

CHAPTER 1

1.1 Introduction

The purpose of this project “Hotel Management System” involves maintaining various operations like admin can login to this system using correct admin username and password and admin can add the room or manage the room and also view the room booking list and cancellation of the room, cash billing and room service. user can check the room availability and also book the room through online.

We are going to develop this project using this modules like Admin module admin login, manage rooms, view room booking, In user module view room, check room availability and book room etc.

To implement this project we will use full stack languages like HTML, CSS, JavaScript, Bootstrap, PHP, MySQL database etc. Some changes could be done here to make it more reliable, more automatic and providing more features.

1.2 Capstone project Scope Document

Capstone project Title: “Hotel Management System”

Problem Statement:

As per the my market survey there is no such type of system are available. In some hotel online room booking facilities are not available. And in some hotel online room booking facilities are available. So for the user we are going to develop this project by this user can book the room by call and visit the room. By this user can save the time.

Objectives:

User can book the room from any location.

Reduce the time.

Reduce manual work.

Can connect with the user easily.

Increase your online visibility.

Capstone project description:

The purpose of this project “Hotel Management System” involves maintaining various operations like admin can login to this system using correct admin username and password and admin can add the room or manage the room and also view the room booking list and cancellation of the room, cash billing and room service. user can check the room availability and also book the room through online. We are going to develop this project using this modules like Admin module admin login, manage rooms, view room booking, In user module view room, check room availability and book room etc. To implement this project we will use full stack languages like HTML, CSS, JavaScript, Bootstrap, PHP, MySQL database etc. will be used to construct this project.

Capstone project Deliverables:

Admin Login
Manage Rooms
View Room Booking
View Rooms
Check Room Available
Book Room

Key milestones:**Admin Modules**

Admin Login
Manage Rooms
View Room Booking

User Modules

View Rooms
Check Room Available
Book Room

Constraints:

Time line for project is limited.
Online payment process is not possible.
Cost is limited.
Required limited employees for project.
Easy to use.

Estimated Capstone project Duration: 12 weeks (72 days)

Estimated Capstone project cost: 45,500/-

Table 1.2.1

Estimation	Cost (Rupees)
Labour Cost	25,000
Cost of Material	8,500
Net Profit	12,000
Total	45,500

CHAPTER 2

2.1 Capstone project planning

2.1.1 Work Breakdown Structure

In this project there are two main modules: - Admin and User.

Admin Modules

Admin Login :- Using user name and password admin can login.

Manage Rooms :- Admin can add the room type, room photo room information (add or delete).

View Room Booking :- In this module admin can view the list of booking.

User Modules

View Rooms :- In this module user can view the room.

Check Room Available :- In this module user can check the room availability.

Book Room :- In this module user can book the room

Activities and Tasks :-

Admin Login

Tasks

Admin must login with correct details.

Entered details must validate properly.

Entered detail must store in database

Manage Rooms

Tasks

Admin must add or delete room type, room photo, room information.

Added room type, room photo, room information must store in database.

View Room Booking

Tasks

Admin must able to view the booked room.

Entired details in view room booking page is stored in database.

View Rooms

Tasks

User must able to view the rooms.

Rooms and their information must displayed correctly.

Check Room Available

Tasks

User must able to check the available rooms.

Available rooms are must displayed properly.

Book Room

Tasks

User must able to book the room.

Booked Room details are must store in database.

2.1.2 Timeline Development-Schedule

Activities & Task

Admin Login (10 Days/70 hours)

Tasks

Admin must login with correct details.

Entered details must validate properly.

Entered detail must store in database

Collect requirement gathering and requirement analysis for above tasks

To design admin login form using HTML, CSS, Bootstrap.

Validation of admin login form using JavaScript.

Perform database connection using PHP and MySQL.

Testing admin login form to check validation and database.

This whole task is done by Shruti Gurav.

Manage Rooms (10 Days/70 hours)

Tasks

Admin must add or delete room type, room photo, room information.

Added room type, room photo, room information must store in database.

Collect requirement gathering and requirement analysis for above task

To design add room form using HTML, CSS, Bootstrap.

Validation of add room form using JavaScript.

Perform database connection using PHP and MySQL.

Testing add room page to check validation and database.

This whole task is done by Aishwarya Patil.

View Room Booking (10 Days/70 hours)

Tasks

Admin must able to view the booked room.

Entired details in view room booking page is stored in database.

Collect requirement gathering and requirement analysis for above tasks

To design view room booking page using HTML, CSS, Bootstrap.

Validation of view room booking page using JavaScript.

View the room booking page database connection using PHP and MySQL.

Testing room booking form to check validation and database.

This whole task is done by Shruti Gurav.

View Room (10 Days/70 hours)

Tasks

User must able to view the rooms.

Rooms and their information must displayed correctly.

Collect requirement gathering and requirement analysis for above tasks

To design view room page using HTML, CSS, Bootstrap.

Validation of view room page using JavaScript.

View the rooms database connection using PHP and Mysql.

Testing view room page to check validation and database.

This whole task is done by Pranay Hiremath.

Check Room Available (10 Days/70 hours)

Tasks

User must be able to check the available rooms.

Available rooms are must displayed properly.

Collect requirement gathering and requirement analysis for above tasks

To design Check room available page using HTML, CSS, Bootstrap.

Validation of Check room available page using JavaScript.

Check the available rooms database connection using PHP and MySQL.

Testing room availability page to check validation and database.

This whole task is done by Vaidehi Deval.

Book Room (10 Days/70 hours)

Tasks

User must be able to book the room.

Booked Room details are must store in database.

Collect requirement gathering and requirement analysis for above tasks

To design room booking form we are using HTML, CSS, Bootstrap.

Validation of room booking form using JavaScript.

Book the room database connection using PHP and MySQL.

Testing room booking form to check validation and database.

This whole task is done by Aishwarya Patil

2.1.3 Cost Breakdown Structure

Analyse your Work Breakdown Structure

Admin Login

Manage Room

View Room Booking

View Room

Check Room Available

Book Room

Estimate the cost of materials

The cloud hosting server cost is of 5000.

PHP Designer license key of rupees 3450

AMPP Severe is of free Source

Overhead costs

If requirement increase then cost increase.

Build contingency into your CBS

As per the estimated cost we need deliver the project

Final-check

As per the client budget we have to define project cost.

Table 2.1.3.1

Estimation	Cost (Rupees)
Labour Cost	25,000
Cost of Material	8,500
Net Profit	12,000
Total	45,500

Estimate the labour cost of work

Table 2.1.3.2

Modules	Tasks	Time			Total Cost (hour* Amount per hour)
		Hours Per Task	Hours	Amount Per Hour (Rupees)	
Admin Login	Admin must login with correct details.	24	70	60	70*60=4,200
	Entered details must validate properly.	24			
	Entered detail must store in database	24			
Manage Rooms	Admin must add or delete room type, room photo, room information.	35	70	60	70*60=4,200
	Added room type, room photo, room information must store in database.	35			
View Room Booking	Admin must able to view the booked room.	35	70	60	70*60=4,200
	Entired details in view room booking page is stored in database.	35			
View Rooms	User must able to view the rooms.	35	70	60	70*60=4,200
	Rooms and their information must displayed correctly.	35			
Check Room Available	User must able to check the available rooms.	35	70	60	70*60=4,200
	Available rooms are must displayed properly.	35			
Book Room	User must able to book the room.	35	70	60	70*60=4,200

2.1.4 Risk assessment

Home page must display Admin login, Rooms, Reservation, Contact US, About Us and Home all of these must be displayed and work properly.

Admin login form must work without any error.

Admin must be able to Manage the all rooms add the room type, room photo, room information.

Admin must be able to see all the reserved room list.

Admin must able to see the user information.

User must be able to view all the rooms with details and picture.

User must be able to check the room availabilities.

User must be able to book for the room in which they are interested.

Room Booked details must store properly.

2.2 Requirements Specification

2.2.1 Functional requirements:

Admin Aspects

Admin login into the system

Managing the rooms

Add the rooms

View the room booking list

Admin can logout

User Aspects

View room

Check room available

Book room

2.2.2 Non-functional requirements:

Usability

Usability is a quality attribute used to access how easy the interface is to use. Usability is ease of use. It tells how user friendly the interface is. It includes memorability, learnability, and satisfaction. Our software interface has all the above quality. Any kind of user can easily understand the interface.

Reliability

Reliability is how much the system is consistent in different platforms. The ability of an apparatus, system to consistently perform its required function, on demand and without degradation or failure.

Integrity

Integrity means doing the right thing in a reliable way. Data integrity is a fundamental component of security. In its broadcast use, “Data Integrity” refers to the accuracy and consistency of data stored in a database, data mart or another construct. Data integrity is the overall completeness, accuracy and consistency of data

Performance

Performance is also a major non-functional requirement. Performance Requirements about resources required, response time, transaction rate or anything else having to do with performance

2.2.3 User inputs

Admin:

Username and password

Manage the room

Add the room

View room booking list

User:

View the room and it's information

Check the available rooms

Book the room

2.2.4 Technical Constraints

Requirements for developing: -

Average I3 processor.

Average 4GB RAM.

Average 512GB hard disk.

Hardware requirement for deployment: -

Average I3 processor.

Average 2GB RAM.

Average 256GB storage.

Software requirement for development: -

OS (Window's, MAC, Linux).

XAMP Server.

VS Code.

Chrome.

Software requirement for deployment: -

OS.

Chrome.

Language used or technology: -

Content language (client-side language) (HTML, CSS, JS)

Backend (server-side language) (PHP).

Database (MYSQL).

2.3 Design Specification

2.3.1 Chosen System Design

System architecture diagrams provide a visual illustration of a system's various components and show how they communicate and interact with each other. These diagrams document a system's structure and architecture.

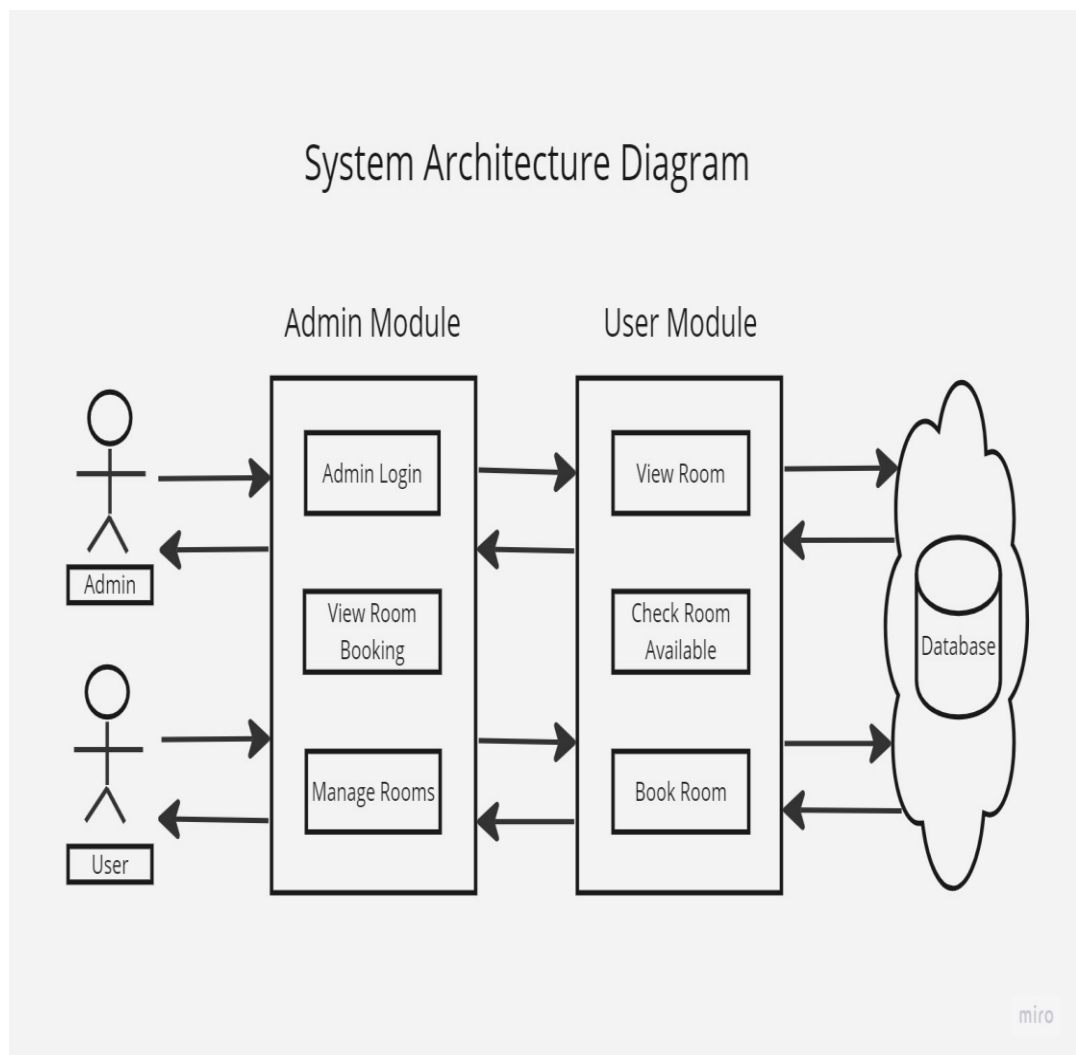


Figure 2.3.1.1

2.3.2 Discussion of Alternative design

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination.

