VISVESVARAYA TECHNOLOGICAL UNIVERSITY

JNANASANGAMA, BELAGAVI – 590018

DBMS Mini Project Report

On

HOTEL ROOM RESERVATION SYSTEM

Submitted in partial fulfillment for the award of degree of

Bachelor of Engineering

In

Artificial Intelligence and Machine Learning

Submitted by

AKSHAY SATHEESH

1RN20AI010

RNS INSTITUTE OF TECHNOLOGY

(**AICTE** Approved, **VTU** Affiliated and **NAAC 'A'** Accredited)
(**UG programs** – CSE, ECE, ISE, EIE and EEE are **Accredited** by **NBA** up to **30.6.2025)**
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098

Department of AI & ML

2022 – 2023

RNS INSTITUTE OF TECHNOLOGY

(**AICTE** Approved, **VTU** Affiliated and **NAAC ‘A’** Accredited)
(**UG programs** – CSE, ECE, ISE, EIE and EEE are **Accredited** by **NBA** up to **30.6.2025)**
Channasandra, Dr. Vishnuvardhan Road, Bengaluru - 560 098

Department of AI & ML

CERTIFICATE

Certified that the Project entitled Hotel Room Reservation carried out by Mr. **Akshay Satheesh** USN **1RN20AI010** a bonafide student of V Semester BE, **RNS Institute of Technology** in partial fulfillment for the Bachelor of Engineering in AI & ML ENGINEERING of the **Visvesvaraya Technological University**, Belagavi during the year 2022-23. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report. The Project report has been approved as it satisfies the academic requirements in respect of Database Management System with Mini Project Laboratory prescribed for the said Degree.

Course Teacher
Ms. Sajitha N
Asst. Professor
Department of AI&ML
RNSIT, Bengaluru

HoD
Dr. Harsha S
Department of AI & ML
RNSIT, Bengaluru

Name & Signature

Examiner 1:

Examiner 2:

ACKNOWLEDGEMENT

At the very onset, we would like to place on record our gratitude to all those people who have helped us in making this project work a reality. Our Institution has played a paramount role in guiding us in the right direction.

We would like to profoundly thank **Sri. Satish R Shetty**, Chairman, RNS Group of Institutions. Bangalore for providing such a healthy environment for the successful completion of this project work.

We would also like to thank our beloved Principal, **Dr. M K Venkatesha**. for providing the necessary facilities to carry out this work.

We are extremely grateful to **Dr. Harsha S**, Head of the Department of Artificial Intelligence and Machine Learning, for having accepted to guide me in the right direction with all his wisdom.

We would like to express our sincere thanks to our Course Teacher **Ms. Sajitha N** Assistant Professor, Department of Artificial Intelligence and Machine Learning for her constant encouragement that motivated us for the successful completion of this project work.

We would like to thank our guide, **Ms. Kavyashree H L** Assistant Professor, Department of Artificial Intelligence and Machine Learning for her continuous guidance and constructive suggestions for this Project work.

Last but not the least. we are thankful to all the teaching and non-teaching staff members of the Artificial Intelligence and Machine Learning Department for their encouragement and support throughout this work.

Signature

Akshay Satheesh

1RN20AI010

ABSTRACT

The Hotel Room Reservation System is a project implemented for Alpha Royal Hotel, which is an imaginary hotel. It provides people all Over the world with an easy and fast way to book hotel rooms online. The interface of The Hotel Room Reservation System is Web pages that can be accessed with a Web site browser. The system is implemented in PHP (Hypertext Preprocessor) and HTML (Hyper Text Markup Language).Users can perform room booking activities at Alpha Royal Hotel anytime and anywhere by accessing it via Internet. The Hotel Room Reservation System is an easy-to-use application. Everyone who knows how to use a Web browser can easily carry out booking, change the booking details, cancel the booking, change the personal profile, view the booking history, or view the hotel information by following its simple and clear GUI (Graphical user interface) design.

TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW OF DATABASE MANAGEMENT SYSTEMS

The essential feature of database technology is that it provides an internal representation (model) of the external world of interest. Examples are, the representation of a particular date/time/flight/aircraft in an airline Reservation or of the item code/item description/quantity on hand/reorder level/reorder quantity in a stock control system.

The technology involved is concerned primarily with maintaining the internal representation consistent with external reality; this involves the results of extensive R&D over the past 30 years in areas such as user requirements analysis, data modelling, process modelling, data integrity, concurrency, transactions, file organisation, indexing, rollback and recovery, persistent programming, object-orientation, logic programming, deductive database systems, active database systems... and in all these (and other) areas there remains much more to be done. The essential point is that database technology is a CORE TECHNOLOGY which has links to:

- Information management / processing
- Data analysis / statistics
- Data visualization / presentation
- Multimedia and hypermedia
- Office and document systems
- Business processes, workflow, CSCW (computer-supported cooperative work)

Relational DBMS is the modern base technology for many business applications. It offers flexibility and easy-to-use tools at the expense of ultimate performance. More recently relational systems have started extending their facilities in directions like information retrieval, object- orientation and deductive/active systems which lead to the so-called 'Extended Relational Systems'.

Information Retrieval Systems began with handling library catalogues and then extended to full free-text by utilizing inverted index technology with a lexicon or thesaurus. Modern systems utilize some KBS (knowledge-based systems) techniques to improve the retrieval.

Object-Oriented DBMS started for engineering applications in which objects are complex, have versions and need to be treated as a complete entity. OODBMSs share many of the OOPL features such as identity, inheritance, late binding, overloading and overriding. OODBMSs have found favours in engineering and office systems but haven’t been successful yet in traditional application areas.

Deductive / Active DBMS has evolved over the last 20 years and combines logic programming technology with database technology. This allows the database itself to react to the external events and also to maintain its integrity dynamically with respect to the real world.

CHARACTERISTICS OF DATABASE APPROACH

Traditional form included organising the data in file format. DBMS was a new concept then, and all kinds of research was done to make it overcome the deficiencies in traditional style of data

management. A modern DBMS has the following characteristics −

- Real-world entity − A modern DBMS is more realistic and uses real-world entities to design its architecture. It uses behaviour and attribute too. For example, a school database may use students as an entity and their age as an attribute.

- Relation-based tables − DBMS allows entities and relations to form tables.

  A user can understand the architecture of a database by just looking at the table names.

- Isolation of data and application − A database system is entirely different than its data. A database is an active entity, whereas data is said to be passive, on which the database works and organizes. DBMS also stores metadata, which is data about data, to ease its own process.

- Less redundancy − DBMS follows the rules of normalization, which splits a relation when any of its attributes has redundancy in its values. Normalization is a mathematically rich and scientific process that will reduces the data redundancy.
- Consistency − Consistency is a state where every relation in a database remains consistent. There exists methods and techniques, that can detect an attempt of leaving database in an inconsistent state. DBMS can provide greater consistency as compared to earlier forms of data storing applications like file-processing systems.

- Query Language − DBMS is equipped with query language, which makes it more efficient to retrieve and manipulate data. A user can apply as many and the filtering options as required to retrieve a set of data. Traditionally it was not possible where file-processing system was used.

- ACID Properties − DBMS follows the concepts of Atomicity, Consistency, Isolation, and Durability (normally shortened as ACID). These concepts are applied on transactions, which manipulate data in a database. ACID properties help the database to stay healthy in multi-transactional environments and also in case of failure.

- Multiuser and Concurrent Access − DBMS supports multi-user environment and allows them to access and manipulate data in parallel. Though there are restrictions on transactions when users attempt to handle the same data item, but users are always unaware of them.

- Multiple views − DBMS offers multiple views for different users. A user in the

  Sales department will have a different view of the database from the person working in the Production department. This feature enables the users to have a concentrate view of the database according to their requirements.

- Security − Features like multiple views offer security to certain extent when users are unable to access the data of other users and departments. DBMS offers methods to impose constraints while entering data into the database and retrieving the same at a later stage.

DBMS offers many different levels of security features, which enables multiple users to have different views with different features. For example, a user in the Sales department cannot see the data that belongs to the Purchase department. It can also be helpful in deciding how much data of the Sales department should be displayed to the user. Since a DBMS is not saved on the disk as traditional file systems, it is very hard for miscreants to break the code.

