

**HOSTEL MANAGEMENT SYSTEM**

SDA PROJECT

SECTION: BCS-5K

**MISS ROMASHA KHURSHID**



GROUP MEMBERS

NAIMATULLAH KHAN (21K-4540)

MOHSIN KHAN (21K-4530)

MUHAMMAD NOHAIL (21K-4608)

**Project Background:**

This project addresses the need for an efficient hostel management system for students at our university. It aims to showcase the essential functionalities involved in hostel management.

**Project Scope:**

The system serves as an educational tool for students to understand practical management systems. Additionally, it provides hostel owners with a platform for designing and implementing their management processes.

**Not In Scope:**

The project excludes a chat box for student assistance, salary management of wardens, and rent management for the entire hostel building.

**Project Objectives:**

To give an idea of how management is done, how many functionalities are involved in seemingly simple tasks.

**Stakeholders:**

Stakeholders include users (students and wardens), administrators, and database developers.

**Operating Environment:**

The system operates through web browsers on laptops and computers.

**System Constraints:**

* Web server limitations.
* Dependency on internet connectivity.
* Opposition from students not residing in hostels or those against hostel living.
* Institutional reluctance towards private hostel management.
* Limited skilled personnel in the deployment environment, particularly in database management.
* User constraints, where students and wardens may not be cooperative.

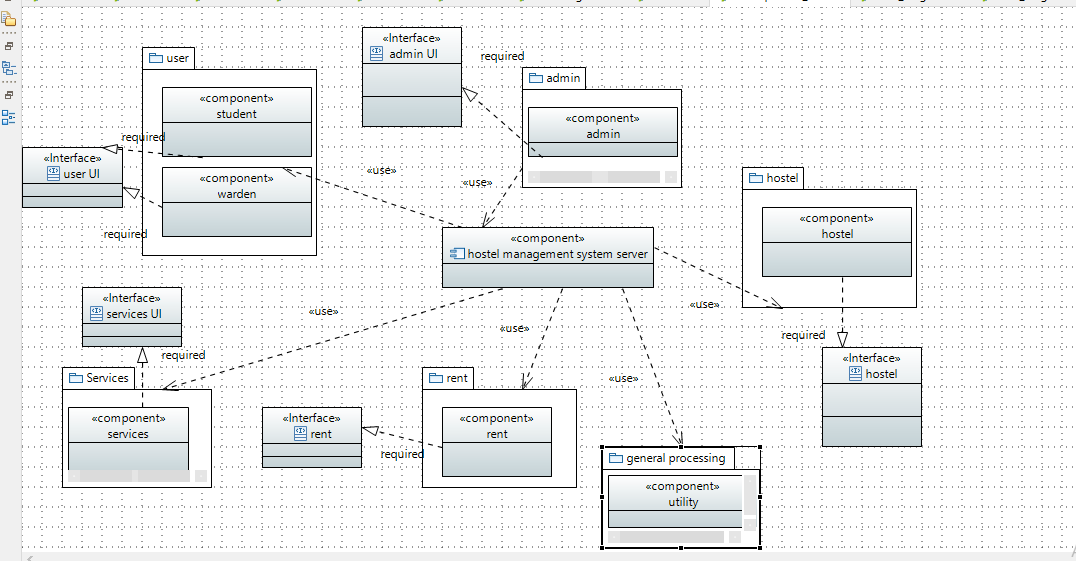
**Assumptions & Dependencies:**

We are dependent on our database system, due to which we can store and retrieve so much data. We are dependent on the internet for this system to work.

**Hardware Interfaces:**

A computer is essential for managing and viewing the database, serving as the primary hardware interface for the system.

**Software Interfaces:**

The following component diagram describes the components of the classes, their connections, their dependencies and uses.

**Functional Hierarchy:**

The functionalities are either by the student or the warden.

For warden:

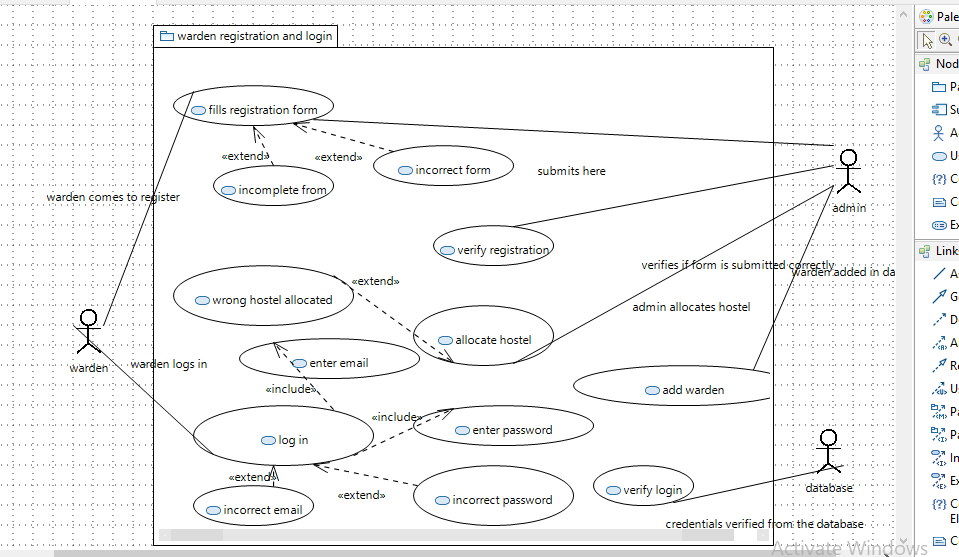
* Registration
* Log in
* Add student
* Add/Delete facility in hostel
* Check student rent status
* Generate rent voucher
* Validate student payments

For student:

* Registration
* Log in
* Avail/Discontinue facility
* Complain regarding a facility
* Check rent voucher
* Pay rent

**USE CASES**

1. **Warden registration and login**



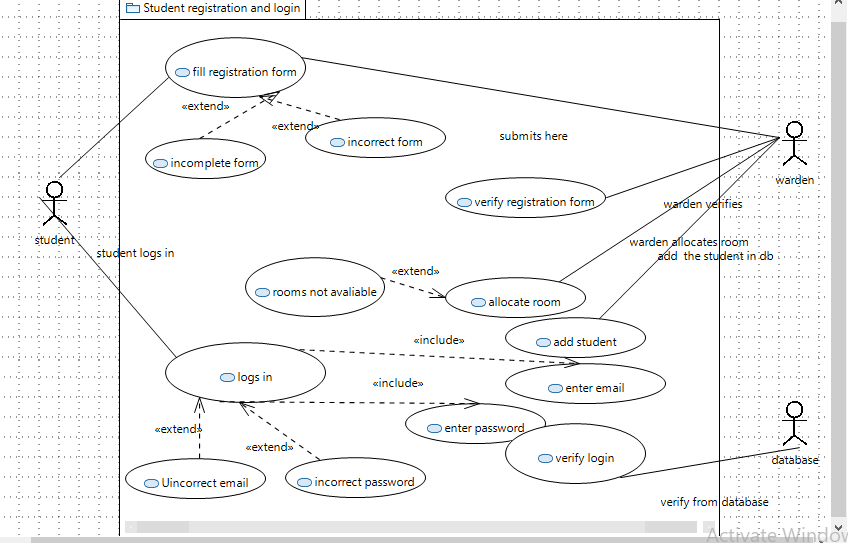
|  |  |
| --- | --- |
| ID | UC1 |
| USE CASE NAME | Warden registration form |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The system will add the warden in the system |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Student, database |
| PRECONDITION | The user is connected to the internet. |
| POSTCONDITION | The Warden has successfully registered. |
| MAIN SUCCESS SCENARIOS | 1. The warden fills a form in which he enters his information. 2. The warden submits his form. |
| EXTENSIONS | \*a. At any time, the system crashes. |
|  | 1a. The warden does not fill a necessary column.  2a. The warden forgets to submit the form. |

