Software Requirements Specification for Online Bakery System

Version 1.0

Prepared by: Varun Kumar Meda, Mihir Santosh Sanjay, Paladugula Lakshmi Snigdha, Pragathya .B

Organisation: PES University

Date Created: 30/08/2022

Index:

7. Other Requirements

Appendix A: Glossary
Appendix B: Field Layouts
Appendix C: Requirement Traceability matrix

Online Bakery System:

1.Introduction:

1.1 Purpose:

The Online Bakery system is a web application that allows various customers to order food online and get it delivered. This software is platform-independent and flexibly runs on all operating systems and mobile devices. In addition to this, the Bakery ordering system also allows the users to order meals from other local Bakeries and the same will be delivered based on the user's delivery request.

Bakery owners can provide their bakery information using the web portal. This information will act as the basis for the search results displayed to the user. A receipt is generated which acts as a token for identifying the customer who placed the order through the Online Bakery system.

1.2 Intended Audience and Reading Suggestions:

The purpose of the document is to give a detailed description of the requirements for the "Online Bakery System" software. It will illustrate the purpose, scope and complete description for the development of software. It will also explain external interface requirements and system requirements as well as non-functional requirements. The document is primarily intended to be proposed to a customer for its approval and also for further processing such as additional features to be developed in later releases.

1.3 Product Scope:

The Online Bakery System is a management system developed with the intention of automating the day to day tasks in a bakery like orders, inventory management, bill generation and feedback. This release of the software would deal with these tasks only whereas more areas might be automated in the future versions of this software. The main purpose is to improve the performance of the bakery by eradicating the daily paperwork. With this system the tasks would be performed in less amount of time and more efficiently.

The developed software would also result in reduction of labour which would result in the reduction of expenses of the bakery.

1.4 References:

1. IEEE: IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.

2. Overall Description :

2.1 Product functions:

• Allow Customers to scroll through the menu and select the dishes he/she wants.
• Allow the Customers to cancel/edit the order any time before its prepared.
• Allow Customers to provide feedback regarding the food and overall service of the bakery.
• Allow Customers to request for a bill.
• Allow Customers to ask for help through the system.
• Show dish queues and their status, for Proprietor.
• Allow proprietor to perform CRUD (create, retrieve, update and delete) operations on Staff Members, Menu Items and Inventory.
• Allow Proprietor to mark orders complete.
• Allow the Proprietor to approve cancellation of the dish or order.
• Allow Proprietor to mark the bill as paid.
• Notify the Proprietor. when a particular order is complete.

2.2 User Classes and Characteristics:

There are four types of users that interact with our system(See Appendix B)
Firstly, there are proprietors.

We'll provide an interface for proprietors as well through which they are looking at the status of their order queues, but they will not interact with our system.
2.2.1 Customer Class
Customers interact with our system directly in order to place order, modify order, get bill and give feedback. We do not store any information related to customers in our system. The process of order taking starts from customers placing order and then the other series of events begin.

2.2.2 Proprietor

Proprietor can mark a dish as prepared when a proprietor tells him to do so. He can approve the cancellation of an order whenever a customer edits or removes a dish from his order. He can also assign a dish to a particular proprietor based on the specialty of the proprietor.

The specific proprietor will also provide their input when he marks the bill as paid when customers pay for their order or get the bill printed. Moreover, he gets a notification whenever a particular order is complete, or some customer asks for help through the system. Proprietor can also see tables in the hall and their status i.e. empty or filled.

Proprietor's job is to manage the inventory and other information related to the menu and proprietors in the system.

3.External Interface Requirements:

3.1 User Interfaces:

3.1.1 Customer Interface

The customer interface will contain three screens. All three screen will have a consistent layout.

3.1.1.1 Place Order

In this screen, the system shows a list of cards (UI Elements) of dishes. Each dish will have an image, its price per serving.

3.1.1.2 Timer and Edit/Cancel Order

After confirming the order, the user will be shown a timer screen. In this screen, customers will be shown "Edit Order" and "Cancel Order" buttons and a timer which shows the completion time of the order. There will also be a button to request for a bill.

