
Software Requirements Specification

for

Restaurant Wizard

Version 1.0 approved

Prepared by Nguyen Ha Phuong - 301147401

Gaeun Kim - 301157178

Albert Bota - 301184986

Prashant Sharma - 301175737

Peiter Djaja - 301189675



May 19th 2021

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Product Scope	1
1.5 References	1
2. Overall Description	2
2.1 Product Perspective	2
2.2 Product Functions	2
2.3 User Classes and Characteristics	2
2.4 Operating Environment	2
2.5 Design and Implementation Constraints	2
2.6 User Documentation	2
2.7 Assumptions and Dependencies	3
3. External Interface Requirements	3
3.1 User Interfaces	3
3.2 Hardware Interfaces	3
3.3 Software Interfaces	3
3.4 Communications Interfaces	3
4. System Features	4
4.1 System Feature 1	4
4.2 System Feature 2 (and so on)	4
5. Other Nonfunctional Requirements	4
5.1 Performance Requirements	4
5.2 Safety Requirements	5
5.3 Security Requirements	5
5.4 Software Quality Attributes	5
5.5 Business Rules	5
6. Other Requirements	5
Appendix A: Glossary	5
Appendix B: Analysis Models	5
Appendix C: To Be Determined List	6

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

The purpose of the Restaurant Wizard App is to facilitate the work of managers, staff, and owners of restaurants while enhancing customer experience. Using the app, customers will be able to access the menu, order food, make payment (tips included) and leave comments. The staff will be able to fulfill orders and monitor the pace of their service. Owners and managers will be able to monitor inventory, review orders, and keep track of overall demand.

The back-end portion of the app will review and sort all orders and thereby keep track of items in high demand. It provides analysts past order data upon which they can base predictions regarding future demand as well as analyses of seasonal fluctuations or prevalent fads. As supplies begin to run out, the managers, using such analysis, can generate orders and forward them to the vendors. In addition, the app also helps service managers to prepare ingredients for items likely to be in demand prior to opening the restaurant any given day.

The app also enables take-out and deliveries.

1.2 Document Conventions

1- Terminology:

Entity	Definitiona
GUI	Graphical User Interface
SDLC	Software Development Life Cycle

1.3 Intended Audience and Reading Suggestions

This document is intended for engineers who will develop the software, restaurant owners who commissioned the software, and staff and managers who will be using the software in its finished form.

Project manager: plans, implements, monitors, and controls the progress of the project until the end. This document will help him or her divide the work in a reasonable manner, directing each portion to an individual or a group. The individual or the group will then be responsible for the completion of the proposed portion within schedule according to the requirements. All the parts in this document are important for him or her to research.

Programmer: participates in all phases of the Software Development Life Cycle (SDLC). This document will help him/her understand the basic purpose of the application and specific requirements of the end users. He or she will use appropriate programming languages to design and build the applications that satisfy end user requirements. He/she needs to read the whole document carefully.

Tester: responsible for testing and verifying the quality and usability of a new product or new feature. This document will help him/her understand the product to be tested, plan test strategy, execute tests and find potential problems. All the contents of this document are important for him or her to read.

Restaurant owner/managers/staff: The restaurant owner will use this application to manage the restaurant business while the managers and staff will use it for their daily operation. This document will help them understand the application features clearly and in detail. They can obtain the information from part 1 (Introduction) and part 2 (Overall Description).

1.4 Product Scope

In a typical restaurant, customers expect a paper menu with a picture, a brief description, and price of each food item. This app will eliminate the use of paper and guide customers through an intuitive and user-friendly order interface. Compared to paper menus, electronic menus will be easier for managers to edit and update (with new pictures or price options). Using mobile-based monitors and alert systems, the app will streamline work of chefs and waitstaff and enhance pace of service. To facilitate managerial tasks, the app will be connected to inventory spreadsheets and a database of paid orders (with special codes for each item). An intelligent analysis of the data will eliminate much of the guesswork involved in predicting demand. This will help inventory managers place appropriate orders to vendors and reduce the probability of a particular product running out. The back-end portion of the app will determine the inventory. The app will record the peak and slow times of the day, month and year. It will also predict the quantity needed to be prepped in the morning prior to the restaurant opening. It will also be able to coordinate pick up and drop-offs by delivery drivers.

1.5 References

Beth Pollock Beth Pollock (2017, June 08). 7 menu design and layout mistakes. Retrieved May 21, 2021, from https://blog.restaurantscanada.org/index.php/2017/06/28/7-menu-design-mistakes/?gclid=Cj0KCQjwhr2FBhDbARIsACjwLo171xfdbBoY2g1i4br8f7nfUQ8Bpr9d4jBvDvAKo73_QiizZ-kmjlcAoSfEALw_wcB

Restaurant menu templates. (n.d.). Retrieved May 21, 2021, from <https://www.musthavemenus.com/category/restaurant-menu.html>

50 Restaurant Menu Designs That Look Better Than Food. (n.d.). Retrieved May 21, 2021, from <https://creativemarket.com/blog/restaurant-menu-designs>

2. Overall Description

2.1 Product Perspective

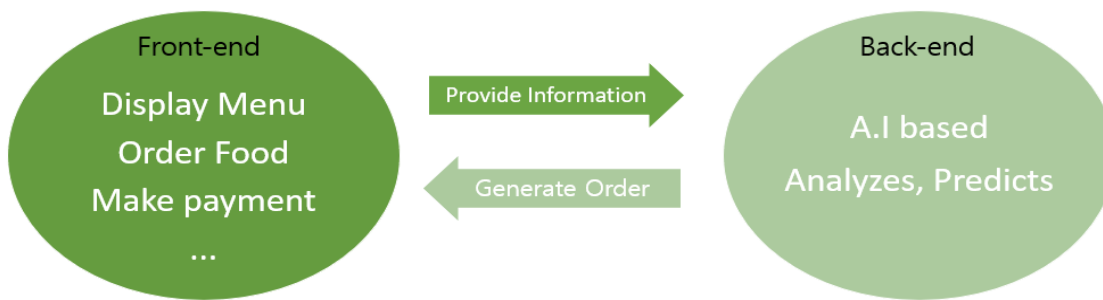
This new software consists of two modules: the front-end and the back-end. This software will interface with the existing restaurant website.

The front-end module displays the menu, and allows customers to order food, leave notes for the chef, make payment, and to comment on the food or service provided. It will also allow the staff to confirm or cancel orders, keep track of orders, and to monitor pace of service.

The back-end module displays the demand data (of food items, inventory) and helps managers generate orders. The software displays customer demand based on past orders, season, and prevalent fashion.

Simple Diagram:

2.2 Product Functions



1. Provide lists and images of the menu
2. Order food
3. Leave note
4. Enable payment
5. Option for take-out and delivery
6. Allow commentary on food and services
7. Estimate and predict the demand for food
8. Help managers generate orders

2.3 User Classes and Characteristics

- **Customers:** will use the software to navigate the menu, order food and beverages, leave comments, and make payments.
- **Restaurant Staff:** will use the software to collect customer input, prepare the order, and monitor speed.
- **Service Managers:** will use the software to track the nature of customer demand and to place appropriate orders to inventory managers. The app will also indicate what and how much to prep for the day.
- **Inventory Managers:** will use the software to keep track of the inventory order and to place appropriate orders to external vendors.
- **Marketing Team:** will use the software to keep track of the fluctuations of customer demand according to season, weather, time of day, fads and fashion etc. They will also use data-driven approach predict the likely nature of demand in the future

- **Human Resource Managers:** will use the software to manage human resources within the restaurant based on the nature of customer demand. This will include plans for hiring, layoffs, transfer of employees etc.
- **Delivery Drivers:** Uses the software to determine which driver can pick up and drop off the fastest based on driver current location and customer destinations.

