

# Assignment 1 (30%)

---

## Overview

This Assignment is worth 30% (*including the marks from Week 3 & Week 4 Assessed Lab Tasks*).



**Due Date:** Assignment 1 is due on Friday of Week 4, 22nd March 2024, at 11:55 PM.

**Interview:** Allocated lab in Week 5.



**Generative AI tools cannot be used in this assessment task**

*In this assessment, you must **not** use generative artificial intelligence (AI) to generate any materials or content in relation to the assessment task.*

In this assignment, you will apply the following skills you will have learnt during Weeks 1-4:

- Java and XML
- Basic UI design
- Adding styles and themes
- Develop a multi-activity Android app (i.e. multiple Views)
- Save data to persistent storage (Shared Preferences)
- Listen to incoming SMS messages on a device
- Send a custom broadcast message for passing data from one Activity to Another

---

## Assignment Theme □



### **Generative AI tools cannot be used in this assessment task**

*In this assessment, you must **not** use generative artificial intelligence (AI) to generate any materials or content in relation to the assessment task.*

Suppose you are working for a software company, and a new client who is in the Events industry wants you to develop an Android application for their company. They need an Event Management Application that will help them see the different events that they organise.

This Events management app needs to take two types of input data (Entities (Objects)) from Event Organisers:

- Event Category: What kind of an event is this?
- Event: What is the event?

Store the above two data points in a tabular format (just one record per object in Assignment 1, we will cover how to implement databases in the next assignment).

**Security:** Your app should have the ability to register new event organisers (signup) and login functionalities.

---

# Assignment 1 Specifications ⚙️



Assignment 1 interview will be conducted in labs in week 5. Please note, a **"no interview no marks"** policy is applied across all assignments.



This is an individual assignment, submission via Moodle only. [What to submit on Moodle?](#)



## **Generative AI tools cannot be used in this assessment task**

*In this assessment, you must **not** use generative artificial intelligence (AI) to generate any materials or content in relation to the assessment task.*

As mentioned in the Assignment Theme section, this assignment is about developing an Events Management App (EMA).

## Entities

### A) Event Category

### B) Event



Use Empty Views Activity template when you create a new project or add a new activity to your project.



~~You must use Guidelines or Barriers in your UI design~~

# Activities (Screens)

## 1) Sign Up/Register (Launcher Activity)

1.1 Develop a signup (registration form) activity that takes as input the following fields and saves them in persistent storage (Shared Preferences).

- Username
- Password
- Password Confirmation

1.2 The username and password can be used later to log in to the app. The app should be able to store the registered username and password in persistent storage.

1.3 The activity must show the status of registration in a Toast (i.e. Success/Failure message). If the registration is successful, redirect the user to the login activity.

1.4 Sign-up activity must remain usable and friendly if you change the orientation to landscape.

1.5 Add a new button to launch Login Activity if the user already registered.

## 2) Login

2.1 Develop a login activity that takes a username and password, and verifies them with the stored

username-password pair in the SharedPreferences.

2.2 On successful login redirect the user to the Dashboard Activity. If login fails the activity must show the failed to login message in a Toast. eg: "Authentication failure: Username or Password incorrect"

2.3 Login activity must remember the last saved username and prefill the username field with the remembered username value.

### 3) Dashboard Activity

3.1 New button to Launch "New Event Category" Activity, see below point 4.

3.2 New button to Launch "Add Event" activity, see below point 5.

### 4) New Event Category

4.1 The "New Event Category" activity should allow users to enter the details of a new category.

- UI element to capture all attributes of Event Category listed under table A above.
- Display generated Category Id after each successful save operation.

4.2 Add a new button to save details of a category to SharedPreferences. For A1, only one category can be saved, no need to store multiple categories in Shared Prefs. In other words, if you try saving another category after saving the first one, previous values will be overwritten.

