2020-2021

JSS MAHAVIDYAPEETA

JSS SCIENCE AND TECHNOLOGY UNIVERSITY

SRI JAYACHAMARAJENDRA COLLEGE OF ENGINEERING

MYSURU-570006

Course Title: Software Design with UML CB430

Event

Report on

TITLE- Student MIS 3

Submitted by

Sl. No.	USN	NAME
1.	**01JST19CB024**	LITESHA NAGENDRA
2.	**01JST19CB042**	SAMANVI SATISH
3.	**01JST19CB052**	AFFAAN AHMED KHAN
4.	**01JST19CB058**	SHOAIB AHMED S

Submitted to

DR. D S VINOD

Associate Professor

Department of IS & E

JSS S&TU, SJCE, Mysuru

CONTENTS

1. ACKNOWLEDGEMENT

We express our sincere thanks to DR. D S VINOD for giving us an opportunity to work on this project, which not only has increased our awareness but also has taught the importance of teamwork.

We thank you sir, for giving us an opportunity to work with the Object Oriented Software Engineering Methodologies to develop this application.

The Design techniques have been very helpful throughout the course of this project and helped for efficient division of work among the teammates even though we were operating from distant places during the course of this lockdown.

It has been a great learning experience and this project has enabled us to thoroughly grasp the concepts of UML Modelling.

Thanks to the practices we employed while developing the solution for the Problem Statement, we all worked at maximum efficiency and shall continue using these practices throughout our academic regime.

PROBLEM STATEMENT

Student MIS 3

The software system is intended to support the office staff in handling the day-to-day academic activities. The
student MIS should provide the facilities:
1. Automation of Hostel accomodation.
2. Maintaining records of absences and attendance.
3. Canteen Management.

Introduction and Solution to the Problem

Student MIS has become important factors in the modern education field. This system should help the institution to streamline the administrative task and provide real-time access to the data.

We have used the design techniques taught in this semester to map out the problem in an organized way and elicititate the requirements for an efficient but feasible solution.

The student MIS should provide the facilities:

1. Automation of Hostel accomodation.

   Rooms are to be allotted automatically upon a new student entry. Various fee related stats, mess bill etc must be accessible to the admin.

2. Maintaining records of absences and attendance.

   Students' attendance is to be marked everyday. On access, the student/ admin must be able to see the number of days a particular student attended the classes.

3. Canteen Management.

   An automation of the Canteen is to be developed to take orders, change stock inventory and to add/remove employees.

Software Requirement Specification

4.1 Objective:

The purpose of this document is to define the requirements of the Student Management Information System. The supplementary specification lists the requirements that are not readily captured in the use cases of the use case model. The supplementary specification and the use case model together capture a complete set of requirements of the system.

4.2 Scope:

This specification defines the non-functional requirements of the system such as reliability, usability, performance, supportability, and security. As well as functional requirements that are common across a no. of use cases.

4.3 Common Functionalities:

- Hostel Accomodation
    - Efficient allocation of hostel rooms
    - Multiple students can register

- Attendance Management
    - Multiple students can mark their attendance
    - Easily track attendance information of students

- Canteen Management
    - The owner can add the employee and update the employee details.
    - The employee manages the sales and order.

4.4 Usability:

The System's interface of the software should be made user friendly and should be efficient.

4.5 Reliability:

The system shall be available 8 hours a day from 9-5pm, 6 days a week to no more than 10% downtime.
The software will not be able to connect to the centralized database in the event that the college LAN fails or in the event of the server being down due to a hardware or software failure.

4.6 Performance:

The system shall support up to 100 simultaneous users against the central Database of any time.

4.7 Security:

The security requirements deal with the primary security. The software should be handled only by the administrator and authorized users. Only the administrator has the right to assign permission like creating new accounts and generating passwords. Only authorized users can access the system with username and password.

Use case model

5.1 Actors

Actor is something external and interacts with the system. Actor may be a human being or some other software system. For our system, Student, Admin and Store Owner are the actors.

5.2 Use cases Use cases represent the functional requirements:

- Hostel Management
  Actors: Admin, Student
  - Allot Room
  - View Funds
  - Modify Student Information
  - Modify Mess Information
  - Student Registration

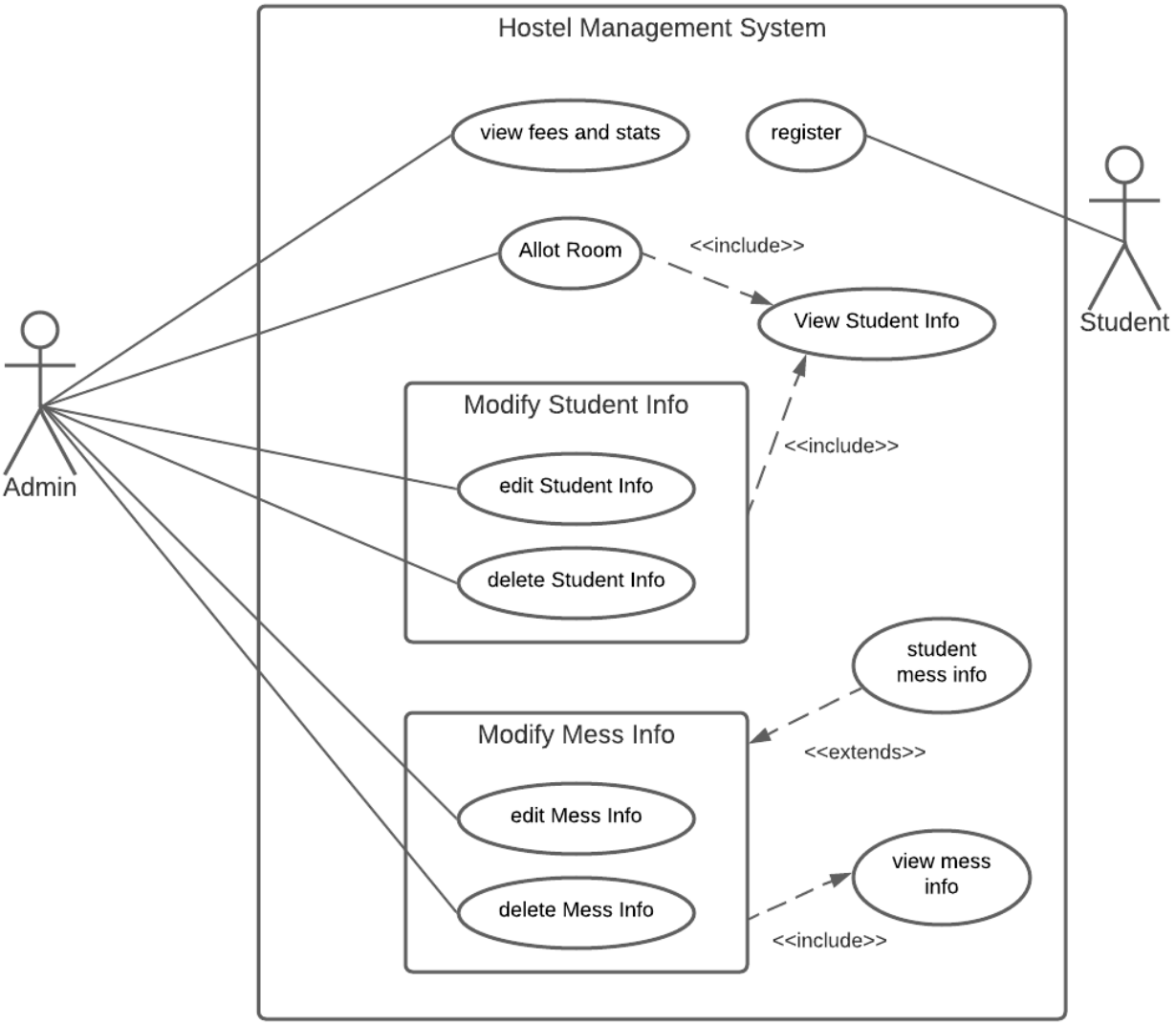
- Attendance Management
  Actors: Student, Admin
  - Add/Remove Students
  - View Student List
  - Checking Presence Count

    - Marking Attendance

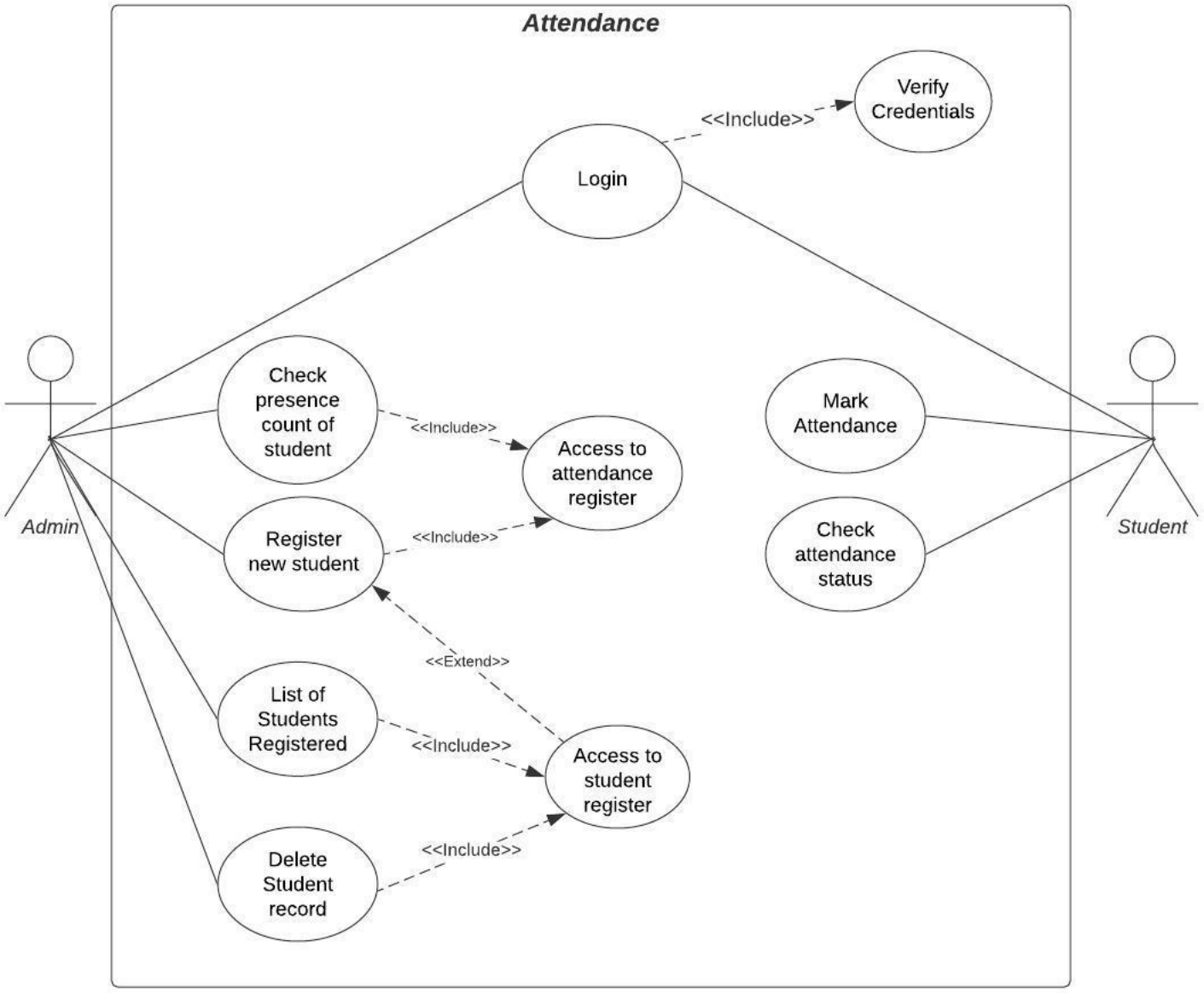
- Canteen Management
  Actors: Owner, Employee, Customer (Student)
    - Add/Remove Employee
    - Edit Inventory Stock
    - Take Order
    - Place Order
    - View Sales Record
    - Delete Sales Record

