

❖ **Acknowledgement**

- I would like to express our deepest appreciation to all those who provided us the possibility to complete this report. A special gratitude I give to our project guide, [Ameem Ma'am], whose contribution in stimulating suggestions and encouragement helped us to coordinate our project, especially in writing this report.
- Furthermore, i would also like to acknowledge with much appreciation the crucial role of the staff of Om Vindhyavasini College of I.T. & Management, who gave the permission to use all required equipment and the necessary materials to complete the task.
- Lastly, i would like to thank our families and friends for their support and encouragement throughout this project.

INDEX

No.	Particulars	Page
1)	Project Profile <ul style="list-style-type: none">• Project Definition• Process of System• System Requirement	
2)	Analysis <ul style="list-style-type: none">• Limitation of Existing System• Scope of the project• Feasibility Study• Use Case Diagram• Entity Relationship Diagram• Data Flow Diagram	
3)	Design <ul style="list-style-type: none">• Data Dictionary	
4)	Implementation and Testing <ul style="list-style-type: none">• Screen Layouts• Testing• Future Enhancement	
5)	Tools and Technology <ul style="list-style-type: none">• What is python.?• What is MYSQL?• What is Android?• Bibliography	

1.Project Profile

1.1 Project Definition

The **Grand Hotel Management System** is a web-based application designed to streamline hotel operations and enhance the user experience for both guests and hotel staff. The system provides functionalities for:

- **Guests:**
 - Viewing room categories and detailed descriptions to make informed decisions.
 - Checking real-time availability of rooms, making reservations, and managing personal profiles for future bookings and preferences.
- **Admins:**
 - Comprehensive management tools for overseeing room details, reservations, user roles, and account security, including the ability to block or unblock users as necessary.
 - Ensuring real-time updates to maintain an accurate representation of room availability and reservation status.

1.2 Process of System

1.2.1 User Interaction:

- **Browsing:** Guests visit the website to explore room categories and view details.

- **Reservation:** Registered users can select room categories, room id, select check-in and check-out dates, and complete the booking form.
- **Profile Management:** Registered users can log in to view their profiles, past bookings, and make new reservations.

1.2.2 Admin Operations:

- **Login:** Admins log into a secure admin panel for system access, ensuring that sensitive operations are protected.
- **Room Management:** Admins can add, update, or delete room , including changing rates and availability in real-time, ensuring accurate representation on the user-facing interface.
- **Reservation Management:** Admins handle all reservations, updating room availability based on confirmed bookings and cancellations, thereby managing operational flow efficiently.
- **User Management:** Admins oversee user roles, ensuring appropriate access levels, block/unblock users when necessary, and manage user accounts for security and compliance.

1.2.3 Reservation Workflow:

- Users select a room category, check available rooms, choose dates, and complete the booking, with real-time validation to enhance user experience.

- Reservations are recorded in the database, and room availability is updated in real-time to prevent overbooking and ensure accurate data handling.

1.3 System Requirements

1.3.1 Hardware:

- **Processor:** Intel core i3 or higher.
- **RAM:** 4GB or more.
- **Hard Disk:** 100GB or more we suggest SSD for smoother access.
- **Client:** Standard web browsers (Chrome) on user devices.

1.3.2 Software:

- **Web Server:** Apache.
- **Database Server:** MySQL.
- **Programming Language:** Python.
- **Framework:** Flask (for Python).
- **Frontend:** HTML, CSS, JavaScript.
- **Dependencies:** jQuery, Bootstrap, SweetAlert.

1.3.3 Network:

- Reliable internet connectivity for accessing the web application.

2. Analysis

2.1 Limitation of Existing System

- **Manual Processing:** Existing systems may rely on manual processes for reservations and room management.
- **Limited Integration:** Lack of real-time updates and integration with other systems.
- **User Experience:** Inefficient user interfaces and poor user experience in existing systems.

2.2 Scope of the Project

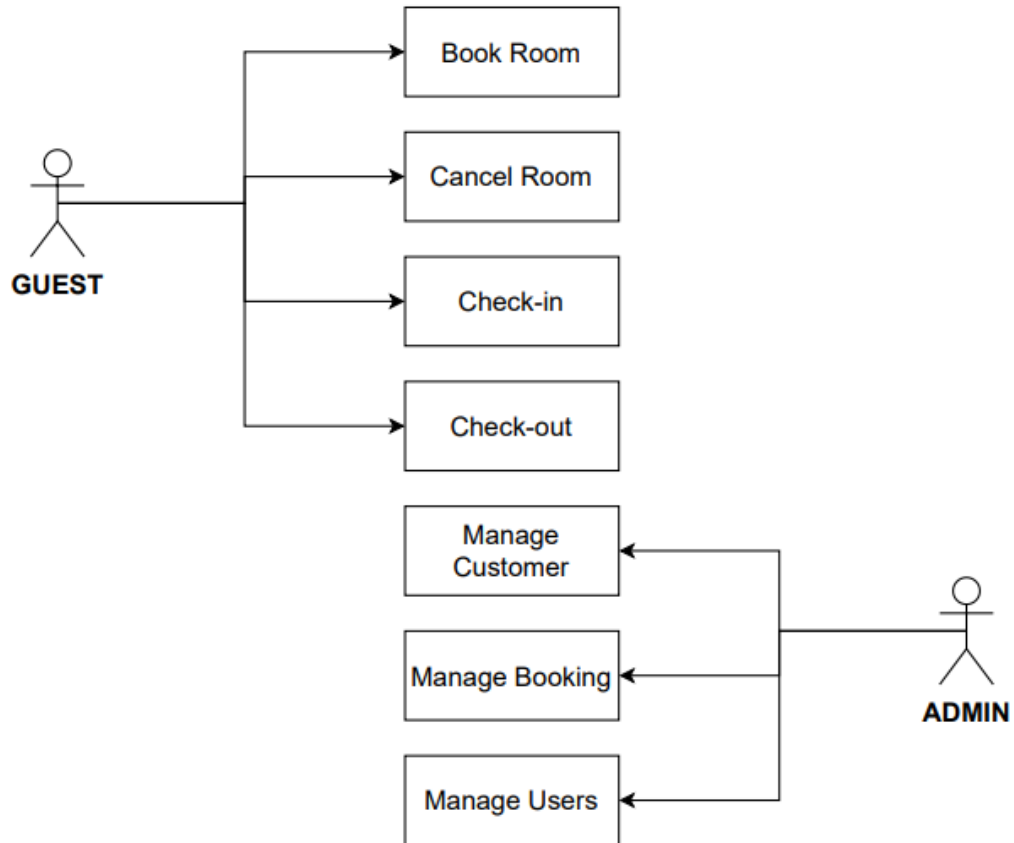
- **Room Management:** Add, update, and delete room details.
- **Reservation System:** Handle room bookings, availability checks, and manage user reservations.
- **User Management:** Admin functions for user role management and account control.

2.3 Feasibility Study

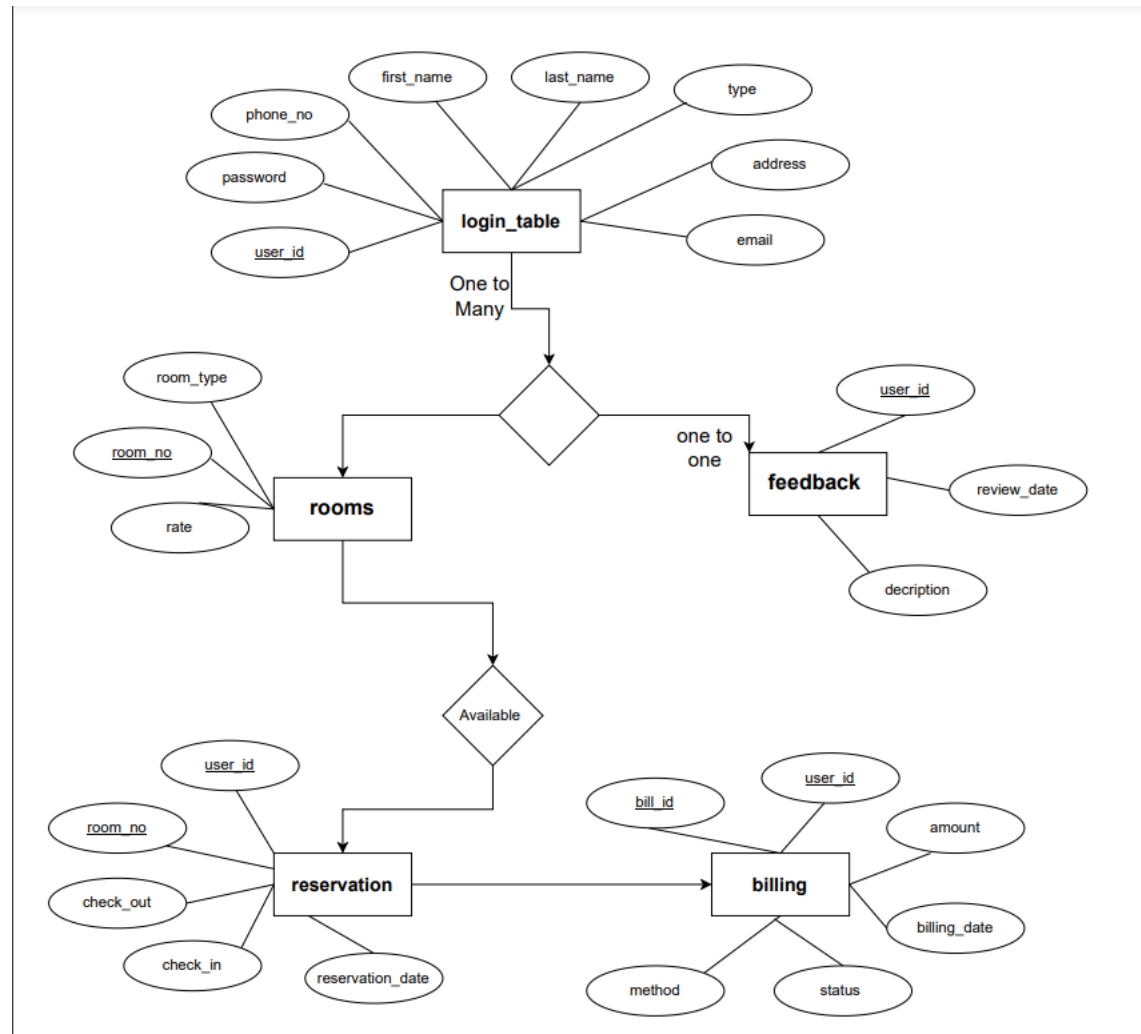
- **Technical Feasibility:** The use of Python, MySQL, and web technologies is technically sound and widely supported.
- **Operational Feasibility:** The system will improve operational efficiency and user experience.