3.1.1.3 Feedback

In feedback screen, at the top right corner a button for "Request Bill" will be shown. Beneath this button we will display a form which will have different multiple-choice questions and a submit feedback button.

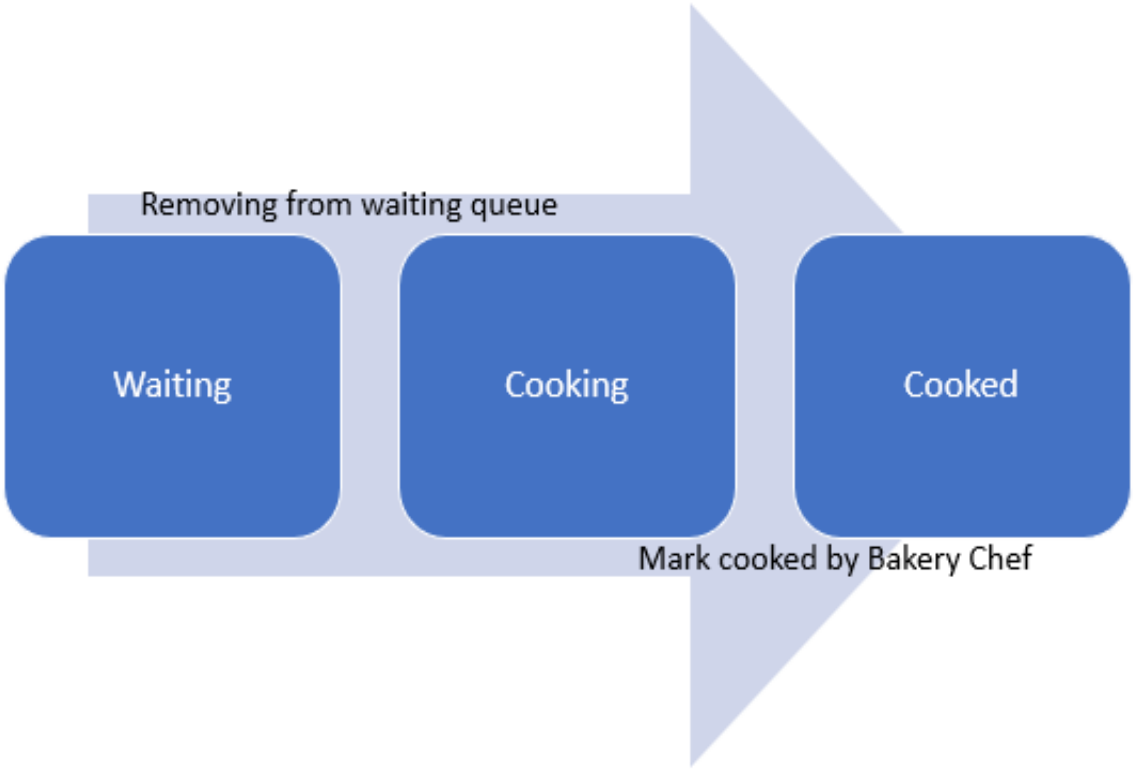
3.1.2 Proprietor

The system will show all the current orders in detail i.e. all the dishes of a particular order. In each order, there is a button which will be used to mark that dish cooked. Moreover, when customer wants to remove a dish from his order, system will show proprietor a notification to approve the removal of the dish.

