Report #1: Part 2

ChefBoyRD

Restaurant Automation

GROUP #6:

Richard Ahn
Zachary Blanco
Benjamin Chen
Jeffrey Huang
Jarod Morin
Seo Bo Shim
Brandon Smith

GITHUB:

https://github.com/ZacBlanco/ChefBoyRD
http://blanco.io/ChefBoyRD

SUBMISSION DATE:

February 12, 2017

Individual Contributions Breakdown

Responsibility Matrix									
	Team Member Name								
Task	Possible Points	Richard	Zachary	Benjamin	Jeffrey	Jarod	Seo Bo	Brandon	Completed Points
Project Management	10								
Sec.1: Customer Statement of Requirements	9	10%	25%	15%	0	15%	35%	0	9
Sec.2: User Story	6	0	15%	0	35%	15%	0	35%	6
Sec.3: Functional Requirements Specification	30	23%	22%	0	0	40%	15%	0	30
Sec.4: User Interface Specs	15	10%	15%	0	37%	0	1%	37%	15
Sec.5: Domain Analysis	25								
Sec.6: Plan of Work	5								
Total Points	100	9.3	12	1.35	7.65	14.3	7.8	7.65	60

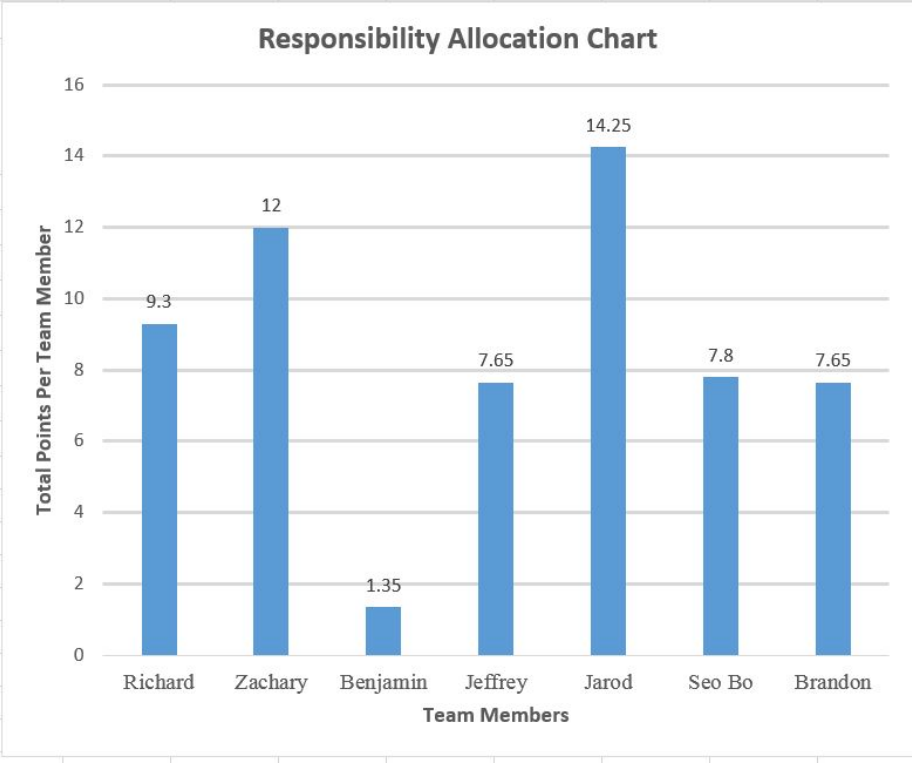


Table of Contents

1. Customer Statements of Requirements

a. Problem Statement

As an established restaurant, we wish to incorporate more technology to help run the restaurant more smoothly and to help us compete in the saturated restaurant market. Our understanding is that with the help of technology we are able to track information on large scales. The broad problem that restaurants like us face is creating meaningful contextual relationships using this information, which can then be analyzed to improve the restaurant overall. But more specifically, we would like to have the proposed software application address some specific problems. The problems we would like to address are improving the customer experience, and increasing the efficiency of running the restaurant and bringing more customers, which ultimately leads to the financial success of the restaurant.

PREDICTING INGREDIENT USAGE

Chef:

As a chef, one of my responsibilities is gauging the amount of food that needs to be prepared for a typical day. This plays a crucial role in the day-to-day operation of a typical food business. As the restaurant gets busier throughout the day, our kitchen has to predict which dishes will be ordered and begin preparing them in advance to keep up with customer demand. However, if we predict incorrectly, we risk preparing too many dishes and wasting both time and money with the wasted dishes. According to the NRDC, approximately 40% of food gets wasted, and a major contributor to that number is restaurants.[1] Our kitchen can attempt to prepare fewer dishes, but if we get too much business and have not made enough dishes, our customers will have to wait longer to receive their orders. This makes customers unhappy and discourages them from returning to our restaurant in the future.

With prior experience, it is possible to predict (to a certain degree) which dishes and how much food should be prepared from day to day. However, this method is very prone to error for even experienced chefs. A system that can analyze our restaurant's metrics and order histories would be extremely useful, as it will use this data to determine the amount of food we should prepare for a given day. This would not only reduce our restaurant's waste but also increase our profit margins. This allows us to invest more into our business and make our customers and employees happier.

As a chef, if I am constantly busy around the kitchen and preparing food, we would like this predictive system to not only be available within the software application; but also available to use and communicate with through a voice-activated service such as Amazon Alexa. Being

able to utilize this feature through Amazon Alexa's voice services means I will be able to work in the kitchen while receiving updates and predictions from the service on the day's food production. It should be able to provide the same type of estimates or similar to the ones available by the software application.

Manager:

We would like to reduce waste and excessive use of ingredients, and what would help is we can view exactly how much of each individual ingredients were used over different time spans. With the aid of this system, chefs can more accurately estimate the ingredients needed on a particular day, expediting the food preparation process and minimizing the waiting time for customers to receive their orders. We would like the application to give us detailed analyses and reports about the restaurant's order history over the course of days, weeks, months or years. We would like the application to display order history information in an easy-to-view format, such as dashboards with graphs and charts. This system is also helpful for me as a manager so that I can understand the day-to-day performance and long-term trends of the restaurant. The ability to see this data helps me determine overall business trajectory and make informed decisions to keep the restaurant financially successful.

We would also like the application to use the accumulated data to give the chef an estimate - an actual number - of the type and quantity of specific ingredients that are required throughout a typical day. The estimates should be able to tell how much of certain ingredients will be consumed within certain hours of business operation. Ultimately, the quantities for food preparation will be determined by the chef, but the system’s estimate should serve as a general benchmark from which the chef can base his decisions.

Overall, this should result in a more efficient use of food resources, especially with time-sensitive ingredients. It can also allow us to prepare food in advance with more time-fragile, but fresher, ingredients.

We do however recognize that one of the drawbacks of using the predictive software is the lack of reliability in its initial stage due to insufficient data. Therefore, we would like the system to provide a simple interface to allow us to sync our historical records from our existing POS system to the new one. This way, we can quickly get the prediction software running as best as possible. This preliminary step of obtaining data is extremely important because a predictive software that is unreliable for the first few weeks or even months is highly undesirable.

Overall, this system can help us serve our customers faster by telling us how much food to prepare at a given time. This system will also allow our business to waste fewer ingredients, resulting in greater profits for our restaurant. Our managers will also be provided with a clear and simple visualization of their restaurant’s financial status to make their own analyses and business decisions. Eventually, the profits from this system can be invested back into our business to provide higher quality ingredients, higher employee salaries, or even lower prices for customers.

CUSTOMER FEEDBACK

Waiter:

Customer feedback is very important in how I'm presenting myself to my customers. Some nights I receive awful tips, or no tips at all and I have no idea why this is the case. Though every customer is different, if I had some concrete feedback to work with, I could improve my customer service skills or communicate any problems to the management. Different customers use different signals to get my attention when they want something. Usually people will wave me over or call out when I pass. Sometimes when the customer thinks that I am being inattentive or am ignoring them, it is just that I didn't recognize that the signal they were using indicated that they wanted my attention. If people communicated more details about their expectations and the signals they used instead of simply rating the service as excellent or poor, it would allow me to learn these new signals and prevent me from repeating my mistakes.

Chef:

When I test new dishes, it would be very helpful to know if people like them. It's never easy to find out what exactly the customers are thinking, but if people are leaving unhappy or are absolutely raving about my dishes, they usually make their opinions clear. My waiters and the manager already do a good job of communicating what the customers think about my dishes, but it would be nice to have something concrete to work with.

Customer:

The waiter usually asks us to complete an online survey about our experience when we receive the check. The website is written on the receipt, and there are usually prizes, but I've never actually completed one. Entering the website on mobile is cumbersome so I prefer to respond on a home computer. However, by the time I've gotten home I've usually forgotten about the receipt. Sometimes going to a restaurant isn't the last thing I do while I am out and I don't remember the details about my dining experience by when I return home I would be happy to offer my opinion if the restaurant provided a solution so that customers could take the survey before leaving the restaurant.

Manager:

A critical task that I have to perform as a manager is finding the source of problems that I see. If the restaurant is only half-full on a Friday night, I have to find the cause behind the problem. Could it be an employee? Is there something wrong in the kitchen? This is where customer feedback is useful, where I can learn the perspective of the customers and act

accordingly to find solutions. In fact, it would be even more helpful to have contextual information about a specific feedback. Who was on shift during this time? What did they order?

I need to have improved methods of collecting feedback, because the current methods do not suffice. Verbal feedback is easy to understand, but difficult to quantify. Online surveys require patrons to visit a website and submit a specific form that restricts users entry. Though they are detailed, they are unpopular because they are so long!

Our previous implementation was to use a suggestion box, but the results were underwhelming. Very few customers offered suggestions and the results were disorganized. We have switched to an online form that requires customers to create an account to participate. To compensate for the added inconvenience, we provide incentives to respond. These prize rewards are also advertised on receipts. We have received more feedback as a result, but the increase in responses does not justify the cost of the prizes.

I want to improve the quantity and quality of the feedback collected, perhaps make feedback submission convenient to the customer and easily trackable. Then once I can receive the feedback, I have to make business decisions based on them and eventually report them to the restaurant's owner. Tracking feedback helps me by providing evidence for my decision-making process. If I need to justify my managerial decisions to the owner, it would be extremely helpful to have supporting data, such as the customer feedback.

I also want to make sure that feedback is sincere and addresses a problem. I would like the ability to keep track of feedback and determine whether the feedback is useful or not.

If a customer is blatantly dissatisfied with a service, in my experience they usually communicate this directly to me or the owner. However, this situation can allow minor or subtle problems to exist undetected. What if customers have feedback that is not substantial enough to warrant calling me or a staff member over?

This could be something like: the table is wobbly, the light is too dim at times, the seats feel a little hard. Individually, these bits of feedback may not be important, but collectively they could make a difference. Some of these items may not be important in the big scheme of things, but making these subtle changes are key to providing an excellent customer experience that exceeds expectations.

TABLE MANAGEMENT

Hostess:

As a hostess, one issue that I constantly deal with is seating customers. Each table seats a different number of people and different-sized groups of people arrive all at different times. Updating this is a tedious, error-prone process, especially during peak hours when the restaurant is busiest. I want to be able to quickly assign people to tables, and change

Customer:

Long wait times at restaurants are inconvenient and annoying to deal with. There are times I've had to wait outside because there were too many people lined up inside the restaurant.

But that's what I always reserve my spot at restaurants now. I usually call in, or make a reservation online. Even then, I do get a wait time estimate but I can wait in the comfort of my home and plan accordingly. But some of these online reservation systems are not convenient at all, I have to enter my full name, address, credit card number, when I just wanted to reserve a spot. And there are times where after I reserve a spot and come into the restaurant, I find out that my spot wasn't reserved at all. It would be nice if I could be more confident that the online-reservation is working. A simple confirmation message with a generic message just doesn't do it for me.

Also, the waiting times are not very accurate and can be inconvenient. I'll be told that I have to wait an hour, then when I arrive on time, I'm told I have to wait another 30 minutes! It would be nice to see where I am in line, and how many people are before me.

Overall, I just want to ensure that I'm not wasting my time when I decide to eat out at a restaurant. I'd like to have short, accurate waiting times, with minimal errors in how I'm being seated.

b. Glossary of Terms

Chefs: A critical member of the kitchen staff who directs kitchen activities including food preparation, cooking, and presentation. Also responsible for creating new recipes.

Customer: A person who uses, or intends to use the services and products of the restaurant.

Customer Satisfaction: A measurement of whether a patron's enjoyment or satisfaction of the restaurant experience meets or exceeds standards. This is impacted by quality of service, food quality, wait time, and overall experience.

Dish(es): A complete food item produced by the kitchen staff.

Feedback: Information that signifies, to any level of detail, a positive, neutral, or negative response to products or services.

Food preparation: A process executed by the kitchen staff between the time the food ingredients are stored, and cooked.

Host/hostess: restaurant employee in charge of greeting customers at the door, and seating arrangements in general

Ingredients: The edible components of a prepared dish before being cooked.

POS (Point-of-Sale): A system that a modern restaurant may use to manage its customer orders,process them to collect payment,and send the orders to the kitchen. These systems often include the cash registers and monitors that employees use to enter in orders.

Reservation: A customer declares and notifies the restaurant that they are arriving at a specified date and time. They usually must be made with customer's name, contact info, date/time, and number of guests.

Restaurant Portal: Where employees can access their role-specific functions in the software.

Employee Portal: Where individual employees can login and manage their shifts, contact information, non-role specific information.

Server: This term will be used to refer to the computer hardware system that hosts the software. This does not refer to the waiter/waitress who is commonly called a server.

2. User Stories

Higher size values indicate a longer expected implementation time.

Role: Manager (ST-MA)

Identifier	User Story	Size (1-10)
ST-MA-1	I can modify the menu based on inventory and customer demand.	6

ST-MA-2	I can pull up economic information (net gain, net loss, revenue, expenses, etc.) at ease	6
ST-MA-3	I can see the inventory that we currently have to make sure that we have enough ingredients and/or materials to continue the quality of service.	7
ST-MA-4	I can receive alerts pertaining to abrupt critical information such as negative feedback, a negative revenue trend, etc	4
ST-MA-5	I can see the customer feedback on specific employees and dishes served	5
ST-MA-6	I can see customer feedback organized into positive/negative categories	7
ST-MA-7	I can see who is working at the moment	2
ST-MA-8	I can find openings in the schedule and ask employees to come and work in those openings	4

Role: Chef (ST-CH)

Identifier	**User Story**	**Size**
ST-CH-1	I can see a queue of orders waiting to be made and their special instructions	4
ST-CH-2	I can view the current inventory levels	2
ST-CH-3	I can see and update inventory levels as needed and the manager can be notified	4
ST-CH-4	I can see a historic record of what and when dishes have been made	5
ST-CH-5	I can view customer feedback about the meals I prepare	6
ST-CH-6	I can receive recommendations on how to prepare the proper amount of ingredients for the restaurant at a certain time	10

Role: Host (ST-HO)

Identifier	User Story	Size
ST-HO-1	I can easily switch to the role of the waiter if we are short on staff.	3
ST-HO-2	I can input the customer's order into the queue for the cook to see	3
ST-HO-3	I can check in both customers that made reservations and customers waiting in line, and then update table availability	4
ST-HO-4	I can cancel a reservation and remove it from the queue list	2
ST-HO-5	I can see all current reservations and their phone number in case I need to call them	4
ST-HO-6	I can manually add an open reservation, specifying the customer's info and arrival info.	3
ST-HO-7	I can use a generated reservation number to refer to a customer's reservation.	2
ST-HO-8	I can easily keep track of moved and merged tables	4
ST-HO-9	I can see the current estimated waiting time for customers in line, to give customers an estimate.	3
ST-HO-10	I can see a list of customers waiting in line and their corresponding contact information	5

Role: Customer (ST-CU)

Identifier	User Story	Size
ST-CU-1	I can reserve a table with ease	4
ST-CU-2	I can cancel a reservation without reason	3
ST-CU-3	I can receive a confirmation to make sure that the	2

	information that I provided for the reservation is accurate.	
ST-CU-4	I can get a text confirmation of my reservation so that I can be confident my reservation was made.	4
ST-CU-5	I can be seated in a timely fashion when I arrive at the establishment	3
ST-CU-6	I can easily and privately submit feedback about my experience at any time	7

Role: Waiter (ST-WA)

Identifier	**User Story**	**Size**
ST-WA-1	I can take customer orders and notify kitchen staff of the order	7
ST-WA-2	I can include special instructions with the orders	3
ST-WA-3	I can easily add items onto a customer's order mid-way through the meal.	5

Role: General Employee (ST-GE)

Identifier	**User Story**	**Size**
ST-GE-1	I can see my work shift and hours worked in daily, weekly, monthly, yearly format as well as wage.	3
ST-GE-2	I can see who else is working at the same time that I will be working	3
ST-GE-3	I can request absence from work	1
ST-GE-4	I can post my shift and take coverages for other people	2
ST-GE-5	I can easily log into my account in the restaurant portal	3

3. Functional Requirements Specification

a. Stakeholders

One of our major stakeholders are restaurant managers and chefs. Since their goal is to raise the efficiency of their restaurant and maximize profits, our product would be very appealing to managers. Chefs in particular will find the freedom of having a voice-operated service that tells them what ingredients they need to prepare very useful.

