

## COS30017 - Software Development for Mobile Devices

Formative Assignment - 05 (Graded as Pass / Fail, Individual Work)



### Objectives

This assignment task has the following objectives,

1. To incorporate what you have learnt throughout the semester
2. Demonstrating competency at using SQLite to perform CRUD functions
3. Applying material design by implementing floating action button
4. Explore location based services using GPS, implement background services and broadcast receiver
5. Communicating with remote server

### Core tasks

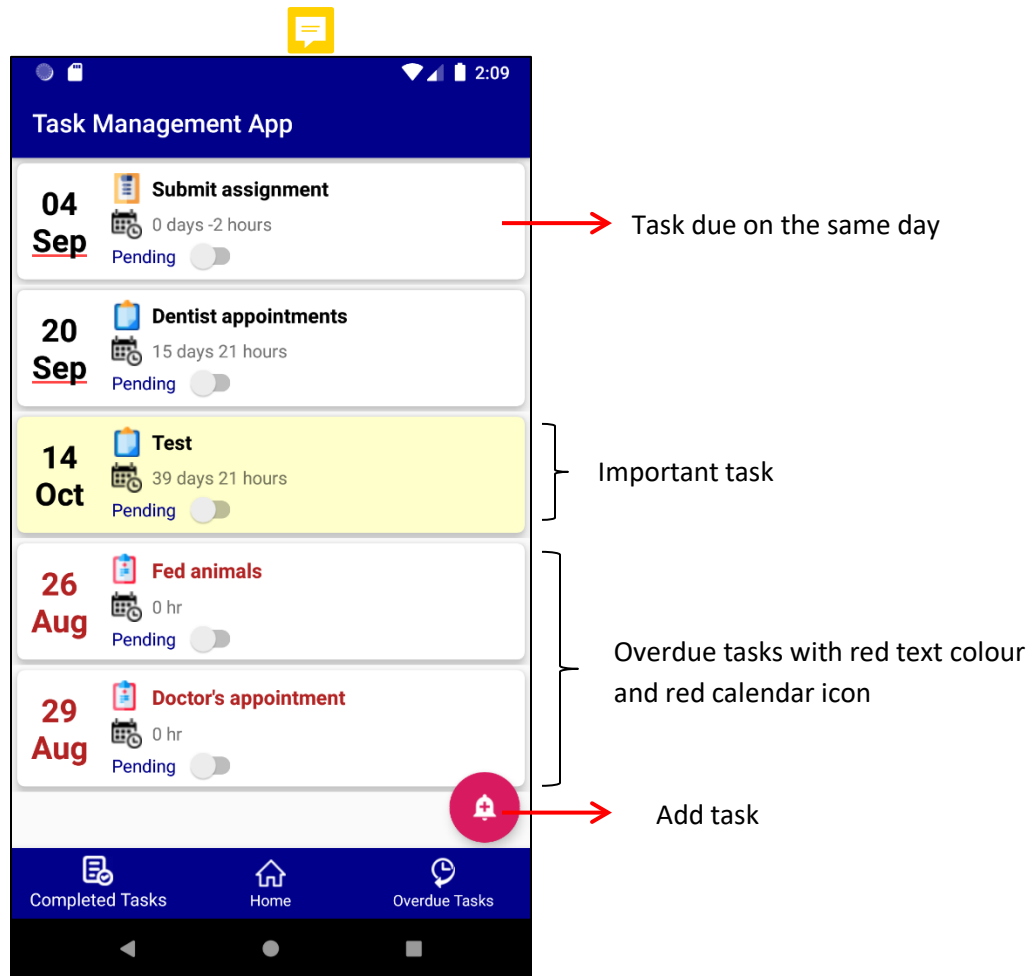
#### Task 1

Create a simple app to help users keep track of the daily tasks. It should make use of *MySQLite* database and have all the essential CRUD operations. The database should store the task's title, task's due date, task's details, task's priority and task's completion status. You may store other required fields as well.

1. The main screen should display the list of tasks on a *Recyclerview* and a *bottom menu* as shown below. Each row is contained in a *cardview*. The completed tasks **should not** be shown on the main screen. The list of tasks should be sorted in ascending order based on the due dates. However, the overdue tasks should be placed at the end of the list.

