# Canteen Food Ordering and Management System

## Software Requirements Specification

Software Engineering Project

**Version 1.0**



**Group Id:** SE-PID-21

**Group Members:** 2018-CS-13, 2018-CS-35, 2018-CS-54

**Supervisor Name :** Ms Taliah Tajammal

## Department of Computer Science
University of Engineering and Technology Lahore

**Table of Contents**

# 1  Introduction

In this SRS we will cover the topics related to our project of Food Ordering and Canteen management system. We will discuss topics related to our project including scope, implementation details, functionality, interfaces and technologies. The main audience of the project will be canteen customers and canteen management.

## 1.1  Scope

The main scope of our product covered in this SRS explains all the functionalities that are related to our project and our product is working on the basis of that. The main and very fundamental reason behind development of this product is to replace the manual food ordering system with a technology based system so that it works smoothly and more efficiently. From Canteen management to customers this product will serve and help all the users to do their work more easily. In this SRS we are not only going to define our product but also the major workflow and usage of technology that is making it easy to use and understand.

## 1.2  Glossary

| Term | Definition |
|------|------------|
| Patron | The person who will order things from the canteen. |
| Admin/ Administrator | System Administrator who is given specific permission for managing and Controlling the System. He will manage incoming orders, customer details and the payment information. |

## 1.3  Technologies to be used

| Technology | Description |
|------------|-------------|
| **Microsoft Visual Studio** | Microsoft Visual Studio is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. |
| **Pencil Tool** | Pencil tool is an easy to understand and use software that is commonly used to design wireframes for a project to show a basic picture of how the screens work. |
| **Microsoft Project Professional** | Microsoft Project is a project management software program, developed and sold by Microsoft, which is designed to assist a project manager in developing a plan, assigning resources to tasks, tracking progress, managing the budget, and analyzing workloads. |

# 2  General Description

## 2.1  User Characteristics

Basically, we have two users for this project:

- ● Admin
- ● Patron

These users will use our product and they will be familiar with the functionality of our product. They know how to move through the application and use different functionalities. They will be familiar with the interface of the system.

**Admin:**

Admin will actually be the canteen owner. Admin will be a user of the system who will login into his account and manage the orders of the customers. He will take care of incoming orders, customer instructions about food and all the billing will be checked by him.

**Patron:**

Patron will actually be the person coming to the canteen and ordering different food items. Using our product they will be able to search for any particular food, select food items according to their current taste requirements, give special instruction regarding food and confirm order after selecting all required items.

## 2.2  General Constraints

Initially our product can be used by only one place at a time as it is not a networked app so there will be only one record system of the application related to it. But on further improvements we can update it to be a networked app used by different places at a time.

## 2.3  Assumptions & Dependencies

Following are some basic assumptions for this system.

- ● As a patron there should be at least one food item must be available to order.
- ● Since it is a web based application, basic internet speed of loading a web page is a critical need because all other things are dependent on it.
- ● All the personal information entered by the patron including billing details should be   kept confidential. It means that there should be a security layer above that data so only the admin has the right to view patron information.
- ● As an admin when orders are confirmed by a patron their detail should be fastly transferred to admin so admin can manage order accordingly. In short, transfer of information must be fast between customer and admin.
- ● Menu should always be updated so currently available stock of food items can be ordered by patrons and there is no problem.

