### Dormitory Management System

**By**

Ashfaqur Rahman Tokee (Exam Roll: 192339)

Md. Shakil Hossain (Exam Roll: 192340)

Mahabubur Rahman (Exam Roll: 192341)

Nahidul Islam (Exam Roll: 192345)

A project report submitted to the Institute of Information Technology

in partial fulfilment of the requirements for the degree of

Bachelor of Science in Information Technology

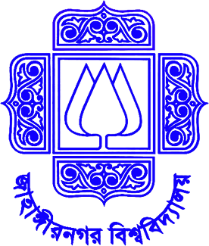
**Supervised By**

[**Fahima Tabassum**](https://juniv.edu/teachers/fahima)

Professor

Institute of Information Technology

Jahangirnagar University



Institute of Information Technology

Jahangirnagar University

Savar, Dhaka-1342

29 March, 2023

**DECLARATION**

This industrial tour report is submitted to the Institute of Information Technology, Jahangirnagar University, Savar, Dhaka in partial fulfillment of the requirements for having the B.Sc. (Hons.) degree in ICT. This is also needed to certify that the project work is under the 3rd Year 2nd Semester course of the IIT “ICT-3200: Project Work and Course Viva”. So, we are here declaring that this project report has not been submitted elsewhere for the requirement of any kind of degree, diploma or publication.

Ashfaqur Rahman Tokee Md. Shakil Hossain Mahabubur Rahman Nahidul Islam  
 192339 192340 192341 192345

CERTIFICATE

The project titled “Dormitory Management System” submitted by Ashfaqur Rahman Tokee (Exam Roll: 192339),Md. Shakil Hossain (Exam Roll: 192340), Mahabubur Rahman (Exam Roll: 192341), Nahidul Islam (Exam Roll: 192345), Session: 2018-2019, has been accepted as satisfactory in partial fulfillment of the requirement for the degree of Bachelor of Science in Information Technology on 30 March, 2023.

[**Fahima Tabassum**](https://juniv.edu/teachers/fahima)

Professor,

Institute of Information Technology,

Jahangirnagar University,

Savar, Dhaka – 1342.

Accepted and approved in partial fulfilment of the requirement for the degree

Bachelor of Science (honors) in Information Technology

BOARD OF EXAMINERS

|  |  |
| --- | --- |
|  |  |
| Dr. M Shamim Kaiser,  Professor,  Institute of Information Technology,  Jahangirnagar University,  Savar, Dhaka – 1342. | Chairman |
|  |  |
| Professor,  Institute of Information Technology,  Jahangirnagar University,  Savar, Dhaka – 1342. | Member |
|  |
|  |  |
| Professor,  Institute of Information Technology,  Jahangirnagar University,  Savar, Dhaka – 1342. | Member  (External) |

### DEDICATION

We dedicate this Dormitory Management System project to all the students who strive for a comfortable and efficient living experience in their college dormitories. Our goal was to develop a system that streamlines the management of dormitories, ensuring that students have easy access to essential services and facilities.

We also dedicate this project to the dormitory managers and administrators who work tirelessly to meet the needs of their residents. We hope that this system will help them to manage their duties more effectively, allowing them to provide a better quality of life for their students.

Finally, we would like to dedicate this project to our families and loved ones who have supported us throughout this endeavor. Their encouragement and belief in us have been instrumental in our success.

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| **Figure No** | **Figure Name** | **Page No** |
| Figure 2.2.1 | Home page of Dhaka University Hall | 3 |
| Figure 2.2.2 | Home page of Shahjalal University Hall | 4 |
| Figure 2.2.3 | Home page of KUET | 5 |
| Figure 3.2.1 | Diagram of Use Case | 7 |
| Figure 3.5 | Entity Relationship | 15 |
| Figure 3.6.1 | Context Level DFD | 16 |
| Figure 3.6.2 | 0 Level DFD | 16 |
| Figure 3.6.3 | 1 Level DFD | 17 |
| Figure 4.1.1 | Student Activity Diagram | 18 |
| Figure 4.1.2 | Admin Activity Diagram | 19 |
| Figure 4.2.1 | Login Operation of Admin | 20 |
| Figure 4.2.2 | Admin Operation Process | 21 |
| Figure 4.2.3 | Student Operation Process | 22 |
| Figure 4.3.5 | Home Page | 24 |
| Figure 4.3.6 | Our Hall Faculty | 24 |
| Figure 4.3.7 | Notice Area | 25 |
| Figure 4.3.8 | All Notice | 25 |
| Figure 4.3.9 | Developer Team | 26 |
| Figure 4.3.10 | Contact | 26 |
| Figure 4.4.4 | Login Page | 28 |
| Figure 4.4.5 | Admin Dashboard | 28 |
| Figure 4.4.6 | Add Faculty Members | 29 |
| Figure 4.4.7 | Faculty Member List | 29 |
| Figure 4.4.8 | Update Faculty Member Information | 30 |
| Figure 4.4.9 | New Admin Request | 30 |
| Figure 4.4.10 | All Admin List | 31 |
| Figure 4.4.11 | All Student List | 31 |
| Figure 4.4.12 | Update Student Information | 32 |
| Figure 4.5.13 | Add Student Part-1 | 32 |
| Figure 4.5.14 | Add Student Part-2 | 33 |

|  |  |  |
| --- | --- | --- |
| **Figure No** | **Figure Name** | **Page No** |
| Figure 4.5.15 | Admin Profile | 33 |
| Figure 4.5.16 | Update Admin | 34 |
| Figure 4.5.17 | Student Payment List | 34 |
| Figure 4.5.18 | Add Student Payment | 35 |
| Figure 4.5.19 | Add Building | 35 |
| Figure 4.5.20 | All Building List | 36 |
| Figure 4.5.21 | All Room List | 36 |
| Figure 4.5.22 | Add Room | 37 |
| Figure 4.5.23 | All Semester List | 37 |
| Figure 4.5.24 | Add Semester | 38 |
| Figure 4.5.25 | Add Notice | 38 |
| Figure 4.5.26 | Student Dashboard | 39 |
| Figure 4.5.27 | All Rooms | 39 |
| Figure 4.5.28 | Room Change Application | 40 |
| Figure 4.5.29 | Room Change Request | 40 |
| Figure 4.5.30 | Student Payment List | 41 |
| Figure 4.5.31 | Problem Request | 41 |
| Figure 4.5.32 | All Replies | 42 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| **Table No** | **Table Name** | **Page No** |
| Table 3.3.1 | Student Manage of Use Case | 8 |
| Table 3.3.2 | Building Manage of Use Case | 8 |
| Table 3.3.3 | Room Manage of Use Case | 9 |
| Table 3.3.4 | Available Room of Use Case | 9 |
| Table 3.3.5 | Update Payment of Use Case | 10 |
| Table 3.3.6 | Check Payment of Use Case | 10 |
| Table 3.3.7 | Create Notice of Use Case | 11 |
| Table 3.3.8 | View Notice of Use Case | 11 |
| Table 3.3.9 | Create Notice of Use Case | 12 |
| Table 3.3.10 | Problem of Use Case | 12 |
| Table 3.3.11 | Manage Apply of Use Case | 13 |
| Table 3.3.12 | Manage Semester of Use Case | 13 |
| Table 5.1.1 | Database Details of Admin | 43 |
| Table 5.1.2 | Database Details of Applies | 44 |
| Table 5.1.3 | Database Details of Building | 44 |
| Table 5.1.4 | Database Details of Members | 45 |
| Table 5.1.5 | Database Details of Notice | 45 |
| Table 5.1.6 | Database Details of Password Reset | 46 |
| Table 5.1.7 | Database Details of Payments | 46 |
| Table 5.1.8 | Database Details of Problems | 47 |
| Table 5.1.9 | Database Details of Replies | 47 |
| Table 5.1.10 | Database Details of Rooms | 48 |
| Table 5.1.11 | Database Details of Semesters | 48 |
| Table 5.1.12 | Database Details of Sliders | 49 |
| Table 5.1.13 | Database Details of Students | 49 |
| Table 5.1.14 | Database Details of Users | 50 |
| Table 5.2.1 | Login Page Test Case Description | 51 |
| Table 5.2.2 | Registration Page of Test Case Description | 52 |

## ACKNOWLEDGEMENT

First of all we would like to thank the Almighty for giving us the opportunity to complete this work successfully. Our acknowledgement is meant to express our sincere gratitude to all those people who have been associated with this project and have helped us with it and by sharing their experiences and valuable opinions through which we received the required information crucial for our project. We are thankful to our parents for their relentless support. Most importantly we are grateful to our honourable supervisor who took time out to guide us and provide us with all the necessary materials and sufficient knowledge that was the major requirement.