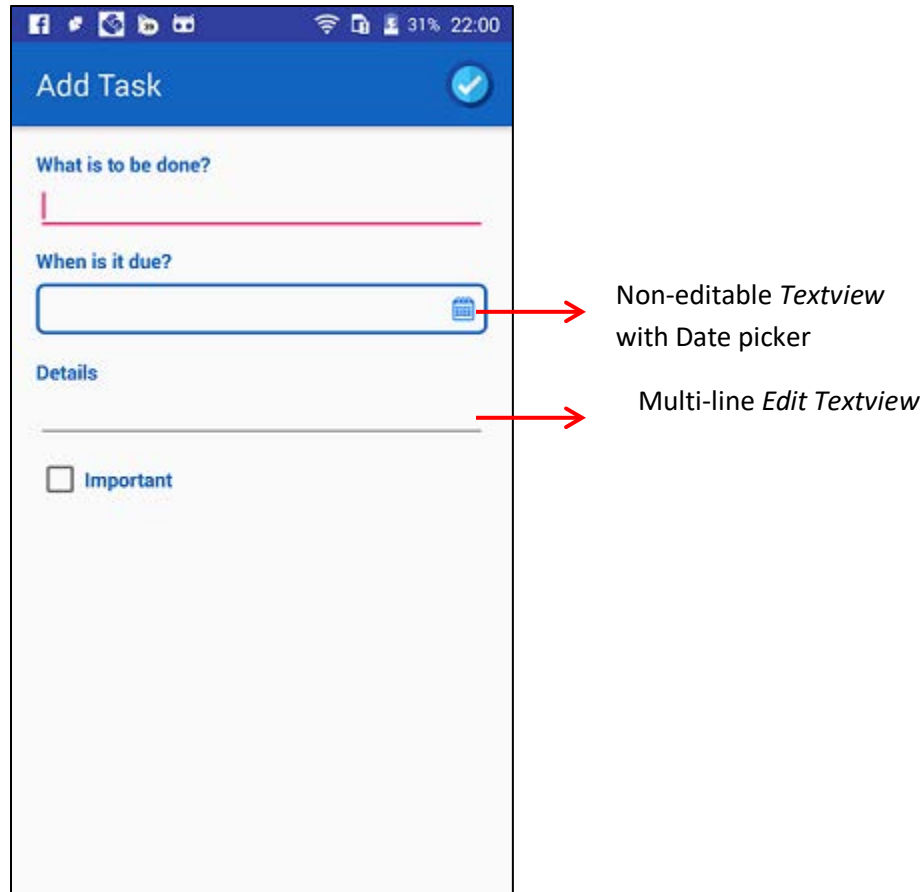
Different coloured icons should be used for the tasks: blue for completed tasks, orange for tasks that are due the same day and red as well as red text colour for the overdue tasks. For the important tasks, set the background colour to yellow.

You should also calculate and display the remaining time left for each task.



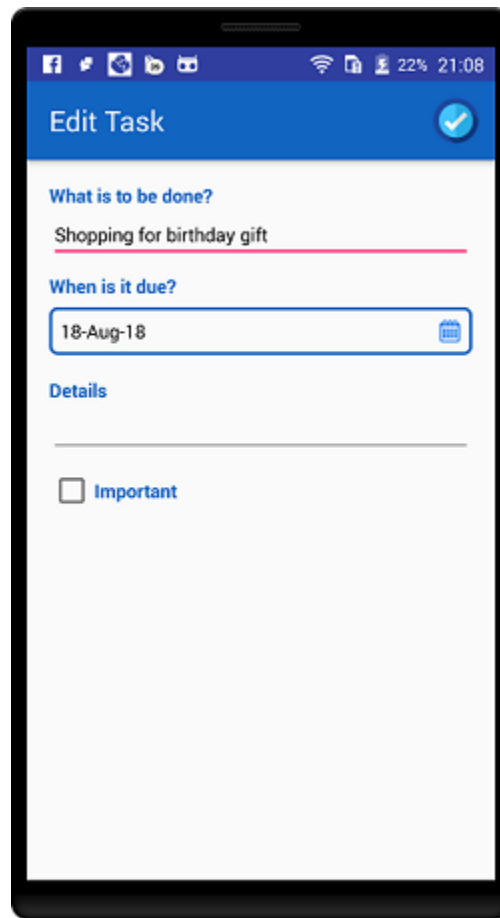
User may click on the switch button to indicate that the task is completed. When this happens, the task should disappear from the list.

