



## *Green University of Bangladesh*

*Department of Computer Science and Engineering (CSE)  
Semester: (Spring, Year: 2025), B.Sc. in CSE (Day)*

---

# **Campus360 Solutions**

---

*Course Title: Integrated Design Project II*

*Course Code: CSE 406*

*Section: 221\_D19*

### Students Details

<b>Name</b>	<b>ID</b>
Afzal Hossain	221002628
Amjad Hossain Piash	221002221
Zakia Haque Zitu	221002224

*Submission Date: 08/05/2025*

*Course Teacher's Name: Mr. Montaser Abdul Quader*

### Lab Project Status

**Marks:**

**Signature:**

**Comments:**

**Date:**

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Overview . . . . .	3
1.2	Problem Domain . . . . .	3
1.3	Motivation . . . . .	4
1.4	Objective . . . . .	5
<b>2</b>	<b>Literature Review</b>	<b>6</b>
<b>3</b>	<b>Methodology</b>	<b>8</b>
3.1	Software Requirements Specification(SRS) . . . . .	8
3.1.1	Hardware requirement . . . . .	8
3.1.2	Software requirement . . . . .	8
3.1.3	Network requirement . . . . .	9
3.1.4	Functional Requirements . . . . .	9
3.1.5	Non-Functional Requirements . . . . .	10
3.2	SDLC . . . . .	11
3.3	Feasibility Study . . . . .	13
3.3.1	Technical Feasibility . . . . .	13
3.3.2	Operational Feasibility . . . . .	13
3.3.3	Financial Feasibility . . . . .	14
3.4	Architectural Design . . . . .	15
3.4.1	High Level Design . . . . .	15
3.4.2	Low Level Design . . . . .	16
3.5	Block Diagram . . . . .	16
3.6	DFD . . . . .	17
3.6.1	Level 0 diagram . . . . .	17
3.6.2	Level 1 diagram . . . . .	18

3.6.3	Level 2 diagram	20
3.7	UML Use Case Diagram	22
3.8	UML Sequence and Communication Diagram	24
3.9	UML Class Diagram	26
3.10	Final Output	28
<b>4</b>	<b>Software Testing Report</b>	<b>35</b>
<b>5</b>	<b>Conclusion</b>	<b>43</b>
5.1	Discussion	43
5.2	Social Impact and Benefit	43
5.3	Limitation	44
5.4	Future Work	44

# Chapter 1

## Introduction

### 1.1 Overview

The Campus360 Solution is a web-based system designed to improve various aspects of university life by offering multiple features. It includes a cafeteria management system where students and faculty can pre-order meals, check estimated serving times, and order using table-specific QR codes. The bus seat management feature helps students check available seats, manage bus occupancy, and use polls to schedule shuttle services effectively. For clubs, it provides dedicated pages to showcase events, allow registrations, and enable members to manage activities. Additionally, the system includes a communication hub where students and alumni can share research and project details, keeping collaboration and experience-sharing within the university community.

### 1.2 Problem Domain

Universities often face many challenges in managing day-to-day services and activities. The faced problem are given below:

The major issue occurs in the cafeteria, where students and faculty cannot always get their desired meals due to limited stock or for long waiting in queues. There is no system to pre-order meals, which leads to uncertainty about meal availability. Additionally, students often have to wait without knowing the estimated time to receive their orders, causing trouble.

The transport system is another area with many problems. Students sometimes cannot find available seats on university buses because there is no way to check seat availability online. Different class schedules mean students finish at different times, making it difficult to manage bus schedules efficiently, especially when buses return from the university. The number of students varies across time slots, and staff often struggle to decide how many buses are needed due to a lack of real-time information.

Many universities do not have a dedicated page for clubs to manage their events. Students miss out on important information about ongoing and upcoming activities, and club members or moderators have no platform to manage or update event details easily.

Another gap is the absence of a communication platform for students and alumni. Cur-

rent students cannot share their research or project achievements, and alumni have no way to find students for assistance with part-time or full-time opportunities. This creates a missed opportunity for professional networking and collaboration within the university community.

## 1.3 Motivation

The Campus360 Solution aims to improve campus life by solving common problems faced by students, faculty, and staff. The key reasons for developing this system are:

- **Improving Cafeteria Management:**

The cafeteria often faces problems like no more meals or waiting in long lines for meals, especially during busy hours. With the pre-order system, students and faculty can reserve meals early, making sure they get the food they want without waiting. The system also shows the estimated time to get the meal, making it easier for users to plan.

- **Improving the Transport System:**

There is no bus system to allow students to check seat availability. This causes confusion is any bus has an available seat or not. The new system lets students see which buses have empty seats. A polling system will also help bus staff understand how many students need the bus at different times, making planning easier.

- **Managing Club Activities:**

Many universities don't have a clear system for managing club events. This makes it harder for students to join or keep track of events. With the new system, clubs will have a dedicated page to show upcoming events and allow students to register easily. Club members and moderators can also update event details. Moderators can create weekly or monthly context on this page.

- **Making Communication Easier:**

Students and alumni often miss opportunities to work together on research or projects because there's no easy way to connect. The communication hub in Campus360 will allow students and alumni to create profiles, share their work, and offer or ask for help on projects. This will help students gain valuable experience while giving alumni a chance to find help for their projects or jobs.

## 1.4 Objective

The primary objectives of the *Campus360 Solution* are:

- **Simplifying Cafeteria Operations:** Enable students and faculty to pre-order meals, track real-time availability, and access estimated serving times. Generate reports for cafeteria staff to manage inventory more efficiently.
- **Improving Transport System Efficiency:** Provide a system to check bus seat availability and introduce polling to help bus staff plan schedules effectively.
- **Enhancing Club Activity Management:** Create web pages for clubs to display events, manage registrations, and update activity details.
- **Building a Communication Hub:** Connect students and alumni through profiles to share research, projects, and opportunities for collaboration.

# Chapter 2

## Literature Review

The Campus360 Solution project aims to build a single web-based system to make campus life easier for students, teachers and staff. This system integrates key features such as cafeteria pre-ordering, club activity management, bus seat reservation, communication hubs, and student and teacher portals.

Many universities provide only basic online services like academic portals and notices. However, they don't have integrated platforms for managing important non-academic activities that are essential for students' daily campus life. Without features like meal management, transportation systems, and club activity pages, students face problems and fewer opportunities to stay engaged.

Green University of Bangladesh focuses on academic details and faculty information but lacks features for managing extracurricular activities, cafeteria pre-ordering, or transportation systems [1]. Similarly, the University of Scholars and Primeasia University offer student portals and academic information but do not include cafeteria management, real-time transportation features, or a system for organizing club events [2] [3]. The State University of Bangladesh provides faculty portal and academic notices but does not address the need for managing bus seat availability or club events [4]. Similarly, Southeast University provides various academic resources, yet it lacks tools for efficient meal pre-ordering, club activity tracking, or communication hubs to connect students and alumni [5].

Website/Software Name	Key Features	Tools and Technology Used	Limitations
Green University of Bangladesh [1]	Provides academic information, faculty directories, and notices.	HTML, CSS, JavaScript, PHP, MongoDB	Lacks features for managing cafeteria pre-ordering, transportation seat tracking, and club activities.
University of Scholars [2]	Offers academic portals and details on programs and events.	HTML, CSS, PHP, MongoDB	No cafeteria management system, no transportation or club activity pages.
Primeasia University [3]	Includes online admission systems and academic resources for students.	HTML, CSS, PHP, Bootstrap, MongoDB	Does not provide real-time transport seat availability or any club event management system.
State University of Bangladesh [4]	Displays academic details, faculty profiles, and notices for students.	HTML, CSS, PHP, WordPress, MongoDB	Lacks cafeteria pre-ordering, bus seat tracking, and a platform for communication between students and alumni.
Southeast University [5]	Offers academic portals, faculty information, and notice boards.	HTML, CSS, PHP, WordPress, JavaScript, MongoDB	No features for managing extracurricular activities, communication hubs, or real-time transport systems.

Table 2.1: Comparison of existing university websites.

The Campus360 Solution fills these gaps by offering key features that address the limitations of current university websites. For example, the cafeteria system lets students and staff pre-order meals, check serving times, and use table-specific QR codes for ordering, making meal access quick and easy. A bus seat management feature allows students to view and book seats online, while staff can optimize shuttle schedules using a polling system. The solution also includes a club activity management system with dedicated pages for clubs, where students can register for events, and moderators can manage event details, encouraging more participation. A communication hub connects students and alumni through profiles highlighting research and projects, fostering collaboration and creating new opportunities. Finally, the integrated student and teacher portals bring together academic resources, faculty directories, and notices in one user-friendly platform, simplifying campus life for everyone.

# Chapter 3

## Methodology

### 3.1 Software Requirements Specification(SRS)

#### 3.1.1 Hardware requirement

- **Scanner**

The bar-code will be attached to the Student ID card. When we will take our meal from the cafeteria, they can scan our ID card by scanner to see our pre-ordered food item. Also When we enter the bus, we will have to scan the ID Card, Which will check whether the person is a student, faculty, or someone else. At that time, it will also count how many people have boarded the bus. Finally show the seat availability on our website.

- **Computer**

- Processor: Core i5 8gen
- RAM: 8 GB
- Storage: SSD 512 GB

#### 3.1.2 Software requirement

- **Code Editor (VSCode)**

A code editor like Visual Studio Code will be used for development, providing an environment for writing, testing, and debugging code.

- **Technologies and Tools Used**

The project will utilize a mix of front-end and back-end technologies:

- **HTML, CSS:** For creating the structure and design of web pages.
- **CSS Framework:** Such as Bootstrap for responsive and streamlined design.
- **JavaScript (React):** For creating dynamic, interactive, and user-friendly front-end interfaces.
- **Node.js:** Node.js is a runtime environment that allows JavaScript to run on the server side. It is lightweight, fast, making it suitable for real-time applications like the Campus360 Solution.

- **Express.js:** Express.js is a web application framework for Node.js that simplifies back-end development.
- **MongoDB:** For database management to store user information, meal orders, bus data, etc.
- **Version Control System (Git & GitHub)**  
Version control will be managed using Git and GitHub to ensure smooth collaboration, versioning, and backup of project files.

### 3.1.3 Network requirement

- **Domain Hosting**  
A registered domain name and hosting service for public access to the system.
- **Web Server:**  
A server capable of hosting this application, ensuring reliable access for users.
- **Network Security:**  
Firewalls and SSL certificates to ensure secure communication and protect user data.
- **Networking Equipment:**  
Routers and switches for secure and uninterrupted connectivity.
- **Cloud Services (Optional):**  
Services like AWS or Google Cloud for scalable deployment and data backup.

### 3.1.4 Functional Requirements

- **Pre-Food Order**  
A system that allows students to place meal orders the day before and get a unique order code for verification.
- **Meal Availability Tracking**  
Displays the amount of available food and snacks in the cafeteria in real time.
- **Estimated Waiting Time** After placing an order, the system shows the estimated waiting time for meal preparation.
- **Table QR Code Ordering** Allows students and faculty to order meals by scanning QR codes at cafeteria tables. Faculty orders receive priority.
- **Bus Seat Availability**  
Allows students to check available seats on university buses by scanning ID cards at bus entry points.
- **Bus Polling System** Provides a poll for students to indicate their need for shuttle buses at specific times, helping staff plan bus schedules.

- **Club Pages**  
Dedicated pages for each club, displaying upcoming events, registration forms, and past activities.
- **Communication Hub** A platform where students and alumni can create profiles, showcase research or projects, and collaborate.

### 3.1.5 Non-Functional Requirements

#### 1. Product Requirements

##### (a) **Usability Requirements**

The system will be designed to be user-friendly and accessible to a wide range of users, including secondary school, high school, and university students.

##### (b) **Portability Requirements**

The system will be compatible with different devices such as mobile phones, tablets, and desktop computers, and will run on various operating systems such as Windows, Mac, Linux and so on.

##### (c) **Reliability Requirements**

The system will ensure user data confidentiality and personal information. Data will be backed up regularly to prevent loss in case of a system failure.

##### (d) **Efficiency Requirements**

###### i. **Performance Requirements**

The system should respond to user inputs quickly, especially when handling real-time data like meal orders and bus seat availability.

###### ii. **Space Requirements**

To minimize storage usage, the system will utilize efficient update frameworks that help reduce space consumption.

#### 2. Process requirement

##### (a) **Process Requirements (Agile)**

The project will follow Agile methodologies, specifically V-shape, to ensure iterative development, frequent updates, and continuous feedback from stakeholders.

##### (b) **Delivery Requirements**

The system will be delivered in phases, starting with the cafeteria and bus systems, followed by other features like club pages and student portals.

##### (c) **Implementation Requirements**

The system will be implemented incrementally to allow testing of each feature before moving on to the next phase.

##### (d) **Standards Requirements**

- ISO
- W3C
- IEEE Standard Association.
- Internet Engineering Task Force (IETF).

- Web Standards Project.

### 3. External Requirements

- **Interoperability Requirements**

The system will work with existing university systems and databases. This will include interfacing with student information systems, academic portals, and other relevant campus technologies.

