



Darshan University

A Project Report on

“Hostel Management System”

Under the subject

Software Engineering (2301CS405)

B. Tech, Semester – IV

Computer Science & Engineering Department

Submitted By

Student Name: Bhalodiya Tanisha

Enrollment No: 23010101021

Academic Year (2024-2025)

Internal Guide

Dean-DIET

Prof. R. B. Gondaliya

Dr. Gopi Sanghani

Darshan University

Darshan University

DECLARATION

We hereby declare that the SRS, submitted along with the **Software Engineering (2301CS405)** for entitled “**Hostel Management System**” submitted in partial fulfilment for the Semester-4 of **Bachelor Technology (B. Tech)** in **Computer Science and Engineering (CSE)** Department to Darshan University, Rajkot, is a record of the work carried out at **Darshan University, Rajkot** under the supervision of R. B. Gondaliya and that no part of any of report has been directly copied from any students’ reports, without providing due reference.

Tanisha Bhalodiya

Student’s Signature

Date: 23-12-2024



**Computer Science & Engineering
Department
Darshan University**

CERTIFICATE

This is to certify that the SRS on “**Hostel Management System**” has been satisfactorily prepared by **Tanisha Bhalodiya (23010101021)** under my guidance in the fulfillment of the course **Software Engineering (2301CS405)** work during the academic year 2024-2025.

Internal Guide
Prof. R. B. Gondaliya
Darshan University

Dean-DIET
Dr. Gopi Sanghani
Darshan University



**Computer Science & Engineering
Department
Darshan University**

ACKNOWLEDGEMENT

I wish to express my sincere gratitude to my project guide Prof. R. B. Gondaliya and all the faculty members for helping me through my project by giving me the necessary suggestions and advices along with their valuable co- ordination in completing this work I also thank my parents, friends and all the members of the family for their precious support and encouragement which they had provided in completion of my work. In addition to that, I would also like to mention the Darshan University personals who gave me the permission to use and experience the valuable resources required for the project from the University premises.

Thus, in conclusion to the above said, I once again thank the faculties and members of **Darshan University** for their valuable support in completion of the project.

Thanking You,

Tanisha Bhalodiya

ABSTRACT

Hostel management system is a system which aims in developing a computerized system to maintain all the daily work of hostel. This will make transfer hostel into digital hostel. This project has many features which are generally not available in general library system like facility of user login, room book , fees payment one click. It also has a facility of admin login through which the admin can monitor the whole system. This system will be designed with the basic features such as administrator can add/view/update/delete students' and rooms' details in it. It has also a facility where student after logging in their accounts can see their own details and update their details. They also can request admin for changing rooms. The administrator has rights to make and update student report , rooms report.

Main purpose of this system is to reduce human efforts as much as possible.

Table of Contents

List of Figures	I
List of Tables	II
1 Introduction	1
Product perspective	1
Product features.....	1
There are three different users who will be using this product	1
The features that are required for the Librarian are	1
The features that are required for the Member are	1
Functional Requirement	1
Administrator	1
Student	2
Warden.....	3
Non-Functional Requirement	3
Usability	3
Accuracy	3
Availability	3
Maintainability	3
2 Design and Implementation Constraints	4
Use case diagram	4
Activity diagram and Swimlane diagram.....	5
Sequence diagram.....	9
State diagram	10
Class diagram.....	11
Data flow diagram	12
Context diagram (level-0).....	12
DFD Level-1.....	12
DFD Level-2.....	13
3 External interface requirement (Screens)	14
Screen-1: Registration Form	14
Screen-2: Login Form	15
Screen-3: Add borrower book	16
Screen-4: Add borrower book	16
Screen-5: Add borrower book	16
4 Database design	17
List of Tables	17
5 Stories and Scenario	18

SRS – Hostel Management System

Story-1: Add new student in Hostel.....	18
Scenario# S1.1	18
Scenario# S1.2	18
Scenario# S1.3	19
Story-2: Manage hostel fees	19
Story-3: Assign room to student	19
6 Test cases.....	20
7 References	24

List of Figures

Figure 2.1-1 Use case diagram for hostel management system.....	4
Figure 2.2-1 Activity diagram for booking Room	5
Figure 2.2-1 Activity diagram for take leave.....	6
Figure 2.2-2 Swimlane diagram for Booking Room	7
Figure 2.3-1 Sequence diagram for Book Issue.....	8
Figure 2.4-1 State diagram of Book	8
Figure 2.4-2 State diagram for Librarian.....	9
Figure 2.5-1 Class diagram for Library management system.....	10
Figure 2.6-1 Context diagram for Library management system.....	11
Figure 2.6-2 DFD level-1 for Library management system.....	12
Figure 2.6-3 DFD level-2 for Issue book.....	13
Figure 3.1-1 Screen-1: Registration Form.....	14
Figure 3.2-1 Screen-2: Login Form.....	15
Figure 3.3-1 Screen-3: Add borrower book.....	12

List of Tables

Table 3.1-1 Screen element of Registration form	10
Table 3.2-1 Screen element of Login form	11
Table 3.3-1 Screen element of Add borrower book	12
Table 4.1-1 Table: Book	13
Table 4.1-2 Table: Borrower	13
Table 4.1-3 Table: Staff	13
Table 4.1-4 Table: Student	13

1 Introduction

1.1 Product perspective

This project is basically updating the manual hostel system into an internet-based application so that the users can know the details of their account, availability of rooms etc. It is a multi-user version and can take care of all the fundamental functions of a Hostel like Cataloguing, Circulation, Accessioning and Housekeeping. It can satisfactorily cater to all the basic functions of a small library.

1.2 Product features

1.2.1 There are three different users who will be using this product:

- Librarian who will be acting as the administrator.
- Member who will be accessing the library.
- Guest who will request for membership.

1.2.2 The features that are required for the Librarian are:

- Control the movement of books and other material and avoid losing the same.
- Search if you have a specific book in your collection based on the title, author etc.
- Print the spine labels for the book.
- Find what a specific person has borrowed from you. It offers the following modules Cataloguing, Circulation, and Queries.
- Can issue a book to the student.
- Can view the list of books available in each category.
- Can take the book returned from students.
- Add books and their information of the books to the database.
- Edit the information of the existing books.
- Can check the report of the issued Books.
- Can access all the accounts of the students.

1.2.3 The features that are required for the Member are:

- Can view the different categories of books available in the library.
- Can view the List of books available in each category.
- Can own an account in the library.
- Can view the books issued to him.
- Can put a request for a new book.

1.3 Functional Requirement

1.3.1 Student

- Registration : Register for self to get admin in hostel.
- Update Details : User can update their details which he or she entered while admission.
- Checking Rooms : User can see how many rooms are available in whole hostel.
- Checking facilities : User can check facilities of rooms available in hostel.
 - Checking cleanliness : User can check facilities and how clean are rooms of hostel.
 - Hiring Request : use request to admit for having admission in hostel.

- Profile request : User can update their own details or personal details if there will be any change in their details.
 - Review : User can review the hostel after visiting and can rate the rooms facilities.
 - Visiting : User can visit the hostel for making sure about it.
- Event Registration : If hostel organize the event then student can register in that particular event.
- Event Notification : Student get message about the event which is organized.
- Choosing Room : Student can choose the type of room available in hostel like AC room and Non-AC room.
- About Parents : Student will give information about their guardians.
- Room Changing Request : If student want to change room for some valid reason then they will request aamin for changing room.
- Room Details : User can view the room details.
- Adding Health Information : Student will add their information about health if there is some huge problem then it will be noticed.
- Fee Payment : Student pay fees of hostel in many different ways in online.

1.3.2 Administrator

- Login : Registered student will login into hostel website or application.
- Room Allocation : Admin will assign room to student with their choice and need.
- Check-in and check-out: When student enter into hostel admin will note time of coming back to hotel.
- Maintaining Room: Admin will maintain room and check every information about that.
- Student Authentication: Admin can authenticate student after their registration.
- Reporting : Admin report and check all the details of student after some certain time duration.
- Add Student: Admin add student which is newly admitted in hostel.
- Update Student: Admin update student which is already exists.
- Delete Student: Admin can delete student and wrong entry must be delete.
- Handle Information of Hostel: Admin check information of rooms of hostel and facilities.
- Hiring Warden : admin will hire warden for hostel.
- Student Status management : admin manage details and information of student.
- Room Maintenance : Room details will be maintain by admin.
- Fees Status : Admin will inform student for fees time and show paid fees status of student.
- Event organize : To organize event admin has this responsibilities.
- Hostel information : To having information and all details about hostel will be manage by admin.
- Handle Hostel staff : Admin will maintain the details of staff.
- Assign warden : If needed new warden will be assign admin.
- Remove Warden : Warden will be removed by admin.
- Exchange Warden : Warden will be exchange by admin.
- Staff Schedule : admin will decide schedule for each member for good time management. Check Account: This use case is used to check account detail

1.3.3 Warden

- Check Student Detail: Warden take care of students in every manner.
- Connecting with parents : To connect with parents warden helps student to talk and meet their parents.
- Tracking Student : Warden has all the information about student where they are going so that they can maintain safety.
- Controlling Student details : Having all the details of student.
- Hiring Cook : Warden hire cook for hostel.
- Handle Cook details : To having all the details of cook.
- Check Healthiness of cook : Warden keep checking of quality of cook.
- Check all groceries : Warden will list and have full stock of groceries.
- Hire Worker or cleaning staff : Hire worker and cleaning staff.
- Tracking information of rooms : Warden have all the information of rooms cleanliness and all the stuff.

