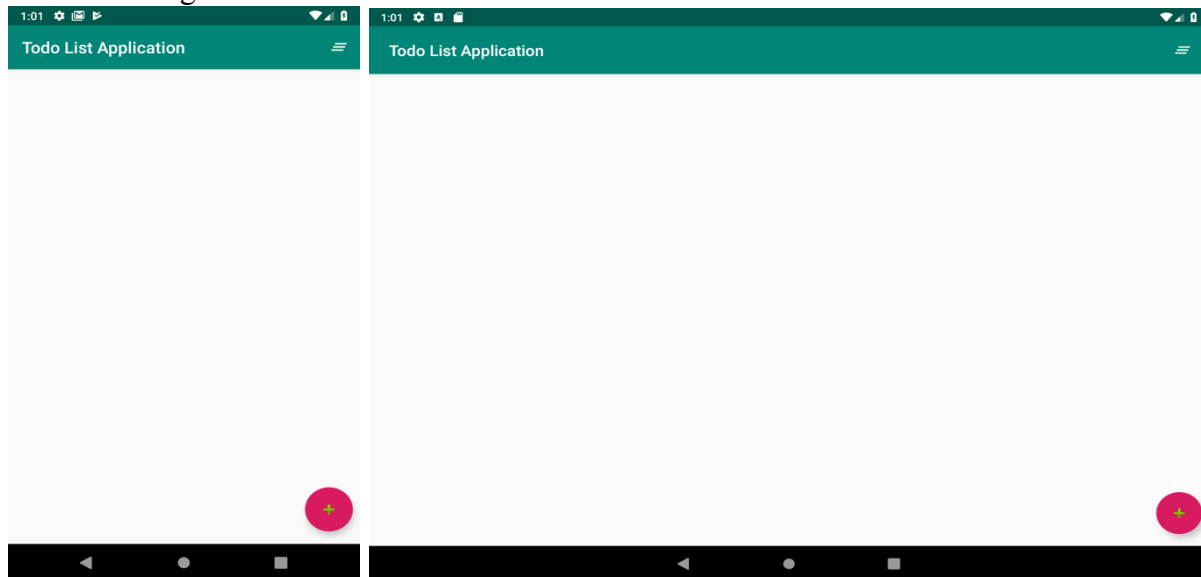# HOMEWORK 2

In this homework, you will create a ToDo manager application. The application creates and manages Todo List Items. You will design this application's UI, including its layout and resource files. You will also need to implement the application's features by yourself. The application should be able to run in both **tablet and phone** and function the same way as described in the **demo video** and the document.