Finally, we convey our regards to our honourable teacher **Professor** [**Fahima Tabassum**](https://juniv.edu/teachers/fahima)Mam for giving us the opportunity to learn the subject particularly practically.

# ABSTRACT

The student hall/dormitory of a university is a main place to student’s daily life, so a hall/dormitory management is a vital part of the university management. But it’s a matter of regret that we have no digitalized management system available right now. Here everything controlled manually. That’s why authorities and students face a lot of problems. We don’t get enough information to allocate a seat of a student in the hall and manage room distribution. We have no digital record of payments of the student. In our current system, we solve our problems manually but we have no strong record, that’s why some problem is not solved in time. In this case, we want to propose an online hall/dormitory management system, which may help the student and authority to some extent.

TABLE OF CONTENTS

Declaration ……………………….…………………………………………………………………...…ii

Certificate ……………………………………………………………………………………………….iii

Dedication ………………………………………………………………………………………………iv

List of Tables and Figures ……………………………………………………………………………....v

List of Abbreviations……………………………………………………………………………... ……vi

Acknowledgement …………………………………………………………………………………….vii

Abstract ………………………………………………………………………………………….....….viii

[CHAPTER 1 INTRODUCTION 1](#_Toc130979283)

[1.1 Introduction 3](#_Toc130979286)

[1.2 Objectives 1](#_Toc130979284)

[1.3 Expected Outcome 1](#_Toc130979285)

[CHAPTER 2 BACGROUND 1](#_Toc130979283)

[2.1 Introduction 3](#_Toc130979286)

[2.2 Related Work 3](#_Toc130979287)

[2.2.1 Dhaka University Hall 4](#_Toc130979288)

[2.2.2 Shahjalal University Hall 4](#_Toc130979288)

[2.2.3 Khulna University of Engineering Technology 5](#_Toc130979289)

[CHAPTER 3 7](#_Toc130979290)

[3.1 Requirement Collection and Analysis 7](#_Toc130979291)

[3.1.2 Non-Functional System Requirements: 7](#_Toc130979292)

[3.2 Use Case Modeling 8](#_Toc130979293)

[3.3 Use Case Description 9](#_Toc130979294)

[3.4 Database Schema 15](#_Toc130979295)

[3.5 Diagram of Entity Relationship 16](#_Toc130979296)

[3.6 Diagram of DFD 17](#_Toc130979297)

[3.6.2 0 Level Diagram 17](#_Toc130979298)

[3.6.3 1 Level Diagram 18](#_Toc130979299)

[CHAPTER 4 SYSTEM DESIGN 19](#_Toc130979300)

[4.1 Diagram of Activity 19](#_Toc130979301)

[4.1.2 Admin 20](#_Toc130979302)

[4.2 Sequence Diagram 21](#_Toc130979303)

[4.2.2 Admin Operation Process 22](#_Toc130979304)

[4.2.3 Student Operation Process 23](#_Toc130979305)

[4.3 Front End Design 24](#_Toc130979306)

[4.4 Back End Design 28](#_Toc130979307)

[CHAPTER 5 IMPLEMENTATION AND TESTING 44](#_Toc130979308)

[5.1 Database Implementation 44](#_Toc130979309)

[5.2 Test Case 52](#_Toc130979310)

[CHAPTER 6 CONCLUSON AND FUTURE WORK 54](#_Toc130979311)

[6.1 Future Works 54](#_Toc130979312)

[6.2 Conclusion 54](#_Toc130979313)

[REFERENCES 56](#_Toc130979314)

* 1. **Introduction**

**CHAPTER 1**

# INTRODUCTION

A Dormitory Management System (DMS) is a software application designed to manage and automate the operations of dormitories or hostels. It's a centralized system that simplifies the management tasks of student boarding facilities, making it easier for administrators to manage students, rooms, inventory, and other aspects of running a dormitory.

The DMS typically consists of several modules that cater to different aspects of dormitory management, such as admissions, accommodation, meal plans, payments, maintenance, and security. The system may also include features such as online booking, room assignment, student profiles, attendance tracking, and communication tools, among others.

The primary objective of the Dormitory Management System is to simplify the management of a dormitory and streamline its operations. By automating routine tasks, reducing paperwork, and providing real-time information, the system helps administrators to save time, reduce errors, and enhance the overall experience of students living in the dormitory.

Dormitory Management Systems are used by educational institutions, such as universities, colleges, boarding schools, and hostels, as well as private organizations that provide student housing. With the growing demand for student housing and the increasing complexity of managing dormitories, DMS has become an essential tool for effective dormitory management.

### Objectives

The objectives of the “Dormitory Management System” are:

* + - To provide a digital management system for JU hall.
    - To reduce time for solving a problem.
    - To increase collaboration of student and authority.
    - To digitalize the problem gathering and solving system.
    - To digitalize the seat distribution system for JU hall.

### Expected Outcome

The expected outcomes of the “Dormitory Management System” are:

* + - A complete digital management system for JU Hall.
    - Digital means of storing student information.
    - Digitalized and Secure communication with authority and student.
    - Reduction of human resource and additional cost of JU Hall.

**CHAPTER 2 BACKGROUND**

### Introduction

To completely digitalize a university hall and providing online-based management system.

In this project, every single things will be automated (digital) and total activities will be complete using new technology. In our country, there are several universities. Our university is developing day by day, increasing with a number of students and hall rooms every semester. Our hall management authority faced troubles to share out room for student. Whereas, this was an lengthy procedure, so this spoiling a lot of time, diligent is not always capable to do the better used of the source. While we are living in the moment of technology, we need to apply for solve this problem.

We endeavor to solve the matter by execute a digital system and successfully we have done this. This project will help Hall authorities and students to locate rooms for students.

### Related Work

* + 1. **Dhaka University Hall**

Digital world is being digital by using information technology like many web-based automation system. A Web-based Computer Experiment Management System is designed and implemented by Dhaka University.

