Assignment 1 – Shopping List App

Total: 5 marks
Due date: 5pm 22nd August 2022 (Week 04)
Submission requirements:

1. **Submit all project files as one zipped file.**
2. **You will demonstrate your app to your tutor during the tutorial time for CC classes or on pre-scheduled time for RE classes.**

In this assignment, you need to extend ToDoList app you started in Tutorials to a Shopping List app which contains items you plan to buy on your selected shopping day.

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

- A **ListView** which displays all the shopping items. Each item consists of an item name and number or amount of items and the time remaining for shopping (in days and hours. e.g., 5 Days 3 Hrs) as an information to the user. If the planned shopping date is already passed you should display "OVERDUE" instead of showing the remaining time. [0.5 mark]

- Additionally, each item should consist of a **checkbox** to tick once the item is purchased. [0.5 mark]

- An "**ADD NEW**" button. Once this button is clicked, the app will switch to the "**Edit/Add Item"** view to add a new item. To update an item, user should click on the item in the list and the app should again switch to "**Edit/Add Item"** view. [0.25 mark]

- Add a long click event to delete an item from the ListView. When user tries to delete the selected item, the app will pop up a message that asks user: "Have you already purchased this item?" If the user clicks "YES", this item will be deleted from both the ListView and local Database. [0.25 mark]

2. The **"Edit/Add Item"** view should contain:

- A **"Text field"** which allows user to type or edit the name of an item to add or update the ListView. [0.25 mark]

- A **"Text field"** which allows user to type or edit the number or amount of items 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 the item to be purchased. [0.5 mark]

- A **“Save”** button used for adding new, or updating the item name, item amount and due date/time of an item in the ListView:
  - If adding a new shopping item, capture both the item name, amount, and due date/time for shopping. To calculate the remaining time to start shopping, take the time difference between the planned shopping 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 item. 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 items and **sorted based on the time remaining** for shopping from the local Database. [0.25 mark]

- The list of items already purchased (with ticked checkbox) should be displayed at the bottom of the list. [0.25 mark]

- When clicking the “Save” button in the “Edit/Add Task” view, the app should add or update the item information and shopping time information in both the MainView and local Database. [0.25 mark]

4. Demonstration

- Demonstrate that the app can be built and run successfully on emulator. This should be the app version submitted to the Canvas. [0.5 marks]
- 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]

Note: Marks for successful implementation of each of the above app features will consider how well you demonstrate the feature and answer questions from the assessor on how you developed those features.