Some equally important stakeholders also include the companies that developed the POS systems that our target restaurants use. These companies would be highly interested in a product that plugs in and adds powerful statistical analysis to their POS systems. Some popular POS systems include Vend or Bindo.

Dining customers will also find our product attractive since we provide the feature of allowing reservations to be made quickly online. In fact, our whole system is designed to reduce the wait time for all of these customers, so we believe that customers will be very supportive of our product.

Overall, we believe that all participants in the flow of a restaurant will benefit from our system.

b. Actors and Goals

Chef:

Chefs are initiators. Their primary goal is to initiate a call through either a physical interaction of the device or through the voice-operating service to give them a list of orders or ingredients to prepare.

Manager:

Managers are initiators. Their primary goal is to initiate a call to give them various statistics of the restaurant's order history for their own analysis. Additionally, they also have access to see which orders need to be prepared.

Waiter:

Waiters are initiators. Their primary goal is to input orders into their POS device.

Order:

Orders are participators. They're passive actors that are only used as data for our use cases.

Host:

Hosts are mainly participators. Their primary goal is to manage customer interaction before they arrive, and just when they arrive at the restaurant. This includes setting up reservations and table management.

c. Use Cases

i. Casual Descriptions

Food/Dish Use Case (FD-UC)

FD-UC1

- Manager or System Sync Settings <<initiates>> the use case
- Current Day's Order Data <<participates>> by providing order data
- Order Data Database <<participates>> by providing and receiving order data and the results of the prediction algorithm
- Administrative Portal <<observes>> by receiving access to the order data and prediction algorithm results

Throughout the day, data relating to the dishes ordered and the associated times is tracked. At the end of the day, this information can be added to the database used for predictive purposes by a manager, or by the program itself. The current data in the database is compiled with the new data and is then analyzed by the prediction algorithm. The results of this process are stored back into the database.

FD-UC2

- Manager or Chef <<initiates>> the use case
- Current Day's Order Data <<participates>> by providing order data
- Order Data Database <<participates>> by providing historical order data
- Administrative Portal <<observes>> by receiving the historical order data

As the data from orders is tracked on a day-to-day basis as a chef or manager it is useful to be able to view historical trends on this data. It can be used not only to make predictions but to make business decisions in terms of marketing and designing the menu for the establishment. By providing easy access to historical data the research and investment required in order to make the most informed decision is lowered.

Feedback Use Cases (FB-UC)

FB-UC1

- Customer <<initiates>> the use case
- Feedback Database <<participates>> by providing and receiving data
- Administrative Portal <<observes>> by noting any significant data trends

A customer completes and submits the in-store survey. The response data, as well as the time of submission and table number, are recorded in the feedback database. Data from the database is retrieved to be compiled with the new submission, and the result is

analyzed for trends of exceptionally poor or excellent ratings. If any such notable tendencies are found, they are displayed on the administrative portal.

FB-UC2

- Customer <<initiates>> the use case
- Feedback Database <<participates>> by providing and receiving feedback data
- Administrative Portal <<observes>> by receiving access to the sorted responses and word cloud

A customer submits an open-ended response relating to their experience at the restaurant. If possible, the submission is sorted into categories based on its content, indicating which part of the dining experience the feedback relates to. General information regarding the responses, such as a word cloud, is accessible through the administrative portal.

Table Reservation Use Case (TR-UC)

TR-UC1

- Customer <<initiates>> the use case and optionally receives a reservation confirmation message
- Seating Schedule Database <<participates>> by providing the current seating schedule and updating this data if necessary
- Host/Hostess Interface <<observes>> by noting any changes that occur in the seating schedule

A customer calls the restaurant to make a reservation, indicating party size and time of arrival. The current table schedule is accessed and compared with the request to see if an opening is available. If so, the table schedule is updated to include the new reservation and a confirmation is sent to the customer. If not, the customer is offered an alternative time. The customer may accept or provide a new time of arrival for their party, repeating the process until either a reservation is accepted or the customer does not provide another request.

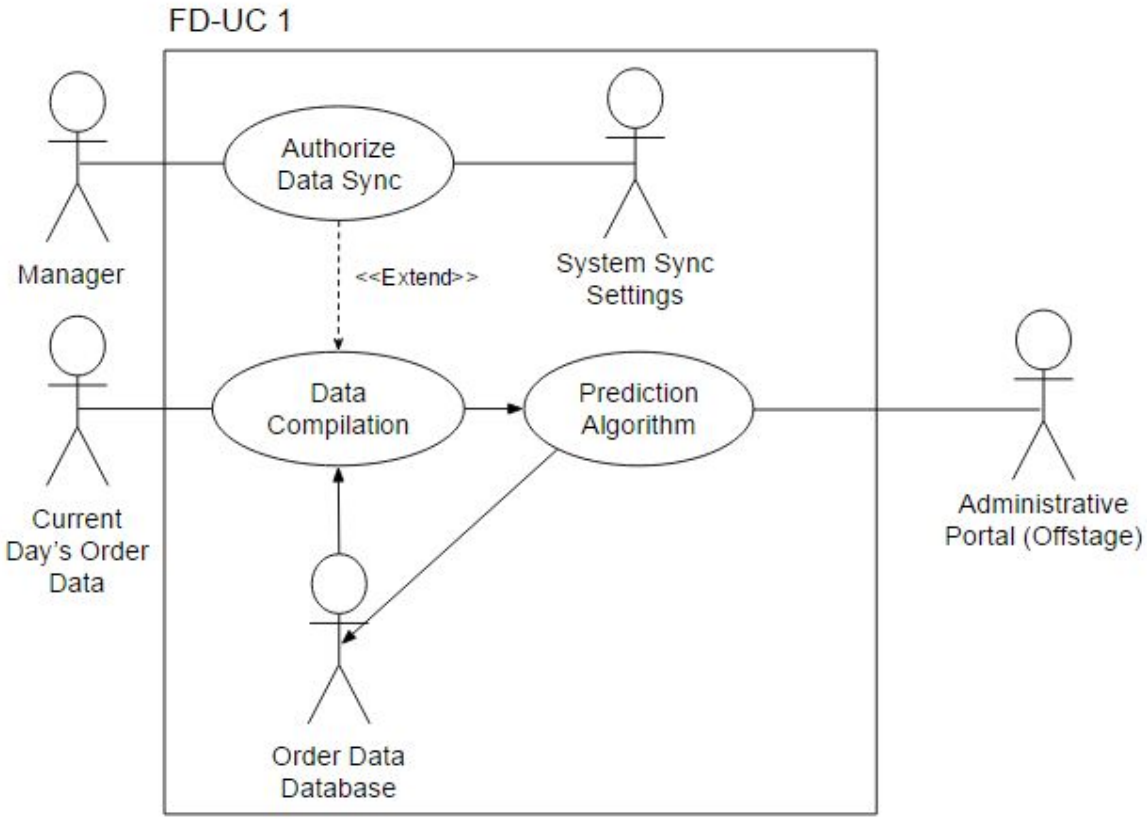
TR-UC2

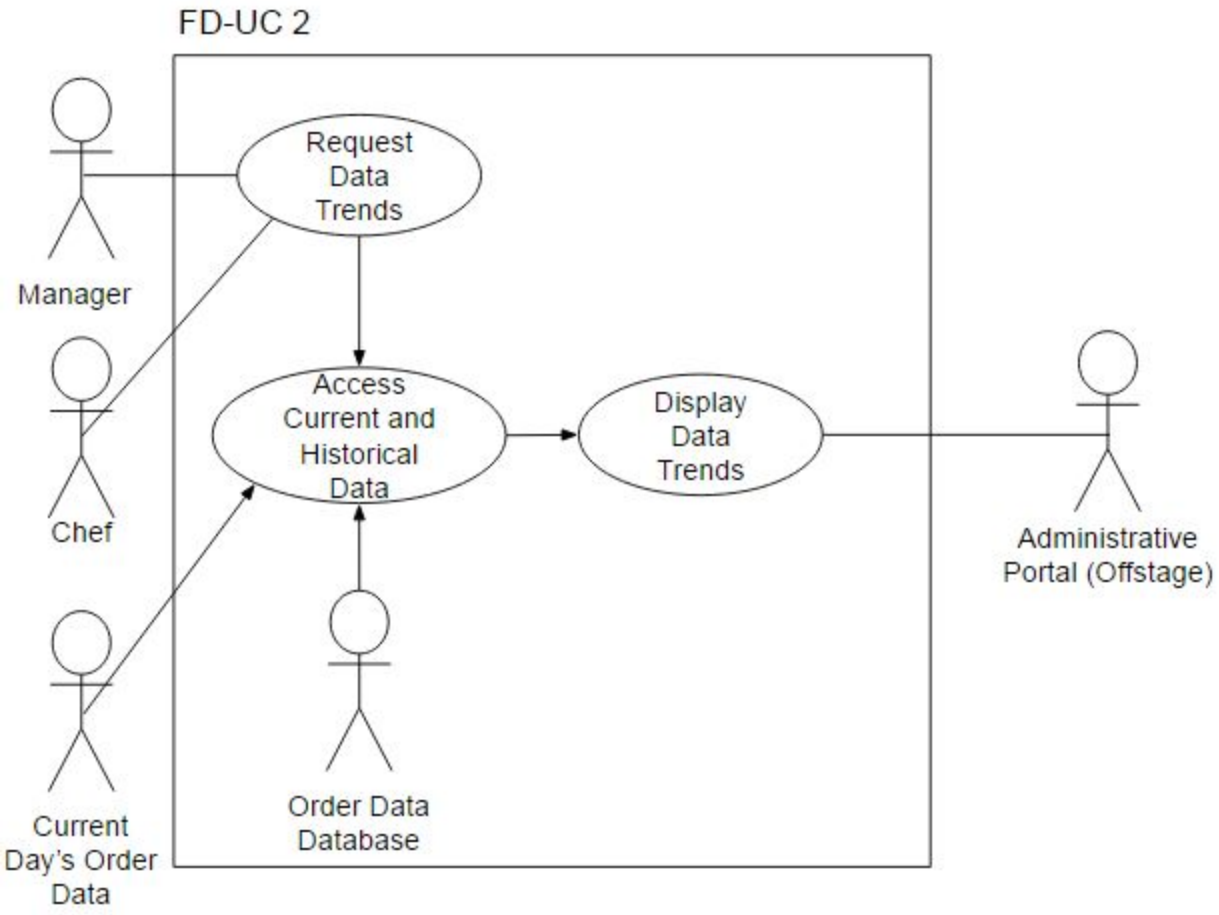
- Customer <<initiates>> the use case
- Seating Schedule Database <<participates>> by providing the current seating schedule and updating this data if necessary
- Host/Hostess Interface <<observes>> by noting any changes that occur in the seating schedule

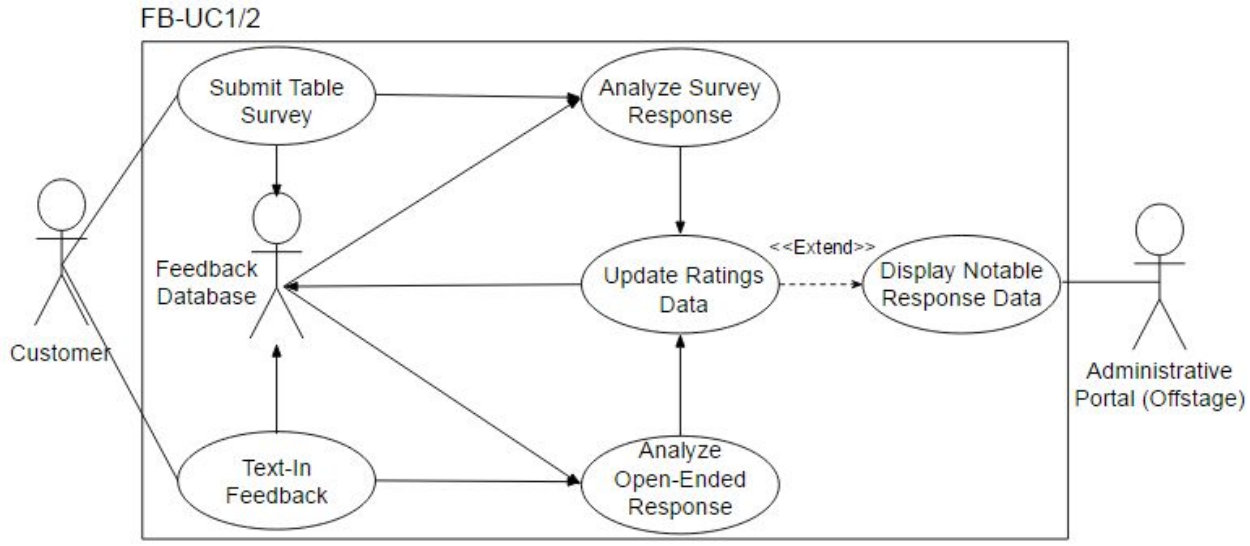
A customer requests to be seated at the restaurant without a reservation, indicating party size. The table scheduling database is accessed and compared with the request to determine if a table is available. If so, the customer is seated and the table schedule is

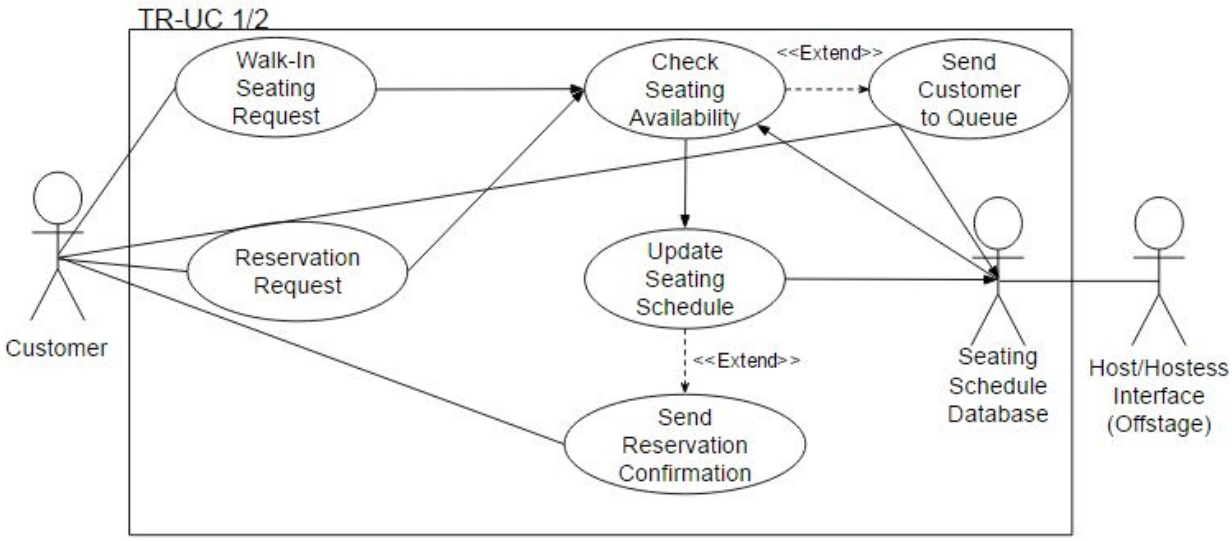
updated to reflect that the table is in use. Otherwise, the customer is allowed to wait for the next available table, placing them into a queue. The table schedule is updated to indicate that the next expected table vacancy will be reserved for the customers currently waiting in the queue.

