

Assignment 1 – Reminder App

Total: 5 marks

Due date: 8pm 30th August 2021 (Week 04)

Submission requirements:

1. **Submit all project files as one zipped file.**
2. **A video with your app walkthrough (screen capture & voice recording) demonstrating the actions/features/functionalities that are required to be developed.**

In this assignment, you need to extend ToDoList app you started in Tutorials to a Reminder app which contains at least two views.

1. The **Main** view should contain:

- A **ListView** which displays all the saved tasks. Each task consists of a title and the remaining time (in days, hours and mins e.g. 1 day 7 hours 15 mins) to complete the task. Clicking a task will switch to the **“Edit/Add Task”** view. [0.25 mark]
- An **“ADD NEW”** button. Once this button is clicked, the app will switch to the **“Edit/Add Task”** view. [0.25 mark]

2. The **“Edit/Add Task”** view should contain:

- A **“Text field”** which allows user to type or edit the title of a task to add or update the ListView. [0.25 mark]
- A **“Date/Time Picker”** that allows user to set or update the due date/time of a task. [0.5 mark]
- A **“Save”** button used for adding new, or updating the title and due date/time of a task in the ListView:
 - If adding a new task item, capture both the title and due date/time of the task. To calculate the remaining time to complete a task, take the time difference between the due date/time and the current system date/time. [0.5 mark]
- A **“Cancel”** button next to the **“Save”** button, used to close the Activity without updating the task. When this button is clicked, the app will pop up a dialog that asks user: “Are you sure to cancel this edit? Your unsaved edit will be discarded if you click YES”. [0.25 mark]

3. Your app should also be able to handle the following data persistence tasks:

- Every time user launches this app, the app loads the list of tasks from the local Database. [0.5 mark]
- The list of tasks should be displayed and sorted based on the remaining time to complete a task i.e. the most urgent task with the least amount of remaining time should be shown on top of the list. If a task is overdue, it should display a text “**OVERDUE**” next to the task title instead of showing the remaining time. [0.5 mark]
- When clicking the “**Save**” button in the “**Edit/Add Task**” view, the app should add or update the task in both the ListView and local Database. [0.25 mark]
- Add a long click event to delete a task from the ListView. When user tries to delete the selected task, the app will pop up a message that asks user: “Do you want to delete this task?” If the user clicks “**YES**”, this task will be deleted from both the ListView and local Database. [0.25 mark]

4. App should be able to be built and run successfully on emulator. [0.5 mark]

5. Coding [1 mark]:

- [0.5 mark] Compliant with the official Android Java code style guidelines (<https://source.android.com/setup/contribute/code-style>), use proper indentation, stick within the 100 characters length for each line of code, no trailing whitespaces and no unused imports.
- [0.5 mark] Documentation: use proper in-line code commenting, and Javadoc standard comments for new public classes, methods, variables and constants.