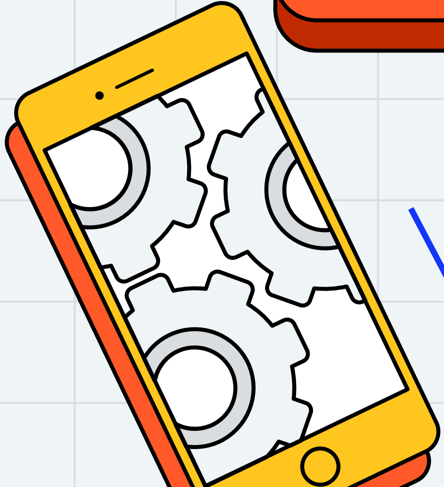
Application Programming

Hend Alkittawi



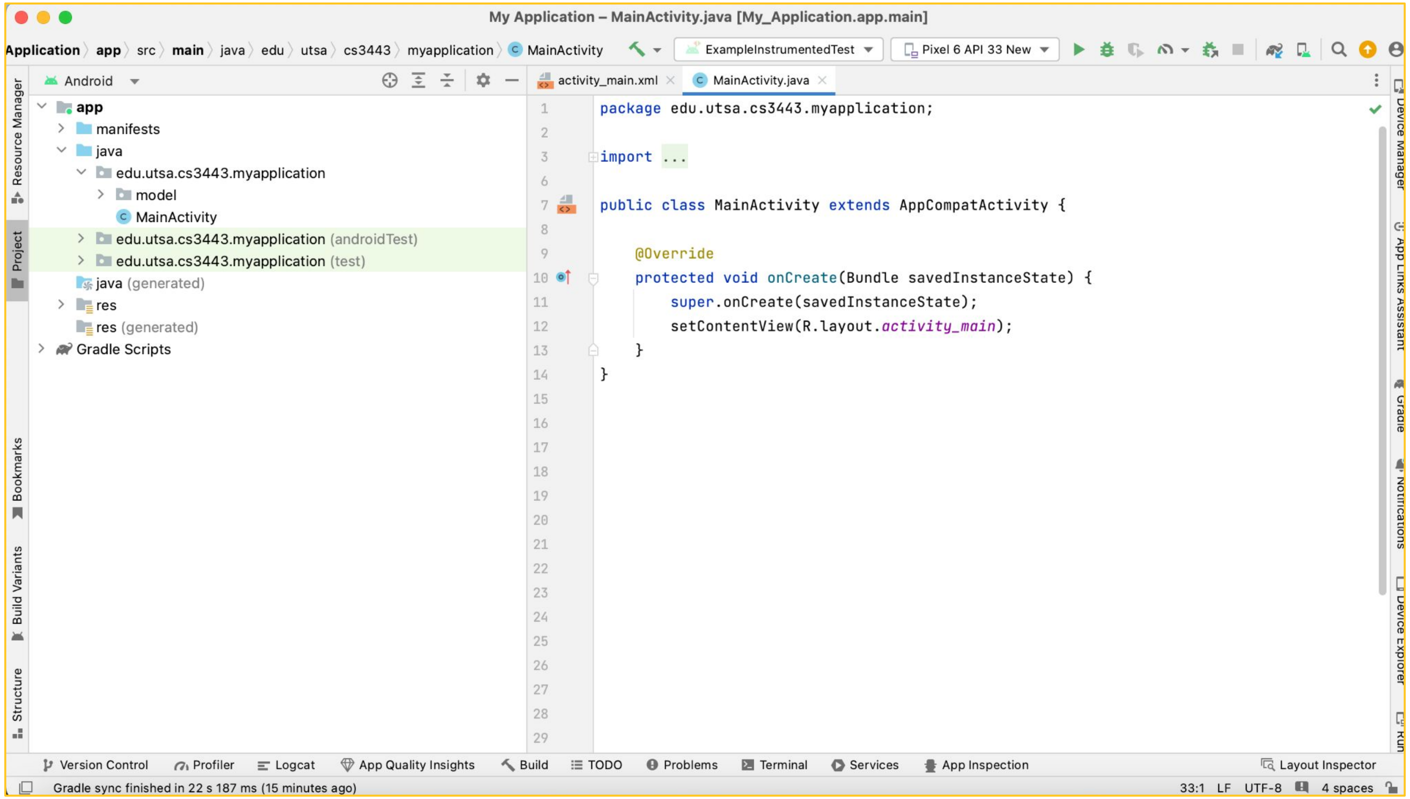
Android Testing

Introduction to Testing Android Apps

ANDROID TESTING

- There are two types of tests for unit testing in Android, both of which use the JUnit testing framework
  - JVM tests for testing your Java code in the project.
    - Execute quickly (milliseconds) on your machine’s JVM
    - Located in the test source set (project > app > src > test)
  - Instrumented tests for testing code that uses the Android SDK
    - May take longer (seconds) on an emulator or Android device directly
    - Located in the androidTest source set (project > app > src > androidTest)

