Software Requirements Specification

for

Cafeteria Ordering System

Prepared by

Jason Ebueng
K. Nguyen
K. Ngo
T. Yau

Table of the Content

1. Introduction

1.1 Purpose

This SRS document outlines the functional and non-functional requirements for Process Impact's Cafeteria Ordering System version 1.0. It is intended to be used by the software development team for implementation purposes and by the quality assurance team for verification purposes. Other stakeholders may use this document for verification purposes as well.

1.2 Scope

1. The software to be implemented is Process Impact's Cafeteria Ordering System.
2. The system will only be used in Process Impact's main campus in Clackamas, Oregon.
3. The Cafeteria Ordering System will
    - 3.1. Grant access to create, modify and delete cafeteria menus to administrators and update services
    - 3.2. Grant employees the ability to order meals from the company cafeteria
    - 3.3. Grant employees the ability to order meals from local participating restaurants
    - 3.4. Trigger the timely delivery of restaurant orders to designated locations on the Process Impact campus
    - 3.5. Trigger the timely delivery of company cafeteria orders to designated locations on the Process Impact campus if requested
    - 3.6. Notify users of order status
    - 3.7. Notify users of errors
    - 3.8. Update its menu listings when
        - 3.8.1. Order items are unavailable or out of stock
        - 3.8.2. Order items are available
        - 3.8.3. New order items are updated
    - 3.9. Allow electronic payments as well as refunds

3.10. Produce receipts listing user order information

3.11. Produce recipes and ingredient lists for custom meals from the company cafeteria

1.3 Definitions, acronyms, and abbreviations

1. COS refers to the Cafeteria Ordering System.
2. In this document, “the system” refers to the Cafeteria Ordering System.

1.4 References

1. *Vision and Scope Document for Cafeteria Ordering System*
2. *Process Impact User Interface Design*
3. *Use-Case Template*

1.5 Overview

The next sections will describe the system’s features, capabilities, characteristics, and constraints.

- Section 2 will provide an overall description for the system.
- Section 3 will outline the use cases for the system.
- Section 4 will detail the external interface requirements for the system.
- Section 5 will describe other non-functional requirements of the system.
- Section 6 will include any other requirements not listed in the previous sections.

2. Overall Description

2.1 Product Perspective

The application that we are envisioning is a cafeteria ordering software system that will essentially improve the food ordering process of our company. It will allow employees to make more efficient use of their time, mitigate the cafeteria operating costs, and reduce the amount of food waste coming out of the cafeteria.

This idea for a cafeteria ordering system originated from various discussions regarding the amount of lunch time employees spend not actually eating lunch. Extensive research has shown that employees spend approximately a third of their time on activities that would be rendered unnecessary by a computerized system, such as walking to and from the cafeteria, selecting meals in person, and paying for the meals at the physical location. In fact, a good proportion of employees prefer to dine out, spending almost twice the amount of time including transportation time and the time needed for the food to be served. In addition to the inefficient usage of time, there is a significant amount of food that gets thrown away, particularly unpurchased foods. Likewise there are often times when employees do not get the food selection of their choice due to either popular demand for a particular menu item or simply the cafeteria running out of specific food items. Some employees have the correct idea of calling the cafeteria directly so that they can avoid all the hassle of ordering in person and simply wait for the food to be ready to pick up.

Our cafeteria order system builds off of the previous points by encouraging employees to order food remotely, through a convenient, user-friendly online system specifically designed to facilitate the food ordering process and help employees save their valuable time. Therefore, we consider this application as an augmentation of the over-the-phone method and a substitute to in-person food ordering, but not necessarily an absolute replacement for in-person food ordering. Rather than ordering through phone which might not be the most dependable method of ordering (cafeteria employees may mishear the orders over the phone especially since the cafeteria is

likely a noisy social setting), the cafeteria ordering system will retain the remote nature of over-the-phone ordering but also possess various features to make ordering food a simpler task. Though the cafeteria ordering system will not be enforced as the "only" option, we highly encourage employees to utilize the system as often as possible not only to make their lives easier, but also for them to quickly gain familiarity with the system.

The cafeteria ordering software application will be integrated into the larger, employee-portal system that we currently have in place, which also houses other components such as a billing/expense system, order history, diet journal (as part of the company's healthy lifestyle initiative) and some other applications commonly used by employees such as an online chat/status update tool. All the transactions that occur through the cafeteria ordering system may be tracked and then sent to the billing system where employees can keep track of their weekly food expenses, as well as allow them to directly deduct food expenses from their payroll (pre-tax) rather than having to pay out of pocket or via credit card. The cafeteria ordering system also produces data on the food orders of each employee and sent to the ordering history system, where employees can view visual representations of their dining patterns and learn about their eating habits. The system will also generate nutritional data that will be visible in the employee health profile system, where employees can keep track of their nutritional intake and make informed decisions about their lunch selections. Yet another external system that the cafeteria ordering system may interface with is the company chat/workplace communication tool, where employees may choose to automatically share their food selections via a status update tool, attracting the attention of their co-workers to an available menu item. That way employees might be more likely to order the same item and so the cafeteria may produce menu items in larger batches and minimize waste.