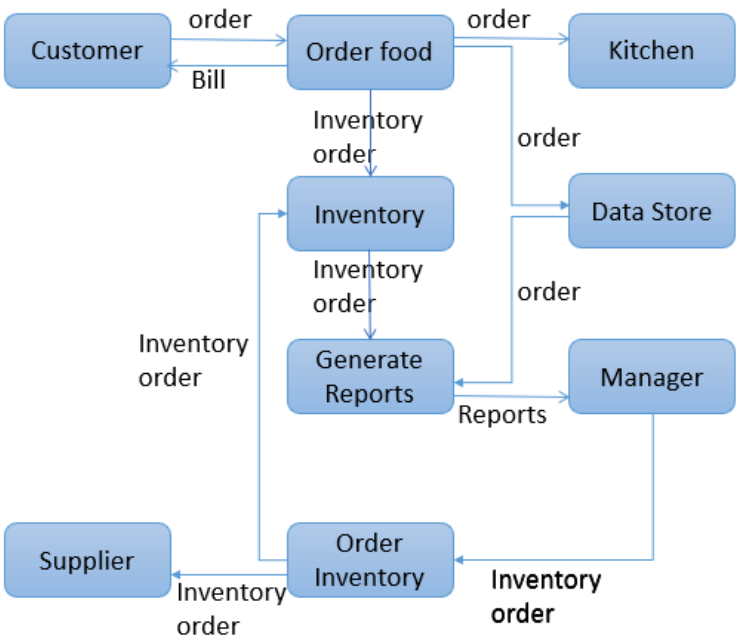
Proprietor will have a screen where he will get notification whenever an order is completed. System will notify the proprietor about the order number and table number. proprietor also has a screen where all orders are listed, and status button to mark the order as paid. Moreover, he also has an interface screen to see and the status of tables in the bakery as free/available.
As proprietor is authorized to perform CRUD operations on Staff Members, Menu Items and Inventory Items. He'll be having three different screens for Staff Members, Menu Items, and Inventory.

4. Analysis Models:

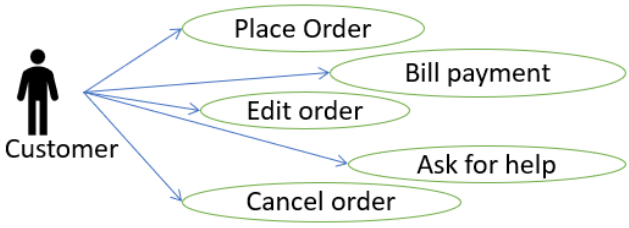
Dish State Diagram:

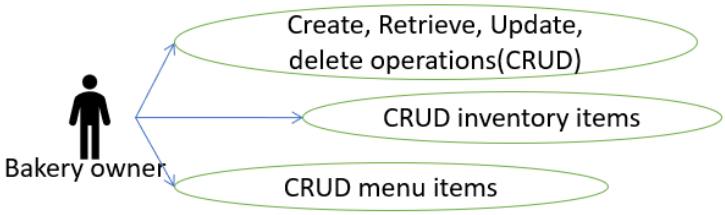


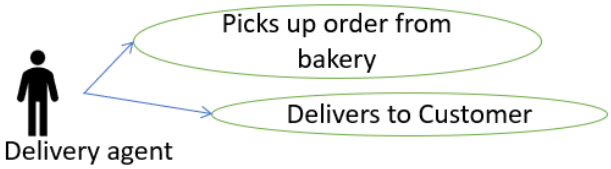
Data Flow Model:



Use case diagram:







5. System Requirements:

5.1 System feature 1:

ORDERED BAKERY ITEMS FROM THE BAKERY MENU TO BE PICKED UP OR DELIVERED:

♦ User shall place an item order from the bakery menu.
♦ User gets the ordered food item delivered to the desired address.
♦ Menu manager is responsible for creating and maintaining the 'Bakery menu.'
♦ In addition to this, users can order multiple items from the menu in a single order.

5.2 System feature 2:

ORDERED ITEMS FROM OTHER LOCAL BAKERY TO BE DELIVERED:

♦ Users shall place a meal order from the local Bakery menu.
♦ Bakery meal deliverer is responsible for delivering the meal to the user's desired address.

5.3 System feature 3:

CREATE, VIEW, MODIFY, AND DELETE MEAL SERVICE SUBSCRIPTIONS:

♦ User can subscribe a meal plan from the bakery menu
♦ Menu manager is responsible for creating and maintaining the various subscription plans.
♦ User can place multiple subscriptions from the menu in a single

5.4 System feature 4:

REQUEST MEAL DELIVERY:

♦ User has the option to schedule his delivery at his desired time.
♦ Users can change his delivery address before the meal is picked up by the delivery agent.