Part A: A basic ToDo manager application

The first time the application runs it will have no ToDo Items and therefore its initial UI will look something like this:
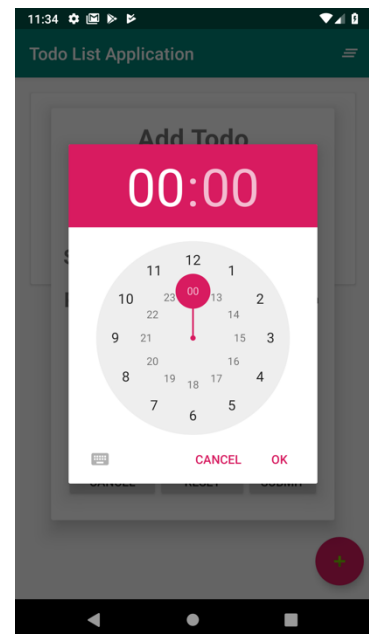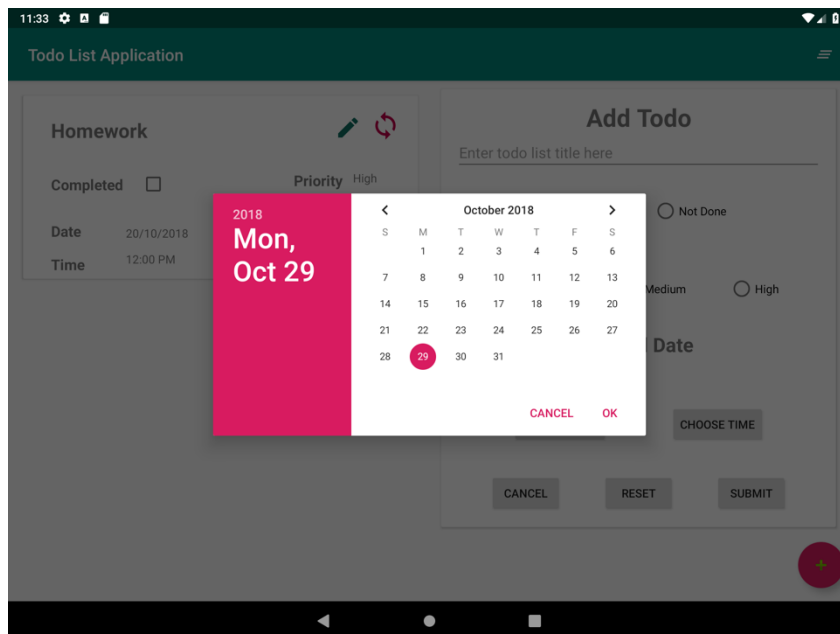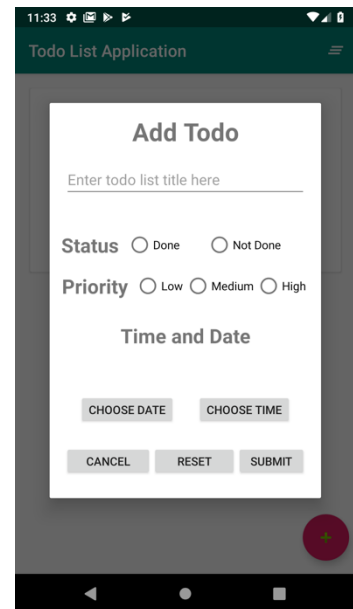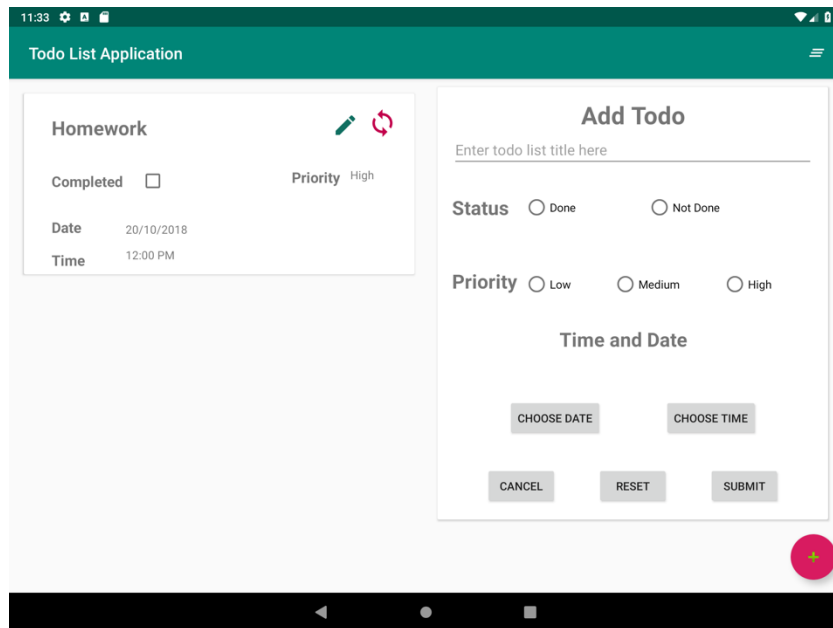

This UI contains a **RecyclerView** for displaying existing Todo Items. As ToDo Items are created, they will be added to this **RecyclerView**. Also, the recycler view is **inside a fragment**.

Initially, the application can either start with an empty list or you can create some dummy ToDo Items. As you work through the Homework, you'll make ToDo Items persist across sessions. The **MainActivity**, should always display a special **plus icon**. When the user clicks on the icon you should do the following.