DFD Level-0

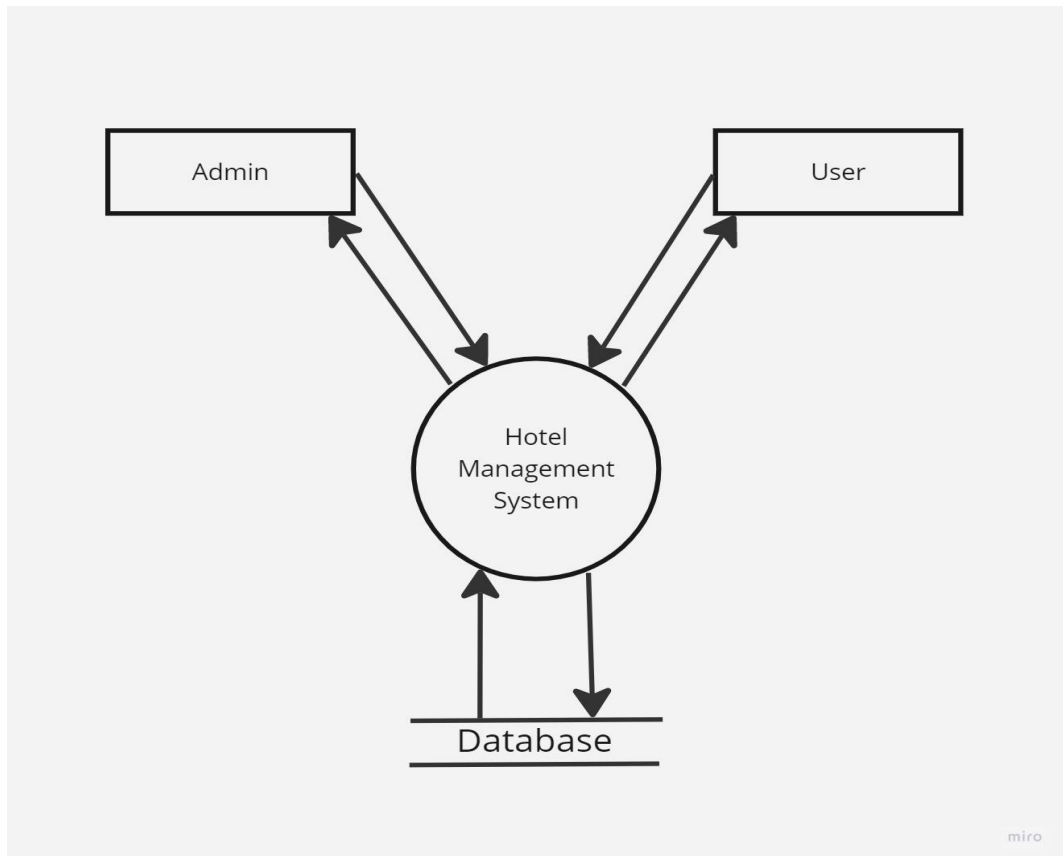


Figure 2.3.2.1

DFD Level-1 Admin

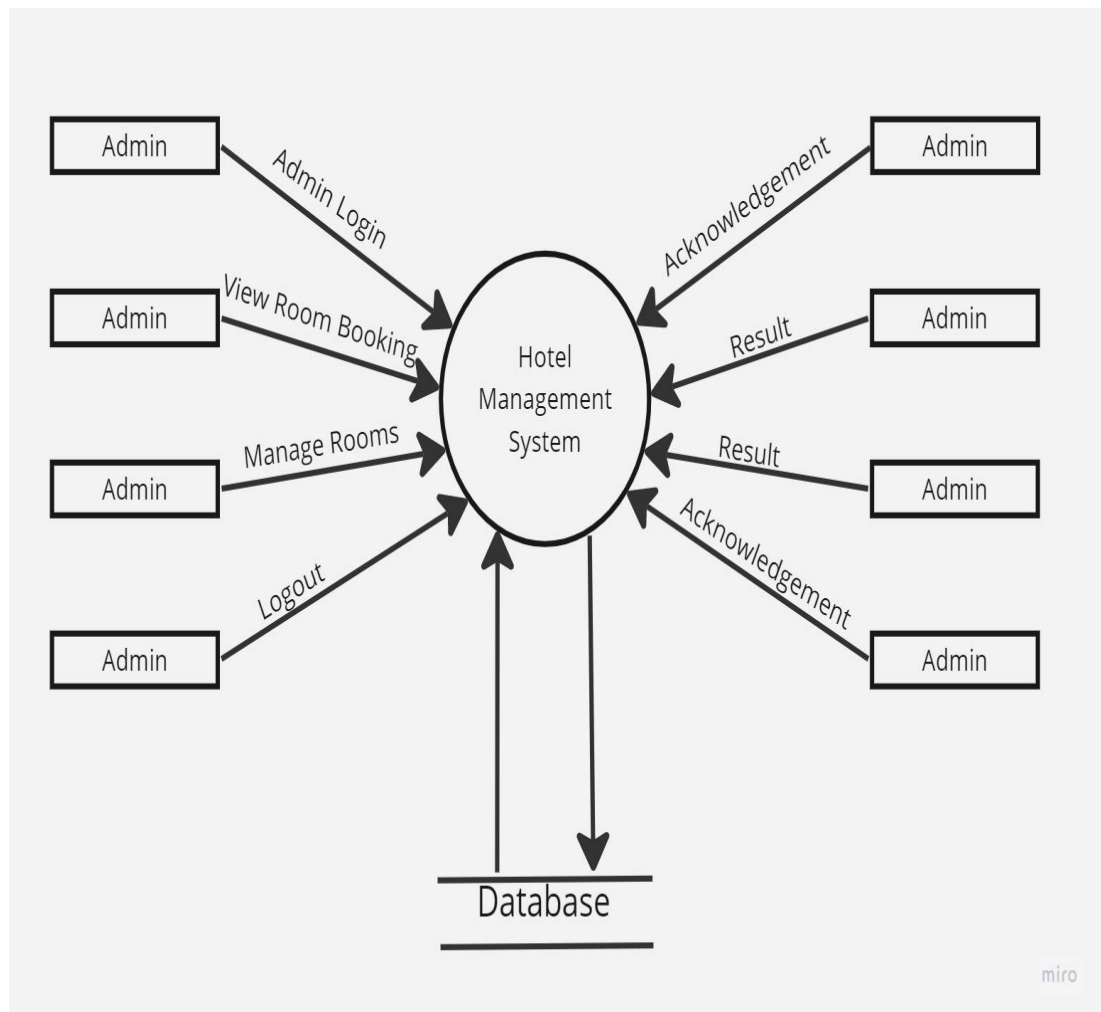


Figure 2.3.2.2

DFD Level-1 User

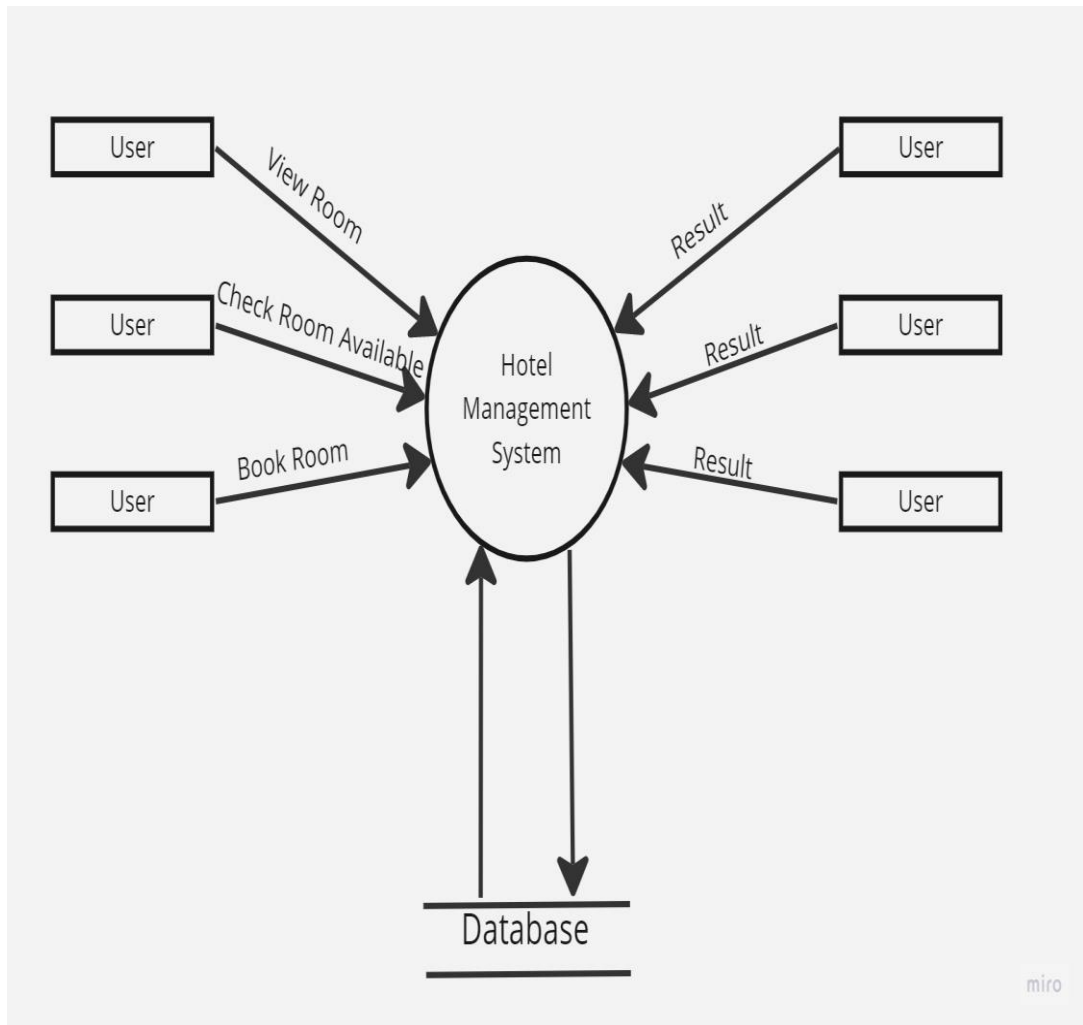


Figure 2.3.2.3

2.3.3 Detailed Description of Components/subsystems

Component diagrams are essentially class diagrams that focus on a system's components that often used to model the static implementation view of a system.