5. Use Case Model

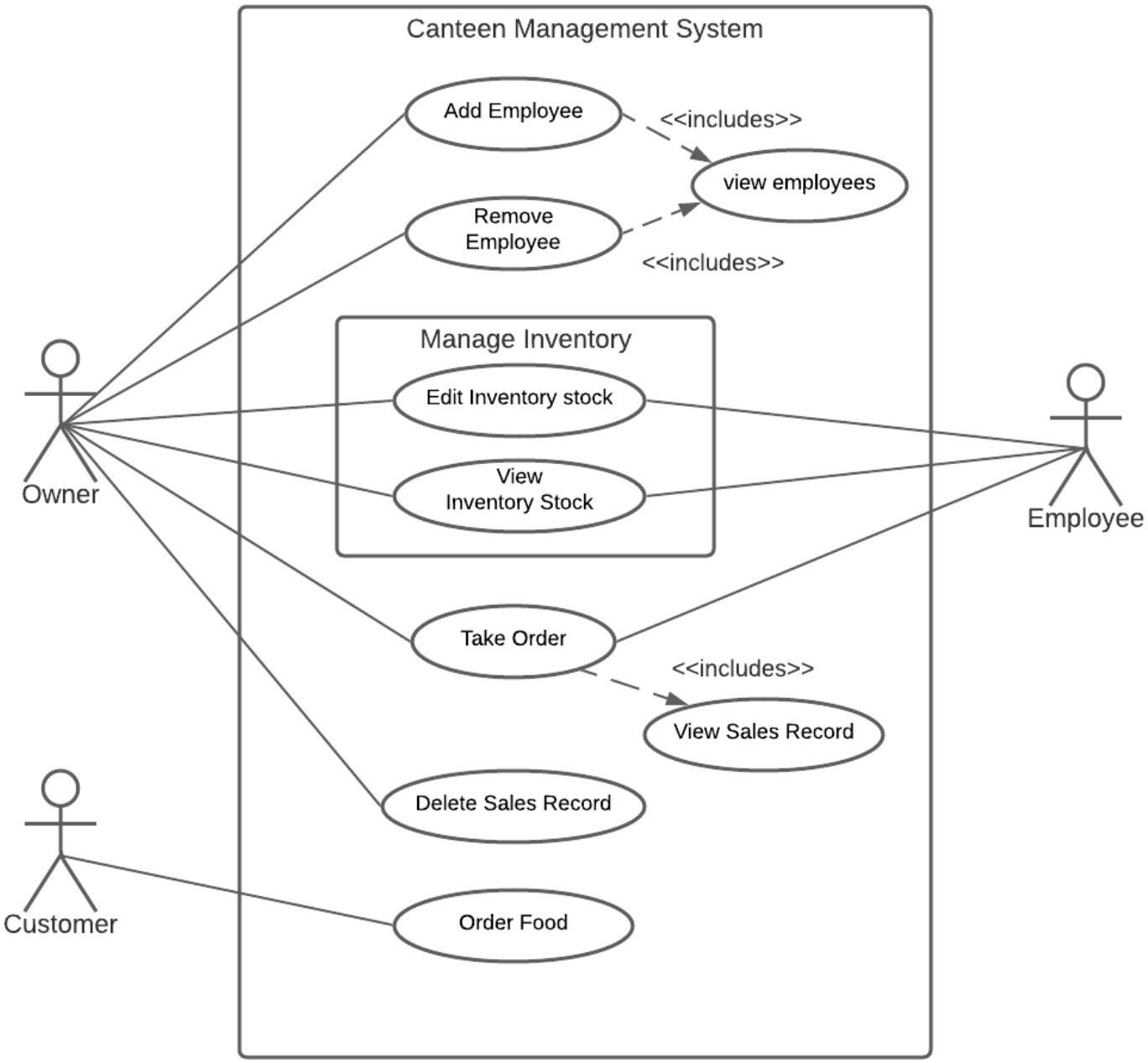
- **Hostel Management System**



- **Attendance Management System**

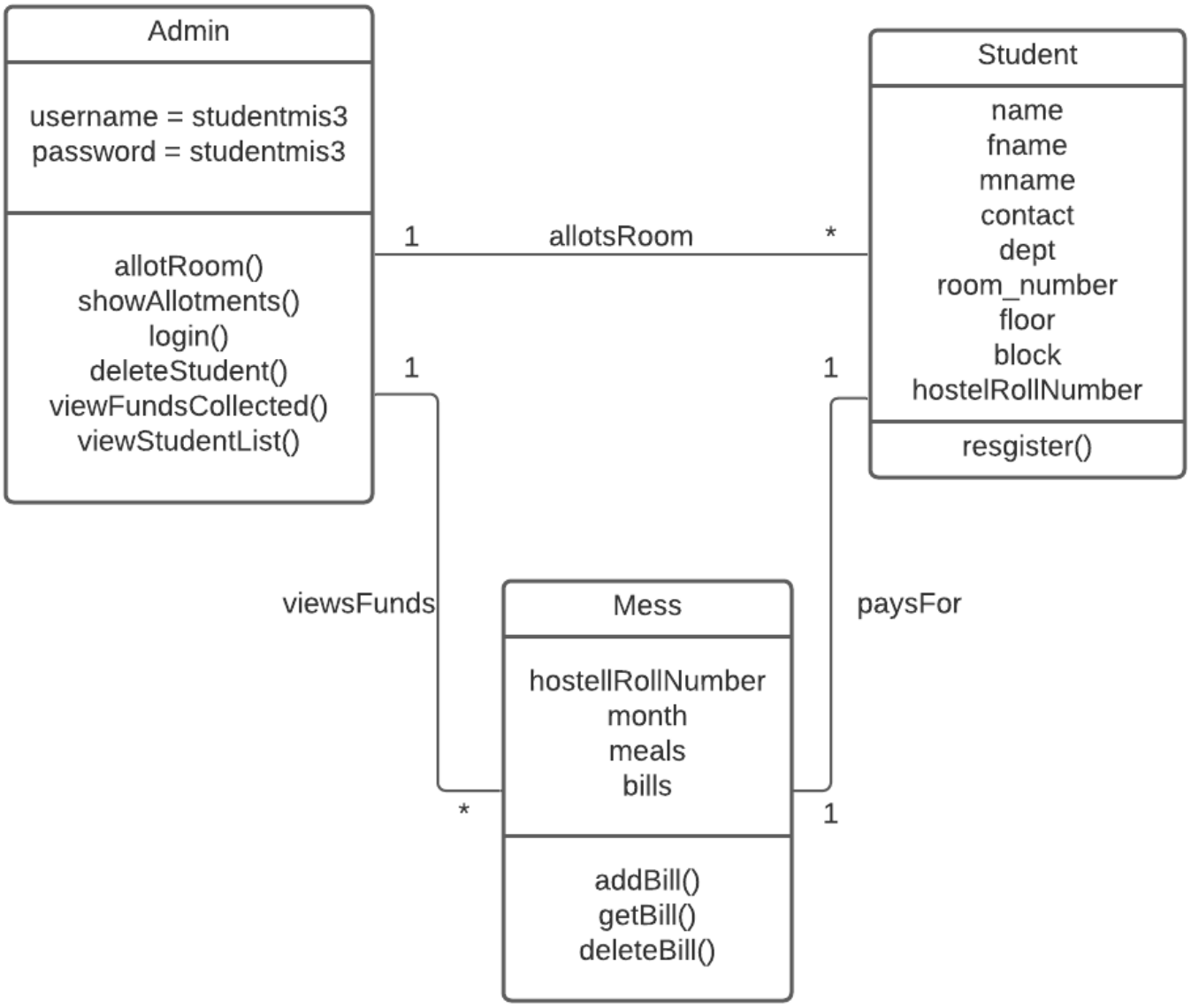


- **Canteen Management**

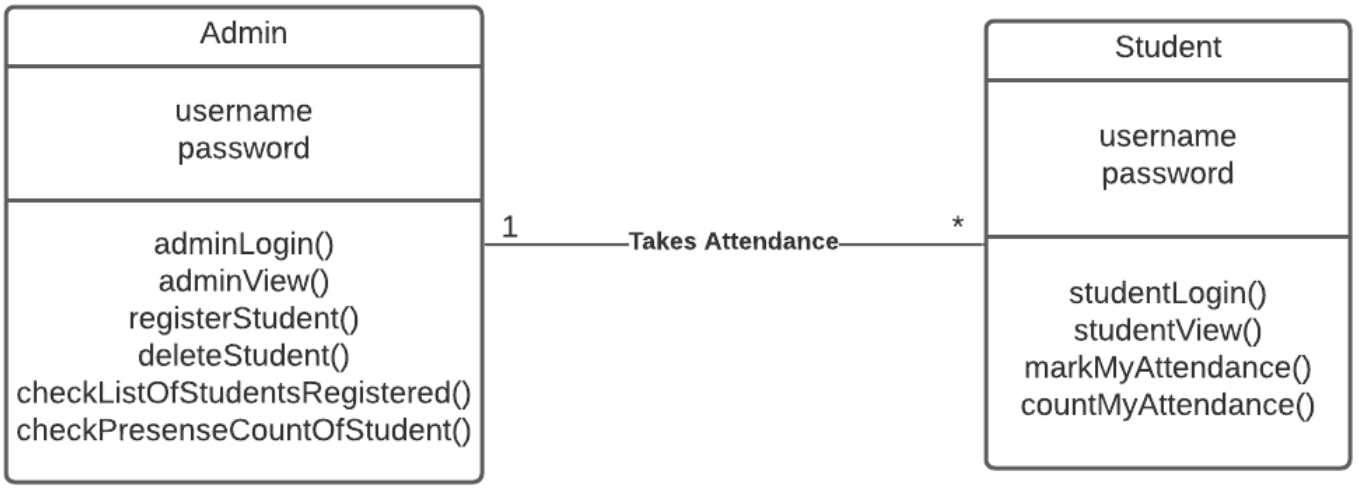


6. Class Diagram

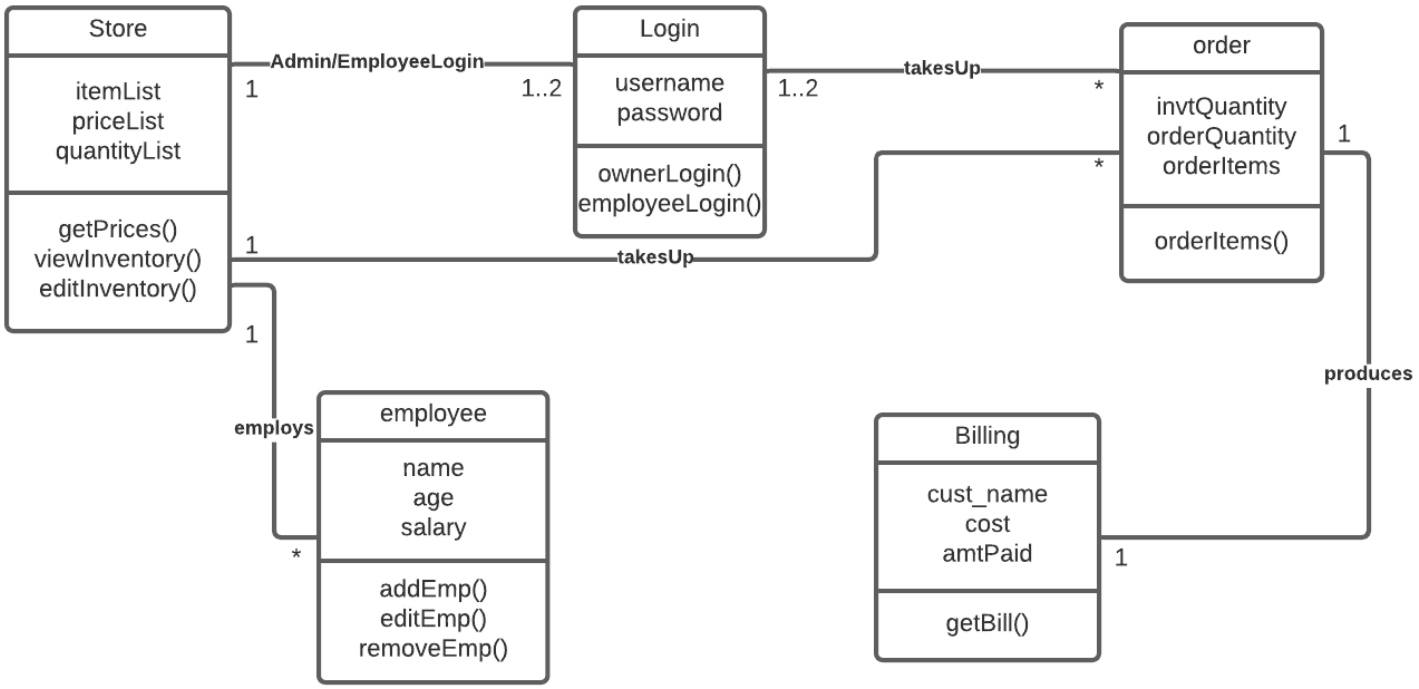