2.4 Operating Environment

The front-end software, intended for customers and staff, will be mobile based. It supports iOS and Android (for the mobile version operating on Linux). It will run on a variety of browsers such as Firefox and Chrome. It will be coded in C# language. The back-end data will be stored on cloud storage powered by Linux. SQL and Python will be used for data analysis. Python will also be used for modeling and prediction.

2.5 Design and Implementation Constraints

Security consideration:

As customers can pay directly via this application, therefore it is very important to keep customer's payment card information secure. Their address and phone numbers are to be kept secure as well when delivering food items.

2.6 User Documentation

When installed and launched, the app provides brief information and images about itself. When the detailed UI appears, the app will provide a description of the main buttons on the screen. In the home page, there will be a page named FAQ (frequently asked questions), which helps users who may still have questions once the tutorial is finished. Users can reactivate the tutorial if they want to review the information again. If there is an error or unresolved questions, they can contact the service center through the homepage.

2.7 Assumptions and Dependencies

Payment gateway: This application will link with bank cards or e-wallets so that customers can pay online more easily.
Card scanner: Customers can scan their cards to make payment.

3. External Interface Requirements

3.1 User Interfaces

Customer Order Menu



Pork dim sum

Wonton wrappers, pork cabbage, soy sauce

Price: Small (4 pieces) : 6.00\$
Medium (6 pieces) : 8.50\$
Large (10 pieces) : 14.00\$



Pork bun

Pork tenderloin, Six spices, hoisin sauce

Price: Small (2 pieces) : 8.00\$
Medium (4 pieces) : 14.50\$
Large (6 pieces) : 22.00\$



Congee

Beef stock, lotus root, olive oil, long grain rice

Price: Small : 8.00\$
Medium : 10.00\$




+

+

+


Customer review order cart



Pork dim sum

Wonton wrappers, pork cabbage, soy sauce

Size: Medium
Quantity: 1



Congee

Beef stock, lotus root, olive oil, long grain rice

Size: Small
Quantity: 2


8.50\$

16.0\$

Total

24.50\$

Submit order



Add note

Add note

Inventory Manager

Item Names || Remaining Stock || Unit || Last Import Date

1. Beef Ribs	3	kg	2021/07/16
2. Ginger Beef	7	kg	2021/07/18
3. Chicken Feet	4	piece(s)	2021/07/17
4. Onions	5	kg	2021/07/17
5. Garlic	2	kg	2021/07/15

Update

Arrange by Time

Arrange by Material

Arrange by Order Type

Arrange by Number
of Remaining Stock

Confirm

Inventory Manager Restock List

Delivery Order



Restaurant Address: 1280 Markham Rd,
Scarborough, ON M1H 3B4, Canada

Current Time: 2021/07/18 15:37

ETA: 2021/07/18 16:10

Delivery Man: Liam Brown

Order Status: Awaiting Pickup

Personal Information

Name: John Wilson

Destination Address: 941 Progress Ave, Scarborough, ON
M1G 3T8, Canada

Phone Number: +1 123 456 7890

Get delivery

Driver's Command Screen

3.2 Hardware Interfaces

Ipad/tablet that is installed with the application for customers to order and make payment.

Card scanner attached to the tablet device for customers to make payment.

Stylus attached to the tablet device for customers to leave comments or navigate the menu.

Driver will have a large screen tablet for delivery orders.

3.3 Software Interfaces

A GUI system for language control, menu display, cart, and submission confirmation

A GUI payment interface (integrated with third party payment system)

A GUI for service manager to confirm or cancel orders

A GUI for inventory manager to update and manage inventory

A GUI for kitchen staff to monitor progress and alert staff and customers on order status

A GUI for managers review paid orders

A GUI for the delivery drivers to inform them of customer information and the destination address (integrated with third party map system)

3.4 Communications Interfaces

The app will alert delivery drivers. It will summon them to pick up orders and instruct them to deliver items to customers by a certain time.

Use cases			
Use Case Name	List of Related Req ID	Actor(s)	Brief Description
Update Menu	FR02	Service Manager	The manager will log into the system as administrator, and go to the Setting -> Update Menu. The software will present the manager the menu table, and two options: "edit" and "add new item." The manager will select "edit" to remove or update item information (such as name, picture, size, price, availability). Besides, he/she can add a new item by uploading a picture and filling the name, price, size, and availability information. After that, he/she will click the save button, and the system will update the menu.
Update Stock	FR03	Inventory Manager	The actor will log in to the

			<p>system with an inventory manager account. (Only the inventory manager and owner have the edit permission for stock.) The main screen will present a list of items with name, remaining stock, and last import date. It will also offer the option to arrange the inventory according to time, material, order type, and remaining stock. The manager can update the stock by clicking “Update.” In the next interface, the system will present the manager the table that lists ingredients as well as two options: “edit” and “add new item.” The inventory manager will select “edit” to remove or update item information (such as name, quantity). When the manager adds a new item in the inventory, the amount of that item will be set to 0 as default. The amount of ingredients that are added, removed, or updated will be recorded with date and time.</p>
Place Order	FR04	Customer	<p>The customer will click on the button representing the menu item. The software will load the item on a “cart”. The customer can remove items from the cart. The customer can choose the dine-in or take-away option. The customer can add a note for the kitchen</p>

			<p>staff regarding the order. Once the customer is satisfied with his or her choice, he or she can go on the cart, and submit the order.</p>
Confirm Order	FR05	Service Manager	<p>The system will send the notification about the new customer orders to the service manager. Every order requires the service manager's approval in case there are any issues or special requests (e.g. adding or omitting ingredients, ordering one kind of food too much). When the service manager opens the notification, the system will present a list of unconfirmed orders. After checking, the manager will press the "confirm order" button. After that, the system will send the orders to the machine inside the kitchen to be displayed.</p>
Cancel Order	FR06	Service Manager	<p>To cancel orders for rational reasons, the service manager will go to Reviews to see the list of confirmed orders, choose the item customers want to cancel, and press the "Cancel Order" button. The software will ask for confirmation, and prompt for notes in case the manager wishes to record the reason for cancellation. After canceling, the system will send the notification to</p>

