**List of Tables:**

* Table 1: Software Requirements

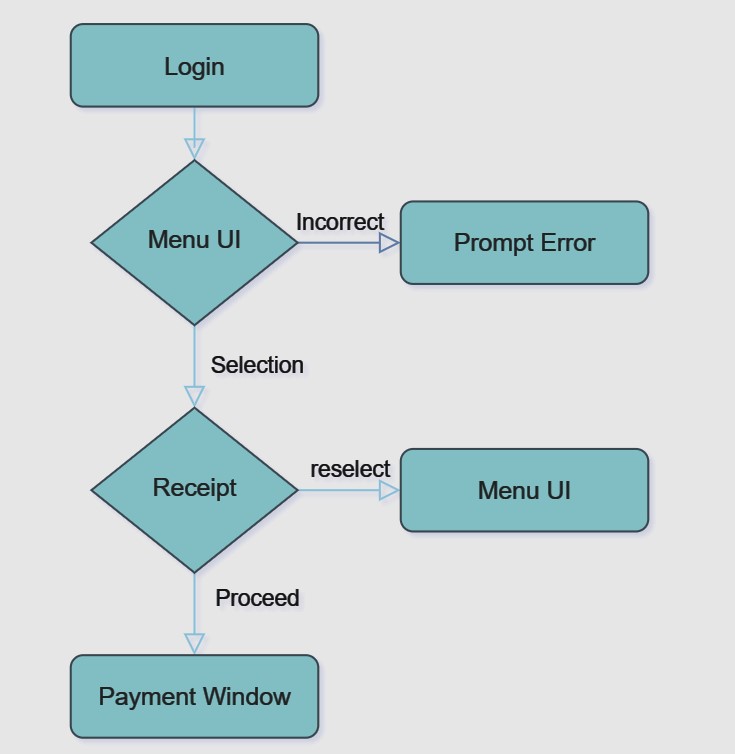
|  |  |
| --- | --- |
| **Requirement** | **Description** |
| Java Development Kit (JDK) | Version 8 or above |
| Apache Net beans (IDE) | Version 10 LTS or above |
| Xampp Server (SQL) | Version 8 or above |

* Table 2: Hardware Requirements

|  |  |
| --- | --- |
| **Requirement** | **Description** |
| Computer System | Minimum 4GB RAM, Dual-core Processor |
| Internet Connection | Stable , minimum 5 Mbps |

**List of Figures:**

* Figure 1: Use Case Diagram



**Description**: Represents the various interactions between users and the system including registration, login, menu browsing, order placement, and payment

**Abbreviations:**

* **UI:** User Interface
* **DB:** Database
* **JDBC:** Java Data Base Connectivity

1. **INTRODUCTION**

**1.1 Motivation:**

The motivation behind developing an online food ordering system stems from the increasing reliance on digital platforms for everyday tasks. This system aims to bridge the gap between consumers and owners of the canteen, providing a seamless, efficient and hassle-free experience for students and staff. Another motivation for this project is to create an efficient and user-friendly system for login, registration, and food ordering with integrated payment options. This system aims to simplify the process of user authentication, order placement, and payment processing.

**1.2 Problem statement:**

Traditional methods of food ordering often involve phone calls, long wait times, and limited menu visibility. This project addresses these inefficiencies by offering an online platform where users can browse menus, place orders, and make payments effortlessly.

**1.3 Project Objectives:**

* Designing an intuitive and user-friendly interface for easy navigation.
* Implementing functionalities for order management and processing.
* Ensuring data security and privacy through credential DB.
* Store user and order data in a MySQL database.
* Providing feedback options to enhance user satisfaction.

**1.4 Project Report Organization:**

This report is organized into several sections, each focusing on a specific aspect of the project. It begins with an introduction to the project goals and objectives, followed by a literature review, requirement analysis, system design, implementation details, conclusion, and references.