2.2 Product Features

The cafeteria ordering system contains a variety of major product features that will help make ordering lunch, dinner, or any other meal much easier. The following is a list of major features that will be a part of the system (which will be described in further detail later in the use cases section) and will be exercisable by the users:

1) The ability to order a meal from the cafeteria menu either for pickup or for delivery
2) The ability to order from restaurants in the local area for delivery
3) The ability to perform CRUD (create, read/view, update/modify, delete) operations on meal service subscriptions
4) The ability to register for various meal paying options
5) The ability to request the delivery of a meal
6) The ability to perform CRUD (create, read/view, update/modify, delete) operations on cafeteria menus
7) The ability to order custom meals outside of the cafeteria menu
8) The ability to produce recipes and ingredient lists for customized meals
9) The ability to save specific menu items into a favorites list for convenient access
10) The ability to search for and acquire discount coupons for local restaurants
11) The ability for admins to manage a rewards program to incentivize employees to use the cafeteria ordering system

2.3 User Classes and Characteristics

Favored User Class: *Employee*

Characteristics of this user class:

- Frequent usage of cafeteria ordering system for all of its functionalities, especially during lunch hours and possibly every day of the week
- High number of employees would potentially use the system
    - 500 employees to use the system on a regular basis
    - 800 employees to use the system on at least a partial basis
- Of the employees that would use the system on a regular basis:
    - 350 employees would pick up the food themselves and eat in a centralized, social setting
    - 150 employees would prefer to enjoy their food at their desk and therefore want their food delivered

User class: *Cafeteria worker*

Characteristics of this user class:

- On the receiving end of orders placed by employees
- Requires a system that will allow them to view all the items ordered in a detailed, organized fashion
- Smaller audience than regular employees
    - 50 total cafeteria workers
- Out of the 50 total cafeteria workers:
    - there will be 5 different cashiers
    - there will be 10 food deliverers who will use the system to keep track of orders
- Will likely use system to perform CRUD (create, read, update, and delete) operations on the menu, as menu items change every day and they will be responsible for updating the menu

2.4 Operating Environment

OE-1: The cafeteria ordering system will be a web-based application that runs on various web browsers, including all currently active versions of:

- Google Chrome (> 69.0.3497)
- Mozilla Firefox (> 3.0)
- Internet Explorer (> 9.0)
- Safari (12.0)

OE-2: The application will be cross-platform, runnable on:

- Windows
- macOS
- Linux

OE-3: Depending on the success of the initial web-based version, a mobile version of the application may also be available in the future. In that case, the application will run on:

- Android (>4.1, Jelly Bean)
- iOS (> 7.0) software versions

OE-4: The application's server side will run on Node.js.

2.5 Design and Implementation Constraints

CO-1: There are several design and implementation constraints that developers need to consider when building the application. The front-end of the cafeteria ordering system will be written in:

- HTML5
- CSS3
- Javascript

and will therefore need to comply with the official HTML5 standards.

CO-2: The back-end of the system will be developed in:

- Node.js

and so it will need to adhere to Node.js best practices.

CO-3: The system will also need to be able to incorporate:

- Redis

as the data storage for all data collected by the application.

These technologies have been chosen based on the overall skill set of the developers that will be assigned to work on this application.

CO-4: Languages supported by the cafeteria ordering system:

- English
- Spanish
- Chinese
- Japanese
- French
- Russian
- Hindi
- Arabic

Any additional languages will be available upon request.

2.6 User Documentation

In order to assist users of the cafeteria ordering system, user documentation will be provided in various forms.

- Online tutorial
    - Users will be provided this option upon opening the application for the first time
    - Will navigate user through different parts of the application and give brief explanations of every feature
    - Completely optional and may be skipped if a new user prefers to learn the application on his/her own accord
- Pop-up tip bubbles
    - Will appear when a user remains idle for a predetermined period of time
    - May be disabled per user option
    - Option for the user to call the support team in case of any difficulties with using the application or for help to locate features within the application
- Official documentation of the cafeteria ordering system
    - Available in the company library as a comprehensive source of information for those who wish to obtain an even deeper understanding of the application

2.7 Assumptions and Dependencies

AS-1: Employees are technologically capable enough to navigate through a simple web application

- The user experience of the application will be optimized with the help of user interface and experience developers/researchers who will determine the best ways to:
    - Implement user flows
    - Decide placement of user interface elements, such as buttons.

AS-2: An employee that uses the application does not have any disability that would hinder their ability to use the application for its most basic purpose

- We will account for color-blindness and select colors for the application appropriately.

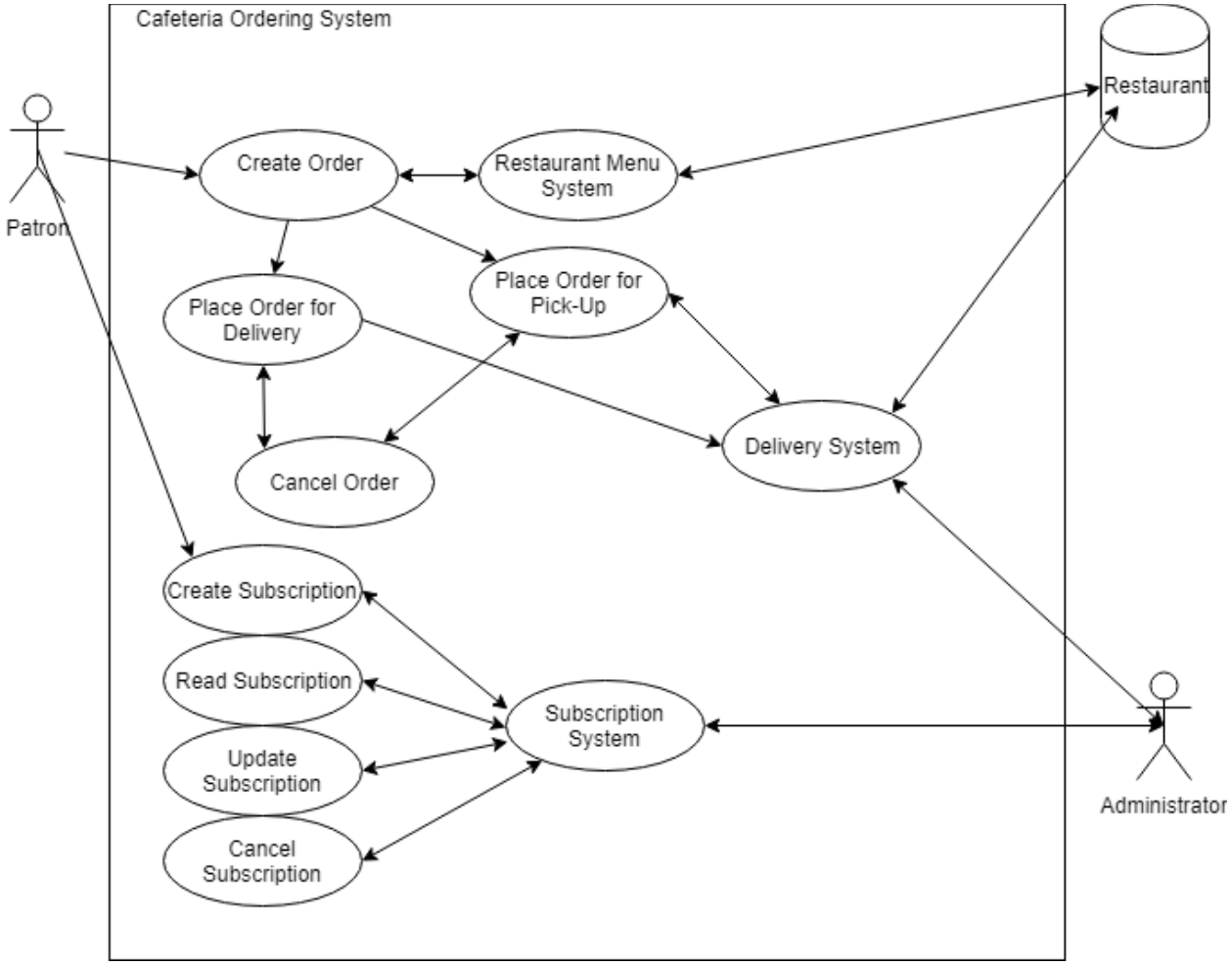
AS-3: Food ingredients and other materials are actually available when food items are ordered

