Introduction

This experience Report is based on the Design Communication Perspective. This involves communicating the design of the model through diagrams and words.

The Application I am designing is a Motorcycle Companion App that you will be able to view weather, track fuel consumption and view a petrol station map.

The Major Feature will be the ability to add your Motorcycles to the App and the Add Fuel Up information to record fuel Consumption and determine how long your Petrol tank will last as most motorcycles do not have fuel gauges like cars. Newer riders will run out of fuel in inopportune places. This app is designed to help all motorcyclists.

With the app you will be able to view weather conditions of the area to determine if it will be ideal conditions to go for a leisurely ride.

Also the App has Google Maps integration so the user can see where all the petrol stations are in relations to their position as well as display all the previous petrol stations they have been to with a label stating the fuel efficient they got at the pump there.

Class Diagram

This is the general Layout for the Class Make Up for the Motorcycle Companion Application. There will only be one activity, the Main Activity which contains all the Fragments and the Action Bar. The Fragments are all accessible through the Action bar with Weather, Maps, Table and Fuel Stats available through the Spinner on the action bar and the add bike and add fuel up through the context menu or on the action bar if enough room.
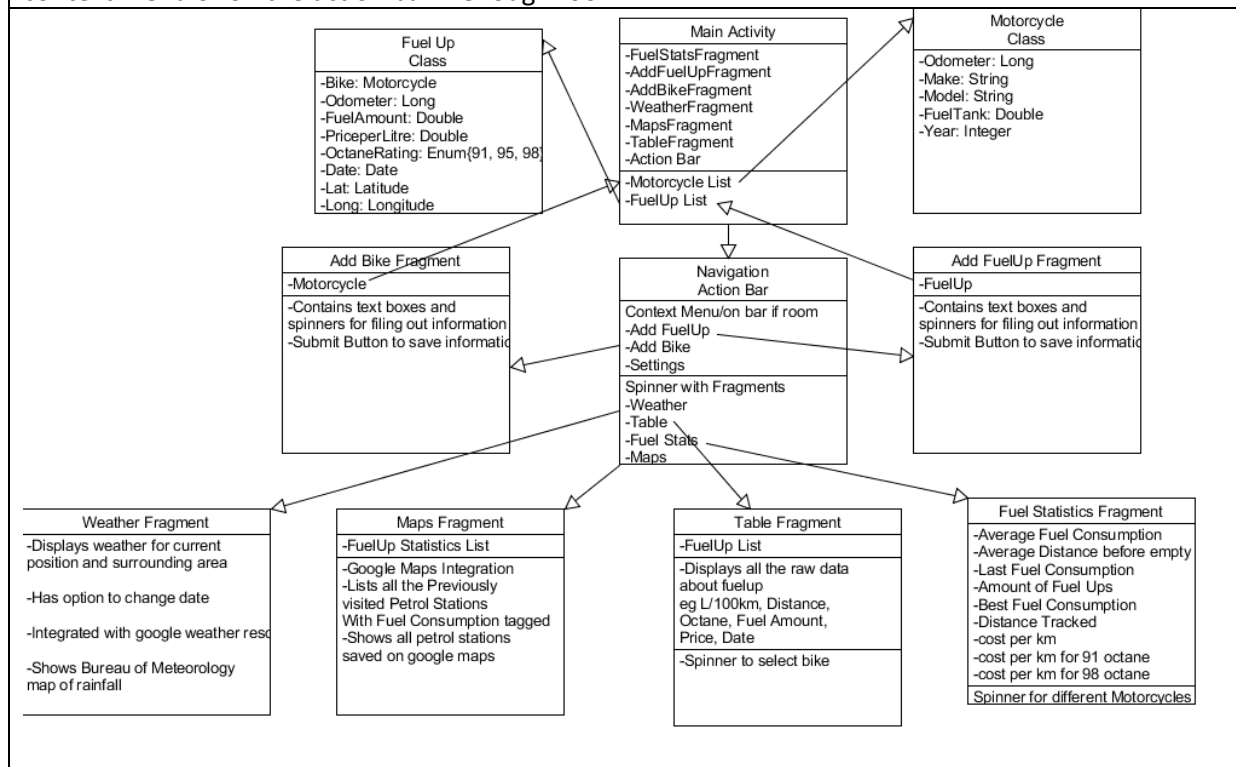
Navigation Model