4.3 Fill Event Category fields using incoming messages using the following format:

**category:Name;EventCount;IsActive**

The SMS starts with 'category' followed by a colon ':', followed by the three parameters of the category separated by semicolons ';'. When such SMS is received, the app should extract its values and place them in the correct fields.

Examples:

- category:Melbourne;20;TRUE
- category:Sydney;300;FALSE

4.4 Show a Toast message for any unrecognised or incomplete SMS received (missing parameters or invalid values). Also, the app should not crash if any invalid message is received.

Invalid message examples:

- category:Melbourne;
- category:Melbourne;20;Y

- category:Melbourne;**ABC**;TRUE

4.5 Upon successful save of a category, show a Toast message containing the generated category Id. e.g. "Category saved successfully: CME-1084".

## 5) New Event

5.1 The "New Event" activity should allow you to enter the details of the new event.

- UI elements to capture all attributes of the Event listed under table B above.
- Display generated Event Id after each successful save operation.

5.2 Add a new button to save details of an event to Shared Pref. For A1, only one event can be saved, no need to store multiple events in Shared Prefs. In other words, if you try saving another event after saving the first one, previous values will be overwritten.

5.3 Fill Event fields using incoming messages using the following format:

**event:Name;CategoryId;TicketsAvailable;IsActive**

The SMS starts with 'event' followed by a colon ':', followed by the four parameters of the event separated by a semicolon ';'. When such SMS is received, the app should extract its values and place them in the correct fields of the "New Event" Activity.

Examples:

- event:AFL Grand Final;CME-1084;20000;TRUE
- event:Melbourne Cup;CGT-1681;3000;FALSE

5.4 Show a Toast message for any unrecognised or incomplete SMS received (missing parameters or invalid values). Also, the app should not crash if any invalid message is received.

Invalid message examples:

- event:AFL Grand Final;20000;TRUE
- event:AFL Grand Final;CME-1084;20000;**TRU**

5.5 Upon successful save of an event, show a Toast with a message containing the generated event Id & category Id. e.g. "Event saved: EPM-10984 to CME-1084".

## Notes

- *For this assignment, only one credential (username and password) is to be managed*
- *To test 4 & 5, please use a new Virtual machine or uninstall all other apps deployed by you before testing the incoming SMS feature.*
- *If your app fails to build tutors will try to resolve bugs to an extent, and you will be penalised for*

*build failures (minimum -2 or no marks for features due to which the app fails to build).*

---

# Marking Rubric ☐☐

## Important Notes



### **Generative AI tools cannot be used in this assessment task**

*In this assessment, you must **not** use generative artificial intelligence (AI) to generate any materials or content in relation to the assessment task.*

- No marks if you do not **submit your code** to Moodle
- Google Drive submissions will **NOT** be considered
- No marks if you miss attending your interview
- No marks for any sections that you cannot explain during the interview
- If you are unable to answer most of the interview questions, your submission might be escalated to Academic Misconduct



