

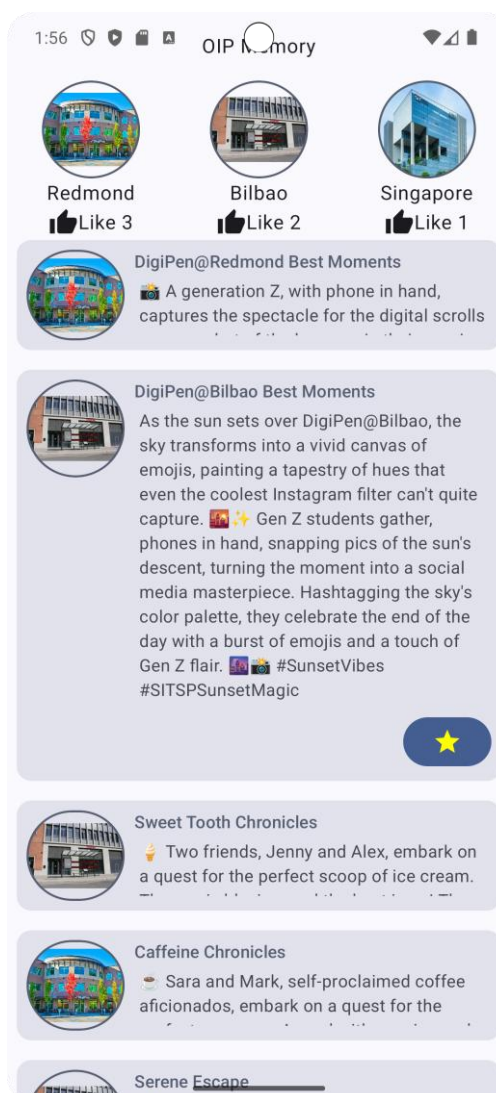
OIP Experiences: Introduction

CSD3156 Mobile and Cloud Computing Spring 2026

Overview

This lab provides exercises and guidelines to gain familiarity with the Android Development using Jetpack Compose.

You are going to build an app to share your (hardcoded) OIP experiences using Jetpack Compose with basic Compose layout elements such as Row, Column, and LazyColumn, UI elements such as Text and Icon, as well as managing basic using remember and MutableState.



Jetpack Compose Basics

<https://developer.android.com/courses/android-basics-compose/unit-1>

<https://developer.android.com/codelabs/jetpack-compose-basics-1-9>

Outcomes

Upon completion of the session, you should be able to:

- Get familiar with Android Development
- Use basic UI elements and layout in Jetpack Compose

Creating a basic UI using Jetpack Compose

The goal of this exercise is to practice basic UI elements/layout and logic in Jetpack Compose.

- Fork the repo `csd3156-lab01-2026`.
- The data is in `data.kt`, please add more entries (more than 10), but don't delete the given two, the texts added should not contain any numbers. It is of the format of `List<MemorableData>`, can be passed as parameter to the screen using `data.momentsList`.
- The app has only one Activity: **MainActivity**.
- Note that the top part is a Row of 3 Columns (campuses) (testTag: **LikeRedmond**, **LikeBilbao**, **LikeSingapore** respectively, use `Modifier.testTag` to set the testTag) with Image, name, Like Icon and Like count. Ensure that each entire column is clickable. Each click will increase the Like count by 1 starting from 0.
- The bottom part is a LazyColumn (testTag: **LazyColumn**), of more than 10 items, each item is a Row (can be held with a Card) with Image, title and text, make the entire row clickable. Whether it's the image or the accompanying text, users should experience seamless navigation by clicking anywhere on the Row to toggle toggle expansion on or off.
- Enhance user engagement with a star button that appears when the one item is clicked to be expanded (full text is displayed). Include a Button (testTag: **Star**, all the Star Icons use the same testTag) with a Star Icon that can be toggled as Stared.
- You can use `Icons.Filled.ThumbUp` and `Icons.Filled.Star` for the icons.
- You need to recreate a UI similar to the screenshot above.
- It's fine if the expanded or starred states are reset after scrolling the LazyColumn, but it would be better if those states could persist.

IMPORTANT: Ensure that the activity is named as MainActivity:

- The Campus Columns are with the testTag "LikeRedmond", "LikeBilbao", "LikeSingapore", e.g., using `Modifier.testTag("LikeRedmond")`.
- In each row in LazyColumn (testTag: LazyColumn), there is a Star Button (testTag: Star)
- Like counts are initiated as 0.
- At least 10 items added to data, the texts contain no numbers.
- Don't change the build.gradle, setting, and Manifest files, otherwise you might not be able to use the test apk to test by yourself.

Lab Exercise 1

Due Date: Wed, Jan 14 2026 2359 hrs

Fork the repo `csd3156-lab01-2026` and then clone it.

1. Implement the UI and UI logic.
2. Test your program using `UIInstrumentedTest` and the test apk.
3. Commit and push all changes to your forked repository.

END OF DOCUMENT