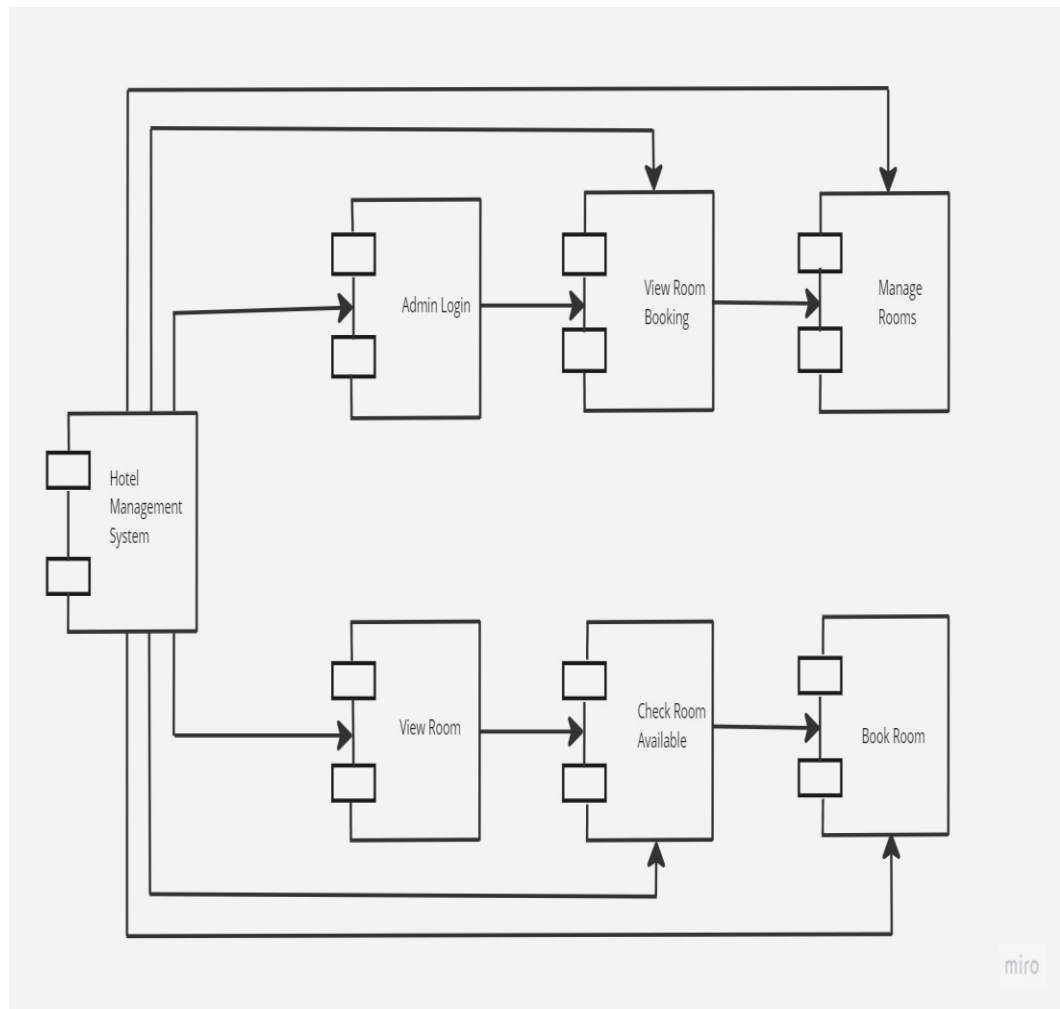


Figure 2.3.3.1

Admin login: -

Here admin can login to the form using admin username and password. After logging in he can add the room, add room information, room type and room photos. After Adding the room admin can view the room booking list.

Manage Room:-

Admin can also manage the rooms he can add the room, delete the room, add room information, add the room type and add the room photo. Admin can also check the booking of room.

View Room Booking:-

After admin logged in to the form admin can easily see the room booking list. Admin can see the user details, there check in and check out date and user needs.

View Room:-

User can view the room view room information, view room type, view room photo for booking of the room.

Check Room Available:-

User can check the availabilities of room and see the information, room type, room photo for there booking purpose.

Book Room:-

After seeing the room, there facilities, room type, there information user can successfully book the room without any issue.

2.3.4 Component 1-n

Admin Login

Manage Room

View Room Booking

View Room

Check Room Available

Book Room

CHAPTER 3

3.1 Approach and Methodology

3.1.1 Discuss the technology

Web technology

A place connected to the internet, where a company, organization, etc. Puts information that can be found on the World Wide Web.

Types of web technology: -

- Browsers
- HTML and CSS
- Programming Languages
- Frameworks
- Web Servers
- Databases
- Protocols
- Lastly, data Formats

Advantages

- We can access from any location.
- No data loss.
- Data can be recovered.
- Be available every time.
- We can save time.

Cloud based technology

To live the project, we need to purchase cloud sever.

Cloud based technology is the use of software and services via the internet. These applications commonly include data storage, networking, servers and databases. User can access their cloud hosted tools with any device that is connected to the internet.

Types of cloud-based technology

SAAS: - Software as a Service

PAAS: - Platform as a Service

IAAS: - Infrastructure as a Service

Advantages

Usability and accessibility

Security

Cost efficient

Convenient sharing of files

Automation

Open-Source web technology

For our project we not require to purchase any software's and libraries.

XAMPP

XAMPP is a cross-platform and open-source tool, which makes it an ideal choice of web developers. It is the acronym of X-cross platform, Apache, MySQL, PHP, and Perl.

PHP

PHP is an open-source scripting language used for creating dynamic and interactive web pages and various digital platforms.

PhpMyAdmin

PhpMyAdmin is an open source and free administration tool for MySQL

Advantages

Community-Driven Reliability

Community-Driven Security

Low Cost on an Ongoing Basis

Better, Community-Based Collaboration

3.1.2 Methodologies

Agile methodology is a process for managing a project that involves constant collaboration and working in iterations. Agile project management works off the basis that a project can be continuously improved upon throughout its life cycle, with changes being made quickly and responsively. Agile is one of the most popular approaches to project Management due to it 's flexible, adaptability to change, and high level of customer input.

Phases of Agile model: -

- Requirement Gathering
- Design the requirements
- Construction or Iterations
- Testing and Quality assurance
- Deployment
- Feedback

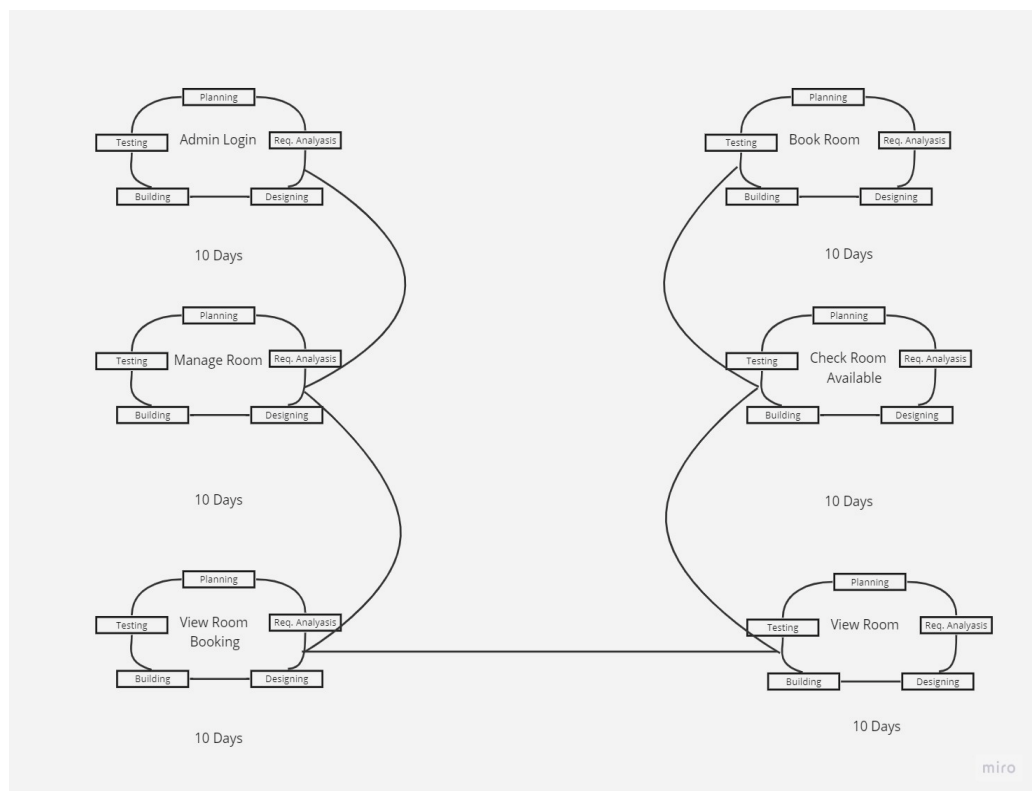


Figure 3.1.2.1

3.1.3 Use Cases

Use case diagram is the primary form of system/software requirements for a new software program underdeveloped. Use cases specify the expected behaviour (what), and not the exact method of making it happen.

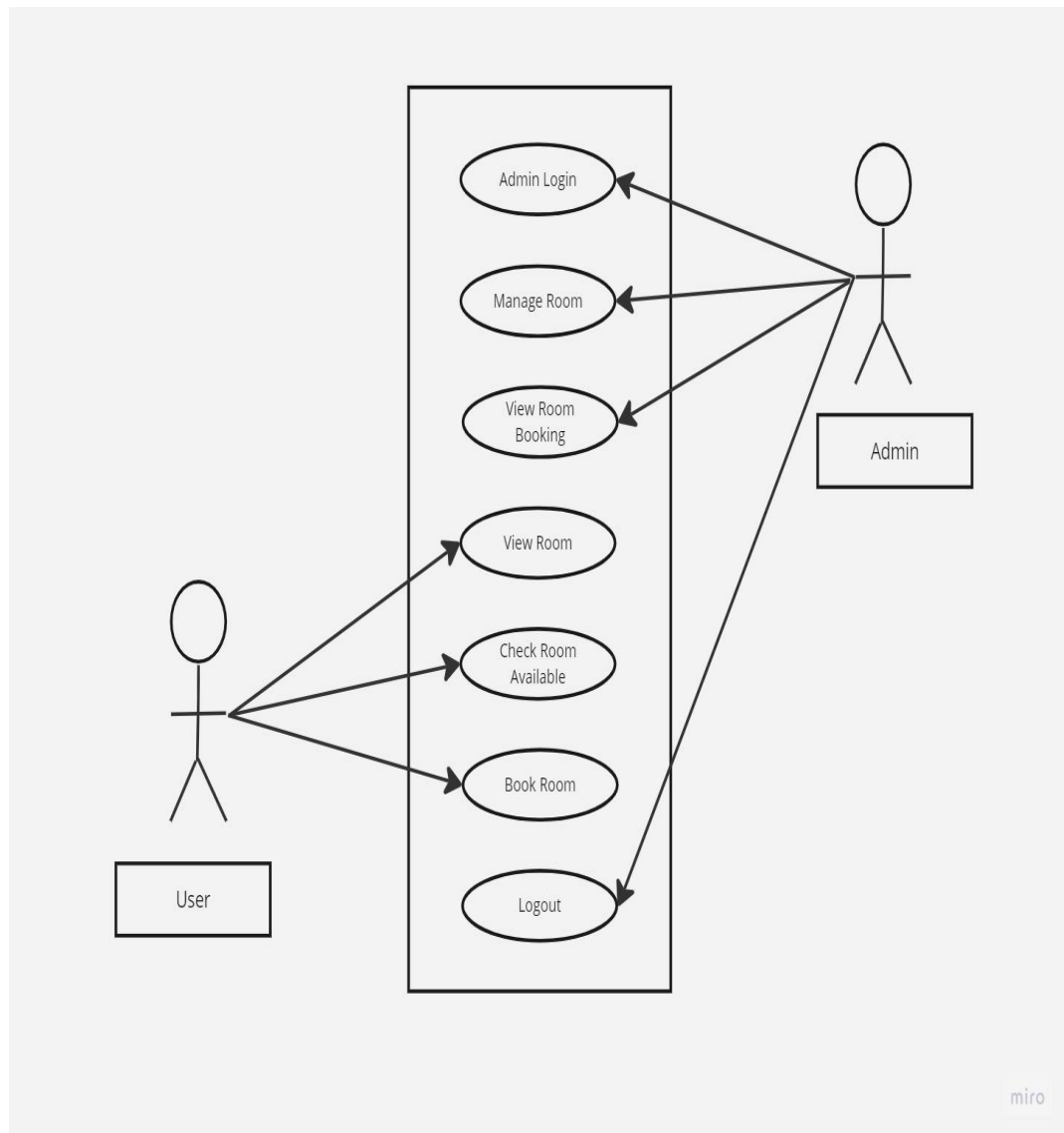


Figure 3.1.3.1

3.1.4 Programming

HTML

It stands for hypertext markup language. It is client-side scripting language. It is used to create static web pages. File extension of html is .html & html contains pre-defined tags.