|  |  |
| --- | --- |
| ID | UC2 |
| USE CASE NAME | Verify registration form |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The admin checks if the form is submitted correctly |
| PRIMARY ACTOR | Admin |
| SUPPORTING ACTOR | Warden, student |
| PRECONDITION | The admin staff is connected to the system  The warden has filled the registration form |
| POSTCONDITION | The form is approved and the warden is added |
| MAIN SUCCESS SCENARIOS | 1. The system checks all the details which are provided in the form. 2. The system approves the form. 3. The system allocates a hostel to the warden. 4. The system enters the warden details in the   database. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. The system is unable to detect an invalid form.  2a. the system is unable to approve the correct form.  3a. the system does not provide hostel to the warden.  3b. The system gives a hostel to the warden which already has a warden.  4a. the system enters incomplete information in the database.  4b. the system enters incorrect information in the database. |

|  |  |
| --- | --- |
| ID | UC3 |
| USE CASE NAME | Warden log in |
| LAST REVISED | 26/11/2022 |
| DESCRIPTION | The warden logs in the system |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Database, student |
| PRECONDITION | The user is connected to the internet.  The user has already registered in the system as a warden. |
| POSTCONDITION | The warden successfully logs in. |
| MAIN SUCCESS SCENARIOS | 1. The warden clicks on the sign in button. 2. The warden enters his credentials 3. The warden clicks on the submit button. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a the warden enters invalid credentials.  3a. The warden forgets to click on the submit button. |

|  |  |
| --- | --- |
| ID | UC4 |
| USE CASE NAME | Verify warden login |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden’s credentials are verified from the database. |
| PRIMARY ACTOR | Database |
| SUPPORTING ACTOR | Admin, student, warden |
| PRECONDITION | The user is connected to the internet.  The warden is registered in the system.  The warden attempts to login to the system. |
| POSTCONDITION | The student is successfully logged In and can see the current status and facilities provided by the system. |
| MAIN SUCCESS SCENARIOS | 1. The warden’s id is verified. 2. The warden’s password is verified. 3. Logged in successfully. |
| EXTENSIONS | \*a. At any time, the system crashes. |
|  | 1a. The warden enters invalid email.  2a. the warden enters an invalid password.  3a. even after correct credentials, the warden is not logged in. |

**2. Student registration and login**



|  |  |
| --- | --- |
| ID | UC5 |
| USE CASE NAME | Student fills registration form |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student comes and registers himself in the system so he can live in the hostel. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Admin, Warden |
| PRECONDITION | The user is connected to the internet. |
| POSTCONDITION | The user successfully fills the registration form |
| MAIN SUCCESS SCENARIOS | 1. The student clicks on the registration button. 2. The student enters his information in the form for registration. 3. The student clicks the submit button. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. The student by mistake does not enter some information, incomplete form gives an error. 2b. The student enters some  wrong information. |

|  |  |
| --- | --- |
| ID | UC6 |
| USE CASE NAME | Verify student registration |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden checks if the student has filled the registration form correctly. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Admin, student |
| PRECONDITION | The user is connected to the internet.  The student has filled the form. |
| POSTCONDITION | The registration has been verified. |
| MAIN SUCCESS SCENARIOS | 1. The warden checks the details entered in the form. 2. The warden approves the registration. 3. The warden allocates a room to the student. 4. The warden adds the student information in the database. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. The admin fails to check the registration form correctly.  2a. The form is checked and is ok but the warden forgets to approve the registration.  3a. The warden forgets to allocate the room to the student.  3b. The warden allocates room which is already taken.  4a. The warden forgets to add the student in the database.  4b. the warden adds incorrect information in the database. |

|  |  |
| --- | --- |
| ID | UC7 |
| USE CASE NAME | Student Log in |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student logs in to the system to check the facilities provided by the hostel and check status. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Admin, warden |
| PRECONDITION | The user is connected to the internet.  The student is registered in the system. |
| POSTCONDITION | The student is successfully logged In and can see the status  and facilities provided by the system. |
| MAIN SUCCESS SCENARIOS | 1. The student enters the ID and password to login. 2. The student clicks on the login button. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. The student enters invalid log in credentials.  1b. the credentials are valid, but  the student is unable to log in. |

|  |  |
| --- | --- |
| ID | UC8 |
| USE CASE NAME | Verify log in |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The login credentials are verified from the database. |
| PRIMARY ACTOR | Database |
| SUPPORTING ACTOR | Admin, warden, student |
| PRECONDITION | The user is connected to the Internet.  The student is registered in the system.  The user has attempted to login. |
| POSTCONDITION | The student is successfully logged In and can see the status  and facilities provided by the system. |
| MAIN SUCCESS SCENARIOS | 1. The email is verified from the database. 2. The password is verified from the database. 3. If both matches, then the user is allowed to login. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. the email was incorrect. 2a. the password was invalid. |

#### **3.Manage hostel facilities**

#### 

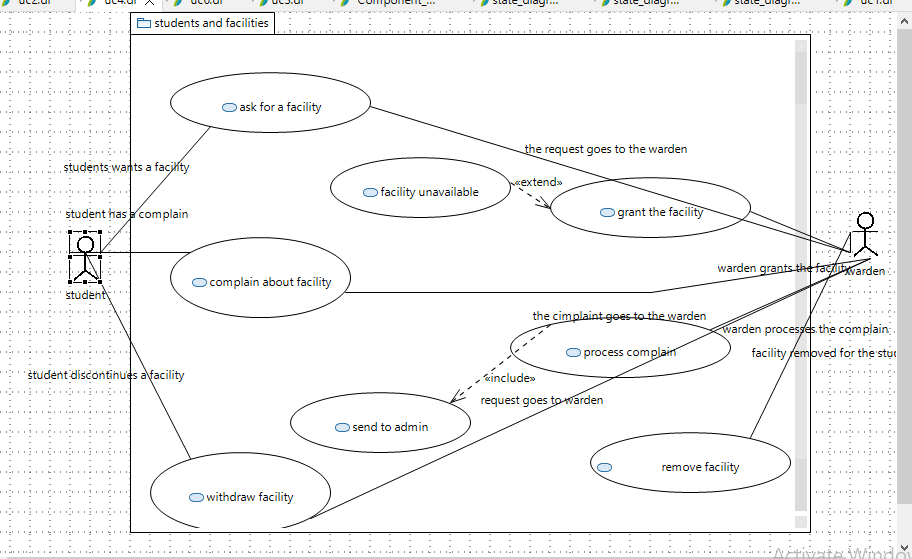
|  |  |
| --- | --- |
| ID | UC9 |
| USE CASE NAME | Display facilities |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The system displays the facilities a warden can provide in his hostel. |
| PRIMARY ACTOR | Database |
| SUPPORTING ACTOR | Student, Admin, warden |
| PRECONDITION | The system is connected to the internet |
| POSTCONDITION | The facilities will be displayed. |
| MAIN SUCCESS SCENARIOS | 1. The facilities button will be present on the front end. 2. The facilities will be displayed on the screen. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. No facilities are displayed on screen.  2b. incomplete facilities list is displayed on screen. |