1.4 Non-Functional Requirement

1.4.1 Usability:

- The UI should be simple enough for everyone to understand and get the relevant information without any special training. Different languages can be provided based on the requirements.

1.4.2 Accuracy:

- The data stored about the books and the fines calculated should be correct, consistent, and reliable.

1.4.3 Availability:

- The System should be available for the duration when the library operates and must be recovered within an hour or less if it fails. The system should respond to the requests within two seconds or less.

1.4.4 Maintainability:

- The software should be easily maintainable and adding new features and making changes to the software must be as simple as possible. In addition to this, the software must also be portable.

2 Design and Implementation Constraints

2.1 Use case diagram

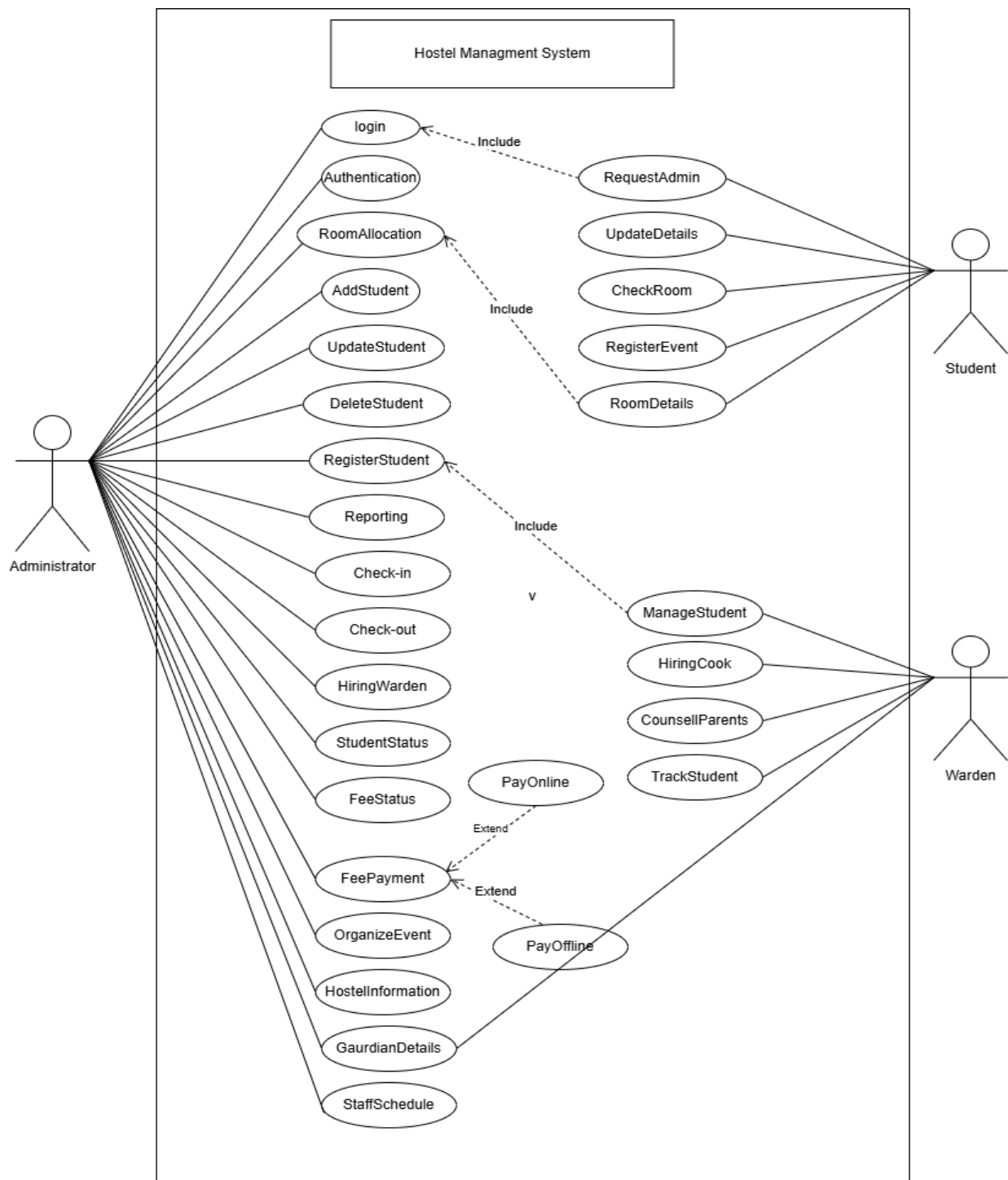


Figure 2.1-1 Use case diagram for hostel management system

2.2 Activity diagram and Swimlane diagram

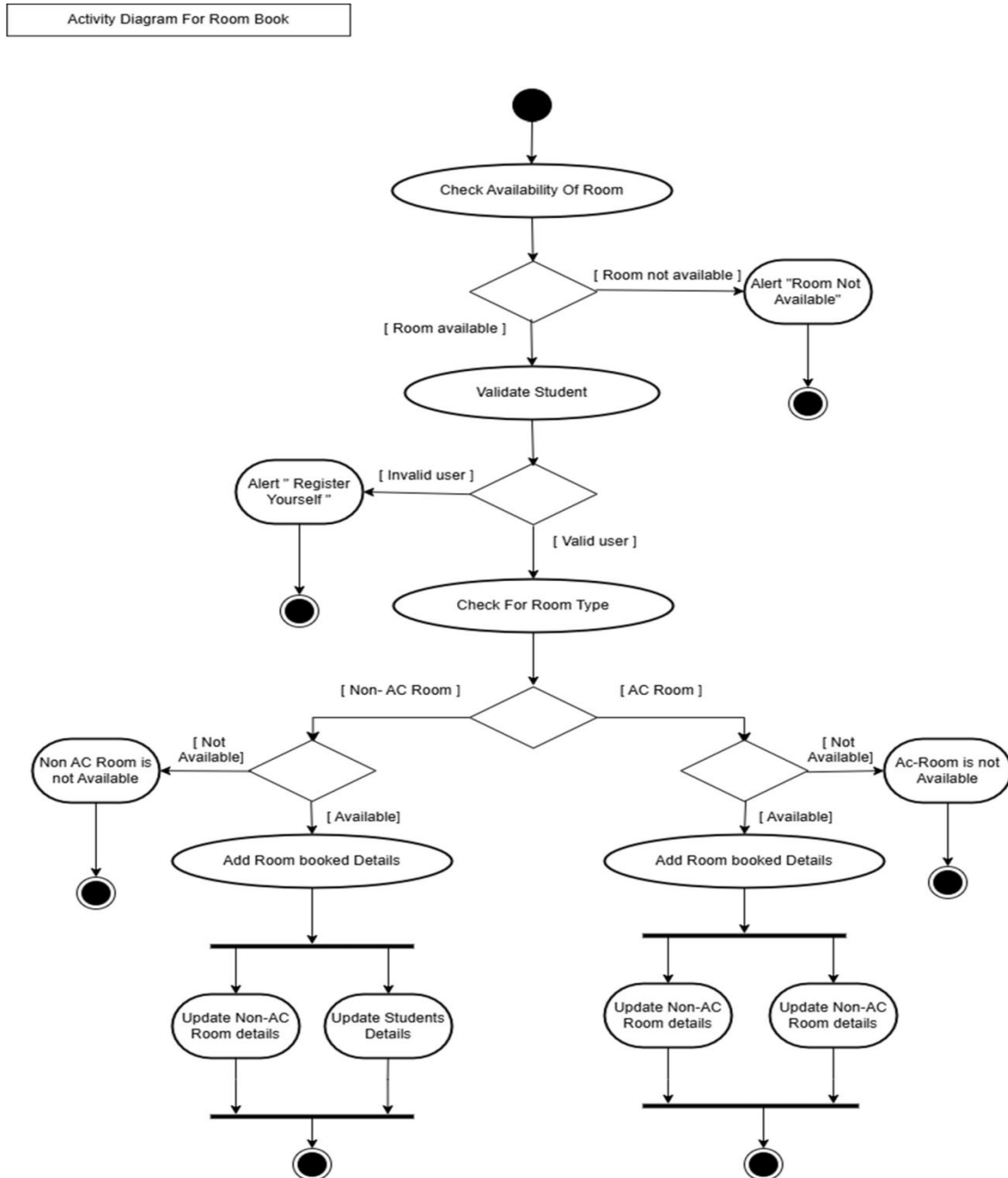


Figure 2.2-1 Activity diagram for Booking Room

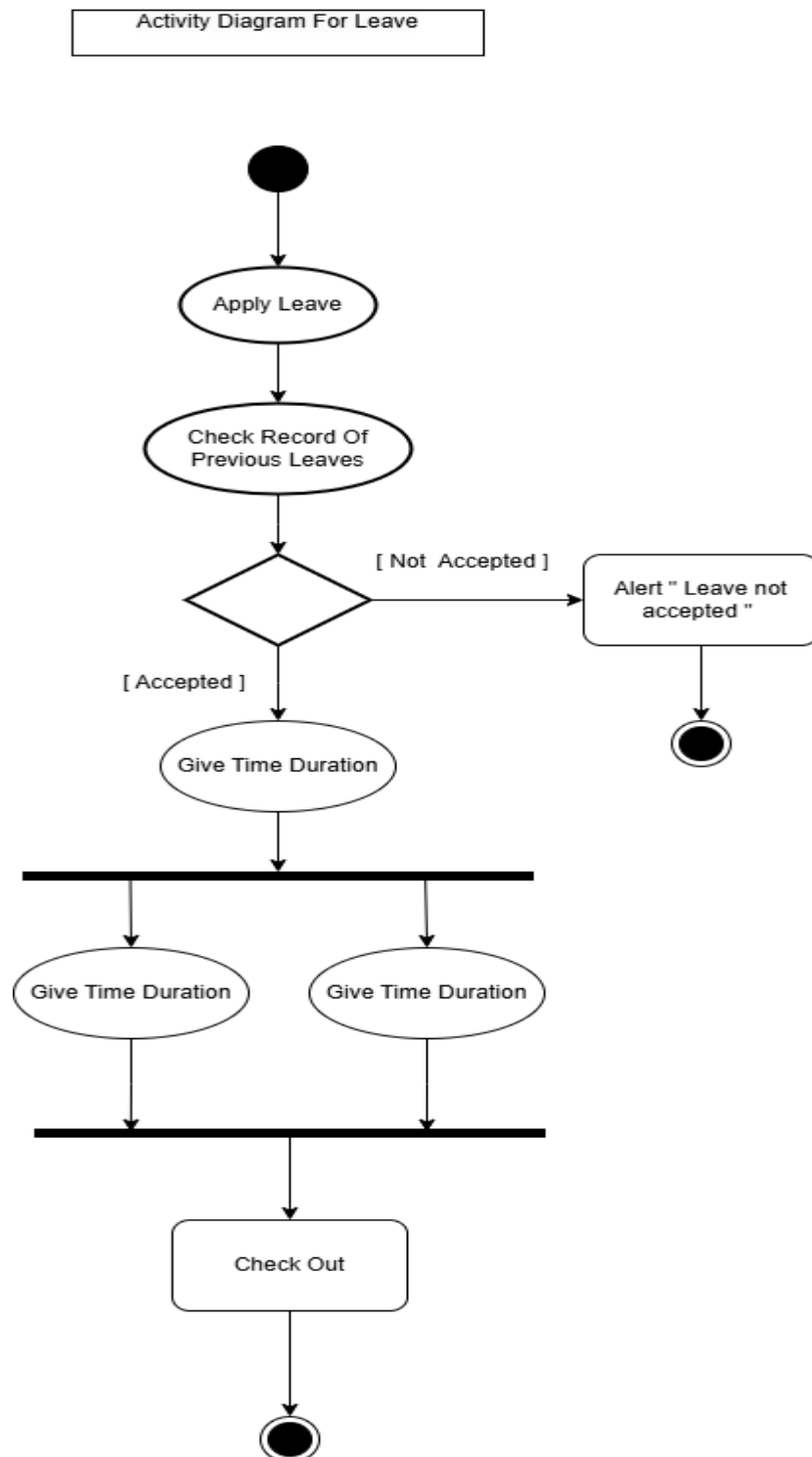


Figure 2.2-2 Activity diagram for leave

SRS – Hostel Management System

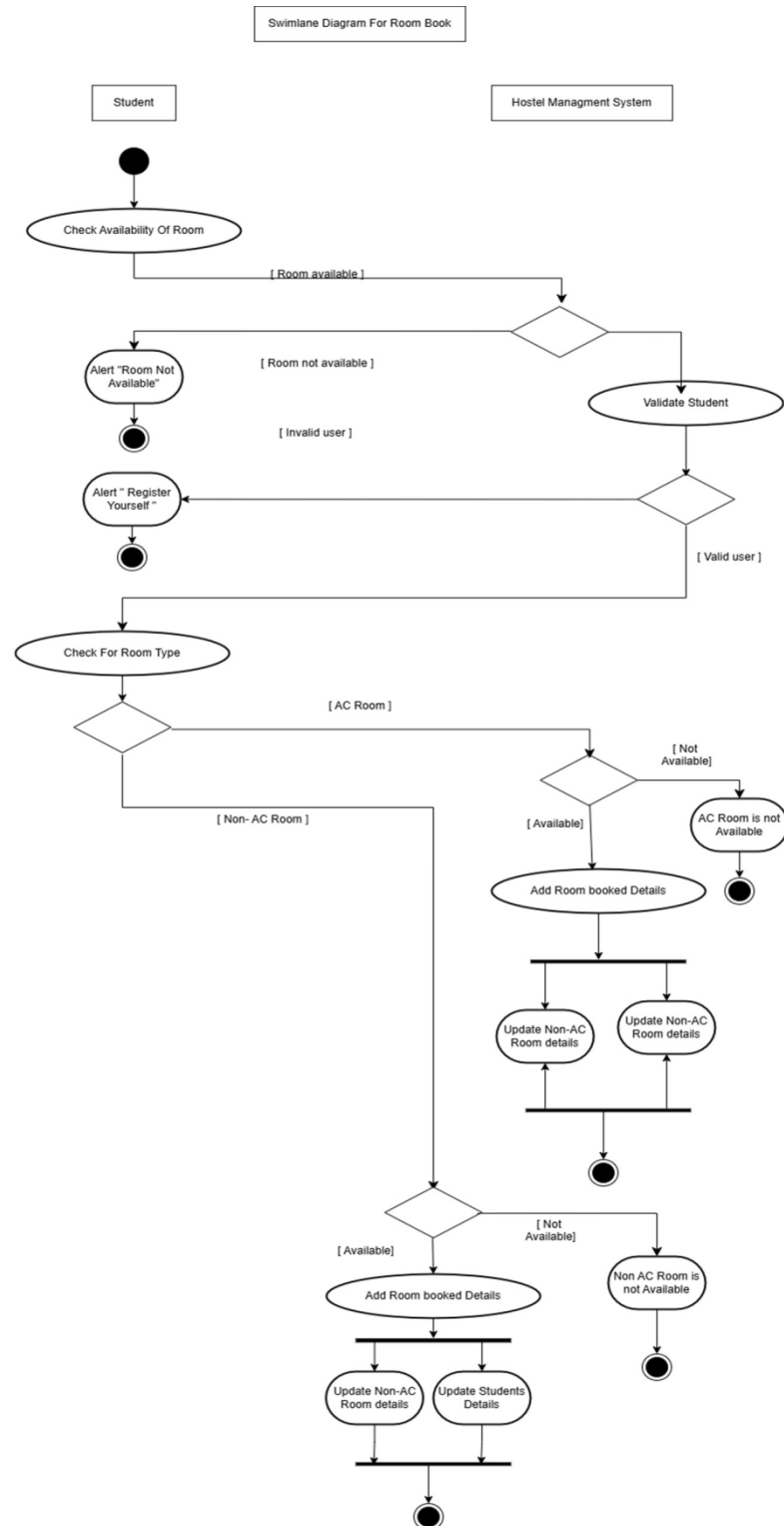


Figure 2.2-3 Swimlane diagram for Room Book

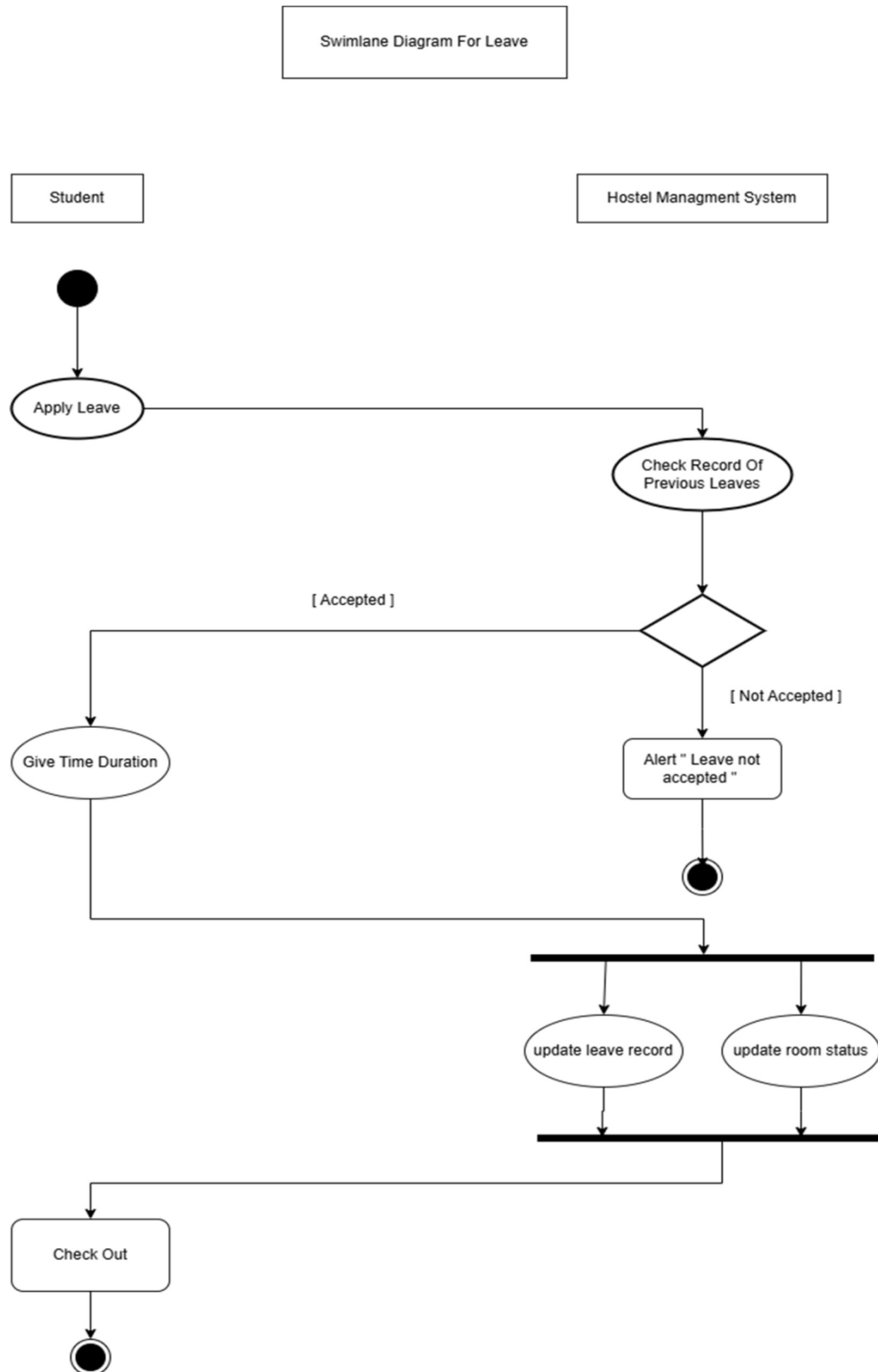


Figure 2.2-4 Swimlane diagram for take leave

2.3 Sequence diagram

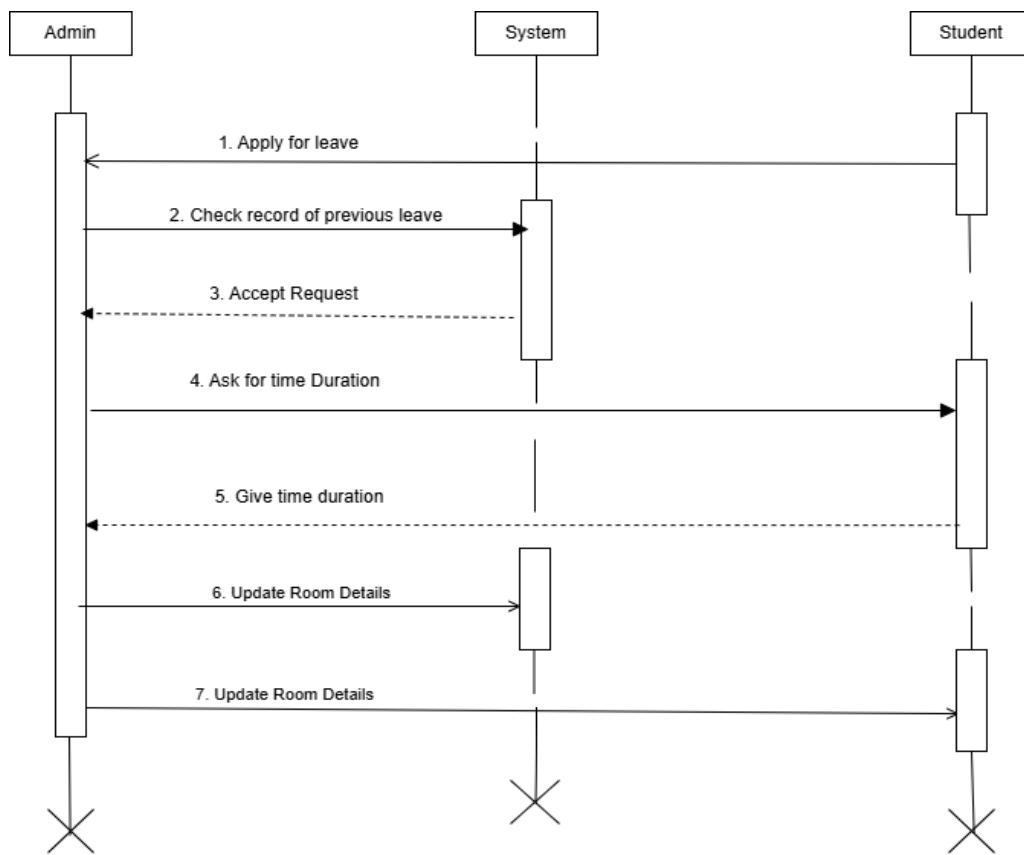


Figure 2.3-1 Sequence diagram for Book Issue

2.4 State diagram

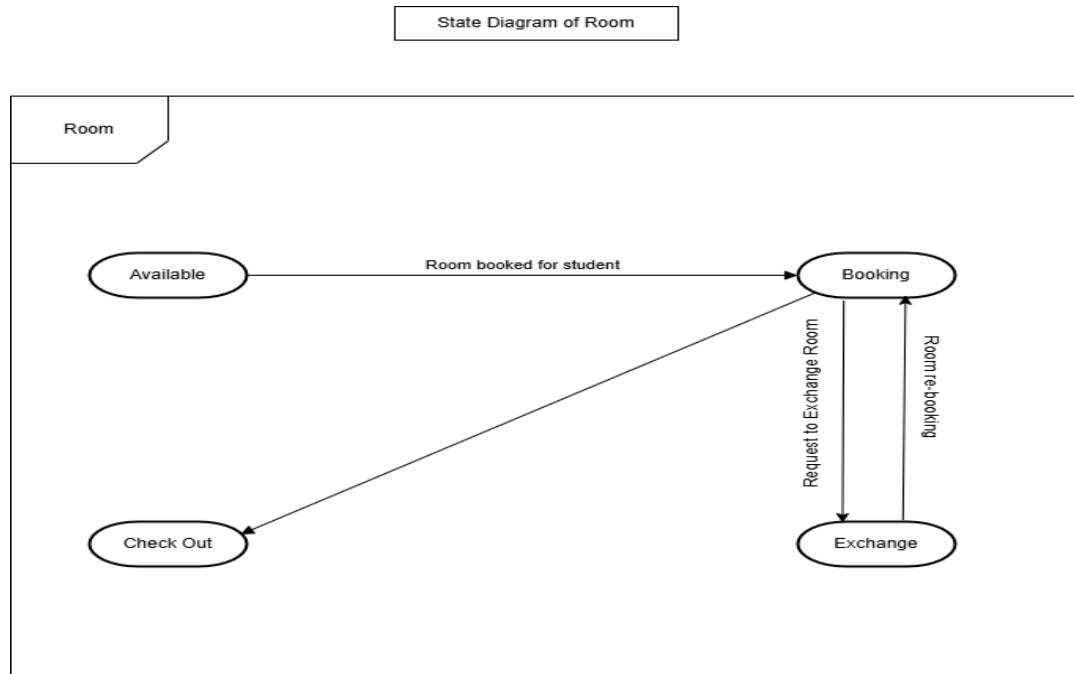


Figure 2.4-1 State diagram of Room

2.5 Class diagram

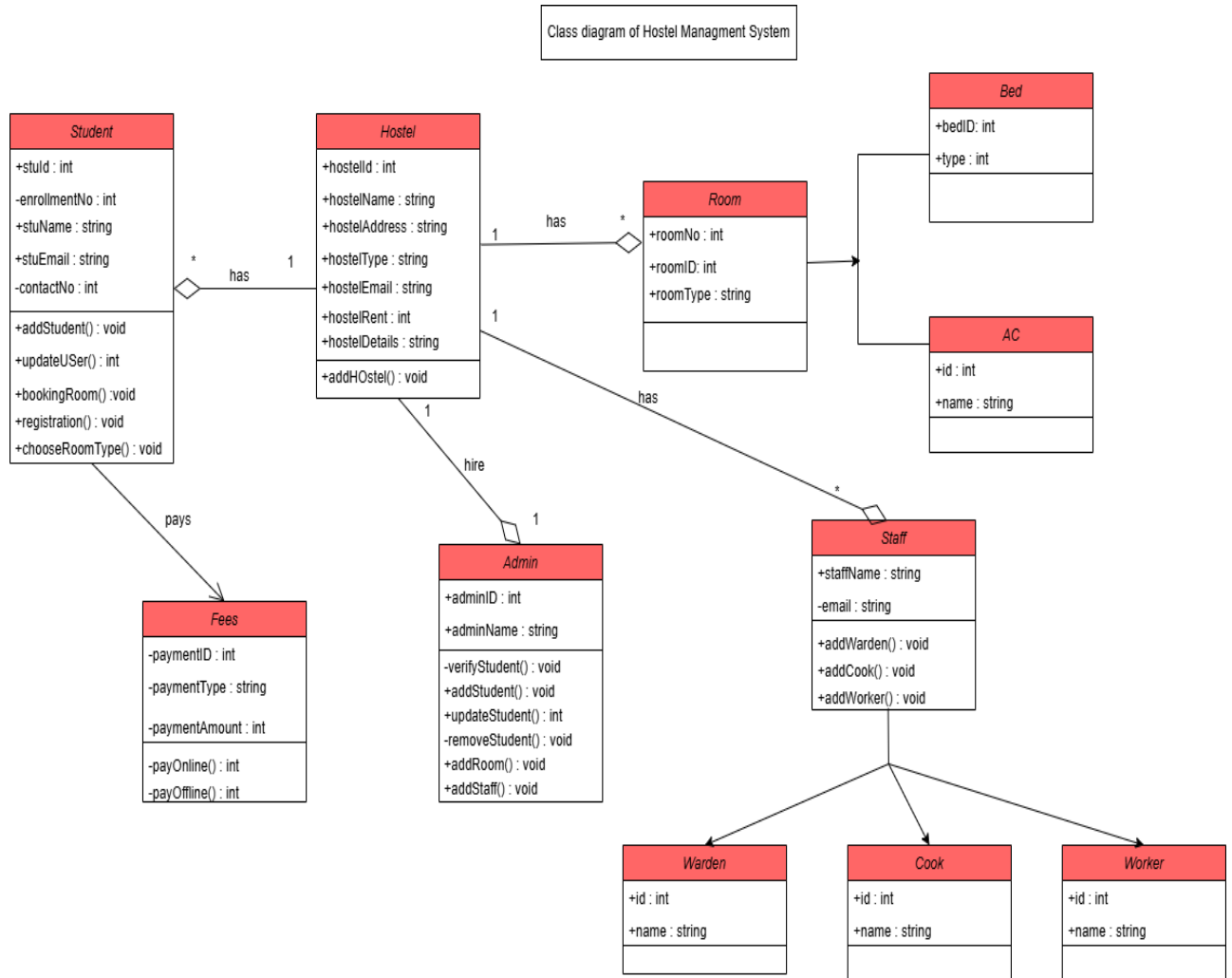