DE-1: The application depends on another system that keeps track of the available cafeteria supplies

DE-2: External payment system that will handle all the transactions such as credit card payments

- This only applies to users who choose not to enroll in an automatic payroll deduction plan

3. Use-Case Diagram with Use-Case Descriptions



UC - 01 Placing an order for delivery

Use Case ID:	UC-01		
Use Case Name:	Placing an order for delivery		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons System		
Description:	As a Patron, they would like to order a meal for delivery.		
Trigger:	A Patron desires an item from the Cafeteria menu.		
Preconditions:	1. User has an account 2. User has verified billing (Activated ATM card or credits) 3. User has access to corporate intranet 4. Item suitable for delivery		
Postconditions:	Minimal Guarantees 1. User receives successful/failure confirmation Success Guarantees 1. User receives receipt 2. System begins charging User 3. System activates User membership		
Normal Flow:	Flow 1. User selects Menu item(s) 2. System prompts User for order confirmation 3. User confirms order confirmation 4. User enters billing / shipping information if not present 5. User selects delivery 6. System confirms if User financial information is legitimate 7. System confirms if User's delivery is valid 8. User receives successful order confirmation 9. User proceeds to pick-up / delivery phase		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In the billing step, the User may have a subscription service activated 1. User will bypass billing 3a. In the order confirmation step, User doesn't confirm their order 1. User will not confirm their menu items 2. User will go back to the menu screen and update their order 8a. In the delivery confirmation step, 1. User will not confirm their delivery choice 2. User will return to the order selection		
Exceptions:	4a. In step 4 of the normal flow, if the user's billing / shipping information isn't saved into the database 1. User will be prompted to enter billing / shipping information		

	2. User will be prompted to save information 3. User will be informed about subscriptions 6a. In step 6 of the normal flow, if the customer isn't approved 1. System will return User to order screen 2. System will inform User about billing failure 3. User returns to Order Confirmation screen 7a. In step 7 of the normal flow, if User's pick-up information isn't valid 1. System will return User to order screen 2. System will inform User about pick-up confirmation failure 3. User returns to Order Confirmation screen
Includes:	1. Ordering for a group 2. System reading catalog items from database
Frequency of Use:	400 - 800 times a day
Special Requirements:	1. Online database infrastructure 2. Billing Verification Service 3. Adequate Preparation Time 4. Delivery Service 5. Online intranet
Assumptions:	1. User understands English 2. User can navigate a web browser 3. User is ordering primarily during Lunch or Dinner 4. User can physically pick-up items 5. User can confirm their order with a receipt 6. Cafeteria Staff is ready and equipped
Notes and Issues:	Placing an order 1. Might have to introduce different types of User classes 2. MIght have to consider vegetarian/vegan options

UC - 02 Placing an order for pick-up

Use Case ID:	UC-02		
Use Case Name:	Placing an order for pick-up		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons System		
Description:	As a Patron, they would like to order a meal for pick-up		
Trigger:	A Patron desires an item from the Cafeteria menu, but Item can only be picked up.		
Preconditions:	1. User has an account 2. User has verified billing (Activated ATM card or credits) 3. User has access to corporate intranet 4. Item is not suitable for delivery		
Postconditions:	Minimal Guarantees 1. User receives successful/failure confirmation Success Guarantees 1. User receives order confirmation receipt		
Normal Flow:	Flow 1. User selects Menu item(s) 2. System prompts User for order confirmation 3. User confirms order confirmation 4. User enters billing / shipping information if not present 5. User can only select Pick-Up option 6. System confirms if User financial information is legitimate 7. System confirms if User's pick-up is valid 8. User receives successful order confirmation 9. User proceeds to pick-up phase		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In the order confirmation step, User doesn't confirm their order 1. User will not confirm their menu items 2. User will go back to the menu screen and update their order 8a. In the pick-up confirmation step, 1. User will not confirm their pick-up choice 2. User will return to the order selection		
Exceptions:	4a. In step 4 of the normal flow, if the user's billing / shipping information isn't saved into the database 1. User will be prompted to enter billing / shipping information 2. User will be prompted to save information 3. User will be informed about subscriptions 6a. In step 6 of the normal flow, if the customer isn't approved		