1. **On a tablet** it should inject **another fragment on the right side** that allows the users to enter the ToDo item info
2. On the Phone you should **pop up a dialog fragment** that allows the user to enter the information of the ToDo Item.
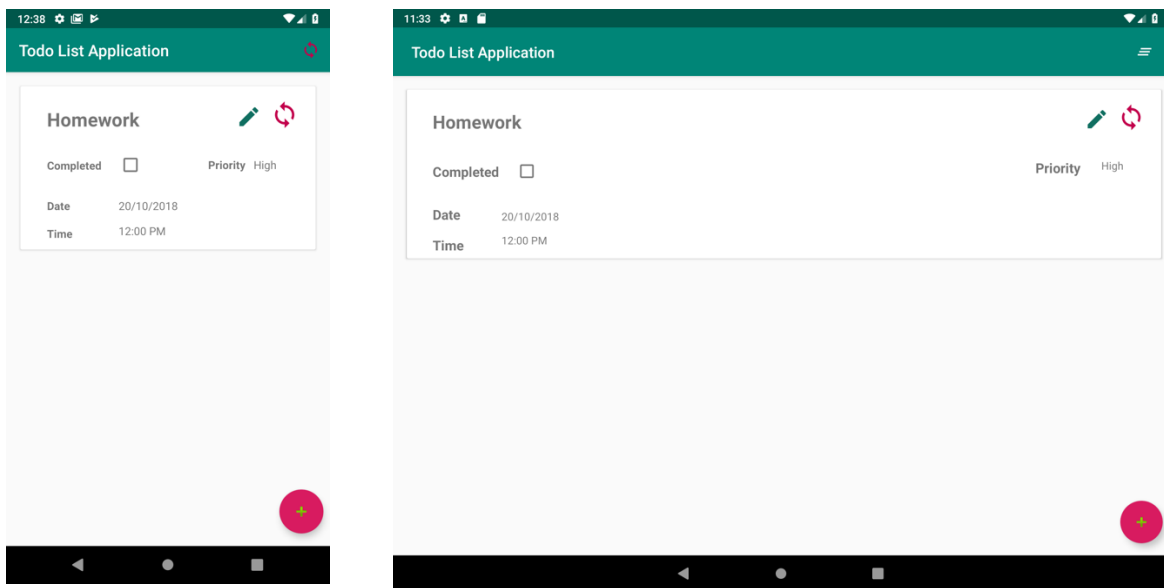
Below are the figures for both cases.

NOTE: **You are free to define your own user interface, but it must include the fields shown in the figure above**. Specifically, **ToDo items** should have the following fields:

1. Title: A user-provided String
2. Status: {Done, Not Done}
3. Priority: {Low,Med,High}
4. Time & Date: A deadline for completing the underlying ToDo Item.   The "Add New

ToDo Item" Activity's UI should also include buttons:

5. Cancel – finish the Activity without creating a new ToDo Item.
6. Reset –Resets the fields of the Todo Item to their default values.
7. Submit – Create a new Todo Item with the user selected data fields & return to the main Activity. When the application returns to the main Activity, the new ToDo Item should appear in the main Activity's RecyclerView. For example, if the user adds a new ToDo Item to an empty ToDo list, as shown below, then the main Activity's RecyclerView should update to display the new ToDo Item.
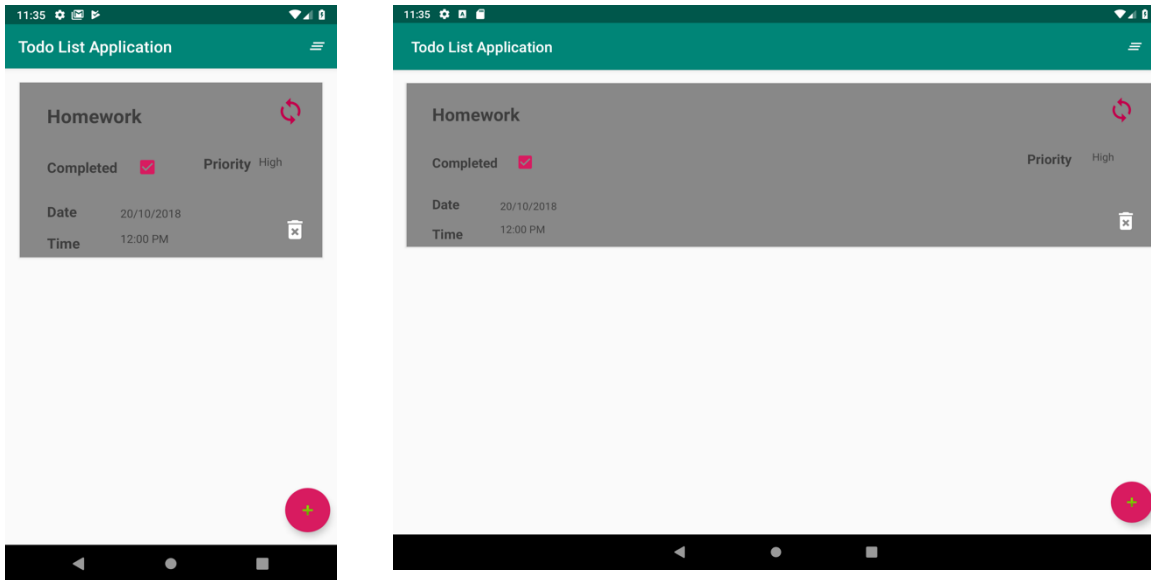
To make Todo Items persist across sessions you can use an ArrayList, you can write Todo items to a file and read existing Todo items from the file on application startup. You can also persist using SharedPreferences or SQL database.