ii. Use Case Diagrams




FD-UC 2
Request Data Trends
Manager
Chef
Access Current and Historical Data
Display Data Trends
Administrative Portal (Offstage)
Current Day's Order Data
Order Data Database


FB-UC1/2
Submit Table Survey
Analyze Survey Response
Feedback Database
<<Extend>>
Update Ratings Data
Display Notable Response Data
Customer
Administrative Portal (Offstage)
Text-In Feedback
Analyze Open-Ended Response


TR-UC 1/2
Walk-In Seating Request
Check Seating Availability
<<Extend>>
Send Customer to Queue
Reservation Request
Update Seating Schedule
<<Extend>>
Send Reservation Confirmation
Customer
Seating Schedule Database
Host/Hostess Interface (Offstage)

iii. Traceability Matrix

	ST Size	FD-UC1	FD-UC2	FB-UC1	FB-UC2	TR-UC1	TR-UC2
ST-MA-1	6	x	x	x	x		
ST-MA-2	6		x				
ST-MA-3	7	x	x				
ST-MA-4	4		x	x	x		
ST-MA-5	5			x			
ST-MA-6	7			x	x		
ST-MA-7	2			x			
ST-MA-8	4	x					
ST-CH-1	4	x					
ST-CH-2	2	x	x				
ST-CH-3	4	x					
ST-CH-4	5	x	x				
ST-CH-5	6				x		
ST-CH-6	10	x					
ST-HO-1	3					x	
ST-HO-2	3	x					
ST-HO-3	4					x	x
ST-HO-4	2					x	x
ST-HO-5	4					x	x
ST-HO-6	3					x	
ST-HO-7	2					x	
ST-HO-8	4					x	x
ST-HO-9	3					x	x
ST-HO-10	5					x	x
ST-CU-1	4					x	
ST-CU-2	3					x	
ST-CU-3	2					x	
ST-CU-4	4					x	
ST-CU-5	3					x	x
ST-CU-6	7				x		
ST-WA-1	7	x					
ST-WA-2	3	x					
ST-WA-3	5	x					

ST-GE-1	3			x			
ST-GE-2	3			x			
ST-GE-3	1			x			
ST-GE-4	2			x			
ST-GE-5	4			x			
	203	60	30	37	30	46	46

*Each User Story was counted multiple times if it was resolved by multiple US Cases

iv. Fully-Dressed Descriptions

Use Case FD-UC1: Orders Prediction

Related Reqs: ST-MA-1, ST-MA-3, ST-CH-1, ST-CH-2, ST-CH-3, ST-CH-4, ST-CH-6, ST-HO-2, ST-WA-1, ST-WA-2, ST-WA-3
Initiating Actor: Manager or Chef
Actor's Goal: Determine the number of ingredients that will be produced in a given time period.
Participating Actors: Order Database, Current Day's Order Data
Preconditions: Order data from previous day is stored within the order database. Manager or chef has been authenticated
Postconditions: The system responds with a set of data based on its prediction algorithm. The system will store its prediction to determine how accurate it the prediction was for future use.
Flow of Events for Main Success Scenario:

1. → Chef or Manager <<initiates>> the request either through the use interface or the voice service by providing the day and time that the actor would like information on.
2. ← The system responds with data representing the ingredients needed for the orders in a given period of time

Flow of Events for Extensions:

1. → Actor requests data for a time in the past
    a. ← System responds with invalid time exception
2. → Actor requests prediction when there is no order database connected or the order database is empty
    a. ← System responds with a message notifying the user that it can't make predictions without data.

Use Case UC-FB-1: Table Survey Analysis

Related Reqs: ST-MA-1, ST-MA-4, ST-MA-5, ST-MA-6, ST-MA-7, ST-GE-1
Initiating Actor: Customer
Actor's Goal: Provide feedback relating to specific aspects of dining experience.
Participating Actors: Feedback Database, Administrative Portal
Preconditions: Time and table number that are submitted with user survey will allow the system to allocate ratings to the appropriate employees using the employee schedule.
Postconditions: Feedback is stored and reflected in employee ratings. If certain thresholds are crossed, the manager is notified.
Flow of Events for Main Success Scenario:

1. → Customer <<initiates>> survey analysis by submitting a survey after a meal.
2. ← System requests feedback data from Feedback Database.
3. ← Feedback data from the Feedback Database is accessed for use in analysis.

4. ← Old and new data are analyzed by the program.
5. ← The Feedback Database is updated to reflect the new rating values.

Flow of Events for Extensions:

6. ← If a significant event occurs regarding the new rating totals after analysis, the Administrative Portal displays a notification.

Use Case TR-UC1: Table Reservation

Related Reqs: ST-HO-3, ST-HO-4, ST-HO-5, ST-HO-6, ST-HO-7, ST-HO-8, ST-HO-9, ST-HO-10, ST-CU-1, ST-CU-2, ST-CU-3, ST-CU-4, ST-CU-5

Initiating Actor: Customer

Actor's Goal: Make an online reservation including time and date of arrival and party size.

Participating Actors: Host/Hostess Interface, Seating Schedule Database

Preconditions: Customer is on the website. The restaurant is open.

Postconditions: Customer's reservation is set in the database and able to be seen by the host

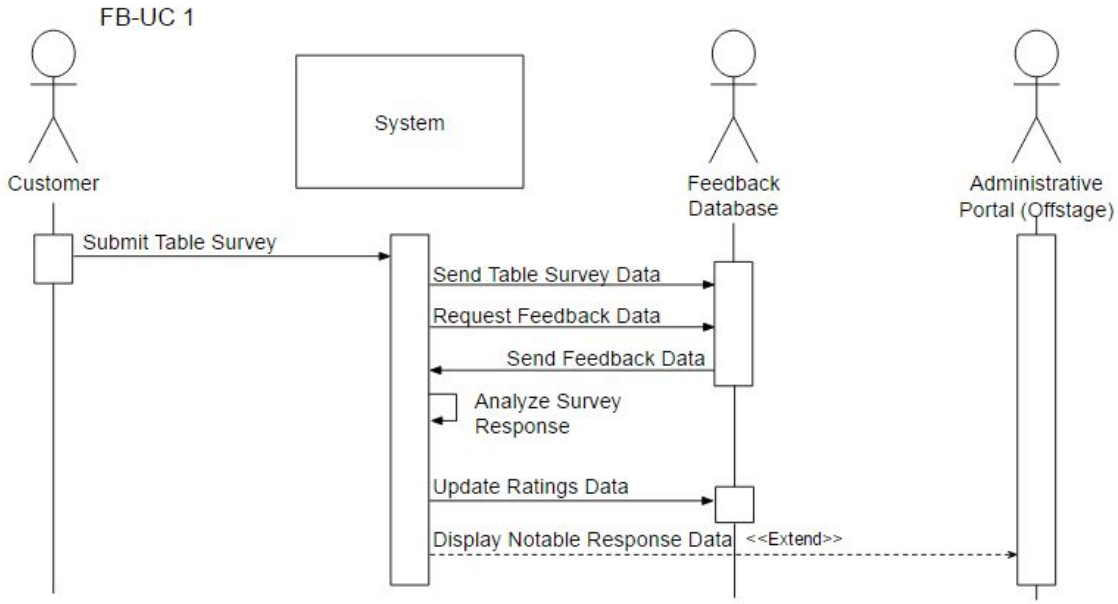
Flow of Events for Main Success Scenario:

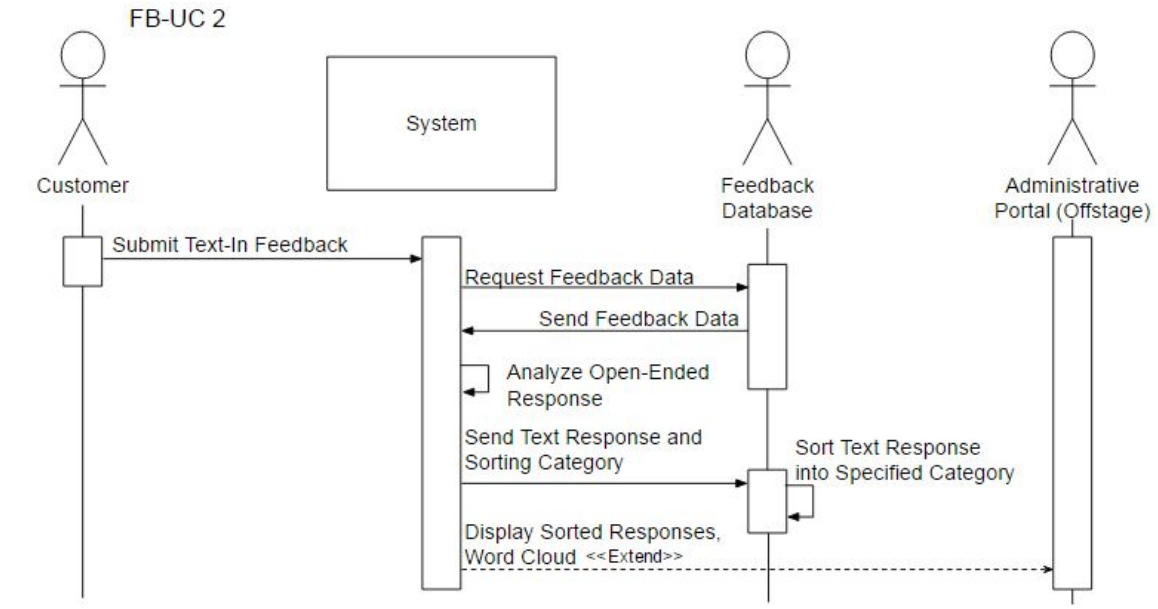
→ 1. Customer accesses ChefBoyRD website, and clicks reservation tab.
← 2. System responds by requesting reservation data from main server.
→ 3. Customer populates information about their reservation. Date/time, phone number.
← 4. System checks the reservation information and confirms the slot is empty.
← 5. System updates reservation information to the database.
← 6. System sends the user a reservation confirmation.

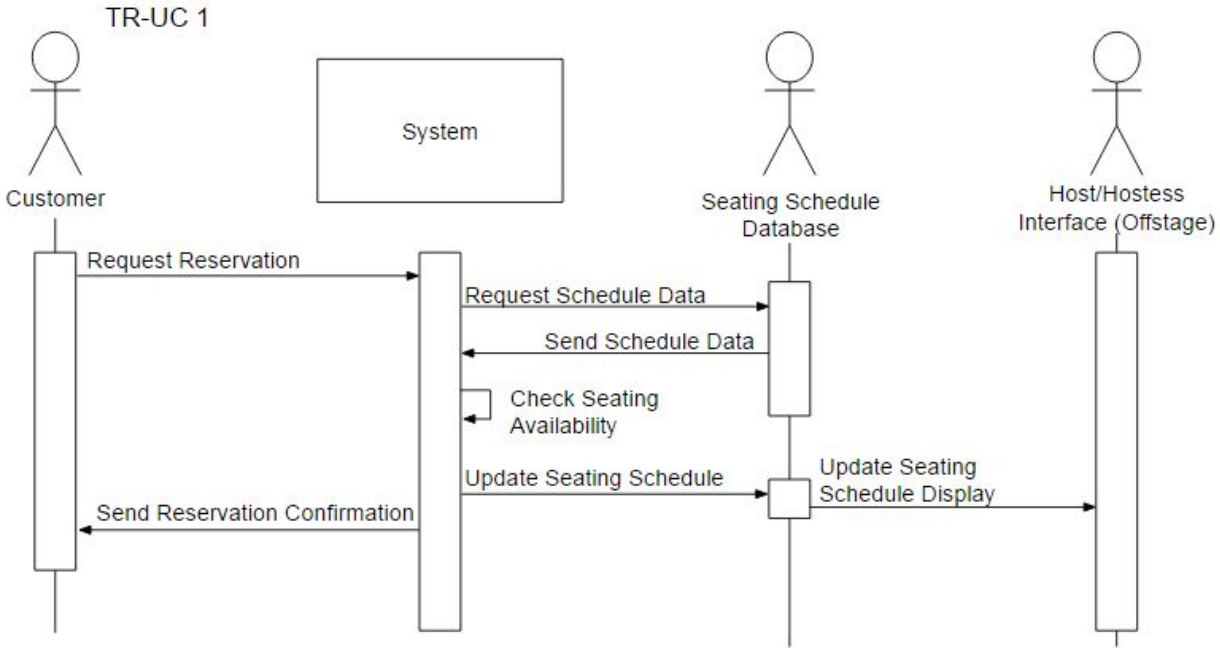
Flow of Events for Extensions:

→ 1. Customer requests a reservation that is already filled
← a. The system responds by checking if the reservation is available.
← b. The system notifies customer that it is not a valid reservation.
→ 2. Customer wishes to cancel a reservation
← a. The system responds by checking to see if the reservation matches the customer who is requesting it. (By phone number)
← b. Upon confirmation, the system makes a change to the reservations list and updates the database.
← c. The system sends a confirmation to the user that reservation has been cancelled.

d. System Sequence Diagrams




FB-UC 2
Customer
System
Feedback Database
Administrative Portal (Offstage)
Submit Text-In Feedback
Request Feedback Data
Send Feedback Data
Analyze Open-Ended Response
Send Text Response and Sorting Category
Sort Text Response into Specified Category
Display Sorted Responses, Word Cloud <<Extend>>


TR-UC 1
Customer
System
Seating Schedule Database
Host/Hostess Interface (Offstage)
Request Reservation
Request Schedule Data
Send Schedule Data
Check Seating Availability
Update Seating Schedule
Update Seating Schedule Display
Send Reservation Confirmation

4. User Interface Specification

a. Preliminary Design

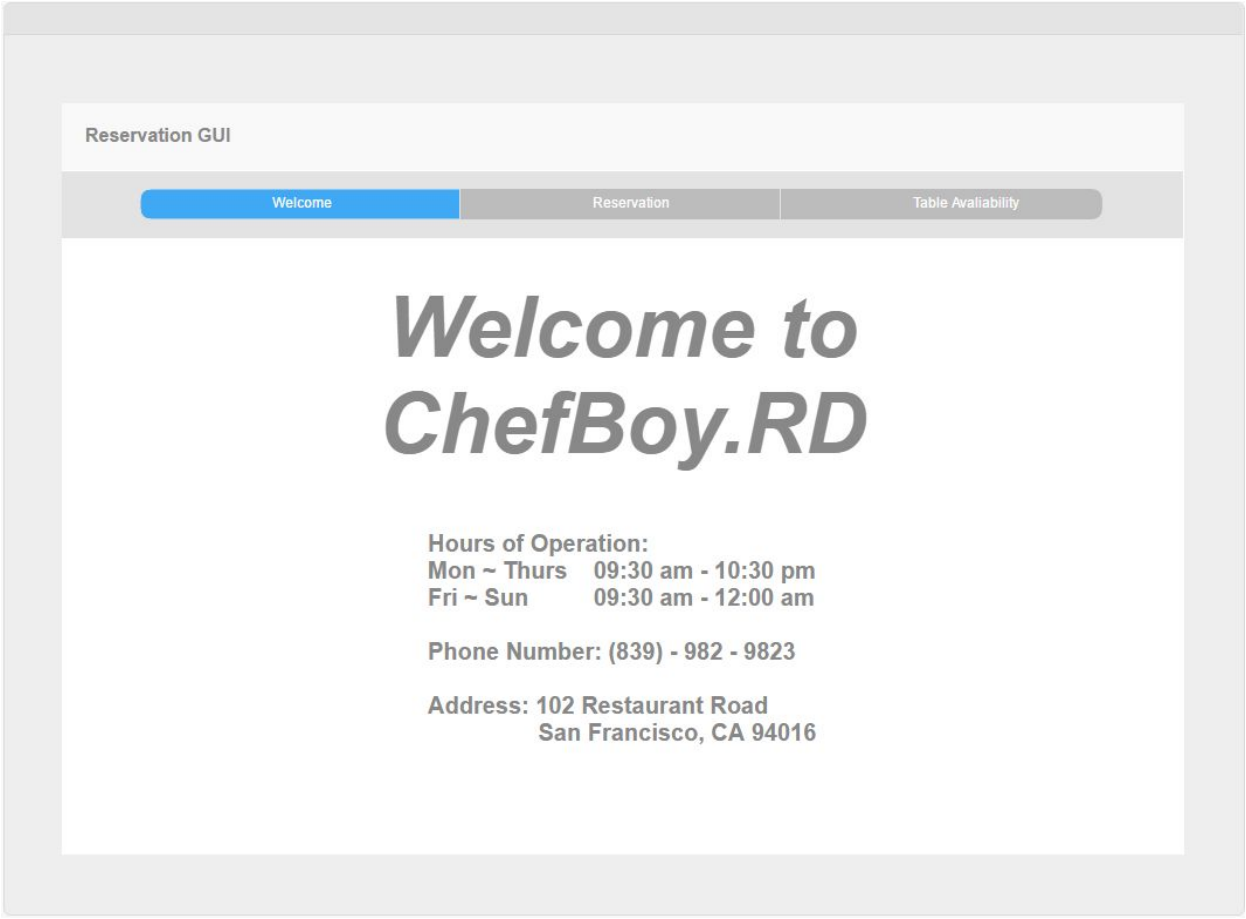


Figure 1: Customer welcome screen to give basic information about the restaurant such as address, phone number, opening and closing hours, reservation times, and table availability.