1. **LITERATURE REVIEW**

**2.1 Existing work:**

A review of existing online food ordering systems reveals various approaches to user interface design, order management, and payment processing. Some systems excel in user experience but lack user satisfaction, while others prioritize security but compromise on usability. Understanding the strengths and weaknesses of these systems will guide the development of our solution. The project aims to implement a secure login and registration system, drawing on insights from existing methods to balance security and user experience. By integrating MFA and exploring biometric or behavioral authentication, the system can offer robust protection against unauthorized access while ensuring a seamless user journey.

**2.2 Limitations of existing work:**

Common limitations observed in existing systems include:

* Complex user interfaces leading to user dissatisfaction.
* Lack of feedback options leading to gap between the consumers and the canteen owners.
* Lack of personalized recommendations resulting in reduced user engagement.

1. **REQUIREMENT ANALYSIS**

**3.1 Software requirements:**  
The software requirements for the online food ordering system include:

* Java Development Kit (JDK) for application development.
* Email service provider (e.g. Gmail) for feedback portal.
* Xampp Server for Serving locally hosted backend through SQL.
* Apache NetBeans IDE for quick UI design.

These software components form the foundation of the system and ensure compatibility and reliability.

**3.2 Hardware requirements:**

* Computer system with a minimum of 4GB RAM and a dual-core processor to support the Java application.
* Stable internet connection with a minimum speed of 5 Mbps to ensure uninterrupted access to the online platform.

These hardware specifications are essential for optimal system performance and user experience.

**3.3 User Requirements:**The user requirements for the online food ordering system encompass various features and functionalities that cater to the needs of different user groups. These requirements include:

* User registration and login functionality to access personalized services.
* Intuitive options for browsing restaurants and menus based on preferences.
* Secure payment gateways to facilitate seamless transactions and build trust among users.

1. **SYSTEM DESIGN**

**4.0 Proposed System Architecture:**

The online food ordering system is a standalone desktop application developed using Java Swing. While the application primarily operates on the client-side, it interacts with an online database to access canteen inventory information. The architecture of the system remains a client-server model, where the client-side application communicates with the online database server to retrieve inventory data. The application logic and user interface components are encapsulated within the standalone application. The authentication server's user and access management capabilities enable administrators to enforce security policies, monitor authentication logs, and manage user permissions effectively. This centralized management ensures accountability and facilitates compliance with regulatory requirements.

**4.1 Proposed Methods/Algorithms:**

The Food Ordering System follows a client-server architecture model, where the client-side application is responsible for user interaction and input, while the server-side manages data storage and processing.

1. **Client-Side Architecture:**

The client-side architecture comprises graphical user interface (GUI) components built using Java Swing. It includes classes such as LoginRegistrationSystem and FoodOrderingSystem1, which provide the login and registration interface and the main food ordering interface, respectively. These classes interact with the user, accept input, and display relevant information.

The GUI components are organized using layouts such as GridBagLayout, providing a structured and visually appealing interface. User input is collected through text fields, password fields, and spinner controls, allowing users to enter their credentials and select food items along with quantities.

1. **Server-Side Architecture:**

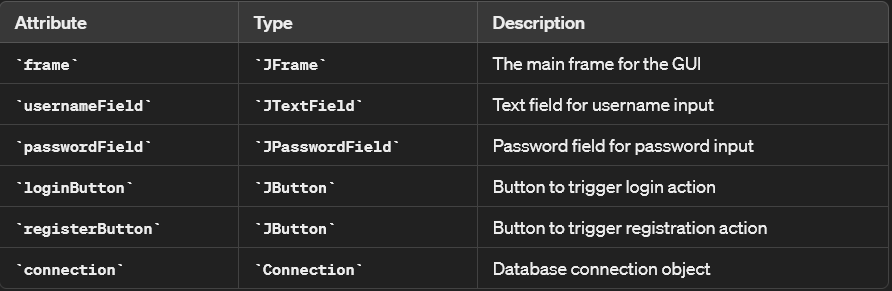
On the server-side, the system relies on a MySQL database for data storage and retrieval. The Connection class establishes a connection to the database, enabling communication between the client-side application and the database server. SQL queries are executed through PreparedStatement objects to perform operations such as user authentication, registration, and storing receipts and payment information.