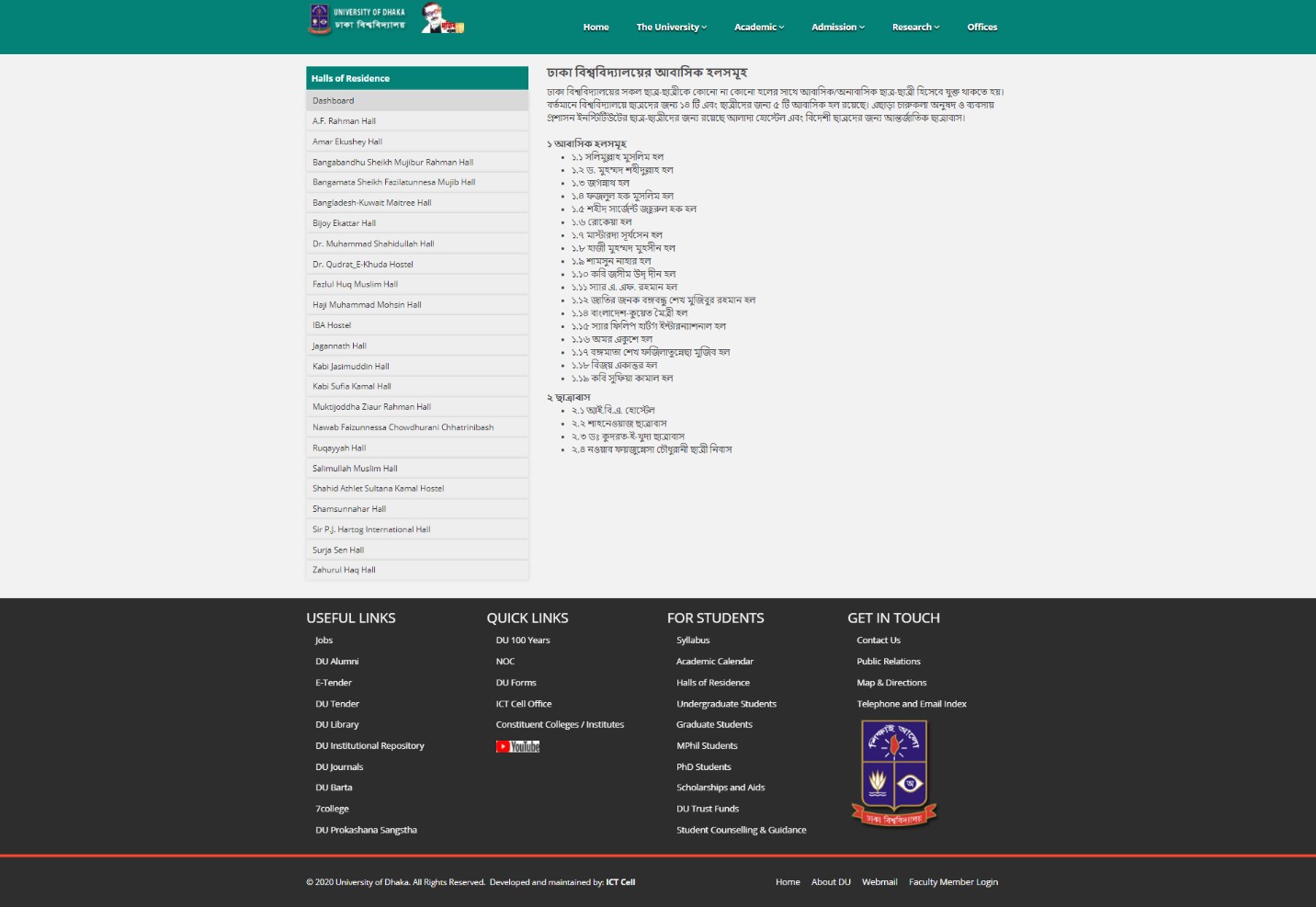


Figure 2.2.1 Home Page of Dhaka University

### Shahjalal University Hall

Digital World is being digital by using information technology like many web-based automation system. A Web-based Computer Experimental Management System is designed and implemented by Shahjalal University.

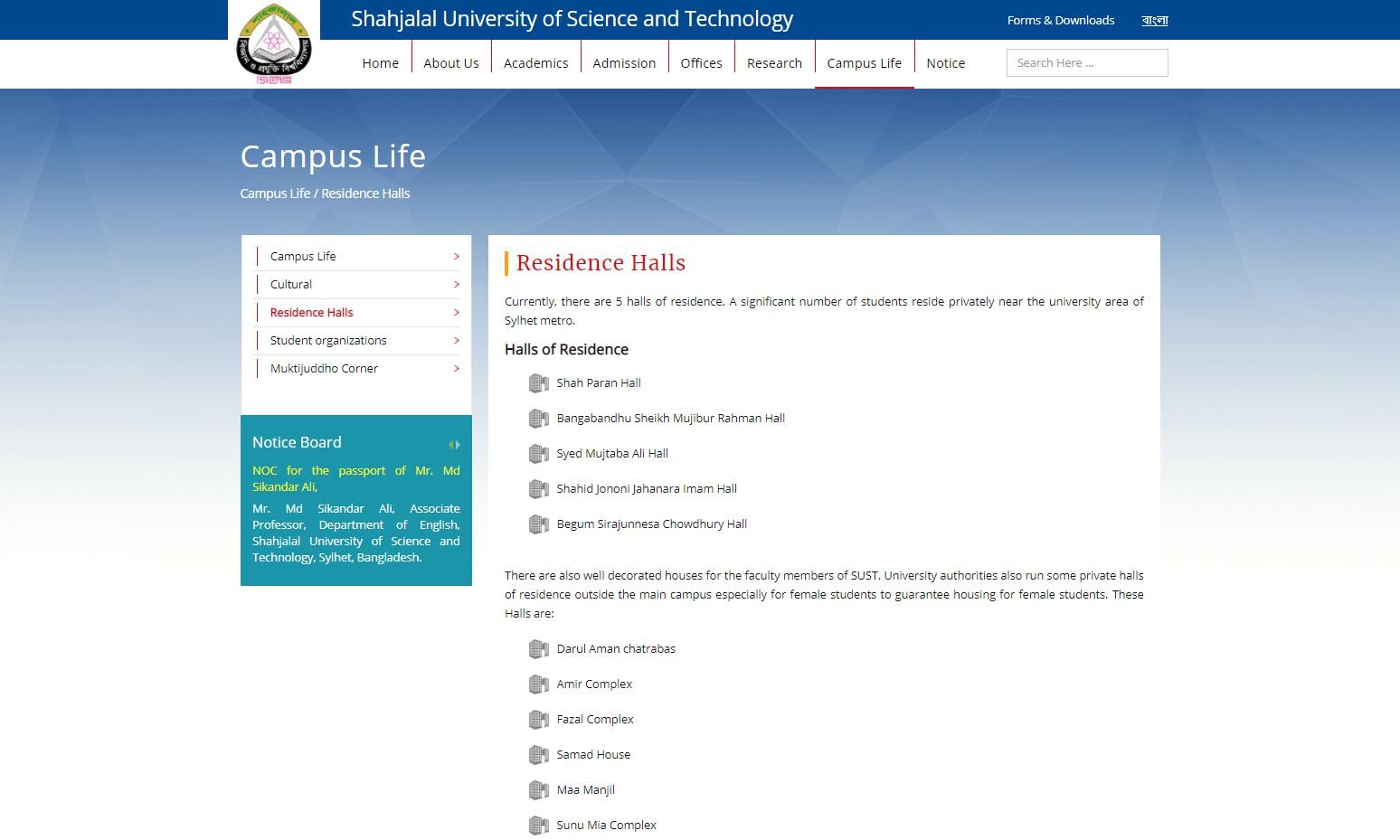


Figure 2.2.2 Home Page of Shahjalal University of Science and Technology

### Khulna University of Engineering Technology

Digital world being digital by using information technology like many web-based automation system. A Web-based Computer Experiment Management System is designed and implemented by Khulna University of Information and Technology.



Figure 2.2.3 Home Page of Khulna University of Engineering and Technology

## CHAPTER 3

**SOFTWARE REQUIREMENTS SPECIFICATION**

### Requirement Collection and Analysis

* + 1. **Functional System Requirement:**

This section gives a functional requirement that applicable to the “Dormitory Management System for JU Hall”.

These are sub modules in this phase.