APPLICATIONS OF DBMS

Applications of Database Management Systems :

- **Telecom**: There is a database to keeps track of the information regarding the calls made, network usage, customer details etc. Without the database system it is hard to maintain such huge amounts of data which gets updated every millisecond.
- **Industry**: Whether it is a manufacturing unit, a warehouse or a distribution centre, each one needs a database to keep the records of the ins and outs. For example, a distribution centre should keep a track of the product units that were supplied to the centre as well as the products that got delivered from the distribution centre on each day; this is where DBMS comes into picture.
- **Banking System**: For storing information regarding a customer, keeping a track of his/her day to day credit and debit transactions, generating bank statements etc is done with through Database management systems.
- **Education sector**: Database systems are frequently used in schools and colleges to store and retrieve the data regarding the student , staff details, course details, exam details, payroll data, attendance details, fees details etc. There is lots of inter-related data that needs to be stored and retrieved in an efficient manner.

- **Online shopping**: You must be aware of the online shopping websites such as Amazon, Flip kart etc. These sites store the product information, your addresses and preferences, credit details and provide you the relevant list of products based on your query. All this involves a Database management system.

1.2 PROBLEM DESCRIPTION/STATEMENT

The Hotel Room Reservation System deals with a simple website where customers can Reserve/book hotel rooms.

This system involves the following functionalities:

- It consists of user categories such as rooms, admins and users.
- Each user can use the website to their own requirements.
- The website acts as a medium between the hotel manager and the user. The customer can log-in and book rooms and admin can view and edit the details .
- Admin Functionality: The admin can add new categories of user, new rooms of the hotel and perform the tasks of the managers.
- User Functionality: The user can access/view the rooms booked but have no other functionalities for better security.
- The website provides options to update the information of each user.
- It also provides information about the hotel such as its total sales for a month, year, etc.

1.3 OBJECTIVES

The primary objective of online hotel reservation systems is to provide customers with a convenient and efficient way to book their accommodations.

This includes providing an easy-to-use interface, secure payment processing, and access to up-to-date information about available rooms.

Additionally, these systems should be able to quickly process reservations so that customers can get the room they need in a timely manner.

Overall, it appears that these objectives are being met by most online hotel reservation systems today.

Customers have access to detailed information about each property before making their booking decision and can easily complete the transaction without any issues or delays.

CHAPTER 2

SYSTEM REQUIREMENTS

2.1 HARDWARE REQUIREMENTS

The Hardware requirements are very minimal and the program can be run on most of the machines.

Processor	:	i5 processor
Processor Speed	:	1.2 GHz
RAM	:	1 GB
Storage Space	:	40 GB
Monitor Resolution	:	1024*768 or 1336*768 or 1280*1024

