Institute of Software Engineering

Diploma in Comprehensive Master Java Developer (CMJD)

Module: Programming Fundamentals
Course Work - PRF
Name: Kavindu Rajapaksha
Batch: CMJD 113

Table of Content

1.Introduction 2
2. System Overview 3
3. Functional 4
3.1 Login 4
3.2 Home Screen 6
3.3 Manage Rooms 12
3.3.1 Add Room 16
3.3.2 Update room 23
3.3.3. Delete Room 26
3.3.4.Search Room 28
3.3.5 View All Rooms 29
3.4 Manage Students 30
3.4.1 Add Student 30
3.4.2 Update Student 37
3.4.3 Delete Student 40
3.4.4 Search Student 42
3.4.5. View all student 43
3.5 Allocated Bed 44
3.6 Vacate Bed 50
3.7 Transfer Bed 53
3.8 View Reports 55
3.8.1. Occupancy Map 55
3.8.2. Vacant Beds By Floor 55
3.8.3. Students Per Room 55
3.8.4. Over Dues 56
3.8.5. Revenue Projection (Daily) 56
4. Conclusion 57

1.Introduction

The HostelMate System is a console-based hostel management application developed entirely using core Java concepts and array-based data structures. It is designed to efficiently manage hostel operations by maintaining accurate and organized records of students, rooms, bed allocations, transfers, and vacating processes without relying on databases or external libraries.

This system provides a centralized platform for hostel administrators to handle day-to-day management tasks such as adding and updating student details, managing room information, allocating and transferring beds, and tracking occupancy and availability. In addition, it includes report generation features to display data such as current occupancy levels, overdue payments, and overall revenue insights.

By implementing structured validation and logical control through Java's fundamental programming constructs — loops, conditions, and arrays — the HostelMate System ensures data consistency, error handling, and smooth user interaction in a text-based environment.

The main objective of this project is to simplify administrative tasks, reduce manual errors, and enhance operational efficiency within hostel management through an easy-to-use and reliable software solution.

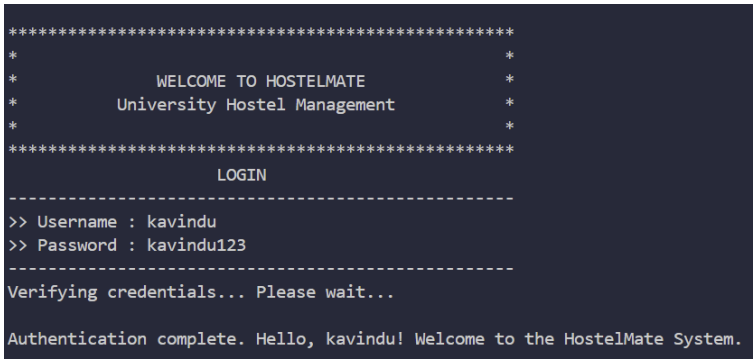
2. System Overview

Feature	Description
Programming Language	Java
Libraries Used	java.time.LocalDate, java.util.Scanner
Data Storage	2D String Arrays
User Interface	Console / Command-Line
Core Functions	Add / Update / Delete / Search / View
Validation	Manual input validation for all fields
Reports	Occupancy Map, Vacant Beds, Students per Room, Overdue Dues, Revenue Projection

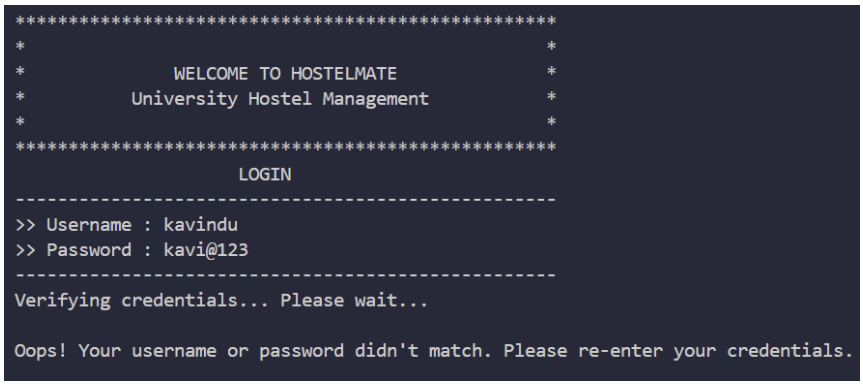
3. Functional

3.1 Login

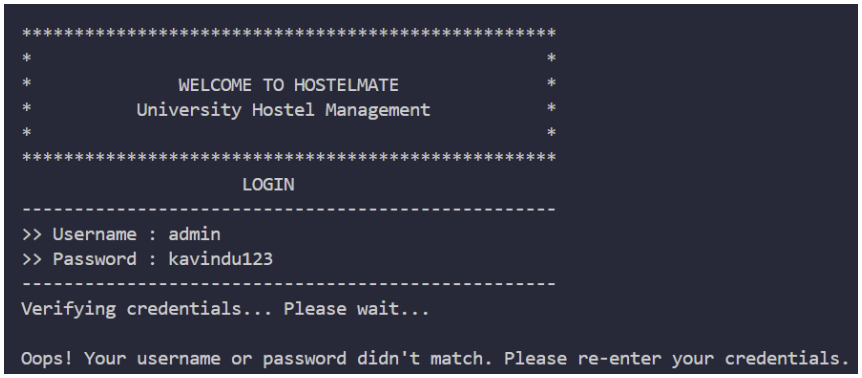
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid login	Username: kavindu Password: kavindu123	“Authentication complete. Hello, kavindu! Welcome to the HostelMate System.”	Pass



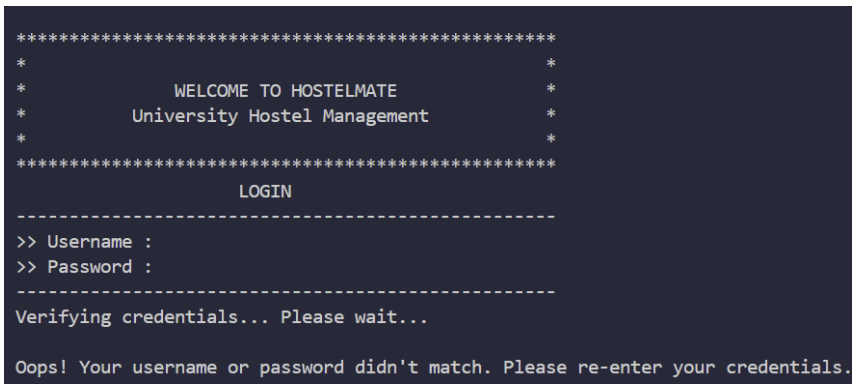
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Invalid password	Username: kavindu Password: wrongpass	Oops! Your username or password didn't match. Please re-enter your credentials.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC03	Invalid username	Username: admin Password: kavindu123	Oops! Your username or password didn't match. Please re-enter your credentials.	Pass

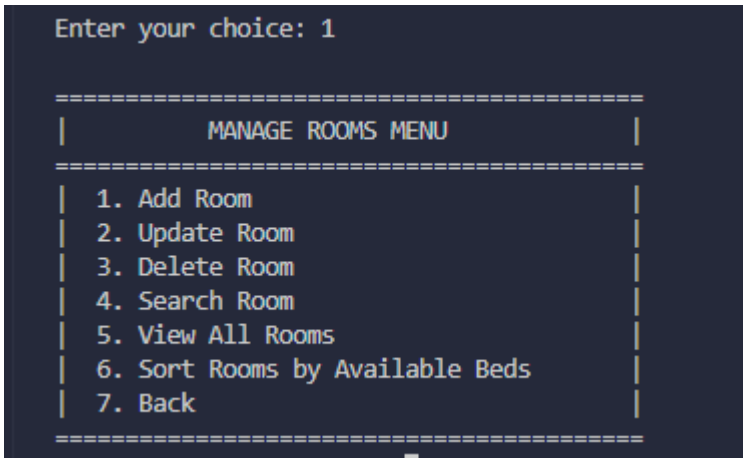


Test Case ID	Scenario	Input	Expected Output	Result
TC04	Empty username and password	Username: *(Enter)* Password: *(Enter)*	Oops! Your username or password didn't match. Please re-enter your credentials.	Pass

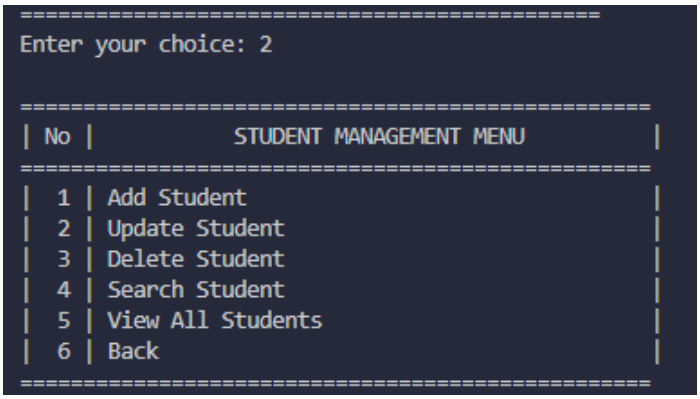


3.2 Home Screen

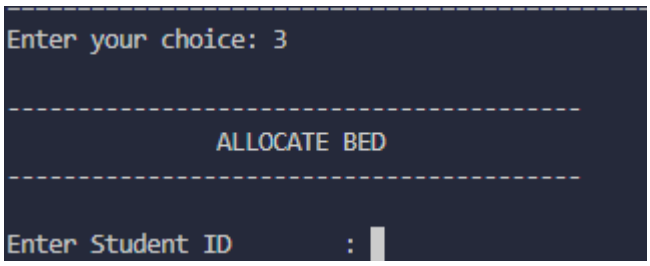
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Option: Manage Rooms	1	Calls manageRooms() method → Displays room management menu.	Pass



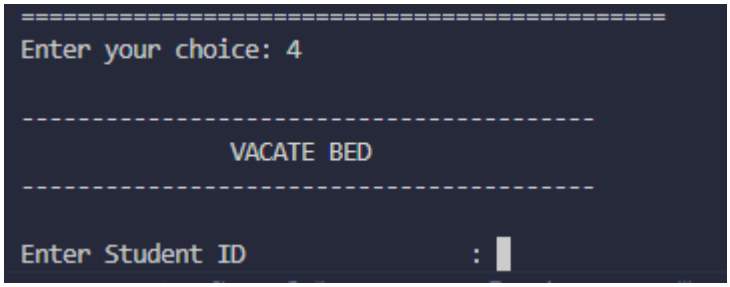
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Valid Option: Manage Students	2	Calls `manageStudents()` → Displays student management menu.	Pass



