For the coursework, we created a UML and a noun-verb and a sequential diagram during our design phase. The reason we decided to use a UML is that its graphical language makes it universal to understand so everyone in the team could understand what class did what. In addition, the visual design was helpful for all the team. The noun-verb gave the team understanding on how each class will work but using subtle information. The sequential diagram was created so that we could decide which objects are connected and the interactions between the objects in a time sequence.

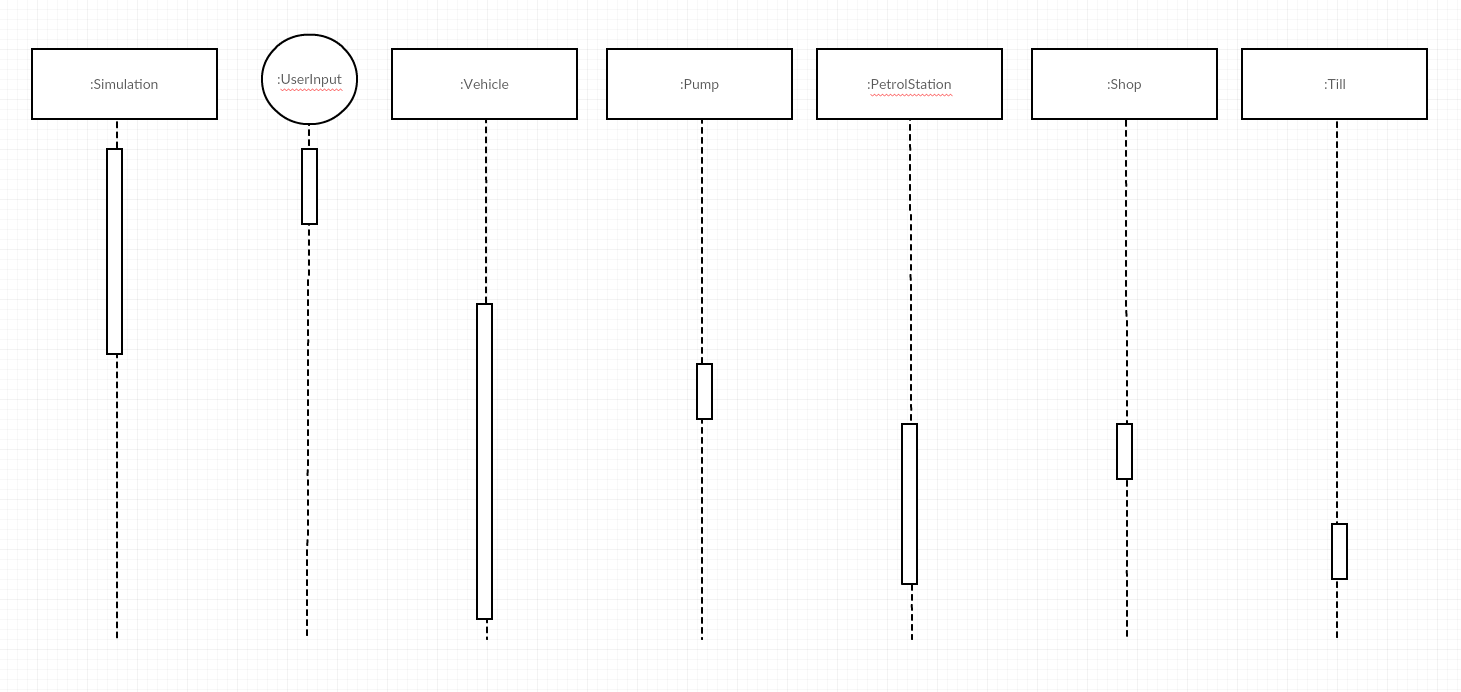
We used client code so that the code could support different scenarios without changing the library classes, for example, we used a vehicle class to as a server and made the vehicle classes the clients because it provides the information needed for the vehicles to function properly, other classes in our code also use client-server coding such as the petrol station class and the GUI classes.

The GUI uses scrollers and buttons to heighten the simplicity of functioning the program for users. We also created a textUI so that users can clearly see output of the program.

To do:

* code results and comparison
* the results of the simulations in tabular form;
* a brief (< 0.5 page) discussion of the results and their implications;
* a complete listing of your source code
* instructions on how to build and run the program.
* (Level 1 only) a sequence diagram for one of the main scenarios in the simulation (1 page);
* (Level 1 only) a brief description of the changes that would be required to your library classes to support simulations with the following properties: more types of vehicles; multiple types of fuel with different prices; parking away from the pump during shopping; vehicles breaking down during the simulation (0.5 page).

Simulation Sequential Diagram



Gets user input

sends vehicle into a queue for an amount of ticks

Vehicle browses shop (based on probability)

Vehicle leaves the pump

Vehicle goes to shop (based on probability)

Adds a vehicle to the pump if there is space

Constructs a vehicle

Sets variables e.g. ticks

Fills up vehicles tank

While the vehicle is filling up