5) style. Resources:
Types XML file (styles.xml)
Significance: Store predefined styles and themes for consistent UI design across your app.

6) Dimen Resources:
Type: XML files (dimens.xml)
Significances store dimension values to ensure consistent spacing and layout in your app.

7) Raw Resources:
Type: Arbitrary files (JSON, XML)
Significance: store raw data files such as configurations files or JSON data which can be read and processed by your app.

7] How does an Android services contribute to the functionality of a mobile application? Describe the process of developing an Android services.

7] How does an Android services contribute to the functionality of a mobile application Provide Describe the process developing an Android services.

→ Background Processing : Services enable the execution of long running operations in the background such as downloading file monitoring sensors or handling network request this ensure switches to another app or keep their services.

Communications services can facilitate communication between different Pocess or app or between different Part of an app or components like activity and broadcast receivers to send and receive data or instruction even when they are not actively visible to the users

Multitasking : Services help in multitasking by allowing apps to perform task concurrently For example music apps use services to play music while user interacts with other Parts of the app or their devices.

Notifications : Services can create notifications to keep users informations about ongoing background activities ensuring a seamless and informative user experience.

GHANSHYAM

→ Developing an android service involves the following steps:

Create a service class: Start by creating a new class that extends the Service class or its subclass.

- Define life cycle methods: Override key lifecycle methods such as onCreate(), onStartCommand() and onDestroy(). These methods control when service starts, what it does and when it stops.

- If a component starts the service by calling startService() the service continues to run until it stops itself with stopSelf() or another component stops it by calling stopService()

- Declaring a services in the manifest.

-You must declare all services in your application's manifest file just as you do for activities and other components.

```
<manifest.....>
    <application .....>
        <service android:name = "ExampleService" />
    </application>
</manifest>
```

6.10.23