- **Hostel Management**



- **Attendance Management**

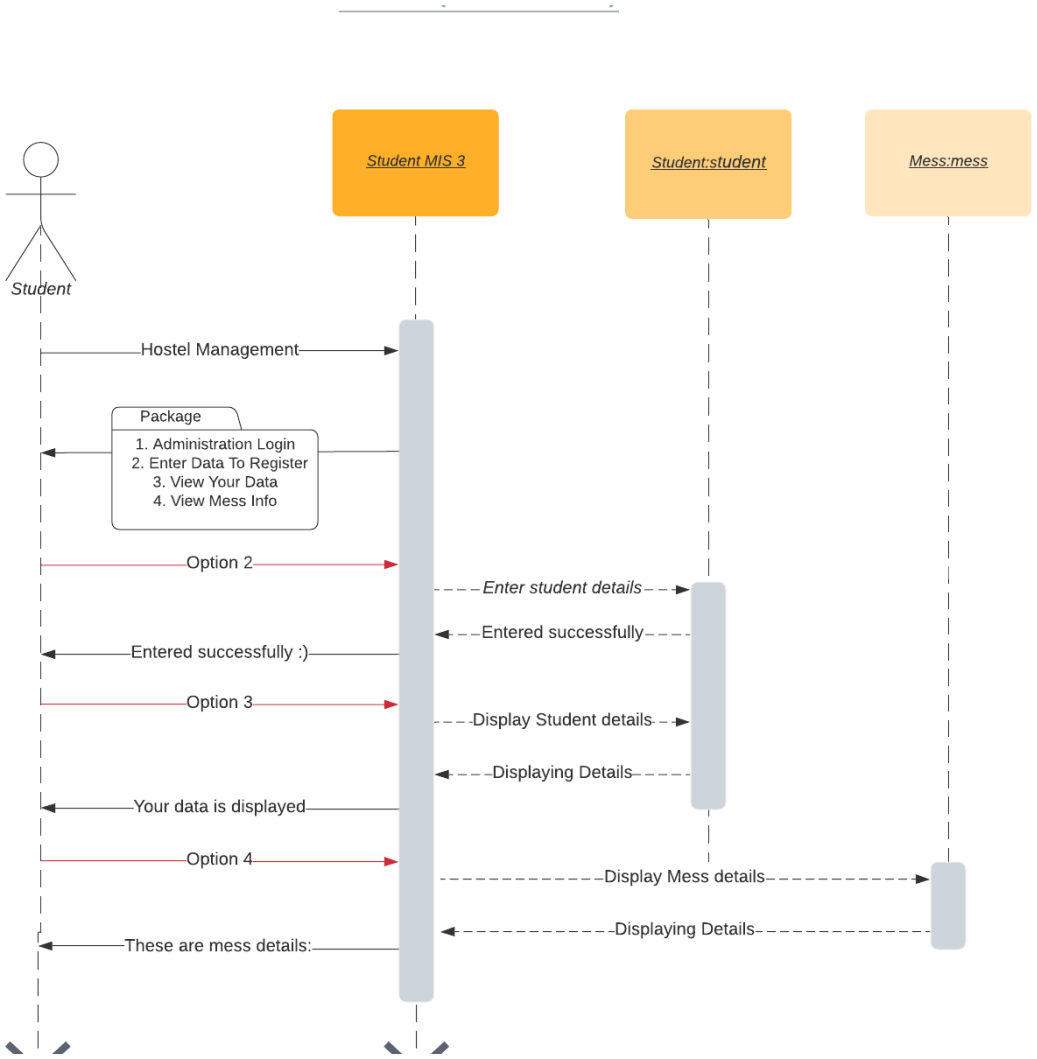


- **Canteen Management**

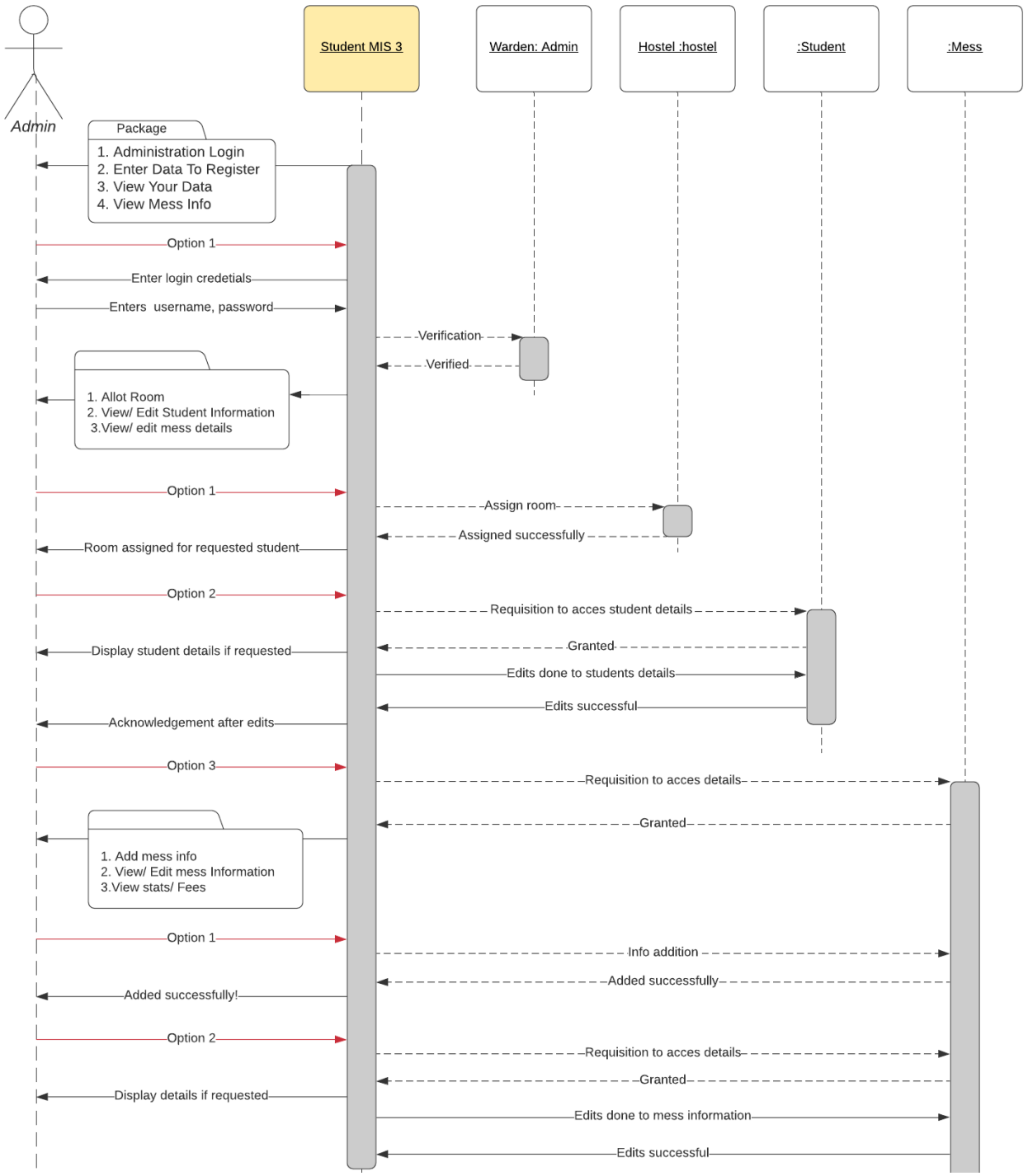


7. Interaction Diagram

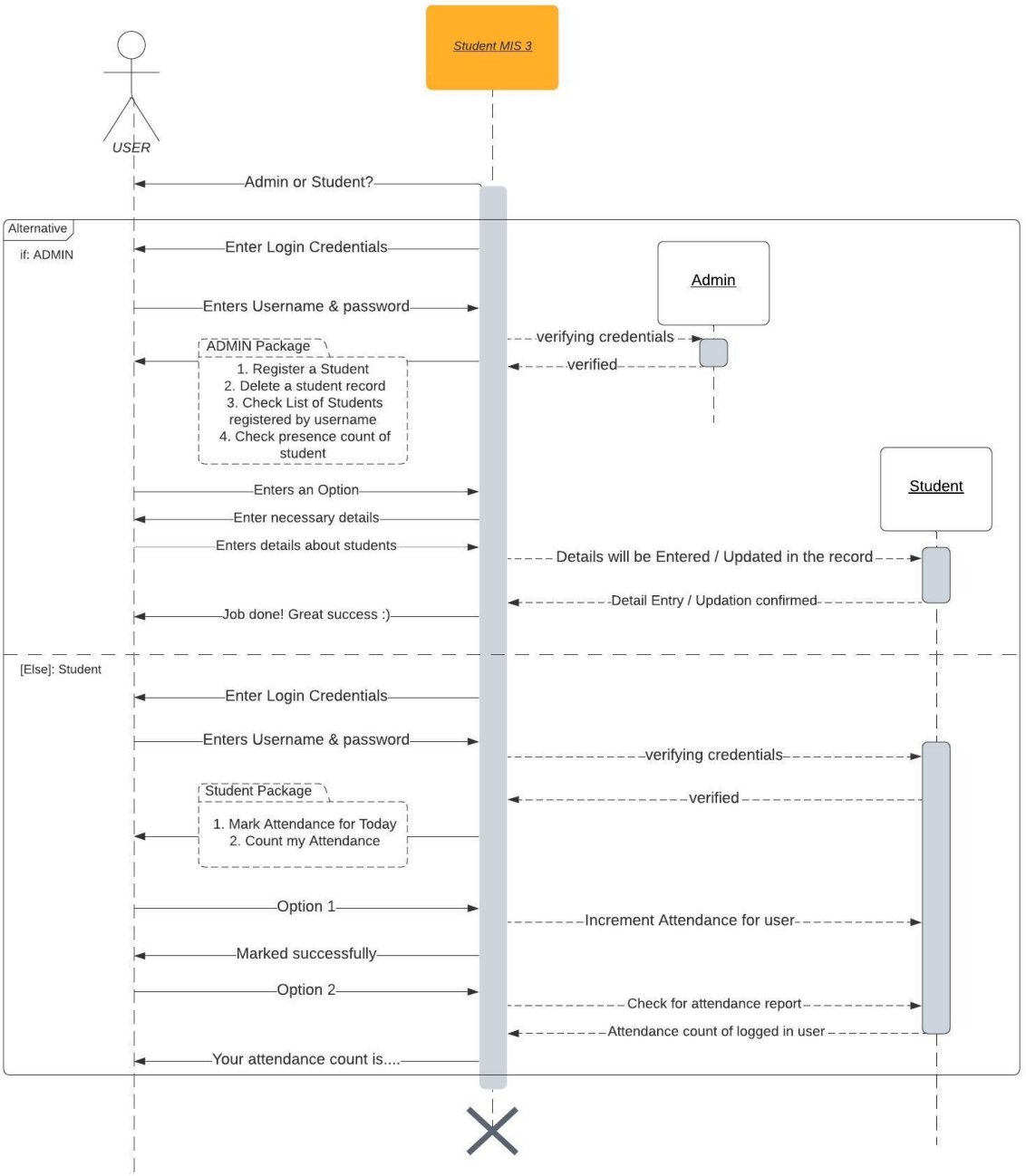
- **Hostel Management**
  - **Student Interaction**