Advantages

It is easy to learn.

Every browser supports HTML Language.

HTML is light weighted and fast to load.

HTML has many tags and attributes which can short your line of code.

CSS

CSS stands for Cascading Style Sheets. CSS is the language we use to style an html document. CSS describes how html elements should be displayed. It can control the layout of multiple web pages all at once. External stylesheets are stored in CSS files.

Advantages

Helps in making creative web pages by making them simple to use.

Improve the browsing speed.

It can be used on various devices.

Wider variety of design options.

Bootstrap

Bootstrap is a free and open-source framework for web development. It uses HTML, CSS, and JavaScript to create responsive and mobile-friendly websites and web applications. It provides a collection of syntax for template designs that make web development easier and faster.

Advantages

Open source

Easy to use

Save lots of time

Compatible with browser

JavaScript

JavaScript is the most popular lightweight, interpreted compiled programming language. It can be used for both Client side as well as server side developments. JavaScript also known as a scripting language for web pages. It is use for validation purpose.

Advantages

- Fast speed
- Easy to learn
- Versatility
- Popularity

PHP

PHP stands for Hypertext Pre-processor. PHP is a server-side scripting language designed specifically for web development. It is open-source which means it is free to download and use. It is very simple to learn and use. The extension for php is '.php'.

Advantages

- It's open-source and free from cost.
- It is platform-independent.
- It has less learning curve because it is simple and straightforward to use.
- It helps in managing code easily.

3.1.5 Analysis

As per the my market survey there is no such type of system are available. In some hotel online room booking facilities are not available. And in some hotel online room booking facilities are available. So for the user we are going to develop this project by this user can book the room by call and visit the room. By this user can save the time.

We are going to develop the hotel management system. The purpose of this project “Hotel Management System” involves maintaining various operations like admin can login to this system using correct admin username and password and admin can add the room or manage the room and also view the room booking list and cancellation of the room, cash billing and room service. user can check the room availability and also book the room through online. We are going to develop this project using this modules like Admin module admin login, manage rooms, view room booking, In user module view room, check room availability and book room etc. To implement this project we will use full stack languages like HTML, CSS, JavaScript, Bootstrap, PHP, MySQL database etc. Some changes could be done here to make it more reliable, more automatic and providing more features.

3.1.6 Process Design

List of tables

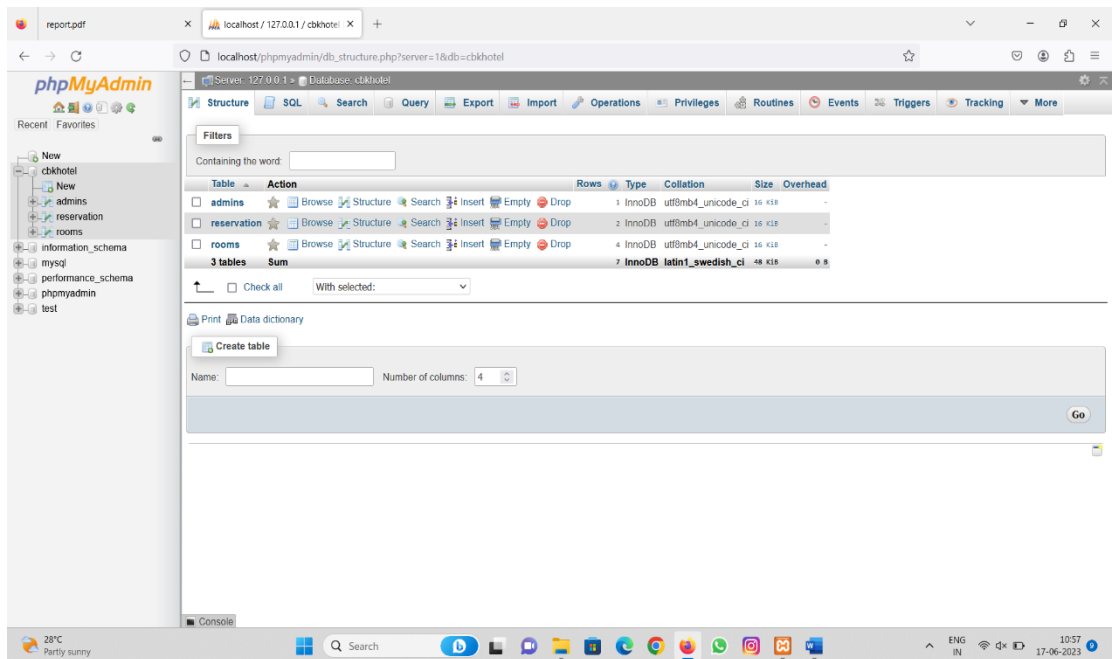


Figure 3.1.6.1

Admin Table

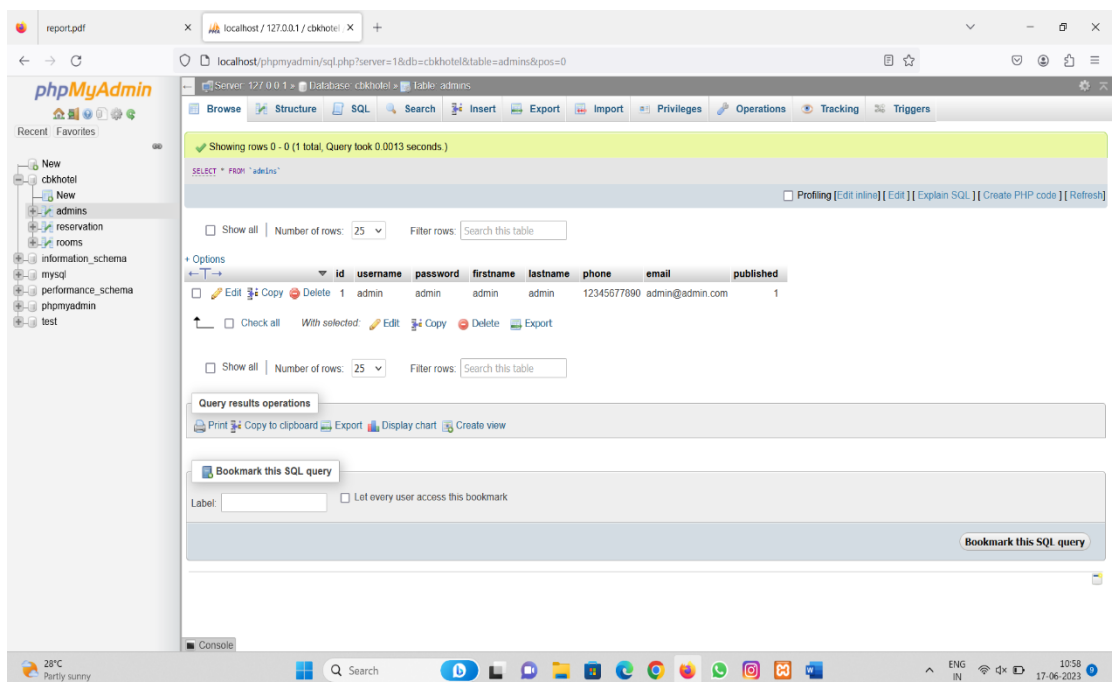
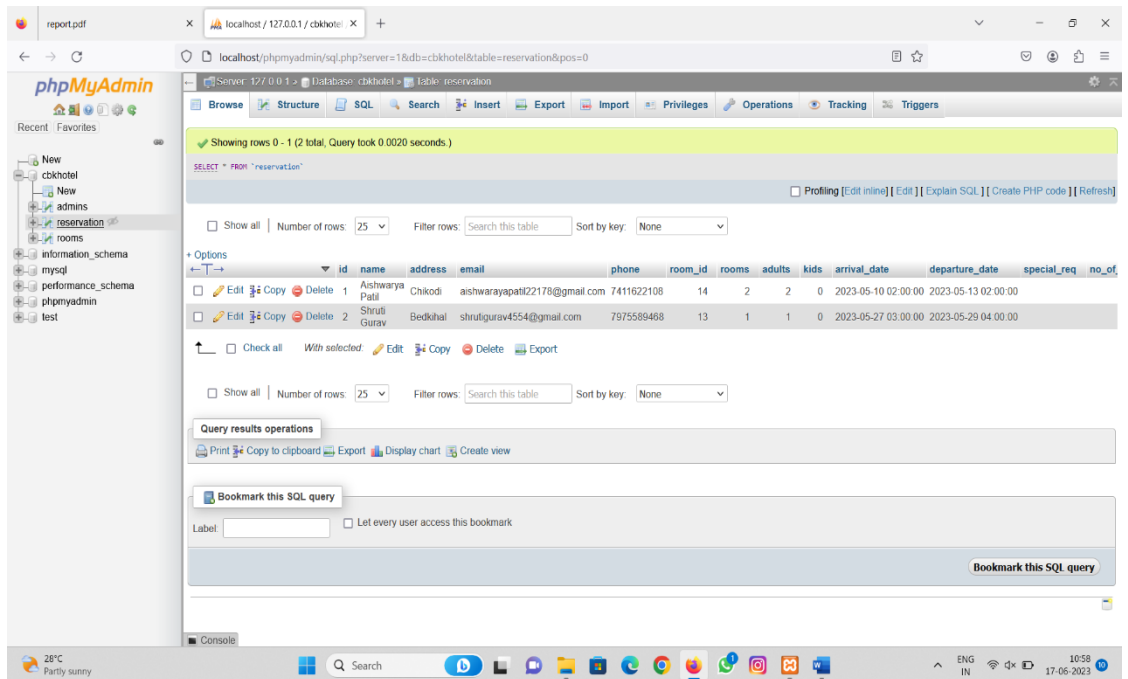


Figure 3.1.6.2

Reservation Table



Showing rows 0 - 1 (2 total, Query took 0.0020 seconds)

```
SELECT * FROM `reservation`
```

