SILVER OAK UNIVERSITY
EDUCATION TO INNOVATION

NAVKAR ENTERPRISE
PAGE NO. 1
Date : / /

Q-1 What is the use of the Android manifest file?

Ans. In the context of Mobile Application Development (MAD), the android manifest file serves as a key component in Android app development.

- It contains essential metadata about the app, defining its components, permissions and overall structure. This file guids the Android system in properly launching and managing the app, ensuring seamless integration and functionality.

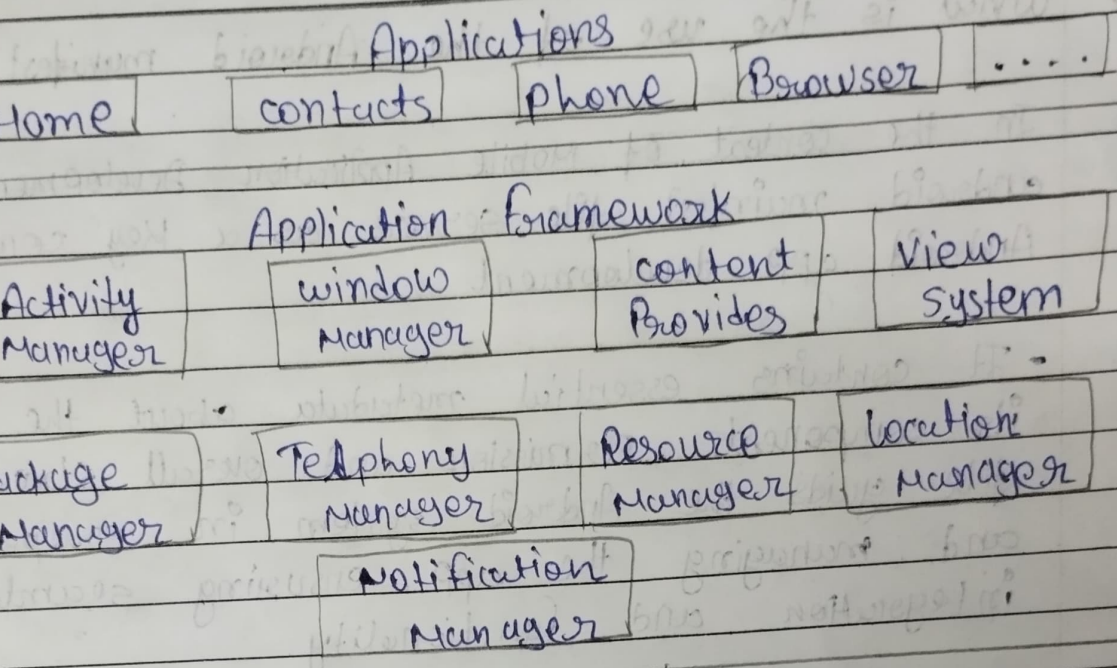
Q-2 Explain Android architecture with a diagrams.

Ans. 1). Linux Kernal :-

This is the kernal on which Android is based. This layer contains all the low level device drivers for the various hardware components of an Android device.

2) Libraries :-

These contains all the code that provides the main SQLite features of an Android OS for example, the SQLite library provides database support so that an application can use it for support so that storage. The webkit library provides functionalities for web browsing.



3. Android runtime :-

At the same layer as the libraries, the Android runtime provides a set of core libraries that enable developers to write Android apps using the Java Programming language. Then android runtime also

Include the Dalvik virtual machine, which enables the Dalvik virtual machine, which enables every Android application to run in its application are compiled. into Dalvik executede Dalvik is a specialized virtual machine designed specifically for Android and optimize for battery powered mobile devices with limited memory and CPU.

4). Application Framework.

Exposes the various capbilities of the Android OS: to application developers so that they can make use to them in their application.