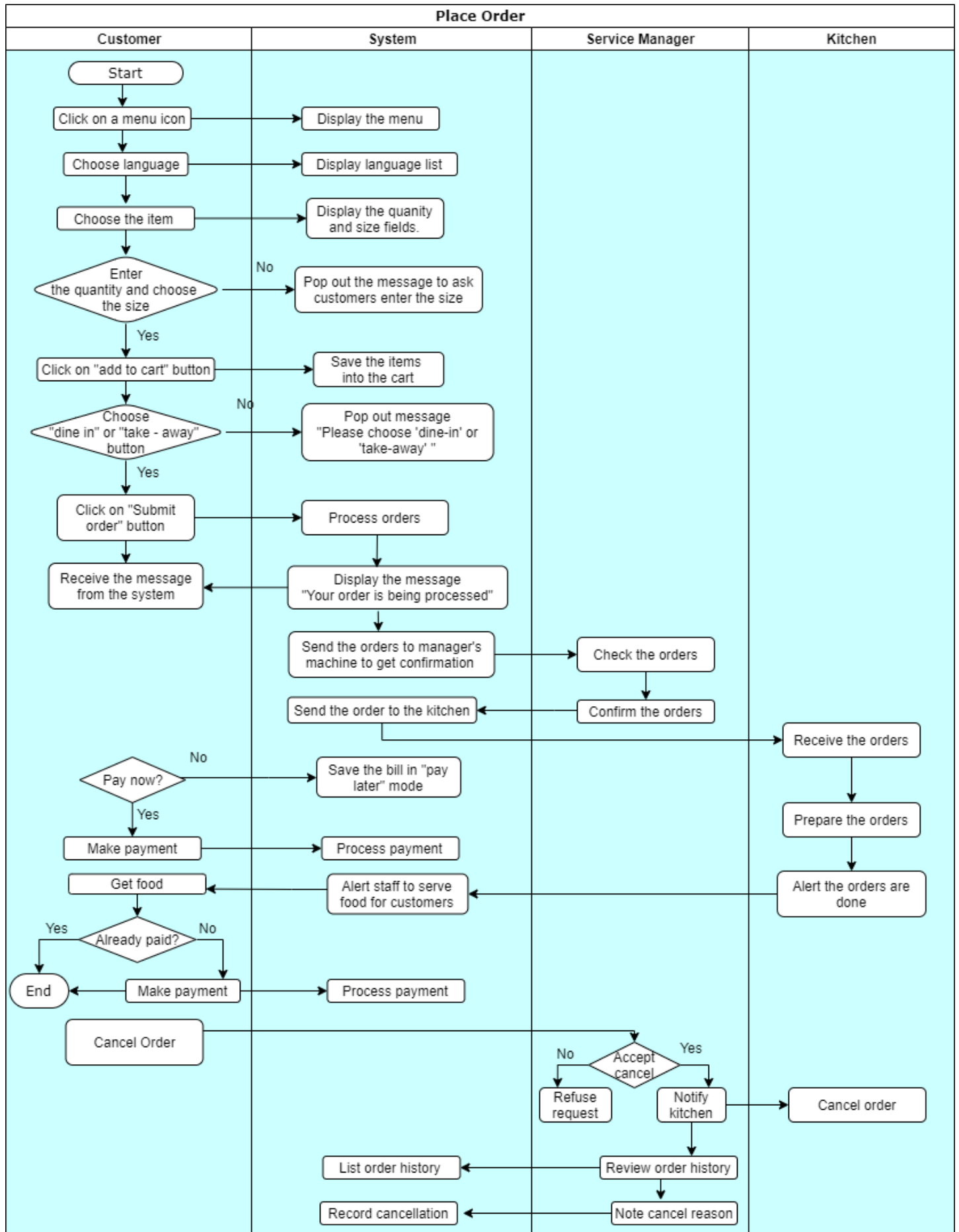
			the kitchen's machine to be printed out automatically.
Choose Language	FR07	Customer, Service Manager	The actor will press the button for the desired language.
Give Feedback	FR08	Customer	After payment, the software will load a feedback screen where the customer can leave comments on food and services. The customer can either type in the textbox, or handwrite the notes using the attached stylus.
Leave Note	FR09	Customer	On the "cart" screen, the customer will be able to add short notes regarding the order ("mild seasoning" or "no onions"). The customer can either type the note in a textbox, or use the attached stylus to handwrite the note.
Process Payment	FR10	Customer, Third Party Payment System	Once the order is submitted, the customer will be given a choice to pay immediately or later. (For take-away or delivery orders, "pay later" choice will be disabled.) If the customer selects "pay now," the software will load the payment screen, and direct the customer to scan a credit or debit card on the card-scanner. If the customer selects the "pay later" button, the software will load a "pay here" button on the screen. Once

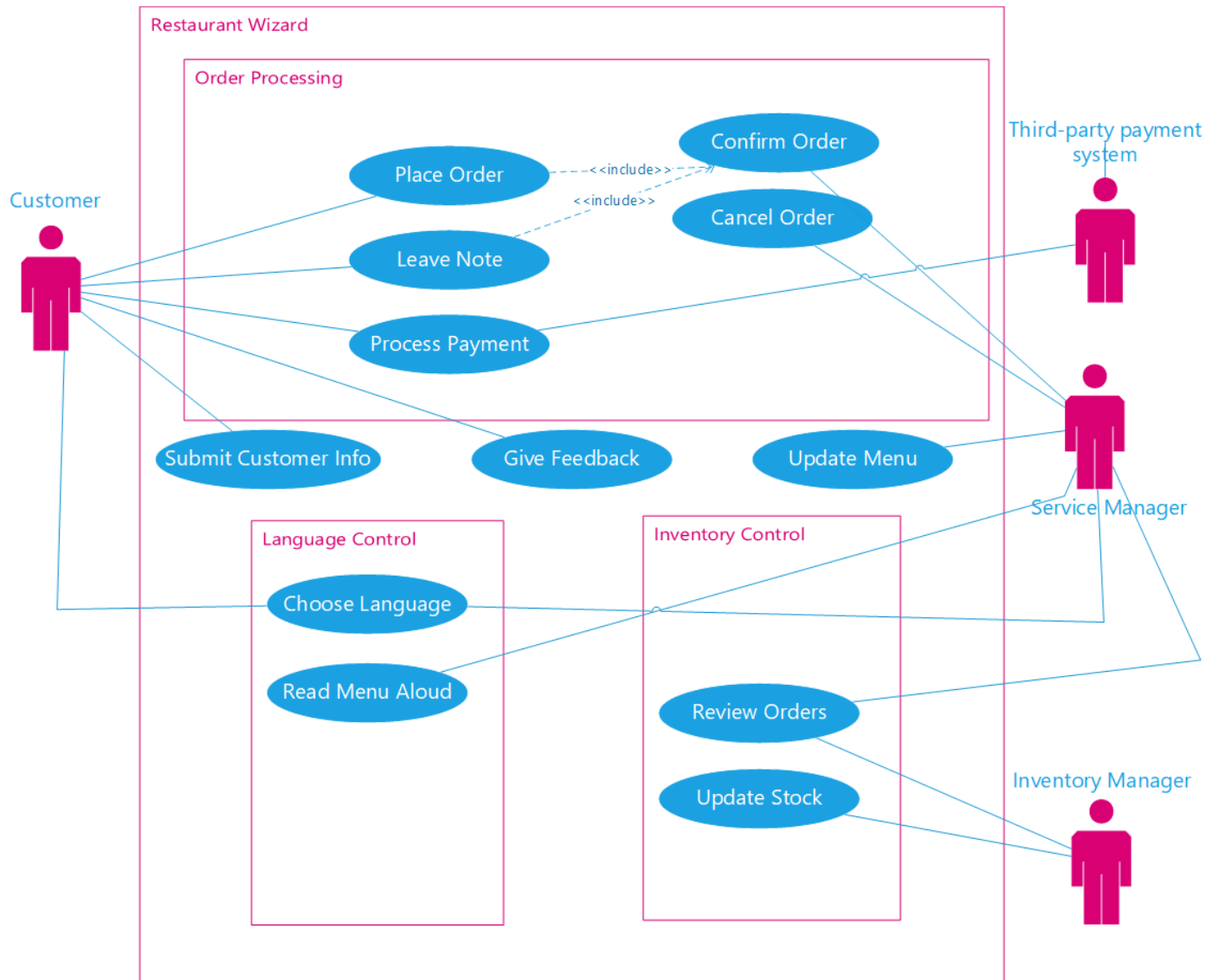
			the customer presses the button at the time of his or her convenience, the software will load the aforementioned payment screen, and direct the customer to scan the card. Once the payment is made, the third party payment system will process the payment.
Review Orders	FR11	Service Manager, Inventory Manager	The actor will press the button “review” to see the list of all the orders for the day, week, and month. The actor will press appropriate buttons to rearrange the list according to order type, time of day, day of week, chronologically, and a number of other ways.
Read Menu Aloud	FR12	Service Manager	The service manager will place the mobile device in front of the blind customer and press the button “Read Aloud.” A computer generated voice will read the menu out loud.
Fill out Customer Information	FR14	Customer	If the customer orders food to be delivered, he or she needs to choose the option “delivery” on the payment screen. Once the order is placed, the system will present the customer an information form. The customer will fill out the date, time, and address for delivery.