Back in the Main Activity, the user should be able to toggle the Done checkbox to indicate that the ToDo Item is Done or Not Done.
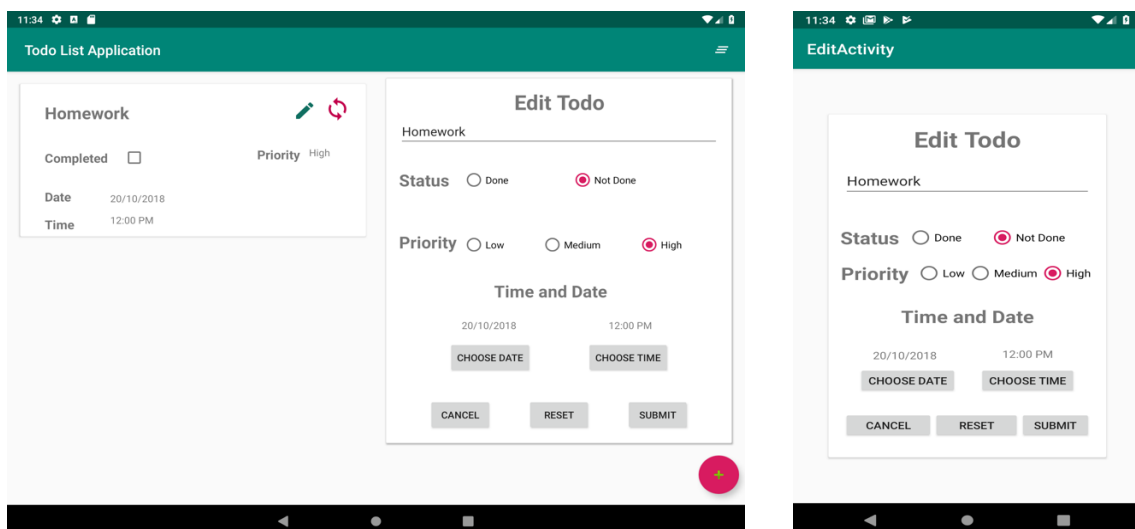
Part B: An improved ToDo manager application

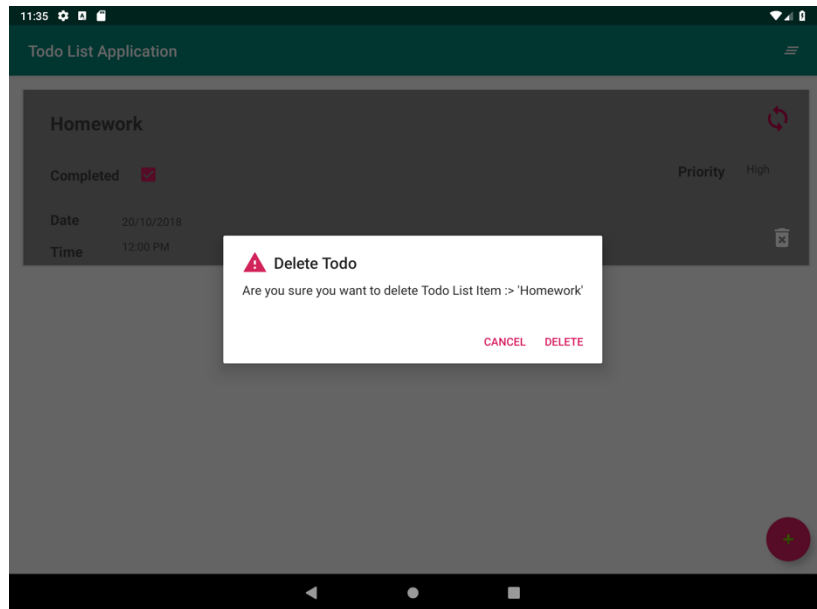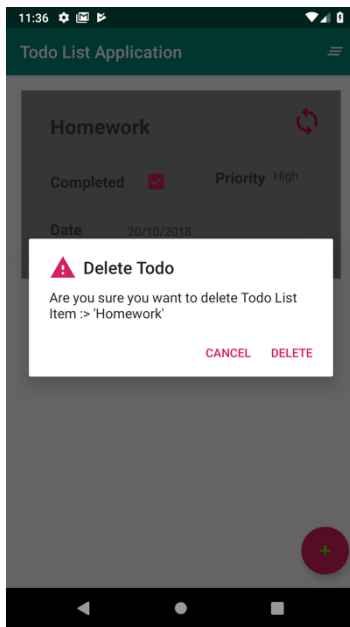In this part, modify your application so that,