- **Economic Feasibility:** Cost of development and maintenance is justified by the benefits of automation and improved functionality.

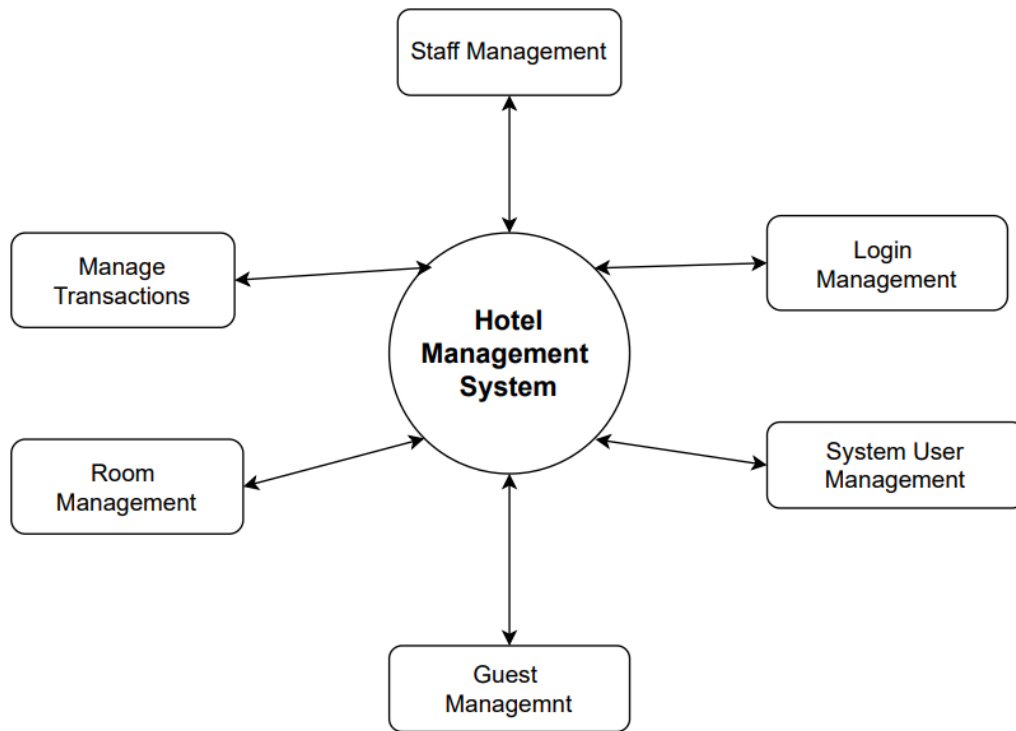
2.4 Usecase Diagram



2.5 Entity Relationship Diagram



2.6 Data Flow Diagram (Level-0)



3. Design

3.1 Data Dictionary

- **Login Table:**

Column	Datatype	Size	Constraint
User_id	Int	-	Primary Key
First_name	Varchar	50	Not null
Last_name	Varchar	50	-
Password	Varchar	50	Not null
Email	Varchar	50	Not null
Phone_no	Varchar	20	Not null
Address	Varchar	255	Not null
Is_admin	Boolean	-	-

- **Reservation Table:**

Column	Datatype	Size	Constraint
Reservation_id	Int	11	Primary Key
User_id	Int	11	Foreign key
Room_id	Varchar	30	Foreign key
Reservation_date	Date	-	Not null
Room_type	Varchar	50	Not null
Check_in_date	Date	-	Not null
Check_out_date	Date	-	Not null
Notes	Text	-	Not null

- **Contact Form table:**

Column	Datatype	Size	Constraint
contact_id	Int	11	Primary key
Name	Varchar	50	Not null
Phone	Varchar	20	Not null
Email	Varchar	255	Not null
Message	Text	-	Not null
Submitted_at	Timestamp	-	-

- **Room Table:**

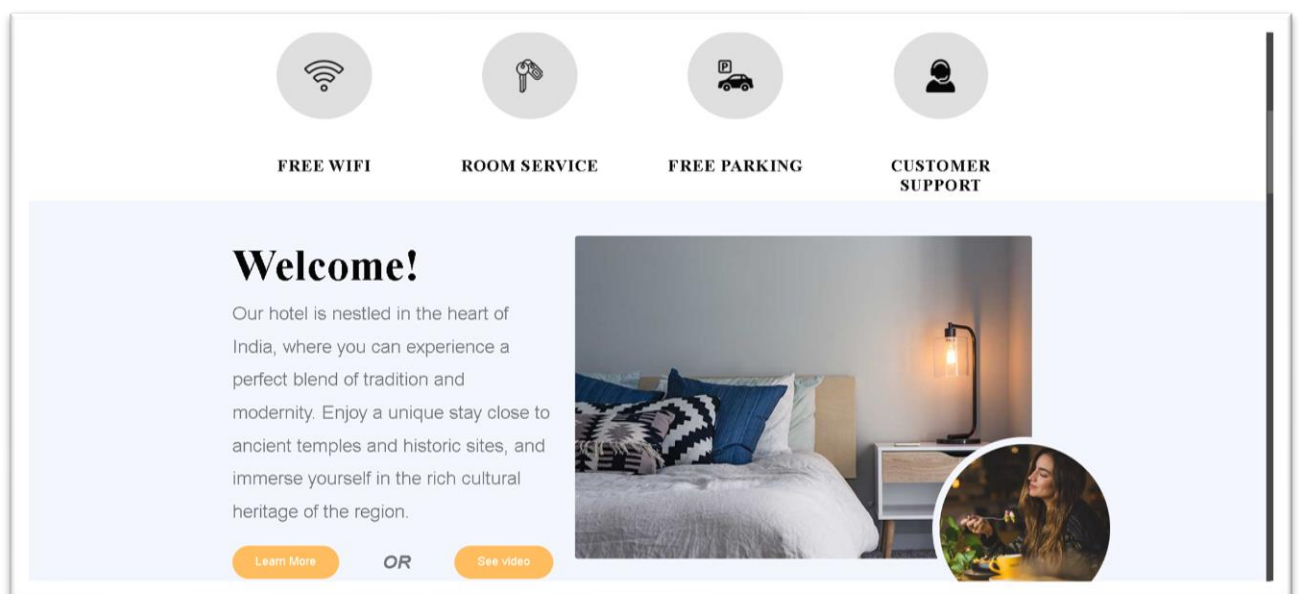
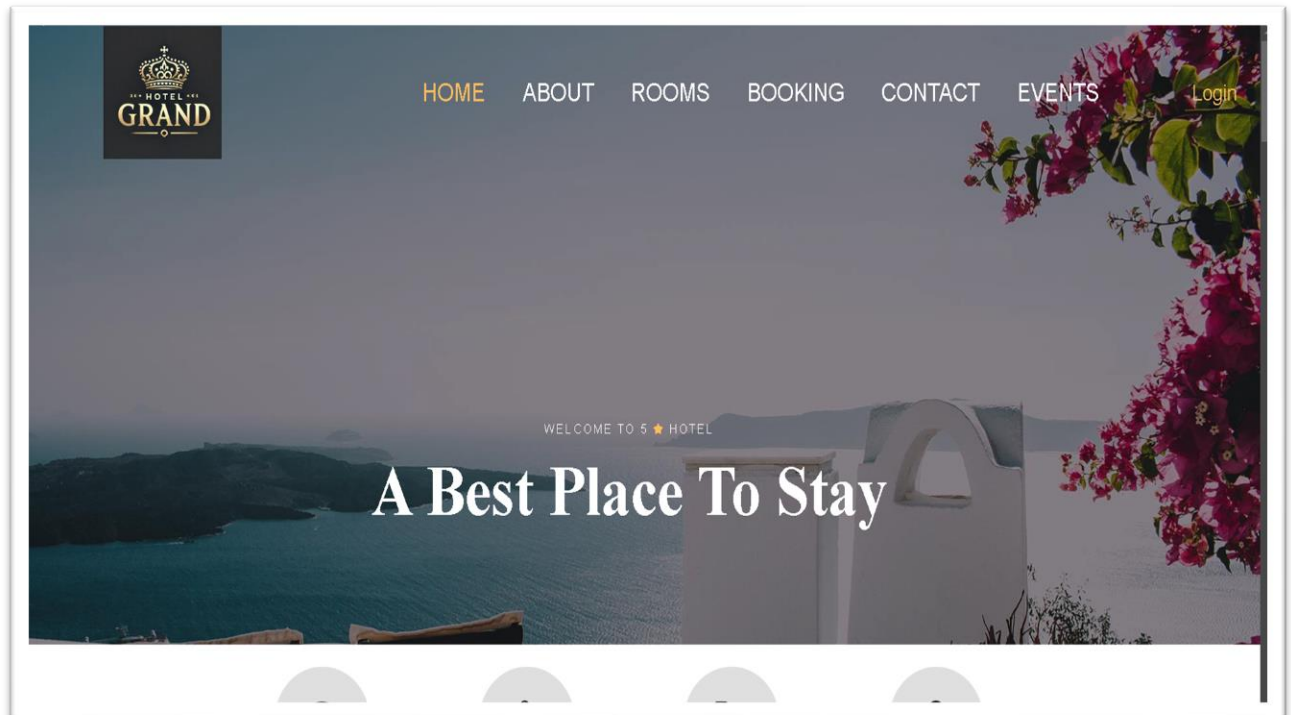
Column	Datatype	Size	Constraint
Room_id	Varchar	30	Primary key
Room_type	Varchar	50	Not null
Rate	Decimal	-	Not null