2. To add a task, the user may click on the floating button. It should load the Add Task page as shown below.



The first two inputs are mandatory fields. Perform input checking to ensure that they filled in. Prompt the user with error messages if they are empty. You are also to ensure that the date selected is valid. The save button is located at the top right. Once it is clicked, the app should return to the main screen and the newly added task should appear on the list. At the same time, you are to keep track of the due date for each task and send notification to the user at a specific time before the due date. Make use of *Alarm manager* and *Broadcast receiver*.

3. To edit the task, the user may simply click on the task on the main screen. It will bring the user to the edit task page, which is similar to the add task page. Refer to the screenshot below:

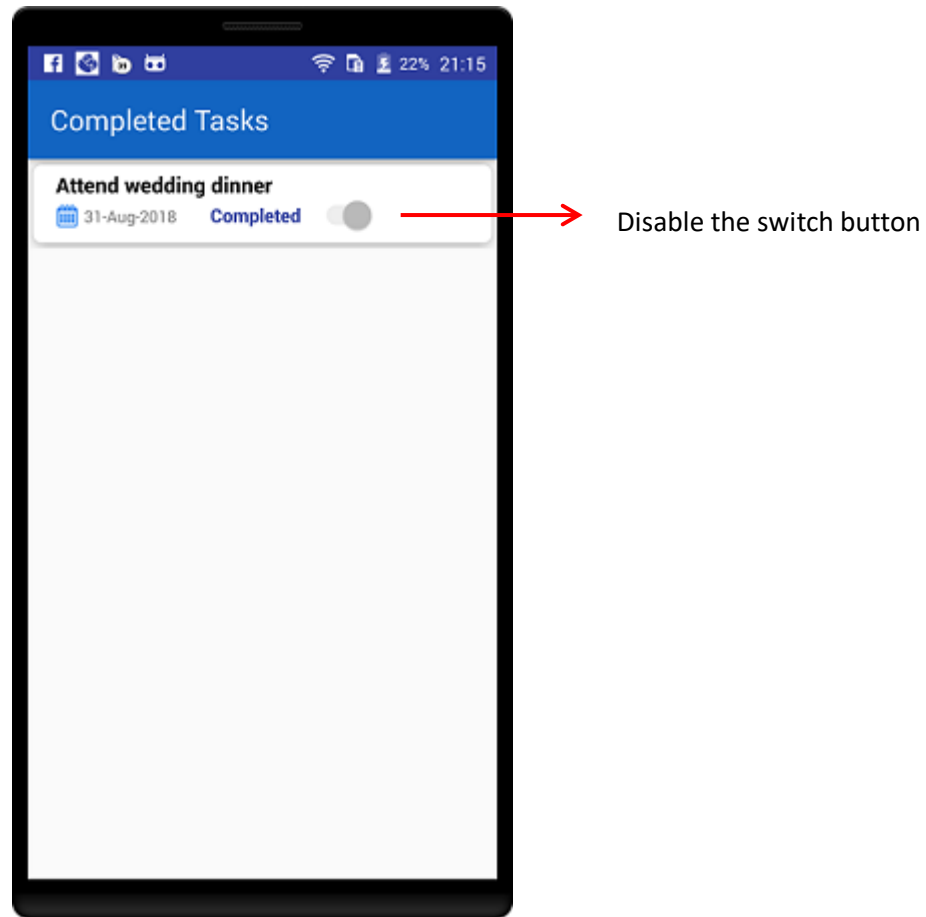


4. To delete a task, the user may simply click on the task for a long time from the *Recyclerview* on the main screen. Remember to confirm the operation by using alert dialog box before deleting the task.
5. User may view the completed and overdue tasks from the menu at the bottom of the main screen.



Back to the main page

For the completed task page, display the list of completed tasks and disable the switch button.