* + - * Administrator module
      * User Module
      * Hostile Module
      * Registration Module

### Non-Functional System Requirements:

* Performance Requirements
* Security Requirements

### Use Case Modeling

#### Use Case Diagram

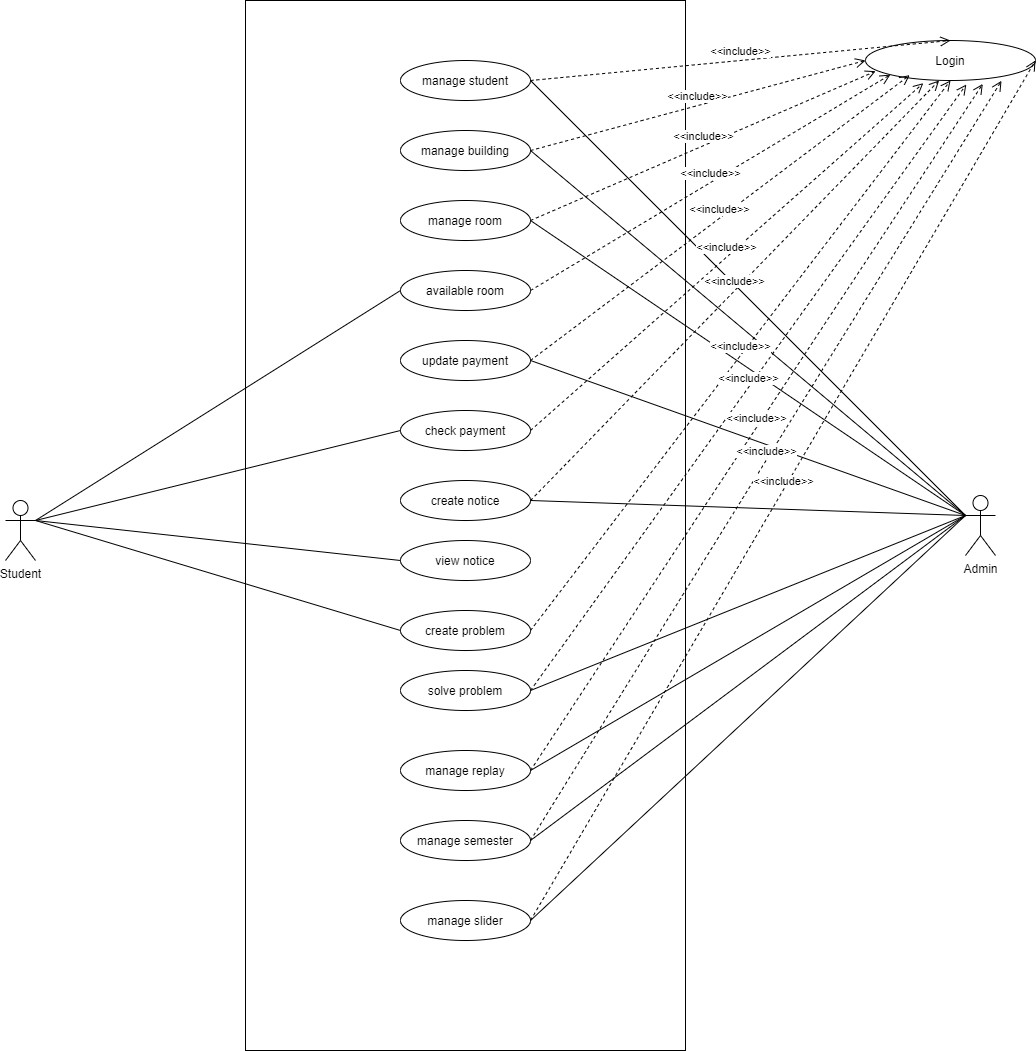


Figure 3.2.1 Use Case Diagram

### Use Case Description

TABLE 3.3.1 Student Manage

|  |  |
| --- | --- |
| **Use Case Term** | Student Manage |
| **Actors** | Admin |
| **Flow of Events** | 1. Affix to Student 2. Remove Student 3.View Student Details |
| **Substitute Flows** | 1. No student found 2. Do not add new student 3.Invalid Information |
| **Pre-Condition** | Login |
| **Post Condition** | Confirm Student, Delete Student |

TABLE 3.3.2 Building Manage

|  |  |
| --- | --- |
| **Use Case Term** | Building Manage |
| **Actors** | Admin |
| **Flow of Events** | 1. Affix to Building 2. Upgrade Building 3. Remove Building |
| **Substitute Flows** | 1. Chosen the wrong building 2. Building not found |
| **Pre-Condition** | Login |
| **Post Condition** | Chosen right building |

TABLE 3.3.3 Room Manage

|  |  |
| --- | --- |
| **Use Case Term** | Room Manage |
| **Actors** | Admin |
| **Flow of Events** | 1. Affix on Room 2. Upgrade Room 3.Remove Room |
| **Substitute Flows** | 1.Chosen false room 2.Delete incorrect room  3.Invalid Input |
| **Pre-Condition** | Login |
| **Post Condition** | Select the right room |

TABLE 3.3.4 Available Room

|  |  |
| --- | --- |
| **Use Case Term** | Available Room |
| **Actors** | Admin, Student |
| **Flow of Events** | 1.Check Available Room 2.Upgrade Room |
| **Substitute Flows** | 1. Chosen false room 2. Don’t updated available room 3.Invalid Input’s |
| **Pre-Condition** | Login |
| **Post Condition** | Select Building, Select Room |

TABLE 3.3.5 Update Payment

|  |  |
| --- | --- |
| **Use Case Term** | Update Payment |
| **Actors** | Admin |
| **Flow of Events** | 1.Affix to Payment 2.Upgrade Payment 3.Remove Payment |
| **Substitute Flows** | 1.Update wrong student payment 2.Don’t delete running student payment  3.Invalid Input’s |
| **Pre-Condition** | Login |
| **Post Condition** | Select correct student |

TABLE 3.3.6 Check Payment

|  |  |
| --- | --- |
| **Use Case Term** | Check Payment |
| **Actors** | Student |
| **Flow of Events** | 1. Check current payment 2. Check payment list |
| **Substitute Flows** | 1. Incorrect student id 2. Invalid Input’s |
| **Pre-Condition** | Login |
| **Post Condition** | Enter Login Information |

TABLE 3.3.7 Create Notice

|  |  |
| --- | --- |
| **Use Case Term** | Create Notice |
| **Actors** | Admin |
| **Flow of Events** | 1.Affix to Notice 2.Remove Notice |
| **Substitute Flows** | 1. Affix wrong notice 2. Invalid Information |
| **Pre-Condition** | Login |
| **Post Condition** | Enter notice title and description |

TABLE 3.3.8 View Notice

|  |  |
| --- | --- |
| **Use Case Term** | View Notice |
| **Actors** | Student |
| **Flow of Events** | 1.View current Notice 2.View previous Notice |
| **Substitute Flows** | 1. Select invalid notice 2. Can’t view future notice |
| **Pre-Condition** | Login |
| **Post Condition** | View notice panel |

TABLE 3.3.9 Create Problem

|  |  |
| --- | --- |
| **Use Case Term** | Create Problem |
| **Actors** | Student |
| **Flow of Events** | 1.Affix to Problem 2.Upgrade Problem 3.Remove Problem |
| **Substitute Flow** | 1. Incorrect student id 2. Invalid Information |
| **Pre-Condition** | Login |
| **Post Condition** | View problem list, view reply of admin |

TABLE 3.3.10 Solve Problem

|  |  |
| --- | --- |
| **Use Case Term** | Solve Problem |
| **Actors** | Admin |
| **Flow of Events** | 1.Reply |
| **Substitute of Flow** | 1. Can’t ignore problem 2. Invalid reply |
| **Pre-Condition** | Login |
| **Post Condition** | View problem solving list |

TABLE 3.3.11 Apply Manage

|  |  |
| --- | --- |
| **Use Case Term** | Apply Manage |
| **Actors** | Admin |
| **Flow of Events** | 1.Confirm Apply |
| **Substitute Flows** | 1. Don’t update previous apply information 2. Invalid Information Input |
| **Pre-Condition** | Login |
| **Post-Condition** | View Apply list |