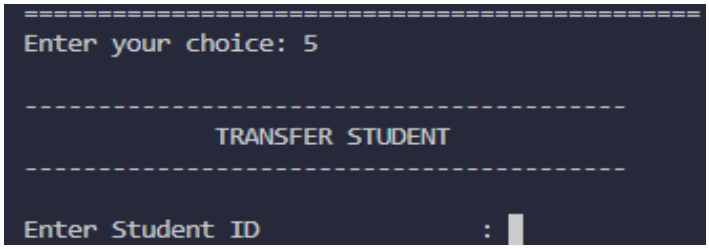
Test Case ID	Scenario	Input	Expected Output	Result
TC03	Valid Option: Allocate Bed	3	Calls `allocateBed()` → Displays allocate bed form.	Pass



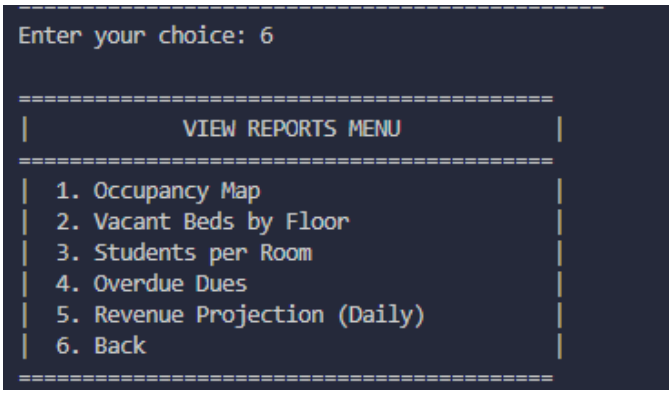
Test Case ID	Scenario	Input	Expected Output	Result
TC04	Valid Option: Vacate Bed	4	Calls `vacateBed()` → Displays **vacate details form**	Pass



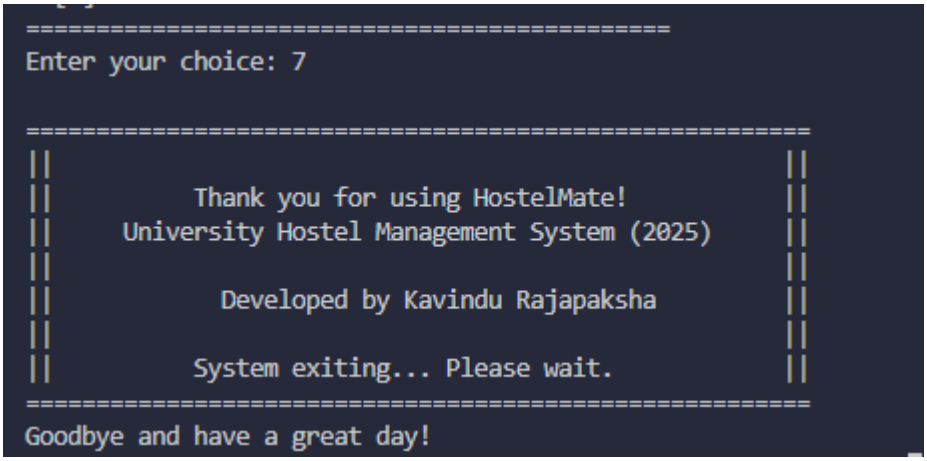
Test Case ID	Scenario	Input	Expected Output	Result
TC05	Valid Option: Transfers	5	Calls `transferBed()` → Displays transfer bed form..	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC06	Valid Option: View Reports	6	Calls `viewReports()` → Displays report view options.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC07	Valid Option: Manage Rooms	7	Calls `Exit()` → Ends the program	Pass



Test Case	Scenario	Input	Expected Output	Result

ID				
TC08	Invalid Option (less than 1)	0	Displays: `Invalid Option..Choose Correct one..`	Pass

```
==============================================
||           HOSTELMATE MAIN MENU           ||
==============================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==============================================
Enter your choice: 0
Invalid Option..Choose Correct one..
```

Test Case ID	Scenario	Input	Expected Output	Result
TC09	Invalid Option(greater than 7)	9	Displays: `Invalid Option..Choose Correct one..`	Pass

```
==============================================
||           HOSTELMATE MAIN MENU           ||
==============================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==============================================
Enter your choice: 9
Invalid Option..Choose Correct one..
```

Test Case ID	Scenario	Input	Expected Output	Result

TC10	Empty Input	(Press Enter)	Displays: `Invalid Option..Choose Correct one.`	Pass

```
==========================================
||          HOSTELMATE MAIN MENU          ||
==========================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==========================================
Enter your choice:
>>> Invalid input. Please Enter valid input.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC11	Alphabetic Input	abc	Displays: `Invalid Option..Choose Correct one.`	Pass

```
==========================================
||          HOSTELMATE MAIN MENU          ||
==========================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==========================================
Enter your choice: abc
>>> Invalid input. Please Enter valid input.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC12	Special Character Input	#	Displays: `Invalid Option..Choose Correct one.`	Pass

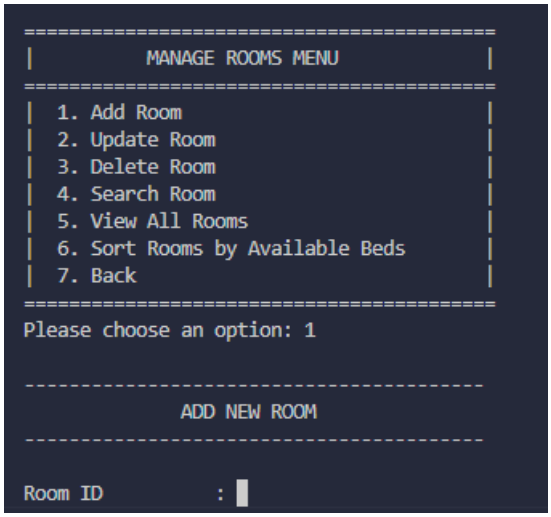
```
==============================================
||           HOSTELMATE MAIN MENU           ||
==============================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==============================================
Enter your choice: #
>>> Invalid input. Please Enter valid input.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC13	Mixed Alphanumeric input	2a	Displays: `Invalid Option..Choose Correct one.`	Pass