	1. System will return User to order screen 2. System will inform User about billing failure 3. User returns to Order Confirmation screen 7a. In step 7 of the normal flow, if User's delivery information isn't valid 1. System will return User to order screen 2. System will inform User about delivery confirmation failure 3. User returns to Order Confirmation screen
Includes:	1. Ordering for a group 2. System reading catalog items from database
Frequency of Use:	30 - 50 times a day 300 times a week 1200 times a month figures are per small to medium sized company
Special Requirements:	Online database infrastructure Billing Verification Service Adequate Preparation Time Delivery Service Online intranet
Assumptions:	User understands English User can navigate a web browser User is ordering primarily during Lunch or Dinner Online Ordering System is for one company Cafeteria Staff is ready and equipped
Notes and Issues:	Might have to introduce different types of User classes MIght have to consider vegetarian/vegan options

UC - 03 System retrieves items from local restaurants

Use Case ID:	UC-03		
Use Case Name:	System retrieves items from local restaurants		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patron System Local Restaurant		
Description:	System retrieves items from local restaurants,		
Trigger:	User views Cafeteria Menu		
Preconditions:	1. System has some sort of read access to local restaurants menu 2. System is equipped with		
Postconditions:	Minimal Guarantees 1. Patron can see available local restaurant items Success Guarantees 1. Item is indicated available		
Normal Flow:	Flow 1. Patron requests Cafeteria Menu/Catalog 2. System collects valid items 3. Patron can view items		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. During this step, the Patron can filter by internal Cafeteria Items 4a. During this step, the Patron can filter by external Local Restaurant Items.		
Exceptions:	3a. In step 3 of the normal flow, Patron can choose to filter catalog by Cafeteria or Local Restaurants.		
Includes:	n/a		
Frequency of Use:	500 - 1500 times a day		
Special Requirements:	Online database infrastructure Local Restaurant Verification Online internet		
Assumptions:	System has access to compatible Restaurant APIs System has contact information on incompatible Restaurants System has functioning networks		

UC - 04 Cancelling an order

Use Case ID:	UC-04		
Use Case Name:	Cancelling an order		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons Cafeteria Staff System		
Description:	User can cancel an on order within 5 minutes		
Trigger:	User orders item, then decides to cancel an order		
Preconditions:	1. User recently created an order 2. User has access to corporate intranet		
Postconditions:	Minimal Guarantees 1. User receives successful/failure confirmation Success Guarantees 1. User receives receipt confirmation 2. User's billing is updated to reflect cancellation 3. System catalogs cancellation		
Normal Flow:	Flow 1. User confirms order 2. User cancels order within 5 minutes		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. During this step, the user's cancellation window may expire. 1. User is on cancellation confirmation phase or prior 2. User's order cancelling window closes before confirming cancellation		
Exceptions:	2a. During this step, User attempts to cancel order after their cancellation window.		
Includes:	Cancelling for a whole group		
Frequency of Use:	30 - 50 times a day		
Special Requirements:	Infrastructure for cancellations and refunds		
Assumptions:	7. User understands English 8. User can navigate a web browser 9. User has ordered		

UC - 05 Patron creates a subscription

Use Case ID:	UC-05		
Use Case Name:	Patron creates a subscription		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patron		
Description:	As a Patron, they would like to establish a subscription plan with the online ordering system.		
Trigger:	A Patron chooses to enlist into a subscription plan.		
Preconditions:	1. User has an account 2. User has verified billing (Activated ATM card or credits) 3. User has access to corporate intranet		
Postconditions:	Minimal Guarantees 1. User receives successful/failure confirmation Success Guarantees 1. User receives receipt 2. User's membership is activated		
Normal Flow:	Flow 1. User confirms Subscription selection 2. User enters billing / shipping information		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In step 2, User may have information entered from a previous Delivery/Pick-Up Order 1. User will have their information filled out automatically		
Exceptions:	In step 2a, the User may have invalid billing / shipping information. 1. User is returned to the billing / shipping input form		
Includes:	Signing-up for a group subscription		
Frequency of Use:	300 times every year 75 times a month		
Special Requirements:	Online database infrastructure Billing Verification Service Subscription Service		
Assumptions:	User understands English User can navigate a web browser User has no active membership Subscription is monthly		
Notes and Issues:	There could be a variety of subscription plans		

UC - 06 Patron can read their subscription status

Use Case ID:	UC-06		
Use Case Name:	Patron can read their subscription		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons		
Description:	Patron wants to read their current subscription status		
Trigger:	A Patron accesses Catalog A Patron accesses their User Profile		
Preconditions:	1. User has an account 2. User has access to corporate intranet		
Postconditions:	Minimal Guarantees User reads an “active” / “in-active” status Success Guarantees 1. User has an extra option in User Profile to subscribe		
Normal Flow:	[Provide a detailed description of the user actions and system responses that will take place during execution of the use case under **normal, expected** conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. E.g., Flow 1. User enters		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In the order confirmation step, User doesn’t confirm their order 3. User will not confirm their menu items 4. User will go back to the menu screen and update their order 8a. In the delivery confirmation step, 3. User will not confirm their delivery choice 4. User will return to the order selection to confirm delivery option		
Exceptions:	[Describe any anticipated **error conditions** that could occur during execution of the use case, and define how the system is to respond to those conditions. e.g. Exceptions to the Withdraw Case transaction 4a. In step 4 of the normal flow, if the user’s billing / shipping information isn’t saved into the database 4. User will be prompted to enter billing / shipping information 5. User will be prompted to save information		