TABLE 3.3.12 Semester Manage

|  |  |
| --- | --- |
| **Use Case Term** | Semester Manage |
| **Actors** | Admin |
| **Flow of Events** | 1.Affix to Semester 2.Remove Semester 3.Upgrade Semester |
| **Substitute of Flows** | 1.Affix false semester 2.search wrong semester  3.Invalid Information |
| **Pre-Condition** | Login, Create a new semester |
| **Post Condition** | View All Semester |

### Database Schema

Admin (A\_ID, Password)

Provost (P\_ID, Name, Email, Phone)

Staff (St\_ID, Name, Phone, Email, Post, Salary, Address)

Student (S\_ID, Name, Phone, Email, Registration, Batch, Department)

Room (R\_ID, Block, Floor, No of Bed, S\_ID)

Dinning (D\_ID, Meal Type, Meal Rate, Menu, S\_ID, St\_ID)

Hall Charge (H\_ID, Charge, A\_ID, S\_ID)

Facility (F\_ID, PR\_ID, ER\_ID, RR\_ID, S\_ID)

Prayer Room (S\_ID, PR\_ID)

Entertainment Room (ER\_ID, S\_ID, TV, Table Tennis)

Reading Room (RR\_ID, S\_ID)

### Diagram of Entity Relationship

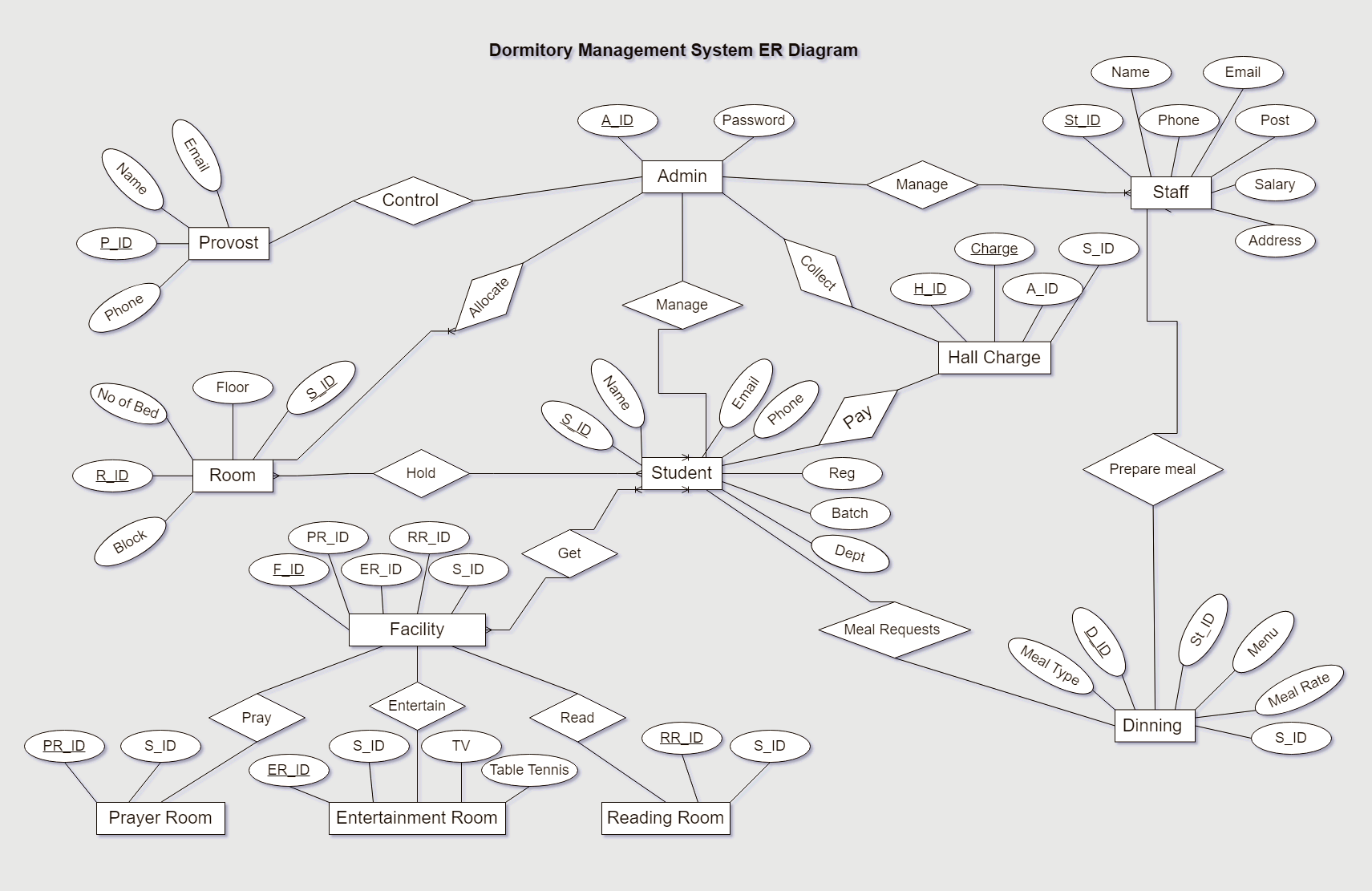
****

Figure 3.5 Entity Relationship

### Diagram of DFD

* + 1. **Context Level**



Figure 3.6.1 DFD Context Level

### 0 Level Diagram

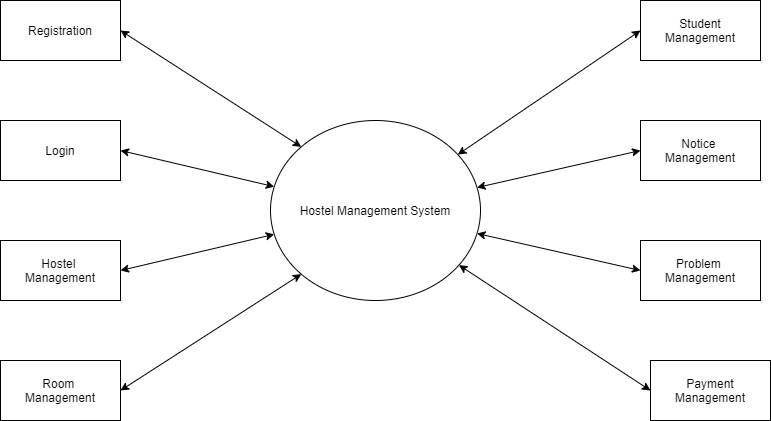


Figure 3.6.2 DFD 0 Level

### 1 Level Diagram

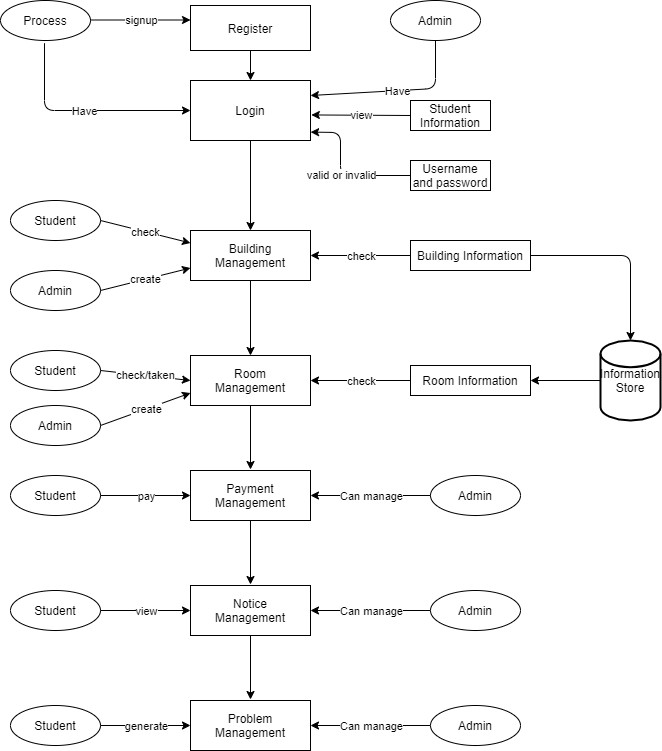


Figure 3.6.3 DFD 1 Level

## CHAPTER 4 SYSTEM DESIGN

### Diagram of Activity

* + 1. **Student**

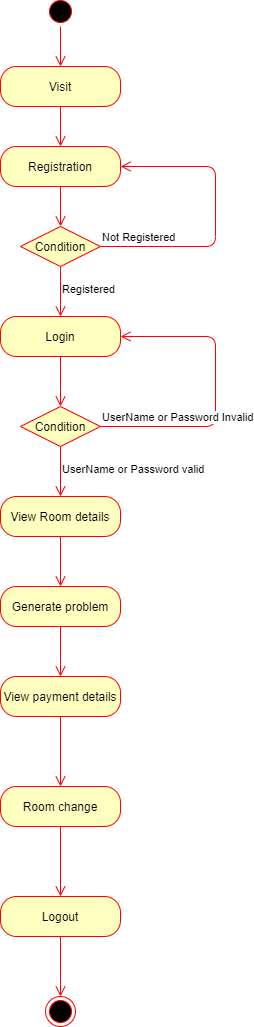


Figure 4.1.1 Student Activity Diagram

### Admin

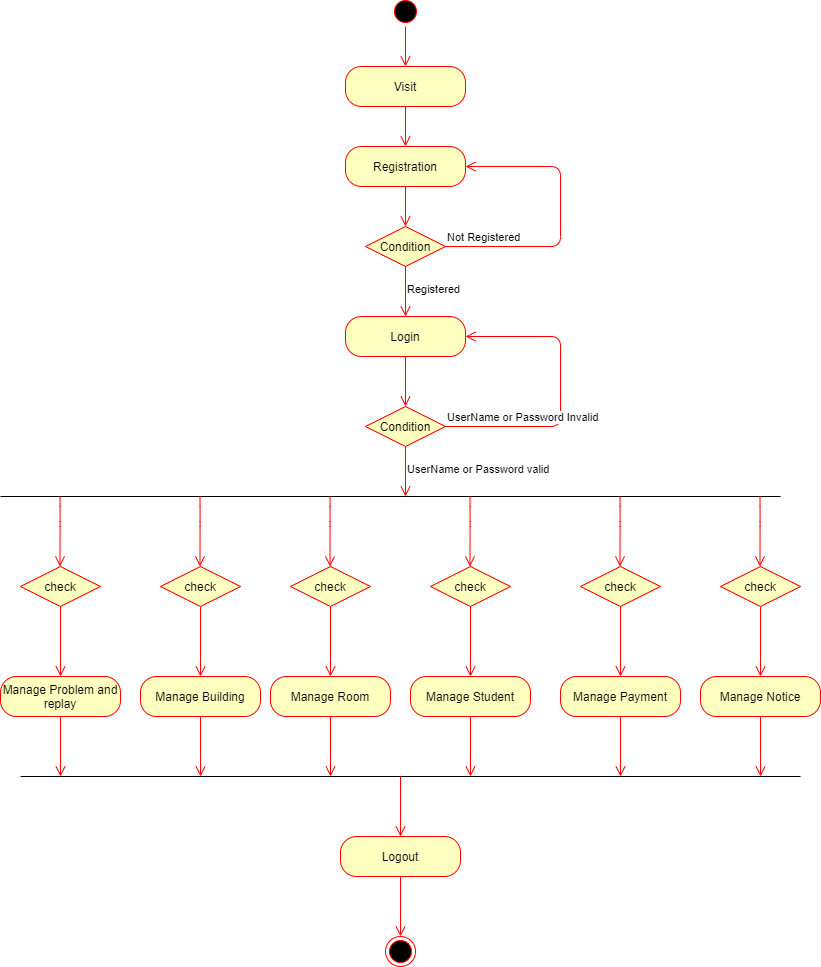


Figure 4.1.2 Admin Activity Diagram

### Sequence Diagram

* + 1. **Login Operation of Admin**

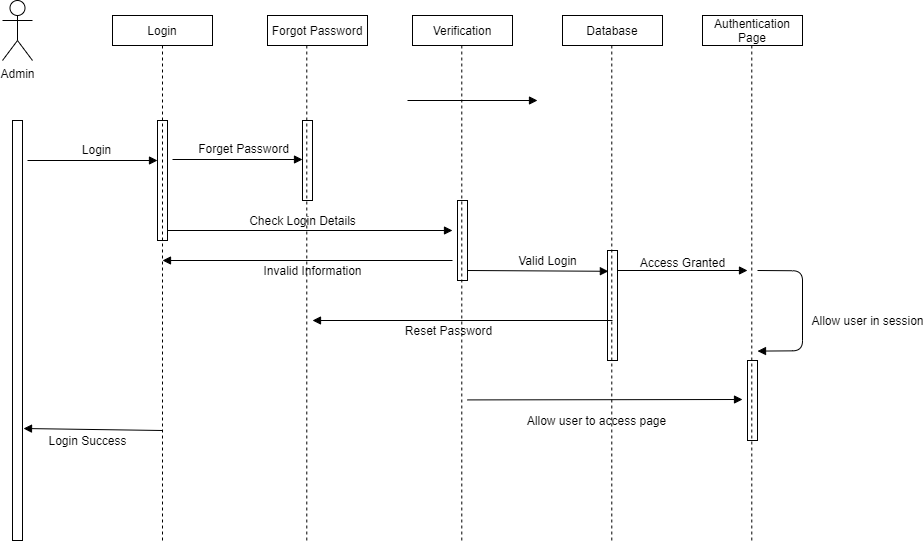


Figure 4.2.1 Login Operation of Admin

### Admin Operation Process

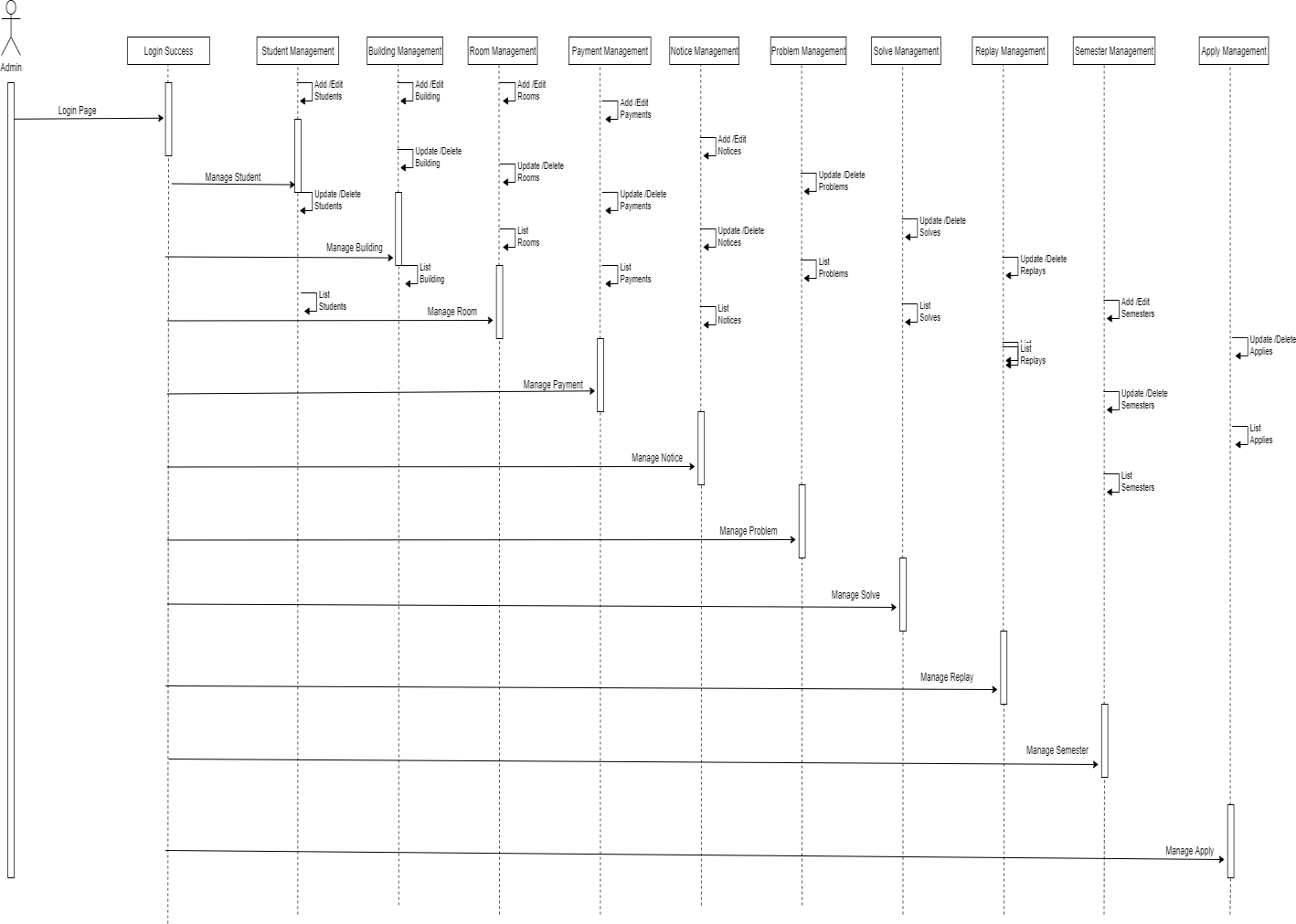


Figure 4.2.2 Admin Operation Process

### Student Operation Process

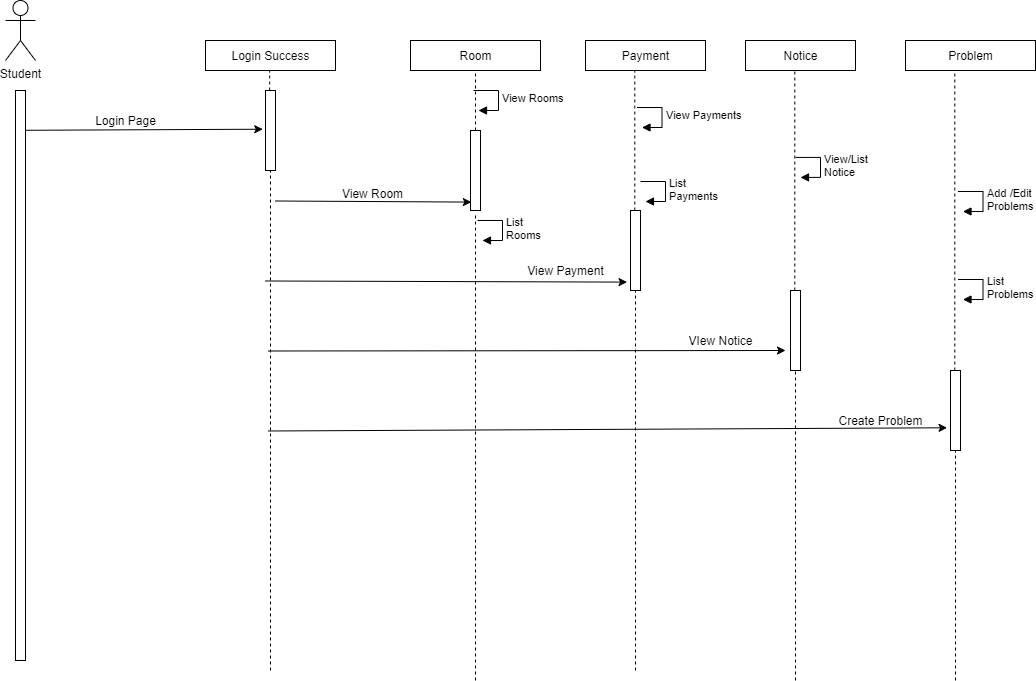


Figure 4.2.3 Student Operation Process