Figure 2.5-1 Class diagram for Hostel management system

2.6 Data flow diagram

2.6.1 Context diagram (level-0)

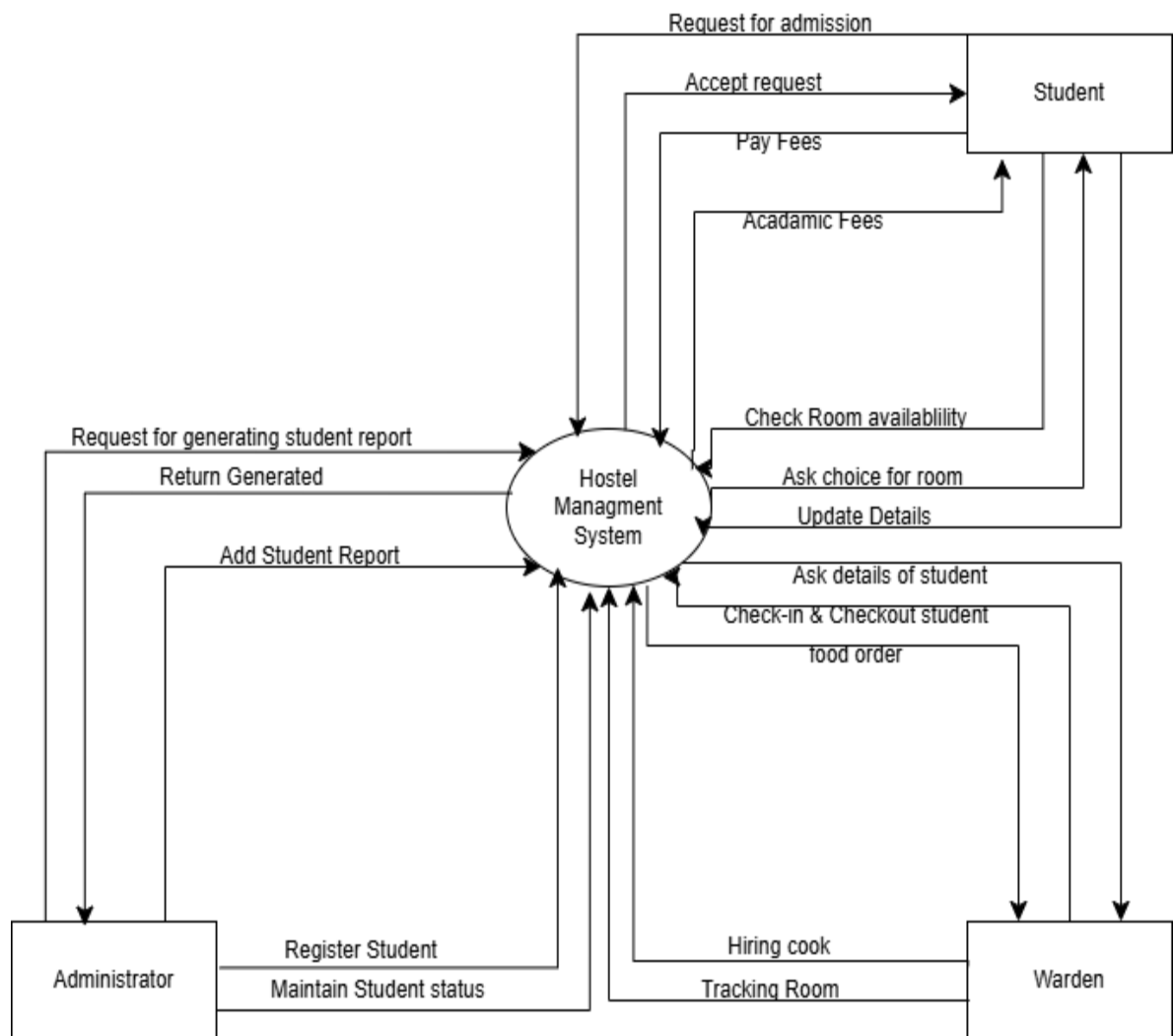


Figure 2.6-1 Context diagram for Hostel management system

2.6.2 DFD Level-1

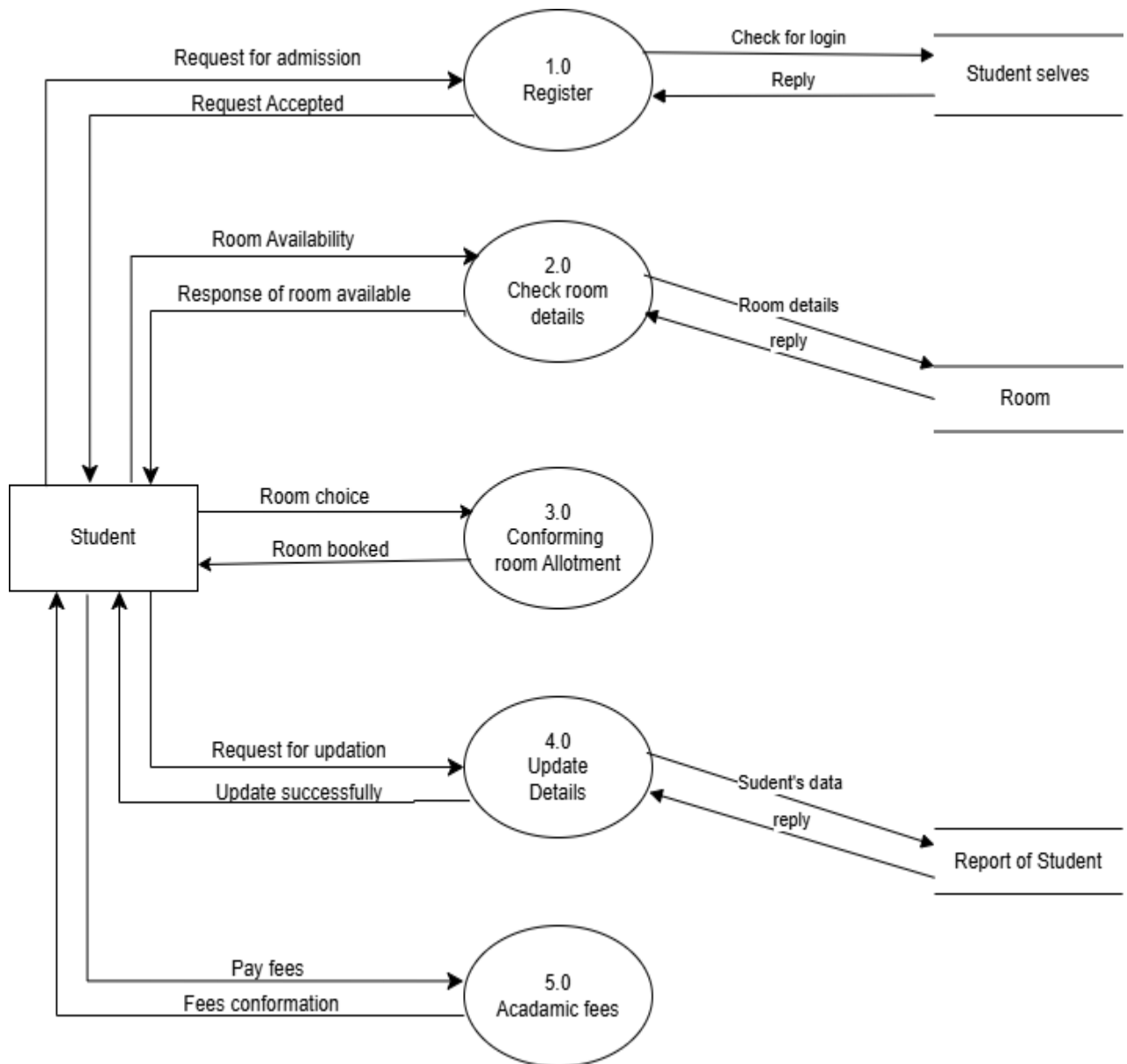


Figure 2.6-2 DFD level-1 for Hostel management system

3 External interface requirement (Screens)

3.1 Screen-1: Update Student Details

Update Student Details

Full Name
Enter your full name

Email Address
Enter your email

Age
Enter your age

Date of Birth
dd/mm/yyyy

Phone Number
Enter your phone number

Address
Enter your address

Update Details

Figure 3.1-1 Screen-1: Update Student Details

Purpose: This form will allow the target end-users to update in the system. To update student, the following information will be encoded in the system.

Table 3.1-1 Screen element of update student

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Full Name	Textbox	M	1	Full Name field should be editable and accept the full name of the student.
2	Email Address	Textbox	M	1	Email Address field should be editable and accept the email address with proper format.
3	Age	Number	M	1	Age field should be editable and accept the age of the student.
4	Date of Birth	Datepicker	M	1	Date of Birth field should be editable and allow date selection.
5	Phone Number	Textbox	M	1	Phone Number field should be editable and accept the phone number with proper format.
6	Address	Textarea	M	1	Address field should be editable and accept the student's address.

SRS – Hostel Management System

7	Update Details	Button	-----	-----	Submit is a button for store the entered data into database.