Options: Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

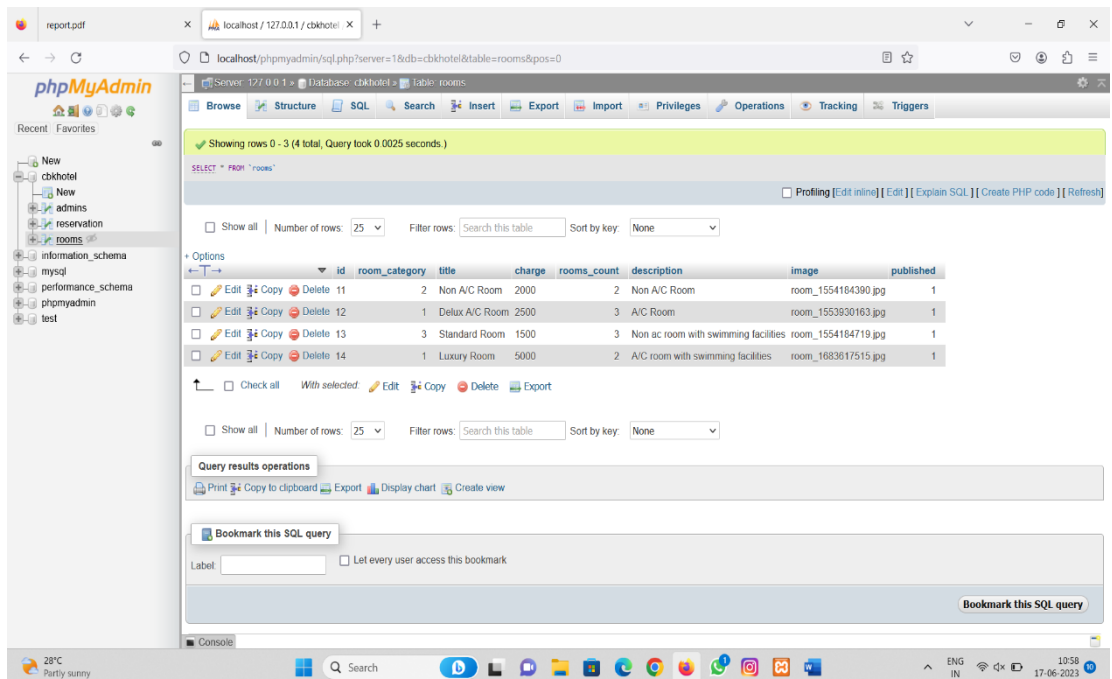
	id	name	address	email	phone	room_id	rooms	adults	kids	arrival_date	departure_date	special_req	no_of
<input type="checkbox"/>	1	Aishwarya Patil	Chikodi	aishwarayapati22178@gmail.com	7411622108	14	2	2	0	2023-05-10 02:00:00	2023-05-13 02:00:00		
<input type="checkbox"/>	2	Shruti Gurav	Bedkihal	shrutigurav4554@gmail.com	7975589468	13	1	1	0	2023-05-27 03:00:00	2023-05-29 04:00:00		

Query results operations: Print | Copy to clipboard | Export | Display chart | Create view

Bookmark this SQL query

Figure 3.1.6.3

Room Table



Showing rows 0 - 3 (4 total, Query took 0.0025 seconds)

```
SELECT * FROM `rooms`
```

Options: Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

	id	room_category	title	charge	rooms_count	description	image	published
<input type="checkbox"/>	11	2	Non A/C Room	2000	2	Non A/C Room	room_1554184390.jpg	1
<input type="checkbox"/>	12	1	Delux A/C Room	2500	3	A/C Room	room_1553930163.jpg	1
<input type="checkbox"/>	13	3	Standard Room	1500	3	Non ac room with swimming facilities	room_1554184719.jpg	1
<input type="checkbox"/>	14	2	Luxury Room	5000	2	A/C room with swimming facilities	room_1683617515.jpg	1

Query results operations: Print | Copy to clipboard | Export | Display chart | Create view

Bookmark this SQL query

Figure 3.1.6.4

3.1.7 Product Design

Home Page

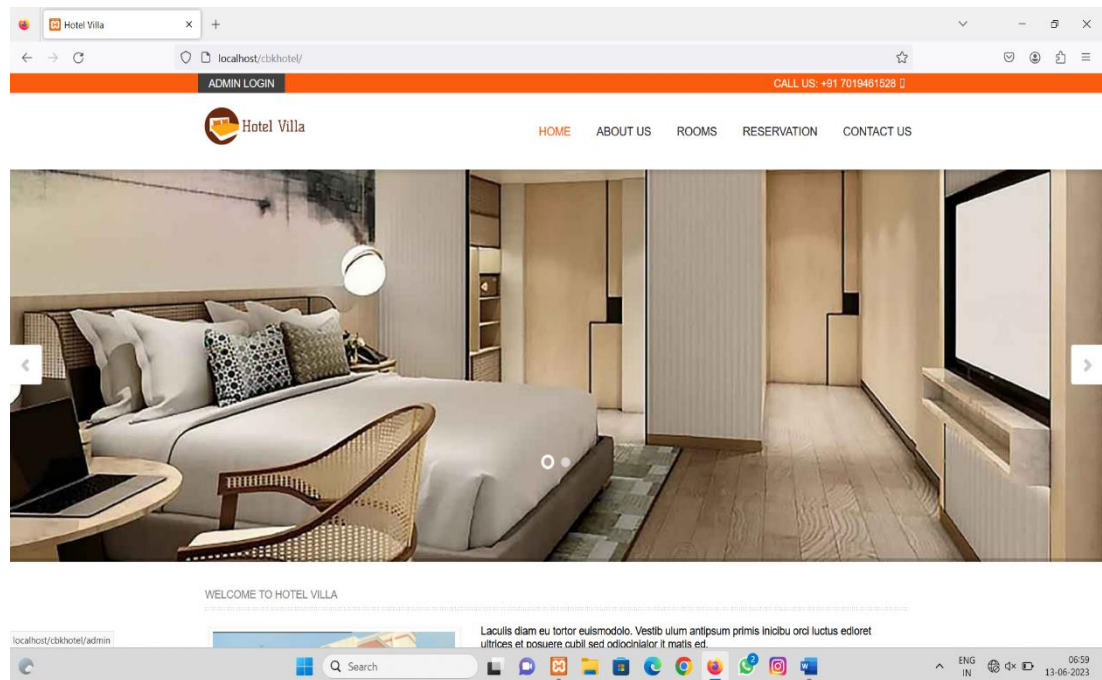


Figure 3.1.7.1

Admin Login Page

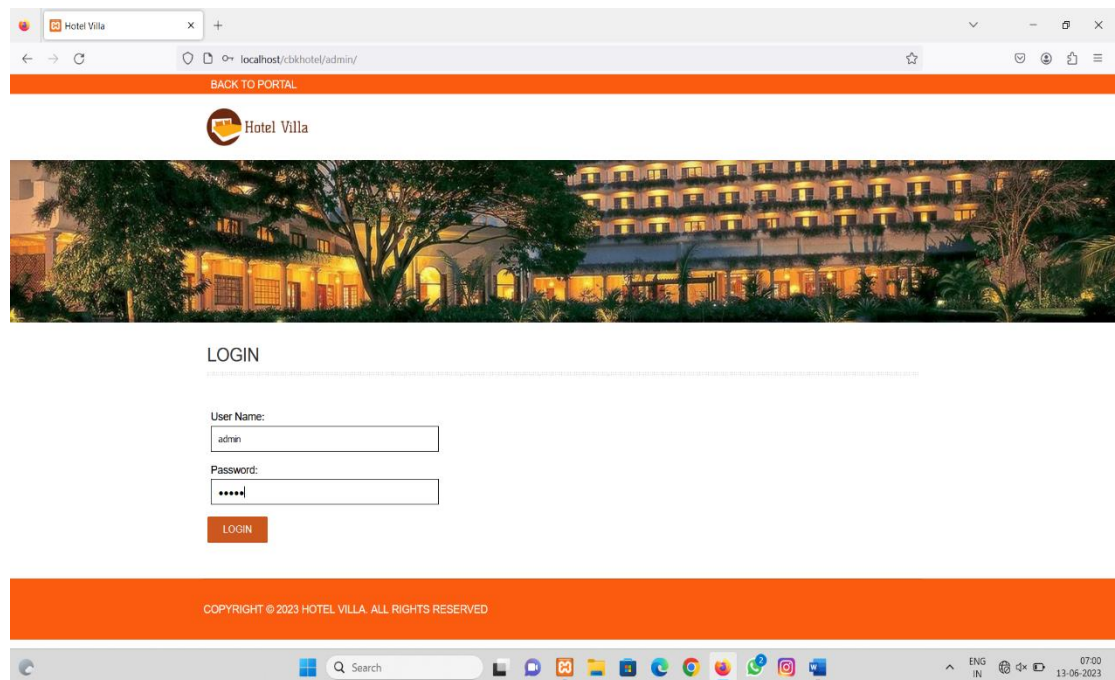


Figure 3.1.7.2

Dashboard

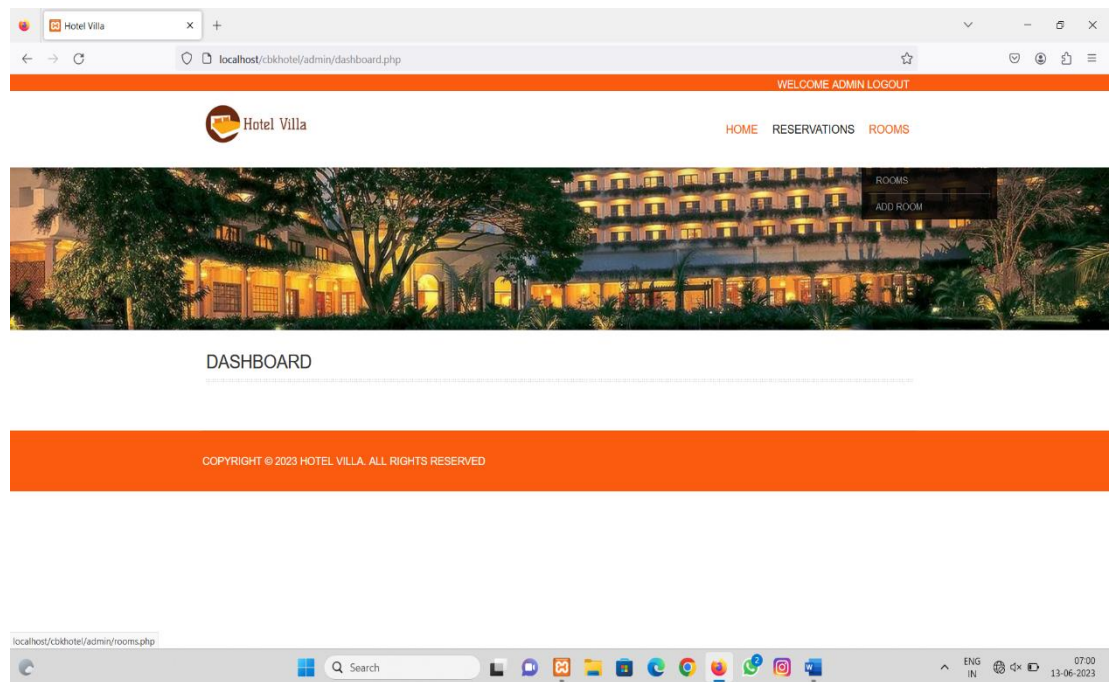


Figure 3.1.7.3

Room List page

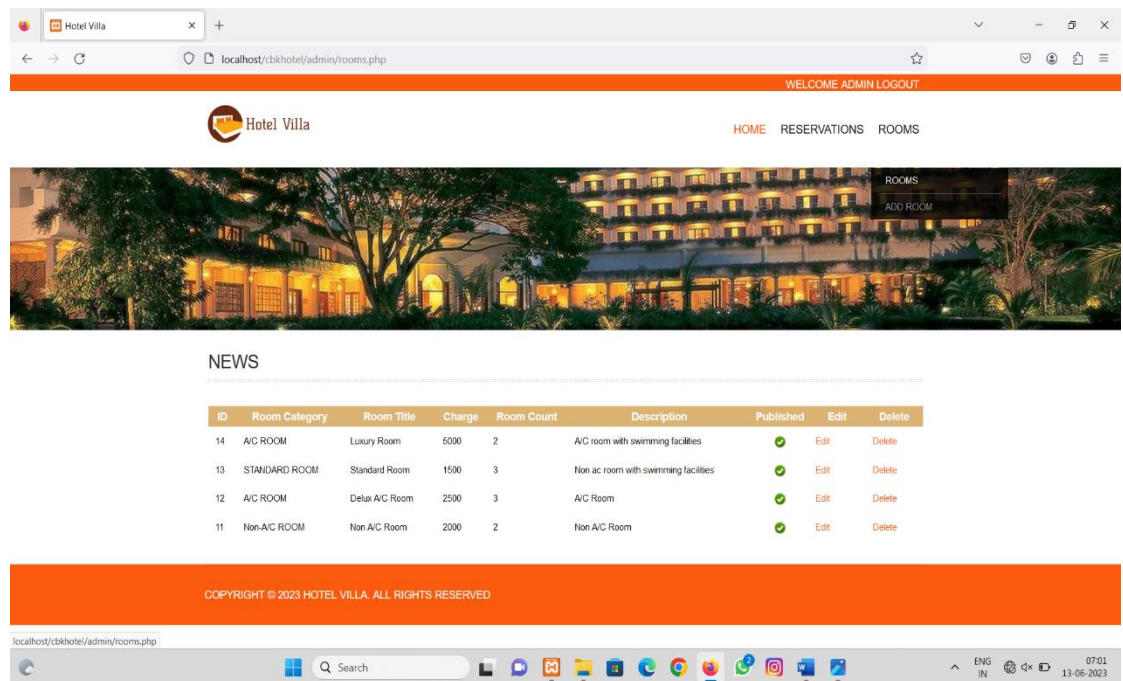


Figure 3.1.7.4

Add Room Page

Room Category:
Select Room Category

Room Title:

Charge:

Room Count:

Description:

Photo:
Browse... No file selected.