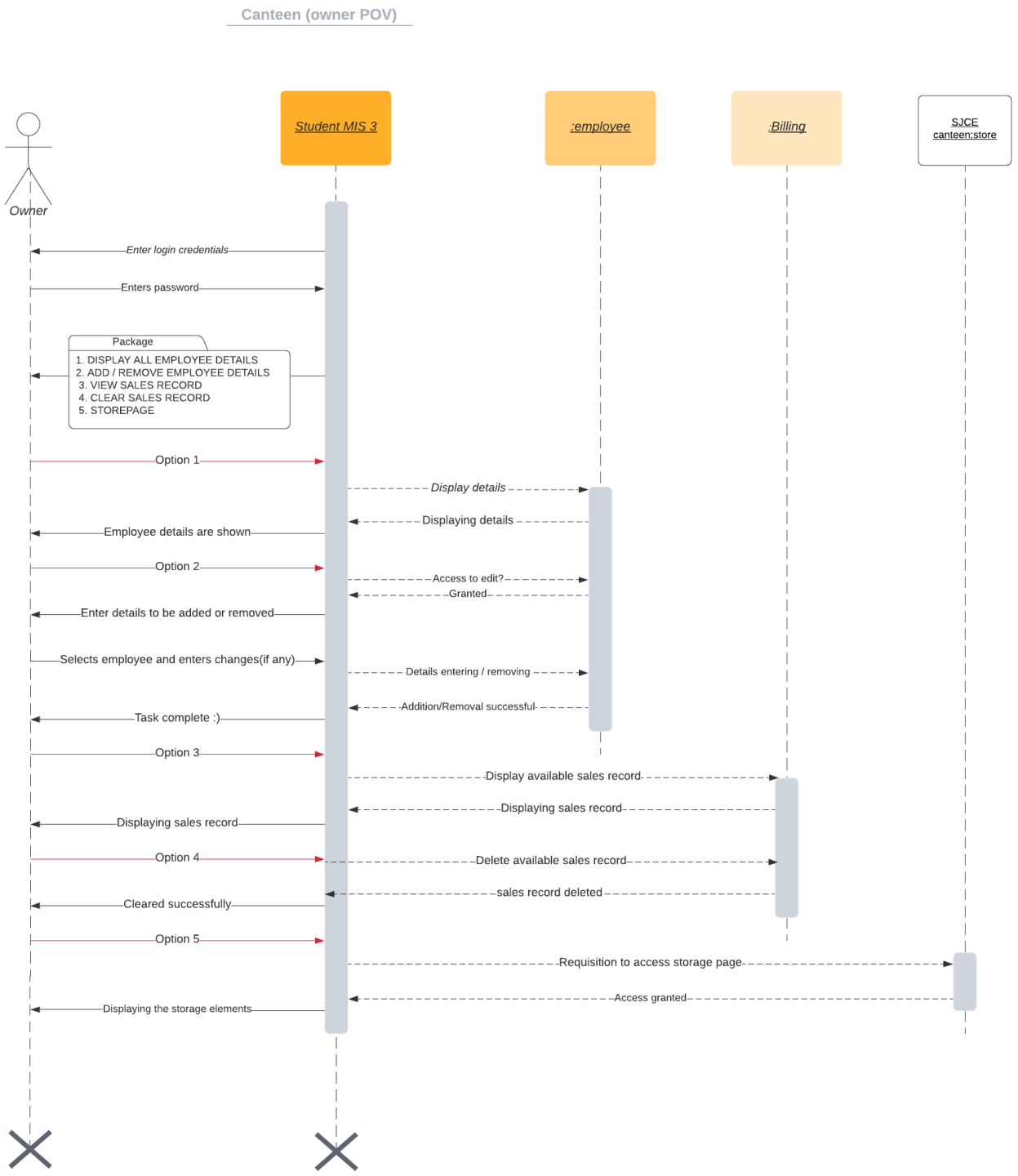
- **Admin Interactions**



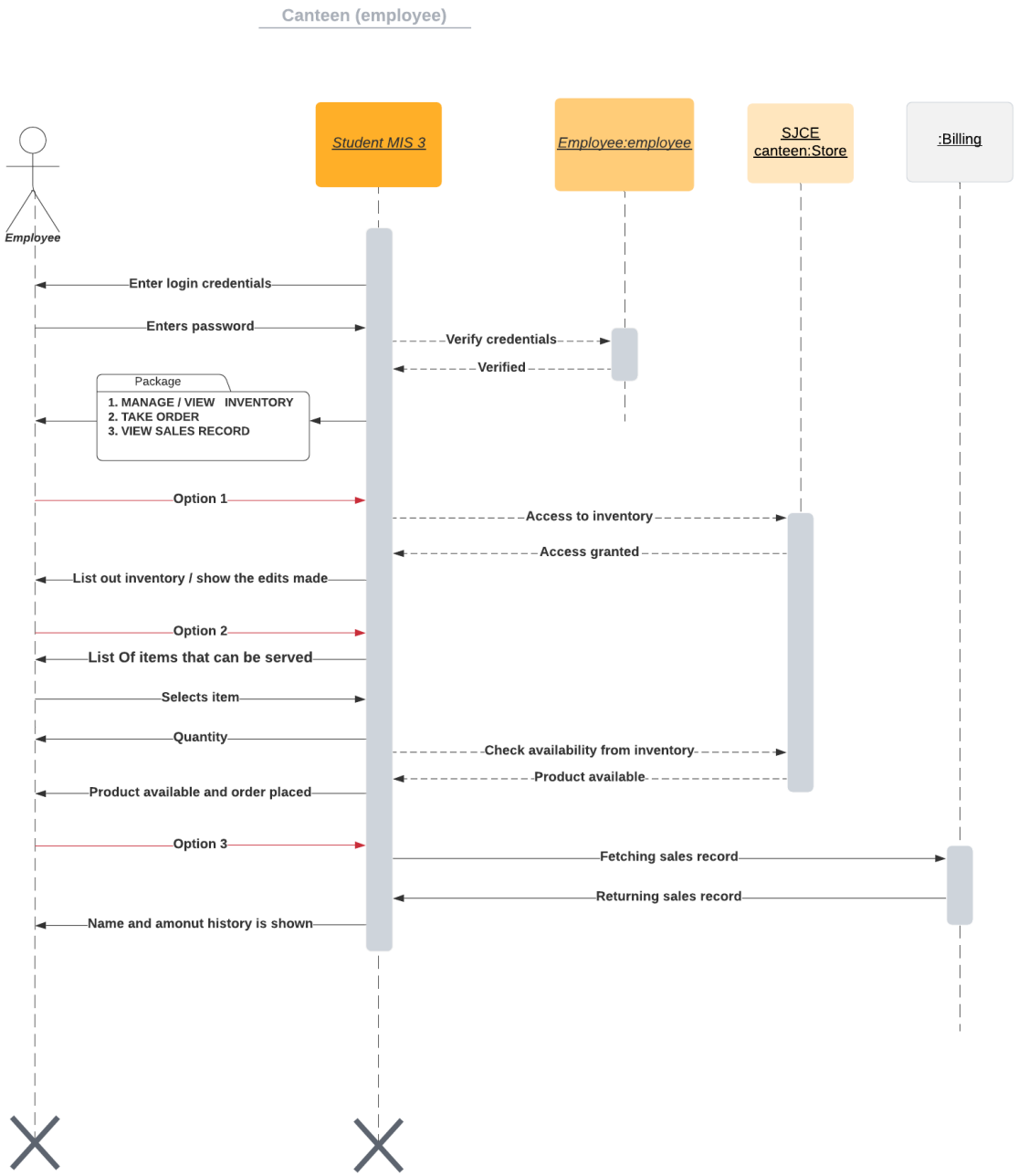
- Attendance Management System



- Canteen Management
  - Owner Perspective

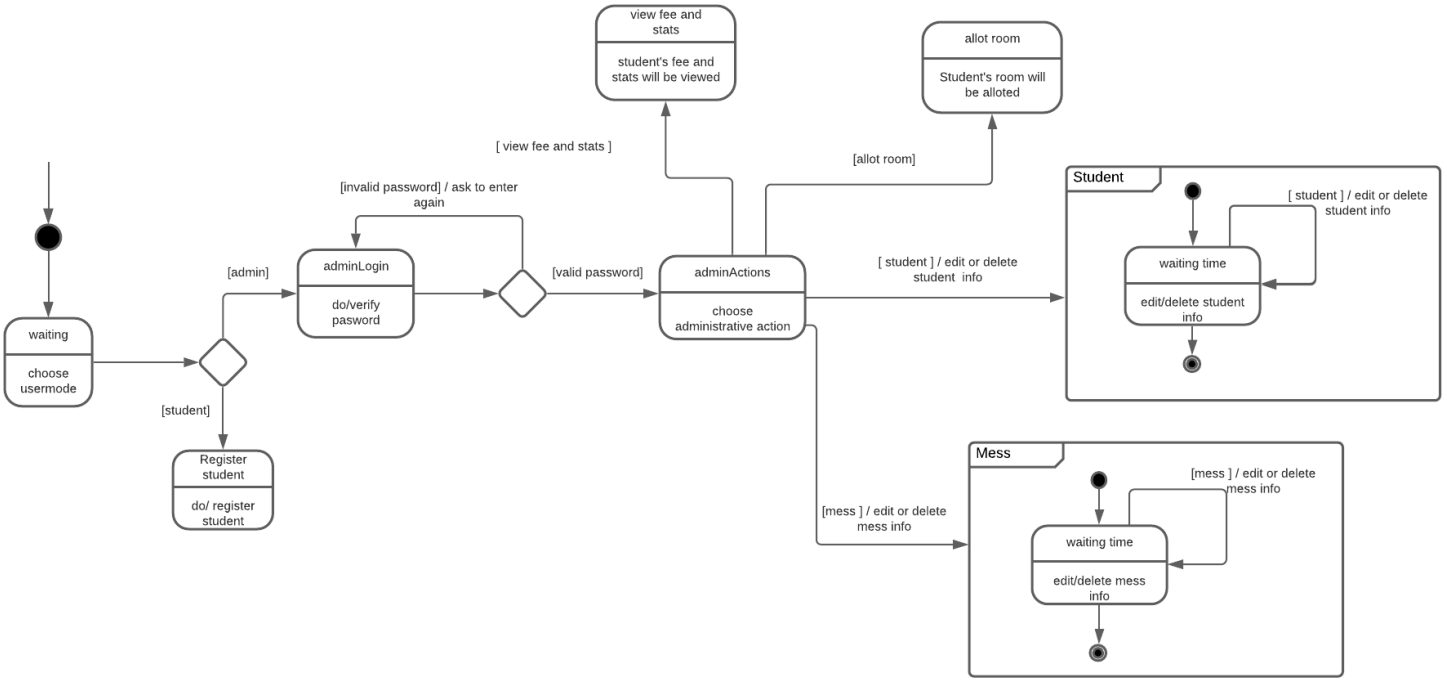


- Employee perspective:-

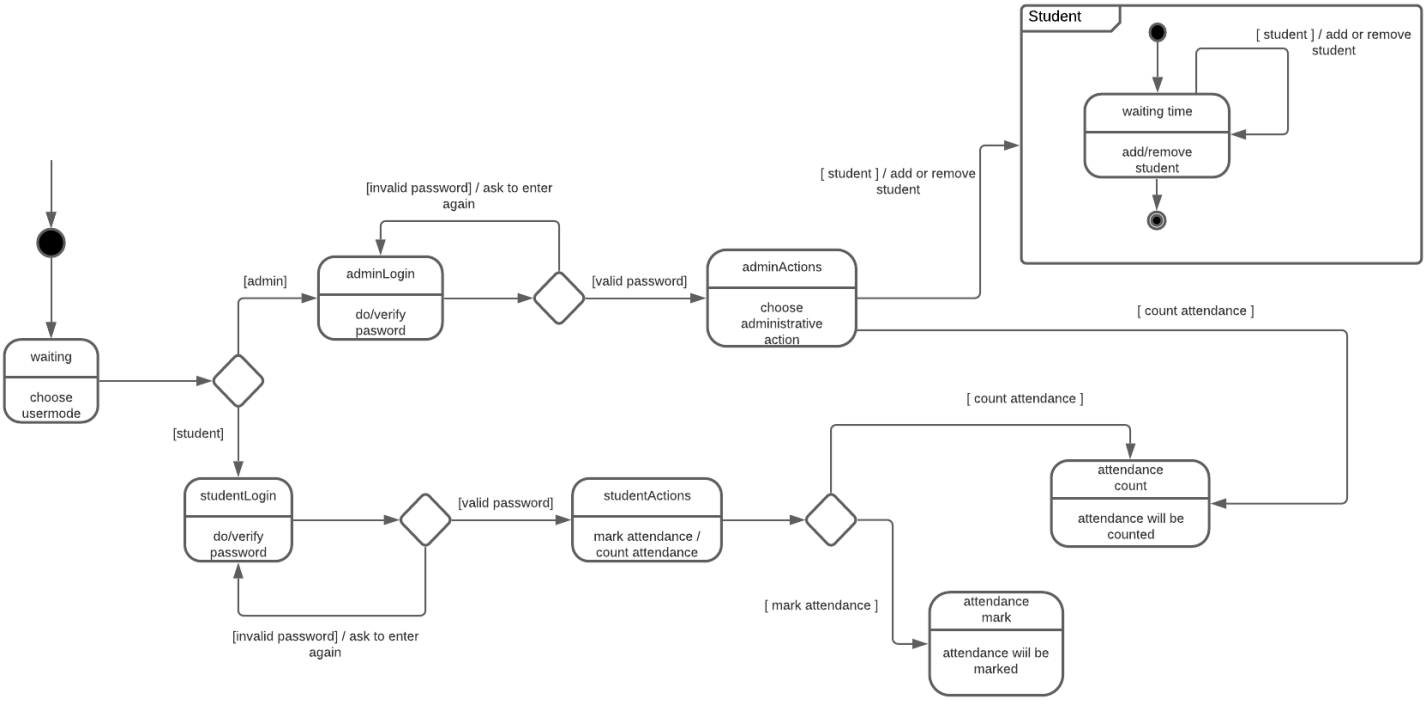


8. State Machine Diagrams

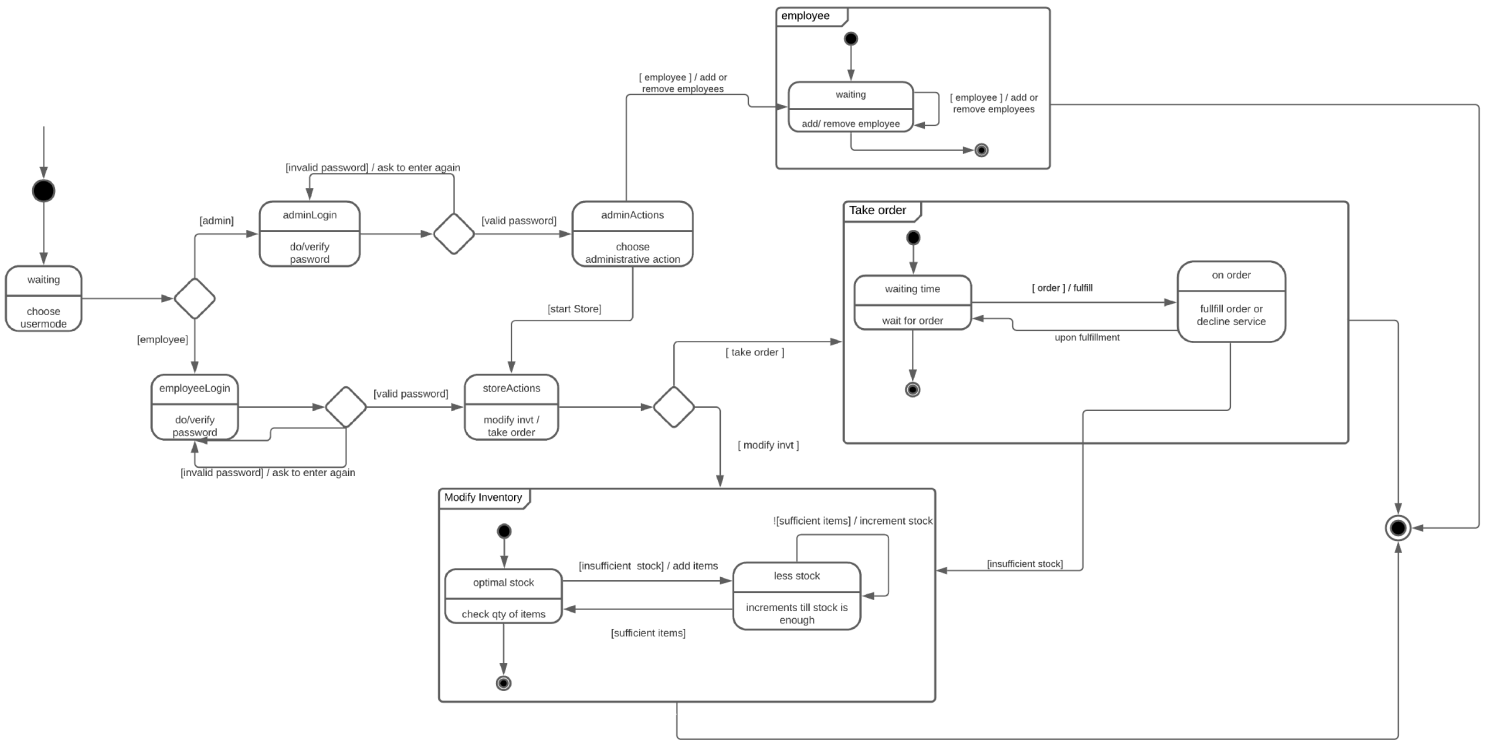
- **Hostel Management System**



- **Attendance Management System**

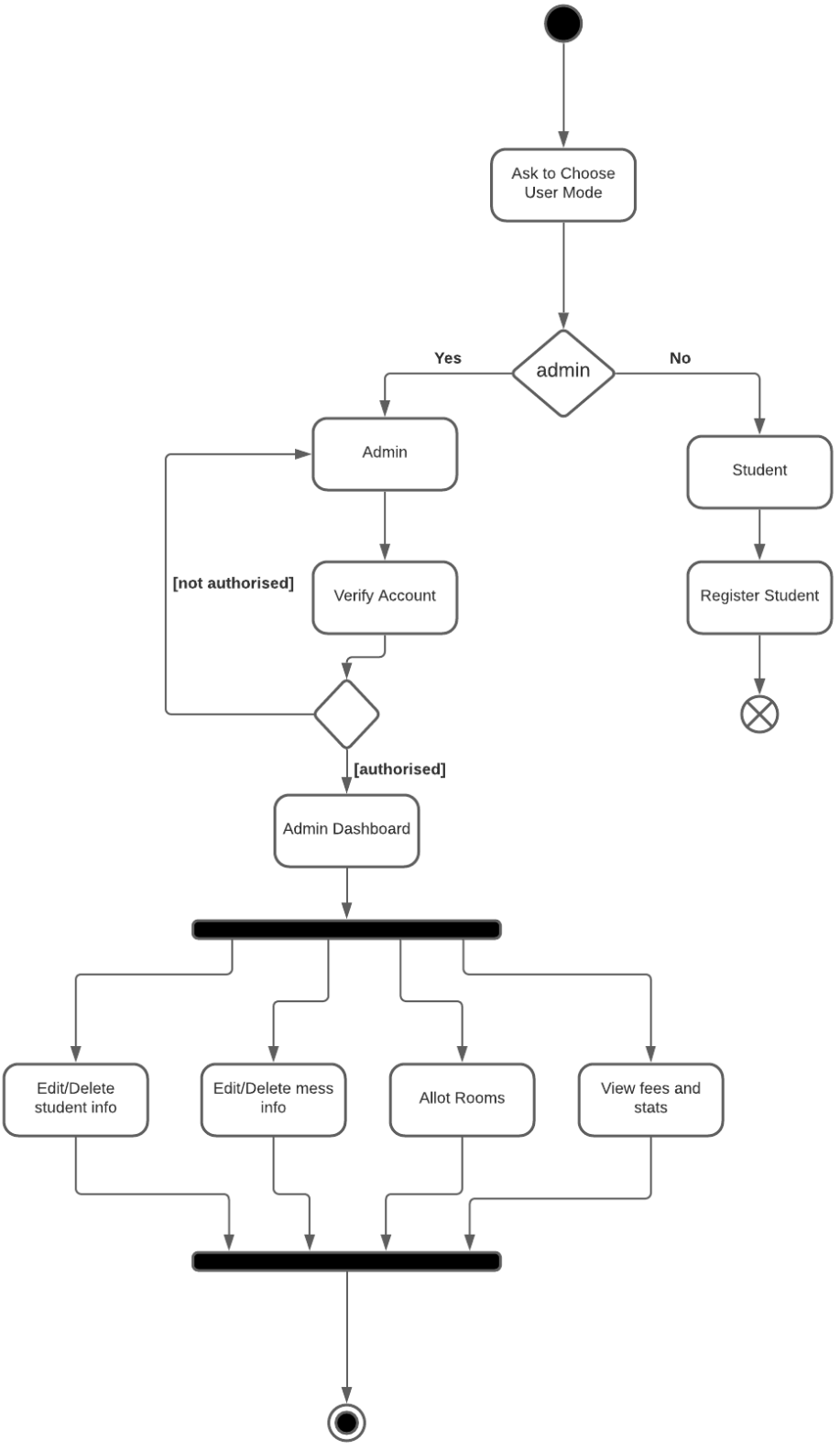


- **Canteen Management**

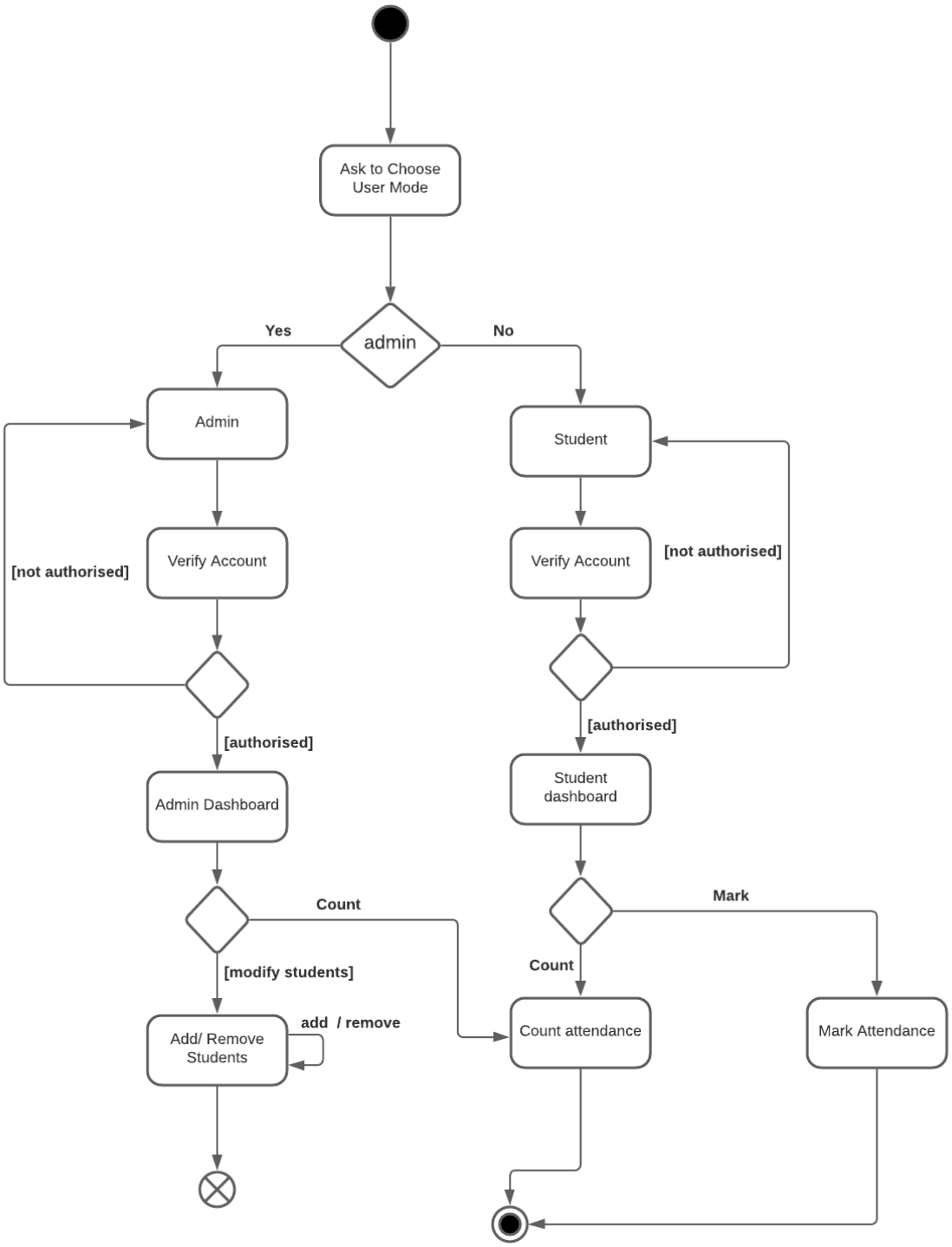


9. Activity Diagram

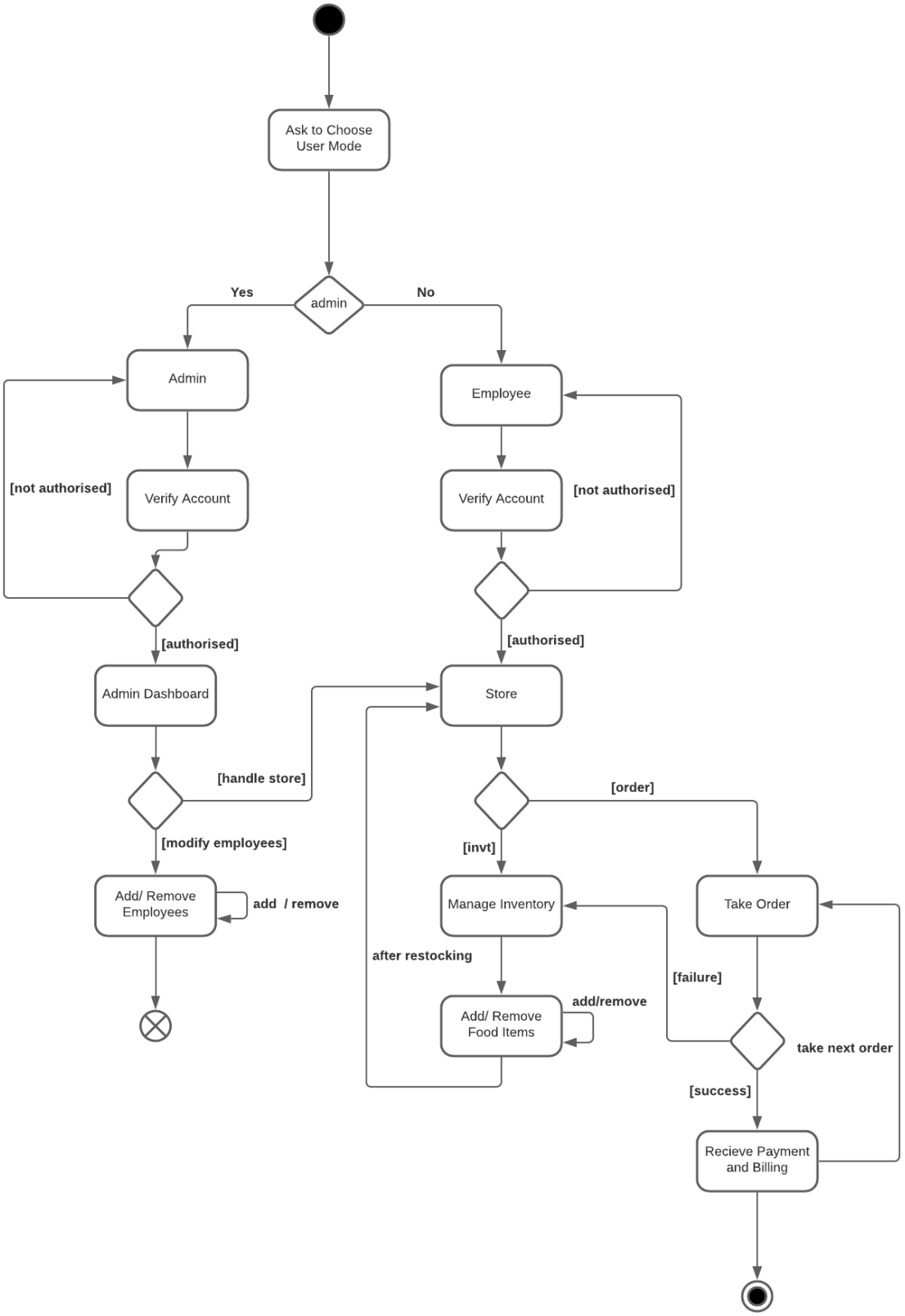
- **Hostel Management System**



- **Attendance Management System**



- **Canteen Management**



10. Necessary Assumptions Made, if any

We have broadly divided the Student Management Information System into 3 parts. The Hostel Management Section, Attendance Management Section and the Canteen Management Section.

Constraints: Hostel Management System