```
==============================================
||           HOSTELMATE MAIN MENU           ||
==============================================
  [1] Manage Rooms
  [2] Manage Students
  [3] Allocate Bed
  [4] Vacate Bed
  [5] Transfers
  [6] View Reports
  [7] Exit
==============================================
Enter your choice: 2a
>>> Invalid input. Please Enter valid input.
```

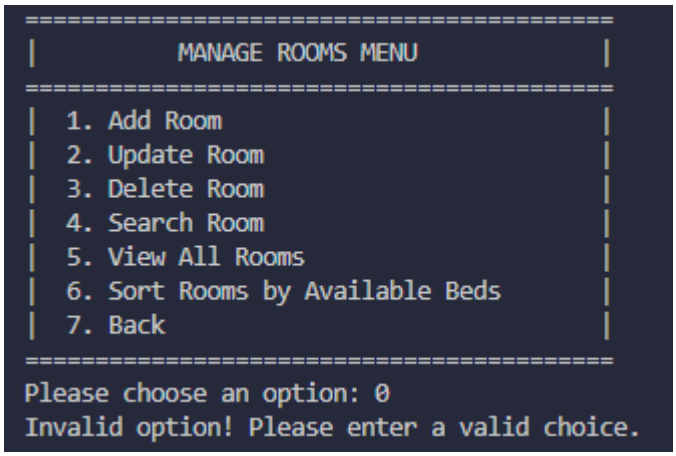
3.3 Manage Rooms

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid option – Add Room	1	Calls `addRoom()` method. Displays “Add Room”	Pass

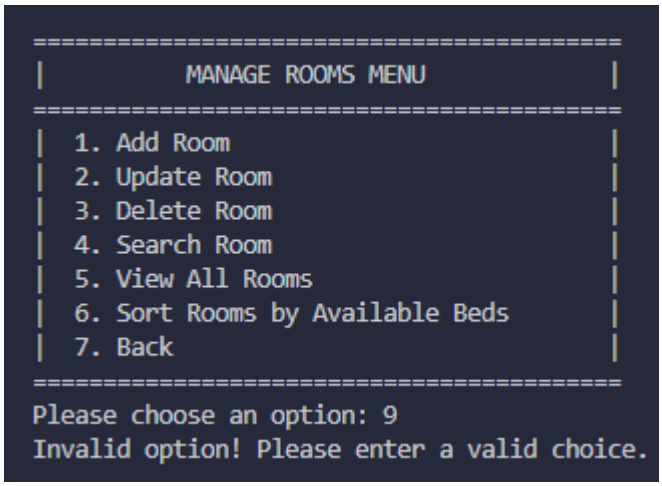


Test Case ID	Scenario	Input	Expected Output	Result
TC02	Valid option – Back	7	Returns to main menu (`home()` method).	Pass

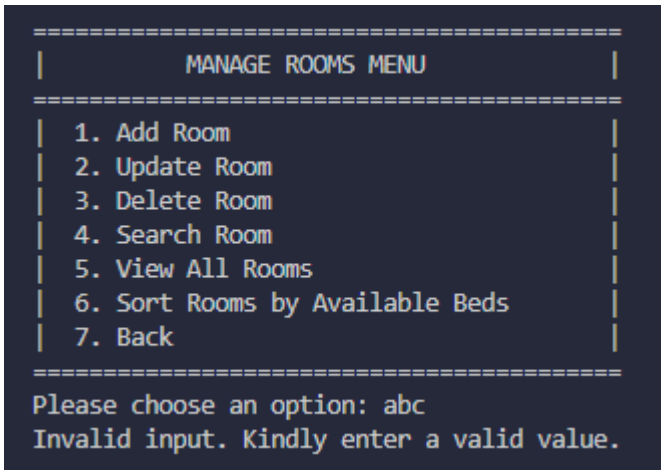
Test Case ID	Scenario	Input	Expected Output	Result
TC03	Invalid option – below range	0	Shows: “Invalid option! Please enter a valid choice.”	Pass



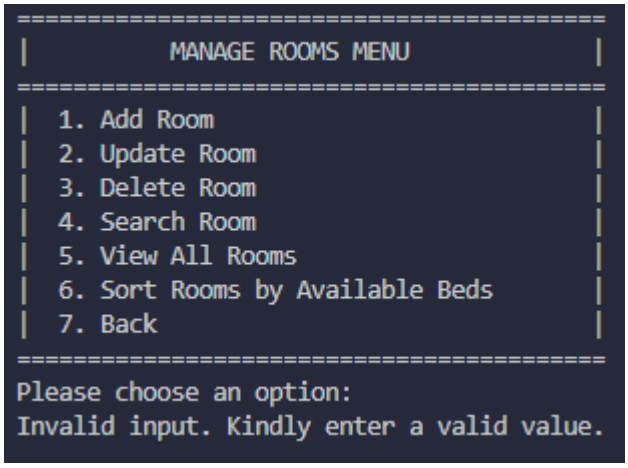
Test Case ID	Scenario	Input	Expected Output	Result
TC04	Invalid option – above range	9	Calls manageRooms() method → Displays room management menu.	Pass



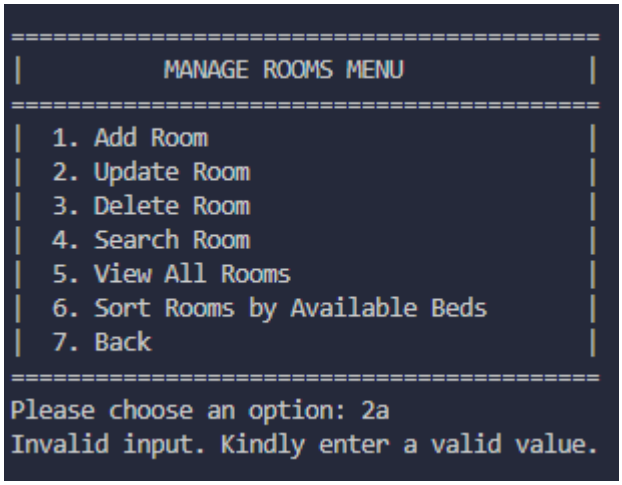
Test Case ID	Scenario	Input	Expected Output	Result
TC05	Alphabetic input	abc	Shows: “Invalid input. Kindly enter a valid value.”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC06	Empty input	(Press Enter)	Shows: “Invalid input. Kindly enter a valid value.”	Pass

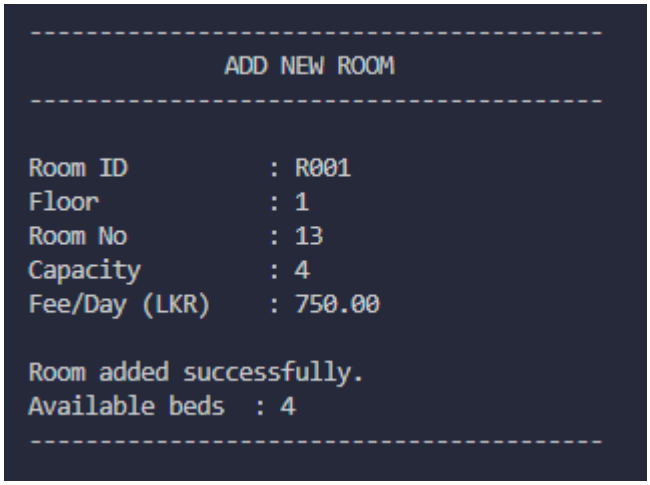


Test Case ID	Scenario	Input	Expected Output	Result
TC07	Mixed input	2a	Shows: “Invalid input. Kindly enter a valid value.”	Pass

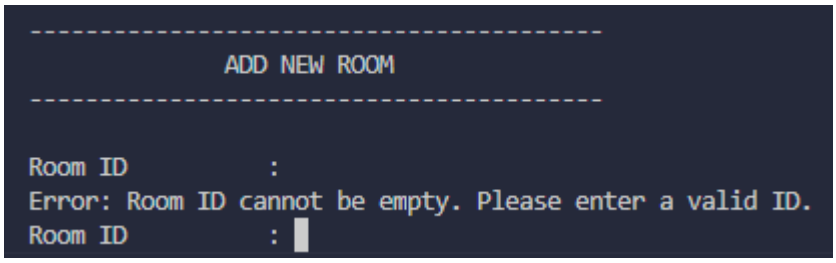


3.3.1 Add Room

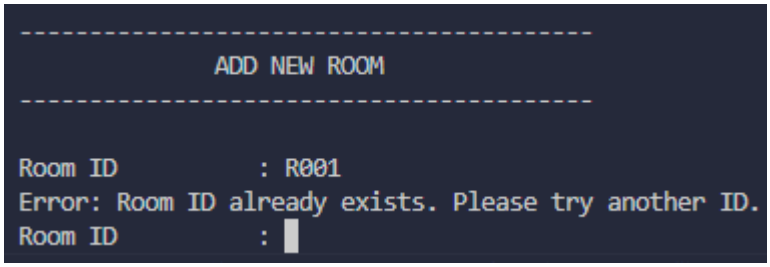
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid room entry	Room ID: R001 Floor: 1 Room No: 13 Capacity: 4 Fee/Day: 750.00	Displays: “Room added successfully.” “Available beds: 4”	Pass



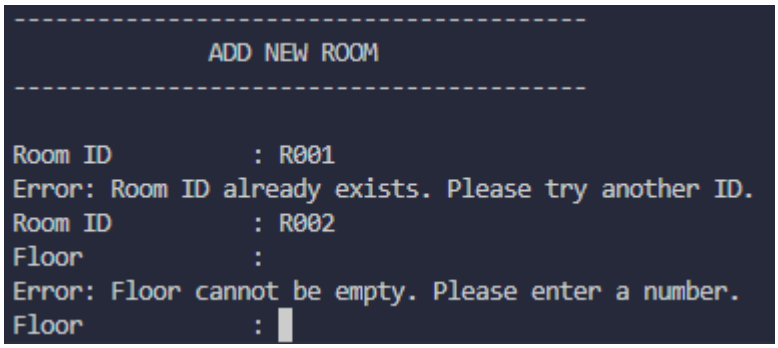
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Room ID	1	(Press Enter)	Pass



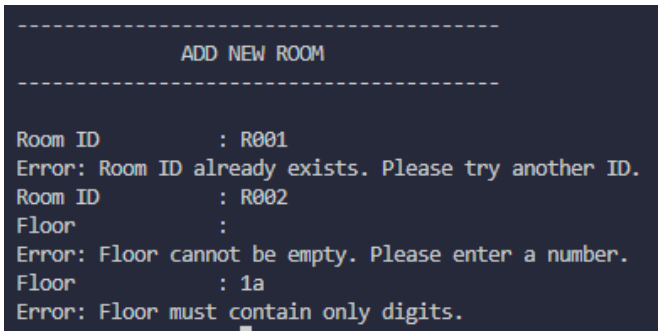
Test Case ID	Scenario	Input	Expected Output	Result
TC03	Duplicate Room ID	R001	Shows error: “Error: Room ID already exists.”	Pass



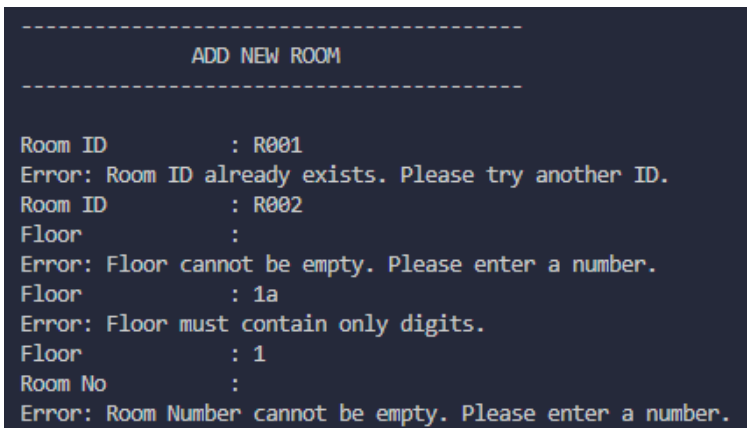
Test Case ID	Scenario	Input	Expected Output	Result
TC04	Empty Floor	Empty Floor	Shows: “Error: Floor cannot be empty.”	Pass



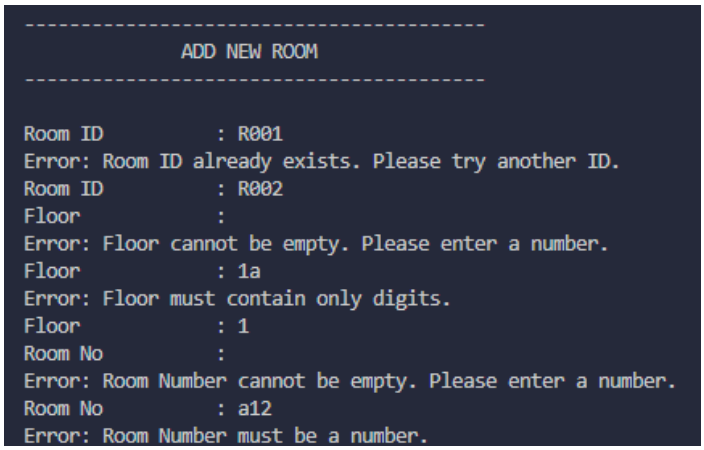
Test Case ID	Scenario	Input	Expected Output	Result
TC05	Non-numeric Floor	1a	Shows: “Error: Floor must contain only digits.”	Pass



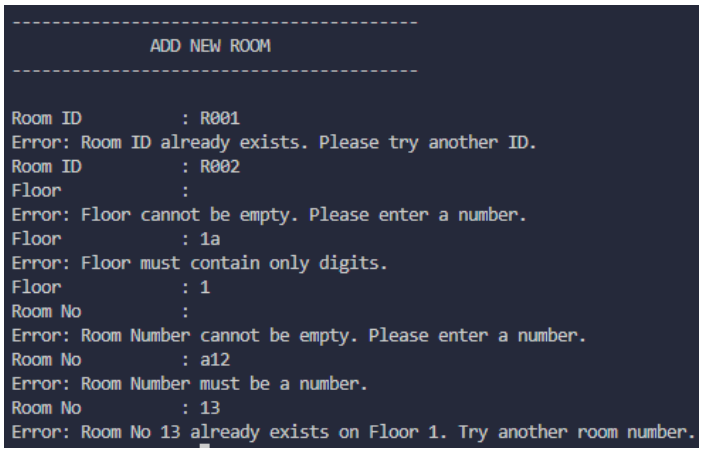
Test Case ID	Scenario	Input	Expected Output	Result
TC06	Empty Room Number	(Press Enter)	Shows: “Error: Room Number cannot be empty.”	Pass



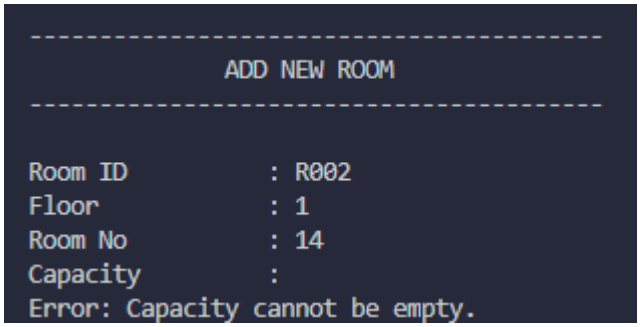
Test Case ID	Scenario	Input	Expected Output	Result
TC07	Non-numeric Room Number	a12	Shows: “Error: Room Number must be a number.”	Pass



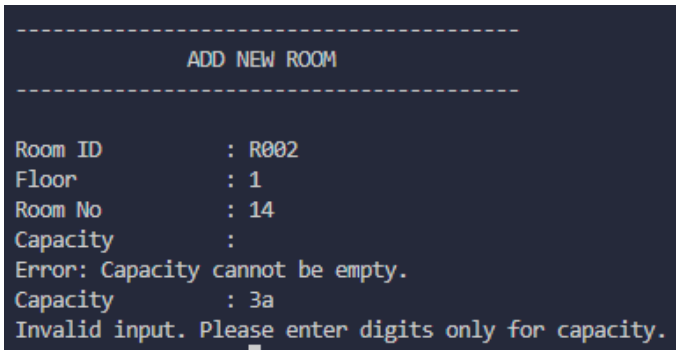
Test Case ID	Scenario	Input	Expected Output	Result
TC08	Duplicate Room on same floor	Floor: 1 Room No: 13	Calls manageRooms() method → Displays room management menu.	Pass



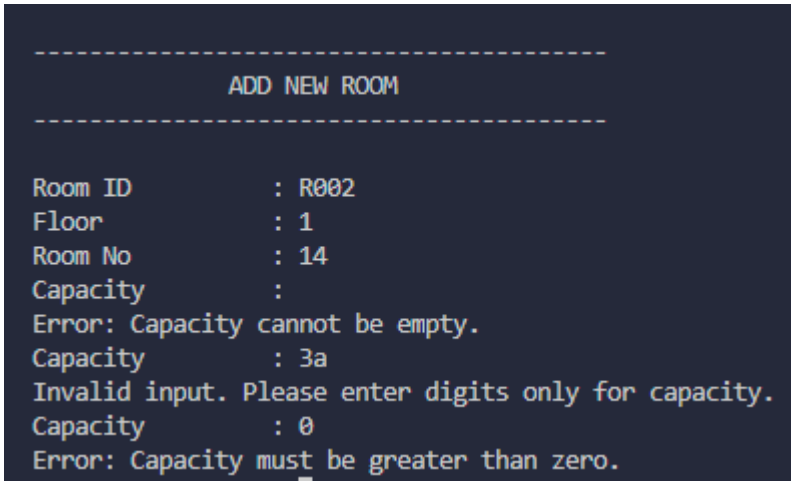
Test Case ID	Scenario	Input	Expected Output	Result
TC09	Empty Capacity	(Press Enter)	Shows: “Error: Capacity cannot be empty.”	Pass



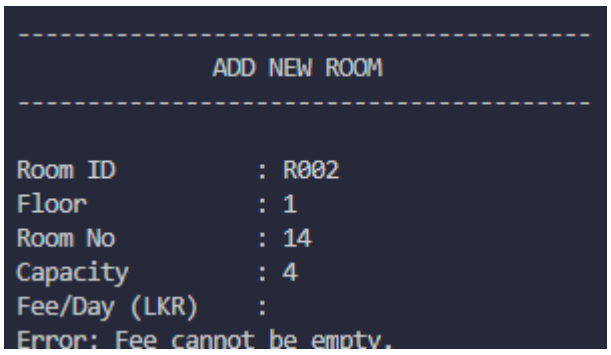
Test Case ID	Scenario	Input	Expected Output	Result
TC10	Non-numeric Capacity	3a	Shows: “Invalid input. Please enter digits only for capacity.”	Pass



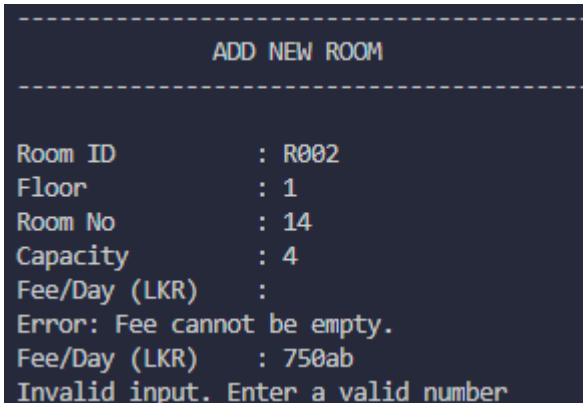
Test Case ID	Scenario	Input	Expected Output	Result
TC11	Zero Capacity	0	Shows: “Error: Capacity must be greater than zero.”	Pass



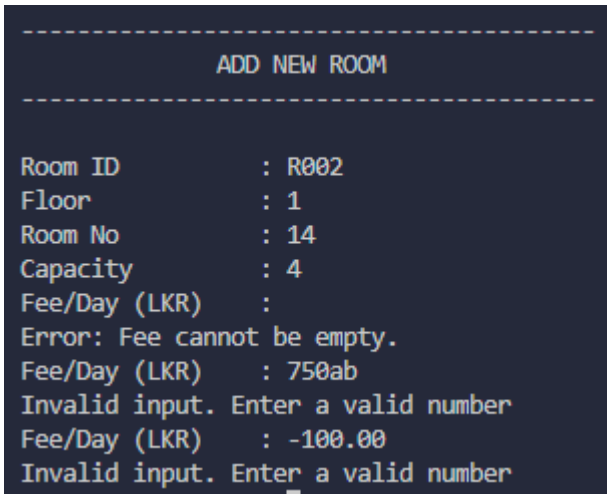
Test Case ID	Scenario	Input	Expected Output	Result
TC12	Empty Fee	(Press Enter)	Shows: “Error: Fee cannot be empty.”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC13	Invalid Fee (letters)	750ab	Shows: “Invalid input. Enter a valid number.”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC014	Negative Fee	-100.00	Invalid input. Enter a valid number.”	Pass



3.3.2 Update room

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid update for both fields	Room ID: R001 New Capacity: 6 New Fee/Day: 3000	Updates both values and shows “Room updated successfully!”	Pass