Published:
Yes

SAVE

Figure 3.1.7.5

View Room Booking page

WELCOME ADMIN LOGOUT

Hotel Villa

HOME RESERVATIONS ROOMS

ROOM RESERVATIONS

Name	Email	Phone	Room	Rooms	Adults	Kids	Arrival Date	Departure Date	Confirm	Delete
Shrus Gurav	shrutgurav4554@gmail.com	7975689468	Standard Room	1	1	0	27 May, 2023	29 May, 2023	✓	Delete
Aishwarya Patel	aishwaryapatel22178@gmail.com	7411622108	Luxury Room	2	2	0	10 May, 2023	13 May, 2023	✓	Delete

COPYRIGHT © 2023 HOTEL VILLA. ALL RIGHTS RESERVED

Figure 3.1.7.6

View Room Page

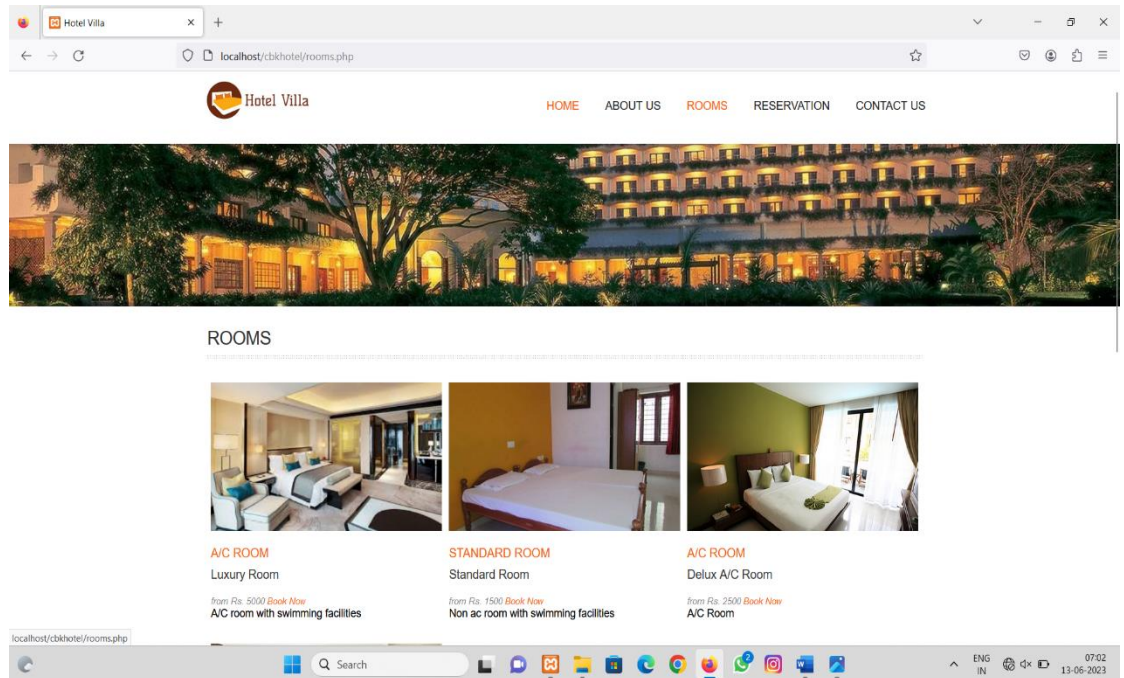


Figure 3.1.7.7

Book Room Page

The screenshot shows a web browser window with the address bar displaying 'localhost/cbkhote/reservation.php'. The page is titled 'SEND REQUEST FOR ROOM RESERVATION'. It contains a reservation form with the following fields:

- Arrival Date:
- Departure Date:
- Choose Room:
- No. of rooms:
- No. of adults:
- No. of Kids:
- Your Name:
- Your Address:
- Your email address:
- Your phone number:
- Special Requirements:

At the bottom of the form is a button labeled 'CALCULATE BILL'. The browser's taskbar at the bottom shows the Windows logo, a search bar, and various application icons. The system clock indicates 07:02 on 13-06-2023.

Figure 3.1.7.8

About Us Page

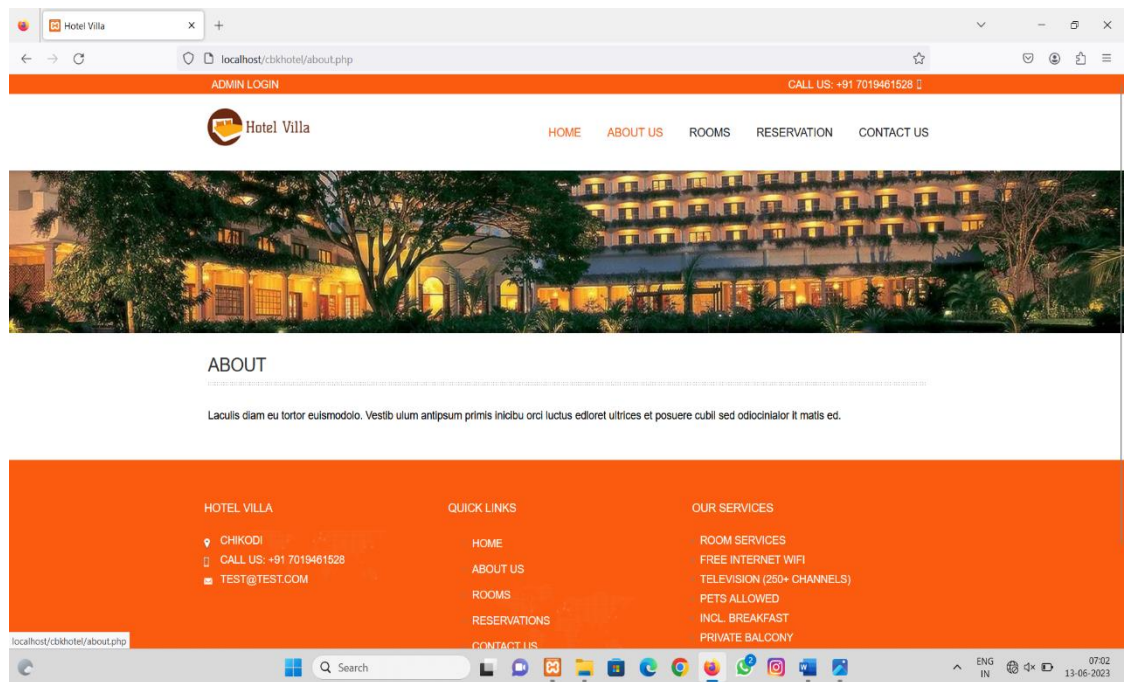


Figure 3.1.7.9

Contact Page Us

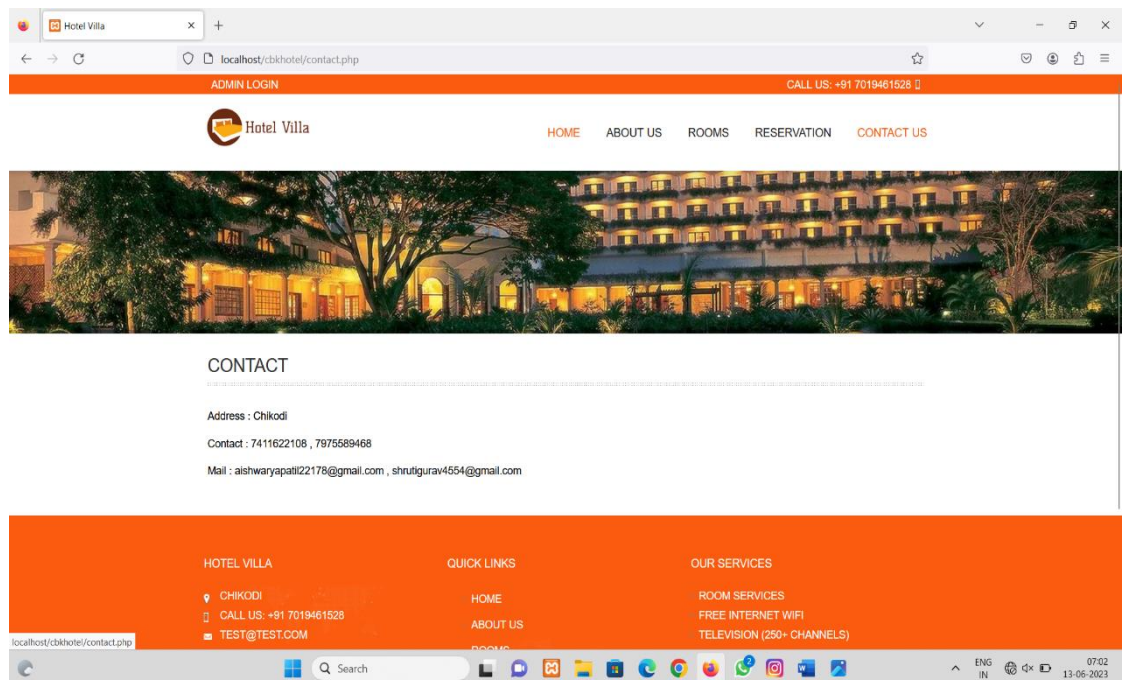


Figure 3.1.7.10

3.1.8 Fabrication

Admin Login

We first gathered the prerequisites for admin login, then we analyzed the information we had. The admin login form is then created. The page is validated after it has been designed, and each field of the form needs to be filled out correctly. After that, we connected the database. After finishing all the processes, the testing is finished.

Manage Room

In this admin is going to manage the room, add the room type, add the room photo, add the room information. For execution of this page, we have collected the requirements & analysed the requirements. Then we designed the page. After designing we have validated this page with entering some details. Then we performed the database connection. After this testing will take place.

View Room Booking

In this admin can view the booked room list, he can view the user details. we first gathered the requirements, then analyzed them, and finally designed the form. Following the evaluation of the designed page, we link to the database, and finally all the steps are finished. The testing is over.

View Room

In this user can view the room and view the room information and room type, photos etc . In this first we have collected the requirements then we analysed the requirements. After that we have designed the page. After that we validated form with each field with filling details. Then performed the database connection. After all this we tested the page comparing with each step of the page.

Check Room Available

In this user can check the availabilities of the room. In this first we have collected the requirements then we analysed the requirements. After that we have designed the page. After that we validated form with each field with filling details. Then performed the database connection. After all this we tested the page comparing with each step of the page.

Book Room

In this user can book the room after seeing the rooms and information about the room and hotel. In this first we have collected the requirements then we analysed the requirements. After that we have designed the page. After that we validated form with each field with filling details. Then performed the database connection. After all this we tested the page comparing with each step of the page.

CHAPTER 4

4.1 Test and Validation

4.1.1 Test Plan

Software testing can be stated as the process of verifying and validating whether a software or application is bug-free, meets the technical requirements as guided by its design and development, and meets the user requirements effectively and efficiently by handling all the exceptional and boundary cases.

