Functional and Non-functional requirements

The objective of this project will be to create a restaurant management system web application which will allow the user to browse through a variety of menu options and place and pay for an order for takeaway. Within this system, staff members of the restaurant will also be able to log in, using credentials based on their roles, and be able to manage incoming customer orders, edit menu items and notify the customer of their order's status.

This part of the document will provide an in-depth discussion of the functional and non - functional requirements associated with this project.

Functional Requirements

- Navigational bar linking to different aspects of the site.
    - Should highlight depending on the page they customer is on.
    - Slide show showing different dishes served by the restaurant.
- Menu Page
    - List of items for starters, mains and desserts along
    - Each item should have its prices listed alongside it (right hand side)
    - Each item should display the following information:
        - Allergen information - if the items are Vegan/Vegetarian or have nuts etc.
        - Ingredients used for the dish.
        - Macro information.
    - Implement a filtering system for the type of meal (starter, mains or desserts).
    - Each item of food should also have a "add to basket" option so that users can choose what they want to eat.
- Basket page
    - Should display all items customers have selected to add to their basket
    - Each item should have a "remove from basket" option.
    - Each item should have an "update" quantity feature.
    - Each item should be listed with an individual price.
    - Full total should be displayed – this should be updated automatically each time a user adds/removes more items.
    - A "pay" button
    - A final confirmation for payment along with continue and cancel button.
- Payment page
    - Upon confirmation of the order page, customer should be directed to a payment page where they can input their card details.
    - A Total price should be displayed along with a form to fill out.
    - A "pay" and "cancel" button should be displayed at the bottom of the form.
    - Successful and unsuccessful submissions should be dealt with responsibly
        - Successful submissions should remove all items from basket
        - Successful submissions should proceed payment successful page
        - unsuccessful submissions should display a message in red, notifying the customer that the payment was unsuccessful.
        - unsuccessful submissions should re-direct the user back to the basket page with no items leaving the basket.

- Employee Log in page
    - Allow Staff to log in and see their specific pages without letting anyone else see
- Kitchen Staff Page
    - Kitchen staff should be able to see orders that have been placed which include:
        - List of items ordered
        - Customer details
        - Time of order
    - Should be able to edit each item to reduce quantity, to facilitate order completion.
    - Each order should have a “Complete order” button to allow staff to remove orders which have been served
    - There should be an undo button in case mistakes are made when completing orders.
    - A log out button on the top of the screen

Non-functional

- High Accessibility/Usability:
    - The user interface for viewing, updating and removing orders should be intuitive and easy to navigate for kitchen staff with minimal training.
    - Customer should be able to use the system intuitively.
    - The website should be compatible with all common web browsers such as Google Chrome, Mozilla Firefox, and Safari.
    - The website should be responsive and display correctly on various devices, including desktops, tablets, and smartphones.
- Security/Privacy:
    - Access to order information should be restricted to authorised staff only i.e. Kitchen staff, with appropriate authentication mechanisms in place.
    - Sensitive information should be stored securely, e.g. stored passwords should be encrypted.
- Scalability:
    - The system should be capable of handling a growing number of orders without degradation in performance.
- Speed/Performance
    - The system should notify the kitchen staff of a new order immediately (< 1sec).
    - Adding and removing items from the basket shouldn't take longer than 1 second.
- Maintainability:
    - The codebase should be well-documented and adhere to coding standards to facilitate future maintenance and updates.
    - The system architecture should be modular to allow for easy integration of new features or modifications.
- Error Handling:
    - The system should provide clear and informative error messages to users in case of input validation errors or system failures. E.g. if the users payment is decline, then the following steps of the systems flow should make the customer aware of this.