```
----------------------------------------
              UPDATE ROOM
----------------------------------------

Enter Room ID to update : R001
New Capacity (or - to skip)       : 6
New Fee/Day (LKR) (or - to skip)  : 3000.00

Room updated successfully!
----------------------------------------
Room ID          : R001
Floor            : 1
Room No          : 13
Capacity         : 6
Fee/Day (LKR)    : 3000.0
Available Beds   : 6
----------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Room ID	(Press Enter)	Shows “Error: Room ID cannot be empty.” and returns.	Pass

```
----------------------------------------
              UPDATE ROOM
----------------------------------------

Enter Room ID to update :
Error: Room ID cannot be empty. Please enter a valid Room ID.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC03	Non-existent Room ID	R005	Shows “Error: Room with ID 'R005' not found.” and returns	Pass

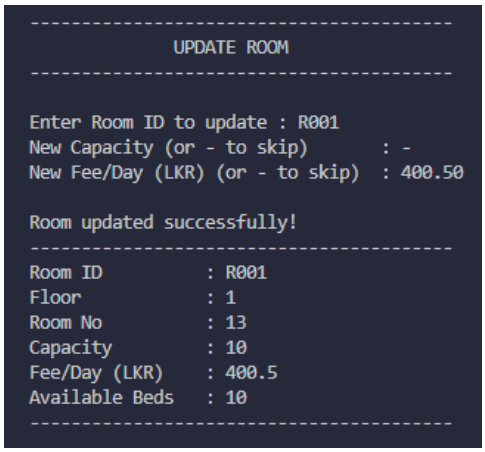
```
----------------------------------------
              UPDATE ROOM
----------------------------------------

Enter Room ID to update : R005
Error: Room with ID 'R005' not found in the system.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC04	Skip Capacity and Fee updates	Room ID: R001 Capacity: - Fee: -	Keeps old values and shows success message.	Pass

```
----------------------------------------
              UPDATE ROOM
----------------------------------------

Enter Room ID to update : R001
New Capacity (or - to skip)       : -
New Fee/Day (LKR) (or - to skip)  : -

Room updated successfully!
----------------------------------------
Room ID           : R001
Floor             : 1
Room No           : 13
Capacity          : 6
Fee/Day (LKR)     : 3000.0
Available Beds    : 6
----------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC05	Update only Capacity	Room ID: R001 Capacity: 10 Fee: -	Updates capacity; fee remains unchanged.	Pass

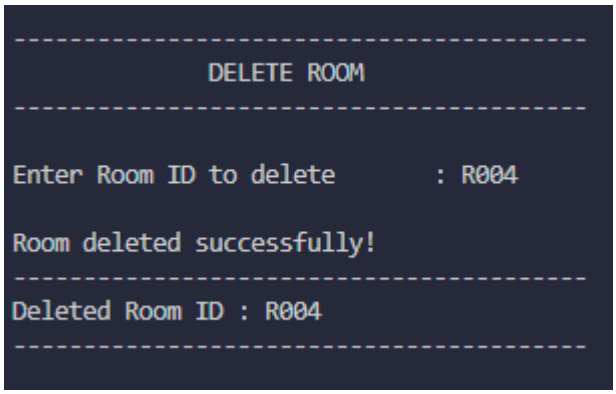
Test Case ID	Scenario	Input	Expected Output	Result
TC06	Update only Fee	Room ID: R001 Capacity: - Fee: 4000.50	Updates fee only; capacity remains unchanged.	Pass



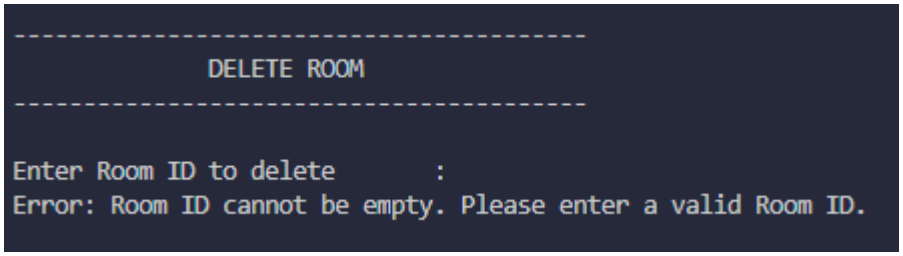
***Other Validations are same as add room**

3.3.3. Delete Room

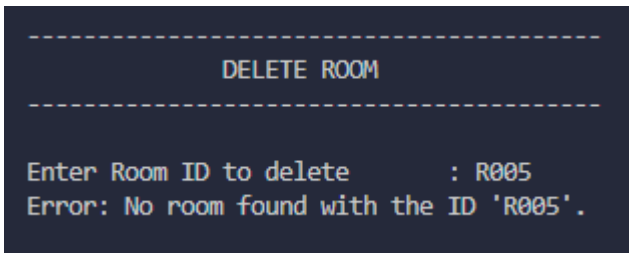
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Deletion – Room exists, no allocations	Room ID: R004	Deletes the room successfully. Shows: “Room deleted successfully!”	Pass



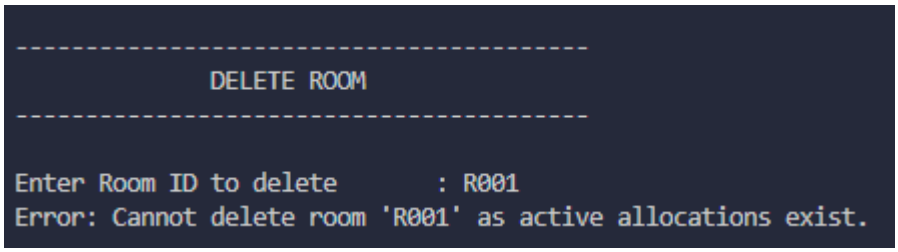
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Room ID	(Press Entry)	Displays: “Error: Room ID cannot be empty.”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC03	Non-existent Room ID	Room ID: R005	Displays: “Error: No room found with the ID 'R005'.”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC04	Room has active allocations	Room ID: R002 (exists in allocations)	Calls manageRooms() method → Displays room management menu.	Pass



3.3.4.Search Room

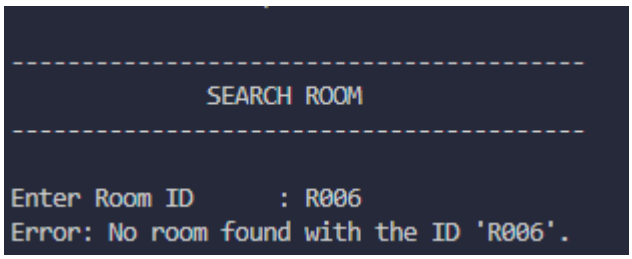
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Room ID	R001	Displays room details in formatted table: “Room found successfully!”	Pass