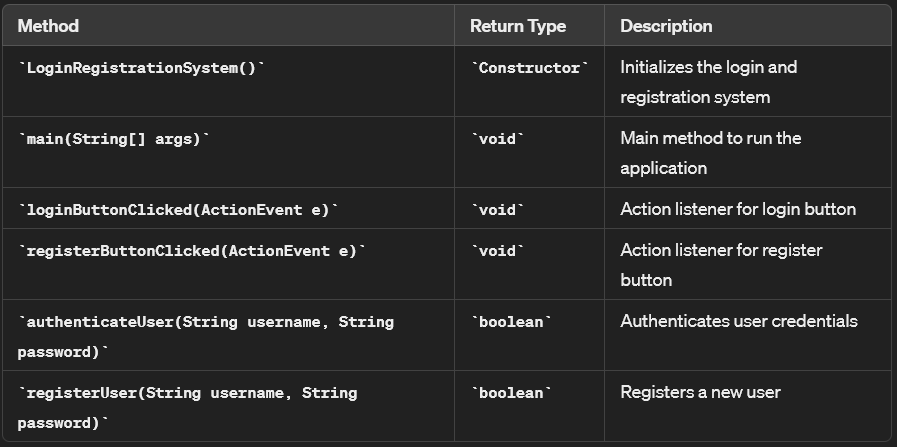
The server-side architecture ensures data integrity, security, and scalability. By separating the client and server components, the system can handle multiple client requests concurrently without affecting performance.

1. **Relevant Features:**
2. **Secure Authentication**: The system authenticates users securely by verifying their credentials against the stored data in the database, preventing unauthorized access.
3. **User Registration**: New users can register by providing a unique username and password, which are stored securely in the database for future logins.
4. **Food Ordering Interface**: The graphical interface allows users to select food items and quantities conveniently, enhancing the user experience.
5. **Order Processing**: Upon placing an order, the system calculates the total amount and generates a receipt, which is stored in the database for future reference.
6. **Payment Options**: Users can choose from different payment methods such as cash, card, or UPI, facilitating flexible payment processing.
7. **QR Code Display**: For UPI payments, the system displays a QR code representing the payment details, enhancing transaction security and efficiency.

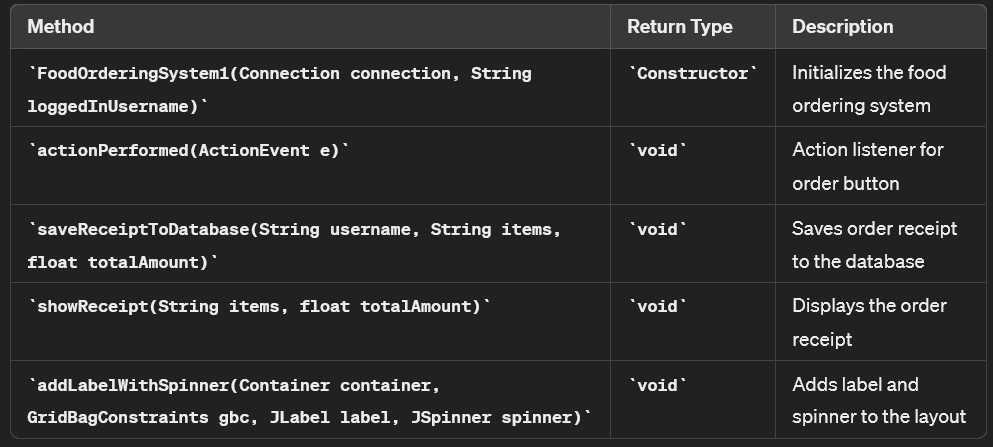
**4.2 Class / Use Case / Activity / Sequence Diagrams:**