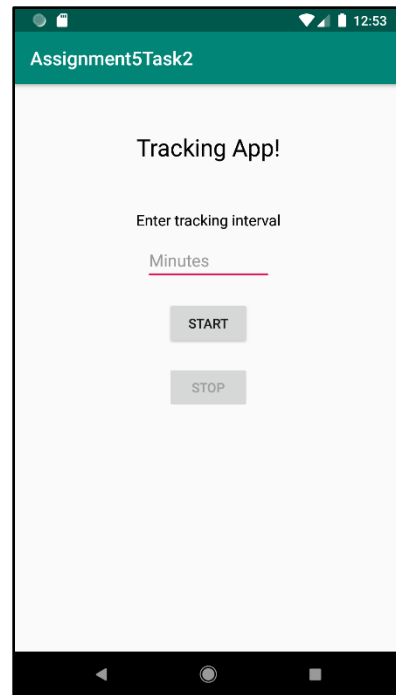


You are expected to use helper classes and adapter class in your implementation.

## Task 2

In this task you are to create a Tracker app that captures the geo-location of your device and post the locations to a server. Create a simple PHP script that resides on the server to receive HTTP post and write it to a text file. The webserver may reside on your local host.

Your *MainActivity* layout should contain an Edit Text View and two buttons as shown below. Disable the two buttons at start up and only enable the **Start** button when permission is granted by the user. When the user starts the tracker, enable the **Stop** button only, and disable the **Start** button.



Write the codes to continuously obtain the latitude and longitude of your device at the interval specified by the user. Post the geo-location after obtaining two coordinate points to the webserver created above.

You are to include the following components/services in your implementation:

- Runtime permission for dangerous permission to support device with API 23 and above
- A background service to obtain the geo-location (latitude and longitude) of the device at specific interval. Post the geo-locations to the server after obtaining two sets of coordinates.
- Create a simple server to receive the geo-locations and save it to a text file. Include the date and time as shown below.

```
2019-10-20 13:06:43 Latitude: 13.58, Longitude: 129.0  
2019-10-20 13:07:44 Latitude: 13.573998333333334, Longitude: 129.5
```

Figure 6

## Core/Extension Tasks

All tasks in this assignment are “core”. You must complete all core tasks, submit for feedback, and achieve a pass for all tasks in order to be eligible for a pass grade in this unit.

## Submission

- i. You are required to submit a printed report:
  - With the assignment cover page
  - The document must have a title (e.g. Submission for Assignment 05)
  - Evidence that shows you completed each task.
  - The document does NOT need a table of contents.
  - Make sure that your codes are properly aligned in your report. Format it properly to ensure that it is readable.
- ii. You are required to submit a softcopy of your report through the Blackboard.

The report is assessed and returned to you in the lab with feedback. You are expected to incorporate the feedback (esp. if changes are required) and submit the changed reports as part of the final portfolio.

Note: This is a formative assignment. That is, an assignment designed to provide feedback. If you fail this assignment, you have **1 week** to make corrections and resubmit to pass. Late submission will reflect on your performance. Any submission after 1 week of the due date will **NOT** be entertained.

## Breach of Academic Integrity

Cases of improper academic integrity includes plagiarism (re-producing in whole or substantial part of the codes/report from book, or the internet) and cheating (copying from your friends). Violation of academic integrity will have its consequences depending on the severity. A repeat offence could lead to a fail in this unit.

## Demonstration

You may be asked to demonstrate your assignment in the lab or during the signed-off sessions. You should be able to do this and explain your code when asked. Failure to do so will have an adverse effect on your performance.

## **FAQ**

### **What happens if a student is unable to submit the assignment?**

If you are unable to submit due to medical reasons, then a doctor's certificate will have to be shown. In exceptional circumstances, an email submission is permitted (with prior agreement with convenor). In normal conditions, all students are expected to make a submission by the due date, else the assignment is graded as a fail.

### **What happens if assignment submission is graded as a 'fail'?**

You will have to repeat the task and submit in the following weeks lab session. Students can repeat the task and submit for feedback up to **twice**. If your submission is graded as 'fail' twice then you may fail this unit.