# 3  Specific Requirements

## 3.1  *External Interface Requirements:*

### 3.1.1  User Interfaces
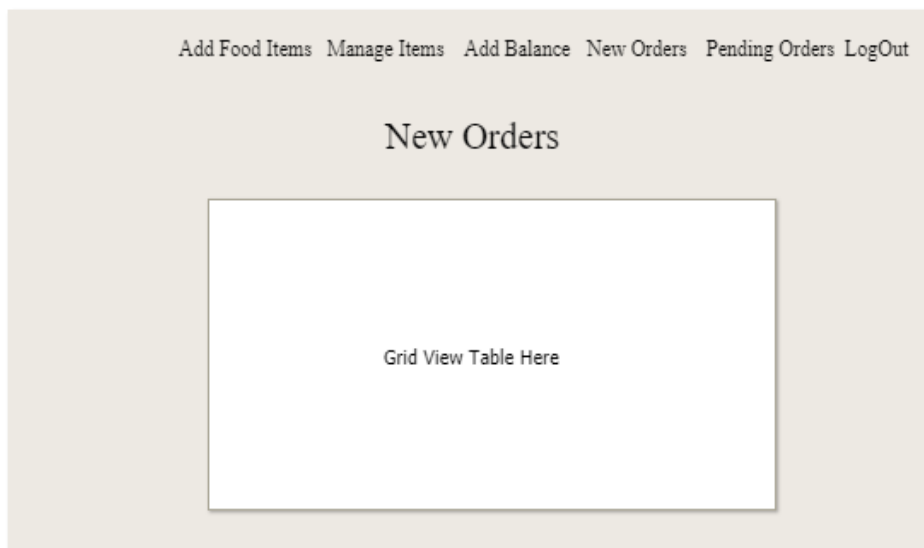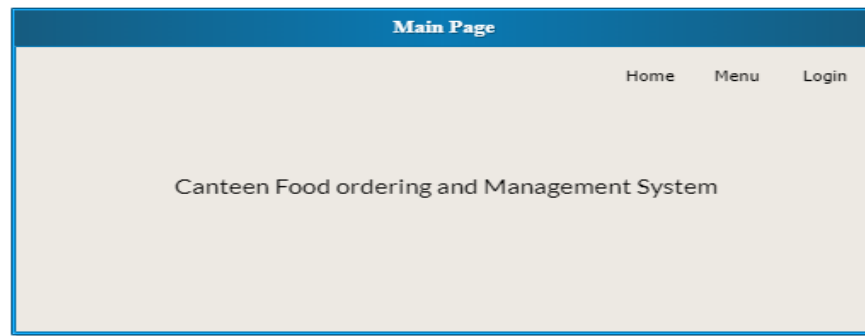
Add Food Items  Manage Items  Add Balance  New Orders  Pending Orders  LogOut

# Manage Items

Soup    Breakfast    Lunch    and more

Food Image Here          Food Description

Edit          Delete

Load More

Add Food Items  Manage Items  Add Balance  New Orders  Pending Orders  LogOut

# Add Balance

User Id

Select ⌄

User Name

Text box

User Wallet Amount

Text box

Amount to Update

Text box

Update

Add Food Items   Manage Items   Add Balance   New Orders   Pending Orders  LogOut

# Pending Orders

Grid View Here

Home     Menu     Order History     LogOut

Food Category
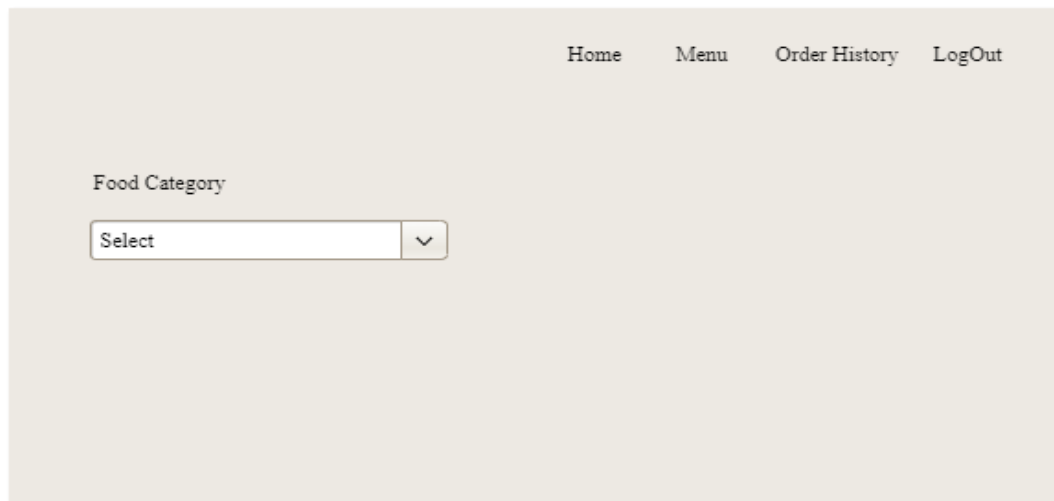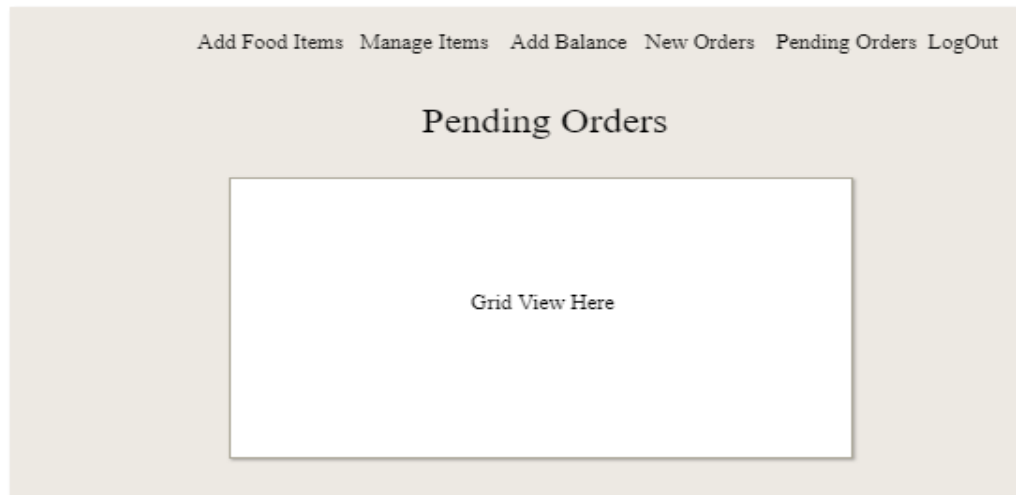
Select

