Assignment :- 1

MEHSANA DISTRICT EDUCATION FOUNDATION SANCHALIT

U. V. Patel College of Engineering

GANPAT UNIVERSITY, KHERVA - 382 711 DIST. MEHSANA. (N.G.)

* MAD *

Name :- Aparnathi Chintan H
Enrollment :- 21012011002
Sub :- MAD
Batch / Class :- SB-1 / CEIT-B

Q1 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 as in the mobile app industry.

AI :- Artificial intelligence and
ML :- Machine Learning

-> AI & ML are integrated into mobile app development to make it smarter.

-> Recently apple launched core ML3 on IOS. Machine learning framework designed to help developer incorporate AI technology into their device apps. Not only this but we also saw google introduce new feature to its Maps app which uses AI to improve user experience.

-> AI / ML have become one of the top mobile app development trends for 2023. In 2023, we can expect AI to reshape how App are built making them more intelligent, more

efficient & capable etc.

→ Augmented Reality and virtual Reality

augmented reality (AR) and virtual Reality (VR) features and this trend is not going away according to research, mobile AR was estimated at $12.6 billion in 2020. By 2030, it's forecast to grow to $184.61 billion ~~2030, it's fo~~

→ Mobile IoT APPS

The integrate of of things has become mainstream in many sectors from healthcare and agriculture to manufacturing & transeportation. This resulted in increased development of mobile IoT APPS

→ Mobile Payments!

→ currently, ecommerce is one of the most rapidly developing markets in the world and mobile shopping is among the top trends consequently the demand for mobile payment is growing.

→ Cloud-Based mobile apps

→ apps that integrate advanced technology such as AI/ML, IoT, etc.. requiring a lot of storage in mobile device's internal memory. But a gigabyte sized, app is not a solution or choice for users

Impacts :- PWA: Progressive web APPS

→ cost-Efficiency & Developing native android app ~~costning - especially~~ can be expensive and time consming especially if a bussiness wants to a more cost effective solution

→ cross-platform compatibility : PWA are not limited to android devices, they can be accessed on various platforms, including ios and desktop browsers

→ Improved user experience : PWAs are designed to provided a seamless user experience similar to native apps they can be installed on home screen, offline, push notification etc

→ other impacts like

→ Performance and speed
→ App distribution
→ App store independence etc.

Q2 what is the purpose of an inflater of layout in android development, and how does it fit into architecture of android layers.

→ In android development, an "inflater" is a crucial component used to create and initialize the contents of a layout XML file into corresponding view objects within your android app. The primary purpose of an inflator is to convert an XML layout file into a view hierarchy that can be displayed on the device's screen. This process is essential for building the user interface of android apps

→ Here's how the Inflater fits into the architecture

① layout XML files: In android, UI components are typically defined in xml layout file

these XML files describe the structure and appearance of the UI elements specifying things like arrangement of widgets their attributes & their hierarchical relationships

② Activity or fragment: In the android app an activity represents a single screen with a user interface, while a fragment represents a portion of an activity's UI The activity or fragment is responsible for inflating and managing the layout

③ Inflation: when the Activity or fragment needs to display the UI defined in an XML layout file, it uses an Inflate to "inflate" the XML layout file Inflation in this context means converting the XML layout into a set of view objects that can be added to the app's UI

④ view hierarchy: The Inflater creates a hierarchy with the root view which is typically a viewgroup This viewgroup contains child view organized as specified in the xml layout file

⑤ Display: once the view hierarchy is create it can be added to the app's UI either by setting it as content view of an Activity or by adding it to a fragment's view hierarchy the android framework then handles reading the UI on the screen

Q 3) Explain the concept of create customDialog-Box in android app provide examples to illustrate its use

→ A custom dialog box in Android a user interface component that allows you to create a customized, pop-up style window with your own layout and content. This is useful when you want to display information, collect input or provide interaction in a manner that differs from standard

① Create xml file

→ <? xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns: android="https://schemas.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="wrap_content"
android:orientation="vertical"
android:padding="16dp" >