5): Applications :-

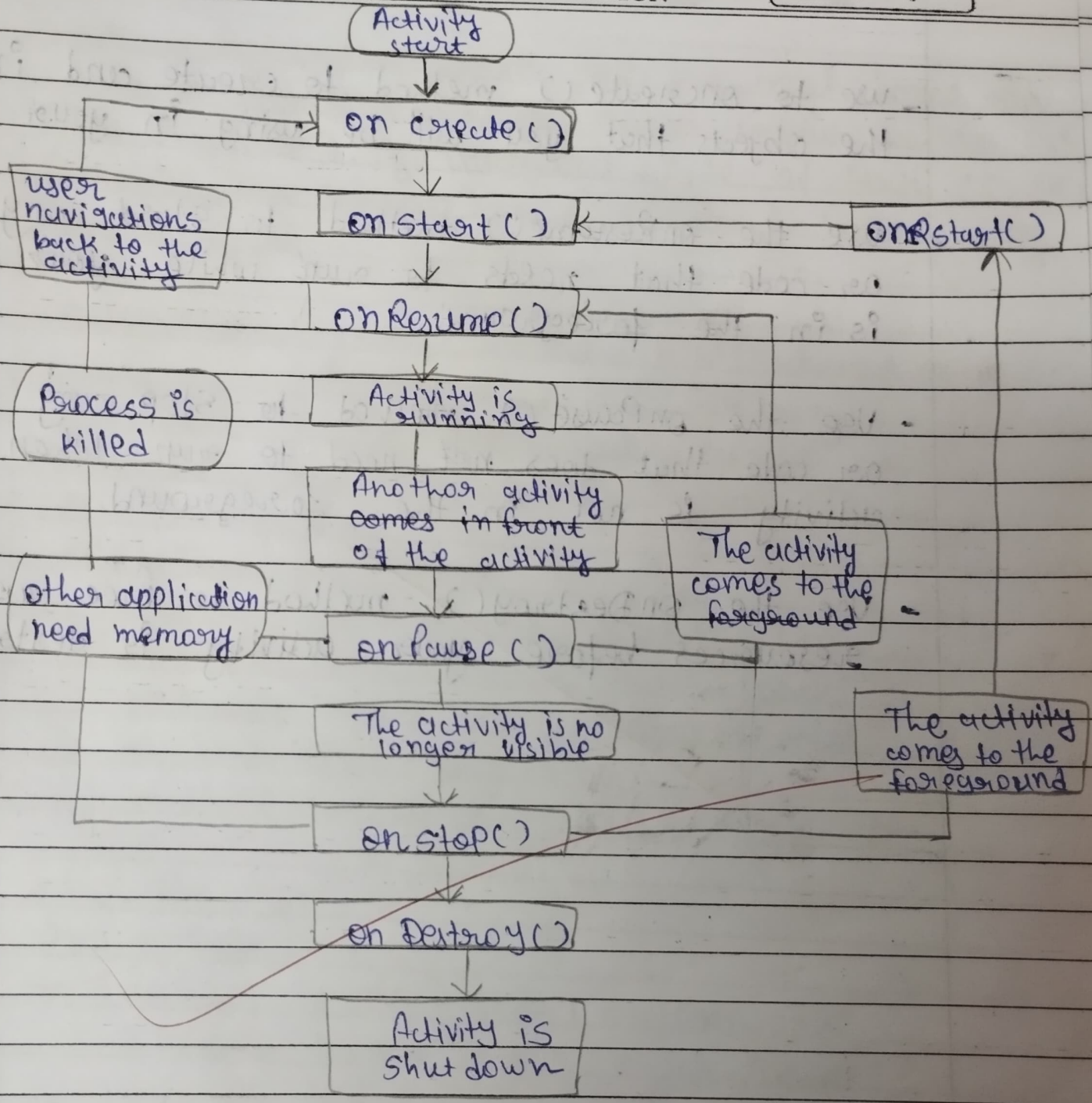
At this top layer, you will find applications that ship with the Android device (such as phone, Contacts, Browser, ect), as well as applications that you download and install from the Android Market. Any application that you write are located at this layer.

What is Activity ? Draw and explain the Activity life-cycle in details.

The Activity base class defines a series of events that govern the life cycle of an activity. The Activity class defines the following events.

- onCreate() :- Called when the activity is first created.
- onStart() :- Called when the activity becomes visible to the user.
- onResume() :- Called when the activity starts interacting with the user.
- onPause() :- called when the activity is being paused and the previous activity is being resumed.
- onStop() :- called when the activity is no longer visible to the user.
- onDestroy() :- called before the activity is destroyed by the system (either manually or by the system to conserve memory)

onRestart() :- called when the activity has been stopped and is restarting again.



By default, the activity created for you. Contains the onCreate() event within this event handler is the code that helps to display the UI element of your screen.

- use to onCreate() method to create and instantiate the objects that you will be using in your application.

- use the onResume() method to start any services or code that needs to run while your activity is in the foreground.

- Use the onPause() method to stop any services or code that does not need to run when your activity is not in the foreground.

- Use the onDestroy() method to free up resources before your activity is destroyed.

Q-4 Create a Registration screen that has input boxes for userName, Password, Address, crender (radio buttons for male & female), Age (numeric), Date of Birth (Date Picker), state (spinner) and a submit button using Relative layout.