4.Implementation and Testing

4.1 Screen Layouts

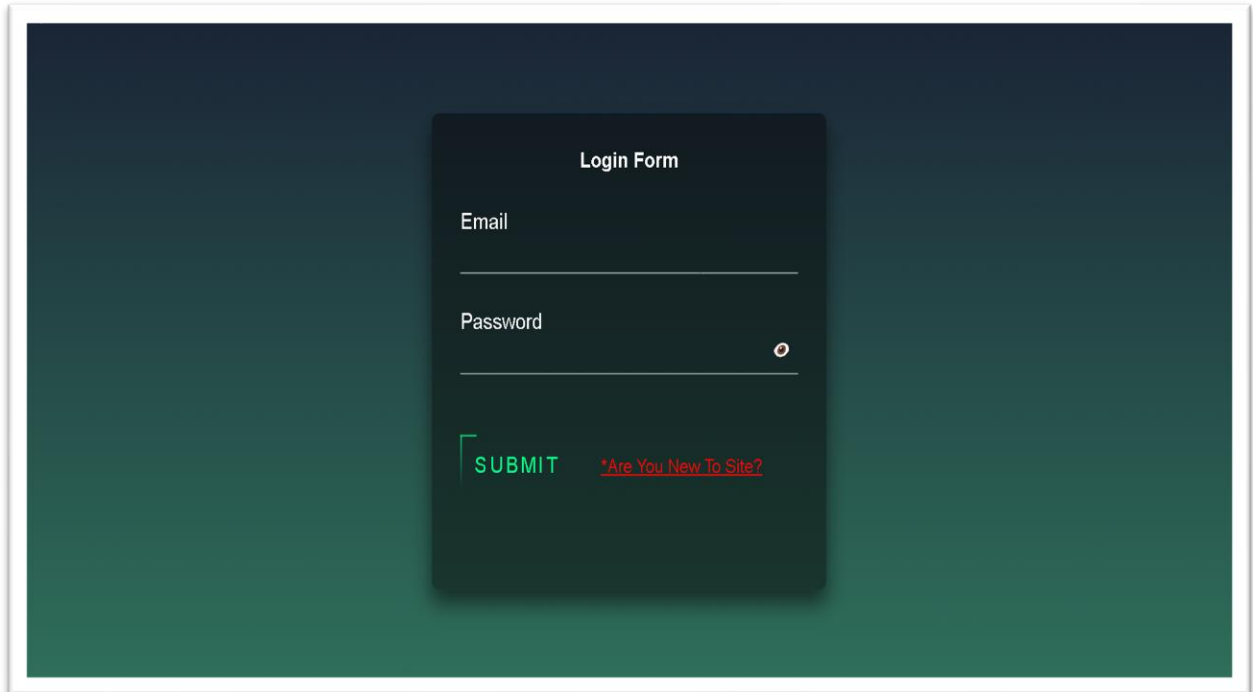
4.1.1. Homepage

- **Hotel Images Display:** Display images of rooms and amenities to attract visitors' attention.



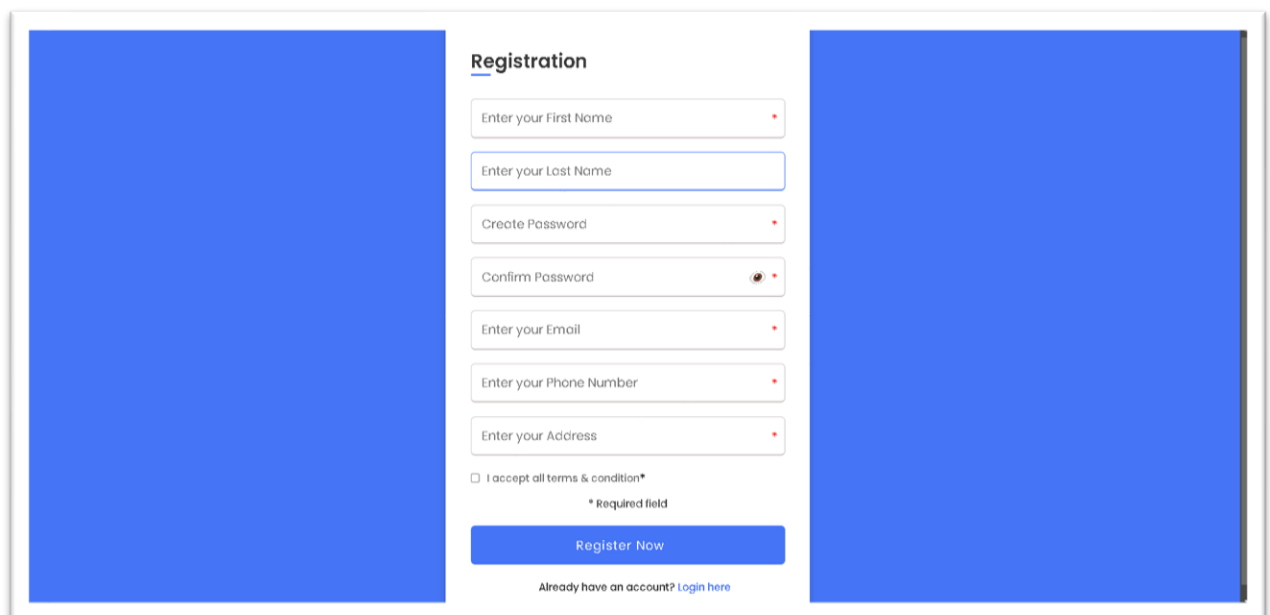
4.1.2. Login Page

- **Login page:** users can login if they are registered.



A dark-themed login form titled "Login Form" is centered on a dark green background. The form contains two input fields: "Email" and "Password". The "Password" field has a toggle icon (an eye) to its right. Below the input fields, there is a green "SUBMIT" button and a red link that says "*Are You New To Site?".

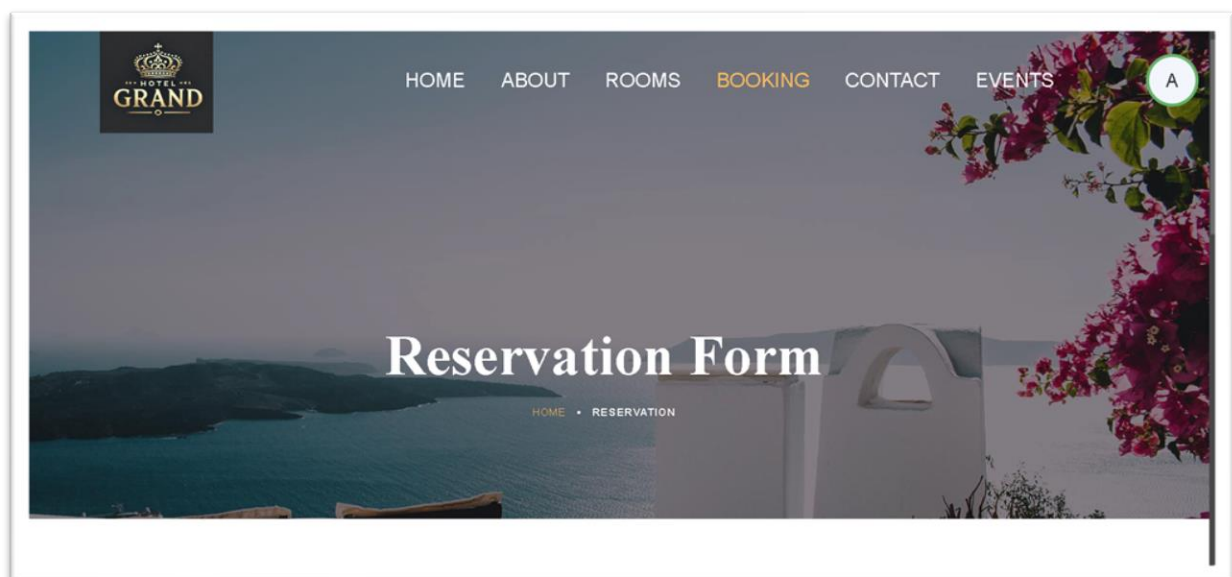
- **Registration page:** New users can register and after they can login.