---	----------------	--------	-------	-------	--

3.2 Screen-2: Room selection form

Hostel Room Selection Form

Full Name

Email Address

Room Type

Room Reference (Optional)

Bed Type

AC or Non-AC Room

Submit Room Selection

Figure 3.2-1 Screen-2: Room selection form

Purpose: This form will be used by the system's users to choose room selection. The users will input the correct rooms to be able to choose to the system.

Table 3.2-1 Screen element of room selection

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Full Name	Textbox	M	1	Full Name field should be editable and accept the full name of the student.
2	Email Address	Textbox	M	1	Email Address field should be editable and accept the email address with proper format.
3	Room Type	Dropdown	M	1	Dropdown field with options for different room types (e.g., Single Room, Double Room, etc.).
4	Room Reference (Optional)	Textbox	O	1	Textbox field for entering a room reference (e.g., room number).
5	Bed Type	Dropdown	M	1	Dropdown field with options for different bed types (e.g., Single Bed, Double Bed).
6	AC or Non-AC Room	Dropdown	M	1	Dropdown field with options for AC or Non-AC rooms.
7	Submit Room Selection	Button	-----	-----	Submit is a button for store the entered data into database.

3.3 Screen-3: Room Allocation

Room Allocation by Administrator

Student Name

Room Type

Bed Type

AC or Non-AC Room

Room Allocation Date

Figure 3.3-1 Screen-3: room allocation

Purpose: This module will allow the system administrator to allocate rooms to student. The admin can add room information and manage it.

Table 3.3-1 Screen element of room allocation