4.1.2 Software testing can be divided into two steps:

Verification: it refers to the set of tasks that ensure that the software correctly implement a specific function

Validation: it refers to a different set of tasks that ensure that the software that has been built is traceable to customer requirements.

4.1.3 Testing Types

Manual Testing: -

Manual Testing is a kind of software testing in which a software tester develops and executes the test cases without using any automated testing tools. The main objective of manual testing is to detect the issues, bugs, and defects of a software application.

Unit Testing: -

Unit testing is a software development process in which the smallest testable parts of an application, are called units. The main objective of unit testing is to isolate written code to test and determine if it works as intended. Unit testing is an important step in the development process. If done correctly, unit tests can detect early flaws in code which may be more difficult to find in later testing stages.

Integration Testing: -

Integration testing is the phase in software testing in which individual software modules are combined and tested as a group. Integration testing is conducted to evaluate the compliance of a system or component with specified functional requirements.

White Box Testing: -

The technique of testing in which the tester is aware of the internal workings of the product, has access to its source code, and is conducted by making sure that all internal operations are performed according to the specifications is known as white box testing.

Black Box Testing: -

The technique of testing in which the tester doesn't have access to the source code of the software and is conducted at the software interface without any concern with the internal logical structure of the software is known as black-box testing.

4.1.2 Test Approach

Table 4.1.2.1

Admin Login

Test Case Id	Test Description	Test Steps	Test Data	Expected Result	Actual Result	Status
TC01	Check Admin Login with Valid Data	Enter Username And Password	Username = admin Password = admin	Admin Login Successfully	As Expected	Pass

Table 4.1.2.2

Manage Room

Test Case Id	Test Description	Test Steps	Test Data	Expected Result	Actual Result	Status
TC02	Check Room title and room charge and room categories	Enter Room title, Enter room charge, Enter Room categories	Room Title=Luxury Room Charge=4000 Room Categories=A/C	Added record successfully	As Expected	pass

Table 4.1.2.3

View Room Booking

Test Case Id	Test Description	Test Steps	Test Data	Expected Result	Actual Result	Status
TC03	Check Room Booking Status with valid data	Check Status	Confirm or Not Confirm	Confirm successfully	As Expected	pass

Table 4.1.2.4

Book Room:-

Test Case Id	Test Description	Test Steps	Test Data	Expected Result	Actual Result	Status
TC04	Check Room Booking Status with valid data	Enter Name, Enter phone number, Enter Check-In Check-Out Date	Name=Aishwarya, phone number=7411622108, Check-In date= 10-06-2023, Check-Out date= 12-06-2023	Booked Room successfully	As Expected	pass

4.1.3 Features Tested

Admin Login

Manage Room

View Room Booking

Book Room

4.1.4 Features not Tested

We have not tested the project on cloud server

4.1.5 Findings

Admin login

If admin doesn't provide any username in username field, we will receive a pop-up message as please enter username, and we expected the same result, hence the test is passed.

If admin doesn't provide any password in password field, we will receive a pop-up message as please enter password, and we expected the same result, hence the test is passed.

Manage Room

If admin doesn't provide room title in room title field, we will receive a pop-up message as please enter room title, and we expected the same result, hence the test is passed.

If admin doesn't provide room charge in room charge field, we will receive a pop-up message as please enter room charge, and we expected the same result, hence the test is passed.

If admin doesn't provide room Categories in room Categories field, we will receive a pop-up message as please enter room Categories, and we expected the same result, hence the test is passed.

View Room Booking

Admin uses the status field to confirm and not confirm in View Room Booking list.

Admin can delete the user

Book Room

If User doesn't provide name in name field, we will receive a pop-up message as please enter name, and we expected the same result, hence the test is passed.

If admin doesn't provide phone number in phone number field, we will receive a pop-up message as please enter phone number, and we expected the same result, hence the test is passed.

If User doesn't provide Check-In date in Check-In date field, we will receive a pop-up message as please enter Check-In date, and we expected the same result, hence the test is passed.

If User doesn't provide Check-Out date in Check-Out date field, we will receive a pop-up message as please enter Check-Out date, and we expected the same result, hence the test is passed.

4.1.6 Inference

Admin login

Admin login is really important in our project because admin manages everything related to users. Admin can add the room or manage the room, he can add the room type and room photos and information. Admin also can view the list of booked room and view the user details.

Manage Room

This page is only visible and handled by admin. Admin can add the room, he can add the room type, he can add the room photo and information. He also can delete the room.

View Room Booking

This page is only visible and handled by admin. Admin can view the list of booked room. And also he can view the user information.

View Room

This page is for the user he can view the rooms, he can view all information about the room and hotel.

Check Room Available

User can check the availabilities of the room. For there booking purpose.

Book Room

This page is for user after viewing rooms and checking the availabilities of room user can book the room which they want. He can book the room with all appropriate details.

CHAPTER 5

5.1 Business Aspects

5.1.1 The market and economic outlook of the project

The purpose of this project “Hotel Management System” involves maintaining various operations like admin can login to this system using correct admin username and password and admin can add the room or manage the room and also view the room booking list and cancellation of the room, cash billing and room service. user can check the room availability and also book the room through online.

We are going to develop this project using this modules like Admin module admin login, manage rooms, view room booking, In user module view room, check room availability and book room etc.

To implement this project we will use full stack languages like HTML, CSS, JavaScript, Bootstrap, PHP, MySQL database etc. Some changes could be done here to make it more reliable, more automatic and providing more features.

5.1.2 Features

Time line for project is limited.

Online payment process is not possible.

Cost is limited.

Required limited employees for project.

Easy to use.

5.1.3 How does the product/service fit into the competitive landscape ?

In order to assess how a hotel management system fits into the competitive landscape, it's important to consider the features, benefits, and unique selling points it offers compared to other existing solutions. While I don't have access to specific information about a particular hotel management system project, I can provide you with a general overview of how such systems typically fit into the competitive landscape. Evaluate the features provided by the hotel management system. Does it offer essential functionalities such as reservation management, room inventory, guest profiles, billing and integration with distribution channels. Assess how these features compare to those offered by competitors. User experience is crucial. A user-friendly interface and intuitive workflows can significantly enhance the system's appeal. Evaluate how the system simplifies complex tasks, streamlines operations, and reduces training time for hotel staff.

5.1.4 Possible capstone project clients / customer

Independent Hotels: Small to medium-sized hotels that operate independently and are looking to streamline their operations, automate processes, and improve guest experiences.

Hotel Chains: Large hotel chains that manage multiple properties and require a centralized system to efficiently manage reservations, room inventory, and guest data across all locations.

Hotel Management Schools/Colleges: Educational institutions offering hospitality or hotel management programs. They could use the hotel management system as a training tool for students to learn about hotel operations and management processes.

Event Spaces and Conference Centers: Venues that host events, conferences, and meetings. They may require a hotel management system that integrates with event management functionalities, such as room booking for attendees, event scheduling, and catering services.

Government Agencies: Government agencies responsible for managing government-owned accommodations, such as lodges or guesthouses. They may require a hotel management system to efficiently handle reservations, invoicing, and reporting for their properties.

5.2 Financial Consideration

5.2.1 Capstone project budget:45,500

5.2.2 Cost capstone projections needed for either for profit/non-profit options.

Table 5.2.2.1

Estimation	Cost (Rupees)
Labour Cost	25,000
Cost of Material	8,500
Net Profit	12,000
Total	45,500

5.3 Conclusion and Recommendations

5.3.1 Conclusion

As per the synopsis we will implement “Hotel Management System” with all the modules like Admin module admin login, Manage rooms, View room booking. In User module in this view room, check room availability and book room etc. All modules name we will implement our project error free and user friendly.

5.3.2 Future Scope

In future we will implement this project to live to an to any hotel.

We will develop the mobile application.

Program Code

Index.php

```
<!DOCTYPE html>

<html lang="en-US">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=1.0, user-scalable=1" />

    <title>Hotel Villa</title>

    <link rel="stylesheet" href="assets/css/layerslider.css"/>

    <link rel="stylesheet" href="style.css"/>

    <link rel="stylesheet" href="assets/css/flexslider.css"/>

    <link rel="stylesheet" href="assets/css/responsive.css"/>

    <link rel="stylesheet" href="assets/css/font-awesome.min.css"/>

    <link rel="stylesheet" href="assets/css/prettyPhoto.css"/>

    <link rel="stylesheet" href="assets/css/woocommerce-modify.css"/>

    <link rel="stylesheet" href="assets/css/skins/brown.css"/>

    <link rel="stylesheet" href="assets/css/flexslider/flexslider.css"
media="screen" />

    <!-- Browser Specical File-->
```

```
<!--[if IE 8]><link href="assets/css/ie8.css" rel="stylesheet" type="text/css"
/><![endif]-->
```

```
<!-- JavaScript -->
```

```
<script src="assets/js/jquery-1.9.1.min.js"></script>
```

```
<script src="assets/js/jquery.modernizr.js"></script>
```

```
<?php
```

```
require_once "hotelhelper.php";
```

```
$msg = "";
```

```
$helper = new HotelHelper();
```

```
?>
```

```
</head>
```

```
<body>
```

```
<div class="body-outer-wrapper">
```

```
<div id="body-wrapper" class="body-wrapper full-width-mode"
```

```
<?php require_once "header.php"; ?>
```

```
<div class="slider-outer-wrapper">
```

```
<div class="main_slider">
```

```
<div id="layerslider" style="width: 100%; height: 500px; margin: 0px auto; ">
```

```
<div class="ls-layer" style="slidedirection: top; slidedelay: 7000; durationin: 1500; durationout: 1500; delayout: 500;">
```

```

```

```
</div>
```

```
<div class="ls-layer" style="slidedelay: 6000;">
```

```

```

```
</div>
```

```
</div>
```

```
</div>
```

```
<div class="shadow-box"></div>
```

```
</div>
```

```
<div class="main-outer-wrapper has-slider">
```

```
<div class="main-wrapper container">
```

```
<div class="row row-wrapper">
```

```
<div class="page-outer-wrapper">
```

```
<div class="page-wrapper twelve columns no-sidebar b0">
```

```
<div class="row">
```

```
<div class="twelve columns">
```

```
<div class="builder-item-wrapper builder-editor">
```

```
<div class="builder-title-wrapper clearfix">
```

```
<h3 class="builder-item-title">Welcome to Hotel Villa</h3>
```

```
</div>
```

```
<div class="builder-item-content row">
```

```
<div class="twelve columns b0">
```


<p>Laculis diam eu tortor euismodolo. Vestibulum ant ipsum primis inibu orci
luctus edloret ultrices et posuere cubil sed odiocinialor it matis ed.</p>

<p>Eu tortor euismodolo. Vestibulum ant psu meop primis inibu orci luctus ed
ultrices et posuere cubil. Cura sed quis nibhut odiocinialor matis ed iaculis ante
sed vel laciniate. Eu tortor euismodolo. Vestibulum ant psu meop primis inibu
orci luctus ed ultrices et posuere cubil. Cura sed quis nibhut odiocinialor matis ed
iaculis ante sed vel laciniate.</p>

<p>Laculis diam eu tortor euismodolo. Vestibulum ant ipsum primis inibu orci
luctus edloret ultrices et posuere cubil sed odiocinialor it matis ed.</p>

</div>

</div>

</div>

</div>

<div class="clear"></div>

</div> <!-- END .row -->

<div class="row">

<div class="twelve columns b0">

<div class="builder-item-wrapper builder-rooms">

<div class="builder-title-wrapper clearfix">

<h3 class="builder-item-title">Highlight Rooms</h3>

View All Rooms →

```
</div>
```

```
<div class="builder-item-content row">
```

```
<div class="twelve columns b0">
```

```
<div class="cpt-items row clearfix isotope">
```

```
<?php
```

```
$helper->getLatestRooms();
```

```
?>
```

```
<?php /*
```

```
<div class="cpt-item four columns isotope-item rooms">
```

```
<div class="thumb-wrapper">
```

```
<div class="thumb-control-wrapper">
```

```
<ul class="thumb-control clearfix">
```

```
<!--
```

```
<li><a title="View Detail" href="#" class="go-detail">Open Detail</a></li>
```

```
<li><a rel="prettyPhoto[gallery1]" title="Room 1 Gallery 1"
```

```
href="assets/images/room_1.jpg" class="go-gallery">Open Gallery</a></li>
```

```
-->
```

```
</ul>
```

```
</div>
```

```
</div>
```

```
<div class="cpt-detail">
```

```
<ul>
```