```
----------------------------------------
              SEARCH ROOM
----------------------------------------

Enter Room ID      : R001

Room found successfully!
----------------------------------------

Room ID    | Floor    | Room No    | Capacity   | Avail Beds   | Fee/Day (LKR)
-------------------------------------------------------------------------------
R001       | 1        | 13         | 10         | 10           | 400.5
-------------------------------------------------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Room ID	(Press Enter)	Shows: “Error: Room ID cannot be empty. Please enter a valid Room ID.”	Pass

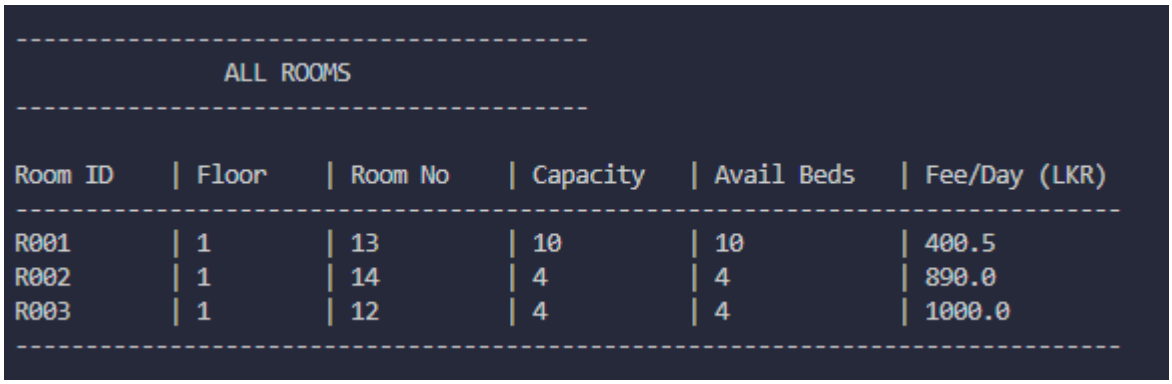
```
----------------------------------------
              SEARCH ROOM
----------------------------------------

Enter Room ID      :
Error: Room ID cannot be empty. Please enter a valid Room ID.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC03	Non-existent Room ID	R006	Shows: “Error: No room found with the ID 'R006'.”	Pass



3.3.5 View All Rooms

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Show all available rooms	-	Successfully show the available rooms	Pass



3.4 Manage Students

3.4.1 Add Student

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Student Record	ID = S001 Name = Kavindu Rajapaksha Contact = 0771234567 Email = kavindu@gmail.com	“Student added successfully!” and record stored.	Pass

```
----------------------------------------
            ADD STUDENT
----------------------------------------

Student ID       : S001
Name             : Kavindu
Contact No       : 0771234567
Email            : kavindu@gmail.com

Student added successfully!
----------------------------------------
Student ID       : S001
Name             : Kavindu
Contact No       : 0771234567
Email Address    : kavindu@gmail.com
Status           : Active
----------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Student ID	(Enter)	Error: Student ID cannot be empty	Pass

```
----------------------------------------
            ADD STUDENT
----------------------------------------

Student ID       :
Error: Student ID cannot be empty. Please enter a valid ID.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC03	Duplicate Student ID	S001	Error: Student ID already exists.	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID       :
Error: Student ID cannot be empty. Please enter a valid ID.

Student ID       : S001
Error: Student ID already exists. Please try another one.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC04	Empty Name	(Enter)	Error: Name cannot be empty.	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID       :
Error: Student ID cannot be empty. Please enter a valid ID.

Student ID       : S001
Error: Student ID already exists. Please try another one.

Student ID       : S002
Name             :
Error: Name cannot be empty. Please enter a valid name.
```

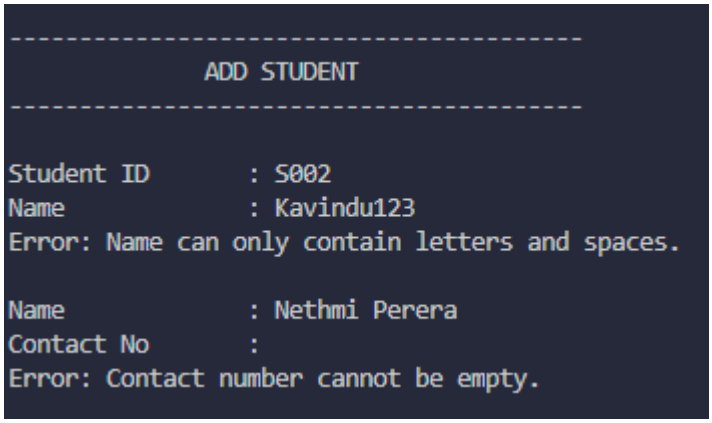
Test Case ID	Scenario	Input	Expected Output	Result
TC05	Name contains digits	Kavindu123	Error: Name can only contain letters and spaces.	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID       : S002
Name             : Kavindu123
Error: Name can only contain letters and spaces.
```

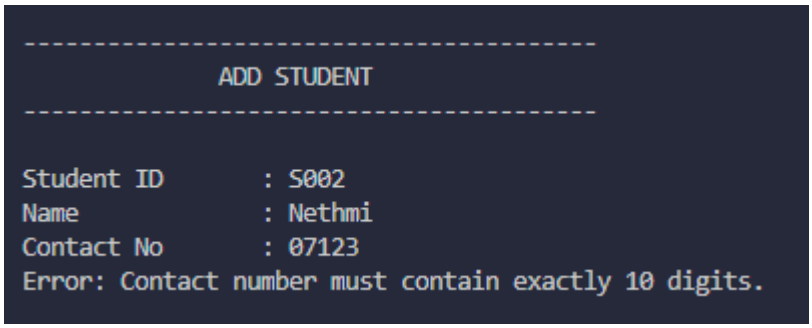
Test Case ID	Scenario	Input	Expected Output	Result
TC06	Valid Name with spaces	Nethmi Perera	Error: Contact number cannot be empty.	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID       : S002
Name             : Kavindu123
Error: Name can only contain letters and spaces.

Name             : Nethmi Perera
Contact No       :
```

Test Case ID	Scenario	Input	Expected Output	Result
TC07	Empty Contact	(Enter)	Error: Contact number cannot be empty.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC08	Contact too short	07123	Error: Contact number must contain exactly 10 digits	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC09	Contact contains letters	07A2345678	Error: Contact number must contain digits only	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID        : S002
Name              : Nethmi
Contact No        : 07123
Error: Contact number must contain exactly 10 digits.

Contact No        : 07A1234567
Error: Contact number must contain digits only.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC10	Duplicate Contact Number	0771234567	Error: This contact number already exists.”	Pass

```
----------------------------------------
              ADD STUDENT
----------------------------------------

Student ID        : S002
Name              : Nethmi
Contact No        : 07123
Error: Contact number must contain exactly 10 digits.

Contact No        : 07A1234567
Error: Contact number must contain digits only.

Contact No        : 0771234567
Error: This contact number already exists. Please try another.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC11	Contact length >10	077123456788	Contact number must contain exactly 10 digits.	Pass

```
----------------------------------------
             ADD STUDENT
----------------------------------------

Student ID       : S002
Name             : Nethmi
Contact No       : 07123
Error: Contact number must contain exactly 10 digits.

Contact No       : 07A1234567
Error: Contact number must contain digits only.

Contact No       : 0771234567
Error: This contact number already exists. Please try another.

Contact No       : 07122345678
Error: Contact number must contain exactly 10 digits.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC12	Empty Email	(Enter)	Error: Email cannot be empty	Pass

```
----------------------------------------
             ADD STUDENT
----------------------------------------

Student ID       : S002
Name             : Nethmi
Contact No       : 0711234567
Email            :
Error: Email cannot be empty. Please enter a valid email.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC13	Missing “@” or “.”	kavindu123gmailcom	Error: Invalid email format. Please include '@' and '.'”	Pass

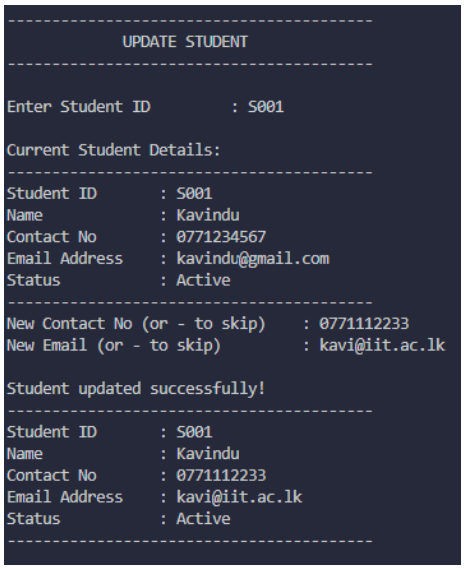
```
Student ID       : S002
Name             : Nethmi
Contact No       : 0711234567
Email            :
Error: Email cannot be empty. Please enter a valid email.

