PROJECT FOOD

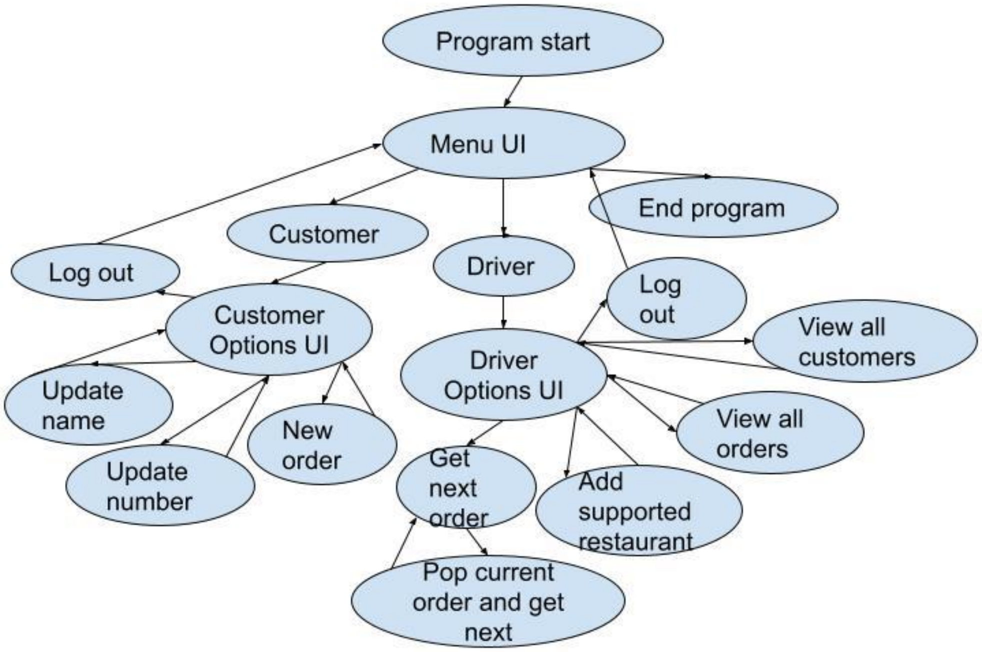
Basic application that the customer can use to put in a new order, and that the client can use to check a list of orders to be done, and mark them complete once finished.

App needs information on the restaurant location and name. There will be some structures to keep track of the customer list and the list of orders.

1. Customer Class//Option
    a. Use the app to submit an order
        i. Create a new order through customer menu
    b. Use the app to edit their name and phone number
        i. Update name/phone number
2. Driver class//option
    a. View list of created orders
    b. Mark orders complete
    c. Add restaurants
    d. Access information on which restaurant to pick up from and where the food is going to, as well as the customer name and phone number
    e. Raymond option (only driver)
        i. Can view the entire list of customers
3. App User unilateral
    a. Saving the information on a text document probably

Iteration	Feature	Areas affected	Relational interactions
1	Restaurant Class	Store name and address of restaurant	Must be accessible by drivers and app users and manipulated by drivers. Must be stored in a text document for later
2	Customer class	Store the name of customer, their phone number, and their address	Must be accessible by drivers and manipulated by customers. Must be stored in a text document for later
3	Queue Class	Basic queue data structure functionality	Should be inherited by the customer class in order to make a queue of orders

4	Save Customer Data	Save all information on text file for later use	A vector of item type customer that stores data and saves it onto a text file
5	Load Customer Data	Load all information from text file onto a vector for instance program use	Text file formatted to be loaded onto a vector of item type customer for program use
6	Order Class	Stores the customer class of the desired customer and the restaurant class for the customers desired restaurant	A class type that has access to both customer and restaurant that will be used as nodes for the queue



Potential TextFile.txt format

```
0110111
Bob
9132868667
222 N 22nd, Kansas City, KS
**
Sl;djkflkdsjf
Dslfjldskfj
Dsflkjdslkfj
Dslfjdsljf
**
Sdlkfjdslkjf
Ldskfjldskjf
Sdlkfjldfj
Sdlfjfdsjlko
**
```

CVS Parser from Rachel Singh.

Post Mortem Analysis

There had to be 3 different data structures and a parser to be able to store the data from the data structures onto a text file and reformat the text into a data structure once we need to retrieve the data for use in the program for later use. The program also had a difficult process in parsing as it might have been easier to create a sql database, but the client had budget problems so we made do. The implementation of the project works and is functional, but the client won’t be able to compete with doordash unless their budget increases and they do something bigger and better