</div>

</div>

<div class="cpt-item four columns isotope-item suites">

<div class="thumb-wrapper">

<div class="thumb-control-wrapper">

<ul class="thumb-control clearfix">

Open Detail

<a rel="prettyPhoto" title="Room 3 Gallery"
href="assets/images/suite_1.jpg" class="go-gallery">Open Gallery

</div>

</div>

<div class="cpt-detail">

<h2 class="cpt-title">SUP. A/C DELUXE ROOM</h2>

<div class="cpt-desc">from Rs. 3000 per night Book
Now</div>

</div>

</div>

<div class="cpt-item four columns isotope-item rooms">

<div class="thumb-wrapper">

```



<div class="thumb-control-wrapper">

<ul class="thumb-control clearfix">

<li><a title="View Detail" href="#" class="go-detail">Open Detail</a></li>

<li><a rel="prettyPhoto" title="Room 3 Gallery"
href="assets/images/room_3.jpg" class="go-gallery">Open Gallery</a></li>

</ul>

</div>

</div>

<div class="cpt-detail">

<h2 class="cpt-title"><a href="#">DELUXE A/C ROOM</a></h2>

<div class="cpt-desc">from Rs.2500 per night <a href="reservation.php">Book
Now</a></div>

</div>

</div>

*/ ?>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

</div>

```

```

<div class="clear"></div>

```

```

</div> <!-- END .row -->

```

</div> <!-- END .page-wrapper -->

<div class="clear"></div>

</div> <!-- END .page-outer-wrapper -->

</div>

</div> <!-- END .main-wrapper -->

</div> <!-- END .main-outer-wrapper -->

<?php require_once "footer.php"; ?>

</div><!-- END .body-wrapper -->

</div><!-- END .body-outer-wrapper -->

<!-- Javascript -->

<script src="assets/js/ddsmoothmenu.js"></script>

<script src="assets/js/purl.js"></script>

<script src="assets/js/jquery.prettyPhoto.js"></script>

<script src="assets/js/jquery.fitvids.js"></script>

<script src="assets/js/jquery.imagesloaded.min.js"></script>

<script src="assets/js/jquery.isotope.min.js"></script>

```

<!-- Included LayerSlider -->

<script src="assets/js/jquery-easing-1.3.js"></script>

<script src="assets/js/jquery-transit-modified.js"></script>

<script src="assets/js/layerslider.transitions.js"></script>

<script src="assets/js/layerslider.kreaturamedia.jquery.js"></script>


<script type='text/javascript'>

/* <![CDATA[ */

var FS =

{"animation":"slide","pauseOnHover":"true","controlNav":"true","directionNav":
"false","animationDuration":"500","slideshowSpeed":"5000","pauseOnAction":"f
alse"};

/* ]]> */

</script>


<script src="assets/js/flexslider/jquery.flexslider.js"></script>


<script src="assets/js/jquery.flexslider.js"></script>

<script src="assets/js/jquery-ui.datepicker.js"></script>

<script src="assets/js/shortcodes.js"></script>

<script src="assets/js/custom.js"></script>


</body>

</html>

```

Admin login .php

```
<!DOCTYPE html>

<html lang="en-US">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0,
maximum-scale=1.0, user-scalable=1" />

    <title>Hotel Villa</title>

    <link rel="stylesheet" href="../style.css"/>

    <link rel="stylesheet" href="../assets/css/responsive.css"/>

    <link rel="stylesheet" href="../assets/css/font-awesome.min.css"/>

    <link rel="stylesheet" href="../assets/css/prettyPhoto.css"/>

    <link rel="stylesheet" href="../assets/css/woocommerce-modify.css"/>

    <link rel="stylesheet" href="../assets/css/skins/brown.css"/>

    <link rel="stylesheet" href="../custom.css"/>

    <!-- Browser Specical File-->

    <!--[if IE 8]><link href="../assets/css/ie8.css" rel="stylesheet" type="text/css"
/><![endif]-->

    <!-- JavaScript -->

    <script src="../assets/js/jquery-1.9.1.min.js"></script>

    <script src="../assets/js/jquery.modernizr.js"></script>
```

```

<?php

require_once "hoteladminhelper.php";

if($_SESSION['userid'])

{

    echo "<script>window.location='dashboard.php';</script>";

}


$error = $_GET['error'];

?>

<script type="text/javascript">

function validate_form()

{

    var username = document.getElementById("username").value;

    var password = document.getElementById("password").value;


    if(username=="")

    {

        alert("Please Enter User Name.");

        return false;

    }

}

```

```

else if(password=="")

{

alert("Please Enter Password.");

return false;


}

}

</script>

</head>

<body>

<div class="body-outer-wrapper">

<div id="body-wrapper" class="body-wrapper full-width-mode">

<?php require_once "header.php"; ?>


<div class="titlebar-outer-wrapper"
style="background:url('../assets/images/titlebar-bg.jpg') no-repeat top center;">

<div class="shadow-box"></div>

</div>

<div class="main-outer-wrapper has-titlebar">

<div class="main-wrapper container">

```

```

<div class="row row-wrapper">

  <div class="page-outer-wrapper">

    <div class="page-wrapper twelve columns no-sidebar b0">

```

```

      <div class="row">

        <div class="twelve columns b0">

          <div class="page-title-wrapper">

            <h1 class="page-title left">Login</h1>

            <div class="page-title-alt right">

              </div>

            <div class="clear"></div>

          </div>

        </div>

      </div>

```

```

    </div>

    <div class="row">

      <div class="content-wrapper eight columns">

        <div class="text-content">

```

```

          <form id="reservation_form" class="reservation_form reservation_form_page"
            action="checklogin.php" method="post" onsubmit="return validate_form();">

            <div class="form-row notice_bar">

```



```

<?php

if($error)

{

echo '<p class="notice notice_error">Login failed!!! Please enter valid
details.</p>';

}

?>

</div>

<div class="row">

<div class="six columns b0">

<div class="form-row field_text">

<label>User Name:</label><br/>

<input id="username" class="input_text required" type="text" value="" name
="username"/>

</div>

<div class="form-row field_text">

<label>Password:</label><br/>

<input id="password" class="input_text required" type="password" value=""
name="password"/>

</div>

</div>

<div class="six columns b0">

```

```

</div>

<div class="clear"></div>

</div>

<div class="form-row field_submit mt20">

<input type="submit" value="Login" id="" class="btn"/>

</div>

</form>

</div> <!-- END .text-content -->

</div>

<div class="clear"></div>

</div> <!-- END .row -->

</div> <!-- END .page-wrapper -->

<div class="clear"></div>

</div> <!-- END .page-outer-wrapper -->

</div>

</div> <!-- END .main-wrapper -->

</div> <!-- END .main-outer-wrapper -->

<?php require_once "footer.php"; ?>

</div><!-- END .body-wrapper -->

</div><!-- END .body-outer-wrapper -->

```

```
<!-- Javascript -->

<script src="../../assets/js/ddsmoothmenu.js"></script>

<script src="../../assets/js/purl.js"></script>

<script src="../../assets/js/jquery.prettyPhoto.js"></script>

<script src="../../assets/js/jquery.fitvids.js"></script>

<script src="../../assets/js/jquery.imagesloaded.min.js"></script>

<script src="../../assets/js/jquery.isotope.min.js"></script>

<script src="../../assets/js/jquery-ui.datepicker.js"></script>

<script src="../../assets/js/shortcodes.js"></script>

<script src="js/custom.js"></script>


</body>

</html>
```

References

Books:-

- [1] “Web Programming”, by ‘Chris Bates’ Wiley Dream tech India, 2nd Edition.
- [2] “Software Engineering”, Ian Somerville, Sixth Edition, Pearson Education Ltd.
- [3] “HTML Complete References” Easy steps to develop web pages.
- [4] “PHP Complete Reference”

Websites:-

- [1] <http://en.wikipedia.org/wiki/PHP> for PHP.
- [2] <http://www.hotscripts.com/category/php/> for PHP.
- [3] <http://www.mysql.com/click.php?e=35050> for MYSQL.
- [4] <http://www.w3schools.com> for information on HTML.

Appendices

Chapter 1

1.1 Introduction	1
1.2 Scope of the capstone project	2-4

Chapter 2

2.1 Capstone project planning	
2.1.1 Work breakdown structure	5-6
2.1.2 Timeline developer-schedule	7-9
2.1.3 Cost breakdown structure	10-11
2.1.4 Capstone project risk assessment	12
2.2 Requirements specification	
2.2.1 Functional	13
2.2.2 Non-functional (Quality attributes)	14
2.2.3 User input	15
2.2.4 Technical constraints	16
2.3 Design specifications	
2.3.1 Chosen system design	17
2.3.2 Discussion of alternative design	18-20
2.3.3 Detailed description of components/subsystems	21-22
2.3.4 Components 1-n	23

Chapter 3

3.1 Approach and methodology	
3.1.1 Discuss the technology	24-25
3.1.2 Methodologies	26

3.1.3 Use cases	27
3.1.4 Programming	28-29
3.1.5 Analysis	30
3.1.6 Process design	31-32
3.1.7 Product design	33-37
3.1.8 Fabrication	38-39

Chapter 4

4.1 Test and validation	
4.1.1 Test plan	40-41
4.1.2 Test approach	42-43
4.1.3 Features tested	44
4.1.4 Features not tested	44
4.1.5 Findings	44-45
4.1.6 Inference	46

Chapter 5

5.1 Business aspects	
5.1.1 The market and economic outlook of the project	47
5.1.2 Features	47
5.1.3 How does the project of service fit into the competitive landscape?	48
5.1.4 Who want the possible client or customer?	48-49
5.2 Financial considerations	
5.2.1 Capstone project budget	50

5.2.2 Cost of capstone projection needed for either profit or nonprofit option	50
5.3 Conclusion and recommendations	
5.3.1 State of completion of capstone project	51
5.3.2 Future work	51
Program Code	52-67
References	68
Appendices	69-73

LIST OF FIGURES

Figure number	Figure name	Page number
Figure 2.3.1.1	System Architecture	17
Figure 2.3.2.1	Data flow diagram level 0	18
Figure 2.3.2.2	Data flow diagram level 1 admin	19
Figure 2.3.2.3	Data flow diagram level 1 user	20
Figure 2.3.3.1	Component diagram	21
Figure 3.1.2.1	Agile methodology diagram	26
Figure 3.1.3.1	Use case	27
Figure 3.1.6.1	List of tables	31
Figure 3.1.6.2	Admin table	31
Figure 3.1.6.3	Reservation Table	32
Figure 3.1.6.4	Room Table	32
Figure 3.1.7.1	Home Page	33
Figure 3.1.7.2	Admin Login page	33
Figure 3.1.7.3	Dashboard	34
Figure 3.1.7.4	Room List Page	34
Figure 3.1.7.5	Add Room Page	35
Figure 3.1.7.6	View Room Booking	35
Figure 3.1.7.7	View Room Page	36
Figure 3.1.7.8	Book Room Page	36
Figure 3.1.7.9	About Us	37
Figure 3.1.7.10	Contact	37

LIST OF TABLES

Table number	Table name	Page number
Table 1.2.1	Cost of the project	4
Table 2.1.3.1	Final check	10
Table 2.1.3.2	Labour cost of work	11
Table 4.1.2.1	Admin Login	42
Table 4.1.2.2	Manage Room	42
Table 4.1.2.3	View Room Booking	43
Table 4.1.2.4	Book Room	43
Table 5.2.2.1	Cost (profit/nonprofit)	50