ANDROID TESTING



```
package edu.utsa.cs3443.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

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

Gradle sync finished in 22 s 187 ms (15 minutes ago)

ANDROID TESTING

- Shortcut to the "create a test" wizard: ctrl+shift+T (or command+shift+T) when your cursor is inside of the class you want to create a test for.
- To run a test
    - Right click on the test class, choose "Run" OR Click ▶ next to the test name to run
    - For instrumented tests, connect a device first (these tests require a device - virtual or physical).

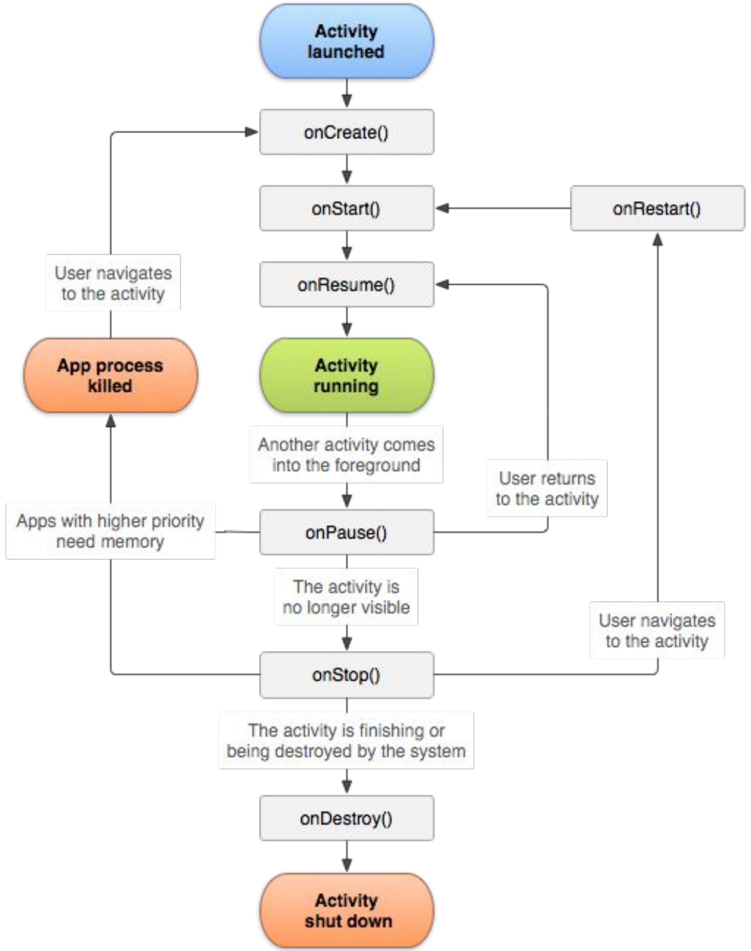
ANDROID TESTING

- JVM Tests
  - Test the Java code in your project
  - Follow the same paradigms as the JUnit tests previously discussed
    - Naming conventions for classes and methods
    - Use of assert statements
- Instrumented tests
  - Test code in your project which uses the Android SDK (e.g. Activity, TextView, etc)
  - ActivityScenario provides APIs to start and drive an Activity’s lifecycle state for testing!
  - Espresso for Android UI testing
    - State expectations
    - Interactions
    - assertions

ACTIVITY LIFE CYCLE

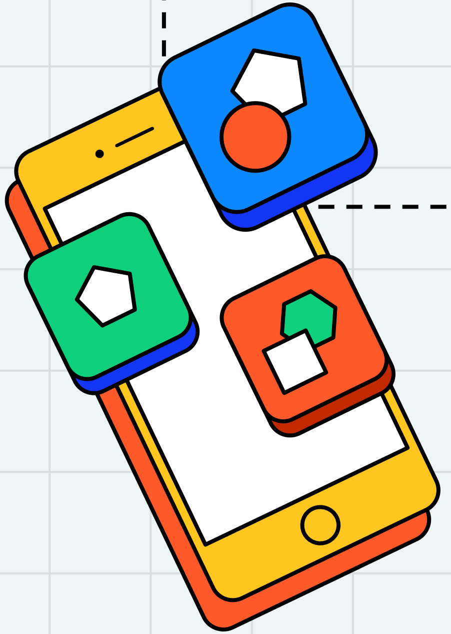
- As a user navigates through, out of, and back to your app, the **Activity instances in your app transition through different states** in their lifecycle.
- The Activity class provides a number of callbacks that let the activity know when a state changes or that the system is **creating**, **stopping**, or **resuming** an activity or **destroying** the process the activity resides in.
- The activity lifecycle | Android Developers

ACTIVITY LIFE CYCLE


Activity launched
onCreate()
onStart()
onRestart()
User navigates to the activity
onResume()
App process killed
Activity running
Another activity comes into the foreground
User returns to the activity
Apps with higher priority need memory
onPause()
The activity is no longer visible
User navigates to the activity
onStop()
The activity is finishing or being destroyed by the system
onDestroy()
Activity shut down

CODE DEMO

- Show how to run Android tests in Android Studio.



THANK YOU!

DO YOU HAVE ANY QUESTIONS?

hend.alkittawi@utsa.edu

By Appointment

Online