A light-themed registration form titled "Registration" is centered on a white background. The form contains several input fields: "Enter your First Name", "Enter your Last Name", "Create Password", "Confirm Password", "Enter your Email", "Enter your Phone Number", and "Enter your Address". Each input field has a red asterisk (*) indicating it is a required field. Below the input fields, there is a checkbox labeled "I accept all terms & condition*" and a red asterisk (*) indicating it is a required field. Below the checkbox, there is a blue "Register Now" button. At the bottom of the form, there is a link that says "Already have an account? Login here".

4.1.3. Reservation Form

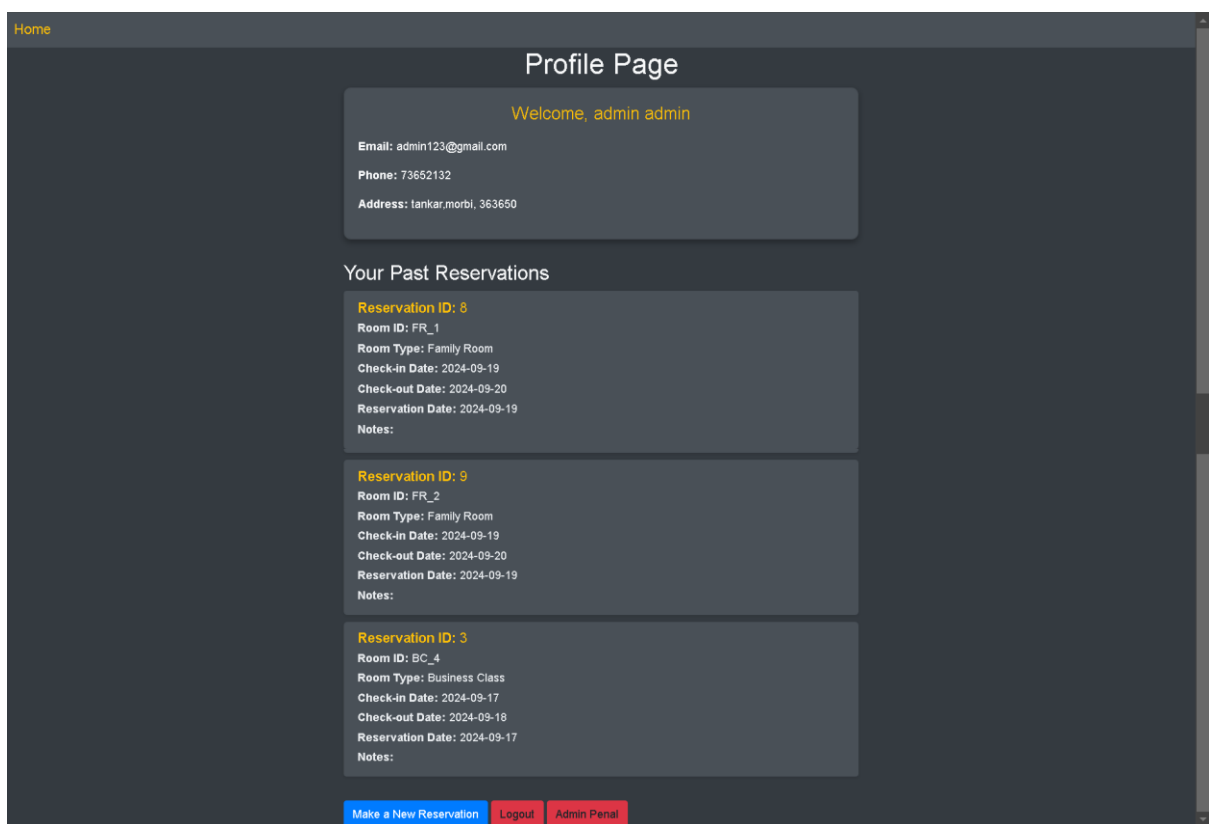
- **Room Selection and Date Pickers:** Users can select their desired room type, input check-in and check-out dates, and view available room options. The form dynamically updates based on the room category selected to show available room numbers.
- **Booking Details and Summary:** Includes fields for a summary of the booking (room type, dates, price). A "Book Now" button confirms the reservation, guiding users to confirmation.
- **Login:** Users must be logged in to book a room. If not it will redirect to login page.



Reservation Date	
<input type="text" value="2024-09-24"/>	
Room Type	Room No
<input type="text" value="Select Room Type"/>	<input type="text" value="Select Room No"/>
Date Check In	Date Check Out
<input type="text"/>	<input type="text"/>
Notes	
<input type="text"/>	
<input type="button" value="Reserve Now"/>	

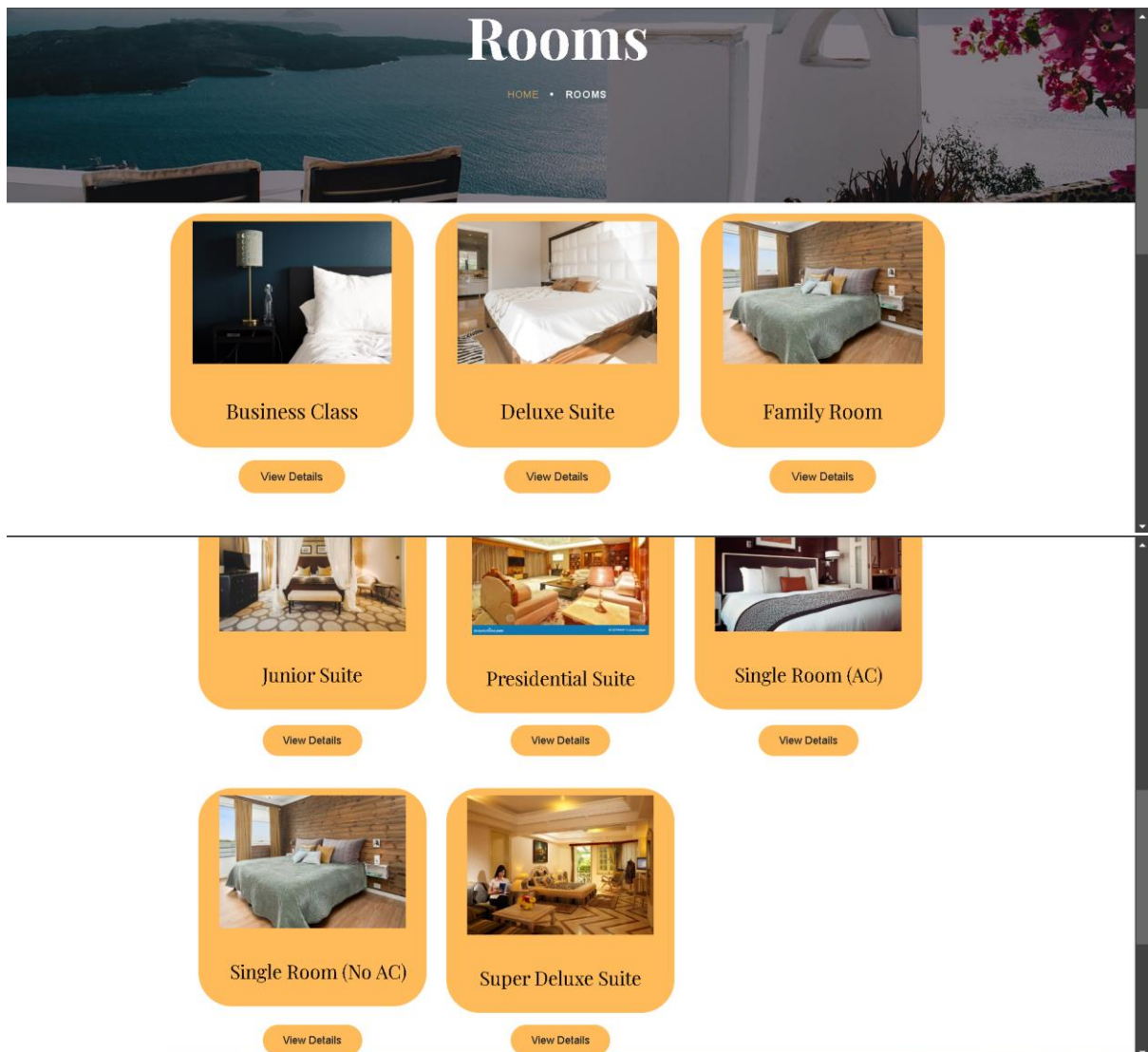
4.1.4. Profile Page

- **User Information:** Displays personal details like name, email, and phone number.
- **Past Bookings:** Lists all previous bookings with details like reservation date, room type, and stay duration. Users can view past stays or manage future reservations directly from their profile.