**Select Item**

|  |  |
|---|---|
| Food1 | Food 2 |

|  |  |
|---|---|
| Food 3 | Food 4 |

**Order History**

Your Cart is empty start shopping now...!

## *3.2  Functional Requirements*

A Functional requirements are the functionality of a system that defines what is the functional purpose of that product .It depends upon the type of software, objectives related to the system, expectation of the user.

### 3.2.1  Requirement 1

*Create canteen* patron *account*

### 3.2.2  Requirement 2
*Login to account while ordering*

### 3.2.3  Requirement 3
*Manage* patron *account*

### 3.2.4  Requirement 4
*Navigate the canteen menu*

### 3.2.5  Requirement 5
*Select desire food with personal choice customization*

### 3.2.6  Requirement 6
*Add item to the cart*

### 3.2.7  Requirement 7
*Manage cart*

### 3.2.8  Requirement 8
*Review bill before order confirmation*

### 3.2.9  Requirement 9
*Place order*

### 3.2.10  Requirement 10
*Receive order ID for further correspondence*

## 3.3  Use Cases:

| Use Case Title | Creation of patron account |
|---|---|
| Use Case Id | 1 |
| Requirement Id | 1 |
| **Description:** This use case is about creating a new patron account to an existing system so that user can order through his account. | |
| **Pre-Conditions:**<br>1. All must-required information about the new patron should be available.<br>2. Database should be available in online mode to save new patron's information.<br>3. Load the sign up page | |
| **Normal Flow** | |
| 1. Patron opens the sign up page for his/her account creation. | |
| 2. System asks for necessary information. | |
| 3. Patron provides all the required information and opts to complete the creation operation. | |

| | 4. System checks that all enter information is valid. |
| --- | --- |
| | 5. System after confirmation adds the new account. |
| | 6. System sends the confirmation message to screen. |

**Alternative Flow**

4a. There is a problem in the data provided; some data needs to be corrected.
- Patron checks the provided information and corrects the error.

Patron continues from the step 4.

**Post Conditions:** A new Patron account is successfully created.

**Open issues:** Patron is trying to create account with information that has already been used by someone for account creation.

**Authority:** Patron

| Use Case Title | Login to Account |
| --- | --- |
| Use Case Id | 2 |
| Requirement Id | 2 |

**Description:** This use case is about login to an existing account so that user can order through his account.

**Pre-Conditions:**
1. Must have create an account.
2. Correct Credential provided for login.
2. Accounts information must be available in database.
3. Load the login page.

**Normal Flow**

| 1. Patron opens the sign in page for his/her account login. |
| --- |
| 2. System asks for necessary information. |
| 3. Patron provides all the required information and opts to complete the login operation. |
| 4. System checks that all enter information is valid. |
| 5. System after confirmation login to account. |
| 6. System then directs the patron to his/her account. |

**Alternative Flow**

4a. There is a problem in the data provided; some data needs to be corrected.
- Patron checks the provided information if the information doesn't match it will alert patron about it.