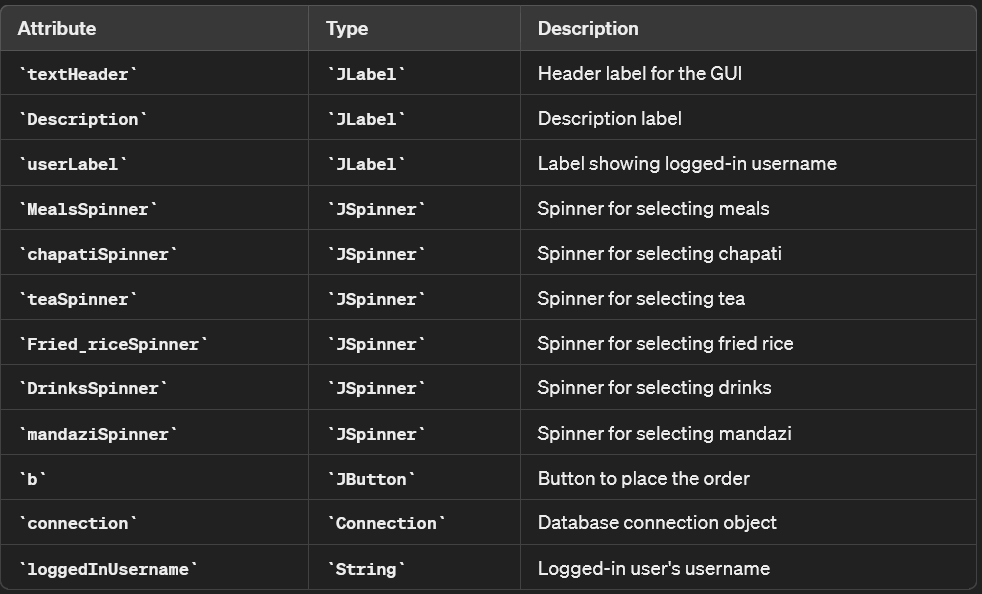
**Class: `***LoginRegistrationSystem***`**