	6a. In step 6 of the normal flow, if the customer isn't approved 4. System will return User to order screen 5. System will inform User about billing failure 6. User returns to Order Confirmation screen 7a. In step 7 of the normal flow, if User's delivery information isn't valid 4. System will return User to order screen 5. System will inform User about delivery confirmation failure 6. User returns to Order Confirmation screen
Includes:	3. Ordering for a group
Frequency of Use:	30 - 50 times a day 300 times a week 1200 times a month figures are per small to medium sized company
Special Requirements:	6. Online database infrastructure 7. Billing Verification Service 8. Pick-Up Location 9. Delivery Service 10. Online intranet
Assumptions:	10. User understands English 11. User can navigate a web browser 12. User is ordering primarily during Lunch or Dinner 13. Online Ordering System is for one company
Notes and Issues:	Placing an order 3. Might have to introduce different types of User classes 4. MIght have to consider vegetarian/vegan options

UC - 07 Patron can update their subscription

Use Case ID:	UC-07		
Use Case Name:	Patron can update their subscription billing information		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons		
Description:	As a Patron, they would like to update their billing information for the subscription service.		
Trigger:	A Patron clicks Update Billing in User Profile		
Preconditions:	1. User has an account 2. User has verified billing (Activated ATM card or credits) 3. User has access to corporate intranet 4. User has an active subscription		
Postconditions:	Minimal Guarantees User receives successful/failure confirmation Success Guarantees 1. User receives receipt 2. System updates billing information		
Normal Flow:	[Provide a detailed description of the user actions and system responses that will take place during execution of the use case under **normal, expected** conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. E.g., Flow 1. User selects Update Billing from User Profile 2. User enters billing information 3. User confirms and submits information		
Alternative Flows: [Alternative Flow 1 – Not in Network]	3a. In the update confirmation step, User chooses to not confirm their information 1. User will not confirm their menu items 2. User will go back to the User Profile screen		
Exceptions:	2a. In step 2 of the normal flow, if the user's billing / shipping information isn't valid 1. User will be alerted of failed billing verification 2. User will be returned to Update Billing screen		
Includes:	Updating subscription for a group		
Frequency of Use:	30 - 50 times a month		
Special Requirements:	Online database infrastructure Billing Verification Service Corporate Intranet		

Assumptions:	User understands English User can navigate a web browser User has active membership for subscription
Notes and Issues:	Administrators and cafeteria staff might need access to this as well

UC - 08 Patron can cancel their subscription

Use Case ID:	UC-07		
Use Case Name:	Patron can cancel their subscription billing information		
Created By:	Jason	**Last Updated By:**	Jason
Date Created:	10/7/18	**Last Revision Date:**	10/7/18
Actors:	Patrons		
Description:	As a Patron, they would like to cancel their subscription.		
Trigger:	A Patron clicks Update Billing in User Profile and selects Cancel Subscription		
Preconditions:	1. User has an account 2. User has verified billing (Activated ATM card or credits) 3. User has access to corporate intranet 4. User has an active subscription		
Postconditions:	Minimal Guarantees User receives successful/failure confirmation Success Guarantees 1. User receives receipt 2. System updates User membership		
Normal Flow:	[Provide a detailed description of the user actions and system responses that will take place during execution of the use case under **normal, expected** conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. E.g., Flow 1. User selects Cancel Subscription from User Profile 2. User confirms cancellation		
Alternative Flows: [Alternative Flow 1 – Not in Network]	2a. In the update confirmation step, User chooses to not confirm their information 1. User will not confirm their cancellation 2. User will go back to the User Profile screen		
Exceptions:	n/a		
Includes:	Updating subscription for a group		
Frequency of Use:	30 - 50 times a month figures are per small to medium sized company		
Special Requirements:	Online database infrastructure Billing Verification Service Corporate Intranet		
Assumptions:	User understands English User can navigate a web browser User has credit-card / billing information		

Notes and Issues:	Administrators and cafeteria staff might need access to this as well

4. External Interface Requirements

4.1 User Interfaces

UI -1. Payment and address fields shall be validated

UI -2. Error message indicates invalidity

UI -3. The system shall provide tooltips when objects are hovered

UI -4. Details of user interface design will be documented on *Process Impact User Interface Design* document

UI -5. The webpages shall permit keyboard navigation

- UI -5.1. TAB navigates down through web pages from left to right
- UI -5.2. SHIFT+TAB navigates to previously “tabbed” element
- UI -5.3. ENTER selects an item
- UI -5.4. UP/DOWN navigates through picklist options

4.2 Hardware Interfaces

HI -1. The system shall allow the printing the following from a printer

- HI -1.1. Receipts
- HI -1.2. Recipes and ingredients list from custom meals

HI -2. The system shall operate in browsers on Windows, Linux, MAC and Android platforms

4.3 Software Interfaces

SI -1. Inventory System

SI -1.1. The system shall transmit data to be printed – receipt, recipes & ingredient list for custom meals

SI -2. Payment System

SI -2.1. The system shall accept payments from customers

SI -2.2. The system shall have the ability to provide refunds

SI -2.3. The printer shall notify the system when it is unable to print

SI -3. Vendor Communication System

SI -3.1. Participating restaurants shall transmit available food choices to the system

SI -3.2. The system shall transmit ordered items and quantities to restaurants

SI -3.3. Restaurants shall notify the system when food items are unavailable

SI -3.4. The system shall transmit payments to participating restaurants

4.4 Communications Interfaces

CI -1. The Cafeteria Ordering System Communication interface will give an option to send message and updates by text messages or email.

CI -1.1. The text message and/or email communication will confirm the order with an online receipt detailing the order and total cost.

CI -1.2. The text message and/or email communication will include delivery information and estimated time.

CI -2. The Cafeteria Ordering System Communication interface will send an additional text message and/or email if there are any problems or delays in the order.