- **Ethical Requirements**

Ensure all users have equal opportunities to access all features and services. Only Faculty can access some special features. The system will not misuse personal data.

- **Legislative Requirements**

- **Privacy Requirements**

The system ensures user data confidentiality and protects personal information. Also backups of regular data are performed to prevent loss in case of system failure.

- **Safety Requirements**

The system will implement secure authentication measures to protect user accounts. Ensuring secure authentication for accessing the platform, and providing safety features related to real-world applications, such as verifying student and faculty identity when boarding buses or during campus events.

## 3.2 SDLC

Evaluation Criteria for SDLC Model Selection are:

- **Well-Known Requirements**

Meal preordering, bus seat management, club activity pages, and student/faculty portals are just a few of the well-defined, constant needs for the Campus360 Solution project. These characteristics are essential and cannot be compromised. These clearly defined requirements must serve as the foundation for the system, so selecting an SDLC model that supports organized and predictable requirements is essential.

- **Technological knowledge**

Developing the project requires expertise in modern web technologies (HTML5, CSS3, JavaScript), backend systems (Node.js, PHP), and databases (MongoDB). The team must understand these technologies for smooth implementation.

- **Efficiency**

Efficiency is critical because the system will handle multiple real-time operations at once. For example, the cafeteria system needs to process pre-orders quickly, and the bus tracking feature must update seat availability on the spot. If the system lags or fails to handle high demand, it would frustrate users and defeat the purpose of campus services. That's why efficiency is a top priority.

- **Risk analysis**

Every system faces potential risks, and this one is no exception. But this project might face some risk such as, while handling multiple orders simultaneously.

- **User Testing Ability**

The system will be used by many different people—students, faculty, cafeteria staff, and club organizers—so it needs to be easy for everyone. User testing ensures that the interface is intuitive and that all features, like meal ordering and event registration, work as expected. By getting feedback from real users during development, the team can identify and fix issues before launch.

- **Dependability and Security**

This project involves sensitive data, like student IDs and personal preferences for meals or event registrations. The system must be trustworthy, ensuring data is safe from breaches and always accessible when needed. For instance, the ID card scanner must work properly.

- **Time consuming**

Meeting deadlines is always important, especially for a project designed to improve daily campus activities. A time-efficient approach to development ensures the system is ready to use when students and staff need it most.

Table 3.1: 1: Comparison matrix with different models

Priority	Criteria	Waterfall	V-Shape	Iterative	Spiral	Agile	Prototype
6	Well-known requirements	Yes (6)	Yes (6)	No (0)	No (0)	No (0)	No (0)
5	Technological knowledge	Yes (5)	Yes (5)	No (0)	No (0)	No (0)	No (0)
7	Efficiency	No (0)	Yes (7)	Yes (7)	Yes (7)	No (0)	Yes (7)
6	Dependability & Security	No (0)	Yes (6)	No (0)	Yes (6)	No (0)	No (0)
4	Risk Analysis	No (0)	No (0)	No (0)	Yes (4)	No (0)	No (0)
5	User Testing Ability	No (0)	No (0)	Yes (5)	Yes (5)	Yes (5)	Yes (5)
3	Time Consumption	Yes (3)	Yes (3)	No (0)	Yes (3)	No (0)	No (0)
<b>Total</b>	<b>Overall Score</b>	<b>14</b>	<b>27</b>	<b>12</b>	<b>25</b>	<b>5</b>	<b>12</b>

In this project, we developed a comparison matrix to select the most suitable SDLC model, considering key criteria such as well-known requirements, technological knowledge, efficiency, dependability and security, risk analysis, user testing ability, and time consumption. Each criteria was assigned a priority score based on its importance to the project. The comparison matrix helped us analyze which model best supports these priorities. Based on this evaluation, the scores were as follows: Waterfall scored 14, V-Shape scored 27, Iterative scored 12, Spiral scored 25, Agile scored 5, and Prototype scored 12.

Among these models, the V-shaped model is the most suitable for our project. It strongly supports critical priorities such as well-known requirements, technological knowledge, efficiency, dependability, and security.

### 3.3 Feasibility Study

### 3.3.1 Technical Feasibility

The technical feasibility focuses on whether the proposed system can be developed with the current technology and resources available.

- **System Accuracy:** The new system will be created by modern web technologies (e.g., HTML5, CSS3, JavaScript, and a backend like Node.js) to ensure high accuracy in functionalities such as meal pre-ordering and bus seat management. The use of database systems like MongoDB will enhance data accuracy and consistency.
  - **Load ability:** The system can handle an average number of users, clubs, and cafeteria items. We will implement cloud-based infrastructure (e.g., AWS) that allows for dynamic scaling based on demand.
  - **User-Friendly interface:** A user-friendly interface will be employed to ensure easy to use for students and staff.

### 3.3.2 Operational Feasibility

The project is designed to be easy to use and maintain. Students, faculty, and staff will find the system simple to navigate, with features like pre-ordering meals, checking bus seat availability, and managing club activities all integrated into a single platform. No special training is needed to operate the system, as it will have a user-friendly interface. The project is operational for any operating system. It will be responsive for mobile, tab & desktop.

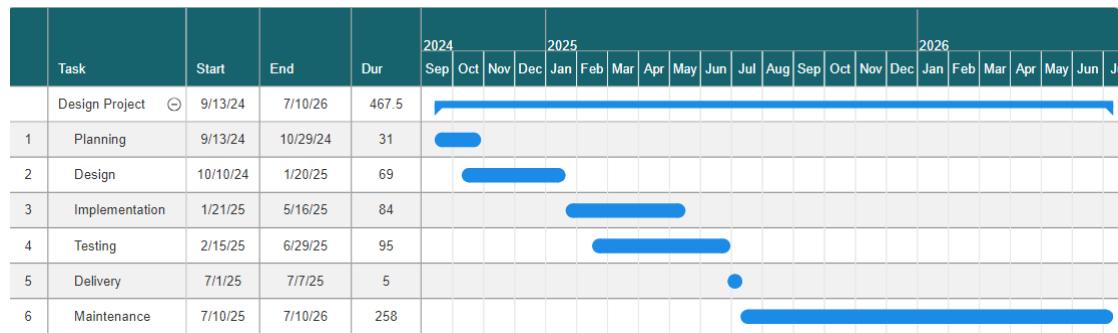


Figure 3.1: Project Development Timing

The operational feasibility of the *Campus360 Solution* evaluates the system's ability to function effectively in real-world scenarios. To ensure smooth development and deployment, a detailed Gantt chart outlines the project timeline from **September 13, 2024**, to **July 10, 2026**, spanning six key phases:

- **Planning (31 days):** This phase focuses on identifying project requirements and goals.
  - **Design (58 days):** Tasks include creating diagrams and defining system architecture.

- **Implementation (84 days):** This phase involves developing key modules, such as cafeteria management, bus seat tracking, and club activity management.
- **Testing (19 days):** Ensures system reliability, accuracy, and error-free functionality.
- **Delivery (5 days):** Deploys the system for use by students, faculty, and administrators.
- **Maintenance (258 days):** Focuses on system updates, technical support, and performance monitoring.

### 3.3.3 Financial Feasibility

The project will have a low starting cost because it is being developed by three students using free tools like VS Code and GitHub. We will use our own computers and internet for the project, so there's no need to buy extra hardware right now. However, for the cafeteria and bus systems, we will need to buy scanners to read student ID cards. These scanners will be the main initial expense. The system is easy to use, so students and faculty won't need any training, which means no extra cost for that. Later, if the system grows or is used for business, we will need to pay for things like domain hosting and servers.

Table 3.2: Budget Details for Campus360 Solution

SL	Criteria	Cost Specification	Estimated Cost (BDT)
1	Office Cost	Team meetings	15,000
		Project meetings	15,000
		First Aid	1,000
2	Website Cost	Website maintenance	2,000
3	Software & Equipment Cost	Computers	4,00,000
		Laptops	1,50,000
		Server setup	3,00,000
		Internet services	50,000
		Scanner	50,000
4	Salary Cost	Team Leader	1,50,000
		System Designer	75,000
		Software Engineer	1,00,000
		Frontend Developer	50,000
		Backend Developer	70,000
		UI/UX designer	60,000
		Database Admin	80,000
		Technical Support	30,000
		Security	10,000
<b>Total Cost</b>			<b>15,98,000</b>

The budget for the Campus360 Solution project is divided into four main categories: Office Costs, Website Costs, Software & Equipment Costs, and Salary Costs. The total estimated budget is 15,98,000 BDT, covering all necessary expenses for project development and maintenance.

- **Office Costs:** Includes team and project meeting expenses along with basic necessities like first aid, totaling 31,000 BDT.
- **Website Costs:** Website maintenance expenses are estimated at 2,000 BDT.
- **Software & Equipment Costs:** Major expenses like computers, laptops, servers, scanners, and internet services amount to 9,50,000 BDT.
- **Salary Costs:** Compensation for the development team, including roles like team leader, designers, developers, and technical support, totals 6,15,000 BDT.

This budget ensures that all resources required for the project, including tools, manpower, and operational costs, are accounted for, making the system feasible to develop and deploy.

## 3.4 Architectural Design

### 3.4.1 High Level Design

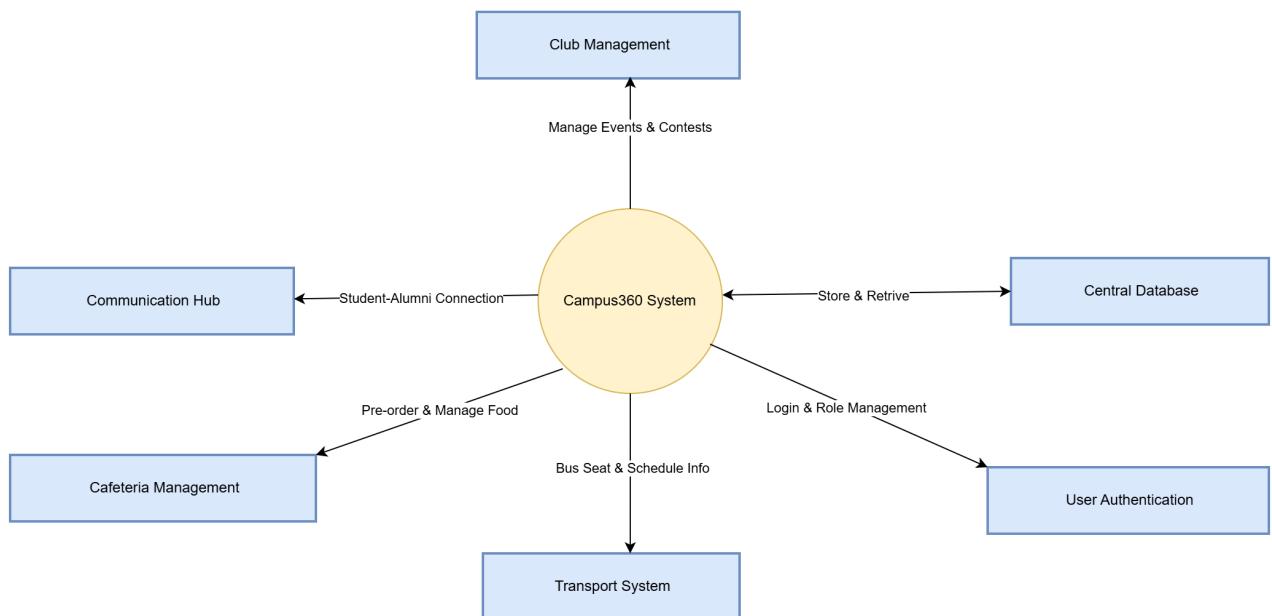


Figure 3.2: High Level Design

### 3.4.2 Low Level Design

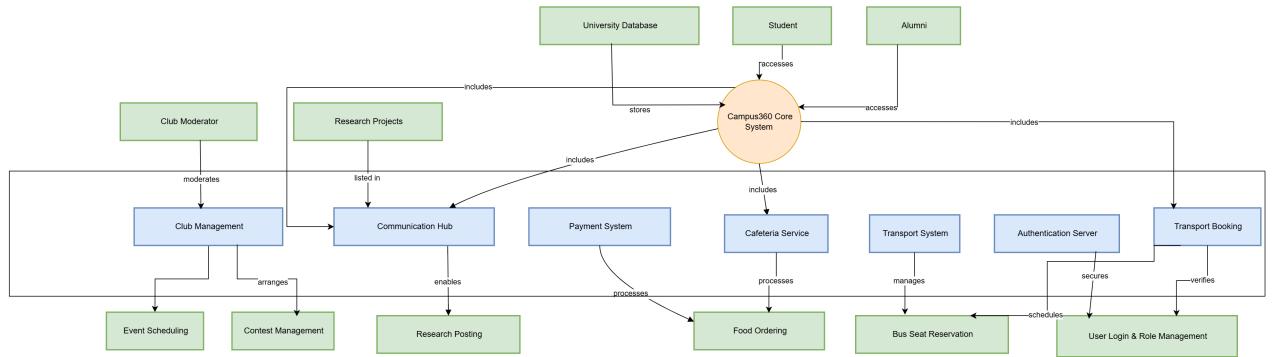


Figure 3.3: Low Level Design

## 3.5 Block Diagram

The block diagram of the Campus360 Solution project begins with the Start point, where the system initiates user verification through the User Authentication process. At this stage, the system checks whether the user belongs to the campus. If the user is verified as a campus member, they proceed as a Regular User. Otherwise, access is denied, and the system redirects the user to the Refuse to Entry block.