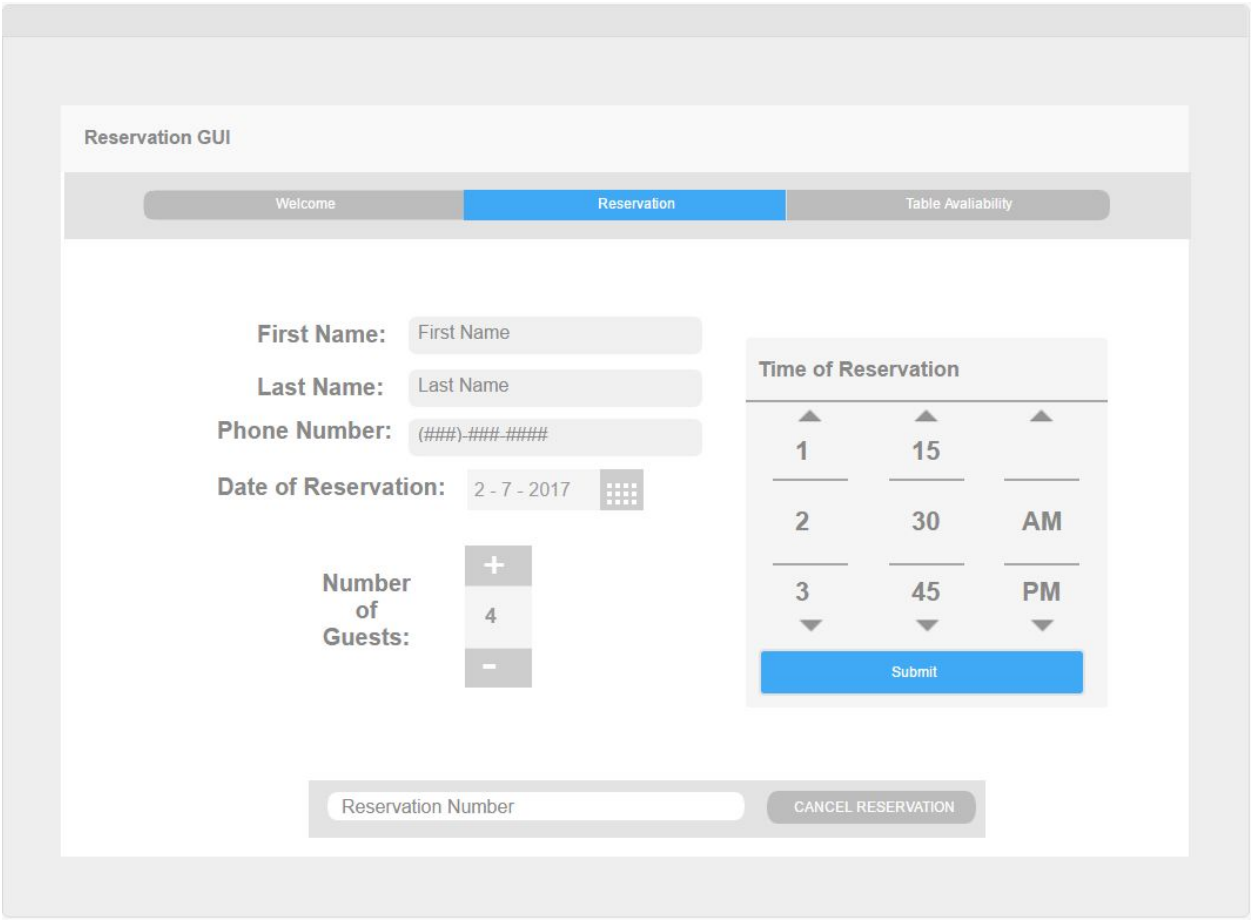


Figure 2: Customers make reservations by typing their first and last name, phone number, the number of guests to be seated, the date of reservation and the time of reservation. Customers can also cancel reservations if they have their reservation number

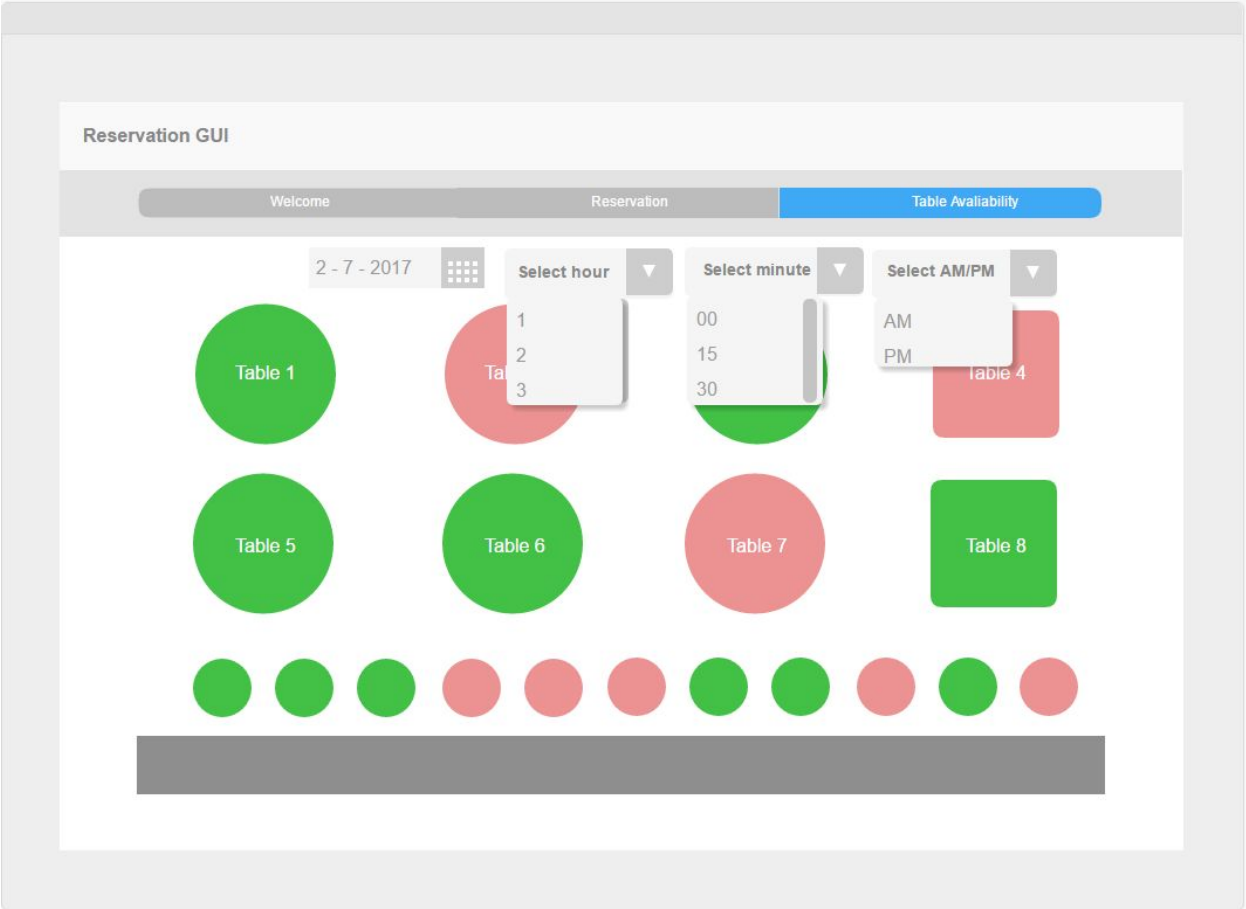


Figure 3: Customers can see the tables that are available depending the date. Red tables signify that the table is reserved or occupied and green tables signify that they are available. Smaller seating options are indicated along the bar and tables 4 and 5 are normally used to accommodate larger groups of guests (seating will vary from restaurant to restaurant).

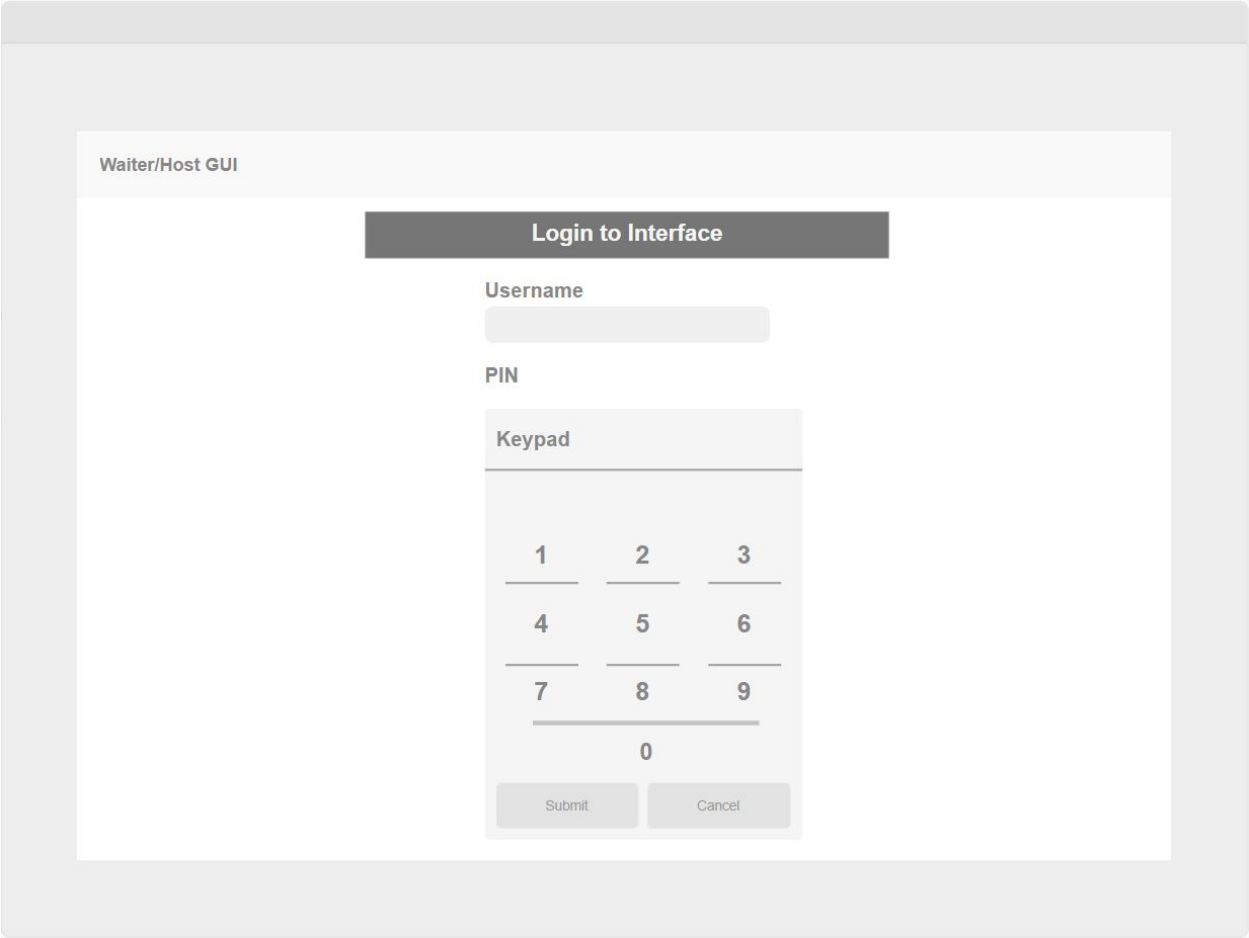


Figure 4: Hosts and waiters log-in into the interface using a username and a pin for easy access. This will prevent any overlapping information between waiters and hosts. Having unique operators for an interface will prevent miscommunication.

Figure 5: Host will have the visual representation of the restaurant table availability just to allow them to have a general idea of how populated their restaurant is. From the navigation bar, the host also has the ability to make a reservation, manage customer walk-ins, and see the customer queue list. The host is also able to put the table into occupied or vacant mode depending on if the customer wants to sit at a different table as shown in the blue pop-up bubble.