4.1.5. Rooms Page







- **Room Display:** A visually engaging presentation of all room categories, each with high-quality images, descriptions, amenities, and price details. This page helps users explore different options before booking.
- **Detailed Room Information:** Clicking on a View Details Button of room type reveals more specific details, including room size, bed options, and included services. This section helps users make informed choices by providing all necessary information upfront.





Business Class

Our Business Class rooms are designed to cater to all your professional needs while providing a comfortable stay.

-  King-size bed
-  Work desk and chair
-  Complimentary breakfast
-  High-speed internet
-  Access to the executive lounge
-  Television with satellite channels

Price: Rs. 5000.00 per night

[Book Now](#)

4.1.6. Contact Page

- **Guest Review:** Shows guests reviews.
- **Inquiry Form:** A simple form allowing users to send messages or requests directly to the hotel. Fields typically include name, email, phone, and message, with a "Send message" button to send inquiries.

Full Name Phone

Email

Write Message

[Send Message](#)

Om Shanti English Medium Sc...
Opp. Sardar Baug, Shanala Road,
Punt Nagar, Morbi, Gujarat 363641
4.0 ★★★★★ 310 reviews
[View larger map](#)

Map showing the location of Om Shanti English Medium School and surrounding areas in Morbi, Gujarat. Landmarks include Sardar Baug, Om VVM College, State Bank of India, and various local businesses.

What Our Guests Say



"I came here for a business trip, and I was pleasantly surprised by the peaceful environment and excellent service. Will definitely return!"

— Ananya Pandey



"The resort exceeded my expectations. The staff went out of their way to ensure we had a memorable stay. The food was incredible!"

— Jatin Patel



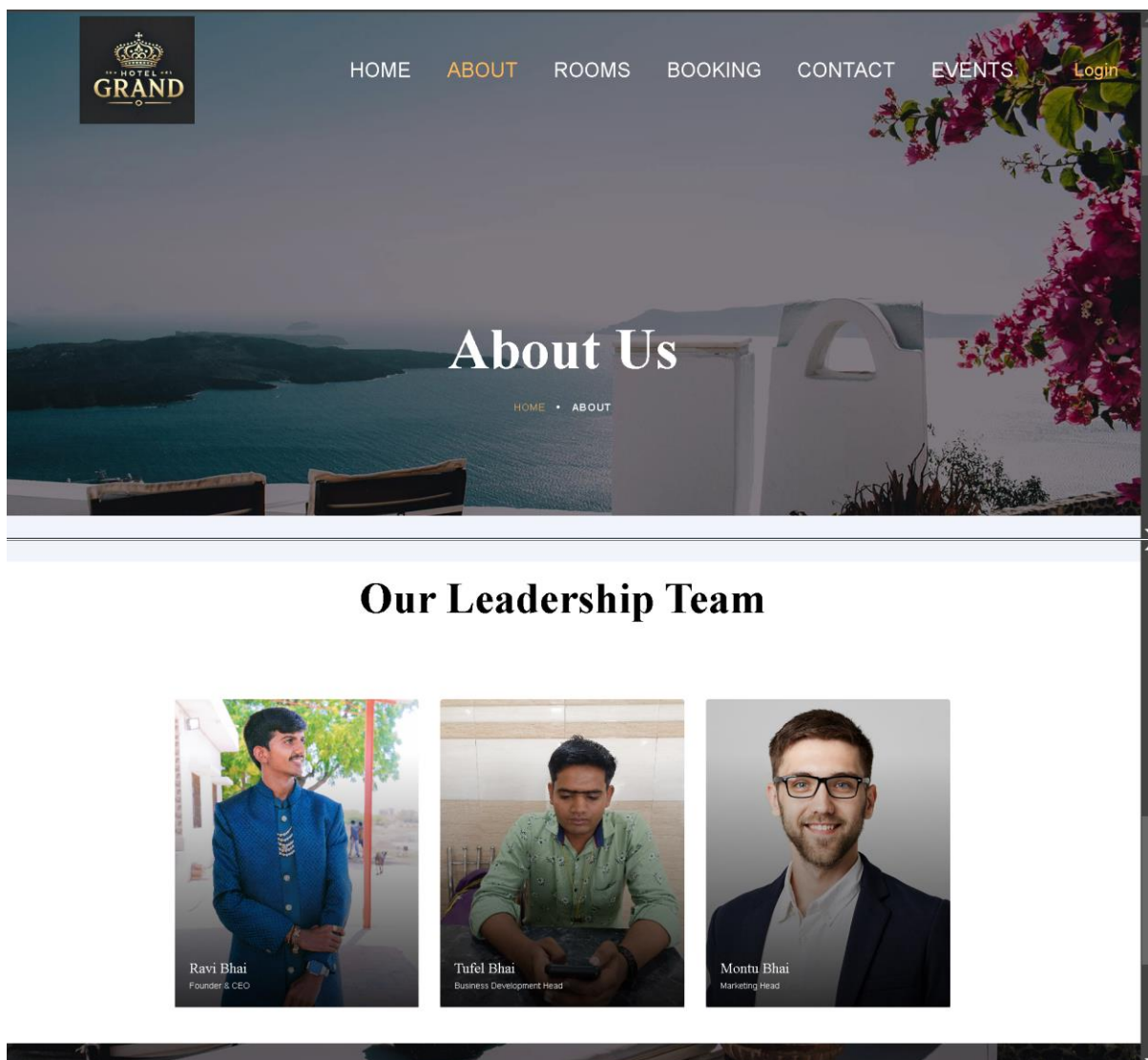
"We had an amazing family vacation here. The rooms were spacious, and there were plenty of activities for the kids. We will definitely be back!"

— Abhishek Patel

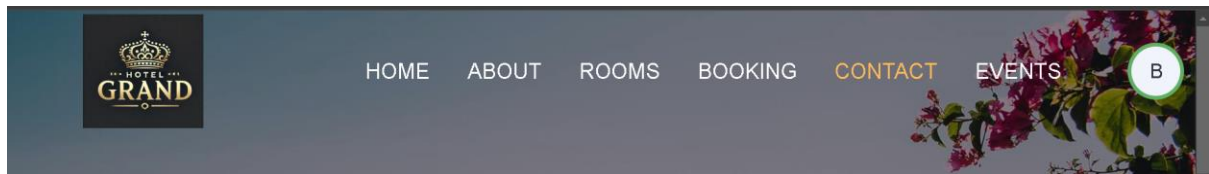


4.1.7. About Us Page

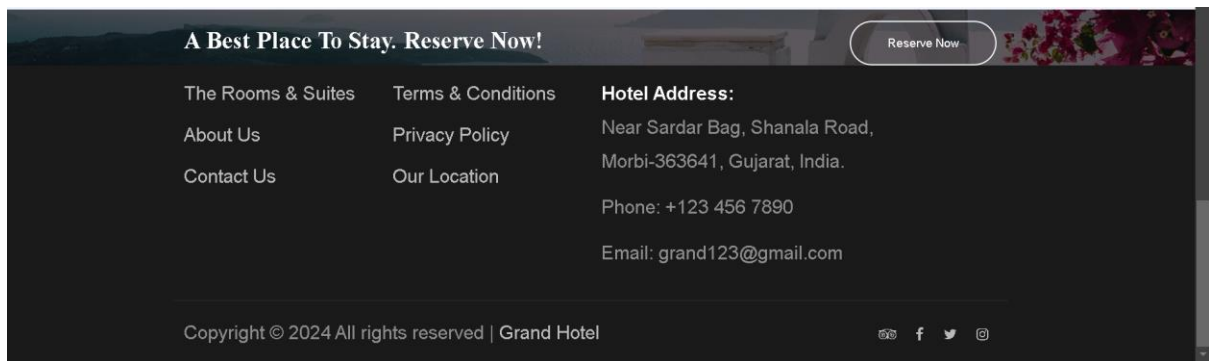
- **Hotel History and Mission:** A narrative section describing the hotel's background, core values, mission, and what sets it apart. This helps build a connection with potential guests by sharing the story behind the brand.
- **Services Overview:** Highlight key services offered, such as dining, spa, event hosting, and other unique amenities. This section serves as an overview of what guests can expect during their stay.