2.1.1 SOFTWARE REQUIREMENTS

1. Operating System used: Windows 11

2. Technologies used: HTML, CSS, PHP, Bootstrap

3. XAMPP Server: MySQL, PhpMyAdmin

4. IDE used: Visual Studio Code

5. Browser that supports HTML

CHAPTER 3

SYSTEM DESIGN

3.1 E R DIAGRAM

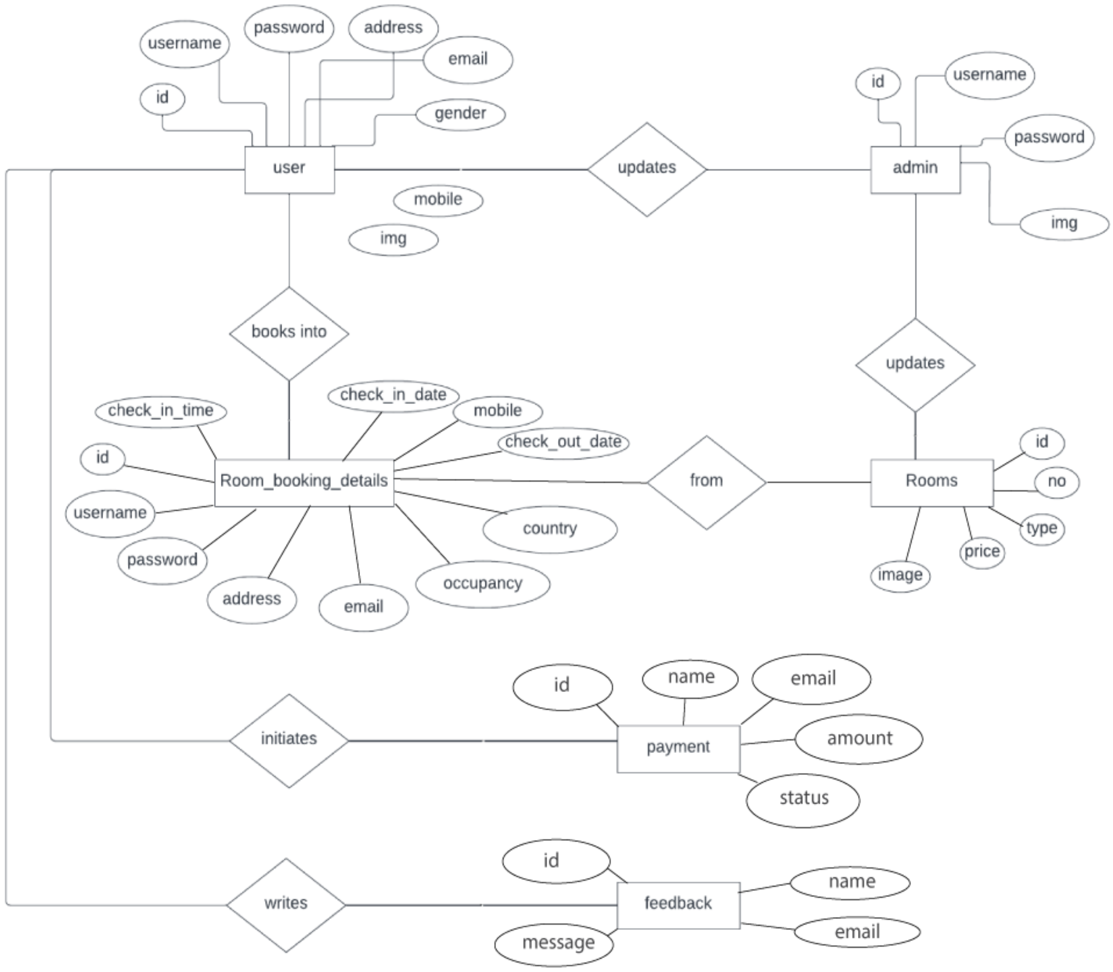


Figure 3. 1 E R Diagram

3.2 SCHEMA DIAGRAM

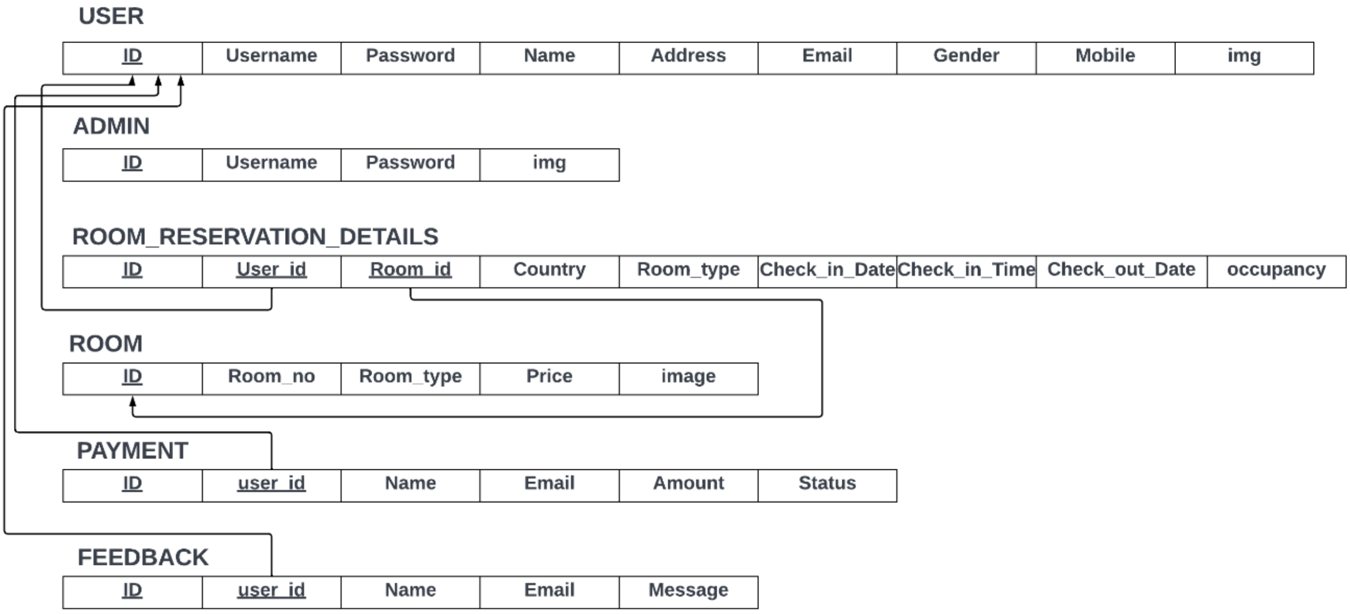


Figure 3. 2 Schema Diagram

3.3 OVERVIEW OF GUI

3.3.1 HTML

Hypertext Markup Language (HTML) is the standard markup language for creating web pages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web. Web browsers receive HTML documents from a web server or from a local storage and render them to multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

HTML elements are the building blocks of HTML pages. With HTML constructs, images and other objects like interactive forms can be embedded into the rendered page. It provides a way to create structured documents by denoting structural semantics for the text like headings, paragraphs, lists, links, quotes and other items. HTML elements are delimited by tags that are written within angle brackets. Tags such as <img /> and <input /> introduce content into the page directly. Other tags such as <p>...</p> surround and provide information about document text and may include other tags as sub-elements. Browsers do not display the HTML tags, but use them to interpret the content of the page.

HTML can also embed programs written in a scripting language such as JavaScript which affect the behaviour and content of web pages. Inclusion of CSS defines the look and layout of content.

3.3.2 CSS

Cascading Style Sheets (CSS) is a style sheet language which is used for describing the presentation of a document written in a markup language. Although most often its used to set the visual style of web pages and user interfaces written in HTML and XHTML, the language can be applied to any XML document, including plain XML, SVG and XUL, and is also applicable to rendering in speech, or on other media. Along with HTML and JavaScript, CSS is a cornerstone technology used by most websites to create visually engaging webpages, user interfaces for web applications, and user interfaces for many mobile applications.

CSS is designed primarily to enable the separation of presentation and content, including aspects such as the layout, colours, and fonts. This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple HTML pages to share the formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

3.3.3 PHP

PHP is a server-side scripting language designed primarily for web development but is also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team. PHP originally stood for Personal Home Page, but it now stands for the recursive acronym PHP: Hypertext Pre-processor.

PHP code can be embedded into HTML or HTML5 markup, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server software combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code can also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

The standard PHP interpreter, powered by the Zend Engine, is a free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers, on almost every operating system and platform, free of charge. The PHP language evolved without a written formal specification or standard until 2014, leaving the canonical PHP interpreter as a de facto standard. Since 2014 work has gone into creating a formal PHP specification. HP development began in 1995 when Rasmus Lerdorf wrote several Common Gateway Interface (CGI) programs in C, which he used in order to maintain his personal homepage. He extended them to work with web forms and to communicate with databases, and called this implementation "Personal Home Page/Forms Interpreter" or PHP/FI.

PHP/FI could help to build simple, dynamic web applications. To accelerate bug reporting and to improve the code, Lerdorf initially announced the release of PHP/FI as "Personal Home Page Tools (PHP Tools) version 1.0" on the Usenet discussion group on June
8, 1995 This release already had the basic functionality that PHP has as of 2013. This included Perl-like variables, form handling, and the ability to embed HTML. The syntax resembled that of Perl but was simpler, more limited and less consistent.

3.3.4 XAMPP Server

XAMPP is an abbreviation where X stands for Cross-Platform, A stands for Apache, M stands for MYSQL, and the Ps stand for PHP and Perl, respectively. It is an open-source package of web solutions that includes Apache distribution for many servers and command-line executables along with modules such as Apache server, MariaDB, PHP, and Perl.

XAMPP helps a local host or server to test its website and clients via computers and laptops before releasing it to the main server. It is a platform that furnishes a suitable environment to test and verify the working of projects based on Apache, Perl, MySQL database, and PHP through the system of the host itself. Among these technologies, Perl is a programming language used for web development, PHP is a backend scripting language, and MariaDB is the most vividly used database developed by MySQL.

3.4 NORMALIZATION

If a database design is not perfect, it may contain anomalies, which are like a bad dream for any database administrator. Managing a database with anomalies is next to impossible. Update anomalies − If data items are scattered and are not linked to each other properly, then it could lead to strange situations. For example, when we try to update one data item having its copies scattered over several places, a few instances get updated properly while a few others are left with old values. Such instances leave the database in an inconsistent state.

Deletion anomalies − We tried to delete a record, but parts of it was left undeleted because of unawareness, the data is also saved somewhere else.

Insert anomalies − We tried to insert data in a record that does not exist at all.

Normalization is a method to remove all these anomalies and bring the database to a consistent state.

3.4.1 First Normal Form

A relation is in 1NF if every attribute is a single-valued attribute or it does not contain any multi-valued or composite attribute, i.e., every attribute is an atomic attribute. If there is a composite or multi-valued attribute, it violates the 1NF. To solve this, we can create a new row for each of the values of the multi-valued attribute to convert the table into the 1NF.

Let’s take an example of a relational table <EmployeeDetail> that contains the details of the employees of the company.

3.4.2 Second Normal Form

The normalization of 1NF relations to 2NF involves the elimination of partial dependencies. A partial dependency exists when any non-prime attributes, i.e., an attribute not a part of the candidate key, is not fully functionally dependent on one of the candidate keys.

For a relational table to be in second normal form, it must satisfy the following rules:

The table must be in first normal form.

It must not contain any partial dependency, i.e., all non-prime attributes are fully functionally dependent on the primary key.

If a partial dependency exists, we can divide the table to remove the partially dependent attributes and move them to some other table where they fit in well.

3.4.3 Third Normal Form

The normalization of 2NF relations to 3NF involves the elimination of transitive dependencies.

A functional dependency X -> Z is said to be transitive if the following three functional dependencies hold:

X -> Y

Y does not -> X

Y -> Z

For a relational table to be in third normal form, it must satisfy the following rules:

The table must be in the second normal form.

No non-prime attribute is transitively dependent on the primary key.

For each functional dependency X -> Z at least one of the following conditions hold:

X is a super key of the table.

Z is a prime attribute of the table.

If a transitive dependency exists, we can divide the table to remove the transitively dependent attributes and place them to a new table along with a copy of the determinant.

3.4.4 Boyce Codd Normal Form

Boyce-Codd Normal Form is an advanced version of 3NF as it contains additional constraints compared to 3NF.

For a relational table to be in Boyce-Codd normal form, it must satisfy the following rules:

means X cannot be a non-prime attribute if Y is a prime attribute.

A superkey is a set of one or more attributes that can uniquely identify a row in a database table.