CI -3. The Cafeteria Ordering System Communication interface will notify that text messages and data rates may apply.

CI -4. The Cafeteria Ordering System Communication interface will encrypt messages.

CI -4.1. Customers' payment card is encrypted for confidentiality.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

PER-1. Items including informational text and images displayed on all pages should be readable and within a size limit to ensure quick loading performance.

PER-2. All pages should load within a reasonable time - about 1-3 seconds - to provide a smooth experience and improve ease of use for the customer.

PER-3. Each selected order item from the user will be instantly added to the cart with an instant pop-up notification.

PER-4. Confirmation notification should be displayed within 1-3 seconds after it's been submitted.

PER-5. System should have a large data capacity, so a large number of employees can use it during lunch hours.

5.2 Safety Requirements

SFR -1. Display warning of any possible food allergy ingredients when users are selecting food items.

5.3 Security Requirements

SCR -1. Users must log in with an employee identification number to use the system

SCR -2. Log in attempts are limited.

SCR -3. Users are limited to the use of the system by only viewing, selecting, ordering.

SCR -4. Only managers are allowed to have access to change menus.

SCR -5. After a confirmation notification order, users may manually log out or the system will automatically log out within one minute.

SCR -6. If the session is open but not in use within 10-15 minutes, the system will automatically log out.

SCR -7. System server will have a malware security protection

5.4 Software Quality Attributes

5.4.1 - Availability

AVA -1. The COS should be available to access during business hours, between 9:00 am to 5:00 pm.

AVA -2. Maintenance or updates to the system or menus must be done before or after business hours.

5.4.2 - Reliability

REL -1. The service would be monitored to make sure it is running efficiently.

REL -2. Once the order is completed, a text message and/or email will be sent immediately to confirm and guarantee delivery.

5.4.3 - Robustness

ROB -1. If server closes or connection is faulty during user's session, data will be saved and recovered when user logs back in within 20 minutes.

5.4.5 - Flexibility

FLE -1. Administrator in charge or manager can modify menu settings

5.4.6 - Portability

POR -1. The COS can be used on different browsers on different devices.

5.4.7 - Usability

USE -1. Navigation will be shown at all times. (home, menu, cart, help)

6. Other Requirements

DAT -1. Database Requirements

DAT -1.1. Models

DAT -1.1.1. User, Patron, Staffer, Administrator, Local Restaurant, Menu, Item

7. Functional Requirements (FRs)

7.1.1 Signing Up Users

System should sign-up Users for ordering with the System. Should establish a User with a status such that the User could interact with the core ordering experience.

Reference(s): UC-01, UC-02, UC-04, UC-05, UC-08

7.1.2 Inputs

Required

User's name, preferred email, employee ID, company ID, password

Optional

personal information for the health profile system

Local restaurant preferences

7.1.3 Processing

System will verify username for compatibility.

System will verify preferred email for compatibility.

System will verify password for strength and compatibility.

System will verify employee/company relationship for system approval

7.1.4 Outputs

Outputs should be a User account with an "Approved" or "Disapproved" status.

Confirmation emails should be sent out to both User and their Companies.

7.1.5 Error Handling

If a User's employee ID doesn't exist, should remain on login page with notification.

If a User's company ID doesn't exist, should remain on login page with notification.

If a Company/User relationship doesn't exist, should remain on login page with notification.

If a User's email isn't verified, should remain on login page with notification.

If a User's name is more than the limit of 255 characters, should remain on login page with notification.

7.2.1 Registering Administrators

System should register Administrators for managing the System. Should establish a User with a status such that it could interact with the core system and manage users.

Reference(s): UC-05, UC-06, UC-07

7.2.2 Inputs

Required

User's name, preferred email, password

Optional

employee ID
company ID
personal information for the health profile system
Local restaurant preferences

7.2.3 Processing

System will verify username for compatibility.
System will verify preferred email for compatibility.
System will verify password for strength and compatibility.

7.2.4 Outputs

Outputs should be an Administrator account.
Confirmation email should be sent out to Administrator.

7.2.5 Error Handling

If a User's employee ID doesn't exist, should remain on login page with notification.
If a User's company ID doesn't exist, should remain on login page with notification.
If a Company/User relationship doesn't exist, should remain on login page with notification.
If a User's email isn't verified, should remain on login page with notification.
If a User's name is more than the limit of 255 characters, should remain on login page with notification.

7.3.1 Registering a Restaurant

The System should add Restaurant data for their catalog and ordering.
Reference(s): UC-01, UC-02, UC-03, UC-04

7.3.2 Inputs

Required

Restaurant's name, preferred email, password, street address

Optional

Menu Catalog
Contact Information
Billing Information

7.3.3 Processing

System will process street address for verification
System will process billing information for verification
System will scan Menu Catalog for compatibility
System will scan username for compatibility
System will scan password entry for strength and compatibility
System will scan contact information for compatibility

7.3.4 Outputs

Outputs should be an Restaurant account. Optional outputs include a Menu.

7.3.5 Error Handling

If street address fails verification, remain on registration page with notification.
If billing information fails verification, remain on registration page with notification.
If Menu Catalog fails compatibility, remain on registration page with notification.
If username fails compatibility, remain on registration page with notification.
If password entry fails strength and compatibility, remain on registration page with notification.
If contact information fails compatibility, remain on registration page with notification.

7.4.1 Placing an Order

When User places an order, the System should register an order.

Reference(s): UC-01, UC-02, UC-03, UC-04

7.4.2 Inputs

Required

Order, User order confirmation

7.4.3 Processing

System should verify Order by processing Menu-Catalog relationship.
System should verify Restaurant Menu.
System should verify Restaurant Pick-Up time.
System should update Restaurant account.
System should update Company account.
System should update User account.