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Student Name	Textbox	M	1	Textbox field to enter the student's full name.
2	Room Type	Dropdown	M	1	Dropdown field with options for different room types (e.g., Single Room, Double Room, etc.).
3	Bed Type	Dropdown	M	1	Dropdown field with options for different bed types (e.g., Single Bed, Double Bed).
4	AC or Non-AC Room	Dropdown	M	1	Dropdown field with options to select AC or Non-AC rooms.
5	Room Allocation Date	Datepicker	M	1	Datepicker field to select the room allocation date.
6	Allocate Room	Button	---	--	Button to submit the room allocation request.

3.4 Screen-4: Check-in & Check-out

Student Check-in & Check-out

Student Name

Room Number

Check-in/Check-out Date

Action (Check-in/Check-out)

Check-in
▼

Update Details

Figure 3.4-1 Screen-4: check-in & check-out

Purpose: This module will allow the system administrator to check-in and check-out students'. The admin can track students.

Table 3.4-1 Screen element of check-in & check-out

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Student Name	Textbox	M	1	Textbox field to enter the student's full name.
2	Room Number	Textbox	M	1	Textbox field to enter the room number.
3	Check-in/Check-out Date	Datepicker	M	1	Datepicker field to select the check-in or check-out date.
4	Action (Check-in/Check-out)	Dropdown	M	1	Dropdown field with options for "Check-in" or "Check-out".
5	Update Details	Button	---	---	Button to submit the check-in or check-out request.

3.5 Screen-5: Handle student details by Warden

Warden - Daily Student Details

Student Name

Room Number

Attendance Status

Date

Update Details

Figure 3.5-1 Screen-5: Handle student details