CHAPTER 4

IMPLEMENTATION

4.1 TABLE CREATION

4.1.1 ADMIN

```
CREATE TABLE `admin` (
  `id` int(11) NOT NULL,
  `username` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,
  `img` varchar(20) NOT NULL
)
```

4.1.2 USER

```
CREATE TABLE `user` (
  `id` int(11) NOT NULL,
  `name` char(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `password` varchar(100) NOT NULL,
  `mobile` bigint(20) NOT NULL,
  `address` varchar(255) NOT NULL,
  `gender` enum('male','female','other') NOT NULL,
  `country` varchar(50) NOT NULL,
  `pictrure` varchar(255) NOT NULL
)
```

4.1.3 ROOMS

```
CREATE TABLE `rooms` (
  `room_id` int(11) NOT NULL,
  `room_no` int(11) NOT NULL,
  `type` varchar(100) NOT NULL,
  `price` bigint(20) NOT NULL,
  `details` text NOT NULL,
  `image` varchar(255) NOT NULL
)
```

4.1.4 ROOM_BOOKING_DETAILS

```
CREATE TABLE `room_booking_details` (
  `id` int(11) NOT NULL,
  `name` char(50) NOT NULL,
  `email` varchar(50) NOT NULL,
  `phone` bigint(20) NOT NULL,
  `address` varchar(255) NOT NULL,
  `city` varchar(100) NOT NULL,
  `state` varchar(100) NOT NULL,
  `zip` int(20) NOT NULL,
  `contry` varchar(50) NOT NULL,
  `room_type` varchar(100) NOT NULL,
  `check_in_date` date NOT NULL,
  `check_in_time` varchar(6) NOT NULL,
  `check_out_date` date NOT NULL,
  `Occupancy` varchar(100) NOT NULL
)
```

4.1.5 PAYMENT

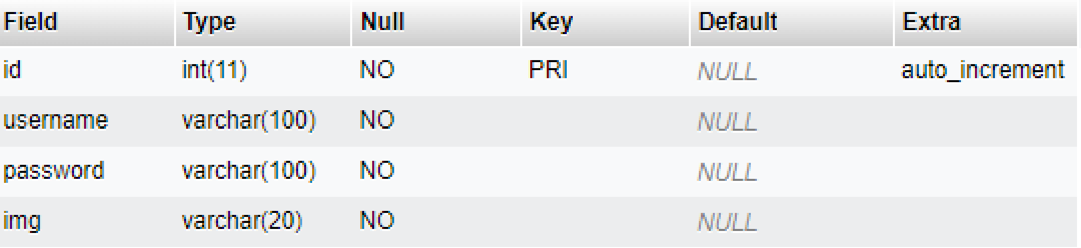
```
CREATE TABLE `payment` (
  `id` int(11) NOT NULL,
  `name` varchar(20) NOT NULL,
  `email` varchar(20) NOT NULL,
  `amount` int(11) NOT NULL,
  `status` varchar(10) NOT NULL
)
```

4.1.6 FEEDBACK

```
CREATE TABLE `feedback` (
  `id` int(11) NOT NULL,
  `name` varchar(100) NOT NULL,
  `email` varchar(100) NOT NULL,
  `mobile` bigint(20) NOT NULL,
  `message` varchar(255) NOT NULL
)
```

4.2 DESCRIPTION OF TABLES

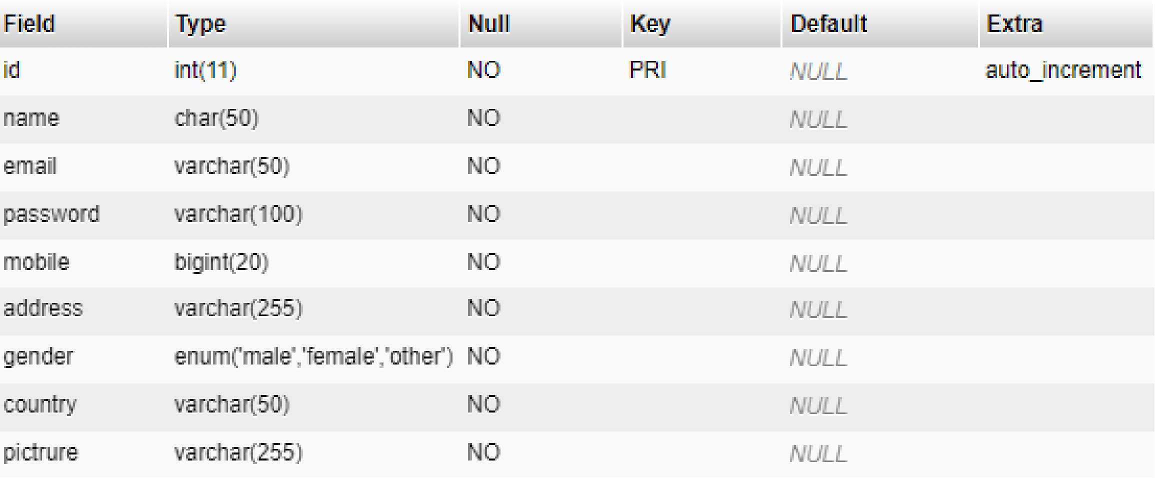
Desc admin;



Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	*NULL*	auto_increment
username	varchar(100)	NO		*NULL*	
password	varchar(100)	NO		*NULL*	
img	varchar(20)	NO		*NULL*	

Figure 4. 1 admin table description

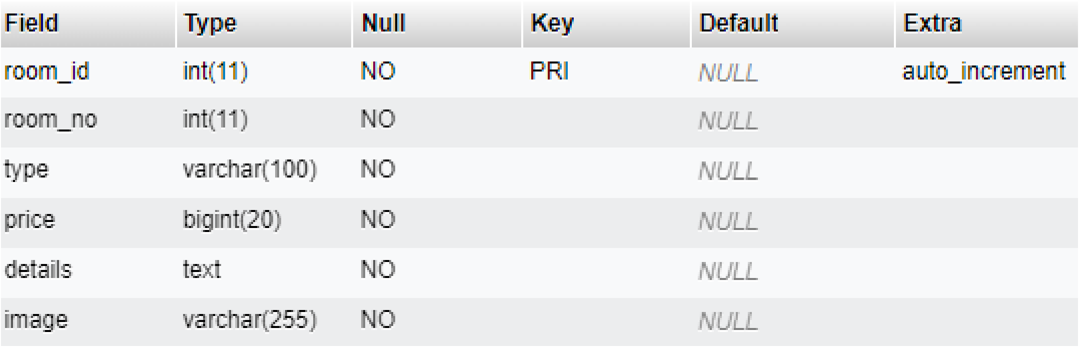
Desc user;



Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	*NULL*	auto_increment
name	char(50)	NO		*NULL*	
email	varchar(50)	NO		*NULL*	
password	varchar(100)	NO		*NULL*	
mobile	bigint(20)	NO		*NULL*	
address	varchar(255)	NO		*NULL*	
gender	enum('male','female','other')	NO		*NULL*	
country	varchar(50)	NO		*NULL*	
pictrure	varchar(255)	NO		*NULL*	

Figure 4. 2 user table description

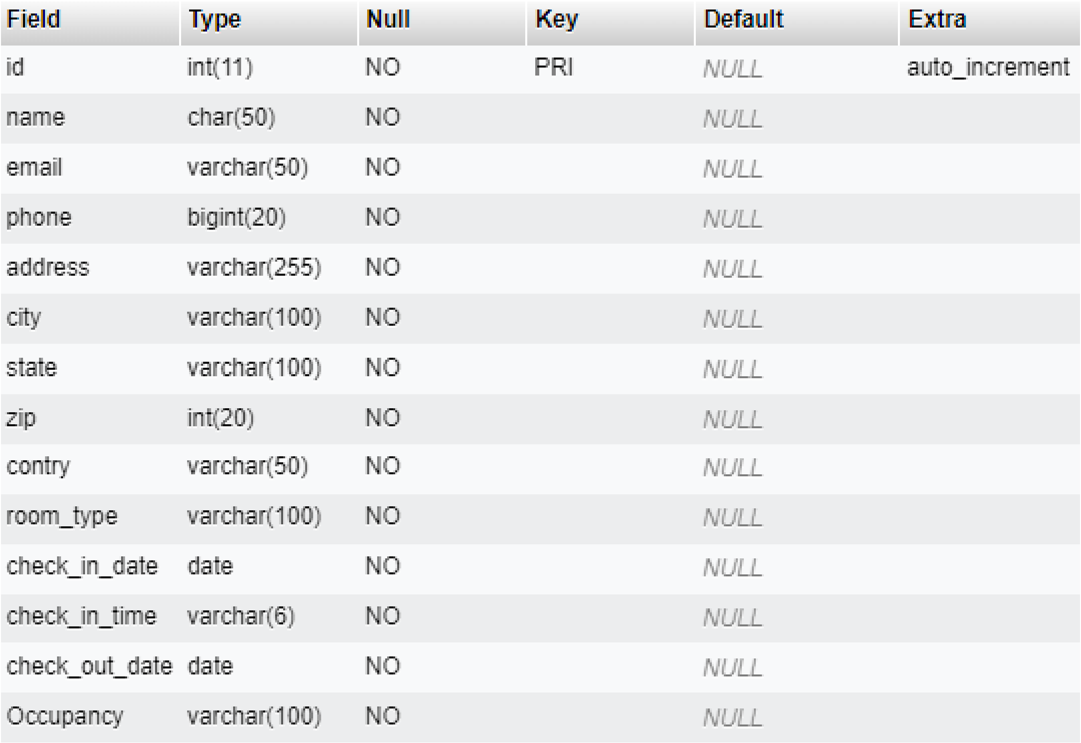
Desc rooms;



