CSCI 234 – Software Engineering | Spring 2019

User Requirements

Riley Kirkpatrick, William Brandes, Abrar Hilal, Mohammed Zakari

Client is the Schaper Sandwich Shop. There is currently one location of this business. The client is considering starting a delivery service in order to keep up with competitors. The client wants to franchise their business. The client also wants to dominate the sandwich business. To do so, it is believed that rather than create a traditional delivery service, it would be better to put the entire sandwich shop in a truck and have the truck drive to customer houses to deliver sandwiches. A sandwich truck will have the entire ability of a brick-and-mortar store.

The client wants to also update their menu. The client has specialty sandwiches that their customers love. They will also determine the best (most desirable) menu based on customer surveys. The menu must contain sandwiches that can be prepared in the limited space of the truck.

The basic operation of a truck will be as follows:

* Customer places an order by phone or online website.
* The address of the customer is used to determine the route the truck will follow. This route is updated each time an order is placed. Routes must not cause indefinite postponement of deliveries.
* Each order is placed in a queue of orders such that each order will be ready before the time the truck reaches the delivery destination. Order preparation must be as close as possible to delivery time so as to keep the sandwiches as fresh as possible.
* The orders are prepared and packaged for delivery.
* When a destination is reached, an employee takes the order to the customer’s door and collects payment.
* If the truck has no orders, it will patrol neighborhoods and make road side sales.
* If the truck runs low of a product, it will schedule a stop at a supply center to restock.
* The truck has a driver, an order taker and deliverer, and an order preparer.

The client wants a software system that will run the business of the truck. To test the concept of a mobile sandwich shop, the client wants a simulation of the operation of the truck. The software must:

* allow for order entry
* Schedule delivery times and the truck route. There must be an easy way to change the heuristics used to create the routes so that the company can experiment with different strategies.
* Scheduling of order preparation and packaging.
* Keep track of the cash register.
* Customers must be able to keep track of the location of the truck.

The client knows that the requirements for this operation are not complete and is relying on the software development company it hires to help flesh out additional details (or even ideas) regarding the mobile sandwich truck.

**Questions**

How will the menu be incorporated?

Text-based interface. We are focusing on feasibility, does not have to be incredible.

What is the priority of location vs distance?

Not sure yet.

Custom sandwiches?

There should be specialty sandwiches, but they can order custom sandwiches. Will be charged for extra stuff.

Minimum purchase?

Not currently.

Other food?

Chips, drinks, etc. WAIT to model those as it is.

Deli services--Sell meat by itself (ham, turkey, steaks, hamburger, etc)?

Yes, eventually.

How much money in cash register and accept cards?

Largest bill in register should be $20, and cards are accepted.

Customers should be able to see the order?

Yes.

How will order numbering/naming be handled?

The orders will be numbered. He wants customers to be able to set up an account in the future which will allow us to also include the customer’s name.

What should be available to be put on sandwiches?

Start with a few basic items, and eventually work towards having as many as possible.

Should the driver always follow the route determined by our heuristics/algorithms?

Yes.

Should gas be worried about?

Yes, we must stop for gas. Possibly need natural gas.

How far from center of operations is the working area?

10 blocks.

When should the distribution center be visited?

When the stock of an item is depleted. If an order requires something that is out of stock, this is a high priority.

What bread types?

Simple for now and later as many as possible.

Hours of operation?

Low priority: Not sure, the client has yet to figure it out. Hour or so before lunch through dinner hours are the current hours. Possibly in the future breakfast sandwiches will be available earlier in the day.

Edit or cancel an order?

Yes, for now. May change.

What if a mistake is made?

Needs to be addressed later, not a high priority.

Will future trucks/franchisees run under the same instance?

All trucks will run the same software, but each will run a different instance that can be configured differently. Configurability should be built in from the get-go.

Will we always know the exact location of the truck (GPS)?

Yes. In the simulation we will do location by address/street location.

How long does preparation of each sandwich take?

Shouldn’t take long; 1-2 minutes.

What are the prices of the sandwiches?

We will have set prices for each item of the sandwich, e.g. bread, each meat by lb, each cheese by lb, etc. (itemized approach to pricing).

What if the order taker is busy?

Driver is the back-up order taker.

How can you order?

Everything: texting, online, at the truck. In the simulation, we will place all of the orders in a file and simulate the ensuing events.

How much cash do we want on hand for giving customers change?

$200 worth of change. Subject to change.

What is the order of preparation for sandwiches?

They should be prepared as close to when they are being delivered as possible.

How should the truck operate when it does not have orders?

Head back to the distribution center, restock, and then hang out there.

How would we deal with obtaining gas?

Need gas stations throughout the map of the area, will add gas to the distribution center later if possible.

Go green?

In the future we might have trucks running on electricity. We would have to deal with how to recharge, possible use of solar panels, and calculate how long we can operate on electricity.

Simulated traffic?

Not yet, worry about route calculation for now.

**Stories**

Story #1:

The neighborhood will be a square that is 20 x 20 blocks. Each block has 9 houses on it. The streets are numbered 0-19 starting from the top left going to the east for vertical streets and to the south for horizontal streets. Each house will be numbered by the street perpendicular to the street they are on that is either the closest street to the north or west times 100 plus the number of houses down the street they are times 10. For example, the third house on the horizontal first street intersecting with the vertical third street is 330 E 1st Street.

The distribution center will be at 910 S 9th Street.

After completion of the first story our product will have the following functionality:

* the ability to generate random addresses in our Sandwich Truck’s neighborhood and save them to a file
* the ability to read a file of random addresses and insert them into a priority queue, where the priority is the distance from current truck location to the address; the truck is at a fixed location (the distribution center)
* the ability to create a representation of the neighborhood with delivery locations indicated

Story #2:

* + The ability to randomly generate a time that an order was made at a given Address, and store that time along with the address
  + Have the addresses stored in order of time
  + Compute a route for the truck to follow, in order of order time, and compute & show the total length of the truck’s route
  + Display a simulation of the truck’s neighborhood and show the truck moving along its route

Story #3:

* Resize the neighborhood so that there are 10 streets in each direction instead of 20. Put the distribution center on 510 East 5th Street.
* Use the strategy design pattern to implement at least 2 routing strategies. Use the strategy that you may have already developed and add a new strategy in which the truck can only make right turns. If you have already implemented this strategy, then create a route in which the truck can only take left hand turns. Your code should handle the creation of more than one truck for future expansion.
* Add a food order to each randomly generated customer event. Let’s keep this simple for now. Customers can order Sandwich 1, 2, or 3, Chips 1 or 2, and drink 1, 2, 3
* Compare the cost effectiveness of each routing method. Assumptions:
* Move from one address to another in 1 unit of time
* A stop at a delivery address takes 5 units of time
* A right-hand turn takes 2 units of time
* A left-hand turn takes 4 units of time
* Time to prepare a food order is 5 units of time
* Compute the total length of each route in distance and time

Story #4:

* Finish all current backlog items
* Change all time units to hours, minutes, seconds
  + Assume truck with travel 30 miles an hour
  + Assume that locations are 0.03 miles apart
  + The values for these assumptions should be easy to change
* Refactor the code so that you apply the Observer design pattern to your solution
* Create an observer that will print the truck location to standard output as it changes
* Your code should be able to easily handle the creation of more than one truck