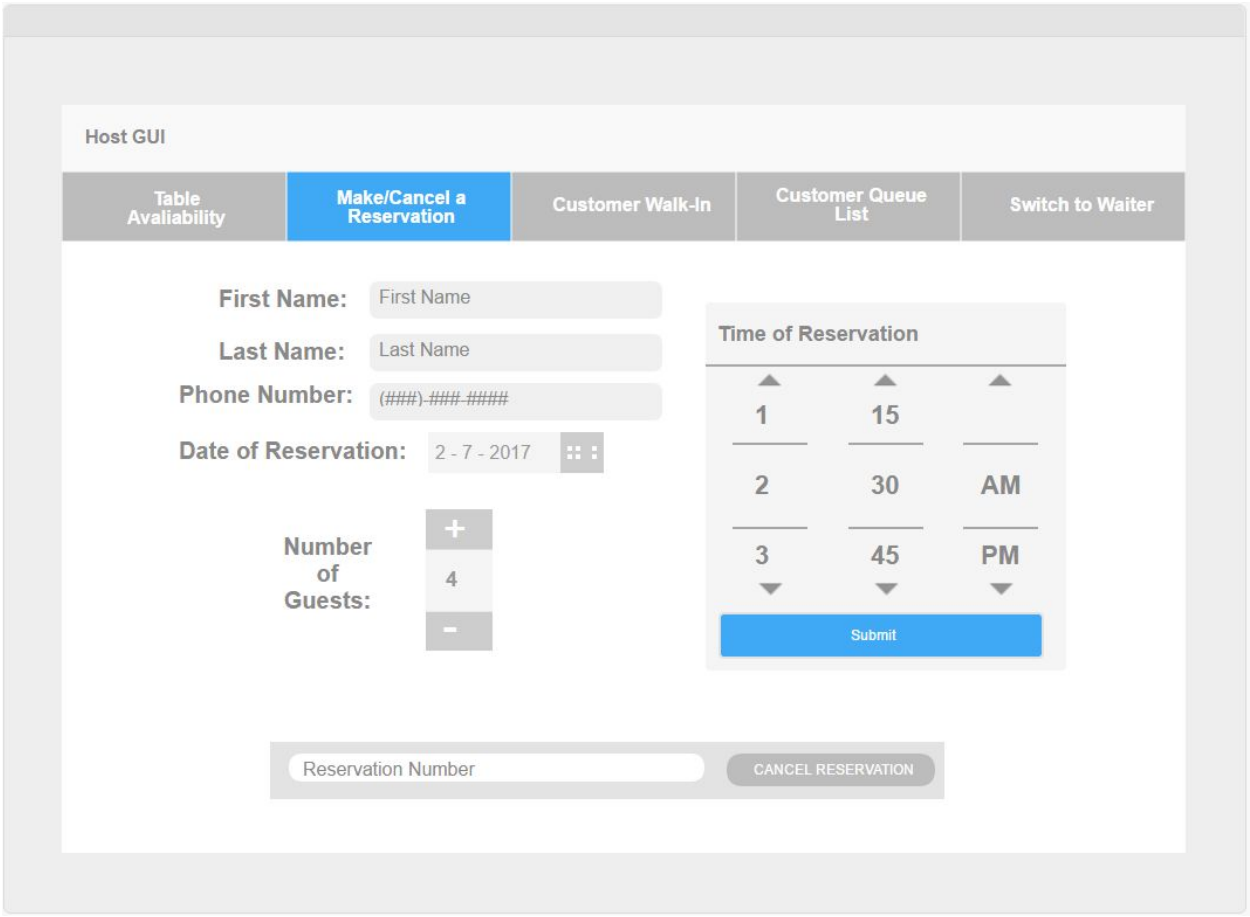


Figure 6: Host has access to making a reservation in order to accommodate for the people that call the restaurants in order to make a reservation instead of going online and making a reservation there. The host will ask the customer for their first and last name, their phone number, the number of guests, and the time of reservation. If the customer wants to cancel the reservation, the host can either input the reservation number and cancel the reservation via this method or a method to be discussed in the customer queue list.

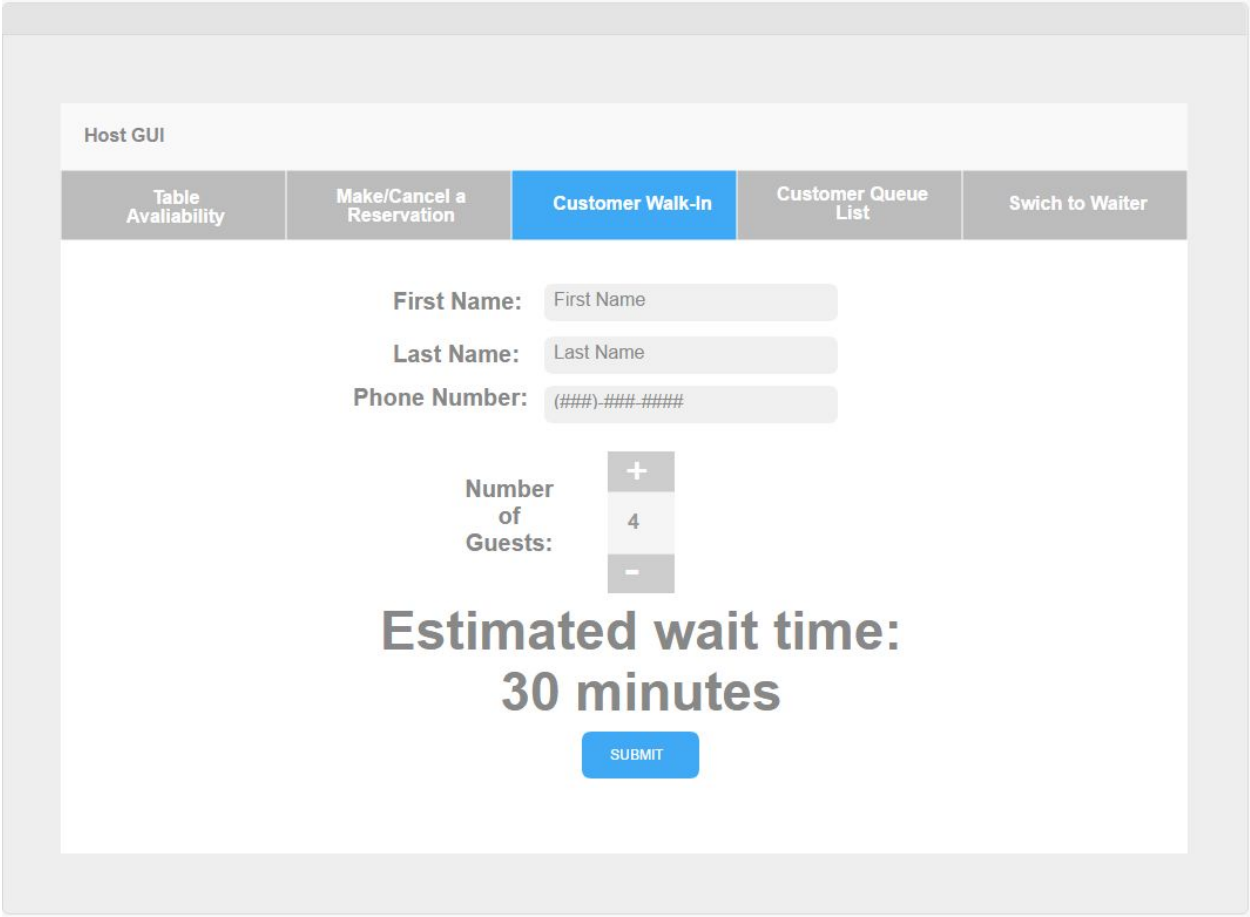


Figure 7: Customers that walk into the restaurant that haven't reserved a table should be seated accordingly by the host. If there are no tables available shown on the table availability, then the customer should be put into the customer queue list using the form shown above. If the customer needs to be placed into the queue list, there should be an estimated amount of time that the customer needs to wait in order to take a seat at a table.

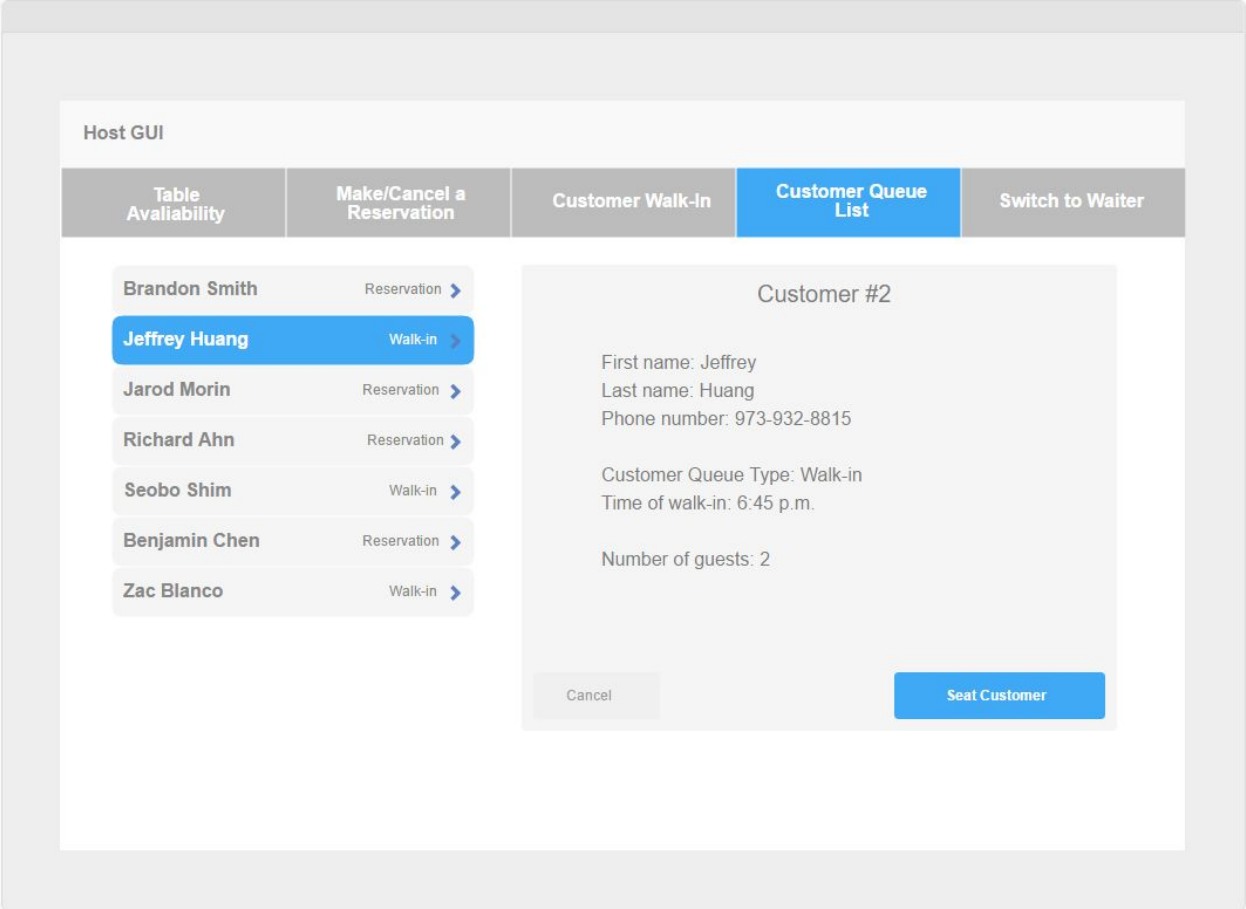


Figure 8: Host has access to the list of people that are currently in the customer queue list in order to keep track of the reservation times. The program will determine the order of which the customers can be seated. Upon selecting a customer in the list, the customer's reservation/walk-in information will be displayed for easier understanding of seating options. The final option "Switch to Waiter" allows the host to switch between the waiter and the host role depending on the situation.

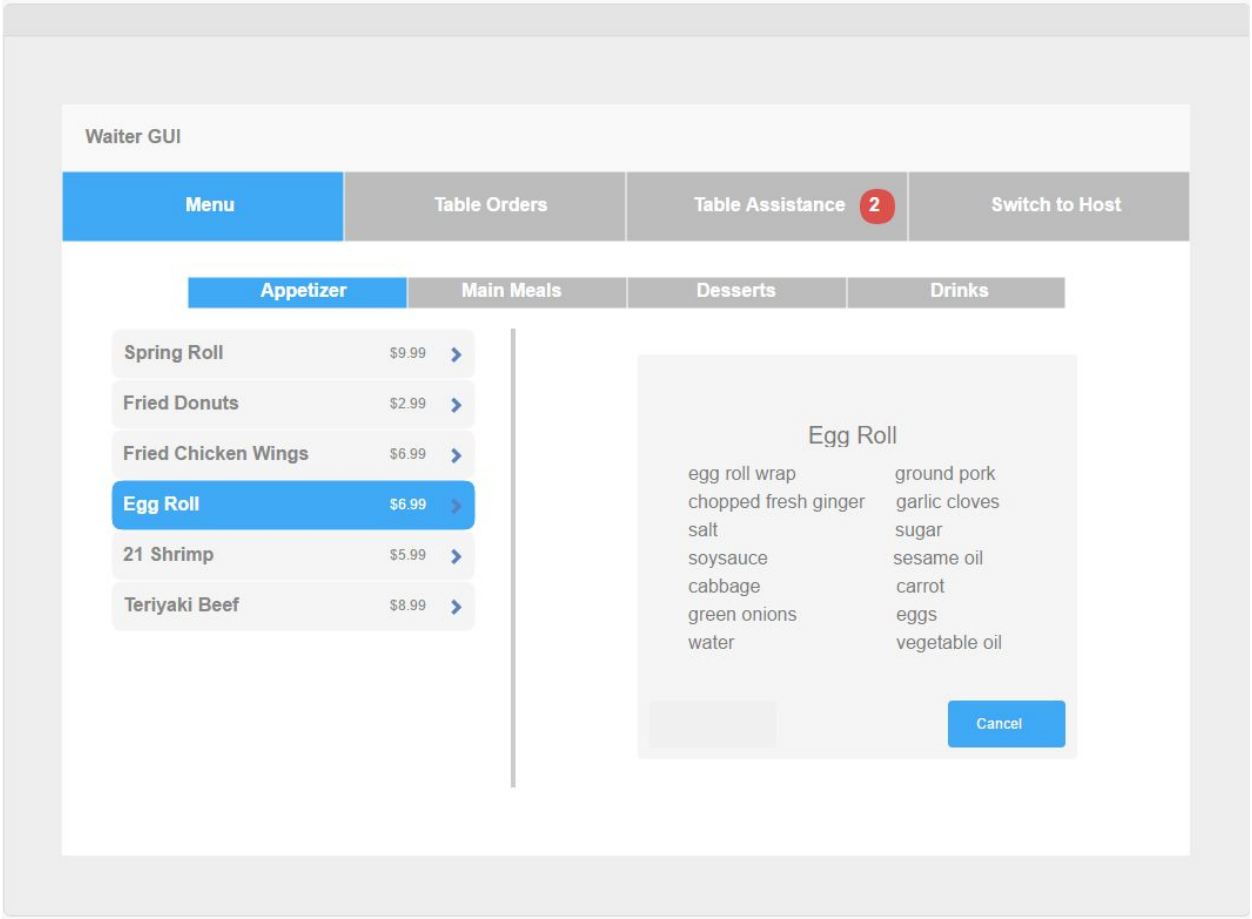


Figure 9: The waiter can carry around a small device with them as they take the orders which can allow for a quickly moving information to the kitchen in order to begin cooking the meals. The menu can display the daily specials as well as the ingredients used within each menu item in the case that customers are worried about allergies or nutritional content.

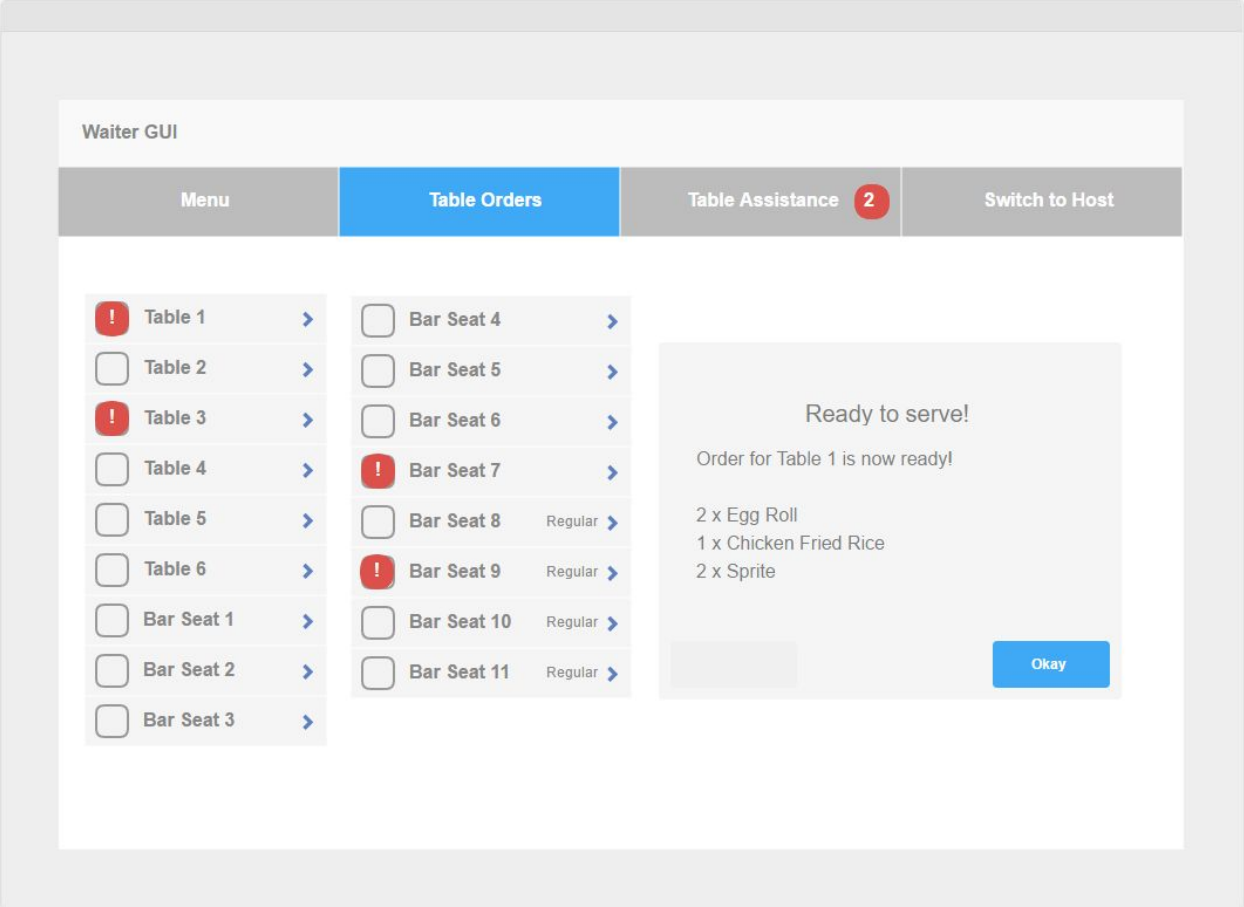


Figure 10: The waiter can also see the orders that have been placed at each table. Chefs can dispatch a notification when the food for each table is ready and waiters will be able to know immediately. This speeds up the communication process between the kitchen and the waiters so that food can be served in a timely manner.