|  |  |
| --- | --- |
| ID | UC10 |
| USE CASE NAME | Add facilities |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden adds the facilities that can be provided to the students. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Admin, student, |
| PRECONDITION | The user is connected to the internet. |
| POSTCONDITION | The warden successfully adds the facilities that his branch of hostel will have. |
| MAIN SUCCESS SCENARIOS | 1. The warden will click on the add facilities option. 2. The warden adds the facilities he wants from   the list of available. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. There is not a list of present. 2a. The warden tries to add the facilities that are not present on the table. |

|  |  |
| --- | --- |
| ID | UC11 |
| USE CASE NAME | Drop facilities |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden drops the facilities that were originally added. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Student, Admin |
| PRECONDITION | The user has already registered in the system.  The warden has added a few facilities for the hostel. |
| POSTCONDITION | The facilities will be deleted. |
| MAIN SUCCESS SCENARIOS | 1. The warden selects the facilities he wants to delete. 2. The facilities will no longer be available for the   students. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. The warden is trying to delete the facility that was never added.  2a. The facility is deleted but still is showing available. |

|  |  |
| --- | --- |
| ID | UC12 |
| USE CASE NAME | Facilities updated in the database |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The facilities added and deleted will be updated in the database. |
| PRIMARY ACTOR | Database |
| SUPPORTING ACTOR | Student, Admin, warden |
| PRECONDITION | The system is connected to the internet.  The warden adds and deletes facilities in their hostel. |
| POSTCONDITION | The data related to facilities will be updated in the database. |
| MAIN SUCCESS SCENARIOS | 1. The facilities that the warden adds are displayed in the database. 2. The facilities deleted will be updated in the   database too. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. the facilities that are added are not shown in the database.  1b. the facilities that are not added are shown in the database.  2a. The facilities that are deleted are shown in the database.  2b. the facilities that are not deleted will not be shown in the database/. |

**4.Student and facilities**



|  |  |
| --- | --- |
| ID | UC13 |
| USE CASE NAME | Asks for facility |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student asks for a particular facility provided by the hostel |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Warden, admin, database |
| PRECONDITION | The student has already registered in the system.  The student is logged in. |
| POSTCONDITION | The request is sent to the warden. |
| MAIN SUCCESS SCENARIOS | 1. The student applies for facilities he wants to avail. 2. The warden allows the facilities if available. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. The student has asked for a facility which is not available in the hostel. |

|  |  |
| --- | --- |
| Id | UC14 |
| USE CASE NAME | Grant facility |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden grants the facility asked by the student |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | student, admin, database |
| PRECONDITION | The warden has registered in the system  The warden is logged in The student has asked for a facility |
| POSTCONDITION | The student successfully gets the facility. |
| MAIN SUCCESS SCENARIOS | 1. The warden checks what facility is asked by the student. 2. The warden checks if that facility is regulated in the hostel. 3. If it is, then it is allowed to the student |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. The facility is not present in the hostel.  3a. The facility is present in the hostel but still the student doesn’t get it. |

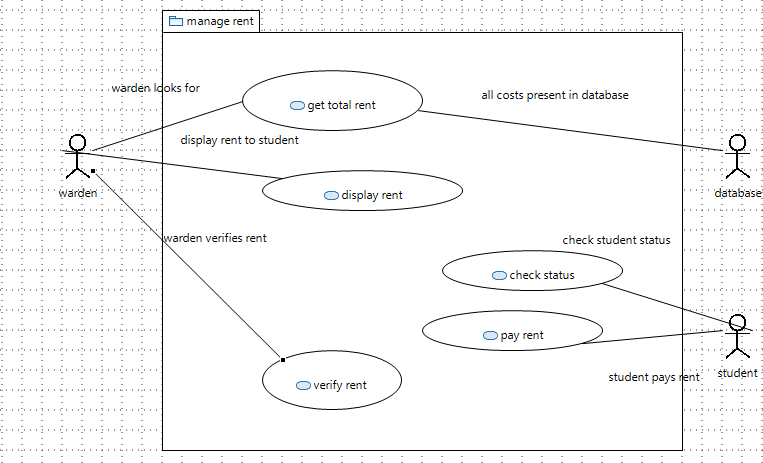
|  |  |
| --- | --- |
| ID | UC15 |
| USE CASE NAME | Complain about facilities |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student can complain about the facilities he is using if there are some problems in it. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Database, warden |
| PRECONDITION | The student is connected to the internet.  The student is logged in.  The student is already using that facility he is about to complain for. |
| POSTCONDITION | The complaint is successfully submitted |
| MAIN SUCCESS SCENARIOS | 1. The student clicks on the lodge complain button.  2 The student enters the facility about which he wants to complain.   1. The student writes down what is the exact issue with the facility. 2. The student submits the complaint. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. The student selects an item for complaint which is not currently in his use.  3a. the student writes invalid or incomplete description.  4a. The student forgets to submit the complaint. |

|  |  |
| --- | --- |
| ID | UC16 |
| USE CASE NAME | Processes the complain |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The complain about the facility is |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Warden, admin, database |
| PRECONDITION | The warden is logged in.  The student has complaint about a facility he is using. |
| POSTCONDITION | The complaint is forwarded to the admin. |
| MAIN SUCCESS SCENARIOS | 1. The warden receives a complaint about a facility the student was using. 2. The warden reads the complaint. 3. The warden forwards the complaint to the admin. |
| EXTENSIONS | \*a. At any time, the system crashes.  3a. the warden receives the complaint but forgets to forward it to the admin. |

|  |  |
| --- | --- |
| ID | UC17 |
| USE CASE NAME | Request to discontinue facility |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student wants to delete the facility he was using earlier. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Database, warden |
| PRECONDITION | The student is connected to the internet.  The student is logged in.  The student is already using that facility he wants to discontinue |
| POSTCONDITION | The facility is discontinued for the student. |
| MAIN SUCCESS SCENARIOS | 1. The student clicks on the facility he wants to discontinue. 2. The student clicks on the discontinues button. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. the student clicks on a facility he is not even using. |

|  |  |
| --- | --- |
| ID | UC18 |
| USE CASE NAME | Discontinue facility |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden discontinues the facility for a student. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Database, student, admin |
| PRECONDITION | The warden is connected to the internet.  The student has requested to discontinue the facility. |
| POSTCONDITION | The facility is discontinued for the  student |
| MAIN SUCCESS SCENARIOS | 1. The warden sees the student request to discontinue the facility. 2. The warden updates in the database \ 3. The facility is removed for that student. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. The warden discontinues the facilities for some other student by mistake.  2b. the warden forgets to discontinue the facility for that student. |