- The Hostel in the picture has various blocks and floors. It's not a single building with 'n' floors.
- The students pay their mess bill time-to-time in real time.

Constraints: Attendance Management System

- We have implemented the Attendance Management System for one course only.
- Every student is assigned an unique username by the system admin to clock their attendance.

Constraints: Canteen Management System

- The food menu is fixed (not dynamic)
- There are 3 food items. (For Demonstration Purposes)

There is no Database for the application, we have created a simulation using the File Handling functionalities of C++.

In real-time implementation,

- We assume that the computers that will use the software will be part of the college LAN.
- Users with administrator access should be careful in deleting or modifying any information knowingly or unknowingly which will lead to inconsistency of the database.
- The end users of this software are assumed to have basic level of computer knowledge i.e. point and click

11. Identify Inputs and Outputs

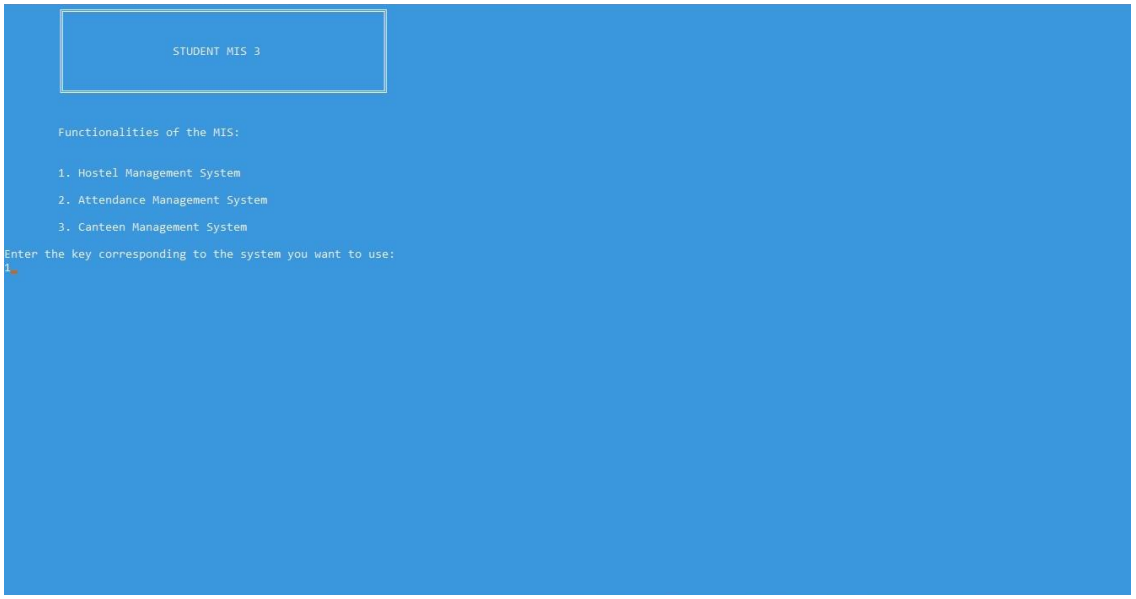
- Hostel Management System
    - Input :
        - Admin logs in by entering his credentials. They can either allot a room, delete a student, view student list, view funds collected, view mess details by entering the appropriate input symbol(s)
        - Students enter the details for registration.
    - Output :
        - The required Administrative operation shall be performed.
        - Information will be stored in the backend (Files in our Simulation).
        - Students get Registered.

- Attendance Management System
    - Input :
        - Admin logs in by entering their credentials. They can add/remove students and can also check attendance status of the student upon entering the appropriate input symbol(s).
        - Students log in by entering their credentials. They can clock their attendance and can also view their attendance status
    - Output :
        - The required Administrative operation shall be performed.