- **ToDo Items** that are Not Done are displayed in the Main Activity with a different colored background than those that are Done. In addition, when the user toggles the Done checkbox, the background color should change as appropriate, the edit icon should hide and the delete button should show.
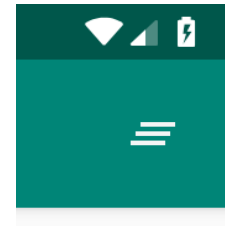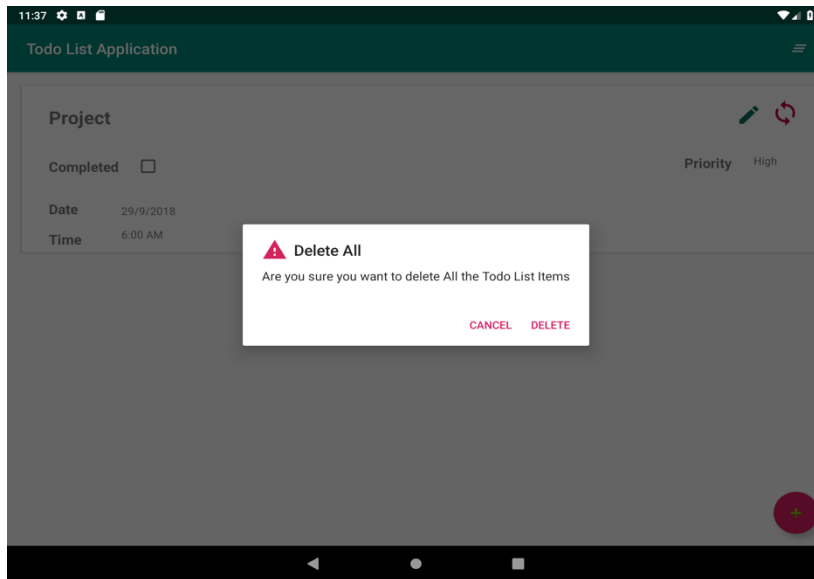

- If the user pressed on the edit button, your application should be able to do the following. For the tablet, it should open a side fragment and populate the data of that todo list item. On the phone, it should open **another activity** that has a single fragment and populate the information. See the two images below.
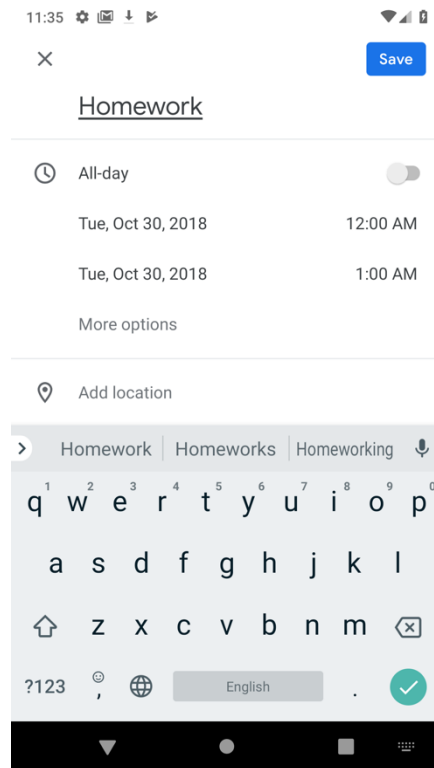
- If the user presses on the delete button, show a warning message before deleting that todo list.

- When the user clicks on the clear all menu icon, it should show a dialog warning the user if they are sure to delete. If they press yes delete all the todos. Otherwise, dismiss the dialog without deleting anything. The below image shows on a tablet, but it should have the same behavior for the phone too.

- Finally, if the user presses on the sync button, it should make a calendar entry of the todo list item on Google calendar or any other calendar configured on the user's device.

Deliverables:

1. Submit your **project code** via the BB as a zip file. Make sure you remove the APK file by cleaning your project before submission. Otherwise, the file size will be huge.
2. Submit the **Testing Sheet [Word file]** with all the screenshots. Do not convert it to PDF.
3. You will be required to demonstrate your work in the Lab so be prepared to answer all questions related to your work.