Ans.

```
<RelativeLayout xmlns:android="http://schemas.android.
                com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
<!-- User Name -->
<EditText
    android:id="@+id/editTextUserName"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="user Name"
    android:layout_marginTop="16dp" />

<!-- Password -->
<EditText
    android:id="@+id/editTextPassword"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Password"
    android:inputType="textPassword"
    android:layout_below="@id/editTextUserName"
    android:layout_marginTop="8dp" />
```

<!-- Adress -->
< Edit Text
android:id = "@+id/edit Text Address"
android:layout_width = "match_parent"
android:layout_height = "wrap_content"
android:hint = "Address"
android:layout_below = "@id/edit Text Password"
android:layout_marginTop = "8dp" />

<!-- Gender Radio Buttons -->
< Radio Group
android:id = "@+id/radio Group Gender"
android:id = "width = "match_Parent"
android:layout height = "wrap_content"
android:layout_below = "@id/edit Text Address"
android:layout_marginTop = "8dp">

< Radio Button
android:id = "@+id/radio Button Male"
android:layout_width = "wrap_content"
android:layout_height = "wrap_content"
android:text = "Male" />

< Radio Button
android:id = "@+id/radio Button Female"
android:layout_width = "wrap_content"
android:layout_height = "wrap_content"
android:text = "Female" />

```
</Radio group>

<!-- Age -->
<Edit Text
    android:id="@+id/edit Text Age"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Age"
    android:inputType="number"
    android:layout_below="@id/radio group ccender"
    android:layout_marginTop="8dp" />

<!-- Date of Birth -->
<Date Picker
    android:id="@+id/date Picker"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@id/edit Text Age"
    android:layoutmarginTop="8dp" />

<!-- State Spinner -->
    android:id="@+id/spinnerstate"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@id/date Picker"
    android:layout_marginTop="8dp" />

<!-- Submit Button -->
<Button
    android:id="@+id/btn Submit"
    android:layout_width="match_parent"
```

```
android:layout_height = "wrap_content"
android: text = "submit"
android: layout_below = "@id/spinnerstate"
android: layout_marginTop = "16dp" />

</RelativeLayout>
```

-5 What is an Intent ? Design an Android application to send SMS using Intent.

Ans. In android an 'Intent' is a messaging object that is used to request an action from another app component it is used to connect components like activities. services and broadcast receivers. either within your app or with components of other apps.

To design an Android application to send an SMS using an 'Intent'. you can create an 'Activity' with a button. and when the button is clicked, an 'Intent' will be used to launch the SMS app with pre-filled data.

Q-6. Write code to display Toast Message on click of Button.

Ans 1). Layout file ('activity_main.xml'):

```
<?xml version="1.0" encoding="utf-8" ?>
<Relativelayout xmlns:android="http://schems.android.com/apk/res/android"
android:layout_width="match_parent"
android:layout_height="match_parent" >

<!-- Button to trigger Toast Message -->
<Button
    android:id="@+id/btnShowToast"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Show Toast"
    android:layout_centerInParent="true" />

</Relative layout>
```

Q-7 Explain basic building blocks/ components of Android Application.

Ans. Android applications are built using various components that work together to provide a seamless user experience.

- The basic building blocks or components:

1) Activity :-
   - An 'Activity' represents a single screen with a user interface.
   - It is the primary building block for the UI and user interaction.
   - Each activity is typically associated with a layout file that defines its UI.

2) Service :-
   - A 'service' is a background process that runs without a user interface.
   - It performs long running operations or processes data in the background.
   - It can run independently of the UI components and other services.

3) Broadcast Receiver :-
   - A 'Brodcast Reciever' is a component that responds to system wide broadcast announcements.

- It allows the application to receive and react to broadcast messages from the system or other applications.

4) Content Provider :-

- A 'content Provider' manages access to a structured set of data.
- It Provides a standardized interface to share data between applications securely
- Commonly used for database operation and accessing shared data.

5) Intent :-

- An 'Intent' is a messaging object that is used to request an action from another app component.
- It can be used to start an activity service or broadcast receiver and to communicate between components.

6) Fragment :-

- A 'Fragment' is a modular section of an activity that has its life cycle and UI.
- It allows creating more flexible and responsive user interfaces, especially for large screens.

7). Layout & virus :-

- layouts define the structure of the UI and how components are arranged
- 'view' are UI elements like buttons text fields .etc : that are placed within layouts.

8). Mainfest file :-

- The 'Android Mainfest.xml' file contains essential information about the application.
- It declares the application's components permission and other configurations.