Purpose: This module will allow the warden to manage student details daily.

Table 3.5-1 Screen element of handle student details

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Student Name	Textbox	M	1	Textbox field to enter the student's full name.
2	Room Number	Textbox	M	1	Textbox field to enter the room number.
3	Attendance Status	Dropdown	M	1	Dropdown field with options for attendance status (e.g., Present, Absent, Leave).
4	Date	Datepicker	M	1	Datepicker field to select the date for which the attendance is being recorded.
5	Update Details	Button	---	---	Button to submit the attendance record.

3.6 Screen-6: Adding health information

Add Health Information

Student Name

Health Condition

Allergies

Emergency Contact

Submit Health Information

Figure 3.6-1 Screen-6: Adding health information

Purpose: This module will allow the student to add health information.

Table 3.6-1 Screen element of handle student health details

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Student Name	Textbox	M	1	Field to enter the student's full name.
2	Health Condition	Textbox	O	1	Field to enter any health conditions the student may have (optional).
3	Allergies	Textbox	O	1	Field to enter any allergies the student may have (optional).
4	Emergency Contact	Textbox	M	1	Field to enter the emergency contact number for the student.
5	Submit Health Information	Button	-	-	Button to submit the entered health information.

3.7 Screen-7: Fees payment

Hostel - Fees Payment

Student Name

Amount Paid

Payment Date

Payment Method

Transaction ID

Remarks

Submit Payment

Figure 3.7-1 Screen-7: Hostel fees payment

Purpose: This module will allow the student to payment fees.