As a Regular User, individuals can access various functionalities. The first key feature is Food Pre-Order, where users can pre-order their desired meals. Within this, users can view their Order History to check previous orders or see the Show Waiting Time option, which provides an estimated delivery time for their meals. This ensures a smooth cafeteria experience.

The next feature is Bus Management, which offers transport-related functionalities. Users can Track Bus to monitor real-time bus locations, view Bus Schedule Info for seat availability, and participate in the Transport Survey to provide input on shuttle bus usage. These features collectively optimize the transport system for efficiency and convenience.

Another important module is Manage Club Activities, which focuses on campus clubs and events. Users can Display Upcoming Events to stay updated on club activities and participate in Contests, such as semester-wise competitions. This encourages student engagement and participation in extracurricular activities.

The final feature is the Communication Hub, which acts as a collaboration platform for current students and alumni. Users can Create Profile to showcase their research work, projects, and achievements. Additionally, others can View Profiles, enabling students and alumni to connect for opportunities such as research assistance or employment, benefiting both parties.

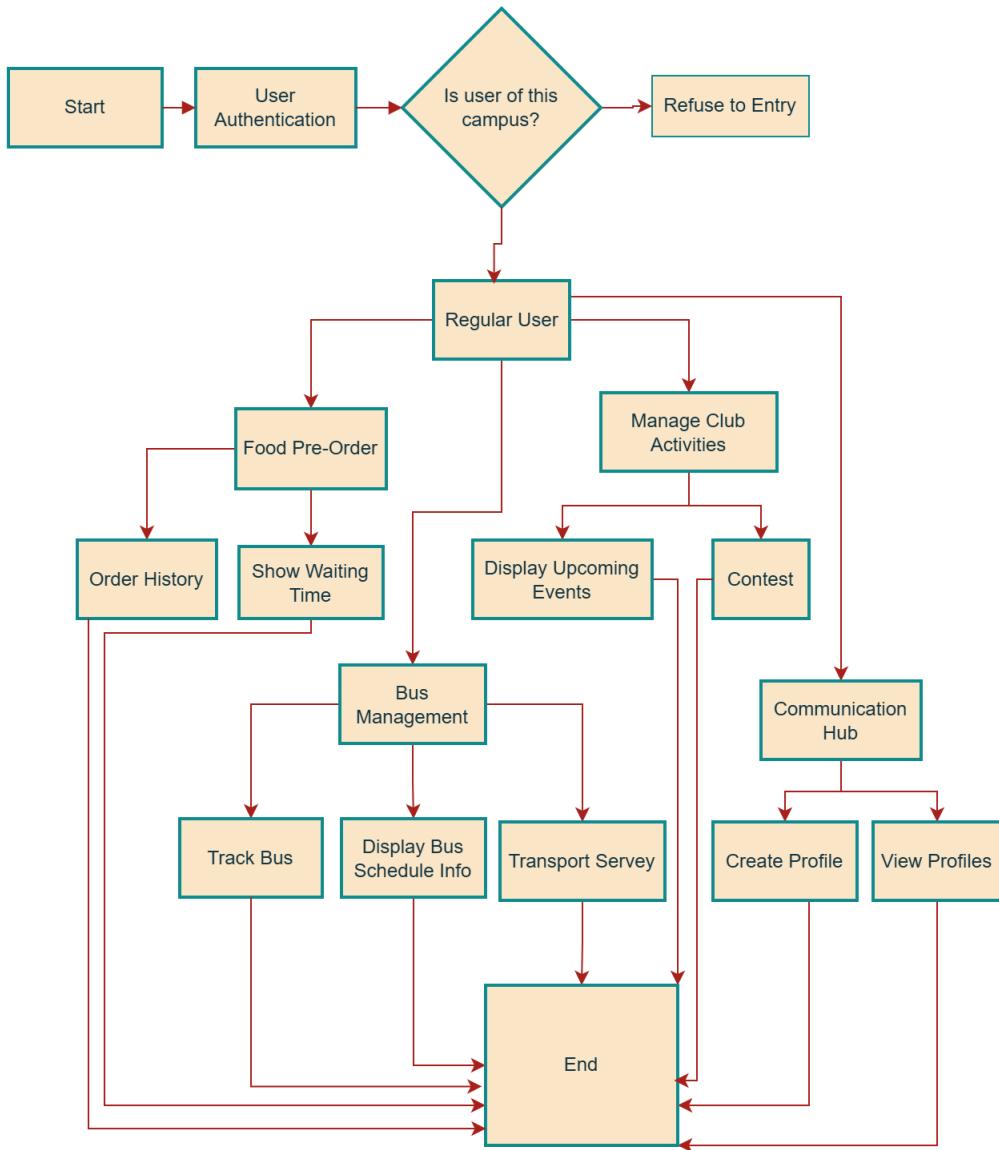


Figure 3.4: Block diagram of the project

## 3.6 DFD

### 3.6.1 Level 0 diagram

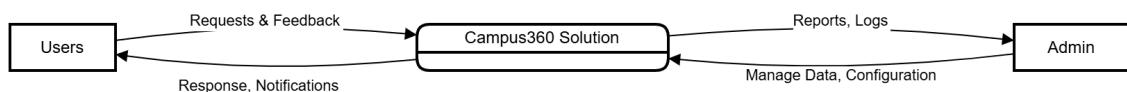


Figure 3.5: Level 0 data flow diagram

The Level 0 Data Flow Diagram (DFD) provides a high-level overview of the Campus360 Solution system. It represents the flow of data between the system and its external entities: Users and Admin. This diagram focuses on the main process and its interactions without breaking it into detailed sub-processes.

The primary external entities and processes are as follows. Users, including students and faculty, send requests such as meal orders, bus queries, or club registrations. They receive responses from the system, including order confirmations, wait times, or event updates. Admins are responsible for updating system data, such as inventory, bus seating, and event details. They also receive reports to monitor system usage and performance. The central process, Campus360 Solution, handles user requests and manages data inputs from admins.

The key data flows within the system include requests sent by users for orders, queries, or registrations, and responses provided by the system, such as order codes, wait times, or confirmations. Admins manage data by updating inventory, seating, and events, while the system generates reports to assist admins in performance analysis and decision-making.

### 3.6.2 Level 1 diagram

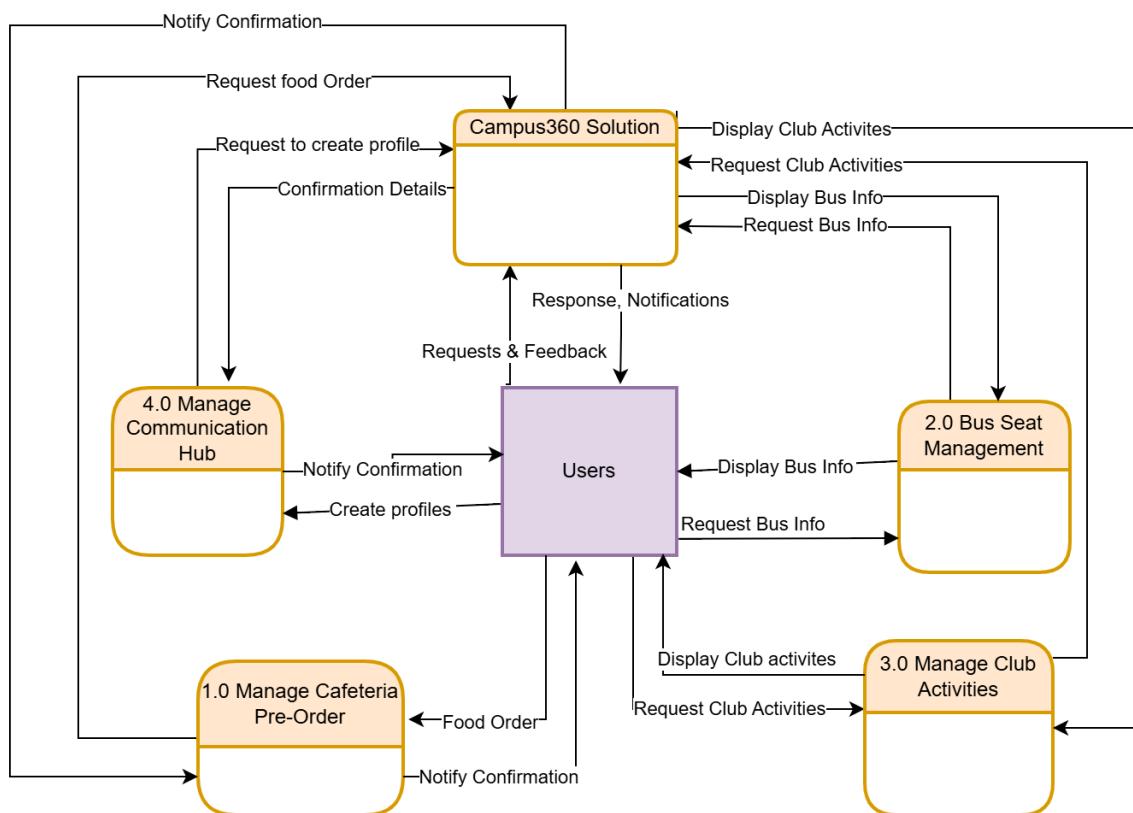


Figure 3.6: Level 1 Data Flow Diagram of Campus360 Solution

## Processes and Their Roles

1. **1.0 Manage Cafeteria Pre-Order:** Handles user requests for food orders and sends confirmation notifications back to users.
2. **2.0 Bus Seat Management:** Processes bus seat-related queries and provides bus seat availability details to users.
3. **4.0 Manage Communication Hub:** Allows users to create profiles and sends profile confirmation notifications.
4. **6.0 Manage Club Activities:** Manages club activity requests and displays information about club events.

## Interactions and Data Flows

- **Users:** - Users interact with the system to request services such as food orders, bus seat queries, and club activity information. - They receive responses like confirmations, bus info, and club activity details.
- **Campus360 Solution:** - Acts as the core process that routes requests to sub-processes and delivers responses back to users.

## Data Flow

The data flows in the Level 1 diagram represent key interactions within the Campus360 Solution. Food Orders and Confirmations are managed by the Cafeteria Pre-Order process, allowing users to place and confirm meal orders efficiently. Bus Info Requests and Updates are handled by the Bus Seat Management process, enabling users to check bus schedules and seat availability. Club Activity Requests and Details are managed through the Club Activities process, where users can access information about events and participate in activities. Lastly, Profile Creation and Notifications are controlled by the Communication Hub, facilitating user profile management and collaboration through timely notifications.

### 3.6.3 Level 2 diagram

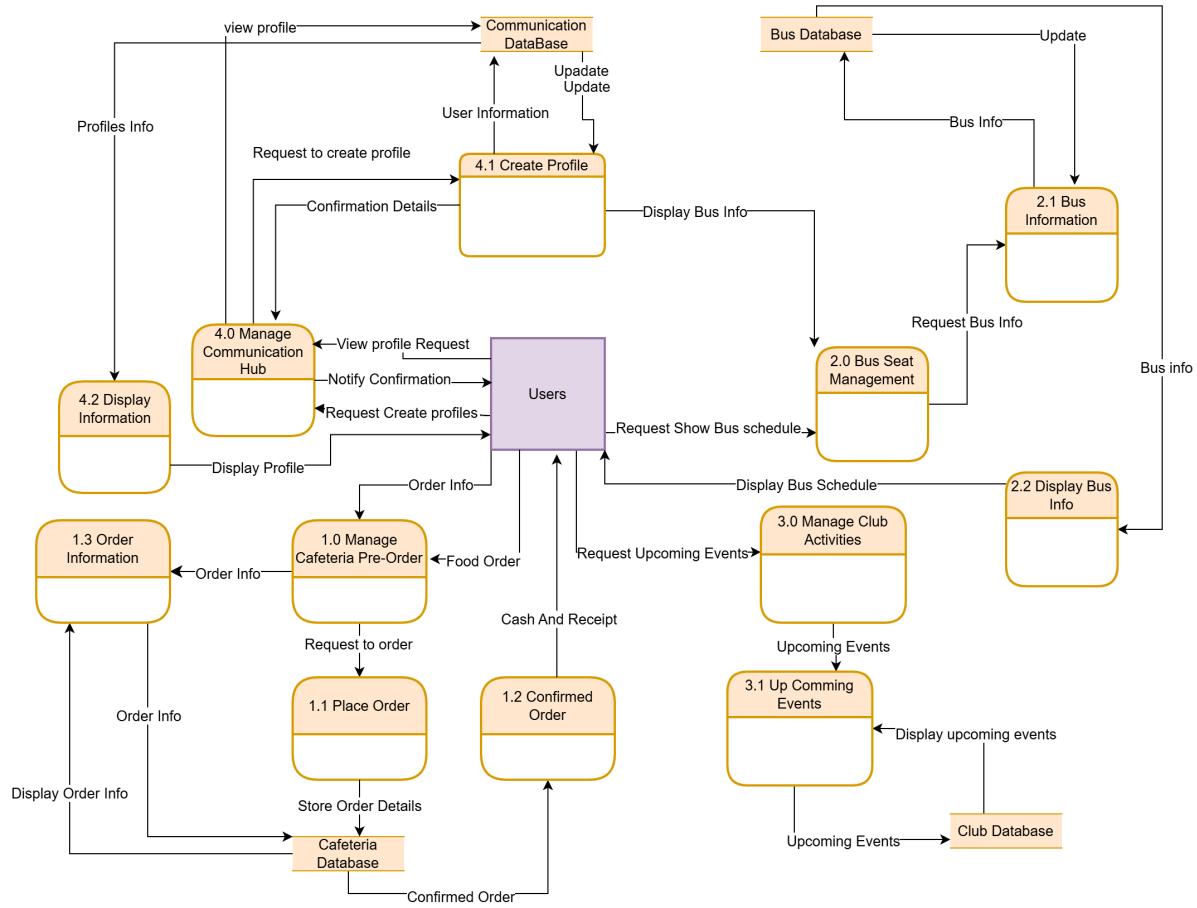


Figure 3.7: Level 2 Data Flow Diagram of Campus360 Solution

The Level 2 DFD for the Campus360 Solution expands the core processes of the system into detailed sub-processes and shows the flow of data between users, processes, and external databases.