4. System Features

USE CASE	
Use Case Name	Place Order
Primary actor (s)	Customers
Goal in context	Order foods and beverages
Preconditions	The system must be fully configured. Customer must click “Submit order”
Trigger	The customer decides to order food
Scenario details	Actor: the Customer
	<ol style="list-style-type: none"> 1. The <u>customer</u> clicks on an <u>icon</u> representing the <u>menu item</u>. 2. The <u>system</u> displays a <u>picture of the item</u> with <u>name, corresponding price for each size (if any) and ingredient details</u>, and prompts the <u>customer</u>: “<u>choose item</u>” 3. The <u>customer</u> clicks “<u>choose item</u>” 4. The <u>customer</u> enters the <u>quantity</u> and chooses the <u>size</u> (if any) of the items then clicks on the “<u>add</u>” button. 5. The <u>system</u> loads the item on the “<u>cart</u>”. 6. The <u>customer</u> can remove an <u>item</u> from the “<u>cart</u>” by going to the “<u>cart</u>” and clicking on the item they want to remove and choose “<u>remove</u>”. 7. The <u>customer</u> can modify the <u>quantity</u> or <u>size</u> of the chosen item (Increase or decrease the quantity, change size). 8. Following steps 1-5, the <u>customer</u> adds another item to the cart. 9. The customer clicks on <u>dine-in</u> or <u>take-away</u> option 10. The <u>customer</u> clicks on “<u>cart</u>” icon 11. The <u>system</u> prompts the <u>customer</u> to leave note for the cook 12. Using an <u>attached stylus</u>, the customer writes “no onion” in the <u>textbox</u>. 13. The <u>customer</u> chooses “<u>finalize order</u>”. 14. The <u>customer</u> clicks “<u>Submit order</u>” 15. The <u>system</u> displays “<u>Your order is being processed</u>”. 16. The <u>system</u> sends the <u>orders</u> to the <u>manager’s machine</u> to get approval. 17. <u>Service manager</u> will check and confirm the <u>orders</u>.
Exceptions	<ol style="list-style-type: none"> 1. If the <u>customer</u> does not enter the <u>quantity</u> of <u>foods</u> or <u>drinks</u> they want to order, <u>the system</u> will automatically add into <u>the cart</u> with the quantity of one.

	<ol style="list-style-type: none">2. If the <u>customer</u> does not choose the <u>size</u> for some <u>dishes</u>, the <u>system</u> will ask them to choose the <u>size</u> of the <u>item</u>.3. If the customer does not choose <u>dine-in</u> or <u>take-away</u> option, <u>the system</u> will regard dine-in as default.4. If the <u>customer</u> does not click the “<u>Submit order</u>” button, their <u>orders</u> will not be processed.
Priority	High priority, first module to be developed
When available	First increment
Frequency of use	Frequent
Channel to actor	Via iPad or tablet device
Secondary Actors	Service Manager





4.1 Restaurant Wizard Menu, Inventory and Delivery Features.

4.1.1 Description and Priority

See 4.1.3 below.

4.1.2 Stimulus/Response Sequences

N/A.

4.1.3 Functional Requirements

[Functional Requirements list](#)

Requirement ID	Requirement title	Short Description	Priority	Requester
FR01	Alert	The system should give the staff a notification when customers submit or cancel their orders.	Expected	Service Manager
FR02	Menu	The system should allow the managers to update the menu (pictures, price, size, availability).	Expected	Service Manager
FR03	Stock	The system should allow the inventory manager to update the amount of ingredients and materials and supplier contact list (used amount, available amount, phone number or email address).	Expected	Inventory Manager
FR04	Place Orders	The system should allow customers to order the food or beverage for both dine-in or take-away.	Expected	Owner
FR05	Confirm Order	The system should allow the service manager to check the orders before sending the orders to the kitchen.	Expected	Service Manager
FR06	Cancel Order	The system should allow the staff to cancel the order (if possible for some rational reasons) and send notification for kitchen staff.	Expected	Service Manager

FR07	Choose language	The system should allow the customers or staff to choose their preferred language when using the application.	Normal	Restaurant Owner
FR08	Feedback	The system should allow the customers to give feedback about the foods or service of the restaurant	Normal	UX/UI specialist
FR09	Note	The system should allow the customers to give some notes (ex: no onion, separate gravy, more chilli,..)	Expected	Service Manager
FR10	Payment	The system should allow the customers to make payment via application after submitting their orders.	Normal	Service Manager
FR11	Review	The system should allow the staff or managers to review the history of paid orders (for all: dine-in, take-away, delivery)	Normal	Inventory Manager
FR12	Accessibility	The system should allow the blind to access the menu by reading it out loud.	Normal	Restaurant Owner
FR13	Advertising	The system should allow the integration of tasteful advertising at the margins of the menu.	Normal	UX/UI expert

FR14	Customer information	Allow the customer to submit name, address, and other information when delivery is chosen.	Normal	Service Manager
------	----------------------	--	--------	-----------------

4.2 System Feature 2 (and so on)

5. Other Nonfunctional Requirements

Nonfunctional Requirements list

<u>Requirement ID</u>	<u>Requirement title</u>	<u>Short Description</u>	<u>Priority</u>	<u>Requester</u>
NFR01	Mobile compatibility	The new system should be compatible with the following mobiles: IOS, Android	Normal	Service Manager
NFR02	Security	The new system should be able to protect customer data (credit/debit card info, bank info). It should be able to resist penetration from malicious entities.	Expected	Database Administrator
NFR03	Aesthetics	The new system should have a colorful look and a simple, intuitive navigation system	Normal	UX/UI expert

NFR04	Data Collection	The new system should collect the following data: user clicks, user orders, order history, user comments, time of day.	Normal	Data Analyst
NFR05	Performance	The new system should be able to process at least 300 orders per day per location, which means 150000 orders per day.	Expected	Owner
NFR06	Reliability	The new system should be able to warn the service manager and backend IT staff if it gets overloaded with orders any time.	Expected	Database Administrator
NFR08	Emergency backup system	Being able to keep the essential system functional in the event that the website is down for any reason.	Normal	Database Administrator
NF09	AI system	Well programmed AI to predict daily demands and track customer's favorite dishes. AI has to be robust.	Normal	Data Analyst

5.1 Performance Requirements

See 5.0 above.

5.2 Safety Requirements

See 5.0 above.

5.3 Security Requirements

See 5.0 above.

5.4 Software Quality Attributes

See 5.0 above.