        - Information will be stored in the backend (Files in our Simulation).
        - Students' attendance will be marked for the day and can also view their attendance.

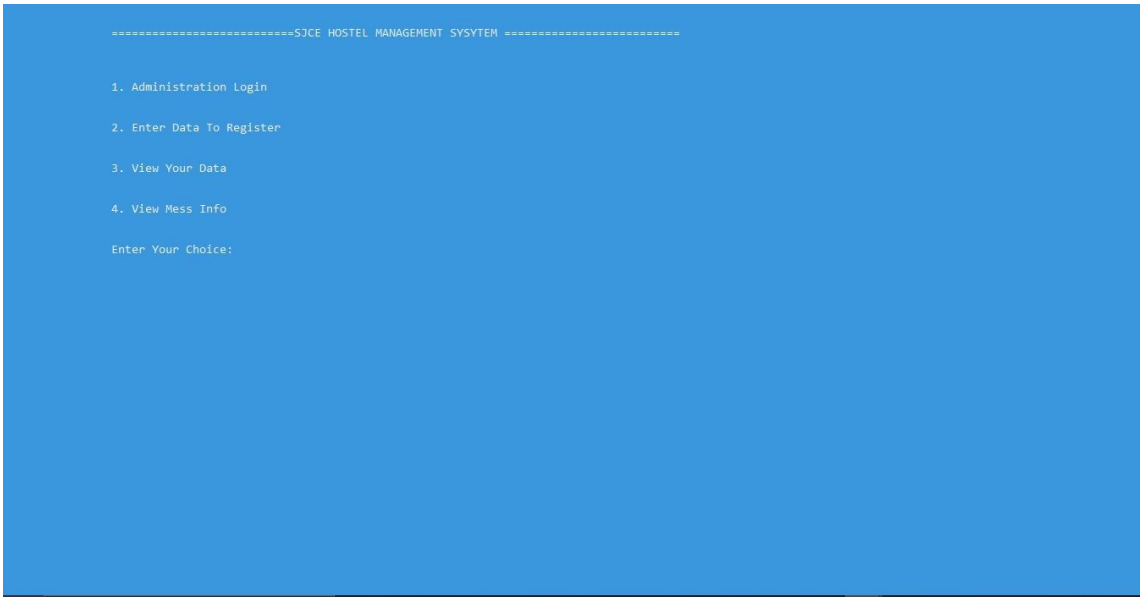
- Canteen Management System
    - Input :
        - Admin/ Store Owner logs in by entering their credentials. They can add/remove employees, edit stock inventory, take up orders and view/delete the sales record upon entering the appropriate input symbol(s).
        - The Employees also log in by entering the “employee” credentials. They can edit stock inventory, take up orders and view the sales record by entering the appropriate input symbol(s).
    - Output :
        - The required Administrative operation shall be performed.
        - Information will be stored in the backend (Files in our Simulation).
        - Employee/Owner shall take up orders, edit inventory etc.

12. Screenshots

- **Home Page**



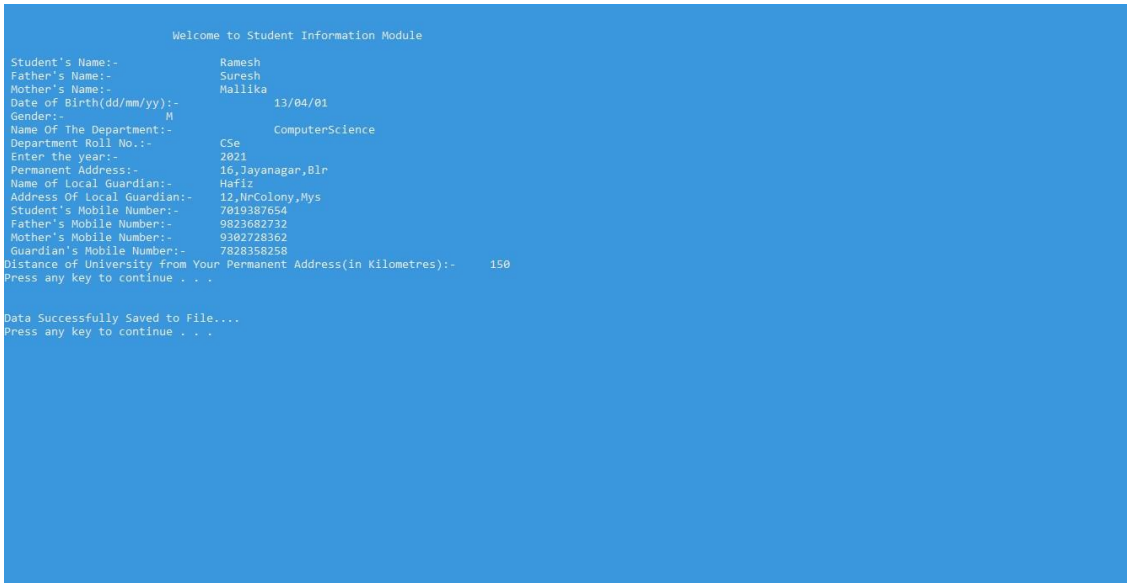
- **Hostel Management System - Home Page**