<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="Custom DialogBox"
android:textsize="18sp"
android:textcolor="@android:color/black"
android:layout_gravity="center_horizontal"
android:layout_marginBottom="8dp" />

< Button
android: layout_width = "match_parent"
android: layout_height = "wrap_content"
android: text = "ok"
android: id = "@+id/dialogBt" />

</linearlayout>

→ kt file

```
import android.app.Dialog
import android.content.Context
import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.widget.Button

class customdialog (context : context) : Dialog (context) {

override fun onCreate (savedInstanceState: Bundle?) {
    super.onCreate (saveInstanceState)

val Inflater = layoutInflater.from (context)
val view = Inflater.inflate (R.layout.custom_dialog
                             , null)

set contentView (view)

val button = view.findViewById<Button> (R.id.dialogBt)

button.setonclicklistener {

        dismiss()
    }
}
}
```

Display box

```
val customDialog = customDialog (this)
customdialog.show()
```

Q4 How do activities services and the Android manifest file work together to make an android app? Can you describe example of how they cooperate to design a mobile app?

→ Activity:

Role: Activity represent the user interface and serve as the building blocks of the app's UI. Each screen or UI component in your app is typically implemented as an activity. Activities manage the interaction with the user, handling user input and displaying UI elements.

Ex → suppose you are building a simple android app with two screen a login screen and home screen you should create two activity one for each screen (authentication, while home screen activity would display the App's main content.

→ class loginActivity : AppCompatActivity() {
// —— code —— //
}
class HomeActivity : AppCompatActivity() {
// —— code //

(2). Service :- Service are background components that Role: perform long-running tasks independently of the UI they are used to execute tasks that should continue running even when the app is not in the foreground. Service typically used for

task such as playing music, handling background network ~~task~~ result

Ex Let's say you want you app to play music in the back ground even when to use switches to another app our locks the devices you could create a service to handle the music playback

```
Class MyService : Service() {
    // Implement music playback login here
}
```

(3) android manifest:

The androidfest.xml file is a crucial Role:- configuration file the provide essential information about your app to the android system. It specifies the app's components permission and how the app should behave when launched our interacts with other parts of the system

Ex In the android manifest file you declare the activities and services you've created and define their properties and relationship you also specify permission your app needs and declare any intent filter for components that should respond to specific system events

→ In practical, these components work together to create a seamless user experience, for example when user launches your app, the ~~and~~ android system reads the manifest file to understand which activity to start, the activity, in turn interact with the user interface. If your app plays music, you can use a service to ensure that music continues to play even when the user

switches to another activity in app

-> Remember that this is a basic overview, and real world app often involve more complexity and may include additional components such as broadcast content provider etc.

Q5 How does the Android manifest file impact the development of an android app? Provide an example to demonstrate it significance

=> The androidmanifest.xml file plays a crucial role in the devlopemment of an android app. It provides essential info. the android system about your app's components permission and behavior. The manifest file is significant for several reasons

(1) Declaring App Components:- The android manifest file is where you declare all the components of your app, including activities services broadcast receivers and content providers. The declarations inforons the android system about the various parts of your app and how they should interact.

(2) Setting entry points:- The manifest file specifies which activity is the main entry point of your app

(3) Defining Intent Filter :- You can define Intent Filter for your app's components in the manifest file. Intent specify type of intents that each component can respond to This is how other apps or system components can communicate with your app

(4) Requesting Permission :- If your app requires certain Permission to access device features or data like camera contacts etc, you must declare these permission in the manifest file

→ Ex :, suppose your app need camera and internet permission so

→ manifest . xml

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

<uses-permission android:name = "android.permission
        .camera" >
<uses-permission android:name = "android.permission
        .INTERNET" >