5.5 System feature 5:

ADD MY FEEDBACK:

♦ User has the feature to add his feedback about the Quality of food, Quality of Service and Experience in ordering the food through the Online bakery system.
♦ Feedback feature will be active to the user after two hours post the delivery.
♦ User's feedback will be monitored directly by the Menu-Manager

5.6 System feature 6:

SPECIFY OPERATING HOURS FOR ONLINE DELIVERY:
♦ Specify the operating hours for each bakery on the website.

5.7 System feature 7:

HIDE UNAVAILABLE PRODUCTS FROM DELIVERY:
♦ Either decide not to show unavailable products on the website or mark them as unavailable.

5.8 System feature 8:

DISPLAY THE ESTIMATED PREPARATION TIME FOR ONLINE ORDERS:
♦ The bakery estimates the time taken to prepare a dish ordered by a customer and the time taken to deliver it to his address and display it on their page of the website.

6. Other Nonfunctional Requirements

6.1 Performance Requirements:

The system must be interactive, and the delays involved must be less. So, in every action-response of the system, there are no immediate delays. Since User Experience (UX) is critical to the success or failure of our system in the market and performance is UX. The system should support more than 1000 user to checkout at the same time.Responses to queries shall take no longer than 7 seconds to load onto the screen after the user submits the query.The system shall display confirmation messages to users within 4 seconds after the user submits information to the system.In case of scrolling through the menu there should be a delay of no more than 2 seconds before the next page of menu items is displayed. The order should be placed in pending orders and be visible to the bakery owner in less than 5 seconds to start the preparation.

Cancel Order/ updates must be made with little delay to avoid delivery delay. Also, when connecting to the Firebase server the delay to make a successful connection should be less for effective real time communication.

6.2 Safety Requirements:

The software is completely environmentally friendly and does not cause any safety violations.

6.3 Security Requirements:

User's personal information like phone number and credit card information should be encrypted before storing in databases.

6.4 Software Quality Attributes

The system is up and running for most of the time and the server is not down for more than a few minutes to avoid inconvenience of the customers.There can be a change in the menu and information stored in the database about employees and inventory. If need arises in the future, software can be modified to change the requirements. Software should be easily repaired if a fault occurs and should be easily installed on devices and would run smoothly according to the requirement. No matter how many orders are placed, the system must give the correct results. Current version can be used in the future versions with more functionality added.

6.5 Business Rules:

- The Bakery owner's interface contains the view of tables that are free, and can just view and doesn't provide any input to the system.
- Once the bill is paid, the bakery owner can mark the order as paid
- The Bakery owner has access to perform add, delete, update operations on the database for menu, inventory, employees and no other person can modify the data in the db.
- Customers can place orders from the list of available items and can update orders and pay bills.
- The Bakery owner assigns orders and can update the queues and has an additional functionality of load balance.
- Delivery agents can only view the orders and cannot remove an order from their queue. Only the owner can interact with the queues containing orders.

7. Other Requirements:

OR-1: Users can set the mobile application to his/her preferred language. (English or Chinese)

OR-2: We should use cache to speed up our application.

Appendix A: Glossary

Cache- It is a high-speed data storage layer which stores a subset of data, typically transient in nature, so that future requests for that data are served up faster than is possible by accessing the data's primary storage location.

Firebase- It is categorized as a NoSQL database program, which stores data in JSON-like documents.

Appendix B: Field Layouts

Field	Length	Data Type	Description	Is Mandatory
Account Number	16	Numeric		Y
ISFC code	11	Alphanumeric		Y
Card Amount	20	Numeric		Y
Status	25	Alphanumeric	Status of rejection	Y
Customer Name	60	String		Y
Reject Reason Code	4	String	Rejection reason code in case mandate in rejected	N
Address	100	Alphanumeric		Y
Phone number	10	Numeric		Y
Email address	60	Alphanumeric		N

Registration Report	Transaction Report
Bank Account Number	Transaction Reference Number
ISFC Code	Bank Account Number
Bank Name	ISFC Code

Account Status	Bank Name
Account Type	Customer Name
Customer Name	Card Number
Card Number	Debit Transaction Amount
SI Start Date	Transaction Date

Appendix C: Requirement Traceability matrix