Introduction:

This Android weather app is designed to be convenient and informative to the user. The app includes information about the weather in the user’s current location, as well as the weather in other locations. The app also includes a feature that allows the user to save their background preference for the app. The app is designed to be simple and easy to use, and is intended to be used by anyone who wants to know the weather in their current location or in other locations.

Design Rationale:

Coursework Requirements:

- There are multiple screens, and users can navigate between screens

- Use both explicit and implicit intents

- Has Menus

- Use RecyclerView

- Use Data storage (e.g., SharedPreference, database, files)

- Use Internet (e.g., fetch images from Internet to update views)

- The app opens and the screens display properly when the device has no connection to the Internet

I based the design of the app around these requirements.

The app is designed to be simple and easy to use, this is so that the user can easily access the information they want. The design code for the application is in the res/layout folder. These are the designs for the different activities and other aspects of the activity such as the weather card and toolbar. There are 2 activities in the app, the main activity and the more information activity. The main activity is the activity that is displayed when the app is initially opened. The main activity displays the weather information for the user's current location and the user is then able to search for other locations. The more information activity is the activity where I have linked the website for the API I used to get the weather information, which is where I implemented the implicit intent. In the more information activity, there is a short description of the app within a card view and also a back button which takes the user back to the main activity, this is where I implemented the explicit intent. For these activities, I decided to use relative layouts as they allow you to arrange UI elements relative to each other. Initially I was going to add a separate activity for changing backgrounds but then decided to use a submenu instead and have the SharedPreferences apply when the user clicks on one of the background option. I also have another submenu which has pre-set cities, such as Exeter, London and Shanghai. When one of these are clicked, it automatically updates the search query and the weather information for that city is displayed. I decided to use a submenu for this as I think it wouldn't look as professional if it was on a separate activity.

Novel Features:

A feature that I have is the more information activity as it has a link to the API that I have used for this project and the website can also be used to check the weather information. This website could then be used when the user is on other devices and use my weather application when using a mobile device.

Challenges:

One of the challenges that I faced was the SharedPreferences wouldn't save and would reset to the default background when the app was closed. I solved this by doing research and trying multiple ways of implementing the SharedPreferences, and eventually I found a way that worked. This was by adding the applySavedPrefs() method to all the activities rather than just having it in main activity.

Another challenge that I faced was that the most of the menu and submenu would appear off the screen, which would therefore make the menu object unusable by the user. This was overcome by adding the following code to the onCreate method in main activity:

getWindow().setFlags(WindowManager.LayoutParams.FLAG\_LAYOUT\_NO\_LIMITS,

WindowManager.LayoutParams.FLAG\_LAYOUT\_IN\_SCREEN);

This code allows the menu to be displayed on the screen and solved that issue.

There was another challenge that I faced, which was not being able to get the pre-set cities to work. At the time, I decided not to implement the pre-set cities feature into the app and focused on perfecting the other features of the app. Closer to the deadline, I decided to try and implement the pre-set cities feature once again and managed to fix the issue and is now a working feature.

Future App Developments:

In the future, I would like to implement the following list of features:

- Users can change the units of the weather information

- Users can add their own pre-set cities

- Background of the main activity changes depending on the weather condition

- Future weather information is displayed

- More specific weather information is displayed such as humidity, feelsLike, etc.

I would like to continue on this project and improve the app as much as possible after the deadline.

Designed and Developed by:

Kevin Liu