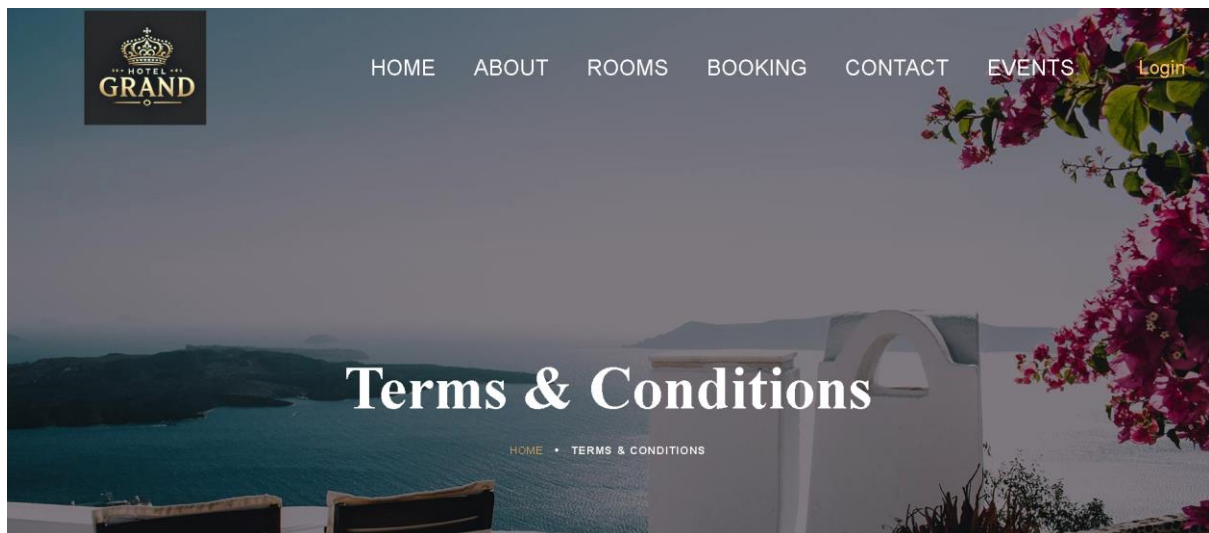
4.1.8. Header



4.1.9. Footer



4.1.10. Terms and Conditions



Terms & Conditions

Introduction

Welcome to Grand Hotel! These Terms and Conditions outline the rules and regulations for the use of our website.

Acceptance of Terms

By accessing this website, you agree to be bound by these Terms and Conditions and our Privacy Policy.

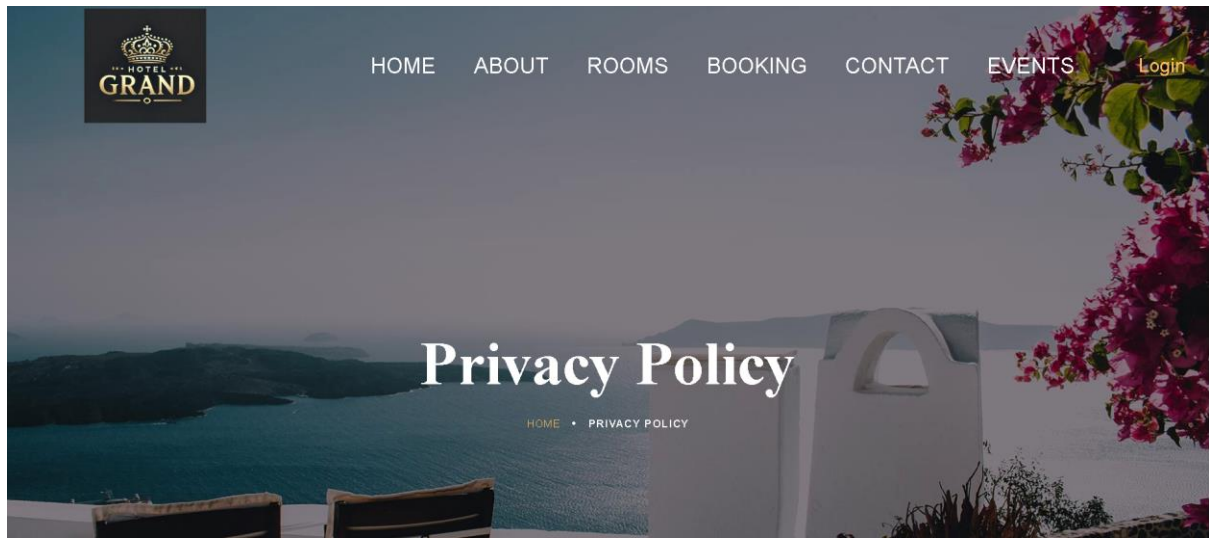
Changes to Terms

We may update these terms from time to time. Continued use of the site signifies acceptance of any changes.

Contact Us

If you have any questions about these Terms and Conditions, please contact us.

4.1.11. Privacy Policy



Privacy Policy

Introduction

Your privacy is important to us. This Privacy Policy explains how we collect, use, and protect your personal information.

Information We Collect

We collect information such as your name, email, and contact details when you use our services.

How We Use Your Information

We use your information to provide and improve our services, and to communicate with you.

Data Security

We implement security measures to protect your data from unauthorized access and misuse.

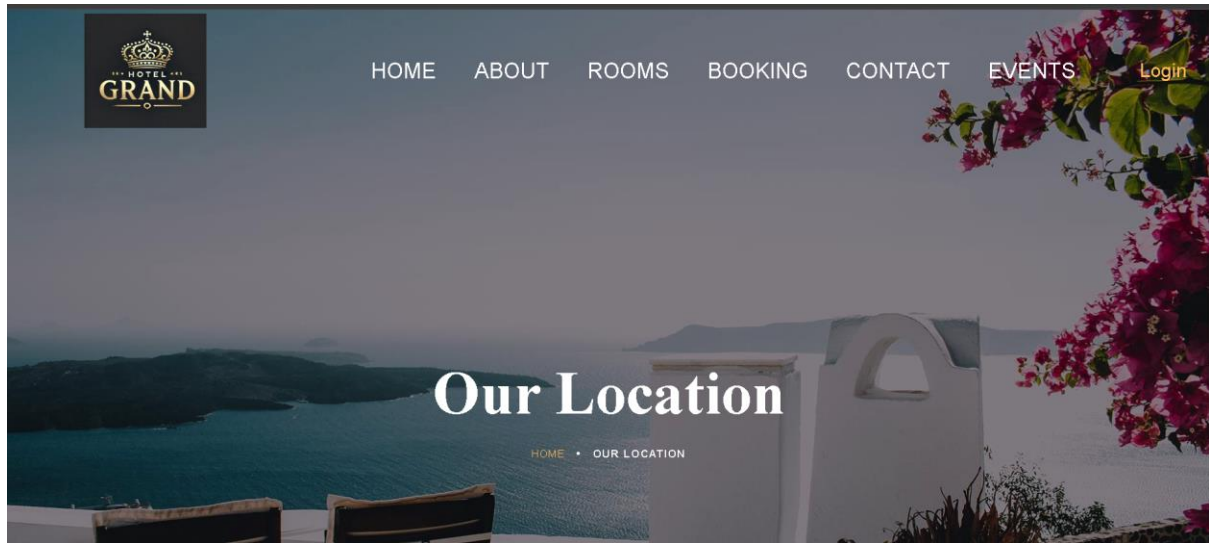
Changes to Privacy Policy

We may update this policy periodically. Please review it regularly.

Contact Us

If you have any questions about this Privacy Policy, please contact us.

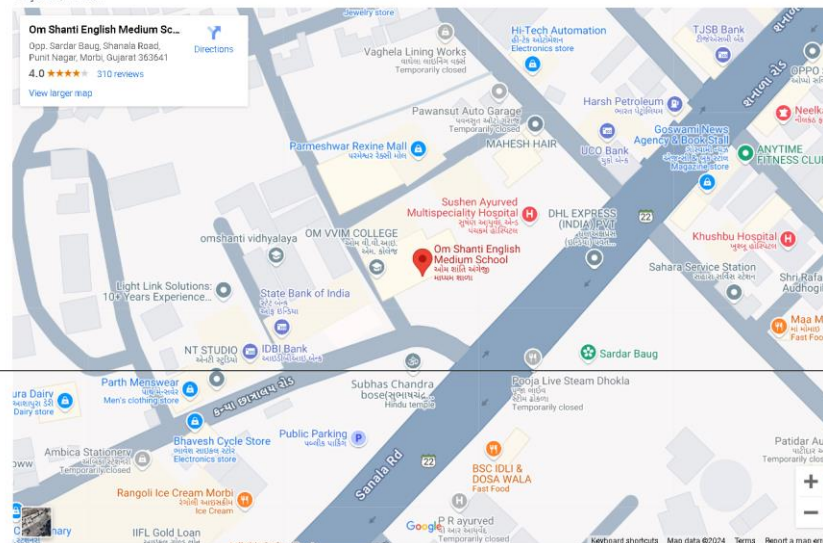
4.1.12. Our Location



Our Location

Hotel Address

Near Sardar Bag, Shanala Road,
Morbi-363641 ,
Gujarat, India.



How to Get Here

Grand Hotel is conveniently located in the heart of Morbi. Here's how you can reach us:

- **By Car:** From Bypass Road, turn onto Sanala Road. Our hotel is on the left.
- **By Train:** The nearest station is Morbi Central. Take a taxi or city bus to Grand Hotel.
- **By Air:** The nearest airport is Morbi International Airport. A taxi ride will take about 15 minutes.