### Front End Design

I use different type of markup language, style sheet, Programming language and some libraries for both front-end and backend.

* HTML 5
* CSS
* Bootstrap4
* JavaScript

Figure 4.3.5 Home Page

Figure 4.3.6 Our Hall Faculty

Figure 4.3.7 Notice Area

Figure 4.3.8 All Notice

Figure 4.3.9 Developer Team

Figure 4.3.10 Contact

### Back End Design

I used some programing languages and frameworks to complete the back-end design of this website.

* PHP
* Laravel Framework
* MySQL

Figure 4.4.4 Login Page

Figure 4.4.5 Admin Dashboard

Figure 4.4.6 Add Faculty Members

Figure 4.4.7 Faculty Member List

Figure 4.4.8 Update Faculty Member Information

Figure 4.4.9 New Admin Request

Figure 4.4.10 All Admin List

Figure 4.4.11 All Student List

Figure 4.4.12 Update Student Information

Figure 4.4.13 Add Student Part-1

Figure 4.4.14 Add Student Part-2

Figure 4.4.15 Admin Profile

Figure 4.4.16 Update Admin Information

Figure 4.4.17 Student Payment List

Figure 4.4.18 Add Student Payment

Figure 4.4.19 Add Building

Figure 4.4.20 All Building List

Figure 4.4.21 All Room List

Figure 4.4.22 Add Room

Figure 4.4.23 All Semester List

Figure 4.4.24 Add Semester

Figure 4.4.25 Add Notice

Figure 4.4.26 Student Dashboard

Figure 4.4.27 All Rooms

Figure 4.4.28 Room Change Application

Figure 4.4.29 Room Change Request

Figure 4.4.30 Student Payment List

Figure 4.4.31 Problem Request

Figure 4.4.32 All Replies

## CHAPTER 5 IMPLEMENTATION AND TESTING

### Database Implementation

TABLE 5.1.1 Database Details of Admin

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Admin | | | | |
| Chart Statement | | This is chart container of Admin records | | | | |
| Area Name | Value Type | Volume | Not  Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| F\_name | Varchar | 250 | √ |  |  | Container contain first name of  admin |
| L\_name | Varchar | 250 |  |  |  | Container contain last name of  admin |