5.5 Business Rules

N/A

6. Other Requirements

Appendix A: Glossary

N/A

Appendix B: Analysis Models

N/A

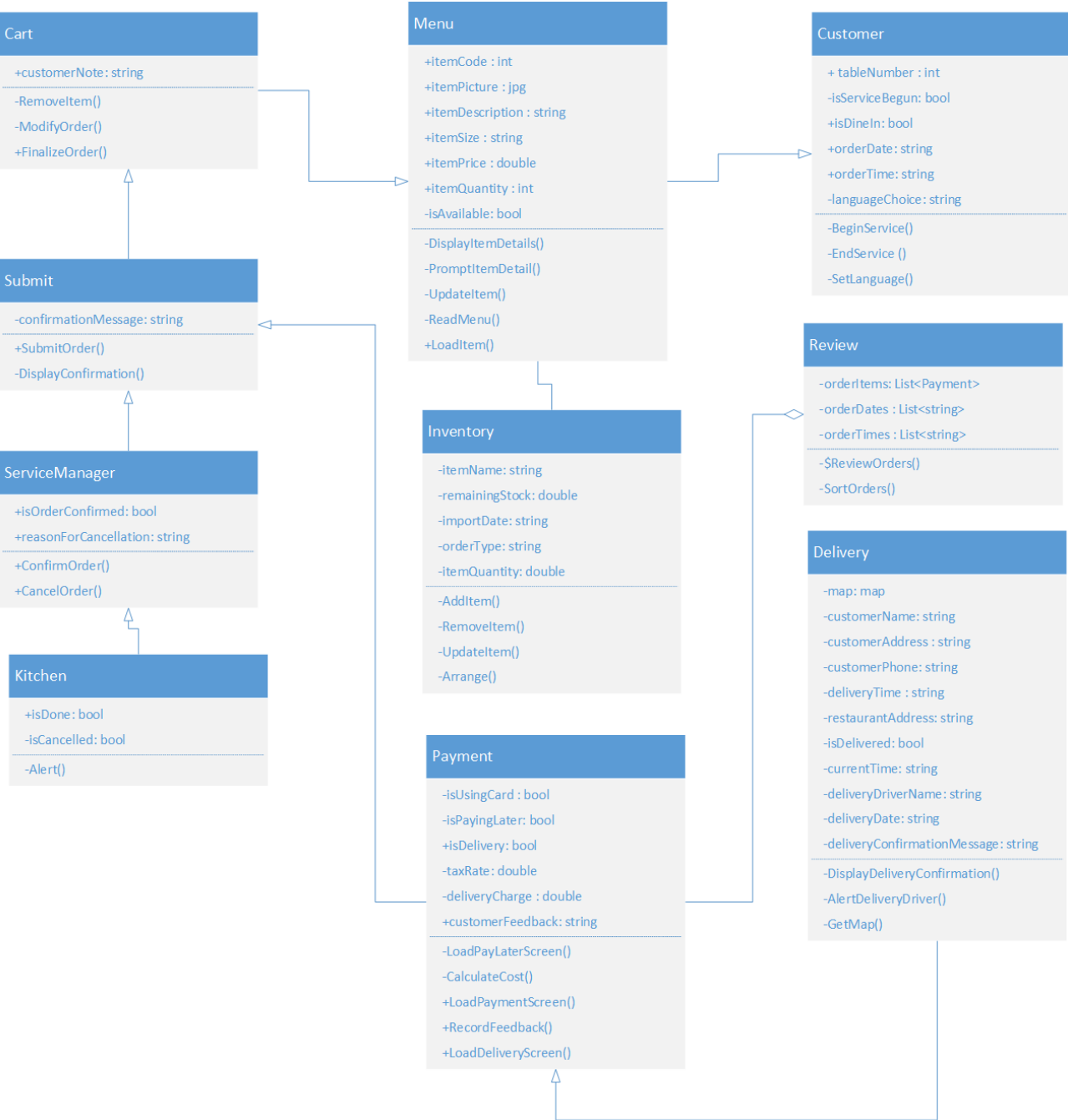
Appendix C: To Be Determined List

N/A

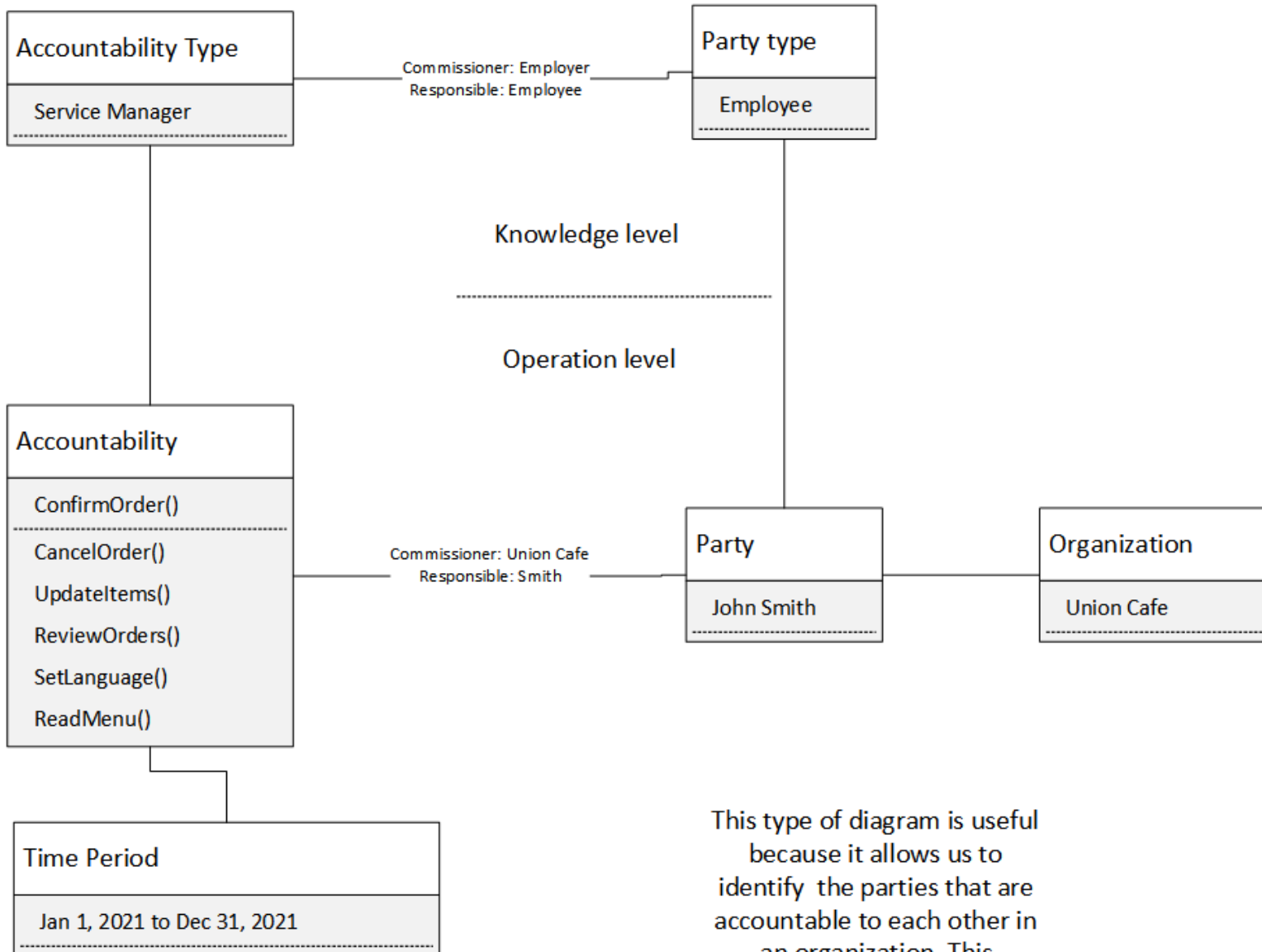
Stakeholder Register					
Stakeholder Name	Stakeholder Position	External/Internal	Stakeholder Contact	Operational/Executive	Interest (high/medium/low)
Balli Batuhan	Restaurant Owner	External	Batuhan.balli566@gmail.com	Operational	high
Johanna Catherine	Service Manager	External	Johanna.cathe347@gmail.com	Operational	high
Maria Isabel	Restaurant Customer	External	Maria48@gmail.com	Operational	medium
Dias Tinoco	Inventory Manager	External	Dias.tino376@gmail.com	Operational	medium
Anand Jashanpreet	Programmer	Internal	Jashan221@gmail.com	Executive	high

Elbo Andrei John	Data Analyst	Internal	Andrei155@gmail.com	Executive	high
Mansur Samirbhai	Database Administrator	Internal	Man.samir54@gmail.com	Executive	medium
Huang Dave	UI/UX expert	Internal	Dave344@gmail.com	Executive	medium