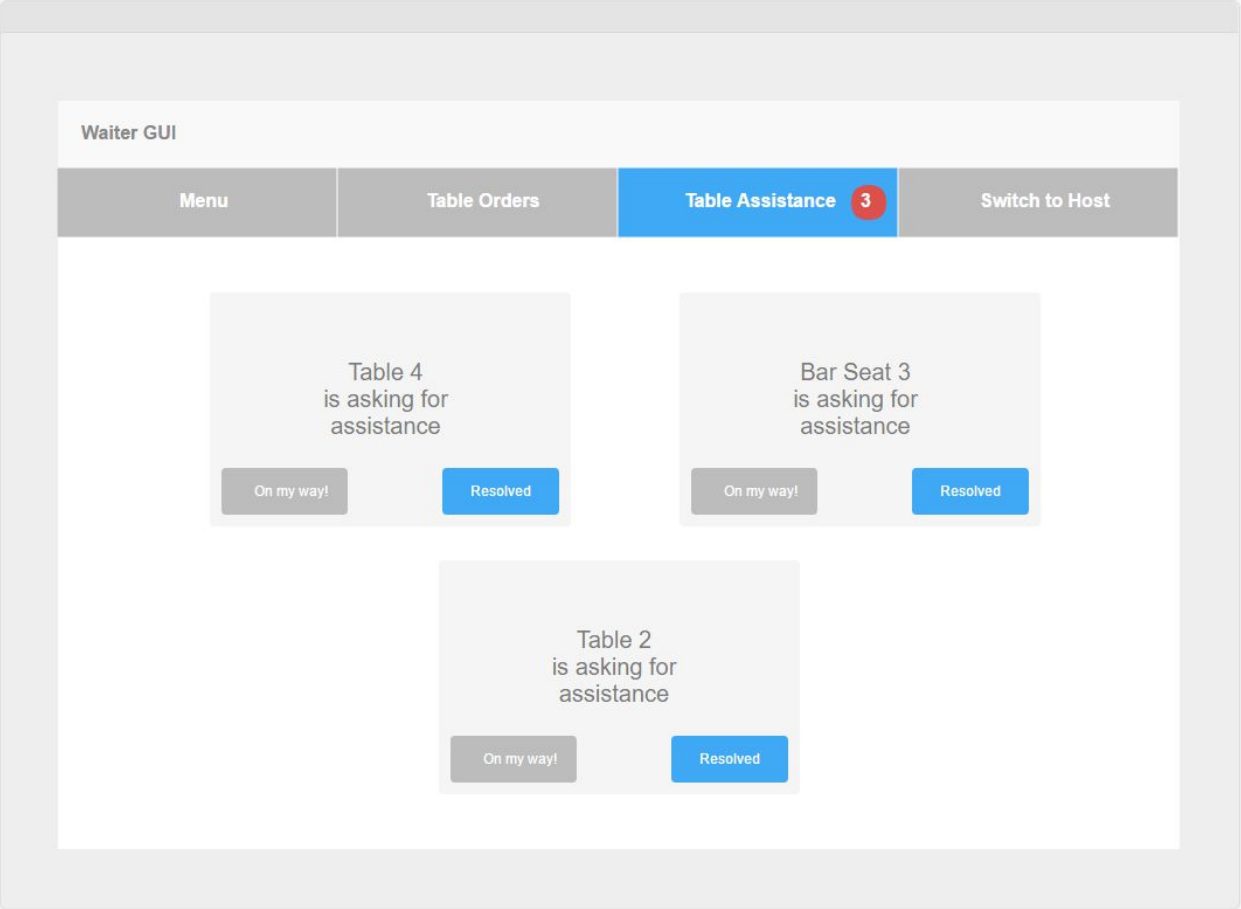


Figure 11: Within the waiters' interface the waiters can receive notifications from tables which need assistance. This can stop the slow servicing of customers because the waiters can always know which tables that need to be tended to.

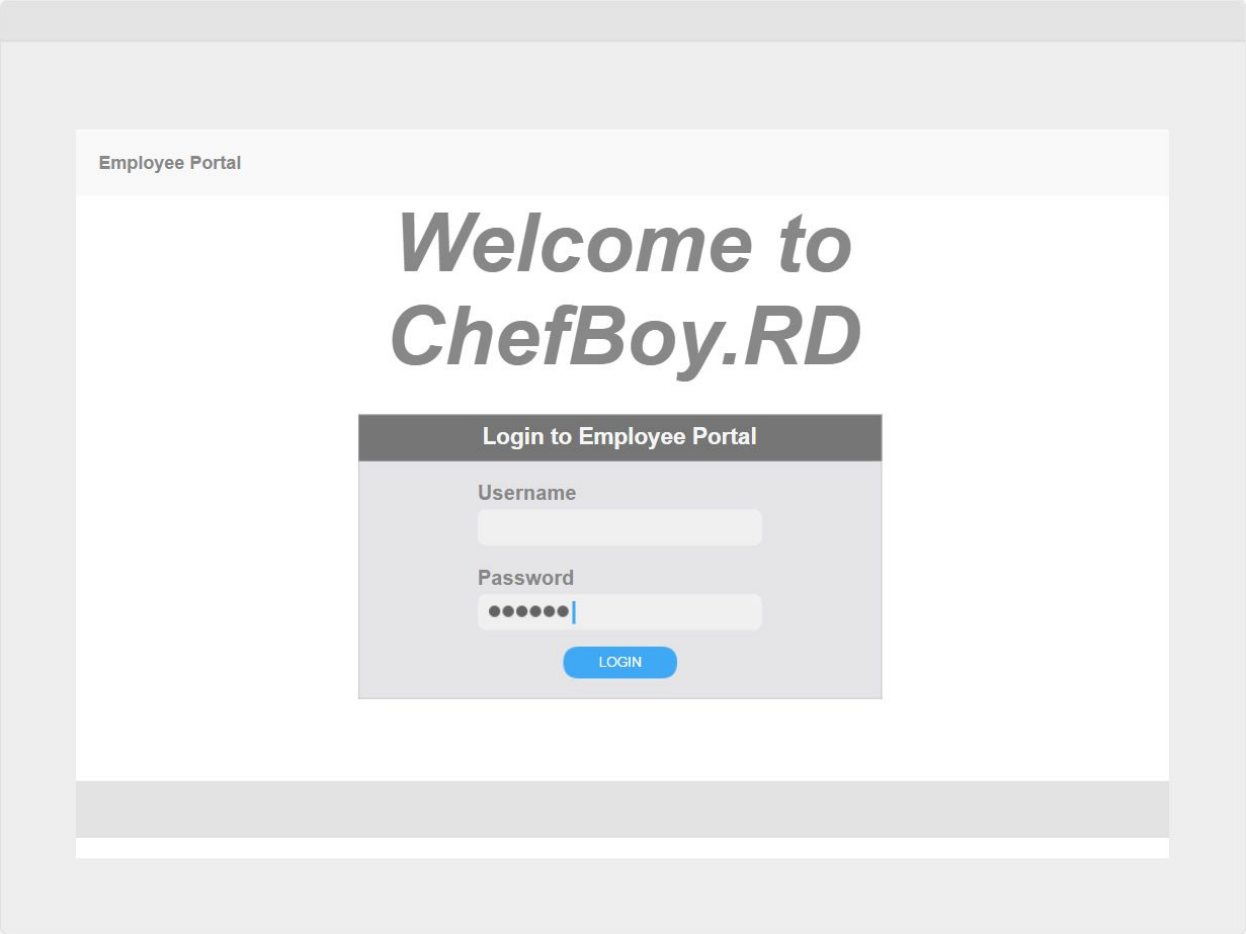


Figure 12: Employees are able to login via a web portal. When hired, an employee will be assigned specific roles which correlate to which menus the employee is able to access after successfully signing in. Username and password will be determined by the manager in terms of how secure they want their login information to be.

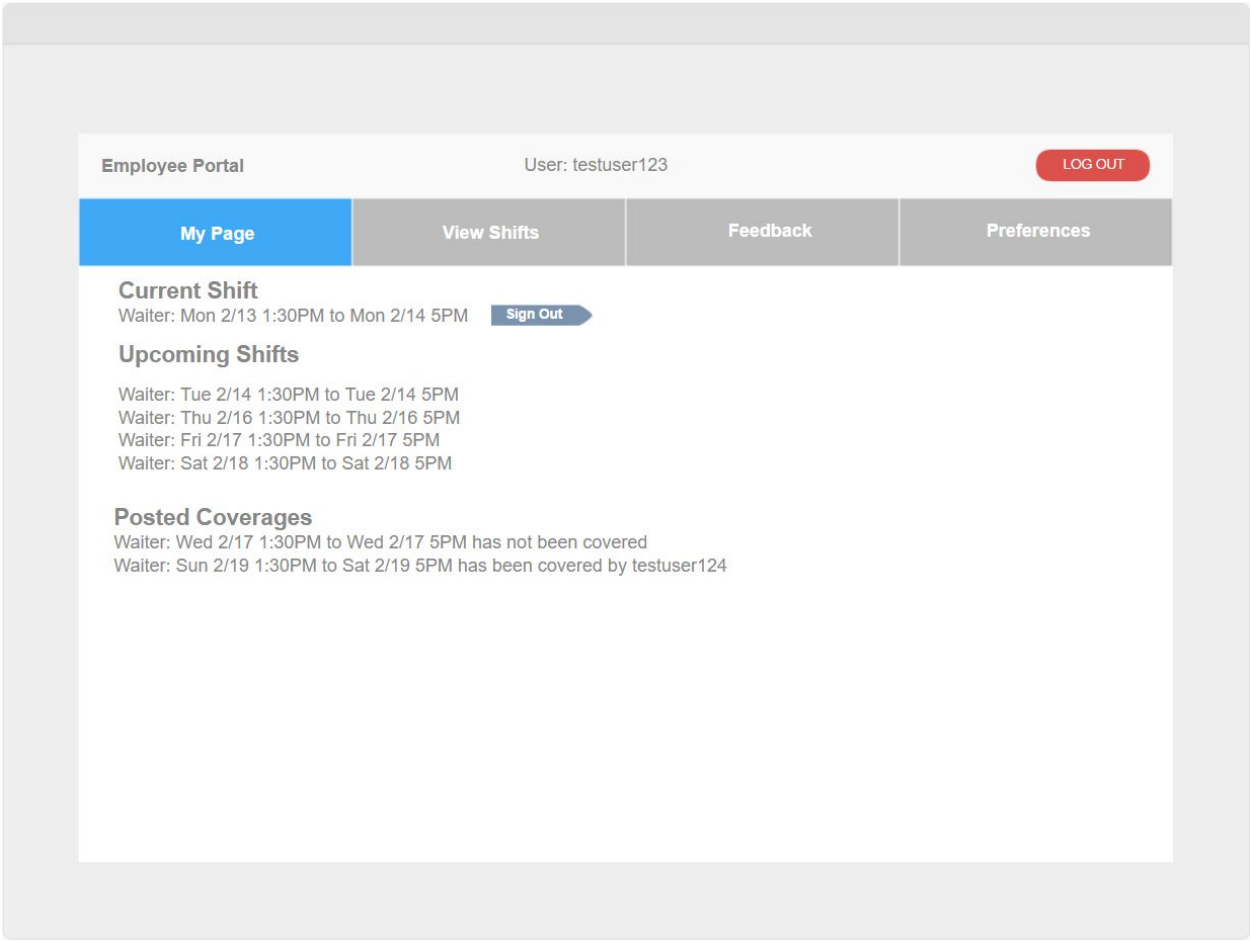


Figure 13: Within the employee portal pages will be available to see which shifts they have signed up for, which shift they may currently be on (if any) as well as requests for others to cover their shift.

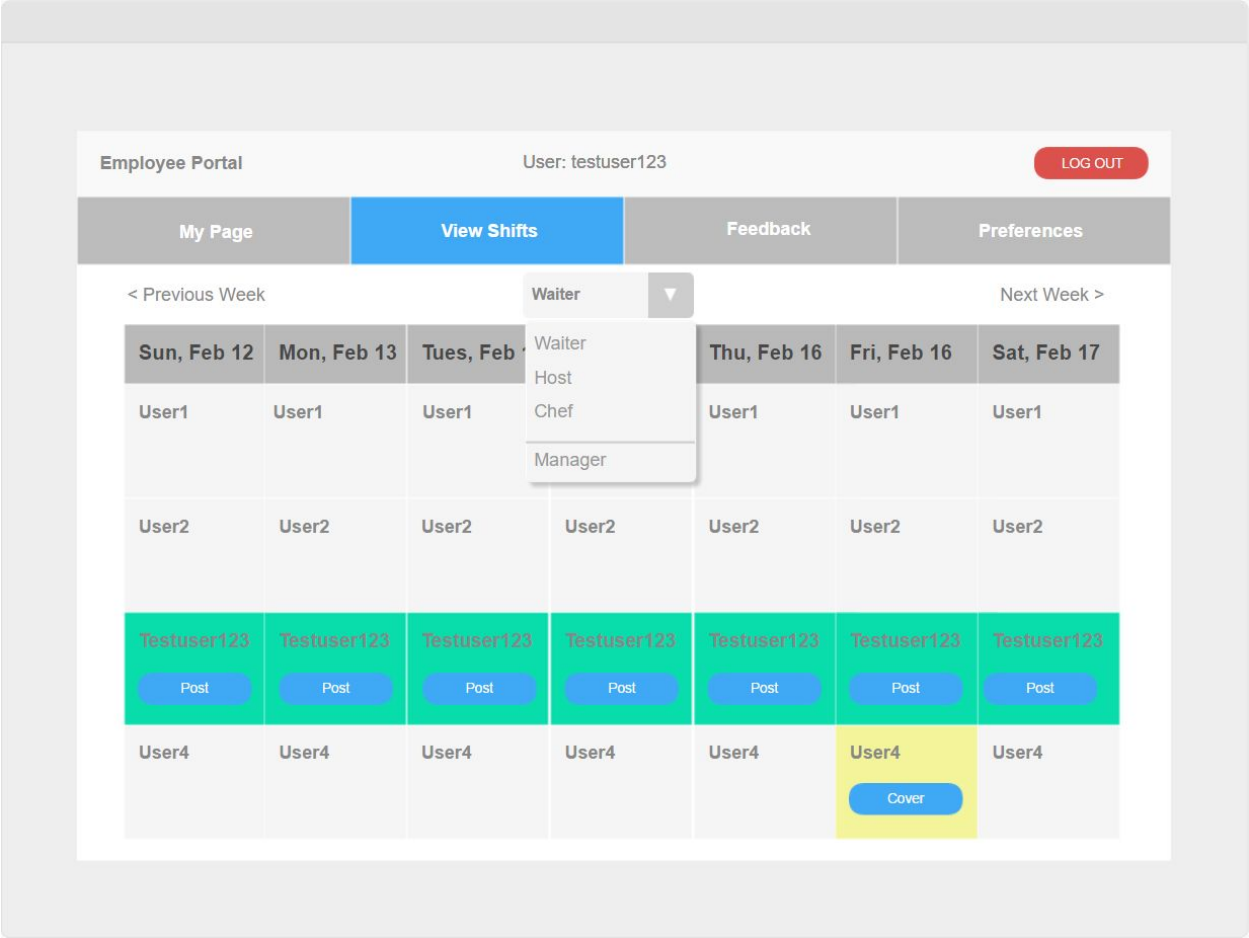


Figure 14: Employees can view the current shifts for their role and who is scheduled for each one. If any shifts are open or need more assistance then they will be able to claim the shifts and the manager(s) will be notified.