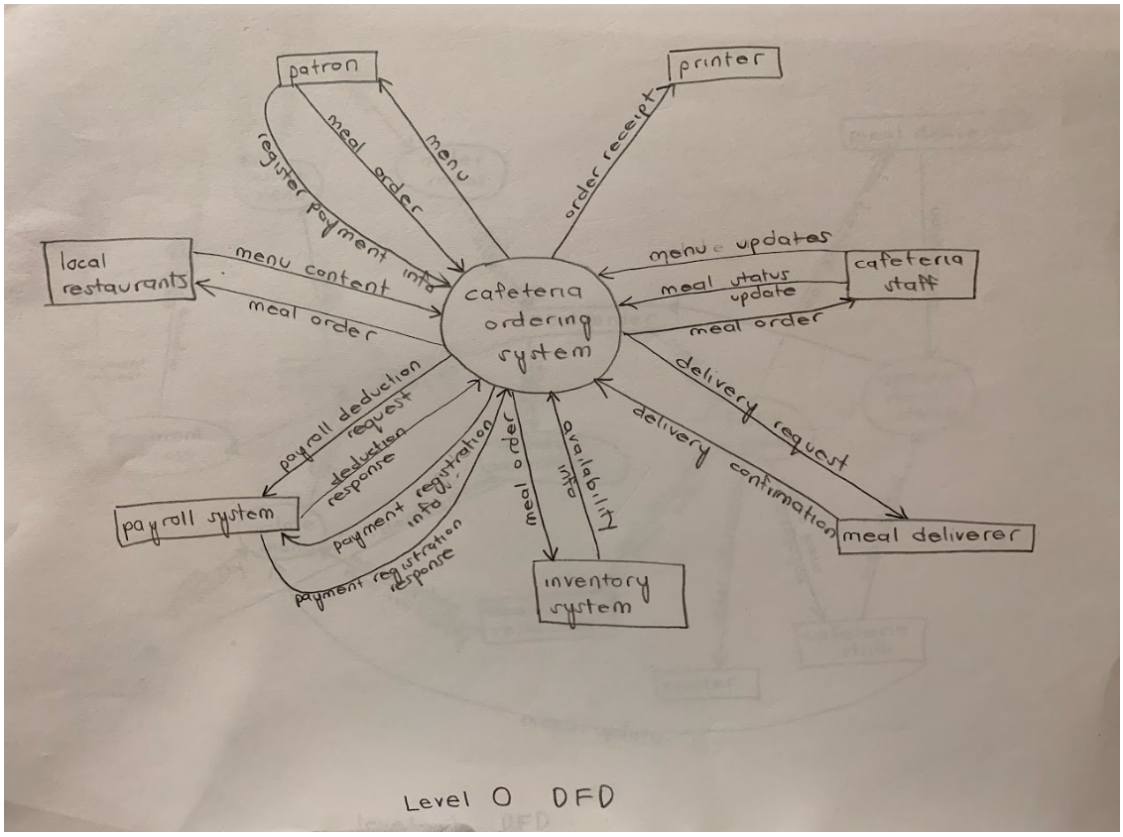
7.4.4 Outputs

User should receive an Order Confirmation Page.
Confirmation email should be sent out to User, Company.
Restaurant will receive email/notifications for ordering.