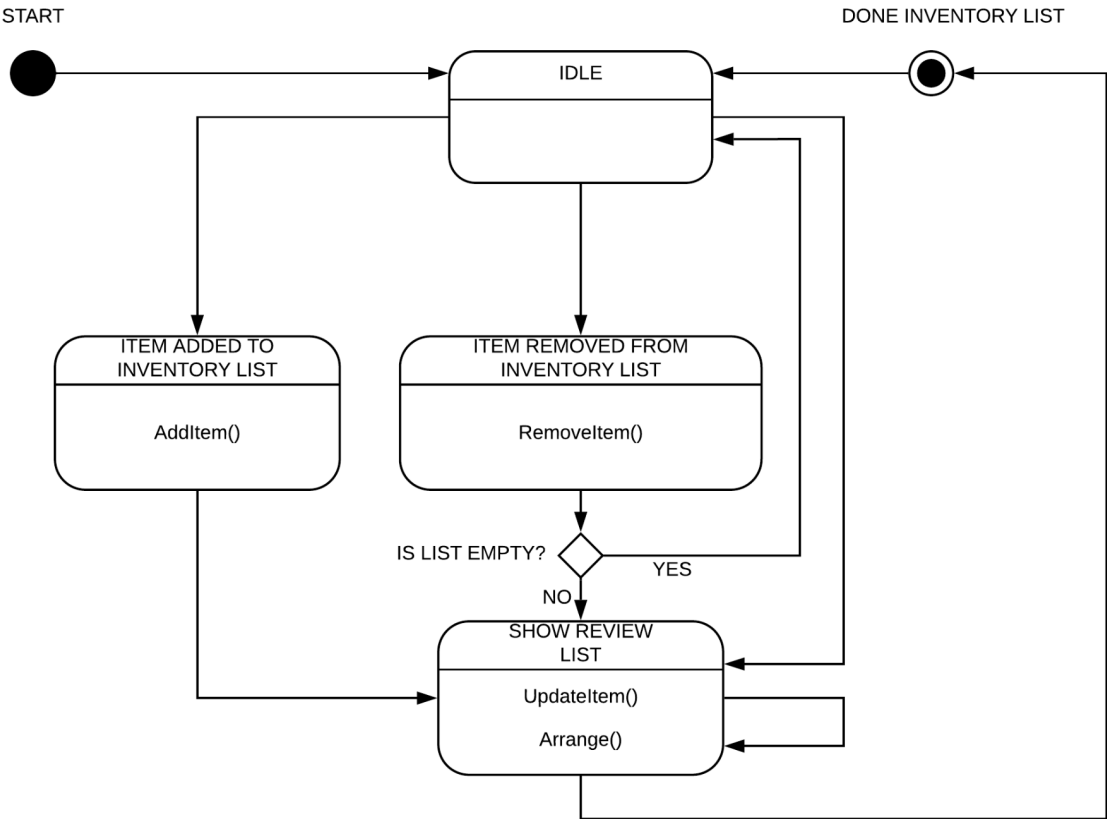
Appendix E



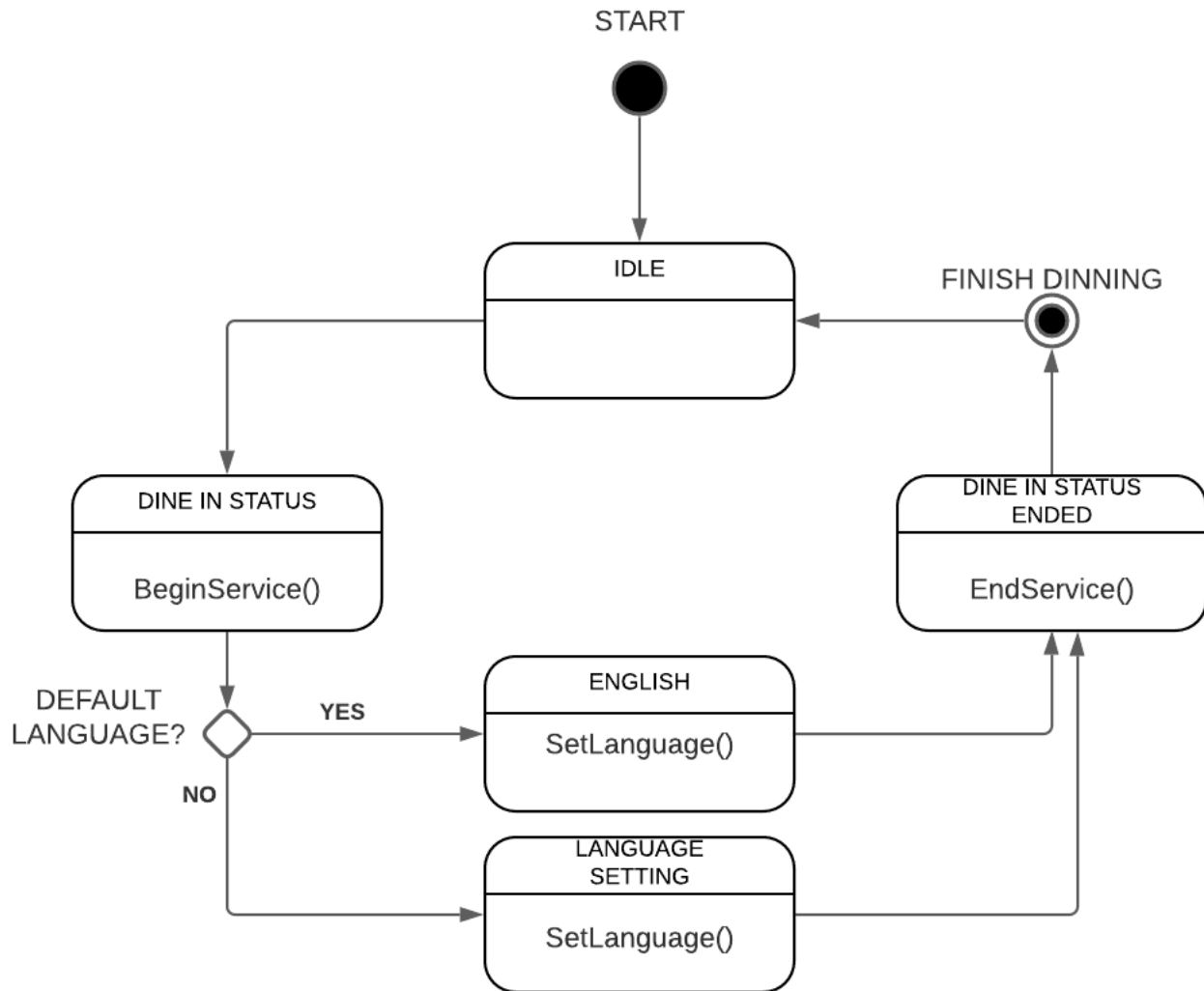
John Smith is a service manager at Union Cafe, Toronto for the entirety of 2021.



