

# Cozy Haven Stay - Hotel Management System

## Business Logic Report

Submitted by: Monish Coumar S

Java & React Batch 2

Trainer: Madhu Kalla

### 1. System Overview

Cozy Haven Stay is a web-based hotel management system that facilitates hotel bookings and management through two primary user roles: Admin and User.

### 2. Core Business Entities

#### 2.1 User Management

Entity: User

- User ID (Primary Key)
- Username
- Email
- Password (Encrypted)
- Role (ADMIN/USER)

#### Business Rules:

- Usernames must be unique
- Email addresses must be unique
- Passwords are encrypted using BCrypt
- Two distinct roles: Admin and User
- Users can register and login
- Session management for logged-in users

#### 2.2 Hotel Management

Entity: Hotel

- Hotel ID (Primary Key)
- Name
- Location

- Description
- Rating (0-5)
- Amenities (List)
- Rooms (One-to-Many relationship)

**Business Rules:**

- Each hotel must have a unique name
- Rating must be between 0 and 5
- Hotels can have multiple amenities
- Hotels can have multiple rooms
- Only admins can create/edit/delete hotels
- Users can view and search hotels

**2.3 Room Management**

Entity: Room

- Room ID (Primary Key)
- Room Number
- Type (Standard, Deluxe, Suite, etc.)
- Price per night
- Description
- Amenities (List)
- Hotel (Many-to-One relationship)

**Business Rules:**

- Room numbers must be unique within a hotel
- Different room types available
- Price must be positive
- Rooms belong to a specific hotel
- Only admins can manage rooms
- Users can view room details

### **3. Business Processes**

#### **3.1 User Registration and Authentication**

Process Flow:

##### **1. User Registration**

- User enters details (username, email, password, role)
- System validates unique username and email
- Password is encrypted
- Account is created

##### **2. User Login**

- User enters credentials
- System validates credentials
- Session is created
- Redirect based on role (Admin/User dashboard)

#### **3.2 Hotel Management Process**

Admin Operations:

##### **1. Create Hotel**

- Enter hotel details
- Add amenities
- Set initial rating
- Save to database

## 2. Edit Hotel

- Modify hotel information
- Update amenities
- Change rating
- Save changes

## 3. Delete Hotel

- Remove hotel and associated rooms
- Handle cascading deletes

### User Operations:

## 1. View Hotels

- List all hotels
- View hotel details
- See available rooms

## 2. Search Hotels

- Search by location
- Filter by rating
- View amenities

### 3.3 Room Management Process

#### Admin Operations:

## 1. Add Room

- Select hotel
- Enter room details
- Set price

- Add amenities
- Save room

## 2. Edit Room

- Modify room information
- Update price
- Change amenities
- Save changes

## 3. Delete Room

- Remove room from hotel
- Update hotel inventory

## User Operations:

### 1. View Rooms

- See room details
- Check availability
- View amenities
- Compare prices

## 4. Security and Access Control

### 4.1 Role-Based Access

#### 1. Admin Access:

- Full hotel management
- Room management
- User management
- Dashboard access

#### 2. User Access:

- View hotels
- Search hotels
- View rooms
- Personal dashboard

## 4.2 Security Measures

### 1. Password Security:

- BCrypt encryption
- Secure password storage
- Password validation

### 2. Session Management:

- Secure session handling
- Session timeout
- Logout functionality

### 3. Access Control:

- Role-based authorization
- Protected endpoints
- Secure redirects

## 5. Data Validation Rules

### 5.1 User Data Validation

- Username: Required, unique
- Email: Required, valid format, unique
- Password: Required, minimum length
- Role: Required, valid role type

### 5.2 Hotel Data Validation

- Name: Required, unique
- Location: Required
- Description: Required
- Rating: Required, 0-5 range
- Amenities: Optional, list of strings

### **5.3 Room Data Validation**

- Room Number: Required, unique per hotel
- Type: Required, valid room type
- Price: Required, positive number
- Description: Required
- Amenities: Optional, list of strings

## **6. Business Rules Implementation**

### **6.1 Service Layer Implementation**

#### **1. UserService:**

- User registration with validation
- Secure login process
- Role-based access control
- Session management

#### **2. HotelService:**

- Hotel CRUD operations
- Search and filter functionality
- Amenity management
- Rating management

#### **3. RoomService:**

- Room CRUD operations
- Price management

- Room type handling
- Hotel-room relationship management