Table 3.7-1 Screen element of handle student fees details

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Student Name	Textbox	M	1	Full name of the student.
2	Amount Paid	Textbox (Numeric)	M	1	Amount of fees paid. Should accept numerical input.
3	Payment Date	Date Picker	M	1	Date of payment (dd/mm/yyyy format).
4	Payment Method	Dropdown/Select	M	1	Method of payment (e.g., Bank Transfer, Cash). Potentially more options could be added.
5	Transaction ID	Textbox	M	1	Transaction ID of the payment.
6	Remarks	Textbox	O	1	Any additional notes or information regarding the payment.

7	Submit Payment	Button	-	-	Button to submit and process the payment information.
---	----------------	--------	---	---	---

3.8 Screen-8: Exchange warden

Exchange Warden

Current Warden Name

New Warden Name

Reason for Exchange

Date of Exchange

Submit Exchange

Figure 3.8-1 Screen-8: Exchange warden

Purpose: This module will allow the admin to exchange warden and manage their details.

Table 3.8-1 Screen element of handle warden

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Current Warden Name	Textbox	M	1	Full name of the current warden.
2	New Warden Name	Textbox	M	1	Full name of the new warden.
3	Reason for Exchange	Textbox	M	1	Explanation for the warden exchange.
4	Date of Exchange	Date Picker	M	1	Date when the warden exchange takes effect (dd/mm/yyyy format).
5	Submit Exchange	Button	-	-	Button to submit and process the warden exchange request.

3.9 Screen-9: Staff schedule

Staff Schedule

Staff Member Name

Enter staff member's name

Schedule Shift

Morning

Start Time

--:-- --

End Time

--:-- --

Assign Schedule

Figure 3.9-1 Screen-9: Staff schedule

Purpose: This module will allow the admin to manage staff schedule.

Table 3.9-1 Screen element of handle staff schedule

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Staff Member Name	Textbox	M	1	Full name of the staff member.
2	Schedule Shift	Dropdown/Select	M	1	Shift assigned to the staff member (e.g., Morning, Afternoon, Evening). Potentially more options could be added.
3	Start Time	Time Picker	M	1	Time the staff member's shift begins.
4	End Time	Time Picker	M	1	Time the staff member's shift ends.
5	Assign Schedule	Button	-	-	Button to submit and assign the schedule to the staff member.

3.10 Screen-10 : Tracking information of rooms

Room Cleanliness & Maintenance

Room Number

Cleanliness Status

Clean
▼

Assigned Staff Member

Maintenance Required

Update Room Status

Figure 3.10-1 Screen-10: Tracking room detail

Purpose: This module will allow the warden to manage and tracking room information.

Table 3.10-1 Screen element of tracking room detail

Sr.	Screen Element	Input Type	O/M	1/N	Description
1	Room Number	Textbox	M	1	Room number to be updated.
2	Cleanliness Status	Dropdown/Select	M	1	Current cleanliness status of the room (e.g., Clean, Dirty, In Progress).
3	Assigned Staff Member	Textbox	M	1	Name of the staff member assigned to the room.
4	Maintenance Required	Textbox	O	1	Details of any maintenance required for the room (optional).
5	Update Room Status	Button	-	-	Button to submit and update the room status.

4. Database design

4.1 List of Tables

- Hostel
- Student
- Administrator
- Warden
- Room
- Fees

Table 4.1-1 Table: Hostel

Column	Data Type	Null	Keys & Constrains	Default Value & Description
HostelID	int	NN	PK (Auto Increment)	
HostelName	varchar(100)	NN		
HostelAddress	varchar(100)	NN		
Foundation_year	int	AN		
HostelDetails	varchar(50)	AN		

Table 4.1-2 Table: Student

Column	Data Type	Null	Keys & Constrains	Default Value & Description
StudentID	int	NN	PK (Auto Increment)	
StudentName	varchar(100)	NN		
Gender	varchar(50)	AN		
StudentEnrollmentNo	int	NN		
StudentEmail	Varchar(100)	NN		
DOB	DateTime	AN		
RoomID	int	NN	FK	Reference of Room table
Contact	int	AN		

Table 4.1-3 Table: Administrator

Column	Data Type	Null	Keys & Constrains	Default Value & Description
AdminID	int	NN	PK (Auto Increment)	
AdminName	varchar(100)	AN		
Email	varchar(100)	AN		
WardenID	int	NN	FK	Reference of Warden table

Table 4.1-4 Table: Warden

Column	Data Type	Null	Keys & Constrains	Default Value & Description
WardenID	int	NN	PK (Auto Increment)	
WardenName	varchar(100)	AN		
Gender	varchar(100)	AN		
Contact	number(10,0)	AN		
AdminID	int	NN	FK	Reference of Admin table

Table 4.1-5 Table: Room

Column	Data Type	Null	Keys & Constrains	Default Value & Description
RoomID	int	NN	PK (Auto Increment)	
RoomType	varchar(100)	AN		
RoomNo	int	NN		
Bed	int	NN		

Table 4.1-6 Table: Fees

Column	Data Type	Null	Keys & Constrains	Default Value & Description
PaymentID	int	NN	PK (Auto Increment)	
Amount	int	NN		
PaymentType	varchar(50)	NN		

5 Stories and Scenario

5.1 Story-1: Add New Student in Hostel

Story # S1	: As a Administrator, I want to add a new student to the hostel database So that they can be assigned a room and receive hostel facilities.
Priority	: High
Estimate	: XL
Reason	: Adding a new student ensure that their details are recorded properly, facilitating smooth room allocation and fee management.

5.1.1 Scenario# S1.1

Scenario# S1.1	: Adding a New Student with Valid Information
Prerequisite	: Administrator is logged into the hostel management system.
Acceptance Criteria	<p>Given: The Hostel Manager is navigated to the student registration page. Valid student information, including name, student ID, contact details, and other relevant information, is entered.</p> <p>When: The administrator selects the "Add New Student" option. The administrator enters valid student details, including name, student ID, contact details, room preference, and other relevant information. The administrator clicks the "Save" button to register the student in the hostel.</p> <p>Then The system successfully registers the student in the hostel.</p> <p>The administrator receives a confirmation message with the student's hostel room number.</p>

5.1.2 Scenario# S1.2

Scenario# S1.2	: Adding a New Book with Invalid Information.
Prerequisite	: The administrator is logged into the hostel management system.
Acceptance : Criteria	<p>Given:The administrator is on the "Student Registration" page.</p> <p>When:The administrator selects the "Add New Student" option. The administrator enters incomplete or incorrect student details (e.g., missing name, invalid contact number, or duplicate student ID). The administrator clicks the "Save" button.</p> <p>ThenThe system displays error messages indicating the missing or incorrect information.</p> <p>The student is not registered until all required details are correctly progress.</p>

5.1.3 Scenario# S1.3

Scenario# S1.3	: Attempting to Add a Duplicate Student
Prerequisite	: The Administrator is logged into the hostel management system and is on the student registration page.
Acceptance Criteria	<p>Given: The student information, including name, student ID, contact details, and other relevant details, is available, and the student with the same ID is already in the database.</p> <p>When: The administrator clicks on the "Add Student" button. Enters the same student details with an existing Student ID.</p> <p>Then the system generates a unique student record and assigns a unique student ID while preventing duplication errors. If multiple entries of the same student are required, the system allows managing multiple hostel stays with different room allocations.</p>

5.2 Story-2: Manage Hostel Fees

Story # S2	: As a Hostel Manager, I want to manage hostel fees of students So that payments are recorded, and dues are tracked.
Priority	: High
Estimate	: M
Reason	: Proper fee management ensures smooth financial operations of the hostel

5.2.1 Scenario# S2.1

Scenario# S2.1	: Recording a Fee Payment
Prerequisite	: The student has been assigned a room and needs to pay hostel fees.
Acceptance Criteria	<p>: Given: The student is enrolled in the hostel, and fee details are available..</p> <p>When: The Hostel Manager navigates to the fee management section.</p> <p>Selects the student and enters the fee payment details.</p> <p>Clicks on "Confirm Payment."</p> <p>Then the system records the payment and generates a receipt for the student.</p>

5.2.2 Scenario# S2.2

Scenario# S2.2	: A Handling Late Fee Payments
Prerequisite	: The student has an overdue payment.
Acceptance Criteria	<p>: Given: The student has an outstanding fee and is notified about the due amount.</p> <p>When: The Administrator navigates to the fee management section.</p> <p>Selects the student and enters the late fee details.</p> <p>Clicks on "Confirm Payment."</p> <p>Then the system updates the outstanding balance and records the payment with an additional late fee if applicable.</p>

5.2.3 Scenario# S2.3

Scenario# S2.3	: Generating a Fee Receipt
Prerequisite	: The student has completed the fee payment.
Acceptance Criteria	<p>Given: The fee payment is successfully recorded in the system.</p> <p>When: The Hostel Manager selects the student in the fee management section.</p> <p>Clicks on "Generate Receipt."</p> <p>Then the system generates a digital receipt with a unique transaction ID and sends a copy to the student's registered email.</p>

5.3 Story-3: Assign Room to Student

Story # S3	: As a Hostel Manager, I want to assign rooms to students So that they have a designated place to stay in the hostel.
Priority	: High
Estimate	: M
Reason	: Assigning rooms ensures proper accommodation for students and efficient hostel space management.