## Where & What to submit? □

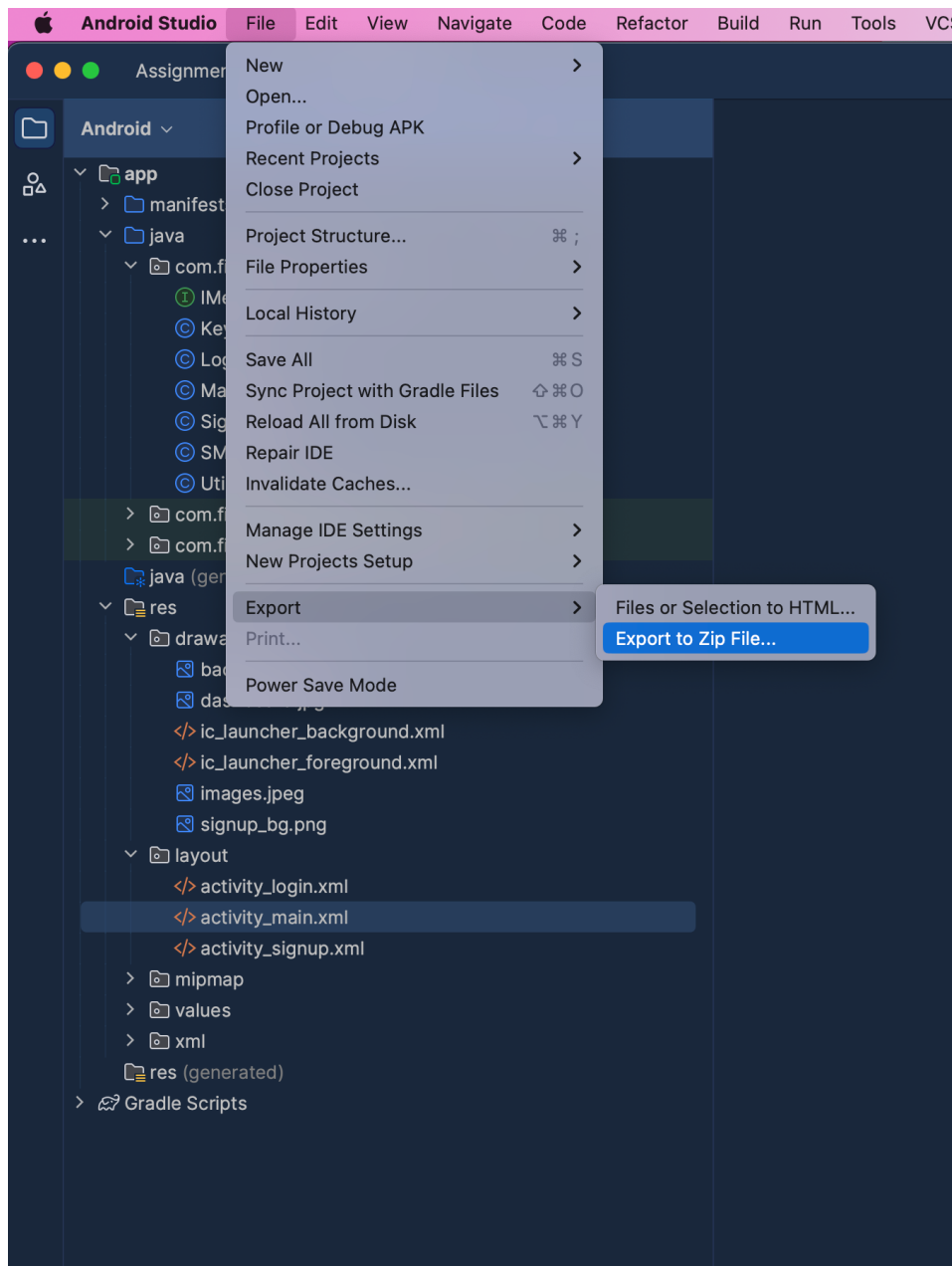
**For Moodle**, you must ZIP your files into a single file and upload it to the assignment-1 submission link that can be found on Moodle under Assessments.



If you directly zip your project folder without choosing the export option below, the generated file will exceed 20MB and Moodle will not accept files larger than 20 MB.

## How to ZIP your Android project?

From Android Studio > File > Export > Export to Zip File.



Save the exported file and upload it to Moodle

---

## Late Submission

Failure to submit your assignment on time will result in a **10% mark penalty for each day late** (including weekends) up to a maximum of 7 days late.