The system begins with Users, who interact with various processes. They can request to create a profile, place food orders, view bus schedules, or access event information. These requests are directed to specific modules within the system.

- **Profile Management:** Users send a request to create a profile, which is handled by the **4.1 Create Profile** process. This process interacts with the **Communication DataBase** to update or store user information. Confirmation details and profile updates are sent back to the **4.0 Manage Communication Hub**, which then notifies the user and displays the updated profile through **4.2 Display Information**.
- **Cafeteria Pre-Order System:** The food order process begins with the **1.0 Manage Cafeteria Pre-Order** module. Users send order requests, which are handled by **1.1 Place Order**. This process stores the order details in the **Cafeteria Database** and sends order confirmations to **1.2 Confirmed Order**. Users can view the status of their orders through **1.3 Order Information**, which displays updated order details.

- **Bus Seat Management:** For bus information, users request bus schedules, which are processed by 2.0 Bus Seat Management. This module interacts with the Bus Database to retrieve or update bus information. The information is displayed to users via 2.1 Bus Information and 2.2 Display Bus Info, allowing users to access the bus schedule or other relevant details.
- **Club Activities Management:** Users can request details about upcoming events through 3.0 Manage Club Activities. This module retrieves event details from the Club Database. The process 3.1 Up Coming Events then displays the event details to users.

### 3.7 UML Use Case Diagram

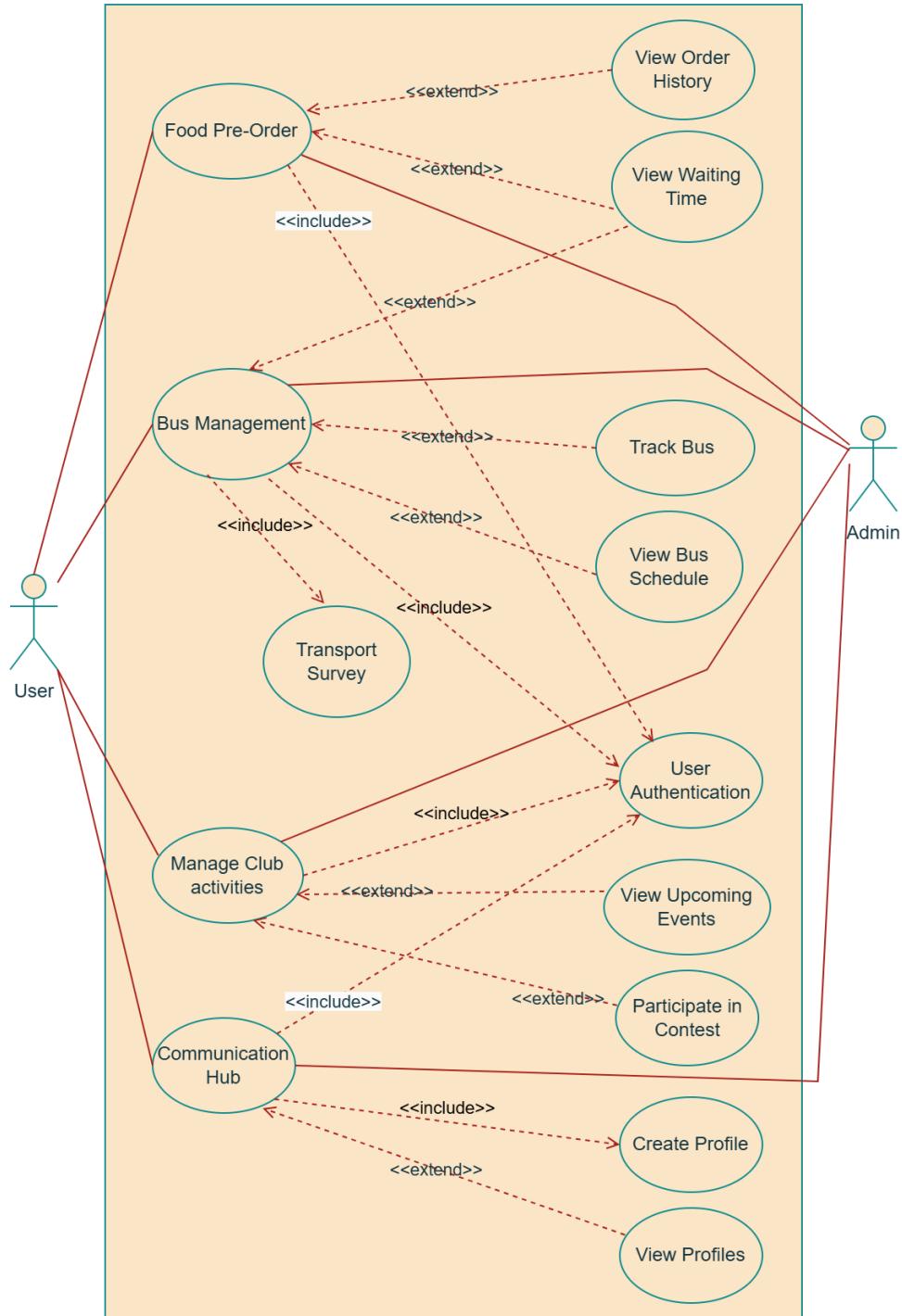


Figure 3.8: Use Case diagram of Campus360 Solution

The Use Case Diagram represents the Campus360 Solution, showcasing its multiple features designed to enhance the campus experience for students, faculty, and administrators. It highlights how different users interact with the system to perform various tasks, ensuring a streamlined approach to managing campus activities.

The system identifies two main actors:

- **User:** Represents students and faculty members who interact with the system for tasks like meal pre-ordering, bus management, club activities, and communication.
- **Admin:** Represents the system administrator responsible for managing tasks such as updating event details, overseeing transport surveys, and maintaining system functionality.

The main use cases include the following:

- **Food Pre-Order:** Users can pre-order meals, check waiting times, and view their order history.
- **Bus Management:** Users can track bus schedules and view seat availability, while administrators use transport surveys to determine bus requirements for shuttle services.
- **Manage Club Activities:** Users can explore upcoming events and participate in contests organized by university clubs. Admins update club activities and manage event details.
- **Communication Hub:** This feature allows users to create profiles showcasing their projects and research activities. They can also view other profiles to foster collaboration and networking opportunities.
- **User Authentication:** Ensures only authorized users can access the system, providing a secure environment for interactions.

The diagram also incorporates relationships between use cases:

The diagram also highlights relationships between use cases to depict task dependencies. The include («include») relationship represents mandatory tasks that must occur as part of a use case. For example, the "Food Pre-Order" use case includes "User Authentication" to ensure secure access for users. On the other hand, the extend («extend») relationship shows optional tasks that depend on specific conditions. For instance, viewing "Order History" extends the "Food Pre-Order" use case, allowing users to access additional information if needed.

## 3.8 UML Sequence and Communication Diagram

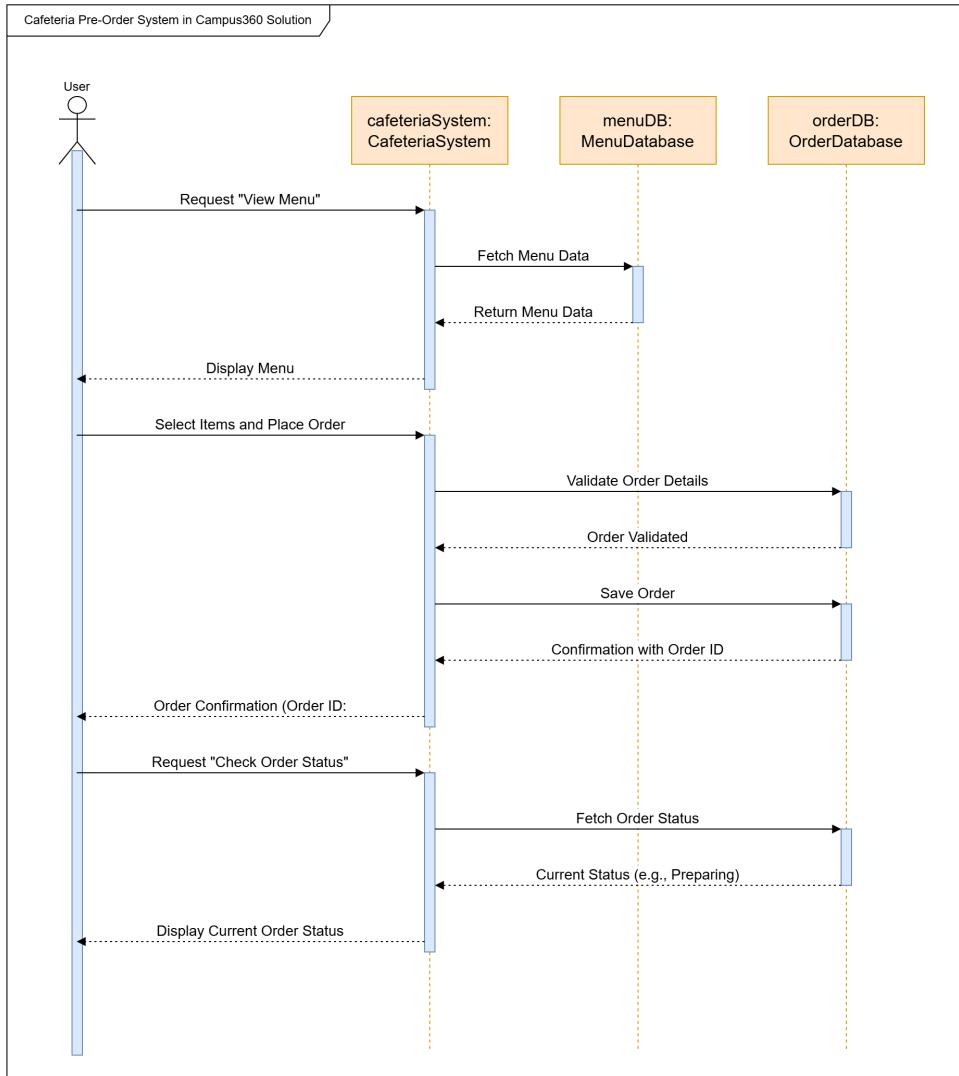


Figure 3.9: Sequence Diagram for Cafeteria Pre-Order System

The sequence diagram represents the interaction flow of the Cafeteria Pre-Order System in the Campus360 Solution. It illustrates how a user interacts with the system to view the menu, place an order, and check the order status. The primary components involved are:

- **User:** The actor initiating the interactions.
- **Cafeteria System:** The main system handling user requests and coordinating with databases.
- **Menu Database:** Stores and provides the menu data.
- **Order Database:** Manages order validation, saving, and status updates.

The key interactions in the system are as follows:

**1. View Menu:** The user requests to view the menu. The Cafeteria System fetches the menu data from the Menu Database and displays it to the user.

**2. Place an Order:** The user selects menu items and places an order. The Cafeteria System validates the order details and saves the order into the Order Database. A confirmation, including an Order ID, is sent back to the user.

**3. Check Order Status:** The user requests to check the status of their order. The Cafeteria System retrieves the order status from the *Order Database* and displays the current order status (e.g., “Preparing”) to the user.