Nearby Attractions

Explore the best of Morbi with these nearby attractions:

- **Mani Mandir** - A must-see for history enthusiasts.
- **Sardar Bag** - A beautiful park for relaxation and recreation.
- **Sky Mall** - Shop till you drop!

4.1.10. Admin Panel

- **Admin Dashboard:** Shows images and recent reservations.

Hotel Admin

Dashboard


Manage Rooms

Manage Reservations

Manage Users

Users Panel

Dashboard



Hotel Admin


Dashboard

Manage Rooms

Manage Reservations

Manage Users

Users Panel



Recent Reservations

Reservation ID	User ID	Room ID	Check-in Date	Check-out Date
8	11	FR_1	Thu, 19 Sep 2024 00:00:00 GMT	Fri, 20 Sep 2024 00:00:00 GMT
9	11	FR_2	Thu, 19 Sep 2024 00:00:00 GMT	Fri, 20 Sep 2024 00:00:00 GMT
5	6	BC_3	Mon, 30 Sep 2024 00:00:00 GMT	Mon, 30 Sep 2024 00:00:00 GMT
6	7	SR_NAC_1	Sun, 29 Sep 2024 00:00:00 GMT	Mon, 30 Sep 2024 00:00:00 GMT
7	7	SR_NAC_1	Sun, 29 Sep 2024 00:00:00 GMT	Sun, 29 Sep 2024 00:00:00 GMT
3	11	BC_4	Tue, 17 Sep 2024 00:00:00 GMT	Wed, 18 Sep 2024 00:00:00 GMT
1	7	FR_1	Sun, 15 Sep 2024 00:00:00 GMT	Mon, 16 Sep 2024 00:00:00 GMT
2	7	SR_NAC_1	Sun, 15 Sep 2024 00:00:00 GMT	Mon, 16 Sep 2024 00:00:00 GMT

- **Room Management:** Admins can add, update, or delete rooms, including modifying room rates and availability. This section also allows categorizing rooms by type, ensuring the data is up-to-date and reflects accurately on the user-facing pages.

Hotel Admin

Dashboard

Manage Rooms

Manage Reservations

Manage Users

Users Panel

Manage Rooms

Business Class

Add Room

View Rooms

Deluxe Suite

Add Room

View Rooms

Family Room

Add Room

View Rooms

Junior Suite

Add Room

View Rooms

Presidential Suite

Add Room

View Rooms

Super Deluxe Suite

Add Room

View Rooms

Single Room (AC)

Add Room

View Rooms

Business Class

Add Room

View Rooms

Room ID: BC_1

Rate: ₹5000.00

Edit

Delete

Room ID: BC_2

Rate: ₹5000.00

Edit

Delete

Room ID: BC_3

Rate: ₹5000.00

Edit

Delete

Room ID: BC_4

Rate: ₹5000.00

Edit

Delete

Add Room

Room ID:

Room Type:

Business Class

Room Rate:

Save Room

Cancel

Edit Room

Room ID:

BC_1

Room Type:

Business Class

Room Rate:

5000.00

Update Room

Cancel

- Hotel Admin

Dashboard

Manage Rooms

Manage Reservations

Manage Users

Users Penal

Manage Reservations

Reservation ID	User ID	Room ID	Check-in Date	Check-out Date	Reservation Date	Room Type	Actions
1	7	FR_1	Sun, 15 Sep 2024 00:00:00 GMT	Mon, 16 Sep 2024 00:00:00 GMT	Sun, 15 Sep 2024 00:00:00 GMT	Family Room	Delete
2	7	SR_NAC_1	Sun, 15 Sep 2024 00:00:00 GMT	Mon, 16 Sep 2024 00:00:00 GMT	Sun, 15 Sep 2024 00:00:00 GMT	Single Room (No AC)	Delete
3	11	BC_4	Tue, 17 Sep 2024 00:00:00 GMT	Wed, 18 Sep 2024 00:00:00 GMT	Tue, 17 Sep 2024 00:00:00 GMT	Business Class	Delete
5	6	BC_3	Mon, 30 Sep 2024 00:00:00 GMT	Mon, 30 Sep 2024 00:00:00 GMT	Wed, 18 Sep 2024 00:00:00 GMT	Business Class	Delete
6	7	SR_NAC_1	Sun, 29 Sep 2024 00:00:00 GMT	Mon, 30 Sep 2024 00:00:00 GMT	Wed, 18 Sep 2024 00:00:00 GMT	Single Room (No AC)	Delete
7	7	SR_NAC_1	Sun, 29 Sep 2024 00:00:00 GMT	Sun, 29 Sep 2024 00:00:00 GMT	Wed, 18 Sep 2024 00:00:00 GMT	Single Room (No AC)	Delete
8	11	FR_1	Thu, 19 Sep 2024 00:00:00 GMT	Fri, 20 Sep 2024 00:00:00 GMT	Thu, 19 Sep 2024 00:00:00 GMT	Family Room	Delete
9	11	FR_2	Thu, 19 Sep 2024 00:00:00 GMT	Fri, 20 Sep 2024 00:00:00 GMT	Thu, 19 Sep 2024 00:00:00 GMT	Family Room	Delete

- **User Management:** Manage all users (add, remove, block/unblock, change roles). This section provides controls for admins to oversee and maintain the system efficiently.

Manage Users								
User ID	First Name	Last Name	Email	Phone Number	Address	Terms Accepted	Is Admin	Actions
6	Bhavesh	Parmar	bhavesh0720@gmail.com	1234567890	gdvcdbjbhvj	Yes	Yes	<div>Revoke Admin</div> <div>Delete</div> <div>Block</div>
7	dhaval	Parmar	dhaval123@gmail.com	4632883722	rfgbjvcertyhjn6774edf	Yes	No	<div>Make Admin</div> <div>Delete</div> <div>Block</div>
9	ravi	bhatasana	ravi123@gmail.com	01101001001	01276372-1e36r278eev	Yes	No	<div>Make Admin</div> <div>Delete</div> <div>Block</div>
11	admin	admin	admin123@gmail.com	73652132	tankar,morbi, 363650	Yes	Yes	<div>Revoke Admin</div> <div>Delete</div> <div>Block</div>
14	rajaji		rajaji123@gmail.com	81726456	2453678e90	Yes	No	<div>Make Admin</div> <div>Unblock</div>

ADD NEW USER

First Name

Last Name

Email

Password

Phone Number

Address

Terms Accepted:

Admin User:

Add User

4.2 Testing

4.2.1. Unit Testing

- **Purpose:** Unit testing focuses on testing individual components or units of the code, such as functions, methods, or classes, to ensure they work correctly in isolation. The goal is to catch bugs early in the development process, making the application more robust and reliable.
- **Approach:** Typically involves writing test cases using testing frameworks (e.g., pytest for Python, JUnit for JavaScript). Each test targets a specific function or module, checking its outputs against expected results. Unit tests help developers quickly identify and fix issues at the code level, promoting good coding practices and modular design.
- **Test Scenarios:**
 - **Login Module:**
 - Test Case: Valid Email and Password
 - ✓ Expected Result: Successful login and redirection to the user dashboard.
 - ✓ Outcome: Pass
 - Test Case: Invalid Email or password
 - ✓ Expected Result: Error message indicating invalid Email or password.
 - ✓ Outcome: Pass
 - **Sign Up Module:**
 - Test Case: Valid Details for All Fields
 - ✓ Expected Result: Successful registration and redirection to the login page.
 - ✓ Outcome: Pass
 - Test Case: Missing Required Fields (e.g., First Name or Password)
 - ✓ Expected Result: Error message indicating missing required fields.
 - ✓ Outcome: Pass
 - Test Case: Weak Password (e.g., no uppercase letters or special characters)
 - ✓ Expected Result: Error message indicating the password must meet strength requirements.
 - ✓ Outcome: Pass
 - Test Case: Mismatched Passwords (Password and Confirm Password)
 - ✓ Expected Result: Error message indicating passwords do not match.