Field	Type	Null	Key	Default	Extra
room_id	int(11)	NO	PRI	NULL	auto_increment
room_no	int(11)	NO		NULL	
type	varchar(100)	NO		NULL	
price	bigint(20)	NO		NULL	
details	text	NO		NULL	
image	varchar(255)	NO		NULL	

Figure 4. 3 rooms table description

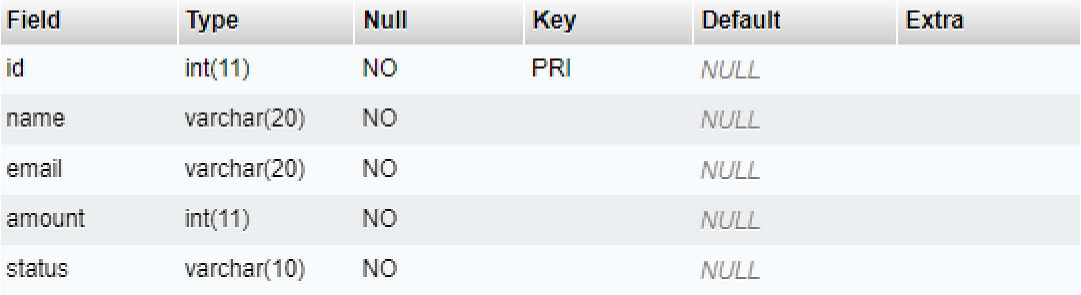
Desc room_booking_details;



Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	char(50)	NO		NULL	
email	varchar(50)	NO		NULL	
phone	bigint(20)	NO		NULL	
address	varchar(255)	NO		NULL	
city	varchar(100)	NO		NULL	
state	varchar(100)	NO		NULL	
zip	int(20)	NO		NULL	
contry	varchar(50)	NO		NULL	
room_type	varchar(100)	NO		NULL	
check_in_date	date	NO		NULL	
check_in_time	varchar(6)	NO		NULL	
check_out_date	date	NO		NULL	
Occupancy	varchar(100)	NO		NULL	

Figure 4. 4 room_booking_details table description

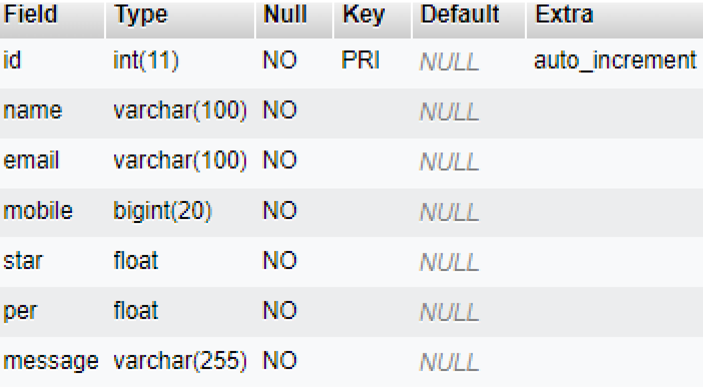
Desc payment;



Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	
name	varchar(20)	NO		NULL	
email	varchar(20)	NO		NULL	
amount	int(11)	NO		NULL	
status	varchar(10)	NO		NULL	

Figure 4. 5 payment table description

Desc feedback;

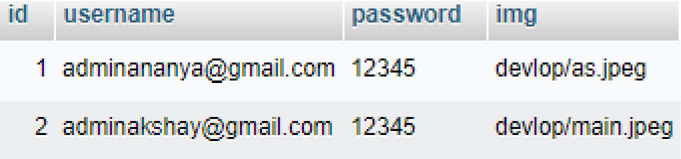


Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
name	varchar(100)	NO		NULL	
email	varchar(100)	NO		NULL	
mobile	bigint(20)	NO		NULL	
star	float	NO		NULL	
per	float	NO		NULL	
message	varchar(255)	NO		NULL	

Figure 4. 6 feedback table description

4.3 POPULATED TABLES

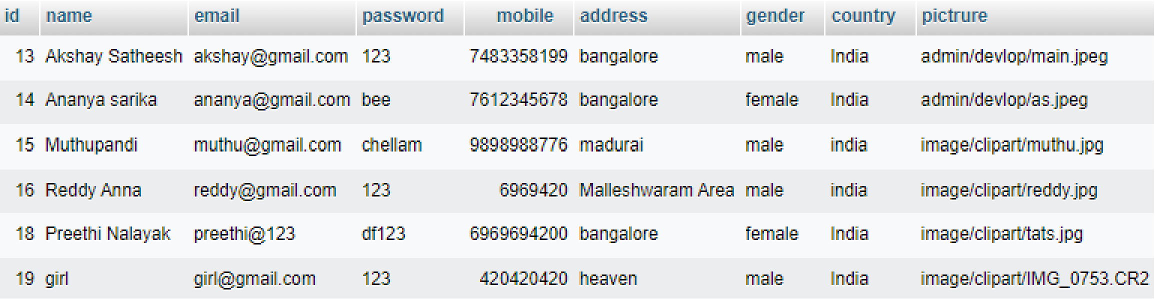
Select * from admin;



id	username	password	img
1	adminananya@gmail.com	12345	devlop/as.jpeg
2	adminakshay@gmail.com	12345	devlop/main.jpeg

Figure 4. 7 admin table values

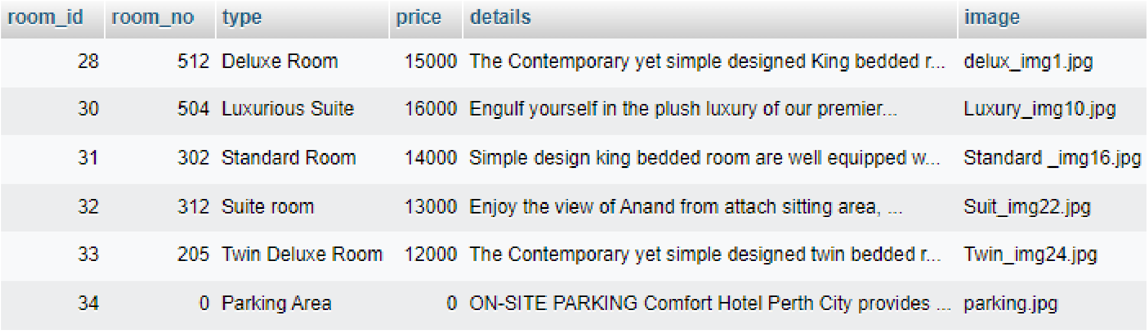
Select * from user;



id	name	email	password	mobile	address	gender	country	pictrure
13	Akshay Satheesh	akshay@gmail.com	123	7483358199	bangalore	male	India	admin/devlop/main.jpeg
14	Ananya sarika	ananya@gmail.com	bee	7612345678	bangalore	female	India	admin/devlop/as.jpeg
15	Muthupandi	muthu@gmail.com	chellam	9898988776	madurai	male	india	image/clipart/muthu.jpg
16	Reddy Anna	reddy@gmail.com	123	6969420	Malleshwaram Area	male	india	image/clipart/reddy.jpg
18	Preethi Nalayak	preethi@123	df123	6969694200	bangalore	female	India	image/clipart/tats.jpg
19	girl	girl@gmail.com	123	420420420	heaven	male	India	image/clipart/IMG_0753.CR2