**5.Manages student fees**



|  |  |
| --- | --- |
| ID | UC19 |
| USE CASE NAME | Get total rent |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | Warden checks the room’s rent plus the facility rent for each  student from the database. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Student, database |
| PRECONDITION | The warden is connected to the internet.  The warden is logged in. |
| POSTCONDITION | The total amount is checked from the database. |
| MAIN SUCCESS SCENARIOS | 1. The warden checks the database. 2. The warden first sees the hostel’s rent for the student. 3. The warden then sees the rent of all the facilities the student is using. 4. The warden adds up the rent. |
| EXTENSIONS | \*a. At any time, the system crashes.  2a. the warden reads incorrect rent from the database.  3a. the warden reads incorrect facility rent from the database. 4a. the rent is added incorrectly. |

|  |  |  |
| --- | --- | --- |
| Id | UC20 |  |
| USE CASE NAME | Display rent |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden displays the total rent on screen. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Database, student |
| PRECONDITION | The warden is registered in the system.  The warden is logged in.  The warden has calculated the total bill. |
| POSTCONDITION | The warden has printed the rent on the screen. |  |
| MAIN SUCCESS SCENARIOS | 1. The warden displays the rent calculated on the  screen. |  |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. the wrong rent is displayed on screen.  1a. no bill is displayed on screen. |  |

|  |  |
| --- | --- |
| Id | UC21 |
| USE CASE NAME | Check status |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student will check his status to view his fee status. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Database, warden |
| PRECONDITION | The student is connected to the internet.  The student is logged in. |
| POSTCONDITION | The student is able to view his rent |
| MAIN SUCCESS SCENARIOS | 2. The student will view the fee details. |
| EXTENSIONS | \*a. At any time, the system crashes. |

|  |  |
| --- | --- |
| Id | UC22 |
| USE CASE NAME | Pay rent |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The student will pay the fees manually. |
| PRIMARY ACTOR | Student |
| SUPPORTING ACTOR | Database, warden |
| PRECONDITION | The student is connected to the internet.  The student is logged in.  The student has seen the rent he needs to pay/ |
| POSTCONDITION | The student pays the fees. |
| MAIN SUCCESS SCENARIOS | 1. The student pays the fees manually. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. the student does not pay the fees.  1b. the student pays the wrong fee. |

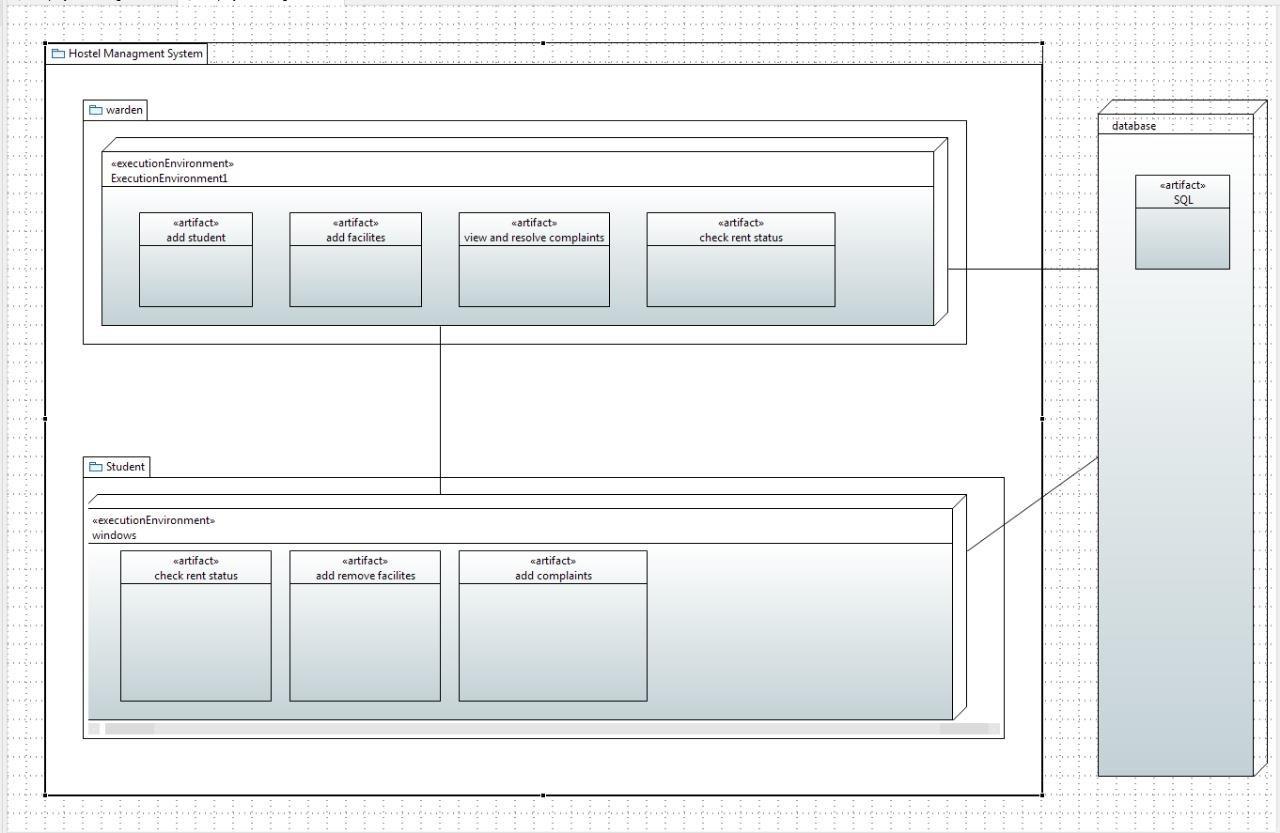
|  |  |
| --- | --- |
| ID | UC23 |
| USE CASE NAME | Verify rent |
| AUTHOR | Naimat, Nohail, Mohsin |
| LAST REVISED | 06/12/2023 |
| DESCRIPTION | The warden will verify whether the student has paid the rent or not. |
| PRIMARY ACTOR | Warden |
| SUPPORTING ACTOR | Database, student |
| PRECONDITION | The warden is connected to the internet.  The warden is logged in  The student has paid the fees. |
| POSTCONDITION | The warden has updated the database |
| MAIN SUCCESS SCENARIOS | 1. The warden checks manually if the student has paid the fees or not 2. The warden updates the database. |
| EXTENSIONS | \*a. At any time, the system crashes.  1a. the warden makes some mistakes in checking.  2a. the fee is paid but warden forgets to update in the database. |

## 

## **System Architecture:**

The hostel management system is done by both maintaining the frontend and the backend. There are two users in our system, the student, and the warden, for both of these users we need to maintain the whole system.