Email            : kaviindu123gmailcom
Error: Invalid email format. Please include '@' and '.'
```

Test Case ID	Scenario	Input	Expected Output	Result
TC14	Duplicate Email	kavindu@gmail.com	Error: This email already exists	Pass

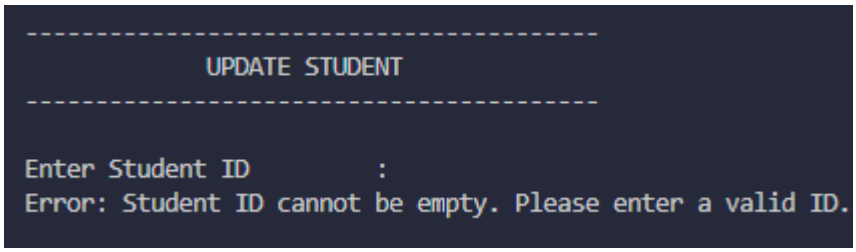
```
----------------------------------------
             ADD STUDENT
----------------------------------------

Student ID       : S001
Name             : kavindu
Contact No       : 0771234567
Email            : kavindu@gmail.com
Error: This email already exists. Please use another email.
```

3.4.2 Update Student

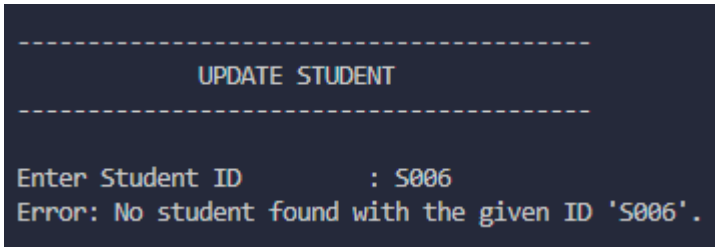
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid update for contact & email	Student ID: S001 New Contact: 0771112233 New Email: kavi@iit.ac.lk	Updates both successfully, displays “Student updated successfully!	Pass



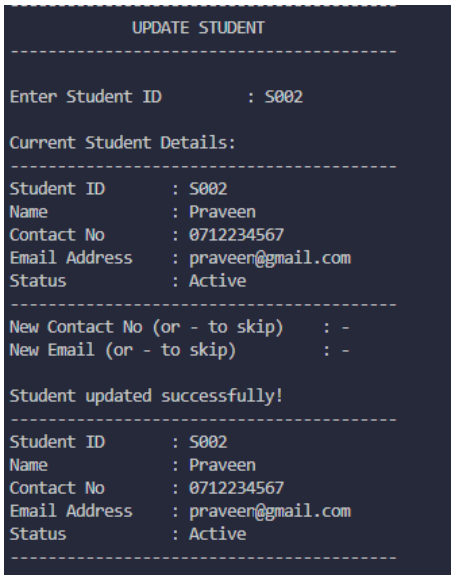
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Student ID	(Enter)	Error: Student ID cannot be empty.” and returns	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC03	Non-existent Student ID	S006	Error: No student found with the given ID 'S006'	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC04	Skip both updates	Student ID: S002 Contact: - Email: -	Keeps existing data unchanged; displays “Student updated successfully!”	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC05	Update only contact number	Student ID: S001 Contact: 0777654321 Email: -	Only contact updated; email remains same.	Pass

```
----------------------------------------
            UPDATE STUDENT
----------------------------------------

Enter Student ID        : S001

Current Student Details:
----------------------------------------
Student ID      : S001
Name            : Kavindu
Contact No      : 0771112233
Email Address   : kavi@iit.ac.lk
Status          : Active
----------------------------------------
New Contact No (or - to skip)    : 0777654321
New Email (or - to skip)         : -

Student updated successfully!
----------------------------------------
Student ID      : S001
Name            : Kavindu
Contact No      : 0777654321
Email Address   : kavi@iit.ac.lk
Status          : Active
----------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC06	Update only email	Student ID: S001 Contact: - Email: newemail@gmail.com	Only email updated; contact remains same.	Pass

```
----------------------------------------
            UPDATE STUDENT
----------------------------------------

Enter Student ID        : S001

Current Student Details:
----------------------------------------
Student ID      : S001
Name            : Kavindu
Contact No      : 0777654321
Email Address   : kavi@iit.ac.lk
Status          : Active
----------------------------------------
New Contact No (or - to skip)    : -
New Email (or - to skip)         : newemail@gmail.com

Student updated successfully!
----------------------------------------
Student ID      : S001
Name            : Kavindu
Contact No      : 0777654321
Email Address   : newemail@gmail.com
Status          : Active
----------------------------------------
```

*** Other validation are same as add student

3.4.3 Delete Student

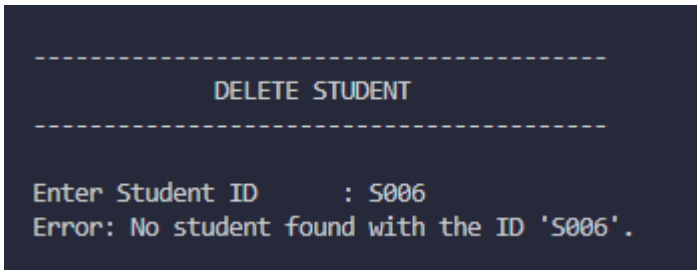
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Deletion – Student exists, no allocation	S003	Student deleted successfully; count decreases by one.	Pass

```
----------------------------------------
            DELETE STUDENT
----------------------------------------

Enter Student ID      : S003

Student deleted successfully!
----------------------------------------
Deleted Student ID : S003
----------------------------------------
```