Patron continues from the step 5.

**Post Conditions:** Patron Successfully login to his/her account.

**Open issues:** Patron has forgotten his/her information.

**Authority:** Patron

| Use Case Title | Manage Account |
| --- | --- |
| Use Case Id | 3 |
| Requirement Id | 3 |

| | |
|---|---|
| **Description:** This use case is about Patron managing his/her account. | |

**Pre-Conditions:**
1. Account must be created.
2. Patron must have login to his/her account to manage it.

**Normal Flow**
1. Patron login to his/her account.

2. Patron opens Management Settings of his/her account.
3. Patron manages his /her account.
4. Patron saves new information.

**Alternative Flow**

3a. There is a problem in the managing information; user may have provoked violation.
   − Patron provides false information or break violation rules so that information cannot be saved.
Patron continues from the step 3.

**Post Conditions:** A Patron manages his/her account successfully.

**Open issues:** Patron is trying to provide false information and break the violation rules. Account does not exists.

**Authority:** Patron

| Use Case Title | Navigation through Menu |
|---|---|
| Use Case Id | 4 |
| Requirement Id | 4 |

**Description:** This use case is about patron navigation through menu so that patron can order through his account.

**Pre-Conditions:**
1. Account must be created.
2. Patron must have login to his/her account to navigate through menu list.
3. Some food must be available in menu.

**Normal Flow**
1. Patron login to his/her account.

2. Patron starts navigating through menu items to select some food.

**Post Conditions:** A Patron successfully select item from menu.

**Authority:** Patron

| Use Case Title | Selection of Desired Item |
|---|---|
| Use Case Id | 5 |
| Requirement Id | 5 |

**Description:** This use case is about Patron Selected his/her desired item from menu.

**Pre-Conditions:**
1. Must have created an account.
2. Login to his/her Account.

| | 3. Navigation through menu. |
|---|---|

**Normal Flow**

1. Patron navigate through items to choose his/her desired item.
2. Patron Successfully selected an item of his/her choice.

**Post Conditions:** Patron Successfully Selected food.

**Open issues:** Food item is not available which patron is looking for.

**Authority:** Patron

| Use Case Title | Add item to Cart |
|---|---|
| Use Case Id | 6 |
| Requirement Id | 6 |

**Description:** This use case is about Adding selected food item to cart so that user can order.

**Pre-Conditions:**

1. Must have created an account.
2. Login to his/her Account.
3. Selected his/her desired item from menu.

**Normal Flow**

1. Patron Login to his/her account.

2. Patron navigate through items to choose his/her desired item.
3. Patron Successfully selected an item of his/her choice.
4. Add item to cart.

**Alternative Flow**

3a. Add the wrong item to the cart
  − Patron discard the item from menu
Patron continues from the step 2.

3b. Item is not available at right moment.

**Open issues:** There is a problem with database the item is not being added in cart.

**Authority:** Patron

| Use Case Title | Patron's Cart Management |
|---|---|
| Use Case Id | 7 |
| Requirement Id | 7 |

**Description:** This use case is about cart management by patron so that patron can manage order before placement.

**Pre-Conditions:**

1. Patron must be login into his account.
2. Cart should not be empty.

**Normal Flow**

1. Patron is in his/her account.

2. After selection of all food he clicks move to cart.

| | |
|---|---|
| 3. | Patron provides all the required information related to food present in his cart like quantity of a particular food and any special note related to any food. |
| 4. | Patron then checks if any extra food or wrong quantity of any food has been added. |
| 5. | Patron remove extra food from his cart. |
| 6. | Patron moves to checkout. |

**Alternative Flow**

2a. The Food patron want to order is not available at the moment.
- A message show to patron related to unavailability of food so he can order something other.

Patron continues from the step 3.

**Post Conditions:** Patron has moved to checkout after order confirmation.

**Open issues:** Patron is trying to enter quantity of some food at that extent which is not available.

**Authority:** Patron

| Use Case Title | Review of Bill |
|---|---|
| **Use Case Id** | 8 |
| **Requirement Id** | 8 |
| **Description:** This use case is about review bill of order so that user can confirm order. | |

**Pre-Conditions:**
1. Patron's cart is not empty.
2. Patron has confirm order.
3. System should be available in online mode to send order to canteen management.