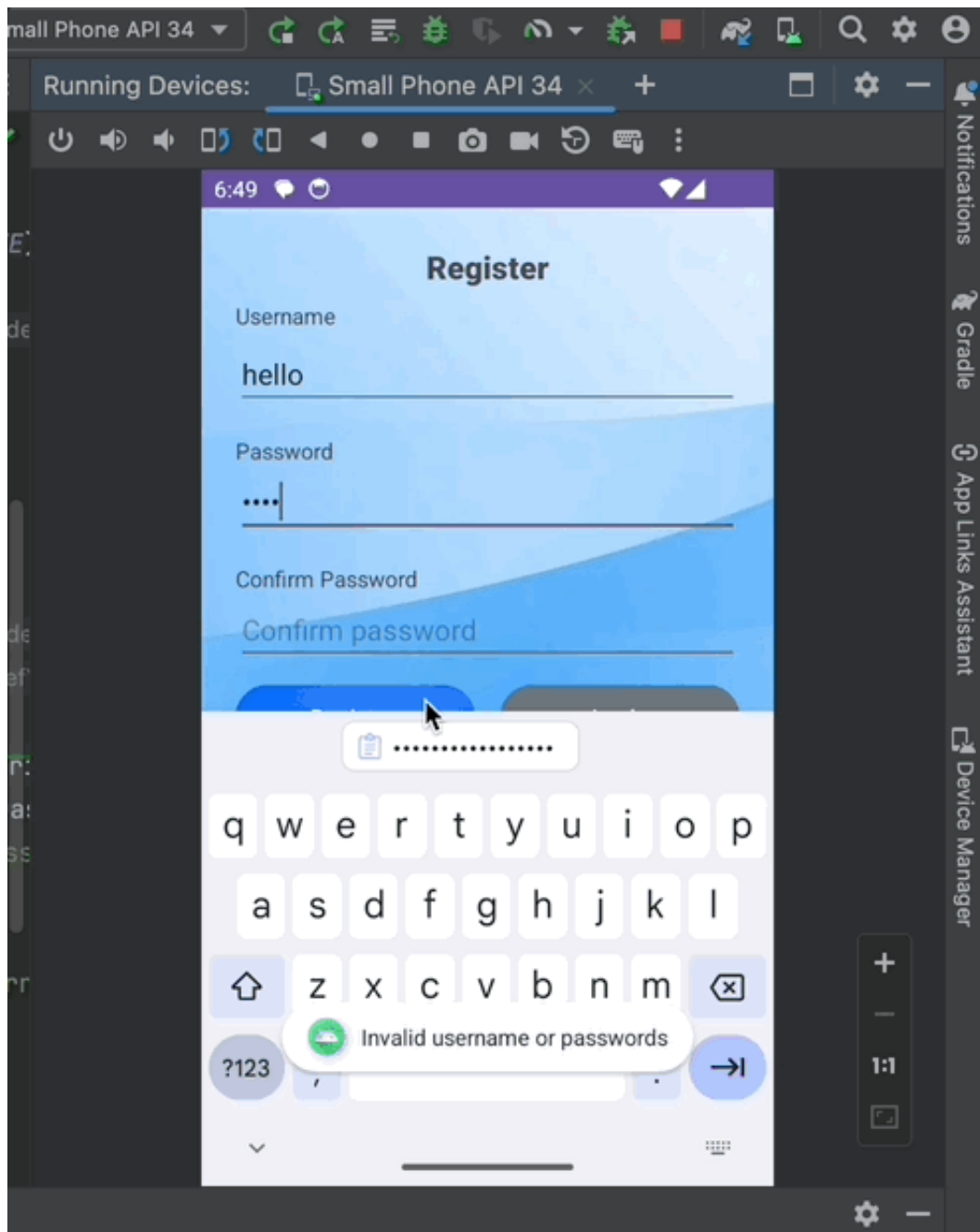
Submissions later than 7 days will receive a mark of 0.