### 3.9 UML Class Diagram

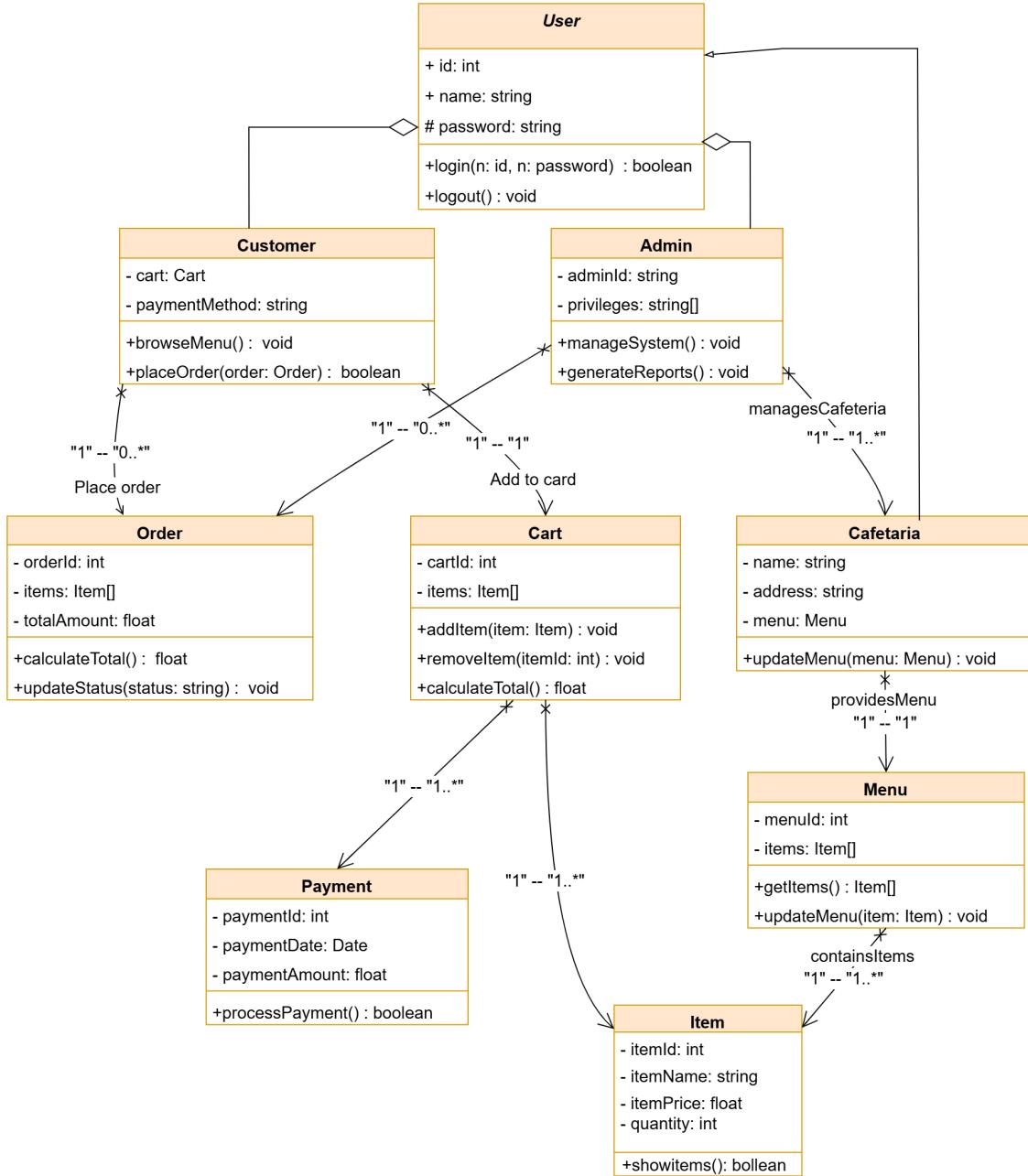


Figure 3.10: Class Diagram of the Cafeteria Management System

#### Class Diagram Overview of the Cafeteria Management System

- The **User** class represents the main entity in the system, acting as a parent class for both **Customer** and **Admin**. It includes attributes such as **id** for a unique identifier, **name** for the user's name, and **password** for login purposes. The **User** class has two main functions: **login** to allow users to access the system and **logout** to exit.
- The **Customer** class inherits from the **User** class and represents the users who can place

orders. It has additional attributes like cart to store selected items temporarily and paymentMethod for choosing how to pay. The Customer class includes functions like browseMenu, which allows the user to view the available food or drink options, and placeOrder, which submits their order.

- The Admin class also inherits from User and is responsible for managing the system. It includes attributes like adminId, a unique identifier for admins, and privileges, which lists the tasks they are allowed to perform. The Admin can manageSystem to control operations and generateReports to analyze system activities.
- The Order class stores details about a customer's order. It has attributes like orderId for the order's unique identifier, items to list the ordered products, and totalAmount for the total price. The class provides the calculateTotal function to determine the order's cost and updateStatus to change the status (e.g., pending, completed).
- The Cart class temporarily holds the items a customer selects before placing an order. It includes attributes such as cartId, a unique cart identifier, and items, which stores the products added to the cart. It has functions like addItem to include an item, removeItem to take items out, and calculateTotal to display the cart's total value.
- The Cafeteria class represents the cafeteria in the system. It includes attributes like name for the cafeteria's name, address for its location, and menu to store the available food and drink options. The function updateMenu allows changes to the cafeteria's offerings.
- The Menu class contains the list of items provided by the cafeteria. It has attributes such as menuId to uniquely identify the menu and items for the list of products. Functions like getItems display the current offerings, and updateMenu allows modifying the list of items.
- The Item class represents individual food or drink products. Its attributes include itemId for a unique identifier, itemName for the product's name, itemPrice for its cost, and quantity for available stock. The function showItems displays details like the product's name and price.
- The Payment class manages all payment-related tasks. It includes attributes like paymentId for identifying payments, paymentDate for recording the date, and paymentAmount for the paid total. The processPayment function finalizes the payment process.

**The relationships among the classes are as follows:** The User class is the parent of both the Customer and Admin classes. A Customer places an Order, which can be paid for using the Payment class. Customers use a Cart to temporarily hold multiple Items before placing an order. The Cafeteria class is associated with a Menu, which lists the available Items. Finally, the Order connects to the Payment class to manage payments.