The main tabs such as Fuel Tracking, Table, Weather and Maps are accessible from the drop down spinner on the Action bar. To add Motorcycles and Add fuel Ups you can do that from the context menu on the Action Bar. This Streamlines the Actions and Tabs by containing them on the Action bar that is needed anyway instead of leaving these actions all over the fragments themselves allowing the user to navigate with simplicity and ease. All tabs and pages data will be stored as a fragment and you can go back to finish filling out information while still in the app.
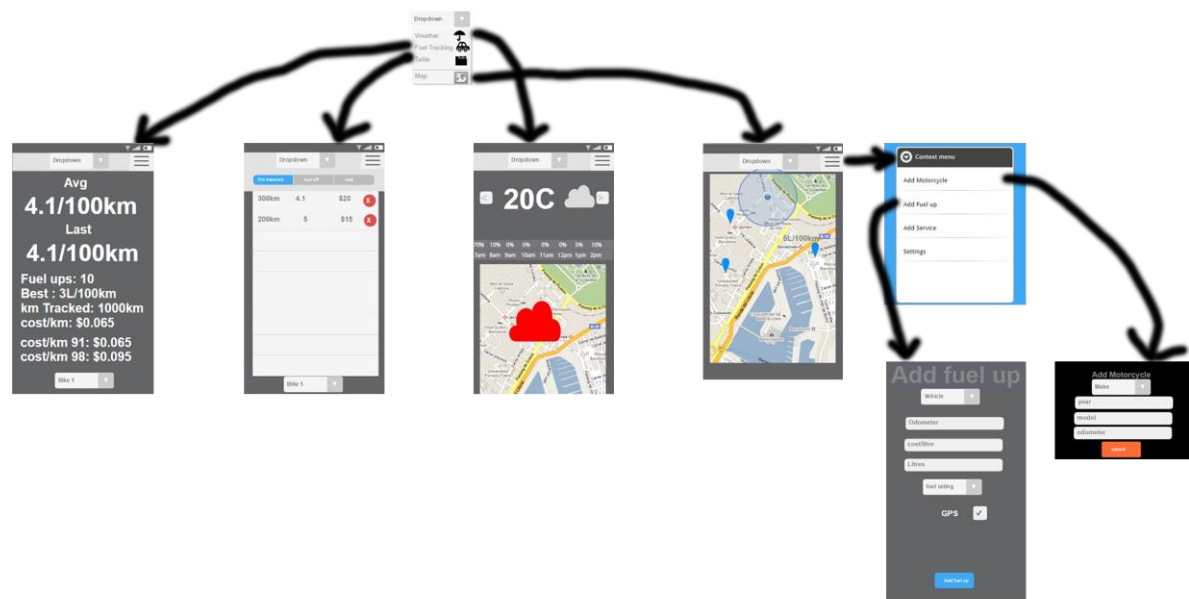
Data Model

All the data for the Application will be stored in either in 2 MySQL Tables or 2 Text Files. If I go with text files I will need code to decode the Data into The classes and values to be stored. Otherwise with tables I just store all the information in the columns and rows. One Document will keep all the Fuel Up class information and calculate extra information to be stored and read by some of the fragments. If I go the Text Document method, I will store each set of information on a new row and read/write them in a specific order.
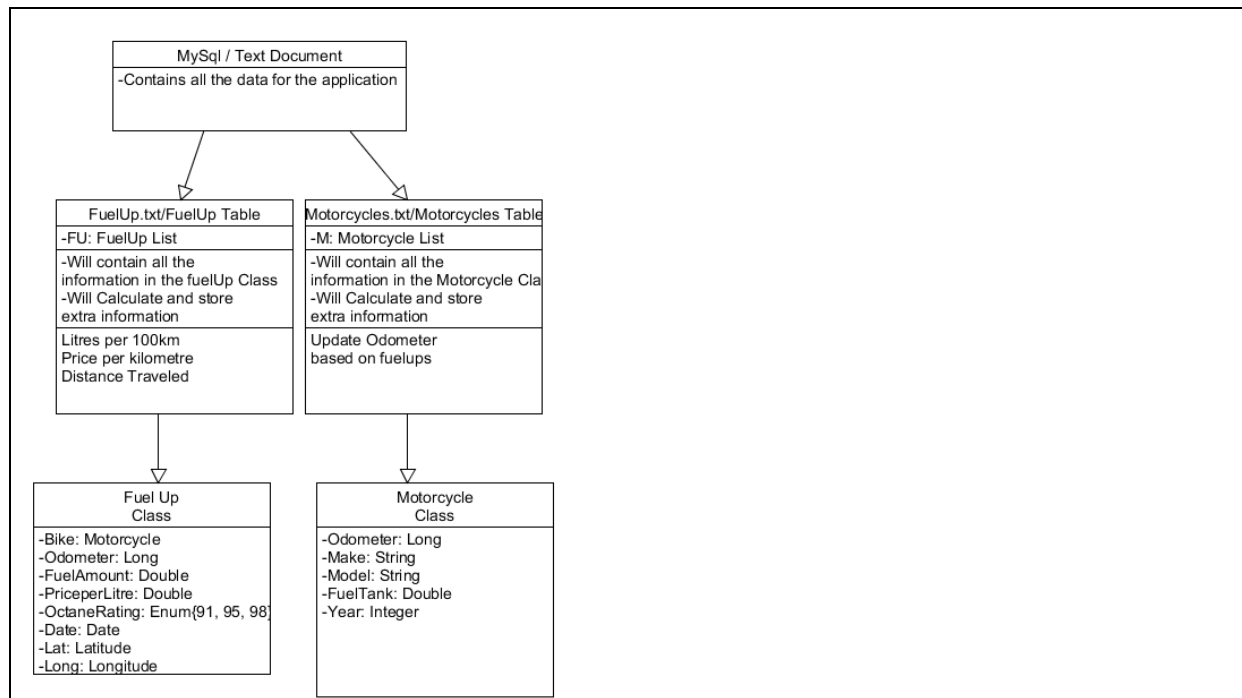
Limitations of API