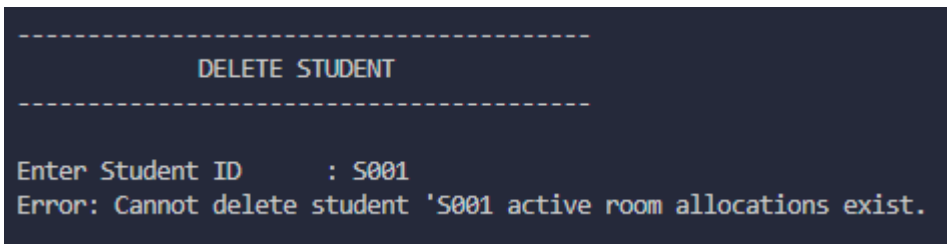
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Student ID	(Press Enter)	Shows “Error: Student ID cannot be empty.”	Pass

```
----------------------------------------
            DELETE STUDENT
----------------------------------------

Enter Student ID      :
Error: Student ID cannot be empty. Please enter a valid ID.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC03	Non-existent Student ID	S006	Displays “Error: No student found with the ID 'S006'.”.	Pass

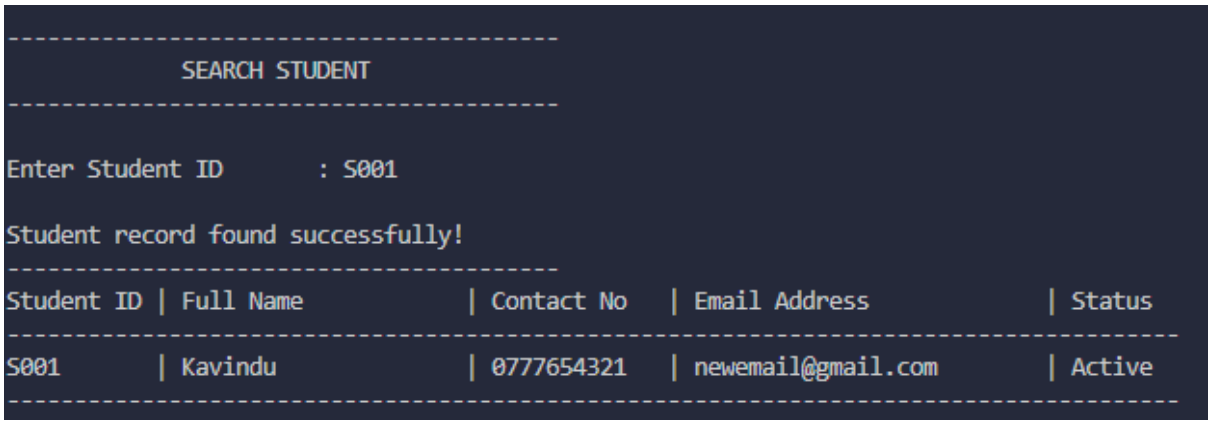


Test Case ID	Scenario	Input	Expected Output	Result
TC04	Student has active allocations	S001 (linked in allocations)	Displays “Error: Cannot delete student 'S002' active room allocations exist..	Pass

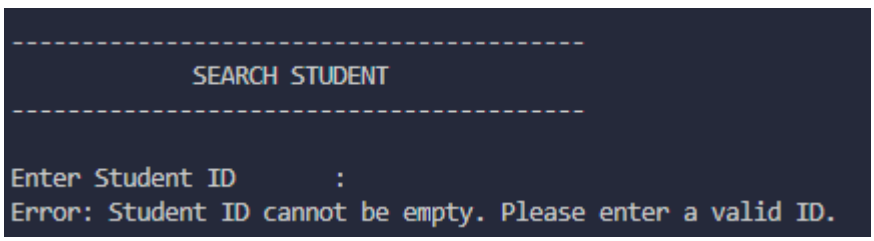


3.4.4 Search Student

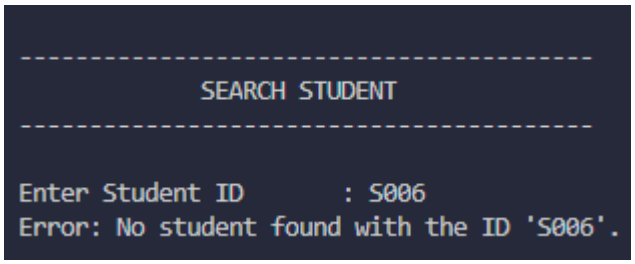
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Student ID (record exists)	S001	Displays: “Student record found successfully!” and shows details in formatted table..	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Student ID	(Press Enter)	Displays: “Error: Student ID cannot be empty. Please enter a valid ID.”.	Pass

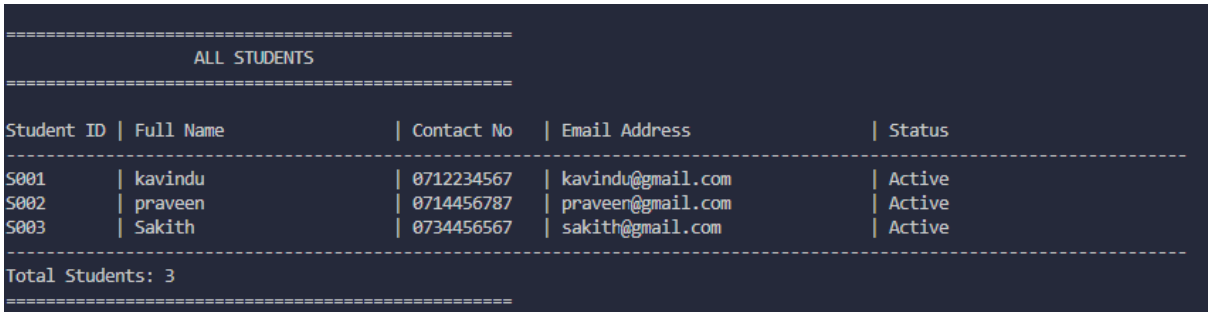


Test Case ID	Scenario	Input	Expected Output	Result
TC01	Non-existent Student ID	S006	Displays: “Error: No student found with the ID 'S006'.	Pass



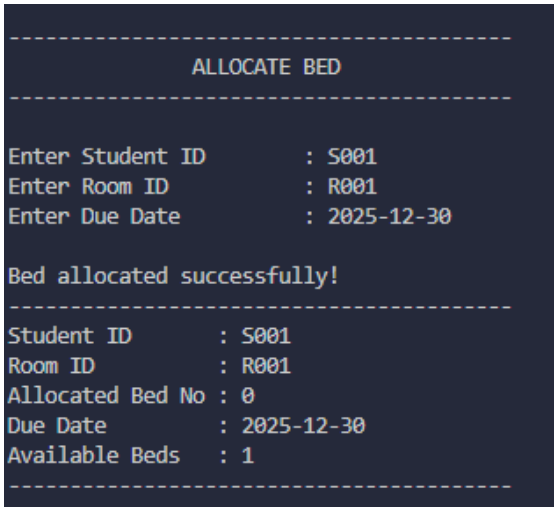
3.4.5. View all student

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Show all the student details		Successfully saved show the data	Pass

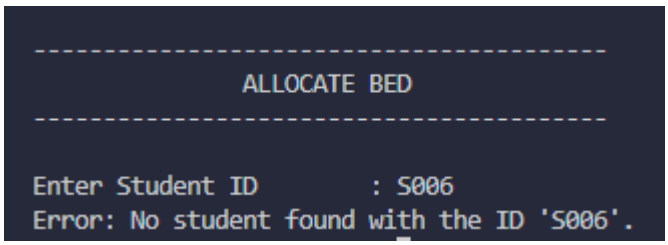


3.5 Allocated Bed

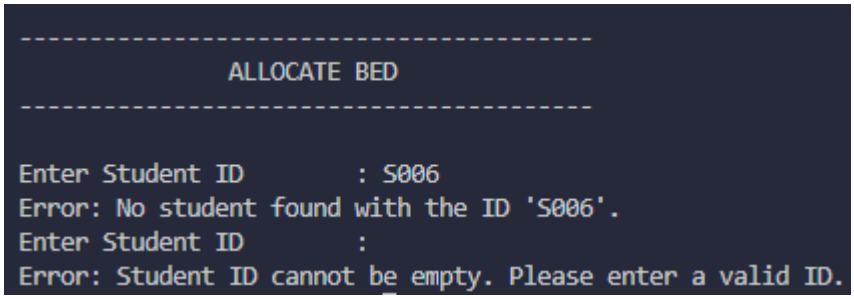
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid Allocation	Student ID = S001 Room ID = R001 Due Date = 2025-12-30	Bed allocated successfully. Bed count decreases by 1.	Pass



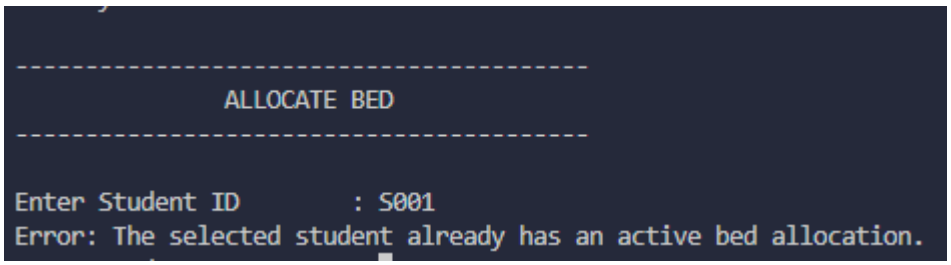
Test Case ID	Scenario	Input	Expected Output	Result
TC02	Non-existent Student ID	S006	Error: No student found with the ID 'S006'	Pass



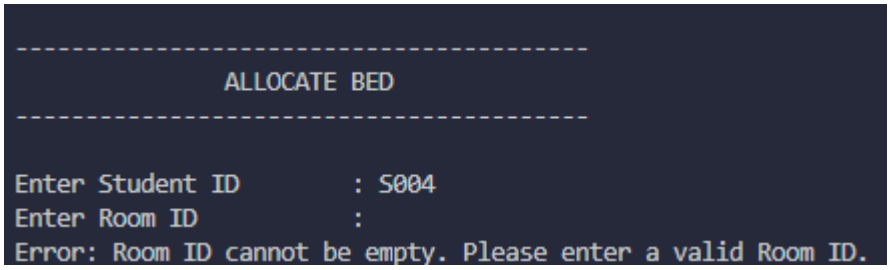
Test Case ID	Scenario	Input	Expected Output	Result
TC03	Empty Student ID	(Press Enter)	Error: Student ID cannot be empty	Pass



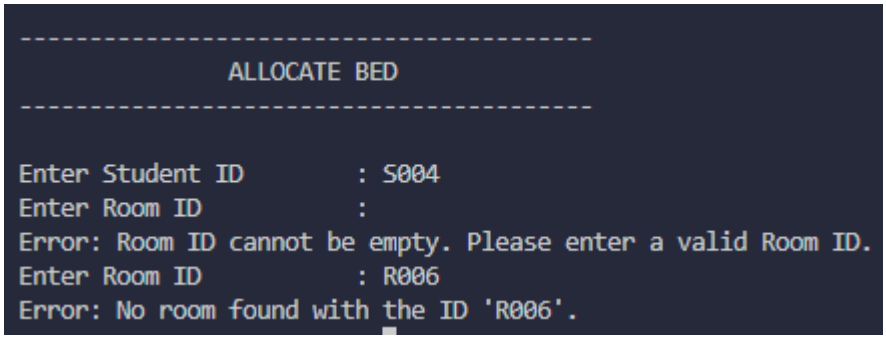
Test Case ID	Scenario	Input	Expected Output	Result
TC04	Student already allocated	Same student already in allocations	Error: The selected student already has an active bed allocation.”	Pass



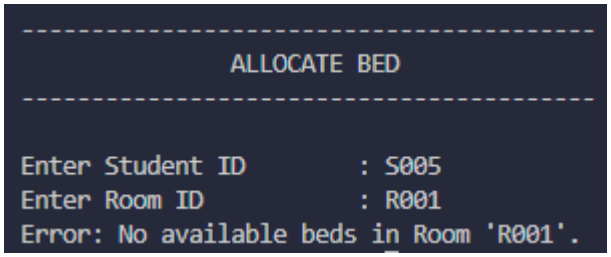
Test Case ID	Scenario	Input	Expected Output	Result
TC05	Empty Room ID	(Press Enter)	Error: Room ID cannot be empty	Pass



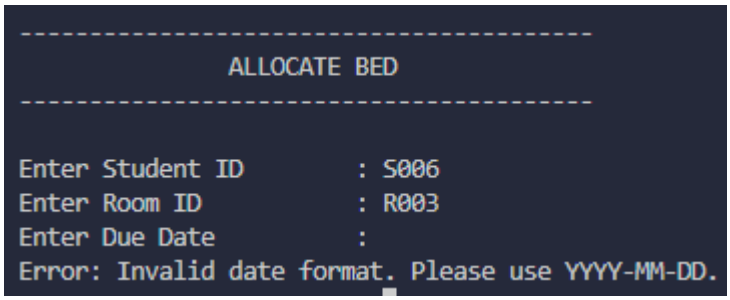
Test Case ID	Scenario	Input	Expected Output	Result
TC06	Non-existent Room ID	R006	Error: No room found with the ID 'R006'.”	Pass



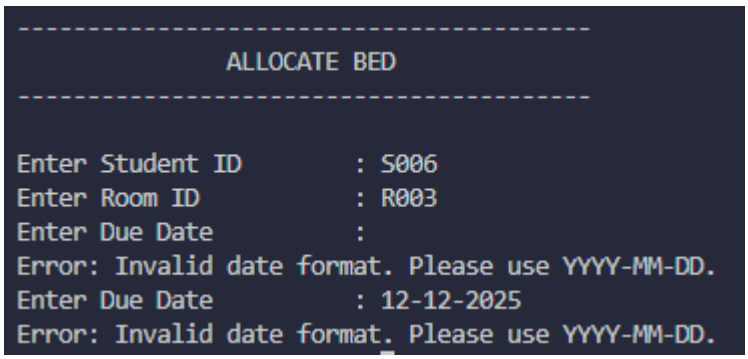
Test Case ID	Scenario	Input	Expected Output	Result
TC07	No available beds	ID- S005 RoomID: R001	Error: No available beds in Room 'R001'.	Pass



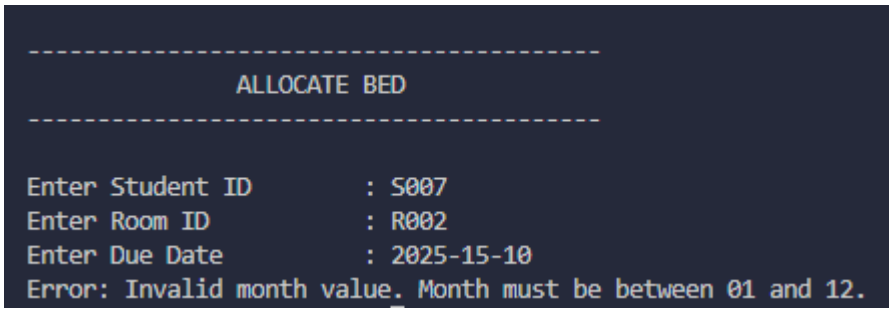
Test Case ID	Scenario	Input	Expected Output	Result
TC08	Empty Due Date	(Press Enter)	Error: Invalid date format. Please use YYYY-MM-DD.	Pass



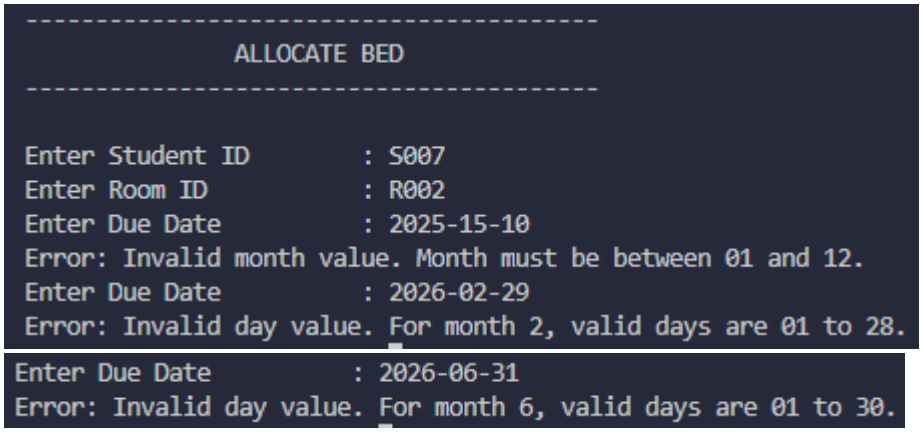
Test Case ID	Scenario	Input	Expected Output	Result
TC09	Wrong date format	12-12-2025	Error: Invalid date format. Please use YYYY-MM-DD.	Pass



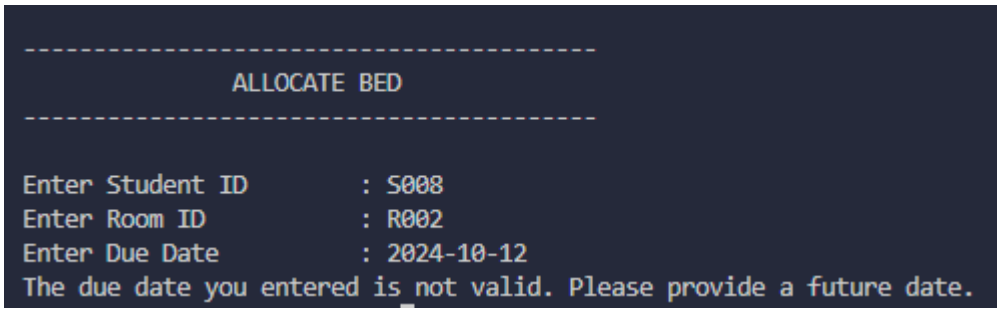
Test Case ID	Scenario	Input	Expected Output	Result
TC10	Invalid month value	2025-15-10	Error: Invalid month value. Month must be between 01 and 12.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC11	Invalid day value	2026-02-29 && 2025-06-31	Error: Invalid day value. For month 2, valid days are 01 to 28.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC12	Past date as due date	2024-10-12	The due date you entered is not valid. Please provide a future date.	Pass



3.6 Vacate Bed

Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid checkout (no overdue)	Student ID = S001 Room ID = R001	Displays success message, frees bed, increases available bed count by 1.	Pass