| Email | Varchar | 250 |  |  |  | Container contain email of  admin |
| Contact | Varchar | 250 |  |  |  | Container contain Contact of  admin |
| Email\_verified\_at | Timestamp |  |  |  |  | Container contain email  verification date of the admin |
| Password | Varchar | 250 |  |  |  | Container contain password of  admin |
| Is\_active | Int | 1 |  |  |  | Container contain active status  of admin |
| Remember\_token | varchar | 250 |  |  |  | Container contain remember  token of admin |

TABLE 5.1.2 Database Details of Applies

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Applies | | | | |
| Chart Statement | | This is chart container of Applies records | | | | |
| Area Name | Value Type | Volume | Not  Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| St\_application | Varchar | 250 | √ |  |  | Container contain student  application |
| St\_id | Int | 30 |  |  | √ | Container contain student id of  apply |

TABLE 5.1.3 Database Details of Building

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Building | | | | |
| Chart Statement | | This is chart container of Buildings records | | | | |
| Area Name | Value Type | Volume | Not  Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| Buiding\_name | Varchar | 250 | √ |  |  | Container contain building name  of building |

TABLE 5.1.4 Database Details of Members

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Members | | | | |
| Chart Statement | | This is chart container of members record | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id (Self  Increment) |
| Name | Varchar | 250 | √ |  |  | Container contain name of member |
| Email | Varchar | 250 | √ |  |  | Container contain email of  member |
| Position | Varchar | 250 | √ |  |  | Container contain position of  member |
| Contact | Int | 30 | √ |  |  | Container contain contact of  member |
| Image | Varchar | 250 | √ |  |  | Container contain image of  member |