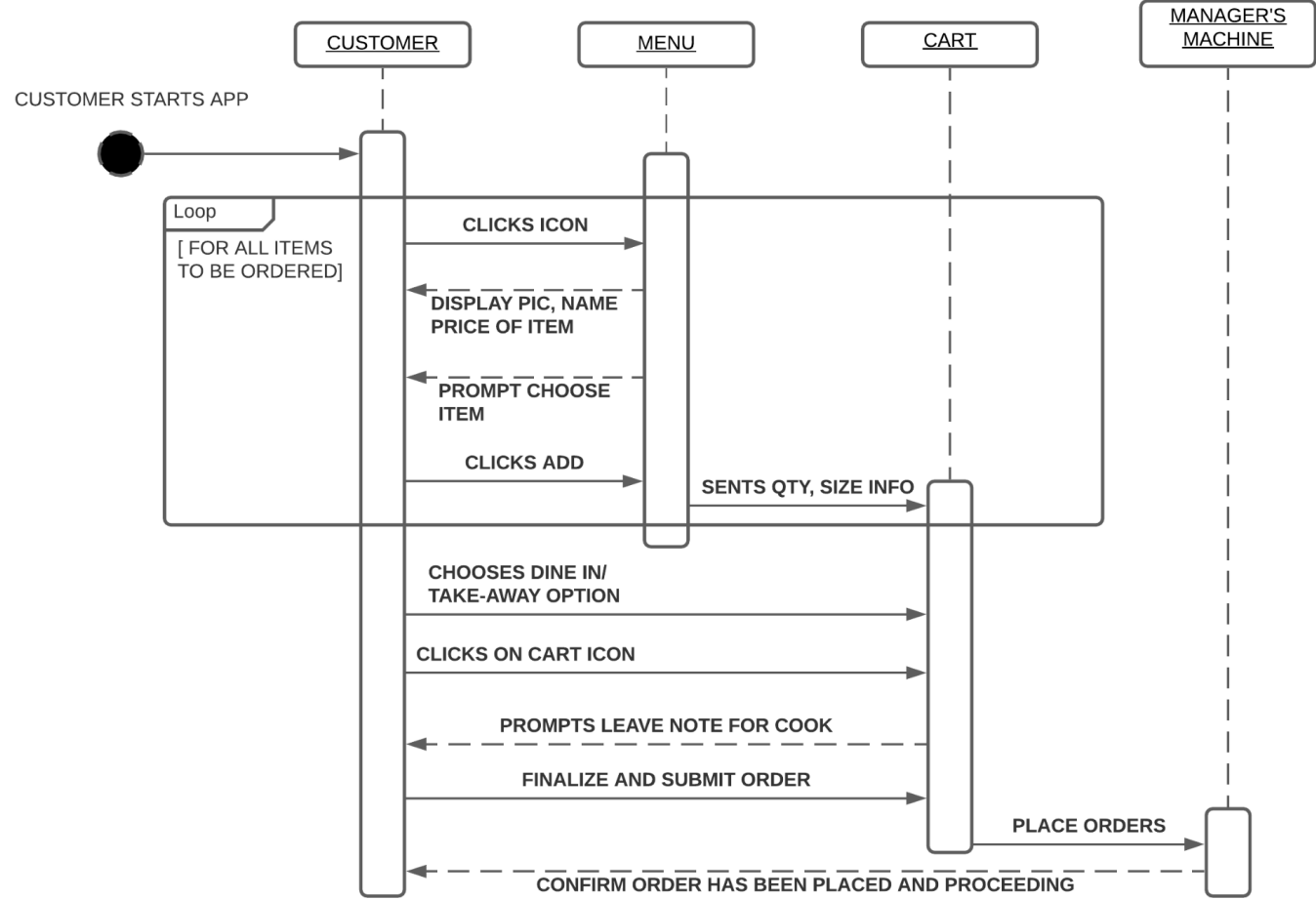
This type of diagram is useful because it allows us to identify the parties that are accountable to each other in an organization. This provides us a framework to model and to measure the interrelationship of actors, objects and entities within any organization in intricate ways.



INVENTORY STATE DIAGRAM



CUSTOMER STATE DIAGRAM



SEQUENCE DIAGRAM: PLACE ORDER USE CASE

Class : Customer		
Description: Begins and ends service; sets language; records date and time		
Responsibility	Collaborator	
	Client	Server
Begins and ends service at a table	+tableNumber -isServiceBegun +isDineIn	-BeginService() -EndService()
Sets language	-languageChoice	-SetLanguage()
Records date/time	+orderDate +orderTime	-Review.ReviewOrders()

Class: Menu		
Description: displays items, prompts for user input, reads the menu aloud for accessibility and allows the service manager to update menu		
Responsibility	Collaborator	
	Client	Server
Displays menu items and details such as picture, name, price, size, availability and ingredients for each item.	+itemCode +itemPicture +itemDescription +itemSize +itemPrice +itemQuantity -isAvailable	-DisplayItem()
Prompts the user to enter size and quantity detail for desired item	+tableNumber +itemCode +itemSize +itemQuantity	-PromptItemDetail()
Reads out the menu items	+itemCode +itemDescription	-ReadMenu()

	+itemPrice	
Update Menu	+itemCode +itemPicture +itemDescription +itemSize +itemPrice +itemQuantity -isAvailable	-UpdateMenu()
Loads and displays item detail on cart	+itemCode +itemPicture +itemDescription +itemSize +itemPrice +itemQuantity	+LoadItem()

Class: Cart		
Description: Loads item on cart, removes item, updates item size or quantity, prompts to leave note, and submits item		
Responsibility	Collaborator	
	Client	Server
Prompts for order modification (Increase or decrease the quantity, change size) Add notes (Ex: Add more gravy, add more chilli,...)	+tableNumber +itemSize +itemQuantity	-ModifyOrder()
Removes item from cart	+itemCode +itemPicture +itemDescription +itemSize +itemPrice +itemQuantity	-RemoveItem()
Calculates total price and prepares order for submission	+itemCode +itemPrice +itemQuantity	+FinalizeOrder

	+tableNumber	
--	--------------	--

Class: Submit		
Description: Submits finalized order, displays confirmation, sends order to service manager, and directs the customer to payment portal		
Responsibility	Collaborator	
	Client	Server
Submits finalized order and sends it to Manager's machine	+itemCode +itemSize +itemQuantity +customerNote +tableNumber	+SubmitOrder()
Displays submission confirmation message and directs customer to payment portal	-confirmationMessage	-DisplayConfirmation()

Class: ServiceManager		
Description: Prompts to either confirm or cancel order, and sends the confirmation or cancellation status to the kitchen and the customer		
Responsibility	Collaborator	
	Client	Server
Confirms Order and notifies the kitchen and the customer of confirmation	+itemCode +itemSize +itemQuantity +customerNote +tableNumber	+ConfirmOrder()
Cancels Order and notifies the kitchen and the customer of cancellation	+itemCode +tableNumber +reasonForCancellation	+CancelOrder()

Class: Kitchen		
Description: Alerts the customer once the order is fulfilled or if the order is cancelled		
Responsibility	Collaborator	
	Client	Server
Alerts staff once the order is ready for the customer	+itemCode +itemSize +itemQuantity +customerNote +tableNumber +isDone	-Alert()
Alerts staff if the order is cancelled	+itemCode -isCancelled +tableNumber +reasonForCancellation	-Alert()

Class: Payment		
Description: Confirms whether the customer is paying now or later, whether the customer is using a card, and whether the customer chooses delivery. Calculates price inclusive tax and service charge. Enables the customer to leave feedback.		
Responsibility	Collaborator	
	Client	Server
Enables the customer to pay later	+tableNumber +isPayingLater +isDineIn	-LoadPayLaterScreen()
Enables the Customer to view the total cost and make payment	+itemCode +itemQuantity +itemPrice -taxRate +isPayingLater +tableNumber -isUsingCard -isDelivery -deliveryCharge +isDineIn	-CalculateCost() +LoadPaymentScreen()

Enables the Customer to leave feedback and passes the feedback to Review class	+tableNumber +customerFeedback	+RecordFeedback()
Prompts for delivery option, customer name, address, delivery time and date	+itemCode +itemQuantity +itemPrice -taxRate +isPayingLater +tableNumber -isUsingCard -isDelivery -deliveryCharge +isDineIn	+LoadDeliveryScreen()

Class: Delivery		
Description: Prompts customer for name, address, delivery time, and date. Displays confirmation message and alerts the driver to deliver the package.		
Responsibility	Collaborator	
	Client	Server
Displays delivery confirmation for customers and the service manager.	+itemCode +itemSize +itemQuantity +itemPrice -deliveryTime -deliveryDate -customerName -customerAddress -customerPhone -isDelivered -deliveryConfirmationMessage	-DisplayDeliveryConfirmation()
Alerts the Delivery Driver	-deliveryTime -deliveryDate -currentTime -customerName -restaurantAddress -customerPhone -customerAddress	-AlertDeliveryDriver()

	-map -deliveryDriverName +itemCode -isDelivered	
Gets map of customer neighborhood	-customerAddress -restaurantAddress	GetMap()

Class: Review		
Description : Lists to total number of paid orders along with associated date and time. Enables sorting and ordering of lists according to various criteria.		
Responsibility	Collaborator	
	Client	Server
List paid orders along with date and time.	+itemCode +itemQuantity +itemPrice +customerFeedback +orderDate +orderTime +isDineIn +isDelivery	-\$ReviewOrders()
Enable Customer to Sort orders to views the list of paid orders easily Ex (by date, by food categories, time,...)	+itemCode +itemQuantity +itemPrice +orderDate +orderTime	-SortOrders()