**System Level Architecture** (Deployment Diagram)

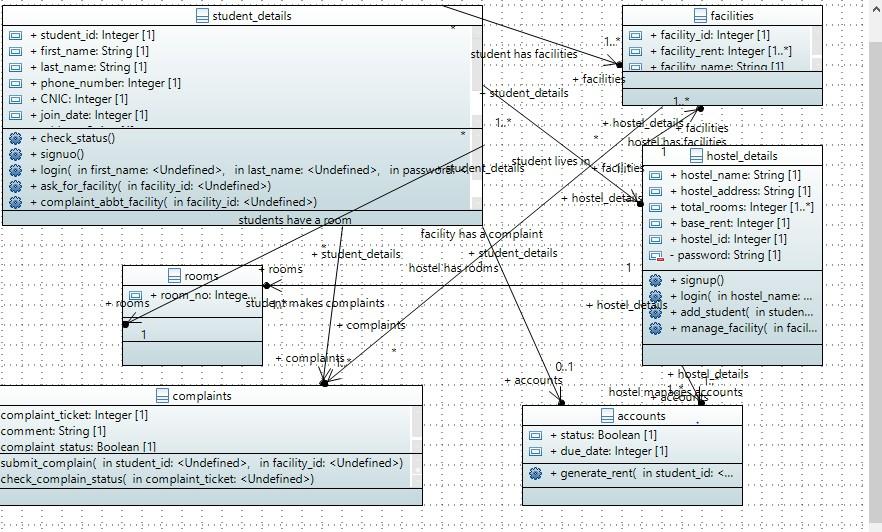
 **Software Architecture**

**Design Strategy**

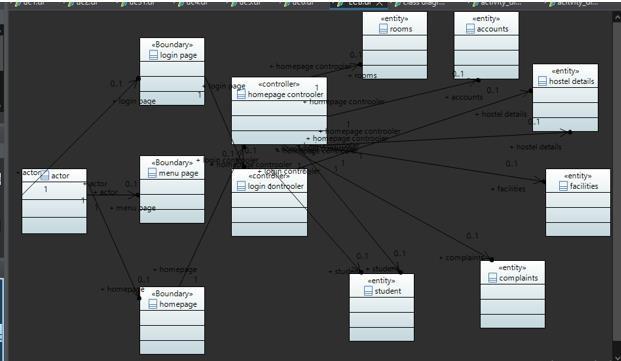
Beginning with the structure of the hostel management, the plan is to provide a portal for both student and warden to manage the hostel in a convenient way. In our design phase, we have