## 3.10 Final Output

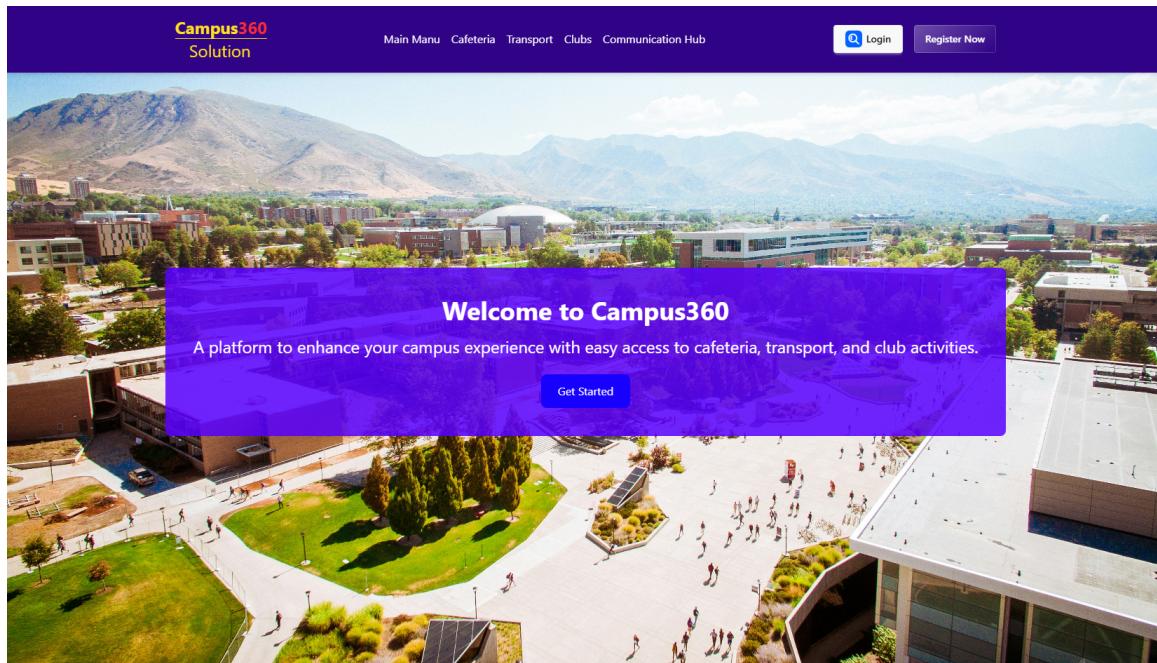


Figure 3.11: Main manu

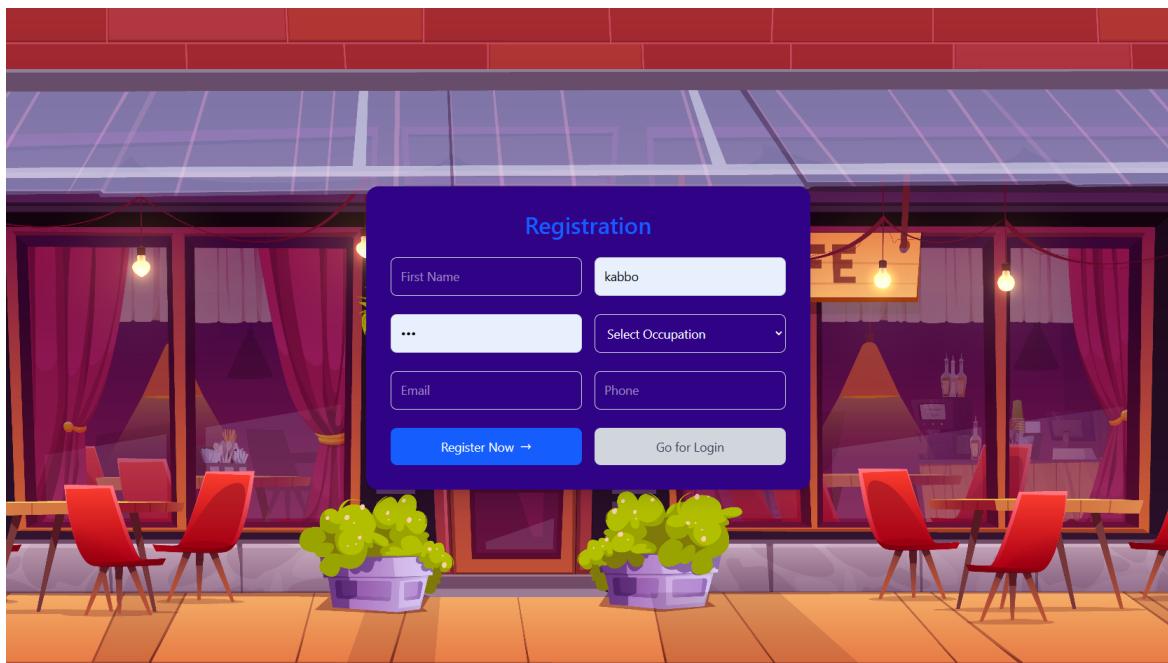


Figure 3.12: Registration Interface

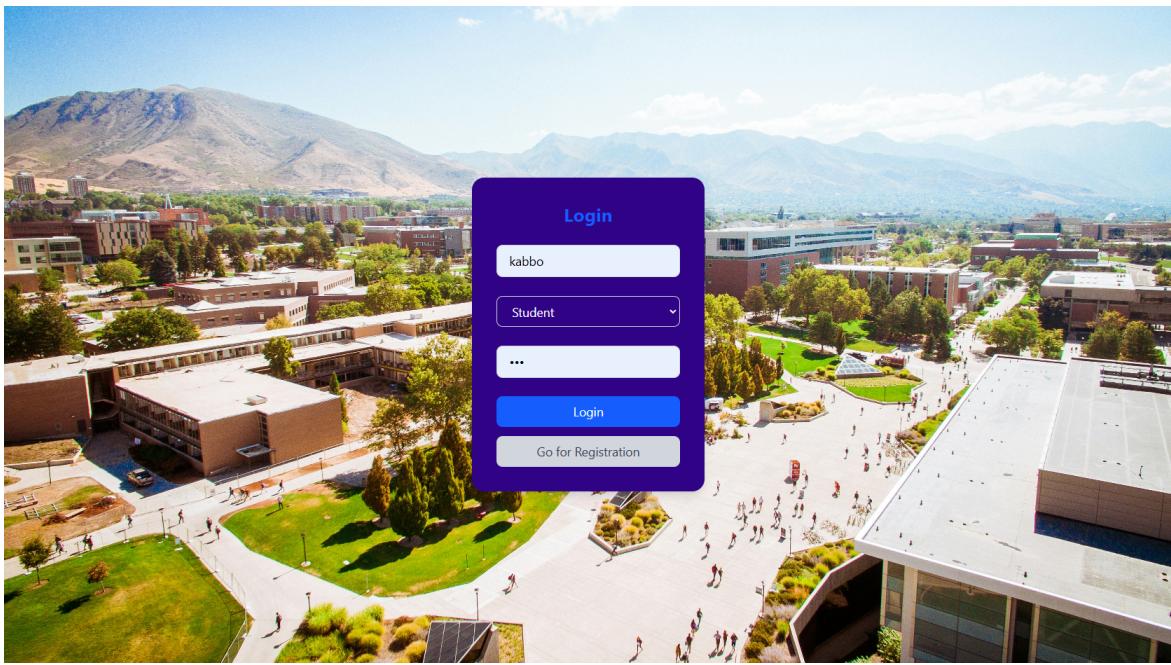


Figure 3.13: Login Interface

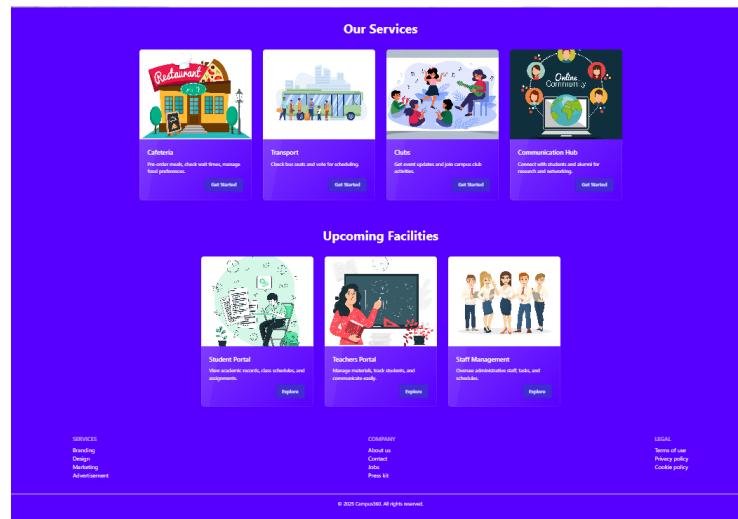


Figure 3.14: All Features

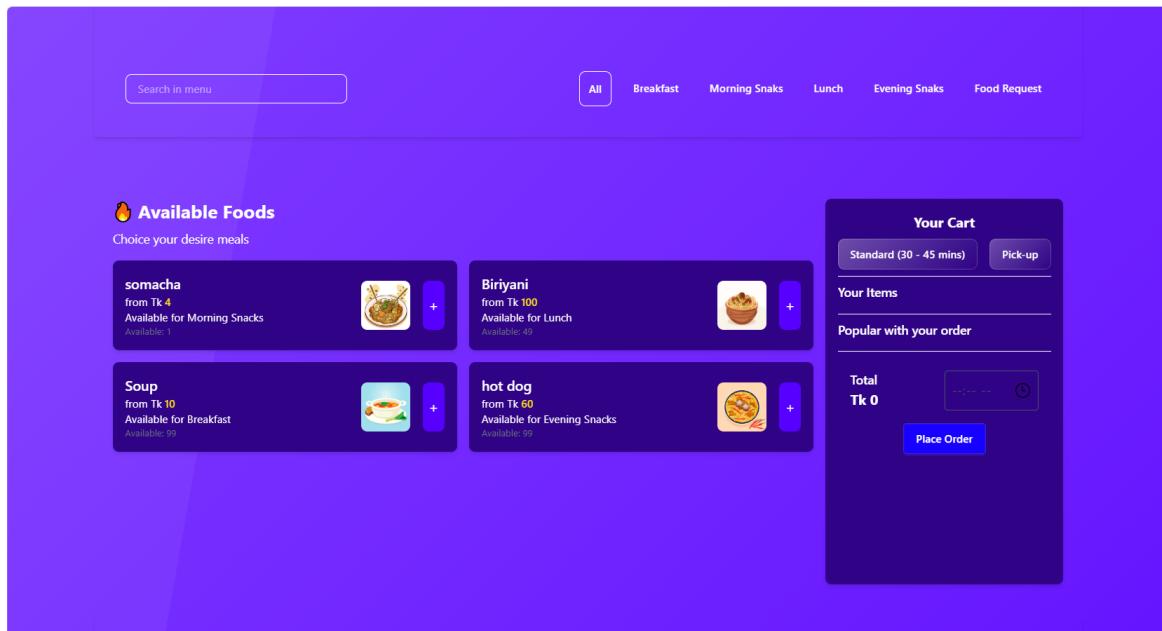


Figure 3.15: Searching for meal and placing order.

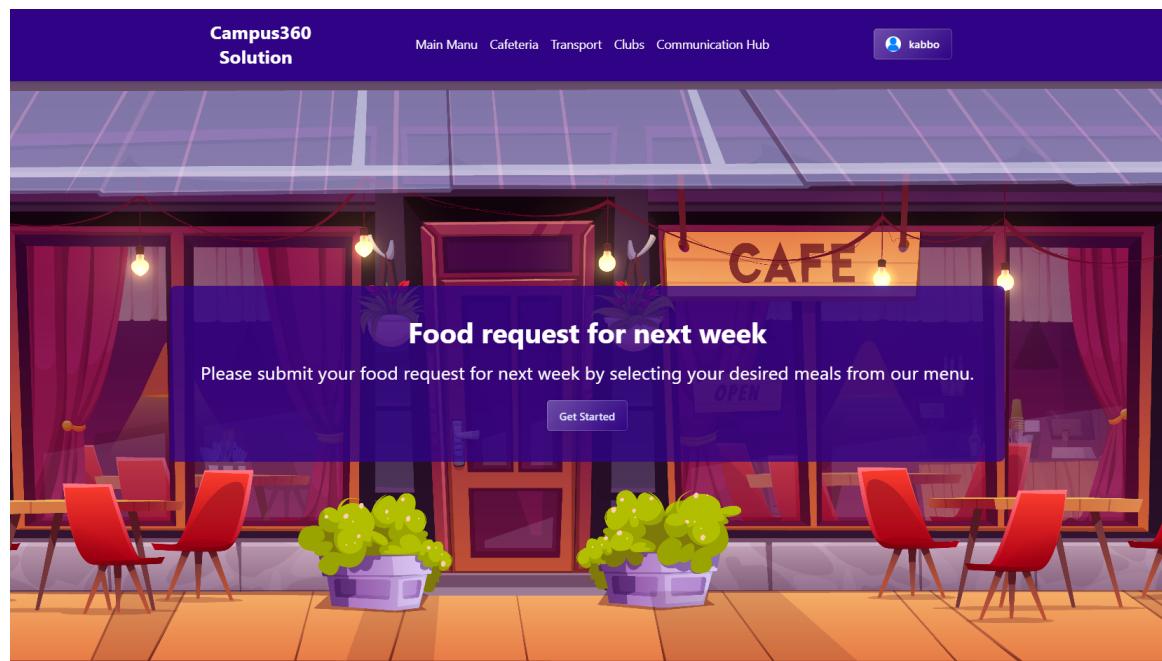


Figure 3.16: Meal Order

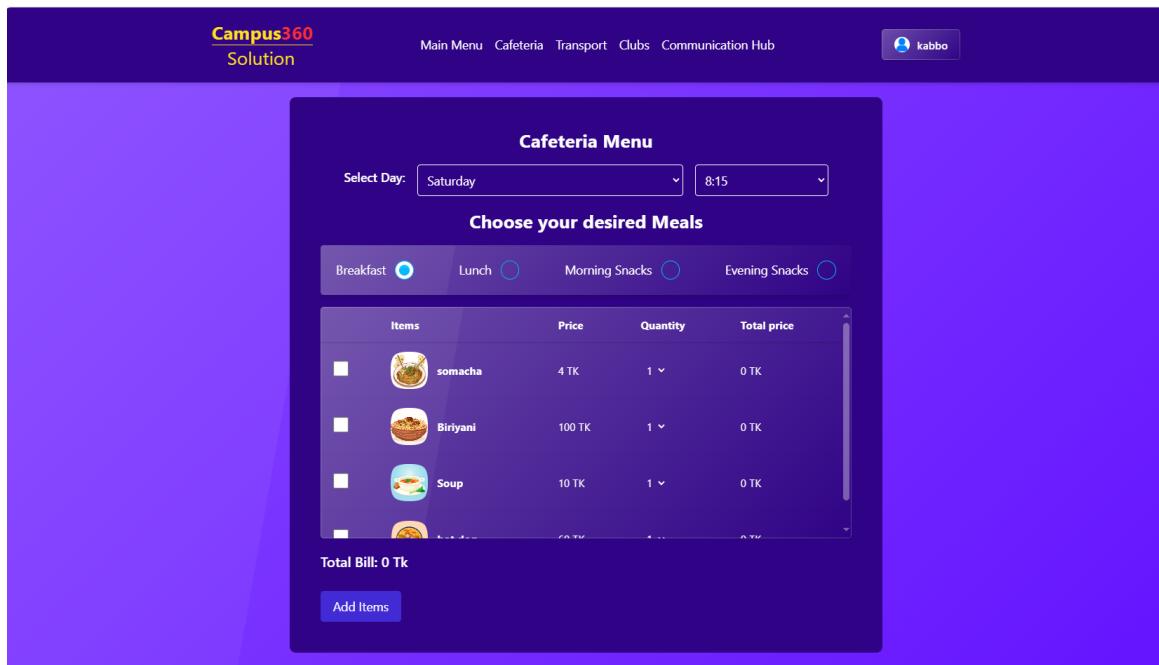


Figure 3.17: Food Request

Awatied Peoples						
	Id	Items	Pickup time	Waiting Time	Status	Need to Pay
1	22100224	somacha	05:20	0 min	Confirmed	4/- <a href="#">Edit/expire</a>
2	2210022	hot dog somacha	06:41	0 min	Pending	9/- <a href="#">Edit/expire</a>
3	123456	hot dog somacha hot dog	12:38	0 min	Confirmed	85/- <a href="#">Edit/expire</a>
4	123	Biriyani hot dog	23:35	27 min	Pending	160/- <a href="#">Edit/expire</a>
5	456	somacha	12:40	0 min	Pending	14/- <a href="#">Edit/expire</a>

Figure 3.18: Requested Meal

Weekly Meal Plan						
Day	Breakfast	Lunch	Morning Snacks	Evening Snacks	Daily Bill	Status
Saturday	• somacha x 1 • fd x 1	• hot dog x 1	• somacha x 1 • Biriyani x 1	• hot dog x 1	152 TK	Paid
Pickup Time	8:15	1:30 pm	1:30 pm	1:30 pm		
Sunday	• hot dog x 1	-	-	-	5 TK	paid
Pickup Time	8:15	-	-	-		
Monday	• hot dog x 1	• hot dog x 1	-	• fd x 1 • hot lcl x 1	120 TK	paid
Pickup Time	8:15	1:30 pm	-	1:30 pm		
Tuesday	• hot dog x 1 • somacha x 1	• hot dog x 1 • somacha x 1	• somacha x 1 • Biriyani x 1	-	122 TK	Unpaid
Pickup Time	8:15	-	1:30 pm	-		
Wednesday	-	-	-	-	0 TK	Unpaid
Pickup Time	-	-	-	-		
Thursday	• hot dog x 1	-	-	-	5 TK	Unpaid
Pickup Time	8:15	-	-	-		
Friday	• fd x 1 • hot lcl x 1	-	-	-	110 TK	paid
Pickup Time	8:00	-	-	-		