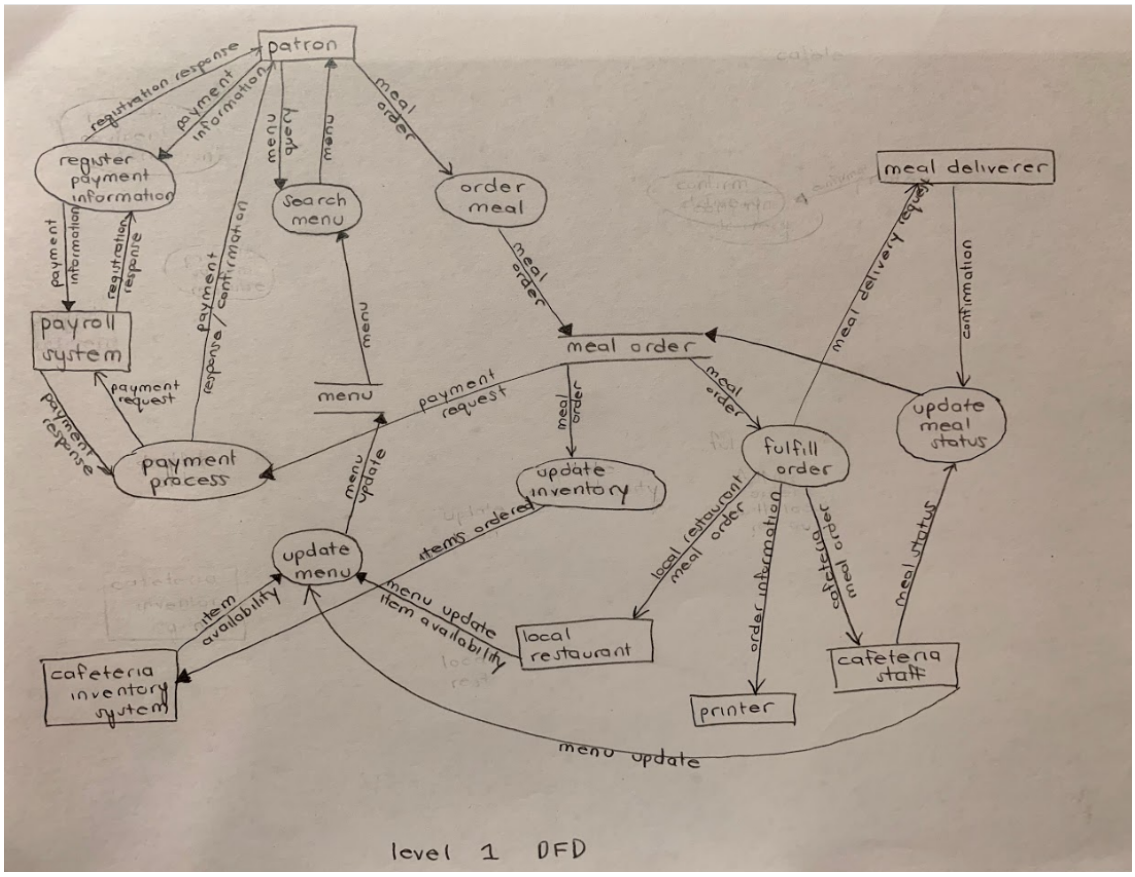
7.4.5 Error Handling

If verification fails processing Menu-Catalog relationship, remain on Ordering page.
If system fails verifying Restaurant Menu and Order, update Cart.
If Pick-Up time fails order window, notify User, Administrator, Company.
If Order takes longer than 15 minutes, cancel order and reset Cart.

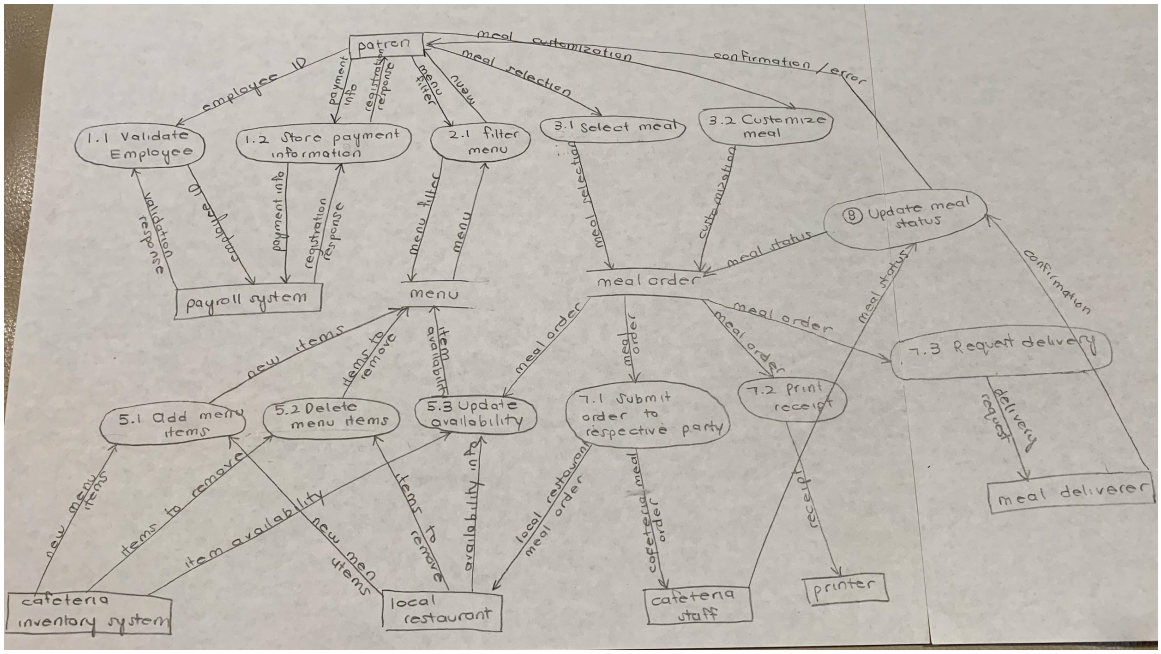
8. Data Flow Diagrams



8.1 Level 0 DFD



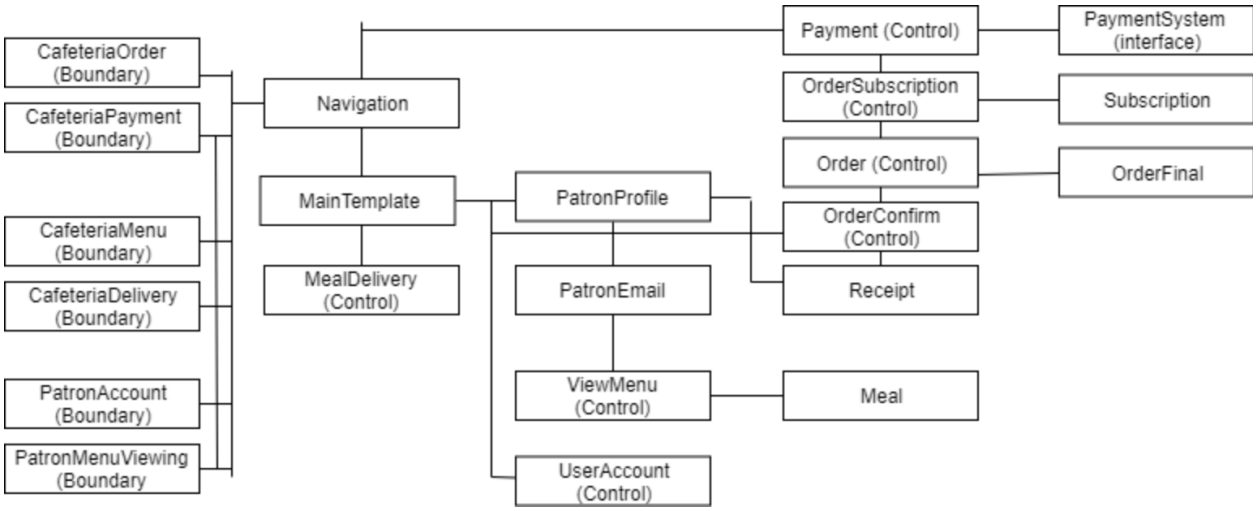
8.2 Level 1 DFD



8.3 Level 2 DFD

9. Initial & Simplified Class Diagrams

9.1 Initial Class Diagram



9.2 Simplified Class Diagram