**Normal Flow**

| | |
|---|---|
| 1. | Patron has moved to checkout section after order confirmation. |
| 2. | Patron cross check the items their quantity and prices added in cart. |
| 3. | Patron review total bill. |
| 4. | Patron select payment method. |
| 5. | System after confirmation of bill moves toward order confirmation. |
| 6. | System sends the confirmation message to screen. |

**Alternative Flow**

2a. There is a problem in the food ordered: mistakenly some food has added in order.
- Patron go back to cart section to remove that item.

Patron continues from the step 4.

**Post Conditions:** Patron has review the bill and moves toward order confirmation.

**Open issues:** Bill of a particular item or total bill is not generating correctly. There is some error in bill calculations.

**Authority:** Patron

| Use Case Title | Order Confirmation |
|---|---|
| Use Case Id | 9 |
| Requirement Id | 9 |

**Description:** This use case is about order confirmation so that order can be proceed by canteen management.

**Pre-Conditions:**
1. Patron has checked out the bill and has confirm it.
2. Database should be available in online mode to save patron's order.

**Normal Flow**
1. Patron is on order confirmation page.
2. System asks for any special note related to order.
3. Patron provides all the instruction in special note section.
4. Patron confirm order.
5. System send order detail to canteen system and save in database
6. System sends the confirmation message to screen.

**Alternative Flow**

All the steps will execute in normal flow.

**Post Conditions:** Patron's order has been placed.

**Open issues:** There is an error with database so that order is not placing properly like details are not saving in database or canteen management is unable to locate order in their system.

**Authority:** Patron

| Use Case Title | Generation of Order Confirmation ID |
|---|---|
| Use Case Id | 10 |
| Requirement Id | 10 |

**Description:** This use case is about getting order confirmation ID for future correspondence.

**Pre-Conditions:**
1. An order has been successfully placed by Patron.

**Normal Flow**
1. Patron has confirm the order.
2. System sends the confirmation message to screen.
3. System provides an ID to Patron for future correspondence.

**Alternative Flow**

All the steps will execute in normal flow.

**Post Conditions:** Patron has receive order ID.

**Open issues:** There is an error with system so that order id is not generating properly.

**Authority:** Patron

### 3.4 Non Functional Requirements:

- **Privacy:**

  The personal information of users enter to system must be confidential.

- **Accessibility on web:**

  The web application must be available to any one any time having internet connection.

- **Accuracy:**

  The data entered by user must be sure to be correct and consistent.

- **Account Management:**

  Since user has to login to the system, so when user enter system must provide them all functionalities.

- **User friendly Interface:**

  System's interface must be easy to understand and use and does not waste user's time.

- **Data safety:**

  System must be able to handle large data so that it does not crash when it has to handle large data.

# 4 Methodology

## 4.1 Adopted Methodology

Keeping in view our product we have decided that we will use agile methodology and its framework SCRUM will be used for the development of our product. This is a framework that supports changing requirements during product development and allows you to adjust those requirements into the product. Scrum is an agile methodology or framework used primarily for software development projects with the goal of delivering new software capability every 2-4 weeks. It is one of the approaches that influenced the Agile Manifesto. As compared to waterfall, agile methodology works in a different way. Here, a version of the project is released after every iteration. This is called SPRINTS. It offers flexibility of covering up missing requirements in the next phase (whereas possible). Agile methods break tasks into smaller iterations, or parts. It does not directly involve long term planning. The project scope and requirements are laid down at the beginning of the development process. Plans regarding the number of iterations, the duration and the scope of each iteration is clearly defined in advance. The division of the entire project into smaller parts helps to minimize the project risk and to reduce the overall project delivery time requirements. Each iteration involves a team. Team works through a full software development life cycle including planning, requirements analysis, designing, coding, and testing before a working product is demonstrated to the client.