Figure 3.19: Weekly Meal Plan

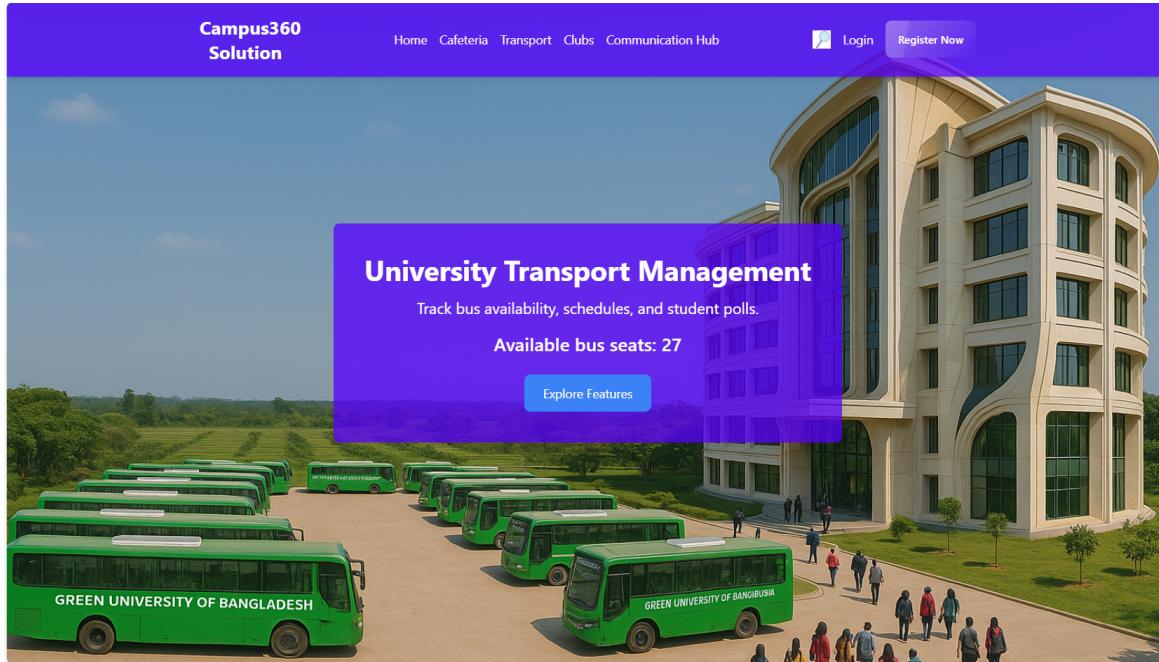


Figure 3.20: Transport user interface

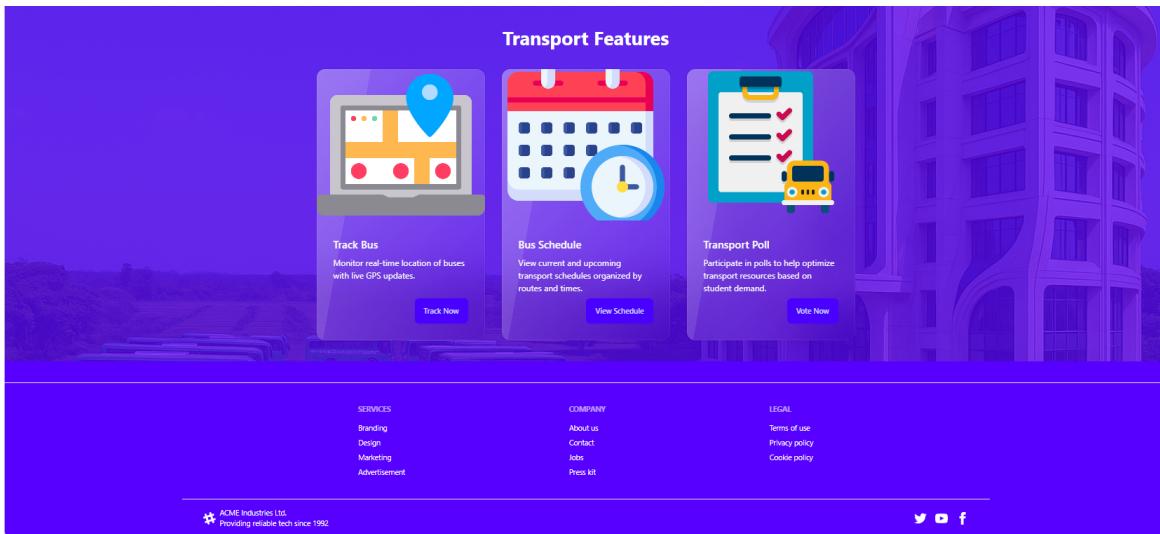
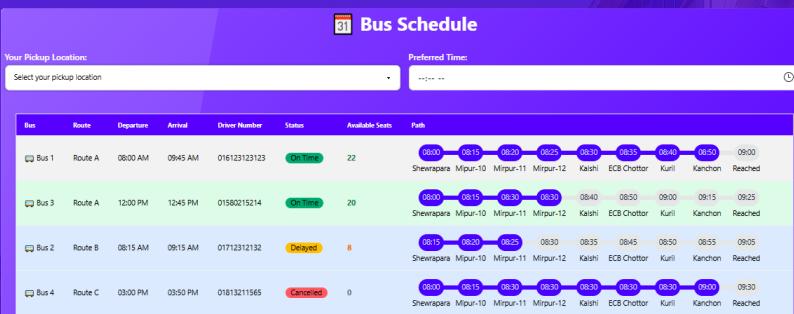


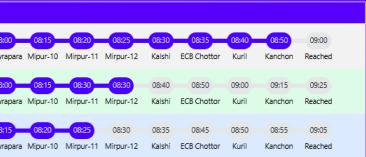
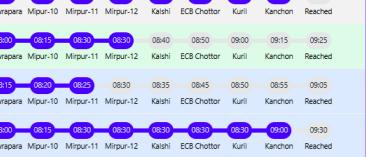
Figure 3.21: Transport Features



**Bus Schedule**

Your Pickup Location:

Preferred Time:

Bus	Route	Departure	Arrival	Driver Number	Status	Available Seats	Route
Bus 1	Route A	08:00 AM	0845 AM	016123123123	On Time	22	
Bus 3	Route A	12:00 PM	1245 PM	01580215214	On Time	20	
Bus 2	Route B	08:15 AM	0915 AM	01712312132	Delayed	8	
Bus 4	Route C	03:00 PM	0350 PM	01813211565	Cancelled	0	

[Close](#)

**SERVICES**

- [Branding](#)
- [Design](#)
- [Marketing](#)
- [Advertisement](#)

**COMPANY**

- [About us](#)
- [Contact](#)
- [Jobs](#)
- [Press kit](#)

**LEGAL**

- [Terms of use](#)
- [Privacy policy](#)
- [Cookie policy](#)

ACME Industries Ltd.  
Providing reliable tech since 1992

[Twitter](#) [YouTube](#) [Facebook](#)

Figure 3.22: Bus Schedule

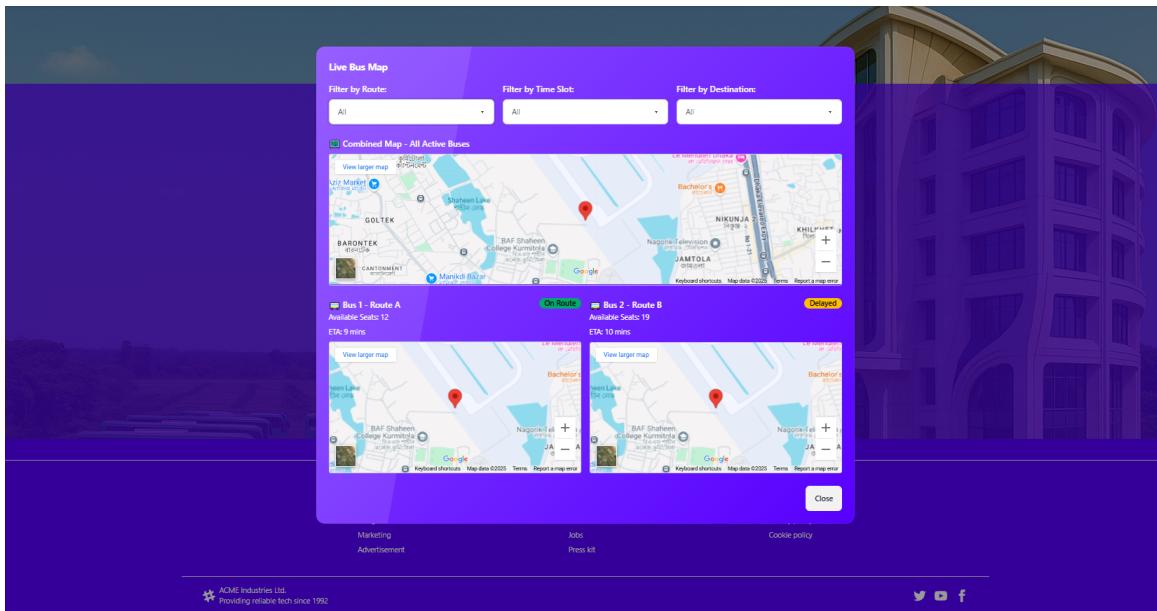


Figure 3.23: Live Bus Map

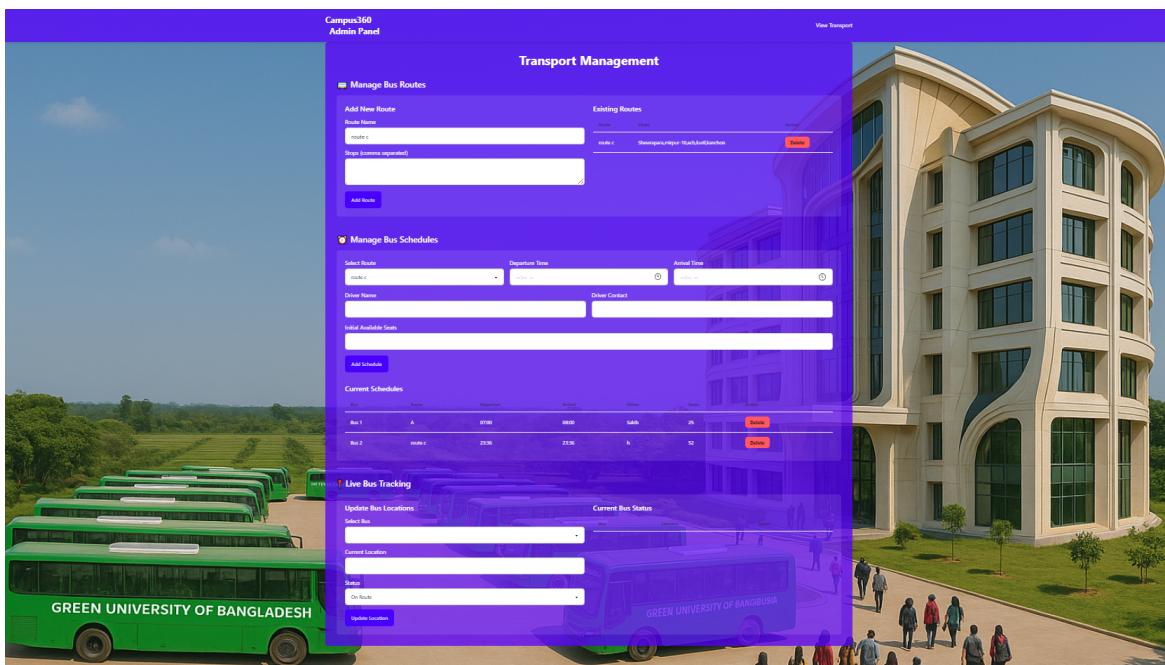


Figure 3.24: Transport Management

# Chapter 4

## Software Testing Report

Criteria	Action	Input (Test Case)	Expected Output	Actual Output	Test Result
Login	User logs in with correct credentials	Valid email and password	Login successful, home screen opens	Login successful, home screen opens	Pass
	User logs in with wrong password	Valid email, wrong password	Login fails with error message	Login fails with error message	Pass
	User logs in with wrong email	Wrong email, valid password	Login fails with error message	Login fails with error message	Pass
	Email field is left empty	No email entered	Login button disabled or error shown	Login button disabled or error shown	Pass
	Password field is left empty	No password entered	Login button disabled or error shown	Login button disabled or error shown	Pass
	Both fields are empty	No input	Login button disabled	Login button disabled	Pass
	"Show password" button works	Tap "Show password" icon	Password becomes visible	Password becomes visible	Pass
	Error message shown on login fail	Invalid credentials	Error message appears	Error message appears	Pass
	Successful login redirects to home	Correct credentials entered	Home screen opens	Home screen opens	Pass
	App remembers session	Logged in once and app restarted	User stays logged in	User stays logged in	Pass
	User logs out from app	Tap logout button	Session ends and login screen appears	Session ends and login screen appears	Pass

	Unregistered user tries to login	Random email	Login fails with error	Login fails with error	Pass
	Login button disabled until fields filled	One or both fields empty	Button stays inactive	Button stays active	Fail
	No internet during login	Disable internet and try login	Error or "No internet" message	Error or "No internet" message	Pass
	Login input allows special characters	Special characters input	Input accepted or handled	Input accepted and handled	Pass
Registration	Submit valid credentials	Valid ID, password	Account created successfully	Account created successfully	Pass
	Submit duplicate ID	ID already used	Registration fails	Registration fails	Pass
	Submit blank form	No input	Show validation error	Show validation error	Pass
	Submit only email	Email only	Show password missing error	Show password missing error	Pass
	Password and confirm mismatch	Two different passwords	Show mismatch warning	Show mismatch warning	Pass
	Use special characters in ID	ID = !@\$	Show input invalid error	Show input invalid error	Pass
	Submit weak password	"123"	Prompt for strong password	Prompt for strong password	Pass
	Registration success redirects to login	Valid details	Redirect to login page	Redirect to login page	Pass
	Existing ID blocks re-registration	Existing student ID	Prevent registration	Prevent registration	Pass
	Click "Register" with empty password	Empty password field	Disabled button or error	Disabled button or error	Pass
	Input only whitespace	Whitespace only	Show validation error	Show validation error	Pass
	Reload form before submit	Reload during input	Reset form fields	Reset form fields	Pass
	Fields accept Unicode chars	Unicode in name field	Accept/encode properly	Accept/encode properly	Pass
	Confirm password autofill works	Autofill by browser	Autofill successful	Autofill successful	Pass
	Form handles slow network	Slow connection	Process smoothly	Process smoothly	Pass

Cafeteria	View available meals	Open meal list page	Show breakfast and lunch items	Meals displayed correctly	Pass
	Place an order successfully	Valid meal, valid time	Order confirmed	Order confirmed	Pass
	Place order without selecting item	Empty selection	Show error	Show error	Pass
	Search a meal by name	Enter "Biriyani"	List matches with "Biriyani"	Matched list shown	Pass
	Show waiting time after order	Submit order	Waiting time displayed	Waiting time displayed	Pass
	Invalid time for pre-order	Time past deadline	Show time error	Show time error	Pass
	Submit weekly food request	Select multiple days/time	Request accepted	Request accepted	Pass
	Reorder previously ordered item	Click reorder	Auto-fill old selection	Auto-fill old selection	Pass
	Place multiple orders	Select 2 items	Place success message	Place success message	Pass
	View confirmed orders	Open "My Orders" page	See status of all requests	Status shown correctly	Pass
	View rejected orders	Open order history	See rejected tag	Rejected orders tagged	Pass
	Order item with quantity limit	0 quantity or invalid	Block action with error	Block action with error	Pass
	Search unavailable item	Enter rare item	Show "not available"	Show "not available"	Pass
	Order with expired session	Place order after timeout	Redirect to login	Redirected to login	Pass
	Network fails during order	Network off	Show "no internet"	Show "no internet"	Pass