<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=".ShareActivity">
    <intent-filter>
        <action android:name="android.intent.action.SEND" />
        <category android:name="android.intent.category.DEFAULT" />
        <data android:mimeType="image/*" />
    </intent-filter>
</activity>
</application>
</manifest>
```

Q6 What is the role of resource in android development ? Discuss the various types of resource and their significance in creating well-structured app Provide example to clarify your point :-

→ Layout Resource :-
This is the structure and appearance of user interfaces in your app. they are define in XML file and describe the arrangement and properties of UI elements.

→ layout resource allow you to create consistent and visually appealing UIs that can adapt different screen sizes & orientation. they help separate the UI design from the code, promoting maintainability and reusability.

</linearLayout> // Element info like textview etc.
</LinearLayout>

② String Resources :- String resources text and string values that are used in your app's UI and code They are typically defined in XML file and can support multiple language and locales

→ String resources enable localization and internationalization of your app, making it accessible to global audience. They also centralize text content making it easier to update and maintain translations

```
<resources>
    <string name="name"> my name </string>
</resources>
```

③ Drawable resources
→ Drawable resource is image, icons, & graphics in your app's UI. They can be in various formats and are stored different drawable directories based on devices screen density

→ Drawable support high-quality graphics across different devices and screen resolution

Ex we can place different icon to app with use of this folder

④ color resources!
→ color Resources define colors that are used in your app's UI and can be easily referenced from XML layout files and code

→ By centralization color definations, color resources make it simple to apply consistent color schemes across your app

<resources>
<colan name = "primary-colan">
#007ACC </colan>
</resources>

③ Dimention Resources :
→ dimension resource store numeric values, such as margins, padding etc.

→ this is easy to maintain consistent spacing and sizing throughout app

<resource>
<dimen name = "text"> 24sp </dimen>
</resource>

④ Style and theme resource
→ style and theme resources define appearance and styling of UI elements. use for creat. custome themes for app

→ it use to maintain consistant look throughout out APP

Ex
<resources>
<style name = "Apptheme" Parent = "Theme.AppCompat.Light.Dark">

<item name = "colorPrimary">@color/primary
</item>

</resources>

(Q7) How does an android service contribute to the functionality of an mobile app Describe the proces of developing andrid service ?

→ an android service contribute to the function-ality of a mobile app

→ an android service plays a crucial role in the functionality of a mobile app by tasks to run into background even item the APP is not actively in use

→ contribution of service

① background Processing: service run tasks in the background ensuring the essential functions like music Playbasic, location, tracking, etc without disrupting the user interface

→ ② long Running oper.:- services are ideal for operations that take a long time to complete such as doonloading large files or performing complex calculation without the app to freeze

③ Forground service:- some services can Run in forground displaying a Persistent notification to keep the user doing tasks like navigation or chat app

④ Inter-component :- service can comm with other app components through interface allocating exchange and connection

→ Developing an android service

(1) create a service class:-
→ Extend the Services class or one of its classes like "IntentService" or "To base service"
→ Implement the service's functionality with the 'onCreate' & 'onStartCommand' method

(2) declare in the manifest
→ Register your service in the android manifest .xml file to make it accessible to the system and other components

(3) service lifecycle:-
→ Understand the service's lifecycle methods [onCreate, onStartCommand, onBind, onDestroy] and, override them as needed
→ service can run in three methods foreground, background, bound

(4) start and stop the service
→ start a service using 'start service' or bind to it using 'bind service (IntentService, connection int)
→ stop a service, when it's no longer needed using 'stop service (intent)' or stopSelf()

(5) foreground service
→ To create a foreground service provide a notification, that informs to user about on going tasks

→ use 'start foregrounds' to start a service in the foreground mode

(6) Thread management

→ when performing time-consuming operations, consider using worker threads or AsyncTask to prevent blocking to main UI thread

(7) Communication :- use intent extra, broadcast receiver, or interface to enable communication b/w services and other app components

(8) Cleanup and Resource management :-

→ Ensure that you release resources and stop the service when it's no longer needed to prevent unnecessary battery

(9) testing :-

Thoroughly test your service to ensure it works as expects including scenarios like app background tasks ~~interruptions~~ and restarts interruptions

Wadj

05.10-23