5.3.1 Scenario# S3.1

Scenario# S3.2	: Assigning a Room to a New Student
Prerequisite	: The student is registered in the hostel management system.
Acceptance Criteria	<p>Given: The student has been added to the hostel database and needs room allocation.</p> <p>When: The Hostel Manager navigates to the room assignment section.</p> <p>Selects an available room.</p> <p>Assigns the selected room to the student and confirms.</p> <p>Then the system updates the student's profile with the assigned room number and notifies them.</p>

5.3.2 Scenario# S3.2

Scenario# S3.2	: Changing a Student's Room
Prerequisite	: The student is already assigned a room but needs to be relocated.
Acceptance Criteria	<p>Given: The student has a valid reason for room change and there are available rooms.</p> <p>When: The Administrator selects the student's current room assignment.</p> <p>Chooses a new available room.</p> <p>Confirms the room reassignment.</p> <p>Then the system updates the student's profile with the new room details and generates a confirmation notification.</p>

5.3.3 Scenario# S3.3

Scenario# S3.2	: Attempting to Assign an Already Occupied Room
Prerequisite	: The Hostel Manager is attempting to assign a room that is already occupied.
Acceptance Criteria	<p>Given: The room is currently assigned to another student.</p> <p>When The Hostel Manager selects an occupied room for assignment.</p> <p>Attempts to assign it to another student.</p> <p>Confirms the room reassignment.</p> <p>Then the system prevents the assignment and displays an error message indicating that the room is unavailable.</p>

6 Test cases

Project Name:	Hostel Management system	Test Designed by:	Tanisha Bhalodiya
Module Name:	Update student details	Test Designed date:	08-03-2025
Release Version:	1.0	Test Executed by:	R. B. Gondaliya
		Test Execution date:	08-03-2025

Pre-condition: Student should be properly updated

Test Case ID	Test Title	Test Type	Description	Test Case ID
TC_001	Update student to web application with valid information	Functional	update to hostel management system web page application through valid credential	TC_001
TC_002	Update student to web application with invalid information	Functional	update to hostel management system web application through invalid credential	TC_002

Test Case Title	Update student to web application with valid information
Test Type	Functional
Test Priority	High
Pre-Condition	Web application should be accessible

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	BUG ID
1	Access Student Update Page	The update page should load properly	Page loaded Successfully	Pass		https://accounts.google.com/ServiceLogin	
2	Enter valid Student ID in the ID field	The ID field should be editable and accept valid Student ID	Student ID input accepted	Pass		Student ID: 123456	
3	Update Student Name	The name field should be editable and accept valid input	Name updated successfully	Pass		Updated Name: Rb gondaliya	
4	Update Student Email	The email field should be editable and accept valid email format	Email updated successfully	Pass	-	Updated Email:rajesh@example.com	
5	Update Contact Number	The contact field should be editable and accept only numeric input	Contact updated successfully	Pass	-	Updated Contact: 9876543210	
6	Update Address	The address field should be editable and accept valid input	Address updated successfully	Pass		Updated Address: 123 Main St, City	

7	Click on "Save" button	Data should be saved successfully and a confirmation message should appear	Confirmation message displayed	Pass	-	-	-
8	Verify Updated Details	Updated details should reflect on the student profile	Updated details displayed correctly	Pass			

Test Case Title	Update student to web application with invalid information
Test Type	Functional
Test Priority	Medium
Pre-Condition	Web application should be accessible.

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	Bug ID
1	Verify that the system does not accept an invalid Student ID	Should display an error message: "Invalid Student ID"	Displayed error: "Invalid Student ID"	Pass		Student ID: 000000	
2	Verify that the system does not accept an invalid email format	Should display an error message: "Invalid email format"	Displayed error: "Invalid email format"	Pass		Email: abc@wrong	
3	Verify that the system does not accept non-numeric characters in the contact number field	Should display an error message: "Invalid contact number"	Displayed error: "Invalid contact number"	Pass		Contact: ABC123	
4	Verify that the system does not accept an empty required field	Should display an error message: "This field is required"	No validation performed	Fail	Validation missing, needs to be fixed	Name: (Blank)	Bug_003

	(e.g., Name field left blank						
5	Verify that invalid address input (special characters) is not accepted	Should display an error message: "Invalid address format"	Displayed error: "Invalid address format"	Pass	-	Address: #@\$!&*	-
6	Verify that clicking "Save" without making any changes does not save data	Should display a message: "No changes detected"		Pass	-		-

Project Name:	Hostel Management system	Test Designed by:	Tanisha Bhalodiya
Module Name:	Adding health information	Test Designed date:	08-03-2025
Release Version:	1.0	Test Executed by:	R. B. Gondaliya
		Test Execution date:	08-03-2025

Pre-condition: Student should be properly updated

Test Case ID	Test Title	Test Type	Description	Test Case ID
TC_001	add student health information with valid details	Functional	Verify that the system allows adding health information with valid student details.	TC_001
TC_002	add student health information with invalid data	Functional	Verify that the system does not accept invalid inputs (e.g., non-numeric characters in the Emergency Contact field).	TC_002

Test Case Title	Add student health information with valid details
Test Type	Functional
Test Priority	High
Pre-Condition	Web application should be accessible

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	BUG ID
1	Verify that the system does not accept an empty Student Name field	Should display an error message: "This field is required"	No validation performed	Fail	Validation missing, needs to be fixed	Name: (Blank)	Bug_001
2	Verify that the system does not accept an empty Emergency Contact field	Should display an error message: "This field is required"	Displayed error: "This field is required"	Pass	-	Emergency Contact: (Blank)	-
3	Verify that the system does not accept non-numeric characters in the Emergency Contact field	Should display an error message: "Invalid contact number"	Displayed error: "Invalid contact number"	Pass	-	Updated Name: Rb gondaliya	-
4	Update Student Email	The email field should be editable and accept valid email format	Email updated successfully	Pass	-	Updated Email:rajesh@example.com	-

SRS – Hostel Management System

5	Update Contact Number	The contact field should be editable and accept only numeric input	Contact updated successfully	Pass	-	Updated Contact: 9876543210	
6	Update Address	The address field should be editable and accept valid input	Address updated successfully	Pass		Updated Address: 123 Main St, City	
7	Click on "Save" button	Data should be saved successfully and a confirmation message should appear	Confirmation message displayed	Pass	-	-	-
8	Verify Updated Details	Updated details should reflect on the student profile	Updated details displayed correctly	Pass			

Test Case Title	Add student health information with invalid details
Test Type	Functional
Test Priority	High
Pre-Condition	Web application should be accessible

Test Step	Test Case Description	Expected Result	Actual Result	Status	Comment	Data	BUG ID
1	Verify that the system does not accept an empty Student Name field	Should display an error message: "This field is required"	No validation performed	Fail	Validation missing, needs to be fixed	Name: (Blank)	Bug_001
2	Verify that the system does not accept an empty Emergency Contact field	Should display an error message: "Invalid name format"	Displayed error: "Invalid name format"	Pass	-	Name: @\$\$\$%	-
3	Verify that the system does not accept non-numeric characters in the Emergency Contact field	Should display an error message: "This field is required"	Displayed error: "This field is required"	Pass		Emergency Contact: (Blank)	
4	Verify that the system does not accept non-numeric characters in the Emergency Contact field	Should display an error message: "Invalid contact number"	Displayed error: "Invalid contact number"	Pass	-	Emergency Contact: ABC123	

SRS – Hostel Management System

5	Verify that the system allows submission with empty optional fields (Health Condition & Allergies)	Data should be successfully submitted	Confirmation message displayed	Pass	-	Name: John Doe, Health Condition: (Blank), Allergies: (Blank), Emergency Contact: 9876543210	
6	Verify that clicking "Submit Health Information" without making any changes does not update data	Should display a message: "No changes detected"	Displayed message: "No changes detected"	Pass	-	-	-
7	Verify UI elements of the health information form	All fields and the submit button should be properly displayed and aligned	Form elements displayed correctly	Pass	-	-	-

7 References

- http://www.w3schools.com/html/html_intro.asp
- <https://www.w3schools.com/php/default.asp>
- <https://www.javatpoint.com/uml>