TABLE 5.1.5 Database Details of Notice

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Notice | | | | |
| Chart Statement | | This is chart container of notice records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain id (Self  Increase) |
| Title | Varchar | 250 | √ |  |  | Container contain notice title of  notice |
| File | Varchar | 250 | √ |  |  | Container contain file of notice |

TABLE 5.1.6 Database Information Details of Reset

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Password Reset | | | | |
| Chart Statement | | This is chart container of password reset records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Email | VARCHAR | 250 | √ |  |  | Container contain email of  password reset |
| Token | VARCHAR | 250 | √ |  |  | Container contain token of  password reset |

TABLE 5.1.7 Database Details of Payments

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Payments | | | | |
| Chart Statement | | This is chart container of payments records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id (Self  Increase) |
| St\_id | Int | 30 | √ |  | √ | Container contain student id of  payments |
| St\_semester | Varchar | 250 | √ |  |  | Container contain student semester  of payments |
| Hall\_fee | Int | 30 | √ |  |  | Container contain hall fee of  payments |

TABLE 5.1.8 Database Details of Problems

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Problems | | | | |
| Chart Statement | | This is chart container of problems record | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| P\_description | Varchar | 250 | √ |  |  | Container contain problem  description of problems |
| St\_id | Int | 30 | √ |  | √ | Container contain student id of  problems |

TABLE 5.1.9 Database Details of Replies

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Replies | | | | |
| Chart Statement | | This is chart container of replies records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| St\_reply | Varchar | 250 | √ |  |  | Container contain student reply of  replies |
| Problem\_id | Int | 30 |  |  |  | Container contain problem id of  replies |

TABLE 5.1.10 Database Details of Rooms

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Rooms | | | | |
| Chart Statement | | This is chart container of rooms records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| Room\_num | Varchar | 250 | √ |  |  | Container contain room name of  rooms |
| Quantity | Int | 30 | √ |  |  | Container contain quantity of  rooms |
| Booked | Int | 30 | √ |  |  | Container contain booked status  of rooms |
| Building\_id | Int | 30 | √ |  | √ | Container contain building id of  rooms |

TABLE 5.1.11 Database Details of Semester

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Semester | | | | |
| Chart Statement | | This is chart container of semester records | | | | |
| Area Name | Value Type | Volume | Not  Null | PK | FK | Statement |
| Id | Int | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| Semester\_name | Varchar | 250 | √ |  |  | Container contain semester name  of semester |

TABLE 5.1.12 Database Details of Slider

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Slider | | | | |
| Chart Statement | | This is chart container of slider records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| Id | BIG (INT) | 30 | √ | √ |  | Container contain supreme id  (Self Increase) |
| Image | VARCHAR | 250 | √ |  |  | Container contain image of slider |

TABLE 5.1.13 Database Details of Student

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | Student | | | | |
| Chart Statement | | This is chart container of student records | | | | |
| Area Name | Value Type | Volume | Not Null | PK | FK | Statement |
| St\_id | Int | 250 | √ | √ |  | Container supreme id (Self  Increase) |
| St\_name | Varchar | 250 | √ |  |  | Container contain student  name of student |
| Email | Varchar | 250 | √ |  |  | Container contain email of  student |
| Password | Varchar | 250 | √ |  |  | Container contain password of  student |
| St\_dept | Varchar | 250 | √ |  |  | Container contain student  department of student |
| Image | Varchar | 250 | √ |  |  | Container contain mage of  student |
| Room\_id | Int | 30 | √ |  | √ | Container contain room id of  student |
| Semester\_id | Int | 30 | √ |  | √ | Container contain semester id  of student |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| St\_contact | Varchar | 250 | √ |  |  | Container contain student  contact of student |
| Remember\_token | Varchar | 250 | √ |  |  | Container contain remember  token of student |

TABLE 5.1.14 Database Details of User

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Chart Name | | User | | | | |
| Chart Statement | | This is chart container of user records | | | | |
| Area Name | Value type | Volume | Not Null | PK | FK | Statement |
| Id | int | 30 | √ | √ |  | Container supreme id (Self  Increase) |
| F\_name | varchar | 250 | √ |  |  | Container contain fast name  of user |
| L\_name | varchar | 250 | √ |  |  | Container contain last name  of user |
| Email | varchar | 250 | √ |  |  | Container contain email of  user |
| Contact | varchar | 250 | √ |  |  | Container contain contact of  user |
| Address | varchar | 250 | √ |  |  | Container contain address of  user |
| Email\_verified\_at | Timestamp |  |  |  |  | Container contain email  verified time of user |
| Password | varchar | 250 | √ |  |  | Container contain password  of user |
| Remember\_token | varchar | 250 | √ |  |  | Container contain remember  token of user |

### Test Case

TABLE 5.2.1 Login Page of Test Case Description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Serial No | Input/Action | Desired Value | Indeed Value | Comment |
| 1 | Permit the field  empty | The field email and  password required | Message “Email and  password is required” | Granted |
| 2 | Taken ineffective  Password | Password is incorrect | Message “Password is  incorrect” | Granted |
| 3 | Taken ineffective  email format | Please enter a valid  email | Message “Please enter a  valid email” | Granted |
| 4 | Taken acceptable  user name or email | Accepted Value | Value accepted | Granted |

TABLE 5.2.2 Registration Page of Test Case Description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Serial No | Activity | Desired Value | Indeed Value | Comment |
| 1 | Permit any field empty | This field is empty | Message “This field  is empty” | Granted |
| 2 | Taken an ineffective email or already used | This email is invalid or try with another | Message “Enter a valid email or try  with another” | Granted |
| 3 | Taken an ineffective  phone number | Phone number is invalid | Message “Phone  number is invalid” | Granted |
| 4 | Taken valid data | Accepted Value | Value accepted | Granted |

## CHAPTER 6

## CONCLUSON AND FUTURE WORK

### Future Works

We just want that our system will have been used in our university campus. Then in future if it is convenient to use by the users, we will try to make it as a versatile system. And the problems we have faced, in future we have planned to solve this problem. We have a well-planned idea about it. Cause it’s so important to build a JU Hall for educational system in our country. Most importantly we want to add payment gateway using API method in our project.

For the payment gateway, we will use:

* + - Bikash
    - DBBL Rocket
    - Nagad
    - One Card and more payment gateway

### Conclusion

Dormitory management systems are software tools that streamline the management of dormitories and student housing facilities. These systems typically include features such as room assignment and scheduling, facility maintenance and repairs, rent payment tracking, and communication tools for residents and staff.

Overall, dormitory management systems can greatly benefit both students and staff by increasing efficiency and reducing administrative workload. By automating many of the tasks associated with managing a dormitory, staff members can devote more time to providing support and resources to students.

Additionally, these systems often provide valuable data and analytics that can inform decision-making around facility management and student programming. For example, usage patterns and feedback from residents can help staff optimize resource allocation and improve the overall living experience for students.

In summary, dormitory management systems offer a variety of benefits for both students and staff, including improved efficiency, better communication, and data-driven decision-making. As such, they are increasingly becoming a standard tool for managing student housing facilities.

## REFERENCES

1. [*http://www.du.ac.bd/home/hall\_admin/hall*](http://www.du.ac.bd/home/hall_admin/hall) *(Dhaka University Hall Management Web Portal)*
2. [*http://www.sust.edu/campus-life/residence-hall*](http://www.sust.edu/campus-life/residence-hall) *(Shahjalal University of Science and Technology Hall)*
3. [*http://www.kuet.ac.bd/index.php/welcome/khajahalldetails*](http://www.kuet.ac.bd/index.php/welcome/khajahalldetails) *(Khulna University of Engineering and Technology)*