

Lecture 3:

Aakash R
CB.SC.P2CYS23011

App components:

Basic Components:

Activities: *Activities are the entry points for user interactions with an app. Each activity represents a single screen with a user interface. Activities work together to form a cohesive user experience but operate independently.*

Services: *Services are components that perform long-running operations in the background. They don't have a user interface but can run in the background even when the user isn't interacting with the app.*

Broadcast receivers: Broadcast receivers respond to broadcast messages from other applications or from the system. These messages are often referred to as events or intents.

Content Providers: Content providers manage access to a structured set of data. They encapsulate data and provide mechanisms for defining data security. Content providers are the standard interface that connects data in one process with code running in another process.

Additional Components:

Fragments: Fragments are modular sections of an activity. They have their own lifecycle and can be added or removed while the activity is running. Fragments make it easier to build dynamic and flexible UI designs on large screens.

Views: Views are the basic building blocks of user interface components in Android. A view occupies a rectangular area on the screen and is responsible for drawing and event handling.

Layouts: Layouts are container views that hold other views (widgets) and define their position on the screen. Android provides several types of layouts like Linear Layout, Relative Layout, and Constraint Layout

Resources: Resources are external elements like strings, images, colors, dimensions, and layouts that you can use in your app. They help in maintaining consistency and support localization.

Manifest: Essential for any Android application. It provides essential information about the app to the Android system, which the system must have before it can run any of the app's code.

Android ▾

▾ app

▾ manifests

AndroidManifest.xml

▾ java

▾ com.example.lecture1

FirstFragment

MainActivity

SecondFragment

▾ com.example.lecture1 (androidTest)

ExampleInstrumentedTest

▾ com.example.lecture1 (test)

ExampleUnitTest

java (generated)

▾ res

▾ drawable

</> ic_launcher_background.xml

</> ic_launcher_foreground.xml

▾ layout

</> activity_main.xml

</> content_main.xml

</> fragment_first.xml

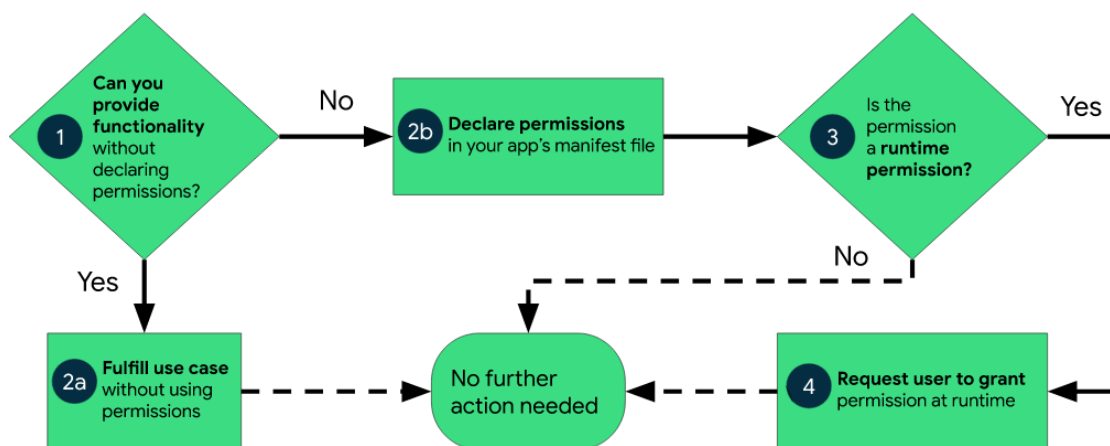
</> fragment_second.xml

Permissions on Android:

App permissions help support user privacy by protecting access to the following:

- **Restricted data**, such as system state and users' contact information
- **Restricted actions**, such as connecting to a paired device and recording audio.

Workflow for using app permissions:



Types of permissions:

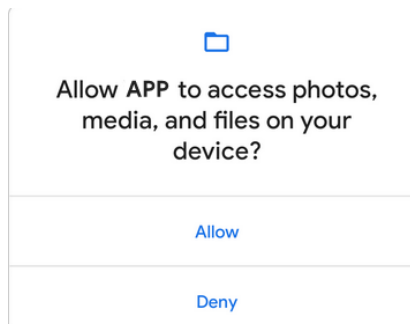
Normal Permissions

- These permissions protect less sensitive data or resources. They are automatically granted by the system upon app installation.
- **Examples:**
 - Accessing the internet: `android.permission.INTERNET`
 - Setting an alarm: `com.android.alarm.permission.SET_ALARM`

Dangerous Permissions (RUNTIME PERMISSION)

- These permissions protect sensitive user data and critical device functions. Users must explicitly grant these permissions.
- Therefore, you need to request runtime permissions in your app before you can access the restricted data or perform restricted actions.
- **Examples:**
 - Accessing the user's location: `android.permission.ACCESS_FINE_LOCATION`
 - Reading contacts: `android.permission.READ_CONTACTS`
 - Accessing the camera: `android.permission.CAMERA`

- **Characteristics:** Require explicit user consent at runtime. If an app needs a dangerous permission, it must include a corresponding <uses-permission> element in the app's manifest file and request the permission at runtime.



Broadcast Receiver:

Broadcast in android is the system-wide events that can occur when the device starts, when a message is received on the device or when incoming calls are received, or when a device goes to airplane mode, etc.

Broadcast Receivers are used to respond to these system-wide events. Broadcast Receivers allow us to register for the system and application events, and when that event happens, then the register receivers get notified. There are mainly two types of Broadcast Receivers:

- **Static Broadcast Receivers:** These types of Receivers are declared in the manifest file and works even if the app is closed.
- **Dynamic Broadcast Receivers:** These types of receivers work only if the app is active or minimized.

Some of the important events:

Intent	Description Of Event
android.intent.action.BATTERY_LOW :	Indicates low battery condition on the device.
android.intent.action.BOOT_COMPLETED	This is broadcast once after the system has finished booting
android.intent.action.CALL	To perform a call to someone specified by the data
android.intent.action.DATE_CHANGED	Indicates that the date has changed
android.intent.action.REBOOT	Indicates that the device has been a reboot
android.net.conn.CONNECTIVITY_CHANGE	The mobile network or wifi connection is changed(or reset)
android.intent.ACTION_AIRPLANE_MODE_CHANGED	This indicates that airplane mode has been switched on or off.