Figure 4. 8 user table values

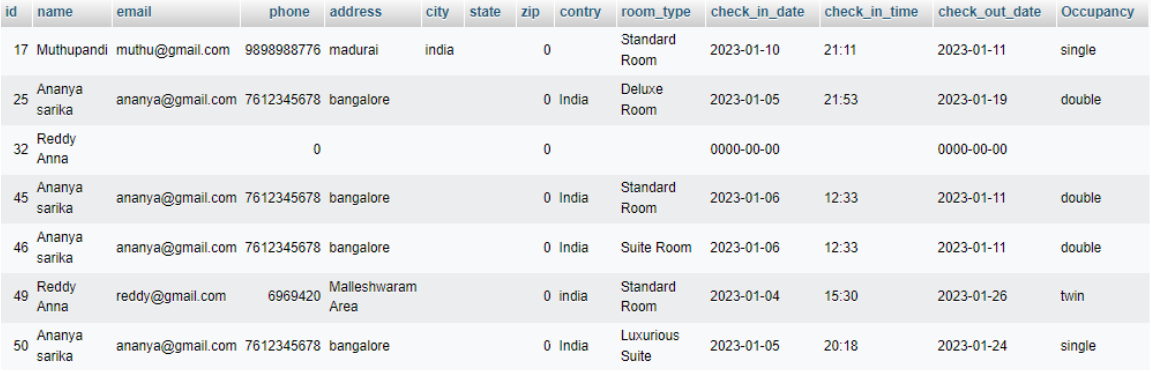
Select* from rooms;



room_id	room_no	type	price	details	image
28	512	Deluxe Room	15000	The Contemporary yet simple designed King bedded r...	delux_img1.jpg
30	504	Luxurious Suite	16000	Engulf yourself in the plush luxury of our premier...	Luxury_img10.jpg
31	302	Standard Room	14000	Simple design king bedded room are well equipped w...	Standard _img16.jpg
32	312	Suite room	13000	Enjoy the view of Anand from attach sitting area, ...	Suit_img22.jpg
33	205	Twin Deluxe Room	12000	The Contemporary yet simple designed twin bedded r...	Twin_img24.jpg
34	0	Parking Area	0	ON-SITE PARKING Comfort Hotel Perth City provides ...	parking.jpg

Figure 4. 9 rooms table values

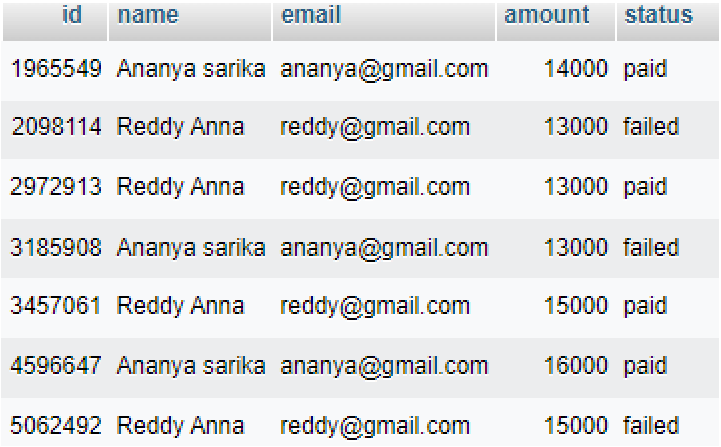
Select * from room_booking_details;



id	name	email	phone	address	city	state	zip	contry	room_type	check_in_date	check_in_time	check_out_date	Occupancy
17	Muthupandi	muthu@gmail.com	9898988776	madurai	india		0		Standard Room	2023-01-10	21:11	2023-01-11	single
25	Ananya sarika	ananya@gmail.com	7612345678	bangalore			0	India	Deluxe Room	2023-01-05	21:53	2023-01-19	double
32	Reddy Anna		0				0			0000-00-00		0000-00-00	
45	Ananya sarika	ananya@gmail.com	7612345678	bangalore			0	India	Standard Room	2023-01-06	12:33	2023-01-11	double
46	Ananya sarika	ananya@gmail.com	7612345678	bangalore			0	India	Suite Room	2023-01-06	12:33	2023-01-11	double
49	Reddy Anna	reddy@gmail.com	6969420	Malleshwaram Area			0	india	Standard Room	2023-01-04	15:30	2023-01-26	twin
50	Ananya sarika	ananya@gmail.com	7612345678	bangalore			0	India	Luxurious Suite	2023-01-05	20:18	2023-01-24	single

Figure 4. 10 room_booking_details table values

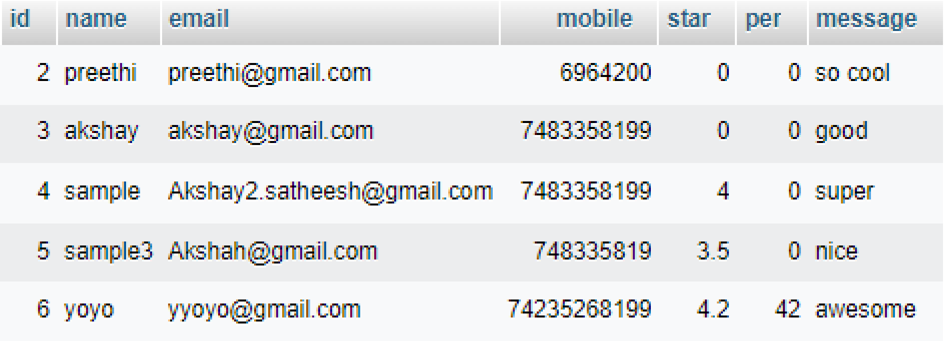
Select * from payment;



id	name	email	amount	status
1965549	Ananya sarika	ananya@gmail.com	14000	paid
2098114	Reddy Anna	reddy@gmail.com	13000	failed
2972913	Reddy Anna	reddy@gmail.com	13000	paid
3185908	Ananya sarika	ananya@gmail.com	13000	failed
3457061	Reddy Anna	reddy@gmail.com	15000	paid
4596647	Ananya sarika	ananya@gmail.com	16000	paid
5062492	Reddy Anna	reddy@gmail.com	15000	failed

Figure 4. 11 payment table values

Select * from feedback;



id	name	email	mobile	star	per	message
2	preethi	preethi@gmail.com	6964200	0	0	so cool
3	akshay	akshay@gmail.com	7483358199	0	0	good
4	sample	Akshay2.satheesh@gmail.com	7483358199	4	0	super
5	sample3	Akshah@gmail.com	748335819	3.5	0	nice
6	yoyo	yyoyo@gmail.com	74235268199	4.2	42	awesome

Figure 4. 12 feedback table values

4.4 SQL TRIGGERS AND STORED PROCEDURES

4.4.1 TRIGGER

A trigger is a stored procedure in database which automatically invokes whenever a special event in the database occurs. For example, a trigger can be invoked when a row is inserted into a specified table or when certain table columns are being updated.

Syntax:

create trigger [trigger_name]

[before | after]

{insert | update | delete}

on [table_name]

[for each row]

[trigger_body]

TRIGGERS USED IN THIS PROJECT

```
CREATE TRIGGER `rat_per` BEFORE INSERT ON `feedback`
 FOR EACH ROW SET NEW.per=(NEW.stars+NEW.star+NEW.starF)*100/15
```

4.4.2 STORED PROCEDURE

A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again.

So if you have an SQL query that you write over and over again, save it as a stored procedure, and then just call it to execute it.

You can also pass parameters to a stored procedure, so that the stored procedure can act based on the parameter value(s) that is passed.

Stored Procedure Syntax

CREATE PROCEDURE procedure_name

AS

sql_statement

GO;

Execute a Stored Procedure

EXEC procedure_name;

4.5 DATABASE CONNECTIVITY

4.5.1 PHP

PHP mysqli_connect()

PHP **mysqli_connect() function** is used to connect with MySQL database. It returns *resource* if connection is established or *null*.