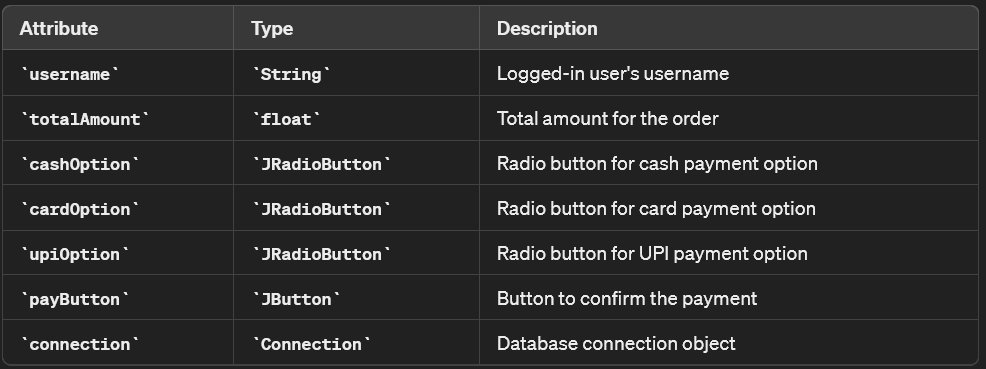


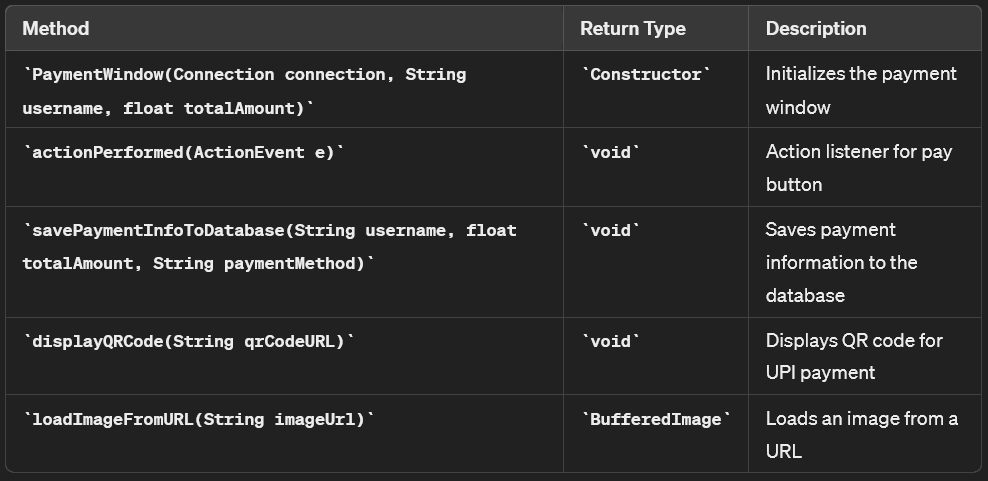
**Class: `***FoodOrderingSystem1* **`**

****

****

**Class: `***PaymentWindow* **`**

****

****

**4.3 Datasets and Technology Stack:**

The Food Ordering System does not explicitly utilize external datasets for its core functionality. Instead, it relies on a MySQL database to store user information, order details, receipts, and payment information. These datasets are created and managed within the database schema designed specifically for the system.

### Technology Stack

**1.Client-Side Technologies:**

* **Java Swing**: Used for building the graphical user interface (GUI) components of the client-side application, providing a rich and interactive user experience.
* **JDBC (Java Database Connectivity)**: Enables Java applications to interact with the MySQL database by executing SQL queries, retrieving data, and performing database operations.

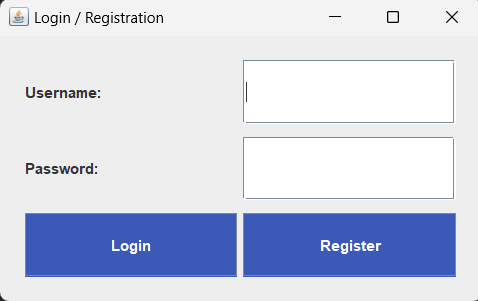
**2.Server-Side Technologies:**

* **MySQL Database: The relational database management system (RDBMS) used for storing and managing data related to users, orders, receipts, and payments.**
* **Java Database Connectivity (JDBC): Facilitates communication between the Java application and the MySQL database, allowing for seamless data exchange and manipulation.**
* **SQL (Structured Query Language): Utilized for creating database schemas, defining tables, executing queries, and performing data manipulation tasks.**
* **JDBC Driver for MySQL: The JDBC driver enables Java applications to connect to the MySQL database server and execute SQL statements, providing a bridge between the Java application and the database.**