Class: Inventory		
Description: Lists the stock of supplies and ingredients. Allows the inventory manager to add, update, and remove items from the inventory and records date and time of addition, removal, or update.		
Responsibility	Collaborator	
	Client	Server

Enable Inventory manager to add Item	-itemName -importDate -orderType -itemQuantity	-AddItem()
Enable Inventory manager to remove Item	-itemName -importDate -orderType -itemQuantity	-RemoveItem()
Enable Inventory manager to Update Item	-itemName -importDate -orderType -itemQuantity	-UpdateItem()
Arrange inventory items	-itemName -importDate -orderType -itemQuantity -remainingStock	-Arrange()

This page intentionally left blank

<p>1) In which countries does your restaurant have branches?</p> <p>2) How many people will use this application?</p> <p>3) What languages should this application support?</p> <p>4)What functionalities do you want in the software?</p> <p>5) Will this application be used for both</p>	<p>Restaurant owner</p>	<p>1) Canada only</p> <p>2) Customers, staff, and managers from more than 500 locations that we have throughout the nation. Currently we serve 150,000 customers per day.</p> <p>3) English, French, Chinese, Hindi</p> <p>4) The software should allow the customers to order food, make payment by card (with a card scanner attached on the device). It should also allow the staff to process orders and managers to keep track of the inventory.</p> <p>5) Yes. Customers need to choose the dine-in or take away option before submitting the orders.</p>
---	-------------------------	---

dine-in and take-away orders?		
<ul style="list-style-type: none">1) What information do you need to update the quantity of ingredients?2) How are the ingredients classified?3) How are vendors classified?4) What criteria do you use to place a particular order	Inventory Manager	<ul style="list-style-type: none">1) The number of orders of each food daily; the amount of ingredients used daily; the quantity of ingredients currently; contact list of suppliers. We would like to review the history of aggregate orders by total customers at the end of every month.2) They are classified by: main ingredients, seasonings, auxiliary ingredients, drinks, supplies, and others. The inventory manager will add and update the ingredients by himself or herself.3) They are classified by: dry-goods; meat; alcohol; non-alcoholic beverage; and a few others. We have several vendors for each category.4) The price quoted, the quality of the ingredients, and the past reliability of the vendor. We also try to minimize the distance

<p>to a particular vendor?</p> <p>5) How much stock do you put in the inventory?</p>		<p>from the restaurant because we don't want ingredients to spoil on the way.</p> <p>5) It depends on the season and the kind of food. If the food is likely to spoil, we stock for 3-4 days only. Otherwise we stock for up to 3-4 months.</p>
<p>1) Do customers have to pay before submitting the order?</p> <p>2) How many orders do you process per day in your branch and what is the expected turnaround time per order?</p>	<p>Service Manager</p>	<p>1) No, they should be able to pay after finishing their meal.</p> <p>2) We serve more than 300 orders per day and each order takes 10 minutes to process. That is a lot, and we would like a robust, streamlined notification system to keep track of each and every one.</p>

<p>3) Do staff often review orders before they are sent to the kitchen?</p> <p>4) Will customers be able to cancel orders and get a refund?</p> <p>5) Do you serve drive-thru customers?</p> <p>6) How do you normally update the items and price in the menu?</p>		<p>3) Yes. The staff check whether orders have any problems (e.g. missing ingredients) or special requests before sending orders to the kitchen.</p> <p>4) It depends on whether the food is cooked. And they cannot cancel by themselves, they need to inform the staff and the staff will cancel on their machine.</p> <p>5) Yes, we serve drive-thru customers, and we would need a large touchscreen outside so that customers can order food without entering the restaurant.</p> <p>6) Until now, we have been printing out new menus with updates. It would be convenient if I could do that on the computer without wasting all that paper.</p>
--	--	---

<ul style="list-style-type: none">1) Do you usually find it difficult to use online computer based menus?2) Which menu layout would be most friendly and intuitive?3) Do you need printed menus as well?4) How can we enhance accessibility to customers that are blind or deaf?	Restaurant Customer	<ul style="list-style-type: none">1) Yes, because some of us are computer illiterate. So we would like the layout to be friendly and intuitive.2) Simple and very shallow branching so customers will not get lost. I would also like to give them special instructions regarding food (e.g. no onion) without too much inconvenience.3) Yes, just in case the customer is computer illiterate or if the network is not working. We would like enhanced accessibility4) For the blind, the menu will be read out loud. Deaf and blind will need printed braille menus.
---	---------------------	---

5) Do you want a delivery option as well?		5) Yes. I would like to give them an address where they can deliver the food items.
---	--	---

<ol style="list-style-type: none">1) What should be the overall “look and feel” of the menu app?2) Can we integrate advertising within the menu app?3) Will the customers be able to leave comments and how convenient will the process be?4) Will customers be able to know what kinds of	UX/UI Expert	<ol style="list-style-type: none">1) It should have a colorful look, with generous whitespace for an airy, spacious feel.2) Tasteful advertising can be placed in the margins of the menu. Separate tab about the company can be created for the customers to browse.3) Customers can easily navigate the menu using a touchscreen and leave star rating both for food and services provided. They can use an attached stylus to leave handwritten notes as well.4) Managers will be able to mark which foods are best sellers when they update the menu on the system.
---	--------------	--

<p>food are best sellers?</p> <p>5) Will customers be able to know whether the foods are not available or sold out?</p> <p>6) How else can we enhance the menu app?</p>		<p>5) If the foods are not available or sold out, the managers will update the status of the food on the system.</p> <p>6) We can enable customers to share menus, pictures, and comments in social media.</p>
<p>1) How do you plan to gather the statistical data needed?</p> <p>2) What programming</p>	Data Analyst	<p>1) I will implement data gathering programs and insert them into the app</p> <p>2) Python and R</p>

languages will you be using?		
3) Database program you will be using?		3) SQL, NoSQL
4) What kinds of patterns will you be looking for in the data?		4) I would like to find out the variation of demand in food items according to season, weather, and time of day. I will determine peak times and low times every day, month, and year. I will be using this data to predict future demand and generate inventories.
5) How much data would you need to start?		5) We would need enough data for AI algorithms to generate patterns. These data include millions of clicks, thousands of comments, and hundreds of thousands of orders. That means several terabytes of data

<p>1) What programming language is appropriate for this project?</p> <p>2) What operating system environment is appropriate for this project?</p> <p>3) What process model is appropriate for this project?</p>	<p>Programmer</p>	<p>1) C#, Javascript, and ASP.NET framework for web-based client-side and server-side scripting.</p> <p>2) Both Android and iOS. iOS is appropriate if the program is to run mainly on iPads.</p> <p>3) Prototyping process model with incremental delivery. The first prototype can include the core user interface applications (menu, service progress monitor, inventory app etc) along with the payment system. The second prototype will allow the software to store data in the cloud. The third prototype can include the analytical component that allows intelligent use of data.</p>
---	-------------------	---

<p>4) Will you be programming the backend as well?</p> <p>5) How long will it take you to get the first, second, and the third prototype to working state?</p>		<p>4) Yes, and I will use C# .NET platform for the project.</p> <p>5) 3 to 4 weeks for each prototype.</p>
--	--	--

<p>1) Where will the data be warehoused?</p> <p>2) What database system will be used?</p> <p>3) Are you concerned that the system might overload or shut down and the service might be disrupted?</p> <p>4) What is your main concern regarding data security?</p>	<p>Database Administrator</p>	<p>1) AWS Cloud services.</p> <p>2) MongoDB</p> <p>3) It is a concern. We might need to shut down the system for maintenance or (in the event of malicious attack) repair. A backup system should be in place so that we can keep the essential restaurant functions running during this time. We would also warn the service managers of possible disruptions and alert them that they might have to operate on a reduced capacity.</p> <p>4) Denial of service attacks, and various forms of unauthorized data infiltration and destruction.</p>
--	-------------------------------	--

5) What hardware needs to be purchased?		5) Network hardware such as switches, routers, firewalls, servers, backup servers.
---	--	--