Syntax

resource mysqli_connect (server, username, password)

PHP mysqli_close()

PHP **mysqli_close() function** is used to disconnect with MySQL database. It returns *true* if connection is closed or *false*.

Syntax

bool mysqli_close(resource $resource_link)

4.5.1 JDBC

JDBC ('Java Database Connectivity') allows multiple implementations to exist and be used by the same application. The API provides a mechanism for dynamically loading the correct Java packages and registering them with the JDBC Driver Manager. The Driver Manager is used as a connection factory for creating JDBC connections.

JDBC connections support creating and executing statements. These may be update statements such as SQL's CREATE, INSERT, UPDATE and DELETE, or they may be query statements such as SELECT. Additionally, stored procedures may be invoked through a JDBC connection. JDBC represents statements using one of the following classes:

Statement – the statement is sent to the database server each and every time.

PreparedStatement – the statement is cached and then the execution path is pre-determined on the database server allowing it to be executed multiple times in an efficient manner.

CallableStatement – used for executing stored procedures on the database.

Update statements such as INSERT, UPDATE and DELETE return an update count that indicates how many rows were affected in the database. These statements do not return any other information.

Query statements return a JDBC row result set. The row result set is used to walk over the result set. Individual columns in a row are retrieved either by name or by column number. There may be any number of rows in the result set. The row result set has metadata that describes the names of the columns and their types.

There is an extension to the basic JDBC API in the javax.sql.

JDBC connections are often managed via a connection pool rather than obtained directly from the driver.

4.5.3 MySQL

MySQL is a Relational Database Management System (RDBMS). MySQL server can manage many databases at the same time. In fact, many people might have different databases managed by a single MySQL server. Each database consists of a structure to hold onto the data itself. A data-base can exist without data, only a structure, be totally empty, twiddling its thumbs and waiting for data to be stored in it.

Data in a database is stored in one or more tables. You must create the data-base and the tables before you can add any data to the database. First you create the empty database. Then you add empty tables to the database. Database tables are organized in rows and columns. Each row represents an entity in the database, such as a customer, a book, or a project. Each column contains an item of information about the entity, such as a customer name, a book name, or a project start date. The place where a particular row and column intersect, the individual cell of the table, is called a field. Tables in databases can be related. Often a row in one table is related to several rows in another table. For instance, you might have a database containing data about books you own. You would have a book table and an author table. One row in the author table might contain information about the author of several books in the book table. When tables are related, you include a column in one table to hold data that matches data in the column of another table.

MySQL, the most popular Open Source SQL database management system, is developed, distributed, and supported by MySQL AB. MySQL AB is a commercial company, founded by the MySQL developers. It is a second generation Open Source company that unites Open Source values and methodology with a successful business model.

- MySQL is a relational database management system. A relational database stores data in separate tables rather than putting all the data in one big storeroom. This adds speed and flexibility. The SQL part of "MySQL" stands for "Structured Query Language." SQL is the most common standardized language used to access databases and is defined by the ANSI/ISO SQL Standard. The SQL standard has been evolving since 1986 and several versions exist. "SQL-92" refers to the standard released in 1992, "SQL:1999" refers to the standard released in 1999, and "SQL:2003" refers to the current version of the standard.

We use the phrase "the SQLstandard" to refer to the current version of the SQL Standard. MySQL software is Open Source. Open Source means that it is possible for anyone to use and modify the software. Anybody can download the MySQL software from the Internet and use it without paying anything. If you wish, you may study the source code and change it to suit your needs. The MySQL software uses the GPL (GNU General Public License), to define what you may and may not do with the software in different situations. The MySQL Database Server is very fast, reliable, and easy to use

4.6 SOURCE CODE(FRONT END)

4.6.1 Index.php

```
<?php
session_start();
error_reporting(1);
include('connection.php');
?>
<!DOCTYPE html>
<html lang="en">
<head><!--Head Open  Here-->
  <title>Alpha Royal Hotels</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>
  <link href="https://fonts.googleapis.com/css?family=Abril+Fatface" rel="stylesheet">
  <link rel="preconnect" href="https://fonts.googleapis.com">
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link
href="https://fonts.googleapis.com/css2?family=Poppins:wght@100;200;300;400;600&d
isplay=swap" rel="stylesheet">
  <link href="css/style.css"rel="stylesheet"/>
</head> <!--Head Open  Here-->
<body style="margin-top:50px;">
  <?php
      include('Menu Bar.php')
  ?>
```

```
<div id="myCarousel" class="carousel slide" data-ride="carousel"> <!--Slider Image Start
Here-->
   <!-- Indicators -->

   <ol class="carousel-indicators">
     <li data-target="#myCarousel" data-slide-to="0" class="active"></li>
     <li data-target="#myCarousel" data-slide-to="1"></li>
    <li data-target="#myCarousel" data-slide-to="2"></li>
   </ol>
   <!--Indicators Close Here-->

   <!-- Wrapper for slides -->

   <div class="carousel-inner" role="listbox">
     <?php
   $i=1;
   $sql=mysqli_query($con,"select * from slider");
   while($slider=mysqli_fetch_assoc($sql))
   {
   $slider_img=$slider['image'];
   $slider_cap=$slider['caption'];
   $path="image/Slider/$slider_img";
     if($i==1)
     {
   ?>
   <div class="item active">
       <img src="<?php echo $path; ?>" alt="Image">
       <div class="carousel-caption">
     <h2><?php echo $slider_cap; ?></h2>
   </div>
     </div>
   <?php
   }
   else
     {
     ?>
   <div class="item">
       <img src="<?php echo $path; ?>" alt="Image">
       <div class="carousel-caption">
       <h2><?php echo $slider_cap; ?></h2>
   </div>
     </div>

   <?php } ?>



   <?php $i++; } ?>




   </div>




   <a class="left carousel-control" href="#myCarousel" role="button" data-slide="prev">
     <span class="glyphicon glyphicon-chevron-left" aria-hidden="true"></span>
     <span class="sr-only">Previous</span>
   </a>
   <a class="right carousel-control" href="#myCarousel" role="button" data-slide="next">
     <span class="glyphicon glyphicon-chevron-right" aria-hidden="true"></span>
     <span class="sr-only">Next</span>
   </a>
   <!-- Left and right controls Close Here -->

</div> <!--Room Info Start Here-->

 <div class="container-fluid"id="red"><!--Id Is Red-->
<div class="container text-center">
```

```
  <h1 style=" font-family: 'Dancing Script', cursive;color:white  ; font-weight:bold;font-size:50px; padding:15px; position: relative; bottom: 390px; filter:drop-shadow(10px 2px 4px rgb(129, 94, 5)); letter-spacing:3px;">Welcome To Alpha Royal Hotel<br><br><p style="font-size:35px;">project by Akshay Satheesh</p></h1><br>
  <div class="row">
    <div class="hov"><!--Hov is Class-->



  <?php
  $sql=mysqli_query($con,"select * from rooms");
  while($r_res=mysqli_fetch_assoc($sql))
  {
  ?>
  <div class="col-sm-4" style="height: 500px;">
                  <img style="width:300px;height:200px"src="image/rooms/<?php echo $r_res['image']; ?>"class="img-responsive thumbnail"alt="Image"id="img1"> <!--Id Is Img-->
      <h4 class="Room_Text"> <?php echo $r_res['type']; ?></h4>
      <p class="text-justify"><?php echo substr($r_res['details'],0,100); ?></p><br>
       <a href="room_details.php?room_id=<?php echo $r_res['room_id']; ?>" class="btn btn-danger text-center">Read more</a><br><br>
    </div>
  <?php } ?>
  </div>
  </div>
</div>
</div>

<?php
  include('Footer.php')
?>
</body>
</html>
```