**Detailed System Design**

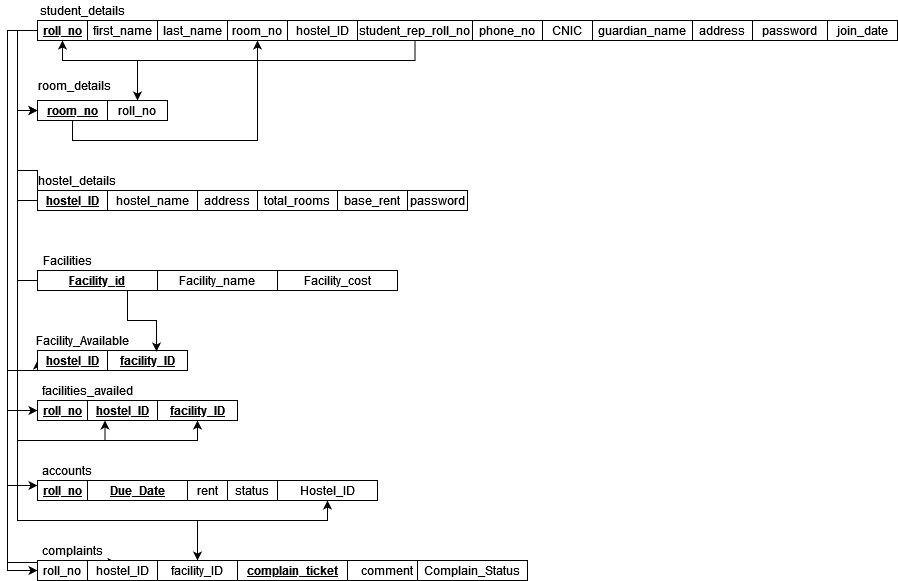
Class Diagram



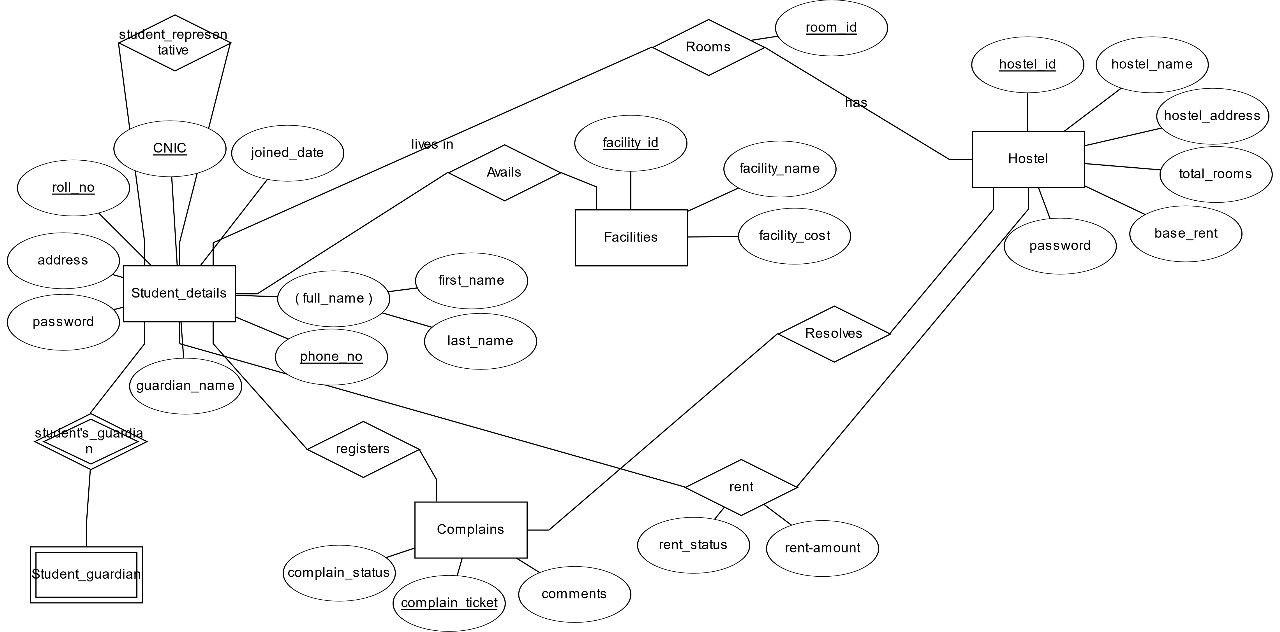
ECB Class Diagram



**Database Design**

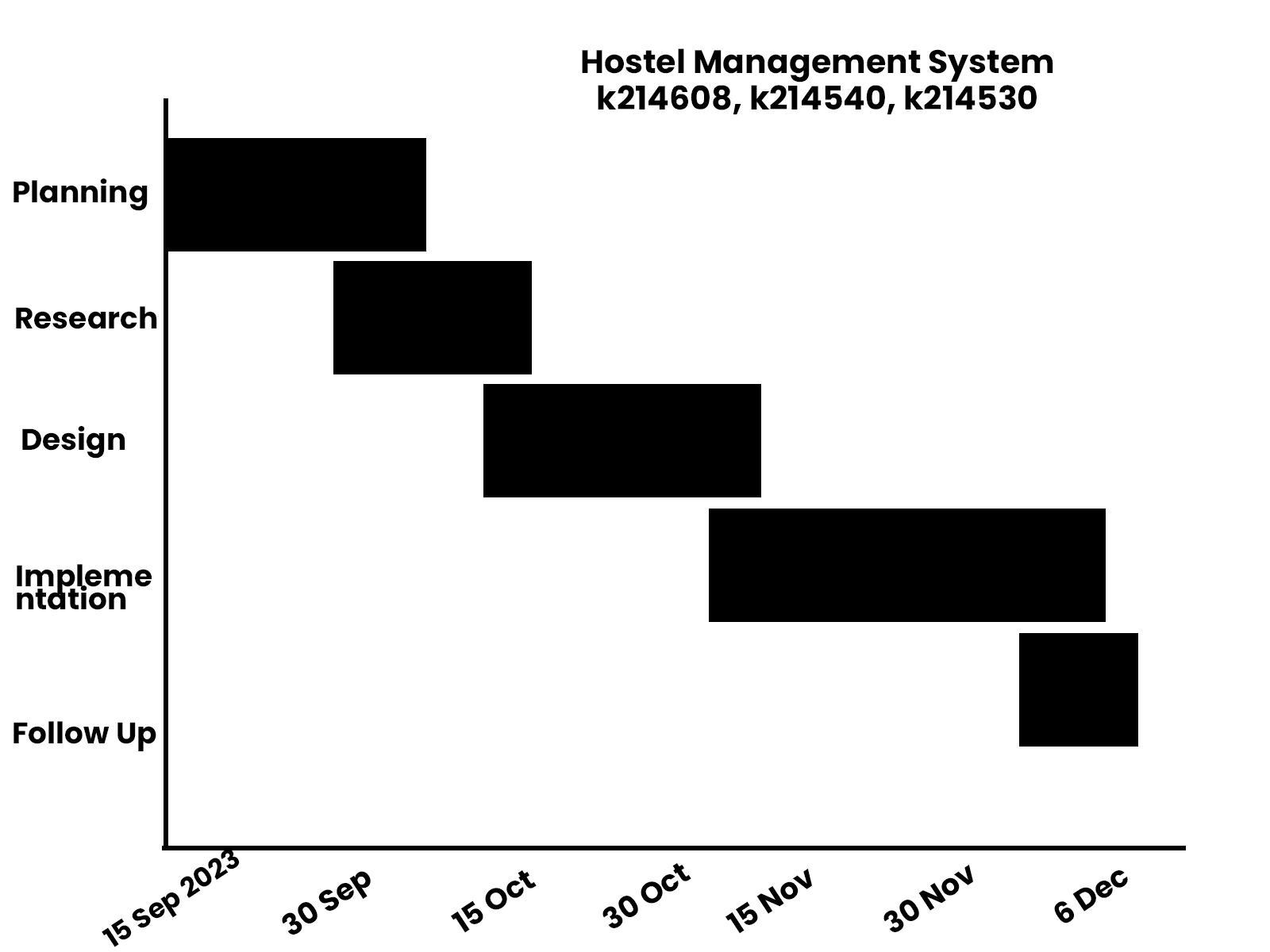


**ER Diagram**

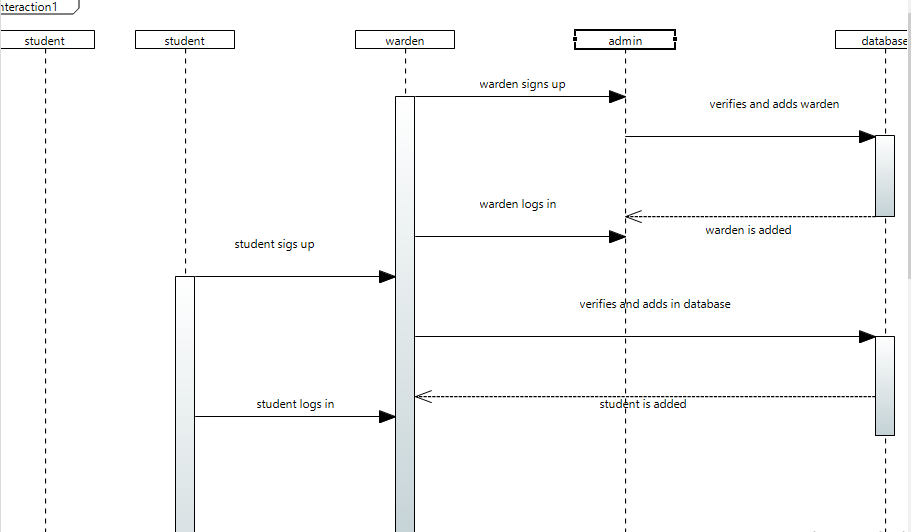


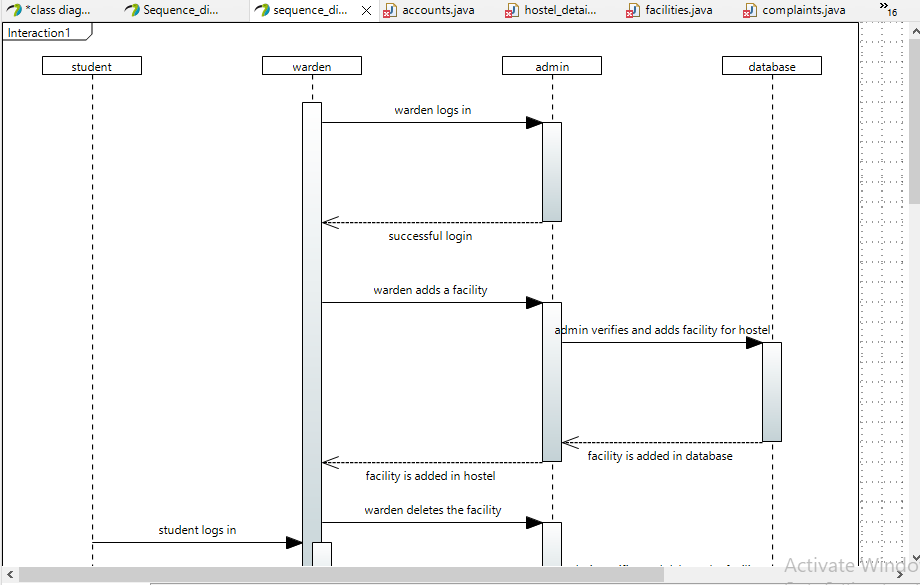
**Application Design**

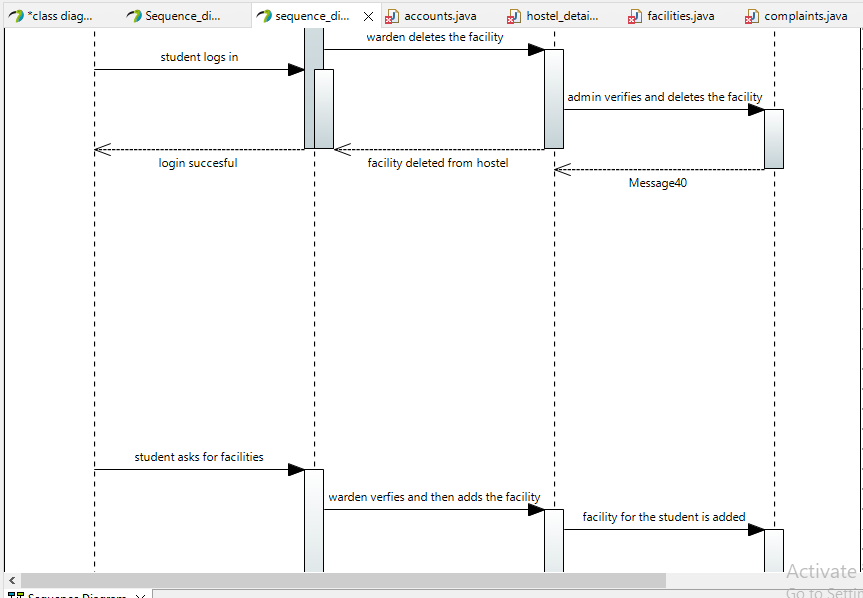
The application was designed in the following stages.