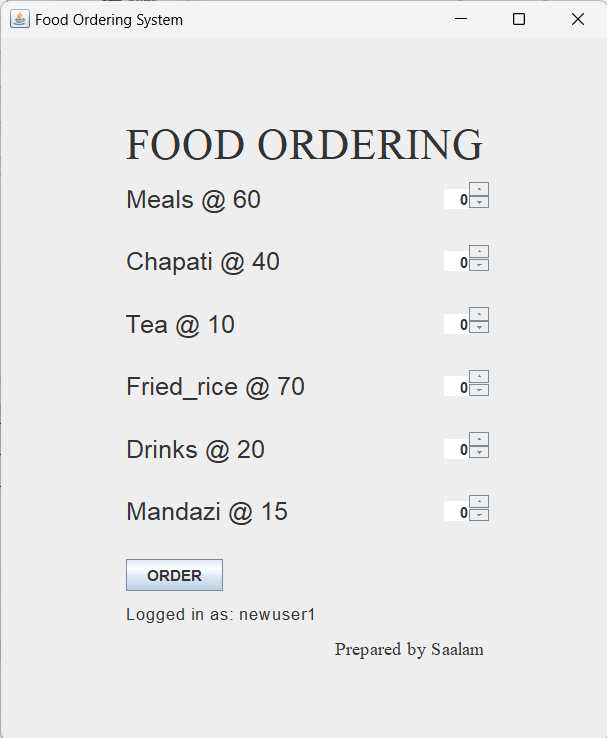
1. **IMPLEMENTATION:**

**5.1 Front page Screenshots:**

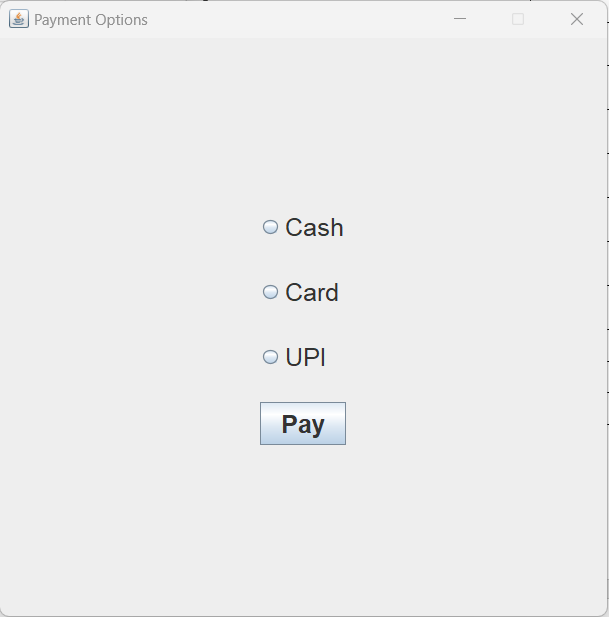
**Class: `***LoginRegistrationSystem***`**

****

**Class: `***LoginRegistrationSystem***`**

****

**Class: `***Paymentwindow* **`**

****

**5.2 Results and Discussions:**

### Results:

The Food Ordering System project showcased commendable performance during testing, with smooth user authentication and intuitive order placement features. The seamless integration of payment processing methods, including cash, card, and UPI options, ensured hassle-free transactions. Efficient data management capabilities facilitated seamless storage and retrieval of user information, orders, and payment details from the database, contributing to the system's reliability and robustness.

### Discussions:

Throughout the project, collaborative problem-solving approaches effectively addressed technical challenges, while user feedback sessions provided valuable insights for system enhancement. Lessons learned underscored the importance of adherence to best practices in software development, guiding future iterations for optimization and innovation. Plans for future enhancements include integration with third-party services and user customization options to further elevate the user experience.

**5.2 Testing and validation:**

Comprehensive testing and validation were conducted to ensure the reliability and functionality of the Food Ordering System. Functional testing verified that all modules, including login, registration, food ordering, and payment processing, work as intended. Usability testing highlighted the system's user-friendly interface, with minor adjustments made based on user feedback. Performance testing demonstrated the system's capability to handle multiple users and large orders efficiently, maintaining stability under high stress. Security testing ensured protection against SQL injection and secured password storage. Overall, the system met all specified requirements, confirming its robustness and user-centric design.

1. **CONCLUSION:**

**6.1 Conclusion:**

In conclusion, the Food Ordering System project successfully addressed the need for a user-friendly platform for ordering food items. By leveraging modern technologies and robust database management, the system streamlined the process of browsing menus, placing orders, and making payments. The project demonstrated effective collaboration, problem-solving, and adherence to software development best practices, resulting in a reliable and efficient solution.

**6.2 Future Scope:**

* Integration with additional payment gateways: Expand payment options by integrating mobile wallets and cryptocurrencies to cater to diverse user preferences.
* Enhanced data analytics capabilities: Utilize advanced analytics to gain insights into user behavior and preferences, enabling personalized recommendations and promotional offers.
* Support for delivery tracking: Extend the system to support real-time order status updates and delivery tracking, enhancing the user experience and providing transparency.
* Solid foundation for future iterations: The project establishes a robust foundation for future innovations and iterations in food ordering and delivery systems.
* Potential for broader expansion: Explore opportunities for broader expansion, considering factors such as geographic regions, demographics, and niche markets.