```
----------------------------------------
              VACATE BED
----------------------------------------

Enter Student ID                : S004
Enter Room ID                   : R003
Checkout completed successfully! The assigned bed has been freed.
------------------------------------------------------------
Student ID                      : S004
Room ID                         : R003
Bed Index                       : 2
Available Beds (R003)        : 2
------------------------------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC02	Empty Student ID	(Press Enter)	Error: Student ID or Room ID cannot be empty	Pass

```
----------------------------------------
              VACATE BED
----------------------------------------

Enter Student ID                :
Enter Room ID                   : R001
Error: Student ID or Room ID cannot be empty. Please enter valid values.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC03	Empty Room ID	(Press Enter)	Error: Student ID or Room ID cannot be empty	Pass

```
----------------------------------------
              VACATE BED
----------------------------------------

Enter Student ID               : S002
Enter Room ID                  :
Error: Student ID or Room ID cannot be empty. Please enter valid values.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC04	Allocation not found	Student ID = S009 Room ID = R001	Error: No active allocation found for Student 'S009' in Room 'R001	Pass

```
----------------------------------------
              VACATE BED
----------------------------------------

Enter Student ID               : S008
Enter Room ID                  : R001
Error: No active allocation found for Student 'S008' in Room 'R001'.
```

Test Case ID	Scenario	Input	Expected Output	Result
TC05	Valid overdue checkout	S001 + R001 (with dueDate before current date)	Displays overdue information block with fine calculation.	Pass

***Change your laptop date to a date later than the due date and check the overdue logic.

```
----------------------------------------
              VACATE BED
----------------------------------------

Enter Student ID               : S002
Enter Room ID                  : R001

----------------------------------------
Overdue Information
----------------------------------------
Overdue Days     : 68
Fee per Day (LKR): 900.0
Total Fine (LKR) : 61200.0
----------------------------------------

Checkout completed successfully! The assigned bed has been freed.
------------------------------------------------------
Student ID                     : S002
Room ID                        : R001
Bed Index                      : 2
Available Beds (R001)        : 1
```

Test Case ID	Scenario	Input	Expected Output	Result
TC06	Correct fine calculation	Fee per day = 900.00 Overdue days = 68	Fine = 61200 displayed.	Pass

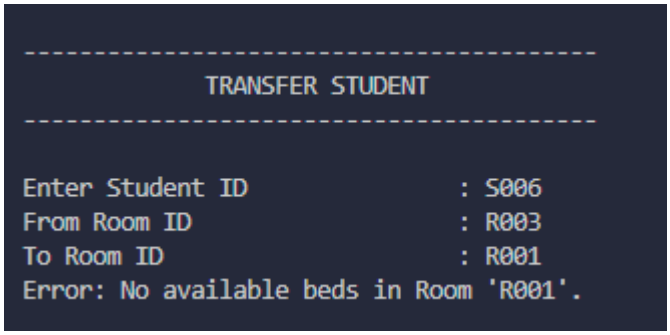
```
----------------------------------------
Overdue Information
----------------------------------------
Overdue Days     : 68
Fee per Day (LKR): 900.0
Total Fine (LKR) : 61200.0
----------------------------------------
```

3.7 Transfer Bed

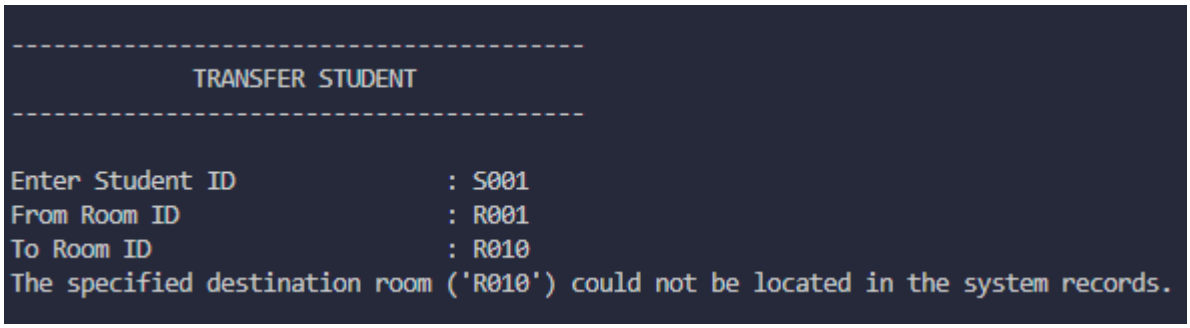
Test Case ID	Scenario	Input	Expected Output	Result
TC01	Valid transfer	Student ID: S001 From: R001 To: R002	Transfer successful. Allocation updated; available bed count adjusted in both rooms.	Pass

```
----------------------------------------
            TRANSFER STUDENT
----------------------------------------

Enter Student ID               : S005
From Room ID                   : R003
To Room ID                     : R001

Student transfer completed successfully!
----------------------------------------
Student ID       : S005
From Room        : R003
To Room          : R001
New Bed Index    : 1
Check-In Date    : 2025-10-16
Due Date         : 2025-10-27
Transfer Date    : 2025-10-16
----------------------------------------
Available Beds   :
 - R003 ? 3 beds remaining
 - R001 ? 0 beds remaining
----------------------------------------
```

Test Case ID	Scenario	Input	Expected Output	Result
TC02	No active allocation in source room	Student: S006 From: R001 To: R002	Error: No active bed allocation found for Student 'R006' in Room 'R001.	Pass



Test Case ID	Scenario	Input	Expected Output	Result
TC01	To room not found	Student: S001 From: R001 To: R888	The specified destination room ('R888') could not be located in the system records	Pass



3.8 View Reports

3.8.1. Occupancy Map

```
----------------------------------------
          OCCUPANCY MAP (Grid)
----------------------------------------
Room ID    Beds
----------------------------------------
R001       [S001] | [S002]
R002       [S007] | [S008] | [empty] | [empty]
R003       [S003] | [S004] | [S005] | [S006] | [empty]
----------------------------------------
```

3.8.2. Vacant Beds By Floor

```
----------------------------------------
         VACANT BEDS BY FLOOR
----------------------------------------
Floor    TotalRooms    TotalBeds    Occupied    Vacant
-------------------------------------------------------------
1        3             11           6           5
-------------------------------------------------------------
```

3.8.3. Students Per Room

```
----------------------------------------
            STUDENTS PER ROOM
----------------------------------------
Room       Count    Students
-------------------------------------------------
R001       1        S001
R002       2        S007,S008
R003       3        S003,S005,S006
```

3.8.4. Over Dues

```
----------------------------------------
            OVERDUE DUES
----------------------------------------
Student    Room       DaysOverdue  Fee/Day      EstimatedFine
-----------------------------------------------------------------
S001       R001       1            900.00       900.00
S003       R003       66           1200.00      79200.00
S005       R003       64           1200.00      76800.00
S006       R003       62           1200.00      74400.00
S007       R002       30           1000.00      30000.00
-----------------------------------------------------------------
```

3.8.5. Revenue Projection (Daily)

```
----------------------------------------
        REVENUE PROJECTION (Daily)
----------------------------------------
Total Estimated Daily Revenue: 6500.00 LKR
----------------------------------------
```

4. Conclusion

The HostelMate – University Hostel Management System was thoroughly evaluated to ensure its accuracy, reliability, and efficiency in handling core hostel operations such as student management, room management, and allocation processes. The testing process included a comprehensive set of functional and validation test cases covering various user scenarios, including valid, invalid, and boundary conditions.

All modules demonstrated strong input validation, data consistency, and logical accuracy, ensuring smooth interaction between components like *Students*, *Rooms*, and *Allocations*. The system effectively prevented data redundancy, handled exceptions gracefully, and maintained accurate real-time updates across all arrays and records.

The results confirm that the HostelMate system fulfills its intended objectives — providing a robust, user-friendly, and efficient digital solution for managing hostel activities. Its modular structure and clear validation logic make it both maintainable and scalable for future enhancements.

Overall, the project can be considered functionally complete and ready for deployment or further integration into a broader university management environment.