Documentation

gitHub with most updated information including this document:
https://github.com/Jsingh11/dueIT

Breakdown:

1. Main Activity

    - The main activity holds our "home screen." On this home screen we have three buttons that will take you to the different ways in which you can see your reminders
    - These three buttons are "daily tasks", "weekly tasks", "monthly tasks"
    - Also at the bottom there are more image buttons that have not been implemented yet but will be

2. Task Screens

    - The task screens are all the same, the only thing that will differentiate them is what tasks are shown in each screeen
    - This is not 100% complete right now but the task screen will include a list view that will display our tasks to the user
    - If you are in the "daily tasks" view, only tasks that are due that day will be shown, all other tasks will be hidden
    - Right now all of the task screens only have a back button and a text view that states what activity they are

3. Floating Action Button

    - The floating action button in the bottom right corner and this will be where users will click to add reminders.
    - When this button is pressed users are taken to a screen where they are asked to type in their reminder and also assign a priority to it
    - Right now the submit and creation of the reminder is undone but will be done by the next sprint

How it works:

For this first sprint Jose took the UI that Dhruval designed and made it actually function. Dhruval had buttons and image buttons placed and Jose made it so that when most of the buttons were pressed you would actually have something happen like in a real app.

The three main buttons all take you to seperate pages and the floating action button also takes you to a page where the user will input the reminder.

The way this works is by using Intents. Each button has an onClick() listener that when pressed sends the user to the next activity. In the next activity there is also a back button that will take the user back to the mainActivity which is our main screen.

This is how that is implemented code wise:

```
1
2  protected void onCreate(Bundle savedInstanceState) {
3          super.onCreate(savedInstanceState);
4          setContentView(R.layout.activity_main);
5
6          // by ID we can use each component which id is assign in xml f
7          // use findViewById() to get the Button
8          Button dailyTaskButton = (Button) findViewById(R.id.dailyTaskB
9
10         // Add_button add clicklistener
11         dailyTaskButton.setOnClickListener(new View.OnClickListener()
12
13             public void onClick(View v)
14             {
15
16                 // Intents are objects of the android.content.Intent t
17                 // to the Android system defining the components you a
18                 // Intent to start an activity called SecondActivity w
19
20                 Intent intent = new Intent(MainActivity.this, dailyAct
21
22                 // start the activity connect to the specified class
23                 startActivity(intent);
24             }
25         });
26     }
27 }
```

This code was for sending users from the 'home screen' to the 'daily task' screen via the 'daily task' button.

For all of the other screens the same logic was followed. The only difference is when we encounter an image button, this occurs with the FAB, since it is an image in our app. This is how it is implemented then, the same logic just with an imageButton

```
1
2  // by ID we can use each component which id is assign in xml file
3          // use findViewById() to get the Button
4          ImageButton fab = (ImageButton) findViewById(R.id.fab);
5
6
7          // Add_button add clicklistener
8
9          fab.setOnClickListener(new View.OnClickListener() {
10
11             public void onClick(View v)
12             {
13
14                 // Intents are objects of the android.content.Intent t
15                 // to the Android system defining the components you a
16                 // Intent to start an activity called SecondActivity w
17
18                 Intent intent = new Intent(MainActivity.this, fabActiv
19
20                 // start the activity connect to the specified class
21                 startActivity(intent);
22             }
23         });
24
```

So once a user is at the screen that we want them to be at, they must go back. In our case we used a simple back button that links our users back to the main activity. This same back button logic is used for every single page.

```
1
2  protected void onCreate(Bundle savedInstanceState) {
3          super.onCreate(savedInstanceState);
4          setContentView(R.layout.activity_daily);
5
6
7          // by ID we can use each component which id is assign in xml f
8          // use findViewById() to get the Button
9          Button backButton = (Button) findViewById(R.id.backButton);
10
11         // Add_button add clicklistener
12         backButton.setOnClickListener(new View.OnClickListener() {
13
14             public void onClick(View v)
15             {
16
17                 // Intents are objects of the android.content.Intent t
18                 // to the Android system defining the components you a
19                 // Intent to start an activity called SecondActivity w
20
21                 Intent intent = new Intent(dailyActivity.this, MainAct
22
23                 // start the activity connect to the specified class
24                 startActivity(intent);
25             }
26         });
27
28     }
29 }
```

In total we have 5 activities, and 5 .java files because at the moment we have 5 'pages' the user can go to.

Things that are still left to do:

- FAB activity needs a submit button and a way for it to take in user reminders as well with the other information that a reminder contains. In our case, the due date, priority level, and of course the reminder
- All of the task pages must be filled with a list view that will display the correct reminders. This would probably involve some sort of logic like:

```
1
2  if (reminder.date == today){
3  dailyTask_listView.add(reminder)
4  }
5
6  if (reminder.date == thisWeek){
7  weeklyTask_listView.add(reminder)
8  }
9
10 if (reminder.date == thisMonth){
11 monthlyTask_listView.add(reminder)
12 }
```

- The bottom image buttons need to have their own activity pages and those must be filled with their functions

Jose Diaz