In the Design of my Application I see little in the way of Limitations of the API in gaining what I want to achieve with this App. I am slightly concerned with integrating weather as I have not done it before but it does not seem hard to perform. Also I have not integrated MySql tables which seems harder than weather but not impossible. If it potentially limits my progress, I could use Text Files to save all information instead.

I should have no issue using the networking API for getting information such as Longitude and latitude and integrating maps as I have done it before with not too much challenge.

Summary

My Motorcycle companion app will be able to record fuel ups, view weather, view petrol station maps and store information saved, to be displayed to inform the user of their fuel consumption.

References

http://www.umlet.com/

https://www.fluidui.com/

Appendix

Scenarios

i)   Joe is planning to go on a long distance trip to Bright which is 340km north east from Melbourne. Joe's petrol tank is only 17L he wants to know how far he could go without filling up by working out his fuel efficiency. He will perform this by adding his motorcycle to the App and then filling up the motorcycle and recording it on the fuel up page to get the estimated fuel efficiency displayed on the Main page which will list his average distance he could travel before his tank will be empty so he can plan his route.

ii)   Justin is planning a Motorcycle trip around Australia; he needs to know the locations of petrol stations around the country so he can plan his trip so that he does not run out of fuel in the middle of nowhere. By looking at the Maps tab he can see all the petrol stations listed on Google maps.

iii)   Jake would like to go for a leisure ride through the Reefton spur on the weekend. He will only ride if the weather is nice with no rain. To check the weather condition he will go to the Weather page and check the weather conditions to see if he could ride on the weekend.

iv)   Karen likes to collect information for her motorcycle to determine how fuel efficient it is as well as checking the consistency of each trip she makes. If it is significantly variable she will take the bike to get serviced since she thinks there might be a mechanical issue or need to be serviced. To work this all out she will record the information into the app with the add fuel up and then click on the table tab to list all previous mileages that have been recorded.

v)   Lee likes to record his fuel efficiency of his motorcycle as well as keeping GPS information to show friends where he travelled to and filled up fuel. The app allows him the choice to record GPS coordinates of where he filled up and displays it on the Maps tab with a marker and lists his fuel efficiency for the specific fill up.

vi)   Simon really wants to know if buying premium fuel actually gets better mileage and more value per dollar than 91 octane. He wants to know if he can buy it and save money or if it costs more with only small fuel efficiency gains. After recording different fill ups with different octane ratings on the fuel up fragment, Simon will look at the Main tab it will list the price for value of each octane which is what Simon wants.

Navigation

The main tabs such as Fuel Tracking, Table, Weather and Maps are accessible from the drop down spinner on the Action bar. To add Motorcycles and Add fuel Ups you can do that from the context

menu on the Action Bar. This Streamlines the Actions and Tabs by containing them on the Action bar that is needed anyway instead of leaving these actions all over the fragments themselves allowing the user to navigate with simplicity and ease. All tabs and pages data will be stored as a fragment and you can go back to finish filling out information while still in the app.