



Project Title	Hotel/Farm House Booking Management System
Technologies	MERN
Domain	Industry
Project Level	Medium

Table

Table of Contents

1. Problem Statement:	2
1.1. What is a Hotel/Farm House Booking System?	2
1.2. Project Objective	3
1.3. Functional and Non-Functional Requirements: -.....	3
1.3.1. Functional Requirements	3
1.3.2. Non-Functional Requirements	3
2. Project Evaluation metrics:	3
2.2. Database:	4
2.3. API Details or User Interface:	4
2.4. Deployment:	4
Deploy the application on your preferred service.....	4
2.5. Solutions Design:.....	4



2.6. System Architecture:	4
2.7. Optimization of solutions:	4
3. Submission requirements:	
.....	4
3.1. High-level Document:	4
3.2. Low-level document:	4
3.3. Architecture:	4
3.4. Wireframe:	5
3.5. Project code:	5
3.6. Detail project report:	5
3.7. Project demo video:	5
3.8. The project LinkedIn a post:	5

1. Problem Statement:

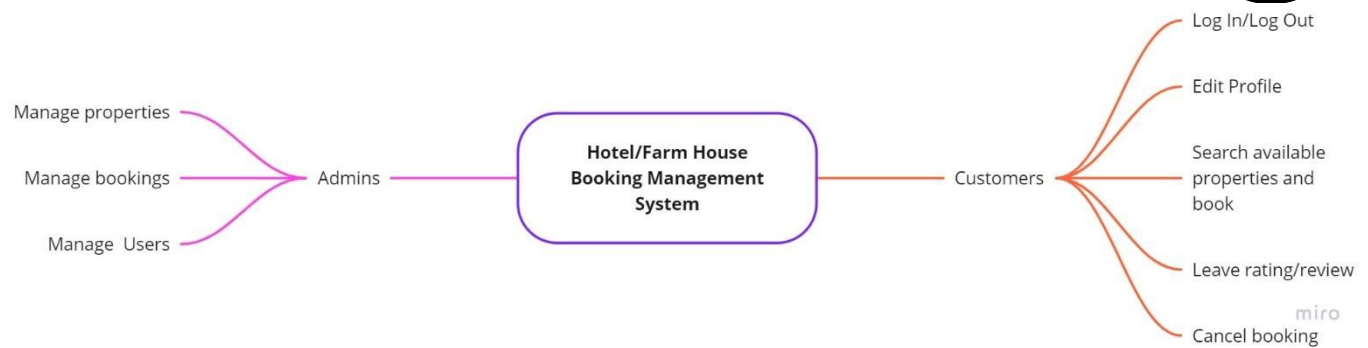
Design a web application “Hotel/Farm House Booking Management System” to allow users to book either a farmhouse or a hotel.

1.1. What is a Hotel/Farm House Booking System?

A web-based application where there are hotels and farmhouses listed and users can visit the site and check the available properties for booking based on their requirements and budget, they can directly book any property for the specified period. Once the booking is confirmed the customer can visit the property on the booking day and stay there for the booking period.

Now that you understand what Hotel/Farm House Booking System is, let’s discuss some of the functionality of the “**Hotel/Farm House Booking Management System**” that you will design.

1. To create a web-based system that will let the users sign up with basic information.
2. There has to be a search form that searches for the properties according to the user’s needs.
3. The search form should have some filters ex: City, and property type (Hotel, Farm House).
4. The customer should have an option to select the period to confirm how long they want to book the property.
5. After the customer has booked the property, they should be notified about their booking status.
6. Once the booking is confirmed customer can then go to the place to stay and once everything is done, the customer can leave a rating/review about the property.



Low-Level system flow

1.2. Project Objective

- The main objective is to manage the details of rooms, bookings, and customers and keep track of available and booked rooms in a clean UI.
- To allow users to book hotels/farmhouses directly online with no intermediaries necessary.

1.3. Functional and Non-Functional Requirements: -

1.3.1. Functional Requirements

- Admin must be able to see the number of booked hotels/Farm House in the admin dashboard.
- Admin should be able to Add/Edit/Remove any Hotel/Farm House from the site.
- Admin should be able to approve or cancel any booking.
- Admin should be able to manage users.
- User should be able to see all the hotels/farmhouses available for booking.
- Users should be able to search for hotels or farmhouses according to their preferences.
- Users can see the details of any hotel or farmhouse and can only book if they are logged in.
- Users should be able to update their personal information.

1.3.2. Non-Functional Requirements

1. **Security:** The system should have a certain level of security such that not anybody can have access to sensitive information and the passwords should be properly encrypted.
2. **Robustness:** If the user's system crashes, a backup of the user data must be stored on remote database servers to enable recovery.
3. **Performance:** The application must be lightweight and the UI should be fast and responsive.
4. **Authentication Requirements:** There should be an authentication system in place which checks the credentials each time a user logs in.



2. Project Evaluation metrics:

2.1. Code:

- You are supposed to write code in a modular fashion
- Safe: It can be used without causing harm.
- Testable: It can be tested at the code level.
- Maintainable: It can be maintained, even as your codebase grows.
- Portable: It works the same in every environment (operating system).
- You have to maintain your code on GitHub.
- You have to keep your GitHub repo public so that anyone can check your code.
- Proper readme file you have to maintain for any project development.
- You should include the basic workflow and execution of the entire project in the readme file on GitHub.
- Follow the coding standards.

2.2. Database:

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas.

2.3. API Details or User Interface:

You have to expose your complete solution as an API or try to create a user interface for your model testing. Anything will be fine for us.

2.4. Deployment:

Deploy the application on your preferred service.

2.5. Solutions Design:

You have to submit complete solution design strategies in High-level Document (HLD), Lowlevel Document (LLD), and Wireframe documents.

2.6. System Architecture:

You must submit a system architecture design in your wireframe and architecture documents.

2.7. Optimization of solutions:

Try to optimize your solution on the code level, and architecture level, and mention all of these things in your final submission.

Mention your test cases for your project.

3. Submission requirements:



3.1. High-level Document:

You have to create a high-level document design for your project. You can reference the HLD form below the link.

Sample link: [HLD Document Link](#)

3.2. Low-level document:

You have to create a Low-level document design for your project; you can refer to the LLD from the link below.

Sample link: [LLD Document Link](#)

3.3. Architecture:

You have to create an Architecture document design for your project; you can refer to the Architecture from the link below.

Sample link: [Architecture sample link](#)

3.4. Wireframe:

You have to create a Wireframe document design for your project; refer to the Wireframe from the link below.

Demo link: [Wireframe Document Link](#)

3.1. Project code:

You have to submit your code to the GitHub repo in your dashboard when the final submission of your project.

Demo link: [Project code sample link](#)

3.2. Detail project report:

You have to create a detailed project report and submit that document as per the given sample.

Demo link: [DPR sample link](#)

3.3. Project demo video:

You have to record a project demo video for at least 5 Minutes and submit that link.

The project LinkedIn a post:

You have to post your project details on LinkedIn and submit that post link in your dashboard in your respective field.