---

Expected Output

Full video of expected output

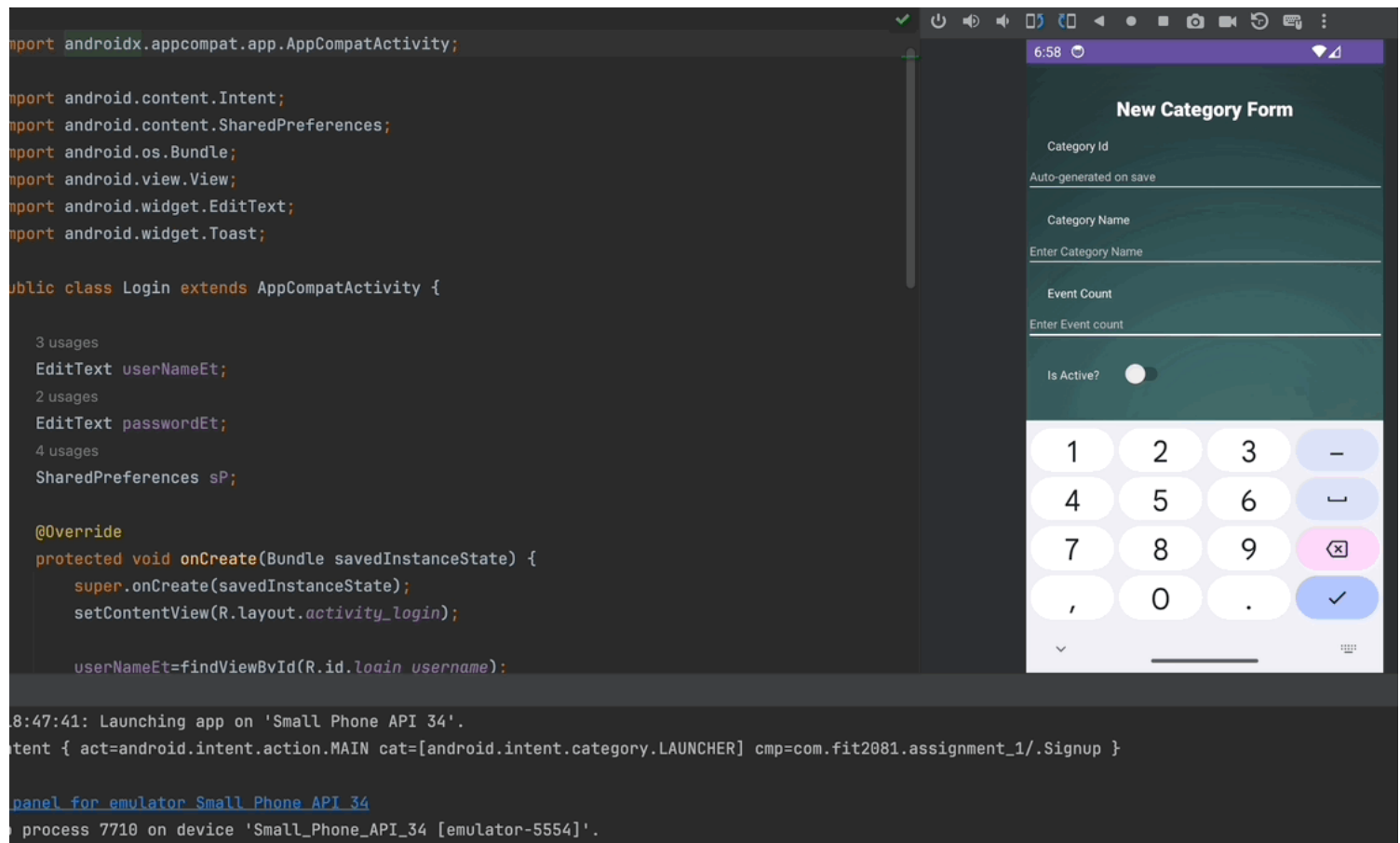
**Sign-up and Register**



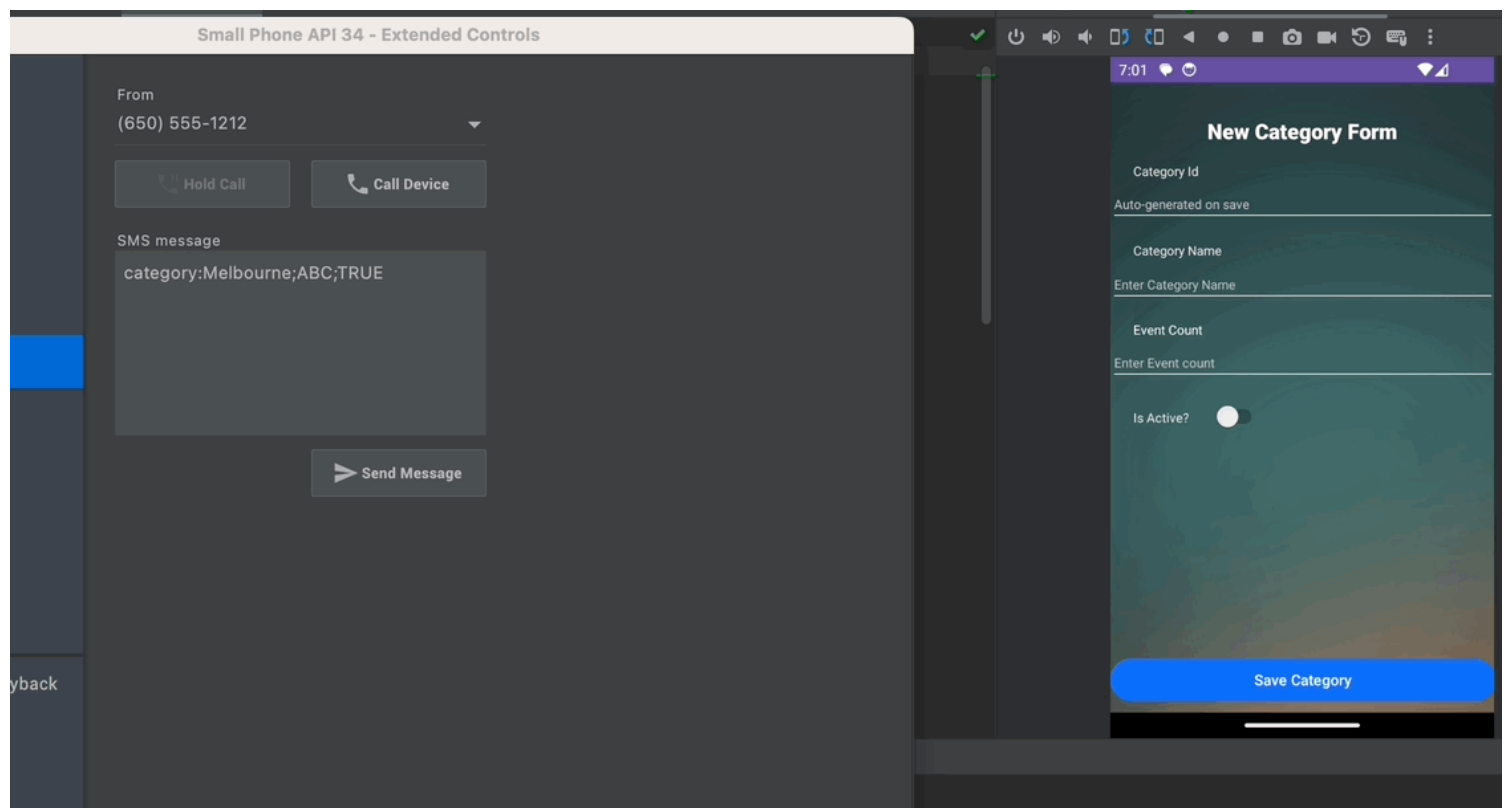
## Event Category UI



**Fill Event Category fields using incoming SMS messages**



## Handling invalid messages



## Adding events using incoming SMS