- **Hostel Management System - Admin Page**

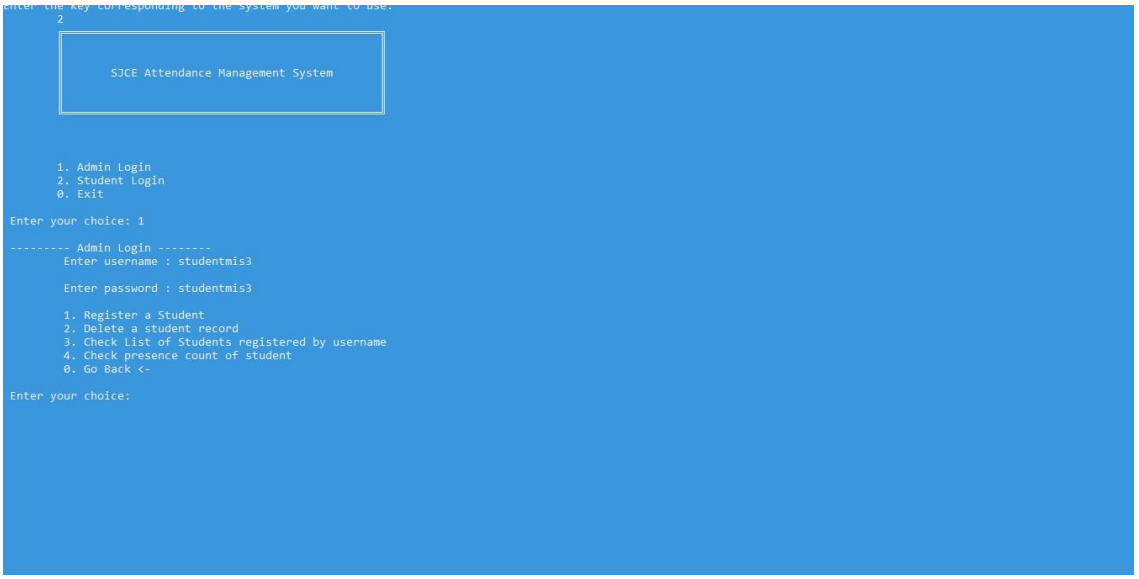
- **Hostel Management System - Allot Room**

- **Hostel Management System - Student Registration**

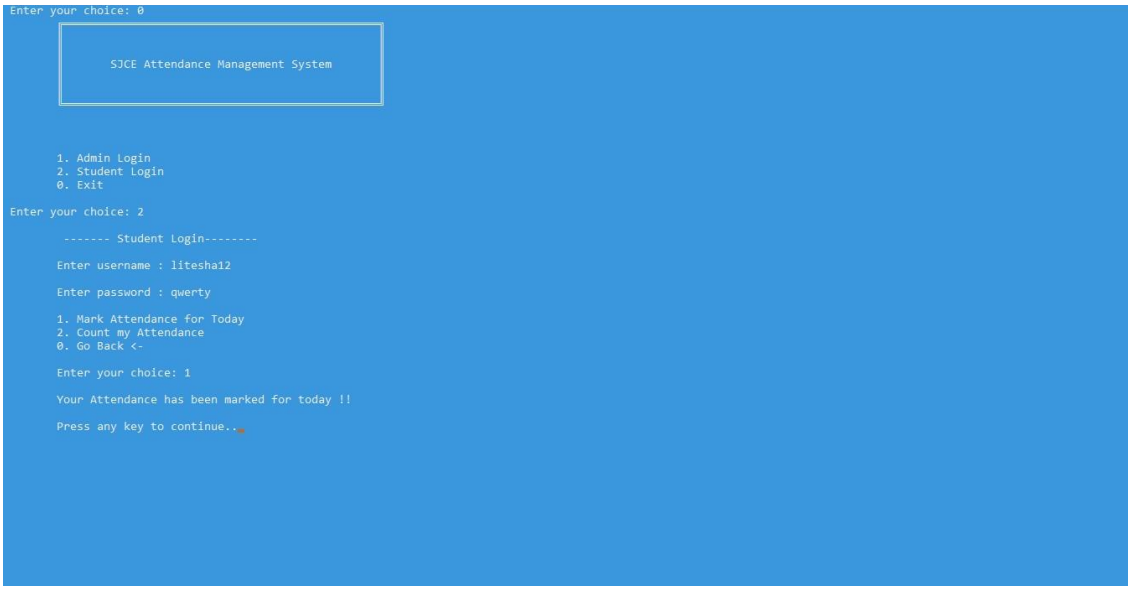


\

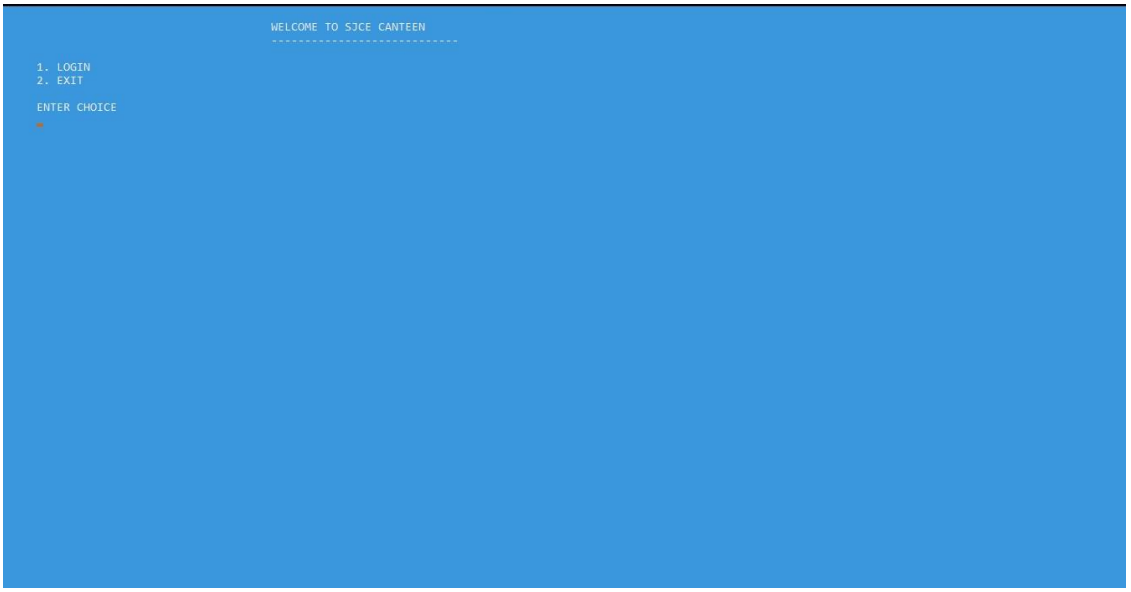
- **Attendance management system : Home page**



- **Attendance Management System - Student Page - Mark Attendance**

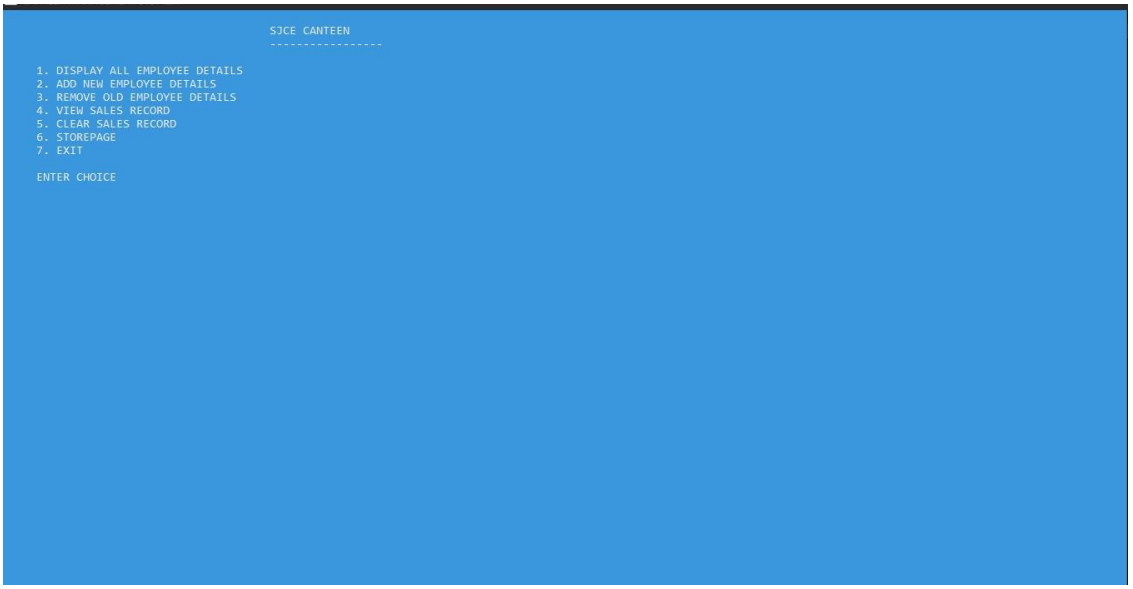


- **Canteen Management System - Home Page**



- **Canteen Management System - Login Page**

- **Canteen Management System - Admin page**



- **Canteen Management System - Admin Section - Adding Employee**

- **Canteen Management System - Admin & Employee Section - Taking Order**

13. Result and Inference

This is the requirements and design document of "Student MIS" using UML notations that defines a standard way to visualize the way our system has been designed.

This is to automate all the pre-existing amenities such as Attendance Management, Hostel Accomodation and Canteen Management so as to maximize efficiency and to keep a record of everything.

Inference:

So finally we have come up with the visual representation of "Student MIS 3". This document portrays the behavior and the structure of the system. This makes the implementation easier for us (Software Engineers). It also helps system architects with the modeling, design and analysis. This document helps us capture:

- Static aspects or structure of a system. Structural diagrams include; Class diagrams, Component diagrams and Deployment diagrams.
- Dynamic aspects or behavior of the system. Behavior diagram includes; Use Case diagrams, State diagrams, Activity diagrams and Interaction diagrams.

Finally, this gives the blueprint of our system which can be used for the development phase.