## 4.2 Diagram of Methodology
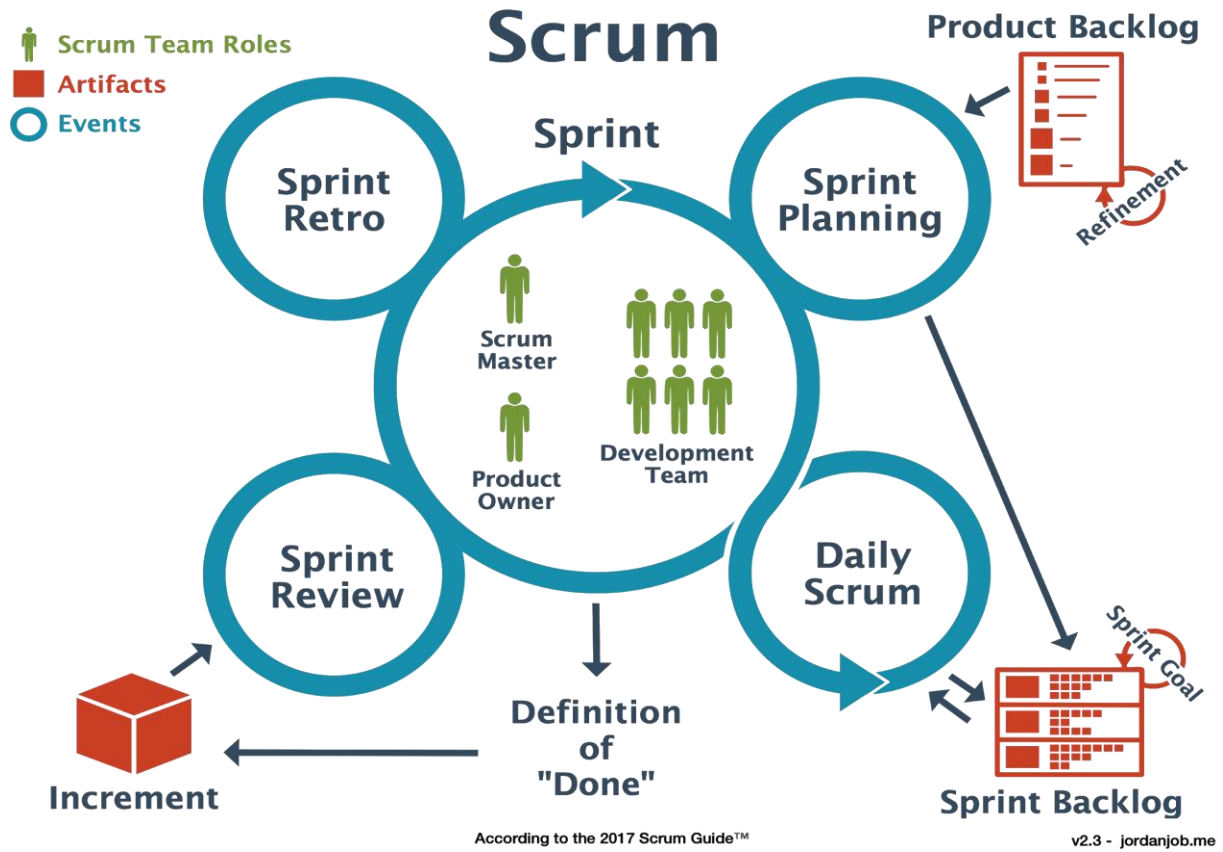
## 4.3 Reason of Choosing Methodology

The basic motivation behind selecting this methodology is that division of big project can be made into smaller units that can be easy to manage and implement. If there is a change in requirement at any stage of development we can adjust that change according to user requirement. Moreover, during the whole development of the product an interaction with the customer continues that makes it easy negotiating with the user on any topic.

# 5  Project Work Plan

| Oct 26 | Nov 9 | Nov 23 | Dec 7 | Dec 21 | Jan 4 | Jan 18 | Feb 1 | Feb 15 |
|--------|-------|--------|-------|--------|-------|--------|-------|--------|

**Food Ordering and Canteen Management System**
October 28, 2020 - February 2, 2021

**Project Proposal**
October 28, 2020 - November 3, 2020

**Software requirement specification**
November 4, 2020 - November 10, 2020

**Requirement Analysis**
October 28, 2020 - November 10, 2020

**Design Document**
November 11, 2020 - November 24, 2020

**Admin Features Implementation**
November 25, 2020 - December 20, 2020

**Patron Features Implementation**
December 21, 2020 - January 10, 2021

**Test Case Development**
January 11, 2021 - January 17, 2021

**Varification & Validation**
January 18, 2021 - January 26, 2021

**Deployment**
January 27, 2021 - February 2, 2021