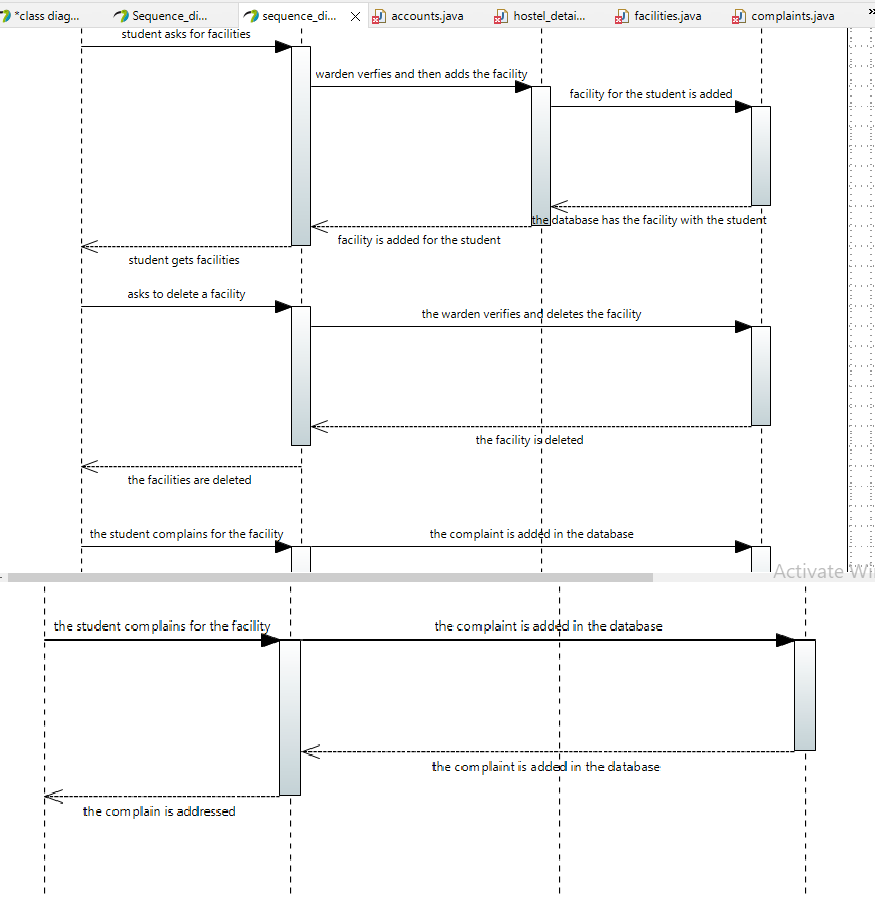


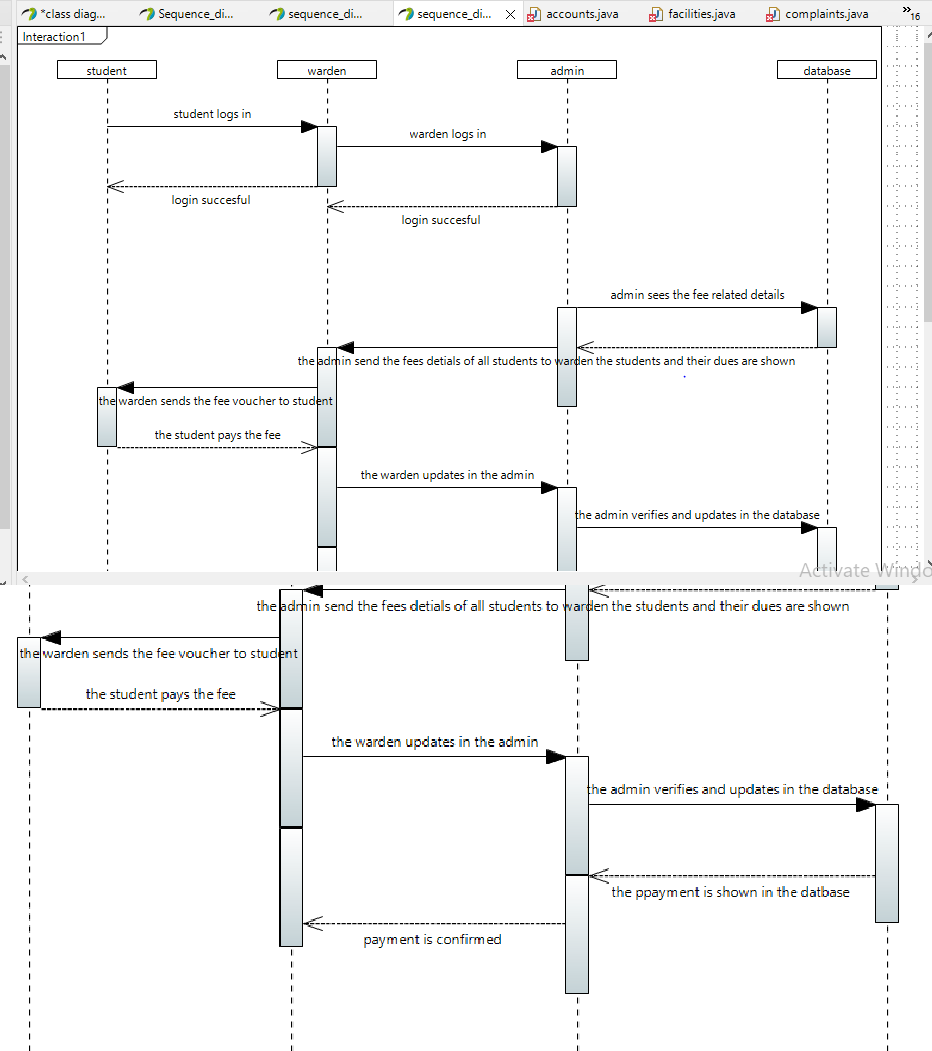
**Sequence Diagrams**



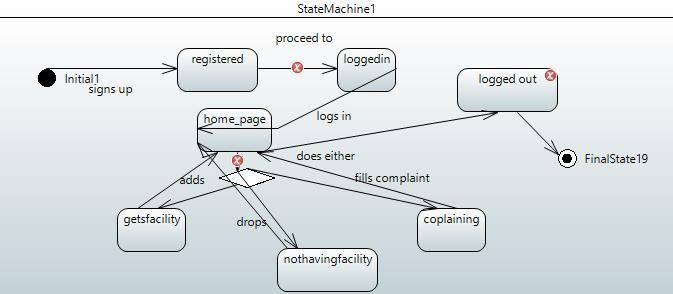


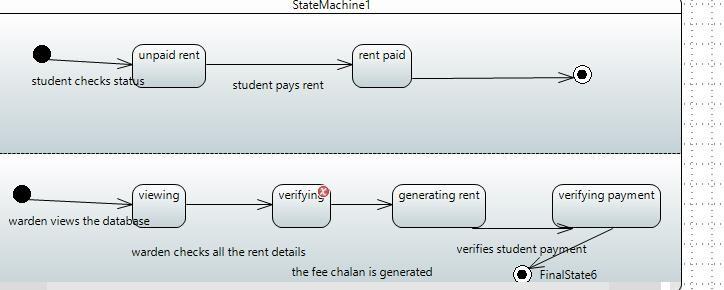


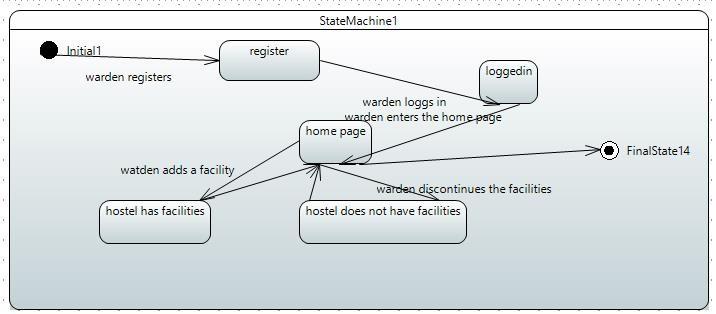




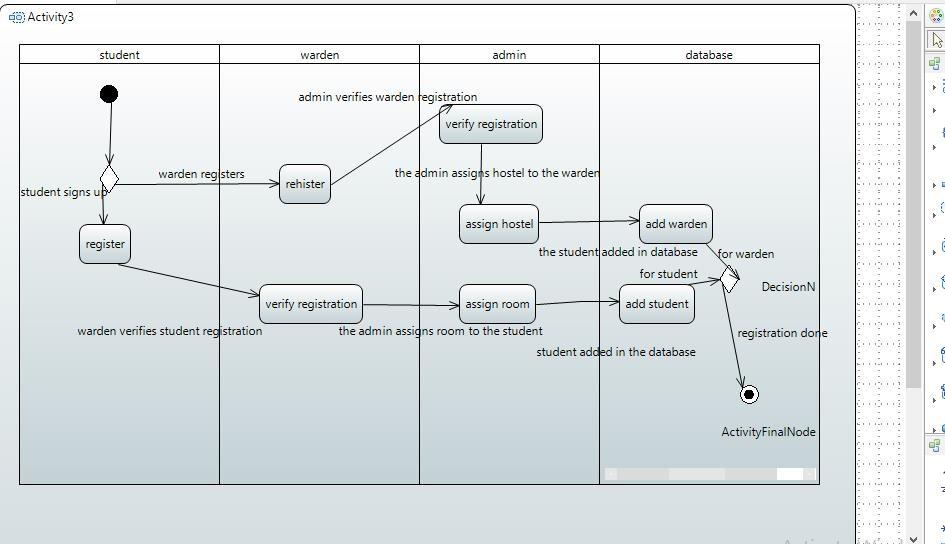
**State Diagrams**

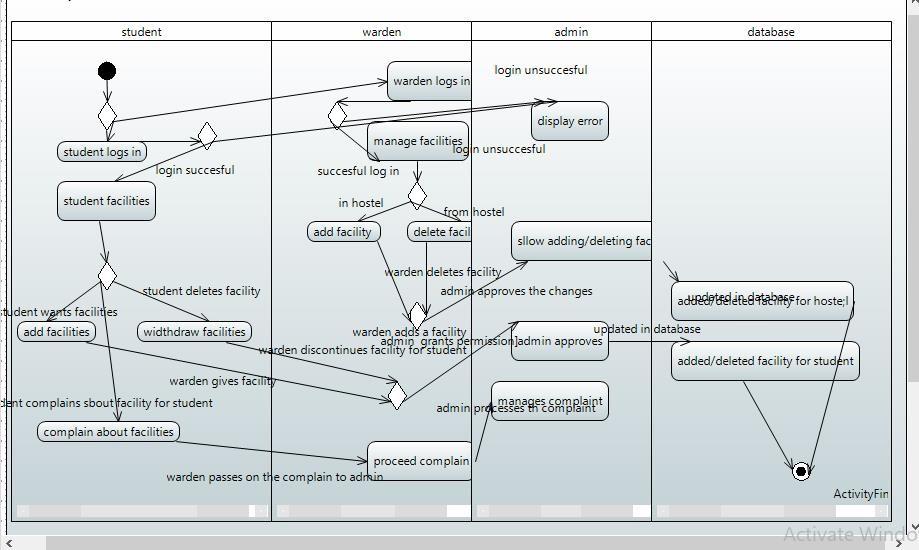


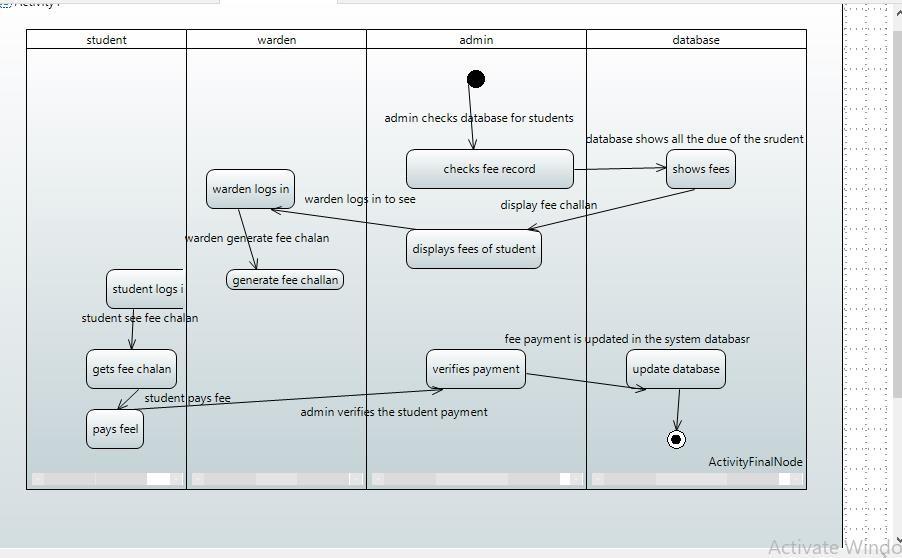




**Activity Diagrams**







**Conclusion:**

In conclusion, the Hostel Management System project undertaken by the BCS-5K group, led by Miss Romasha Khurshid, presents a comprehensive solution to streamline hostel operations at the university.

The detailed system architecture, deployment diagrams, and design strategies not only provide a clear roadmap for implementation but also underscore the depth of knowledge acquired during the course. As the project team, including Naimatullah Khan, Mohsin Khan, and Muhammad Nohail, presents their findings, it is evident that the semester's coursework has been successfully translated into a practical and impactful solution for hostel management

#### 