- ✓ Outcome: Pass
- Test Case: Duplicate Email Registration
 - ✓ Expected Result: Error message indicating the email is already registered.
 - ✓ Outcome: Pass

▪ **Reservation Module:**

- Test Case: Valid Room Selection and Date Range
 - ✓ Expected Result: Successful reservation and confirmation message displayed.
 - ✓ Outcome: Pass
- Test Case: Invalid Date Range (e.g., Check-out date is before Check-in date)
 - ✓ Expected Result: Error message indicating invalid date selection.
 - ✓ Outcome: Pass
- Test Case: Room Not Available
 - ✓ Expected Result: Error message indicating the selected room is not available.
 - ✓ Outcome: Pass
- Test Case: Missing Required Fields (e.g., Room Type(Category), check-in date or check-out date)
 - ✓ Expected Result: Error message indicating missing required fields.
 - ✓ Outcome: Pass
- Test Case: Overlapping Reservation for the Same Room
 - ✓ Expected Result: Error message indicating room is already booked for the selected dates.
 - ✓ Outcome: Pass

▪ **Admin Module:**

- Test Case: Add New Room with Valid Details
 - ✓ Expected Result: Room is successfully added and appears in the room list.
 - ✓ Outcome: Pass
- Test Case: Update Room Details (e.g., Rate Change)
 - ✓ Expected Result: Room details are successfully updated in the system.
 - ✓ Outcome: Pass
- Test Case: Delete Room
 - ✓ Expected Result: Room is successfully deleted and removed from the room list.
 - ✓ Outcome: Pass
- Test Case: Invalid Admin Credentials for Login
 - ✓ Expected Result: Error message indicating invalid credentials.
 - ✓ Outcome: Pass
- Test Case: Attempt to Access Admin Panel Without Admin Rights

- ✓ Expected Result: Access denied message and redirection to the login page.
- ✓ Outcome: Pass
- Test Case: Block User Account
 - ✓ Expected Result: User account is blocked, and blocked status is reflected in the system.
 - ✓ Outcome: Pass
- Test Case: Toggle User Role (Admin/User)
 - ✓ Expected Result: User role is successfully changed, and changes are visible in the user list.
 - ✓ Outcome: Pass

4.2.2. Integration Testing

- **Purpose:** Integration testing examines how different modules or components of the application interact with each other. While unit testing checks individual components, integration testing verifies that these components work together as intended, ensuring data flows smoothly between parts of the system.
- **Approach:** Integration tests often involve scenarios where multiple functions, modules, or APIs are tested together. For example, testing how the reservation form interacts with the database to ensure room availability updates correctly. The goal is to detect any interface defects or mismatches between integrated components.
- **Test Scenarios:**
 - **Login and Sign Up Integration:**
 - Test Case: Successful Registration Followed by Login
 - ✓ Scenario: A new user signs up successfully and then attempts to log in with the newly registered credentials.
 - ✓ Expected Result: User is redirected to the dashboard after logging in successfully.
 - ✓ Outcome: Pass
 - Test Case: Login with Blocked or Disabled Account
 - ✓ Scenario: An admin blocks a user account, and the user then tries to log in.

- ✓ Expected Result: User receives an error message indicating the account is blocked or disabled.
- ✓ Outcome: Pass
- **Reservation and Database Interaction:**
 - Test Case: Room Availability Update After Reservation
 - ✓ Scenario: A user books a room, and the system checks room availability before and after the booking.
 - ✓ Expected Result: Room status changes to unavailable, and booking details are updated in the database.
 - ✓ Outcome: Pass
 - Test Case: Booking with Conflicting Dates
 - ✓ Scenario: A user tries to book a room for dates that overlap with an existing reservation.
 - ✓ Expected Result: System displays an error message, preventing double booking.
 - ✓ Outcome: Pass
- **Admin and User Management Integration:**
 - Test Case: Admin Adds New Room in Category and Updates User Interface
 - ✓ Scenario: An admin adds a new room to a category, and the changes reflect on the user-facing reservation form.
 - ✓ Expected Result: The new room is available for selection in the reservation form.
 - ✓ Outcome: Pass
 - Test Case: Admin Blocking User and Login Restriction
 - ✓ Scenario: An admin blocks a user, and the blocked user attempts to log in.
 - ✓ Expected Result: The user is denied access with a message indicating account restrictions.
 - ✓ Outcome: Pass
 - Test Case: Admin Panel Changes Reflect in User Dashboard
 - ✓ Scenario: An admin updates room rates or availability, and the changes should reflect on the user dashboard.
 - ✓ Expected Result: Updated information is displayed correctly in the user interface without any inconsistencies.
 - ✓ Outcome: Pass

4.2.3. User Acceptance Testing (UAT)

- **Purpose:** UAT is conducted to validate the system against user requirements and ensure the application meets the needs of its end users. It involves real users testing the application in a controlled environment to provide feedback on usability, functionality, and overall satisfaction.

- **Approach:** UAT typically follows predefined test scenarios that reflect real-world usage. Users interact with the system as they would in a live setting, testing key functions such as booking a room, managing reservations, or accessing the profile page. Feedback is gathered on performance, ease of use, and any encountered bugs, leading to necessary adjustments before final deployment.
- **Test Scenarios:**
 - **Room Booking and Reservation:**
 - Scenario: Booking a Room with Valid Details
 - ✓ Steps: The user navigates to the reservation form, selects a room, and provides valid check-in and check-out dates.
 - ✓ Expected Result: Room is successfully booked, and a confirmation message is displayed.
 - ✓ Outcome: Pass
 - Scenario: Booking a Room with Invalid Date Range
 - ✓ Steps: The user selects a check-out date that is earlier than the check-in date.
 - ✓ Expected Result: Error message indicating an invalid date range is shown, preventing the booking.
 - ✓ Outcome: Pass
 - Scenario: Booking an Already Reserved Room
 - ✓ Steps: The user attempts to book a room that is already reserved for the selected dates.
 - ✓ Expected Result: System prevents the booking and displays a message indicating the room is unavailable.
 - ✓ Outcome: Pass
 - **User Profile:**
 - Scenario: Accessing Reservation History
 - ✓ Steps: The user views their past reservations by accessing their profile section.
 - ✓ Expected Result: The reservation history displays all past bookings with relevant details such as dates, room type, and status.
 - ✓ Outcome: Pass
 - **Login and User Access:**
 - Scenario: Logging in with Valid Credentials
 - ✓ Steps: The user enters valid email and password details on the login page.
 - ✓ Expected Result: Successful login and redirection to the user dashboard.
 - ✓ Outcome: Pass

- Scenario: Attempting to Login with Blocked Account
 - ✓ Steps: The user attempts to log in after being blocked by the admin.
 - ✓ Expected Result: Login fails with a message indicating the account is blocked.
 - ✓ Outcome: Pass

4.3 Future Enhancement

4.3.1. Mobile Application

- **Purpose:** Creating a mobile application would provide users with a convenient and seamless experience, allowing them to access hotel services on the go. This would include functionalities like room booking, checking availability, managing reservations, and accessing user profiles directly from their smartphones.
- **Approach:** The mobile app would be designed with a user-friendly interface optimized for touch interaction, featuring a streamlined version of the web functionalities. Technologies like React Native or Flutter could be used to develop cross-platform apps for both Android and iOS. The app would sync with the existing backend, ensuring consistency of data across devices, and could include additional features like push notifications for reservation updates and special offers.

4.3.2. Advanced Analytics

- **Purpose:** Advanced analytics would help the hotel gain valuable insights into customer behavior, booking trends, and operational efficiency. By analyzing data, the hotel can identify peak booking periods, popular room types, and common customer preferences, allowing for data-driven decision-making.
- **Approach:** Implementing analytics would involve setting up data collection from various touchpoints (e.g., reservation forms, user profiles) and integrating it with analytics tools like Google Analytics, Power BI, or custom dashboards. Machine learning algorithms could be used to predict booking trends and personalize offers, enhancing the user experience and optimizing pricing strategies based on demand.

4.3.3. Integration

- **Purpose:** Integrating third-party services would enhance the functionality of the system by adding features like secure payment processing, SMS notifications, and real-time booking confirmations. This would provide users with a smoother and more trustworthy booking experience.
- **Approach:** Integration could involve APIs from payment gateways like Stripe, PayPal, or Razorpay for secure and convenient transactions. Additionally, integrating with calendar services could allow users to add reservations directly to their schedules, and linking with travel platforms like Google Maps could provide location-based services such as directions and local recommendations.

5. Tools and Technology

5.1 What is Python?

- **Definition:** Python is a high-level, interpreted programming language known for its clear syntax and readability, making it easy to learn and use. It emphasizes code readability with its use of significant indentation, allowing developers to write clear and logical code for both small and large-scale projects.
- **Features:** Python supports multiple programming paradigms, including procedural, object-oriented, and functional programming, making it flexible and versatile. It has an extensive standard library and numerous third-party packages that extend its capabilities.
- **Uses:** Python is widely used in various fields such as web development (Django, Flask), data analysis (Pandas, NumPy), artificial intelligence and machine learning (TensorFlow, PyTorch), automation, scientific computing, and more. Its popularity continues to grow due to its broad application range and supportive community.

5.2 What is MySQL?

- **Definition:** MySQL is an open-source relational database management system (RDBMS) that uses Structured Query Language (SQL) for accessing, managing, and manipulating data in databases. It is developed, distributed, and supported by Oracle Corporation.
- **Features:** Known for its high performance, reliability, and ease of use, MySQL offers features like robust security, high availability, and scalability, which make it suitable for both small applications and large-scale enterprise systems.
- **Uses:** MySQL is commonly used in web applications, data warehousing, and e-commerce platforms, powering popular websites like Facebook, Twitter, and YouTube. It provides a solid foundation for managing data in content management systems (e.g., WordPress) and other data-driven applications.

5.3 Bibliography

- **Python Documentation:** The official Python documentation provides comprehensive information about Python's syntax, libraries, and usage examples. Available at <https://docs.python.org/>.
- **MySQL Documentation:** This resource covers MySQL's features, commands, and best practices for managing databases effectively. Available at <https://dev.mysql.com/doc/>.
- **Web Development Resources:** Various online tutorials and official documentation provide guidance on web development using different technologies, supporting a wide range of web-based projects.