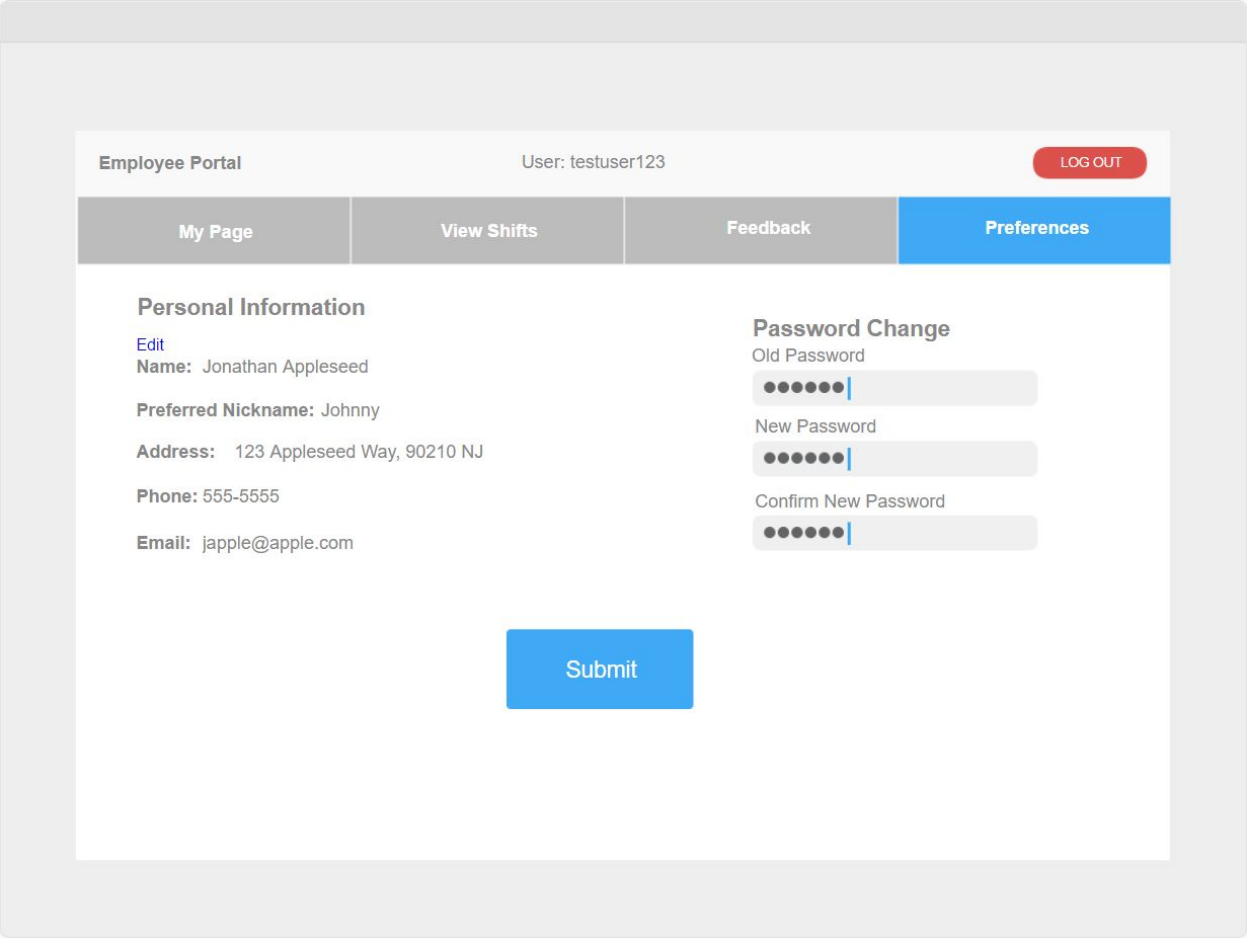


Figure 15: Employees will be able to manage their account. They will be able to set their own password and view the information that the business has on file about them. Any information that is going to be changed will be edited by the manager in person.

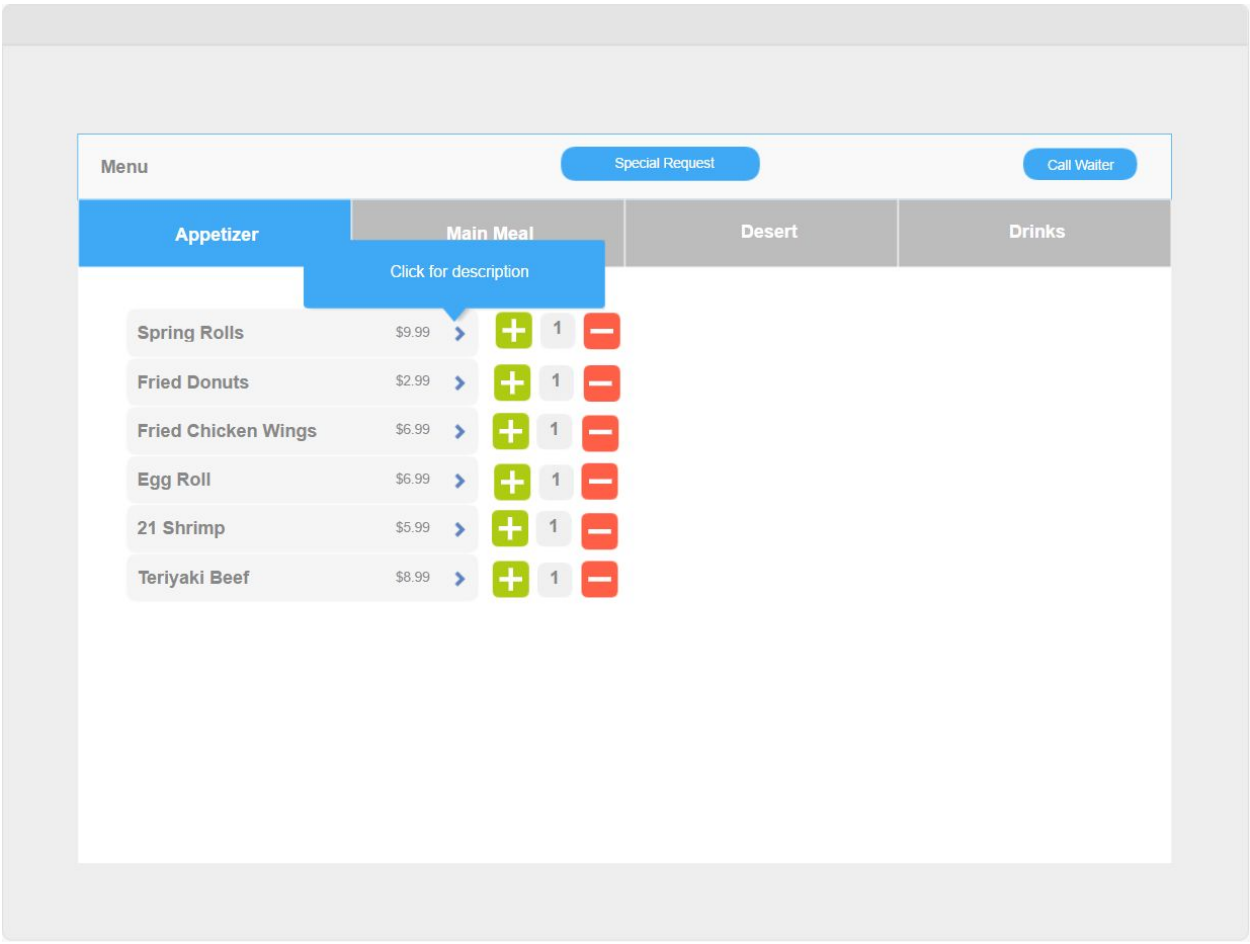


Figure 16: While at their table customers can order food from the menu without having to call the waiter. The menu can be updated daily with specials that the restaurant may be serving or reflect when an ingredient is out of stock. Prices are easily adjusted by management which can be immediately reflected on the menu seen by the customers.

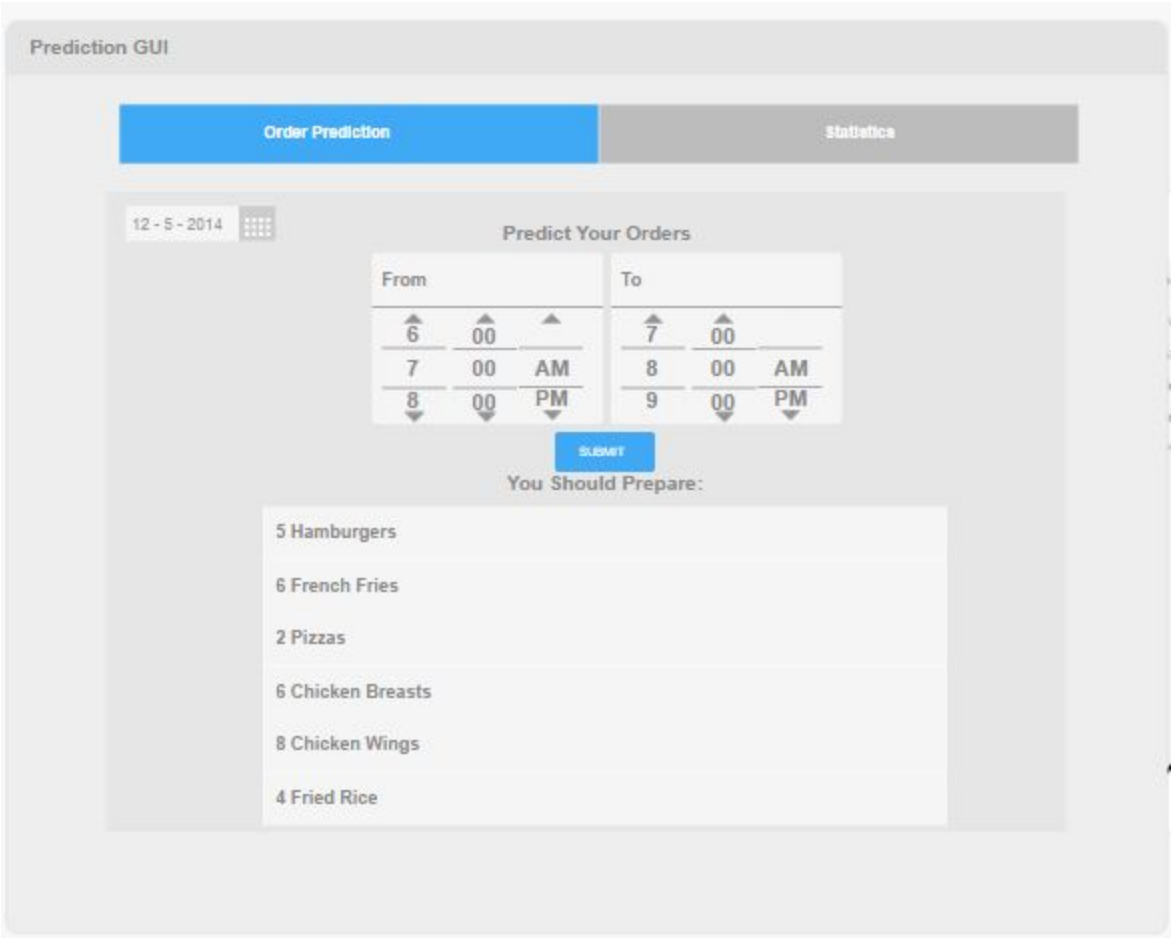


Figure 17: The chef and manager are able to access the prediction interface which will attempt to give a best-guess estimate of the food which will be ordered within a given time period. This gives the chefs a good idea of how much food they need to cook as well as the amount of inventory that the manager may need to order for a given day.

Figure 18: Managers can also get a look at the business statistics which are reported based on different metrics that the restaurant might see. This may include, but is not limited to: The amount of types of different ingredients used over time, the number of orders over time on a given day, and the performance of the prediction algorithms over time.

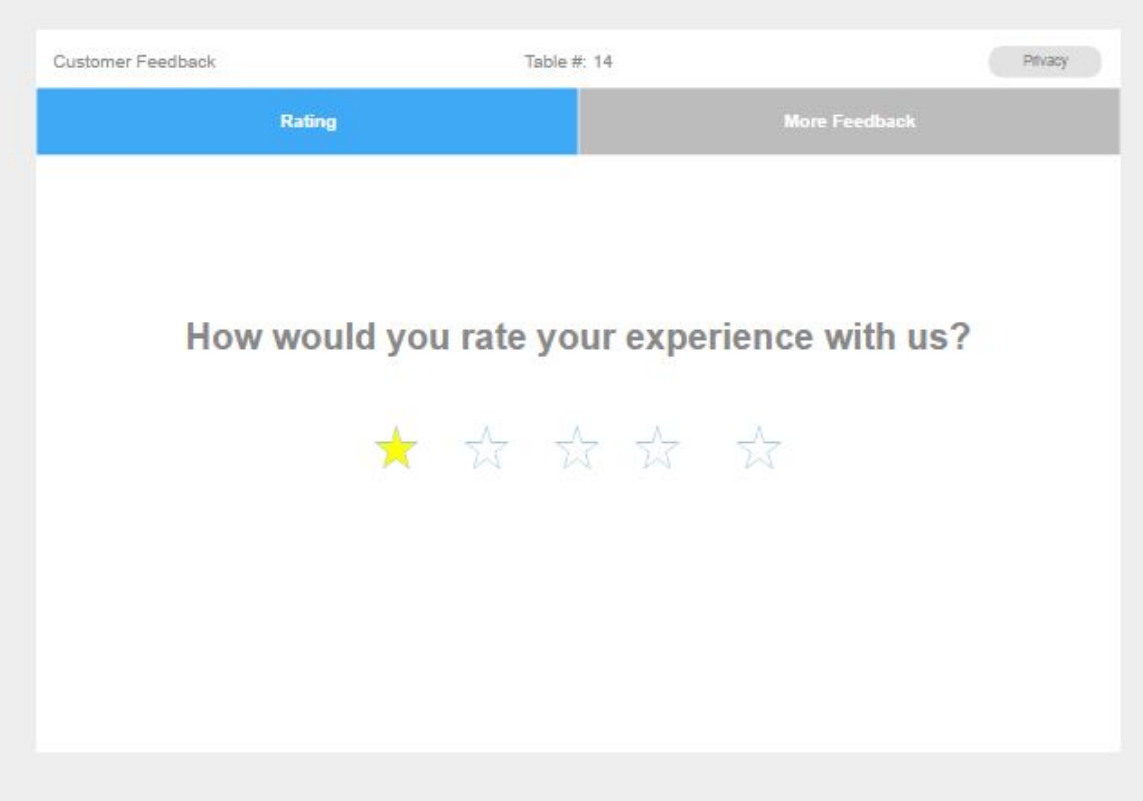


Figure 19: After receiving their order and finishing customers will be presented with a simple screen where they can rate the service they received at the restaurant. This can give managers a general idea of the service which is being provided at given times. This can also be applied to each waiter to see which waiters give the best or worst service to customers.