Transport System	Submit weekly vote	Day + Time + ID	Poll accepted	Poll accepted	Pass
	Submit vote without login	Time only	Block with login message	Block with login message	Pass
	Submit duplicate vote	Same ID + Time	Show already submitted	Show already submitted	Pass
	Vote with invalid ID	Wrong student ID	Show invalid ID error	Show invalid ID error	Pass
	Poll interface loads	Open transport page	Time slot options visible	Time slot options visible	Pass
	Vote on weekend	Select invalid day	Show "poll closed"	Show "poll closed"	Pass
	Access poll without network	Network off	Show connection error	Show connection error	Pass
	Submit poll with empty fields	No time or day selected	Show required field error	Show required field error	Pass
	Show poll summary (admin)	Login as admin	Show poll chart	Show poll chart	Pass
	Access admin panel without login	No session token	Redirect to login page	Redirect to login page	Pass
	Admin sees student counts	Open dashboard	Show total votes per slot	Show total votes per slot	Pass
	Vote with invalid format	Invalid characters	Block submission	Block submission	Pass
	Retry vote after fail	Network back on	Vote submitted	Vote submitted	Pass
	Submit for unknown stoppage	Stop not listed	Show stop not found	Show stop not found	Pass
	Poll records update in DB	Vote submitted	DB stores user choice	DB updated	Pass

Admin Panel	Admin login success	Valid username and pass	Redirect to dashboard	Redirect to dashboard	Pass
	Admin login fail	Wrong credentials	Show error message	Show error message	Pass
	Add new food item	Valid inputs	Item added successfully	Item added successfully	Pass
	Add food with missing field	Missing name/price	Show validation error	Show validation error	Pass
	Edit food item	Change existing food	Updated successfully	Updated successfully	Pass
	Delete food item	Click delete	Item removed from list	Item removed from list	Pass
	View food requests	Click requests tab	List of requests displayed	List displayed	Pass
	Confirm meal order	Click confirm	Status = Confirmed	Status = Confirmed	Pass
	Reject meal order	Click reject	Status = Rejected	Status = Rejected	Pass
	Edit after confirmation	Click edit	Allow change with note	Change saved with note	Pass
	View daily sell report	Open report tab	Show today's total sell	Sell displayed	Pass
	View filtered orders	Apply filter/search	Show matched items	Matched data shown	Pass
	Access admin page without login	No credentials	Redirect to login	Redirect to login	Pass
	Add invalid price input	Non-numeric value	Show input error	Show input error	Pass
Club Management	Add duplicate food name	Existing food name	Show duplicate error	Show duplicate error	Pass
	Create new event	Event name, date	Event added successfully	Event added successfully	Pass
	Create event with missing field	No date entered	Show validation error	Show validation error	Pass
	Edit existing event	Update title	Event updated	Event updated	Pass
	Delete event	Click delete button	Event removed from list	Event removed from list	Pass
	Unauthorized event edit	No login token	Redirect to login page	Redirect to login page	Pass
	View upcoming events	Open club page	Upcoming list visible	Upcoming list visible	Pass
User	Event filter by date	Select date range	Filtered list shown	Filtered list shown	Pass

	Create event with duplicate name	Existing title	Show duplicate error	Show duplicate error	Pass
	Event contest winner display	Top user selected	Name shown on homepage	Name shown	Pass
	Invalid event date	Past date selected	Show validation error	Show validation error	Pass
	Refresh event list	Click refresh	Event list reloaded	Event list reloaded	Pass
	Add long description	1000+ characters	Saved correctly	Saved correctly	Pass
	Cross-browser compatibility	Open in Firefox	Page loads properly	Page loads properly	Pass
	Pagination for events	10+ events added	Paginated list shown	Paginated list shown	Pass
	Highlight current contest	Contest ongoing	Banner shown on top	Banner shown	Pass
Communication Hub	Create profile	Valid ID and data	Profile created	Profile created	Pass
	Duplicate profile submission	Same ID used again	Show "profile exists"	Show "profile exists"	Pass
	View shared research	Open hub page	List of posts shown	Posts visible	Pass
	Upload research	Valid file and login	Upload successful	Upload successful	Pass
	Upload without login	Click upload without login	Redirect to login	Redirect to login	Pass
	View user profile	Click username	Profile details visible	Profile shown	Pass
	Upload unsupported format	Upload .exe file	Show format error	Show format error	Pass
	Create profile without ID	Empty ID field	Show required field	Show required field	Pass
	Show research sorted	Sort by newest	Sorted list shown	Sorted list shown	Pass
	Search post by keyword	Enter keyword	Filtered results shown	Filtered results shown	Pass
	Multiple uploads allowed	Upload 3 posts	All visible in list	All listed properly	Pass
	Data security check	Inspect element disabled	Code not exposed	Code hidden	Pass
	Upload large file	Upload 5MB+ PDF	Upload successful	Upload successful	Pass
	Admin view uploads	Login as admin	All profiles visible	Admin list loaded	Pass
	Post without title	Empty title field	Show title required	Show title required	Pass

User Profile	Create new profile	ID, name, password	Profile saved to DB	Profile created	Pass
	Duplicate ID check	Existing student ID	Show "already exists"	Error shown correctly	Pass
	Edit profile info	Update email	Changes saved	Info updated	Pass
	View user profile	Click on user image	Details displayed	Profile loaded correctly	Pass
	Profile not found	Random ID entered	Show "not found" error	Error displayed	Pass
	Login with profile	Correct ID	Redirect to dashboard	Login successful	Pass
	Upload profile image	Upload .jpg file	Image uploaded	Image shown on profile	Pass
	Missing required field	No name provided	Show "required field"	Warning displayed	Pass
	Change password	Provide old/new pw	Password updated	Updated successfully	Pass
	View all profiles (admin)	Admin login	All profiles listed	Displayed as expected	Pass
	Delete profile (admin)	Confirm delete	Profile removed from DB	Deleted successfully	Pass
	Unauthorized edit attempt	Not logged in	Redirect to login page	Access denied	Pass
	Input invalid email	Enter "abc"	Show email error	Validation error shown	Pass
	Use weak password	"123" as password	Warning displayed	Error shown	Pass
	View my activity log	Click log history	Recent actions listed	Activity log visible	Pass
Transport Polling	Submit poll response	Select time and stoppage	Poll data recorded	Poll saved successfully	Pass
	Submit poll without login	No login info	Show login error	Error message shown	Pass
	Poll form validation	Empty time selection	Show warning	Validation works	Pass
	View poll results (admin)	Admin login	Results displayed	Poll data shown	Pass
	Duplicate poll check	Same user polls again	Show "already voted" message	Error message shown	Pass
	Poll options load	Open poll form	All time slots visible	Time options loaded	Pass
	Invalid input poll	Submit blank stoppage	Show warning	Validation shown	Pass
	View poll stats	Click "View Poll"	Stats chart appears	Displayed properly	Pass
	Poll summary display	Weekly summary request	Chart/summary shown	Summary shown correctly	Pass

Student ID validation	Fake ID used	"Invalid ID" error	Error shown	Pass
Logout after poll	Submit then logout	Return to login screen	Works as expected	Pass
Refresh poll result	Press refresh	New results appear	Updated result shown	Pass
Poll deadline expiry	Submit late poll	Show deadline message	Deadline warning shown	Pass
Backend store data	Submit poll form	Poll data stored in DB	DB updated correctly	Pass
Display user choice	View after submit	Selected option highlighted	Displayed correctly	Pass

## Test case summary

Test ID	Test Steps	Test Input	Expected Result	Actual Result	Status
T01	Step 1: Open login page	Valid email and password	Redirected to home page	Redirected to home page	Pass
	Step 2: Enter credentials				
	Step 3: Click login button				
T02	Step 1: Open login page	Valid email, wrong password	Show error message	Error message displayed	Pass
	Step 2: Enter wrong password				
	Step 3: Submit form				
T03	Step 1: Go to poll section	No poll time selected	Show validation warning	Warning not shown	Fail
	Step 2: Click submit without choosing time				
	Step 3: Wait for response				
T04	Step 1: Open admin panel	Click reject button	Order marked as rejected	Order rejected successfully	Pass
	Step 2: Choose pending order				
	Step 3: Reject the order				
T05	Step 1: Visit upload research page	No login session	Redirect to login page	Page opened without login	Fail
	Step 2: Try uploading project				
	Step 3: Submit form				

Table 4.2: Test Case Summary Table

# Chapter 5

## Conclusion

### 5.1 Discussion

The Campus360 Solution project focuses on designing a web-based system to improve university campus operations through an integrated platform. In this phase, we concentrated on analyzing requirements and creating system design components, including diagrams and models to outline the project structure. The system addresses core features such as Cafeteria Management, enabling meal pre-orders, table-based ordering, and inventory updates; Bus Seat Management, allowing students to reserve and track bus seats efficiently; and Club Activity Management, providing dedicated club pages to streamline event sharing and membership activities. Additionally, the project design includes Student and Teacher Portals for accessing notices, class schedules, faculty information, and communication tools. While the current focus remains on diagrams such as class diagrams and data flow models, these outputs serve as a foundation for future implementation. The overall goal is to create a user-friendly platform that enhances productivity and resource management across the university campus.

### 5.2 Social Impact and Benefit

- **Improved Campus Life:** By streamlining daily operations like meal ordering, bus seat management, and club activities, *Campus360 Solution* creates a more organized, stress-free environment for students and faculty, improving their overall campus experience.
- **Increased Student Engagement:** The system's club activity management and communication hub feature provide a platform for students to actively participate in extracurricular activities and collaborate with alumni, fostering stronger community ties and skill development.
- **Better Resource Management:** Efficient use of resources such as cafeteria meals, bus schedules, and club activities reduces waste and ensures better allocation, benefiting both students and staff.
- **Enhanced Collaboration and Networking:** The communication hub connects students and alumni, creating opportunities for research collaboration and professional networking. This connection enhances academic and career development for students, contributing to their long-term success.

## 5.3 Limitation

1. **Limited Security Features in Design:** The current design does not address detailed security measures, such as encryption for sensitive data or advanced authentication mechanisms.
2. **Space Complexity:** Since the system integrates multiple subsystems, including cafeteria management, bus seat tracking, and club activities, managing and maintaining the system's data may lead to higher space complexity.
3. **Inaccurate Meal Waiting Time:** The system calculates an estimated waiting time for meal orders based on predefined rules. However, it cannot guarantee an exact waiting time due to varying preparation speeds and real-time kitchen workload.
4. **Dependence on Internet Connectivity:** The pre-ordering system and other features, such as bus seat management and product inventory tracking, rely on a stable internet connection. If the connection is lost, these features will not function properly.
5. **Product Sold Tracking Disruption:** In the cafeteria system, tracking sold products depends on the internet. If the connection fails, updates to inventory or sales data will be hampered, leading to inaccuracies.

## 5.4 Future Work

- **Tutor Connect Module:** We plan to implement a Tutor Connect feature, allowing students to post tutoring requests for specific durations, such as a day, a week, or a month. Experienced students can teach their peers as part of mutual academic support, fostering collaboration and enhancing the learning environment.
- **Enhanced Academic Classroom Integration:** In the future, we aim to implement a feature where the system will provide a dedicated classroom module for each section of every batch in each department. This module will include functionalities for both class material sharing and online classes. Currently, students and faculty rely on external tools like Google Classroom for sharing resources and Zoom for conducting online classes. By integrating these features into the Campus360 Solution platform, we will bring all academic activities—such as uploading class materials, managing assignments, and hosting online classes—into a single, unified platform. This will simplify the process, reduce dependency on external tools, and make the system more user-friendly for both students and teachers.
- **Automatic Section Allocation System:** In the future, we plan to implement an automatic section allocation system to address the challenges of manual course selection. Currently, students must manually select sections for each course, which can lead to conflicts and delays. The system will automatically generate all possible combinations of course sections for a given semester and identify conflict-free schedules for students. This feature will save time, reduce errors, and streamline the registration process.
- **Data Analytics and Reporting:** A data analytics module will be introduced to generate insights for the university administration. This feature will analyze cafeteria orders, bus usage patterns, and club activity participation to help in better resource planning and decision-making.

# References

- [1] Green University of Bangladesh home page. <https://www.green.edu.bd/>. Accessed Date: 2024-09-13.
- [2] University of Scholars home page. <https://ius.edu.bd/>. Accessed Date: 2024-09-13.
- [3] Primeasia University home page. <https://www.primeasia.edu.bd/>. Accessed Date: 2024-09-13.
- [4] State University of Bangladesh home page. <https://www.sub.ac.bd/>. Accessed Date: 2024-09-13.
- [5] Southeast University home page. <https://seu.edu.bd/>. Accessed Date: 2024-09-13.