Lauren McKeown's Capstone | Restaurant Flow

Project Objective:

For this project, we have decided to model a sit-down restaurant. Running a restaurant is hard! According to the National Restaurant Association, approximately 60% of new restaurants fail within the first year, and 80% fail within the first five years. Factors such as food prices, labor challenges, management struggles, restaurant layout decisions, and various other elements all contribute to the strain experienced by restaurants. All members of the team have had first-hand experience in this industry as we all worked restaurant jobs during the COVID-19 pandemic! We hope this model can provide clarity to restaurant owners, helping them further understand and optimize their operations in order to improve customer experience and maximize profits. Restaurants are complex and simple decisions regarding layout, kitchen flow and more can have significant effects on customer satisfaction and company profits.

Helpful Context | Model Design:

Our model assumes the typical set up a sit-down-and-order restaurant. There is a dining area and a kitchen in our model, which are connected through the kitchen queue.

The Dining Area:

- In terms of people, there are servers and customers in the dining area
- Customers have a CustomerStatus field which keeps track of what they are doing at any point during their restaurant stay. Customers are either waiting, seated, ordered or ready for the check. Customers enter the restaurant in a Party. The Party sig keeps track of the party size, customers in the party and the groups table as fields of the party sig. Customers will move through the restaurant flow with their assigned party.
- Upon a parties arrival in the restaurant, they are assigned a Table. In this model, tables have a capacity of either 2 or 4. Each table also has a unique table number and a set of customer orders. These are all fields of the Table sig. Additionally, each table can either be Available or Full. Each table must be in one of these status lists.

The Kitchen:

- The Kitchen is our integrated queue data structure for this model to keep the flow of the restaurant going.
- The Kitchen has three important sigs, the Kitchen, which is our pointer for the tail of the queue, the nodes in the queue which represent ticket orders and keep the structure of the queue as they point to the next ticket order to be cooked, and in those ticket orders, they store what the table has ordered in a set of Dishes.
- The Kitchen has two operations, enqueue, dequeue. Enqueue sends a ticket order to the kitchen and adds it to the tail of the queue (FIFO!) so the kitchen can start cooking! Dequeue deletes a ticket order and sends the ticket order's value (set of Dishes), to the table, and the Kitchen then moves the second order in place to the front.

Model Goals & Project Scope

Foundation Goals:

Our foundational goals were to have the customers, tables, menu, staff components defined and set up to resemble a restaurant in its opening state. Additionally, we wanted to have simple restaurant flow operations working, like seating, order taking, and a basic implementation of our first in first our kitchen queue. Given that our Kitchen and Restaurant components are in two files, our foundation goals are that the functionalities of these components work independent of one another.

Target Goals:

Originally, our target goal was to develop a more complex kitchen queue to take into account different factors like table size and and dish complexity when making decisions about what dishes to send through the kitchen first. We have transitioned this goal to our reach goal for the project.

Our new target goal is to fully integrate the kitchen queue with the front of house restaurant flow. We did not anticipate the restaurant & the kitchen being two septate files when starting the project; however, abstracting it this way made the most sense from a design perspective. Fully integrating the kitchen queue to work with the restaurant flow ordering function will be sufficient for reaching our target goal.

Reach Goals:

As stated above, our reach goal is to develop an additional kitchen queue to take into more complex, real life restaurant factors when deciding to process orders for a given table/tables.

Our original reach goals were to model how different sets of kitchen staff effect kitchen queue time. For example, entry-level kitchen staff might only be able to focus on one dish at a time while a veteran who has spent 5 years on staff can make three dishes at a time while focusing on an appetizer. All these factors effect how the speed and method at which the kitchen queue gets taken care of.

However, when creating our model, we quickly identified how many components need to be organized in this type of model so that nothing is conflicting with each other and ensuring that other components keep track of all the other components as well. So, we kept working on towards ensuring our foundation goals a very solid foundation that works well for our wanted model.

Model Visualization | Running our Model:

To view our model, use the built in sterling visualizer and run statement we have at the bottom of the front_of_house.frg file.

There are a few predicates that that can be found in the RUN STATEMENTS for front_of_house.frg section of our front_of_house.frg file that can be run to show our model. Each predicate is well commented about its functionality and relation to the model. To run each of then, place then in the run{} statement at the bottom of front_of_house.frg.

Limits to every model exist, here are the limits to our restaurant model:

- The scope of our model is pre set at a pretty small scale. We made this design choice for run time purposes but expanding this would be best for a more realistic view of a restaurant flow. A concrete example of this is our current table size representation. We limit the table occupancies to a max of two or four. Parties that meet or are smaller than these occupancies can sit at a table. Most restaurants have bigger tables for large groups; however, we decided it was best to keep these bigger examples out of our model for the scope of this project.
- Another limitation in our model is simultaneous movements of parties through the restaurant. Currently, one party moves through the restaurant flow at a time. This is not realistic but something we struggled to model with forge correctly. For

example, ideally when one part has ordered another could be seated in the next state. With all the moving parts in a restaurant, we found it difficult to accurately define rules to let people move through the restaurant without over / under constraining the model.

Stakeholders

Several stakeholders exist for this model.

Servers rely on tips to make an income; however, there are several things about a restaurant guests experience that the server cannot control. A significant factor is the kitchen wait time. Hungry customers are never happy and the time it takes the food to come out can have a large impact on how happy the customers are with their experience and therefore how much they are willing to tip. This model could change the way restaurants decide to operate their kitchen flow, increasing customer satisfaction helping as well as helping servers turn tables faster and make more money!

For restaurant managers , this tool could help them make their staff schedule in order to maximize kitchen flow capabilities. Additionally, during different times of the day, this tool could help managers decide how the kitchen should operate in order to minimize ticket times.

This tool would directly impact the kitchen staff , improving their working conditions by promoting a calmer and more organized environment. It would alleviate some of the heavy burdens associated with working in this high-stress restaurant position.

Restaurant Owners are another vital stakeholder. A faster kitchen queue enables a restaurant to turn tables more quickly, ultimately increasing revenue potential. This tool could assist restaurant owners in learning how to enhance their kitchen flow to maximize profits.

Lastly this tool could improve customers experience at the restaurant!

Here is our demo video!.