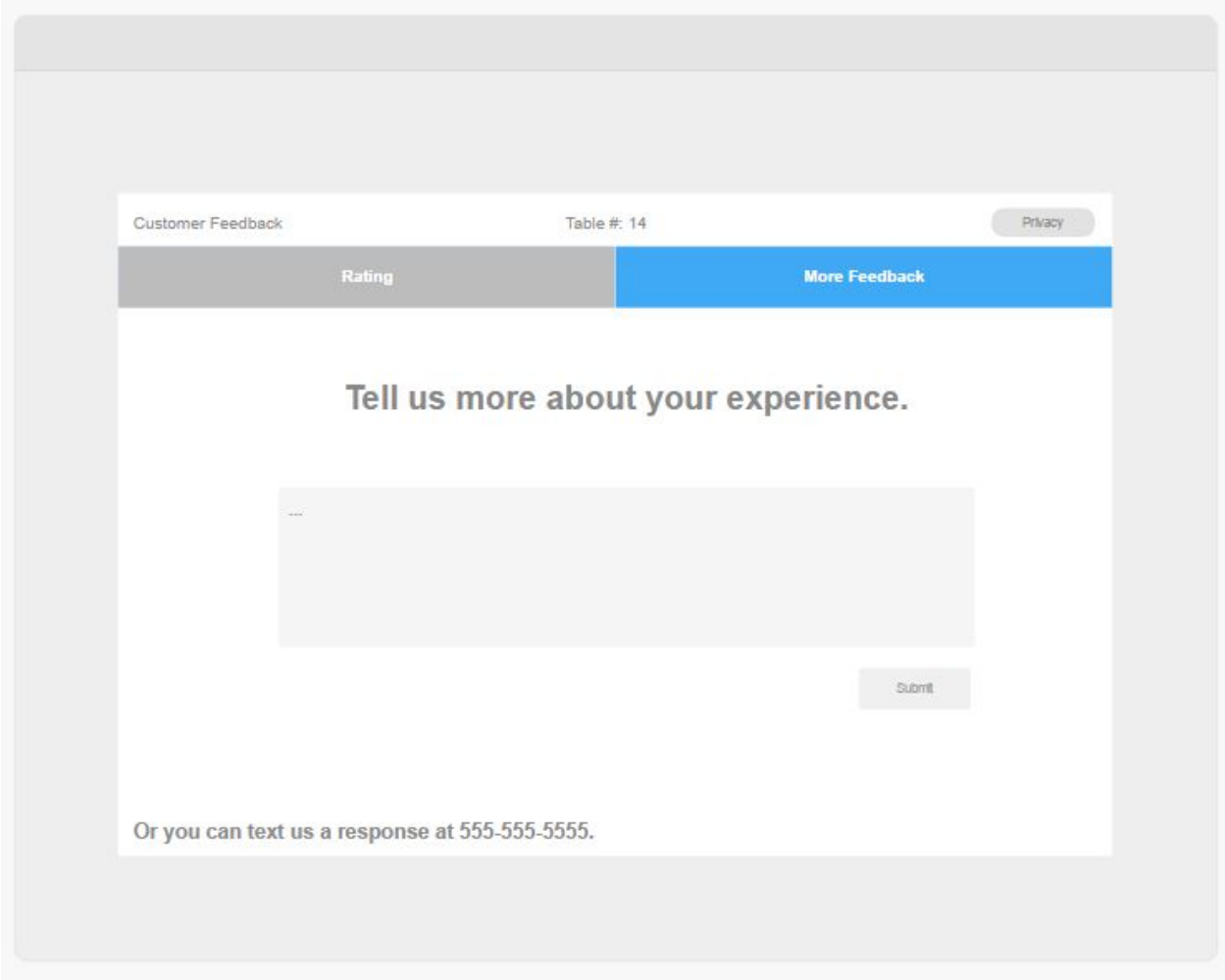


Figure 20: Customer feedback is one of the most important things in a restaurant in order to improve their customer service. They can type their response into the textbox that is provided and then submit their response.

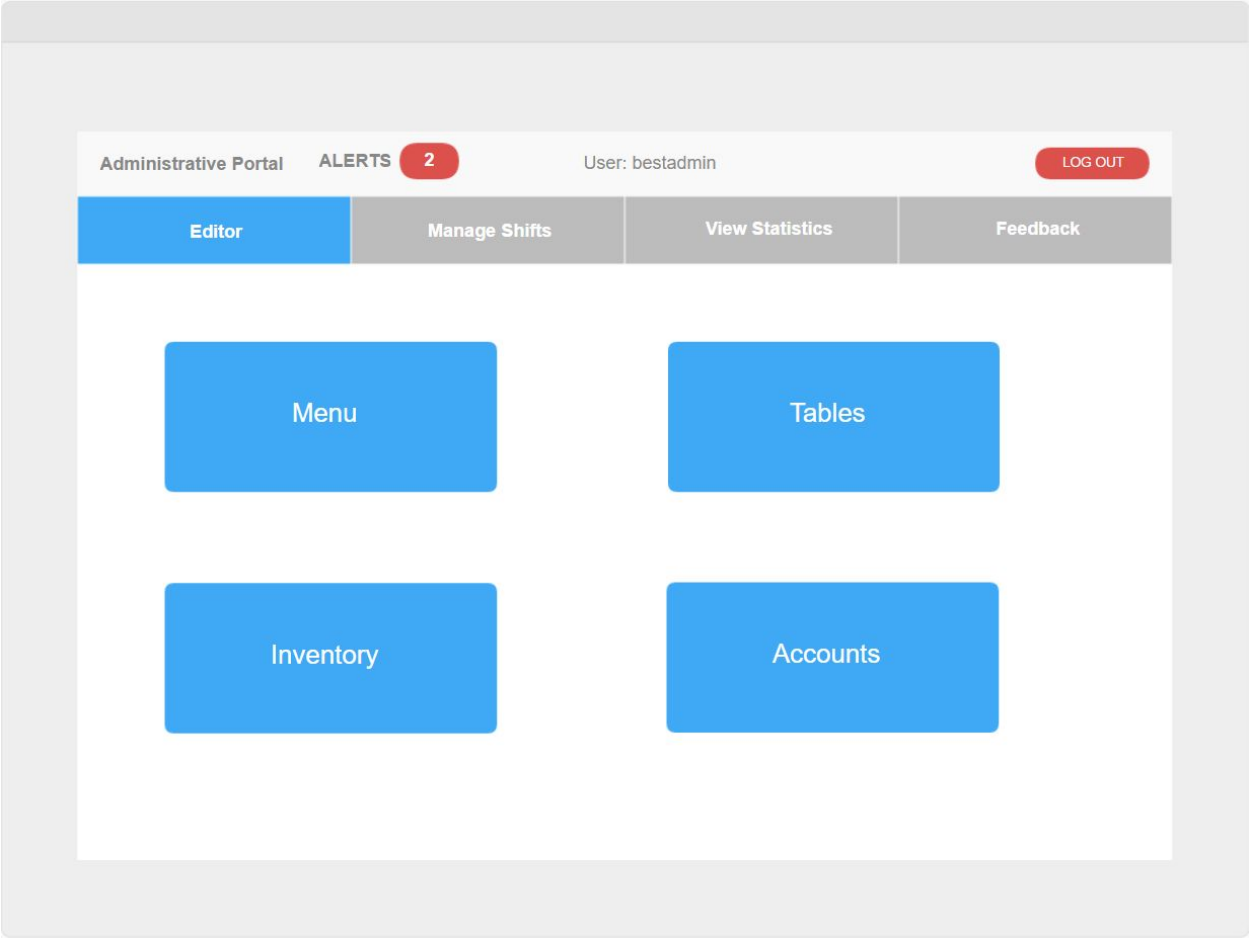


Figure 21: The administrative portal allows managers to edit the menus, the tables, inventory and accounts when they are required to do so. They also can manage shifts when an employee calls in sick or find openings to see if they need to hire more people.

b. User Effort Estimation

Customer/Host reserving a table: Total 2 clicks/presses 10-50 keystrokes

1. Select reservation tab
2. Typing their first name, last name, phone number, selecting date, number of guests, time of reservation.
3. Finalizing reservation by hitting submit

Customer/Host canceling a reservation: Total 3 clicks/presses

1. Select reservation tab
2. Typing their reservation number

3. Hitting “Cancel Reservation”

Customer checking Table Availability: Total 8 - 10 clicks/presses

1. Select Table Availability tab
2. Selecting date, hour, minute, a.m./p.m.

Customer leaving a Rating: Total 2 clicks/presses

1. Select correct tab (Rating)
2. Press appropriate star level

Customer leaving a feedback: Total 3 clicks/presses. 10- 200 keystrokes

1. Select correct table (More Feedback)
2. Selecting the feedback box and typing their response
3. Hitting submit to finalize the submission

Chef view order predictions: Total 2 clicks/presses

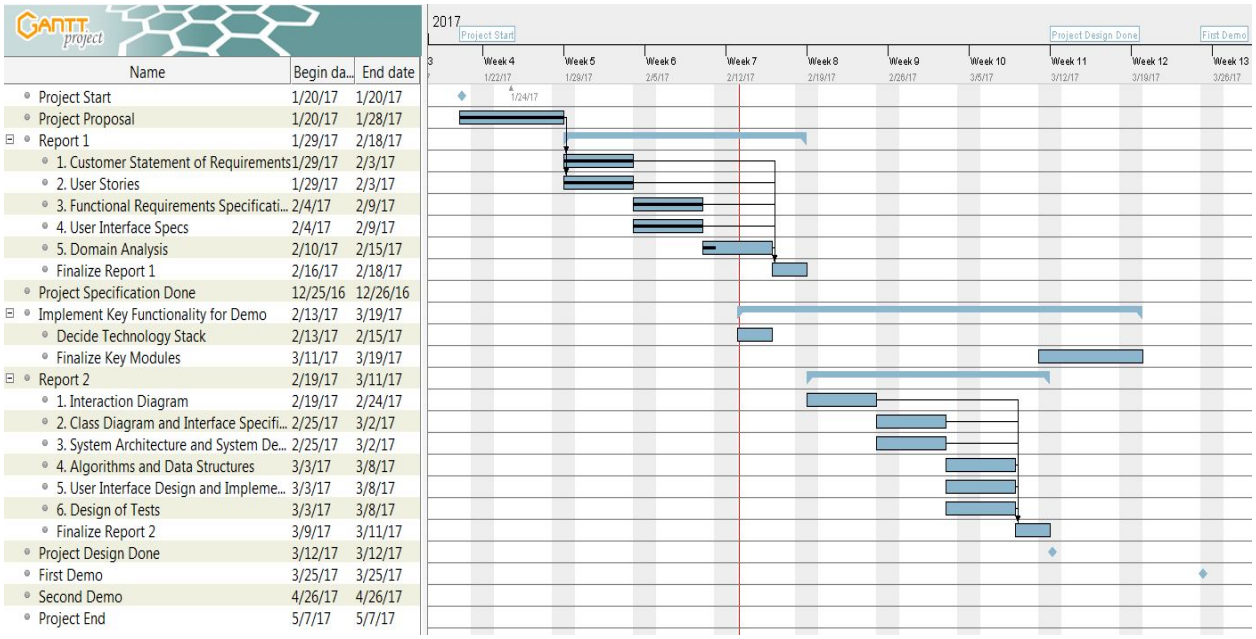
1. Select Predictions section from Chef Dashboard
2. Select correct tab (view predictions)

Manager view statistics: Total 2-3 clicks/presses

1. Select Statistics section from Chef Dashboard
2. Select correct tab
3. Select desired statistic from the dropdown

*Number of keystrokes can vary widely, depending on the type of information being entered.
Lower bound for number of clicks/presses: 22
Upper bound for number of clicks/presses: 25
Lower bound for number of keystrokes: 20
Upper bound for number of keystrokes: 250
On average: clicks/presses account for 14.8% of input, while keystrokes account for 85.2%

6. Plan of Work



Name	Begin da...	End date
Project Start	1/20/17	1/20/17
Project Proposal	1/20/17	1/28/17
Report 1	1/29/17	2/18/17
1. Customer Statement of Requirements	1/29/17	2/3/17
2. User Stories	1/29/17	2/3/17
3. Functional Requirements Specificati...	2/4/17	2/9/17
4. User Interface Specs	2/4/17	2/9/17
5. Domain Analysis	2/10/17	2/15/17
Finalize Report 1	2/16/17	2/18/17
Project Specification Done	12/25/16	12/26/16
Implement Key Functionality for Demo	2/13/17	3/19/17
Decide Technology Stack	2/13/17	2/15/17
Finalize Key Modules	3/11/17	3/19/17
Report 2	2/19/17	3/11/17
1. Interaction Diagram	2/19/17	2/24/17
2. Class Diagram and Interface Specifi...	2/25/17	3/2/17
3. System Architecture and System De...	2/25/17	3/2/17
4. Algorithms and Data Structures	3/3/17	3/8/17
5. User Interface Design and Impleme...	3/3/17	3/8/17
6. Design of Tests	3/3/17	3/8/17
Finalize Report 2	3/9/17	3/11/17
Project Design Done	3/12/17	3/12/17
First Demo	3/25/17	3/25/17
Second Demo	4/26/17	4/26/17
Project End	5/7/17	5/7/17

Short-term:

Subproblem A team - Richard and Zac:

Over the next few weeks, we plan to accomplish:

1. Implementing a configurable dashboard interface
2. Determine details of algorithm analysis for the predictive solutions
    a. Possible using advanced statistical techniques such as multivariate regression analysis, regression support vector machines, artificial neural networks, bayesian regression, or possibly even more.
3. Connecting the voice service to our backend.

Subproblem B team - Ben, Seo Bo, and Jarod:

Over the next few weeks, we plan to accomplish:

1. Implement basic text-message receiving system
2. Brainstorm possible parsing algorithms to analyze received text messages.
3. Design GUI of how feedback analysis will be shown of managers
4. Design basic feedback form

Subproblem C team - Brandon, Jeffrey:

Over the next few weeks, we place to accomplish:

1. Design a basic reservation form and customers can leave the necessary information
2. Implement a basic text message notification system
3. Design a server database model to hold and return required information
4. Brainstorm possible seating algorithms to efficiently seat customers
5. Create a savable graphical interpretation based on restaurant.

Product ownership:

Each team will be responsible for the following features and qualitative property of the subproblem solutions. Throughout the course of the project, the following teams will implement the following functional features and qualitative properties of our project:

Subproblem A team - Richard and Zac:

- A simple easy-to-use interface to view the restaurant business trends
- A predictive service to determine the amount of food that will be needed on a given day.
- An interface to create custom dashboards with analytics based on manager preferences
- Voice commands to report to managers and chefs.

Subproblem B team - Ben, Seo Bo, and Jarod:

- A convenient post-meal survey that customers can complete to provide restaurant feedback.
- Feedback submissions remain confidential via each user being given a unique non-identifiable ID
- Customer receive a confirmation of feedback entry. (maybe include some of the organizing data that was parsed)
- Manager receives a summary and analysis of the feedback included
- Send out notification of action from management to customers who provided relevant feedback

Subproblem C team - Brandon, Jeffrey:

- An efficient and convenient system that allows customers to provide information and reserve a seat.
- Server notification that the table is now reserved as well as text/QR notification that the table is reserved for a guest.
- Manager/Host will receive seating confirmation and reserve the seating to prevent walk-in customers from taking the seat.
- Queueing of walk-in customers depends on the size of the group that comes in, basically table size demand queue.
- Designing a webpage for reservation systems, which implements a graphical interface that allows customers to reserve seating with the knowledge of where they are sitting.

7. References

- [1] - "Statistics on Food Waste in the United States", https://www.nrdc.org/issues/food-waste
- ChefBoyRD Icon “Chef Boy” https://img.clipartfest.com/91a3a64a56686f2093016774a3f0ad63_boy-chef-pizza-chef-boy-clipart_236-236.jpeg