4.6.2 Menu bar.php

```
<?php
session_start();
$eid=$_SESSION['user_logged_in'];
error_reporting(1);
?>
<!--Menu Bar Close Here-->
<nav class="navbar navbar-inverse navbar-fixed-top">
  <div class="container">
    <div class="navbar-header">
           <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#myNavbar">
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>

    </div>
        <div class="collapse navbar-collapse" id="myNavbar" style="position:relative; top:30px; font-size: 18px;">
      <ul class="nav navbar-nav">
        <li><a href="index.php"title="Home">Home</a></li>
        <li><a href="about.php"title="About">About </a></li>
        <li><a href="image gallery.php"title="Gallery">Gallery </a></li>
      </ul>
            <img src="logo/logo2.png" width="100px"style="position:relative;left:25%; bottom:30px;margin-top:5px;margin-bottom:5px;">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
      <ul class="nav navbar-nav navbar-right">
```

```
            <li><a href="admin/index.php"title="Admin Login"><span class="glyphicon glyphicon-user"></span>&nbsp;&nbsp;Admin Login</a></li>

       <?php
     if($_SESSION['user_logged_in']!="")
     {
       ?>
            <li class="dropdown"><a class="dropdown-toggle" data-toggle="dropdown" href="#">View Status <span class="caret"></span></a>
         <ul class="dropdown-menu">
             <li><a href="profile.php">Profile</a></li>
             <li><a href="order.php">Booking Status</a></li>
             <li><a href="logout.php">Logout</a></li>
         </ul>
       </li>
       <?PHP } else
   {
   ?>
         <li><a href="Login.php"title="login"><span class="glyphicon glyphicon-log-in"></span>&nbsp;&nbsp;User Login</a>
       </li>
   <?php
   } ?>
     </ul>
   </div>
  </div>
</nav>

<!--Menu Bar Close Here-->
```

4.6.3. Login.php

```
<?php
session_start();
error_reporting(1);
if($_SESSION['user_logged_in']!="")
{
header('location:Booking Form.php');
}
error_reporting(1);
require('connection.php');
extract($_REQUEST);
if(isset($login))
{
  if($eid=="" || $pass=="")
  {
  $error= "<h4 style='color:red'>fill all details</h4>";
  }
  else
  {
  $sql=mysqli_query($con,"select * from user where email='$eid' && password='$pass' ");
    if(mysqli_num_rows($sql))
    {
    $_SESSION['user_logged_in']=$eid;
    header('location:Booking Form.php');
    }
    else
    {
    $error= "<h4 style='color:red'>Invalid login details</h4>";
    }
  }
}
?>
<!DOCTYPE html>
```

```
<html lang="en">
<head>
  <title>Alpha Royal Hotel</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
                                        <link                                       rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
                                                                                                       <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>
  <link href="css/style.css"rel="stylesheet"/>
        <link        rel="stylesheet"        href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.min.css">
     <link     href="https://fonts.googleapis.com/css?family=Akronim|Libre+Baskerville"
rel="stylesheet">
</head>
<body id="primary"style="margin-top:50px;">
<?php
include('Menu Bar.php')
?>
<div class="container-fluid"><!-- Primary Id-->
  <div class="container" style="">
    <div class="row1"><br>

                    <div      class="col-sm-12      text-center"style="width:450px;height:fit-
content;background-color:white;box-shadow:2px 2px 2px;"><br>

        <h1 style="align:center; font-family: Verdana, Geneva, Tahoma, sans-serif; color:
#212221; position:relative; top:-20px;"><b>User Login </b></h1>
             <img  src="image/clipart/login-user-icon.png"  class="img-circle"  alt="Bird"
width="150" height="150">
        <?php echo @$error; ?>
        <form method="post"><br>
```

```
            <div class="form-group">
                <input type="Email" class="form-control"name="eid"placeholder="Email Id" autocomplete="off"required >
            </div>
          <div class="form-group">
                <input type="Password" class="form-control"name="pass"placeholder="Password" autocomplete="off"required>
          </div>
          <input type="submit" value="Login" name="login" class="btn btn-primary btn-group btn-group-justified"required>
        <div class="form-group forget">
             <a href="Forgot account.php">Forgot Password?</a>&nbsp; <b>|</b>&nbsp;
             <a href="Registation form.php">Create an Account</a>
          </div>
      </form><br>
      </div>
   </div><br>
  </div>
</div>
<?php

?>
</body>
</html>
```

4.6.4. Profile.php

```
<?php
session_start();
error_reporting(1);
include('connection.php');
$eid=$_SESSION['user_logged_in'];
extract($_REQUEST);
```

```
if(isset($update))
{
$que="update user set name='$name',password='$pass',mobile='$mob',address='$add' where email='$eid'";
mysqli_query($con,$que);
$msg= "<h3 style='color:blue'>Profile Updated successfully</h3>";
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Alpha Royal Hotels</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.4.0/js/bootstrap.min.js"></script>
  <link href="css/style.css"rel="stylesheet"/>
 <link href="https://fonts.googleapis.com/css?family=Lobster" rel="stylesheet">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
  <link href="https://fonts.googleapis.com/css?family=Baloo+Bhai" rel="stylesheet">
</head>
<body style="margin-top:50px;">
  <?php
  include('Menu Bar.php');
  ?>
 <?php
     $sql= mysqli_query($con,"select * from user where email='$eid' ");
     $result=mysqli_fetch_assoc($sql);
?>
```

```
<div class="container-fluid"id="primary" style="margin-top: 100px; padding-bottom: 70px;"><!--Primary Id-->

  <div class="container">
    <div class="row" style=" scale:90%;display:flex; background-color: white;">

    <div style=" margin-left:100px; width: 700px;">
         <center><h1 style="margin-left:250px;border-radius:50px;font-family: Verdana, Geneva, Tahoma, sans-serif ;display:inline-block; color: #212221;">User Profile</h1></center><br>
  <center><?php  echo $msg; ?></center>
     <form class="form-horizontal" method="post">
       <div class="col-sm-6">
          <div class="form-group">
           <div class="row">
              <div class="control-label col-sm-4"><h4> Name :</h4></div>
                <div class="col-sm-8">
                          <input type="text" name="name" value="<?php echo $result['name']; ?>"class="form-control"/>
          </div>
        </div>
      </div>


      <div class="form-group">
           <div class="row">
              <div class="control-label col-sm-4"><h4>Email-Id:</h4></div>
                <div class="col-sm-8">
                        <input type="text" value="<?php echo $result['email']; ?>"class="form-control"/readonly="readonly">
          </div>
        </div>
      </div>
      <div class="form-group">
```

```
          <div class="row">
            <div class="control-label col-sm-4"><h4>Password:</h4></div>
              <div class="col-sm-8">
                    <input  type="text"  name="pass"  value="<?php  echo  $result['password'];
?>"class="form-control"/>
         </div>
       </div>
     </div>
     <div class="form-group">
          <div class="row">
            <div class="control-label col-sm-4"><h4>Mobile:</h4></div>
              <div class="col-sm-8">
                      <input  type="text"  name="mob"  value="<?php  echo  $result['mobile'];
?>"class="form-control"/>
         </div>
       </div>
     </div>
     <div class="form-group">
          <div class="row">
            <div class="control-label col-sm-4"><h4>Address:</h4></div>
              <div class="col-sm-8">
                      <input  type="text"  name="add"  value="<?php  echo  $result['address'];
?>"class="form-control"/>
         </div>
       </div>
     </div>
     <div class="form-group">
          <div class="row">
            <div class="control-label col-sm-4"><h4>Gender:</h4></div>
              <div class="col-sm-8">
               <strong><?php echo $result['gender']; ?></strong>
         </div>
       </div>
```

```
      </div>

      <div class="form-group">
           <div class="row">
              <div class="control-label col-sm-5"></div>
                <div class="col-sm-7  ">
                   <input type="submit" value="Update Profile" name="update" class="btn btn-primary"/>
          </div>
        </div>
      </div>

    </div>
<!--User Profile Update Query-->
        </form>
    </div>
       <div> <img style="position:relative;top:100px;right:200px; width: 350px;height: 350px; border-radius: 50%; border: 7px solid #212221;" src="<?php echo $result['pictrure']; ?>"> </div>
      </div>
   </div>
 </div>
<?php

?>
</body>
</html>
```

CHAPTER 5

RESULTS

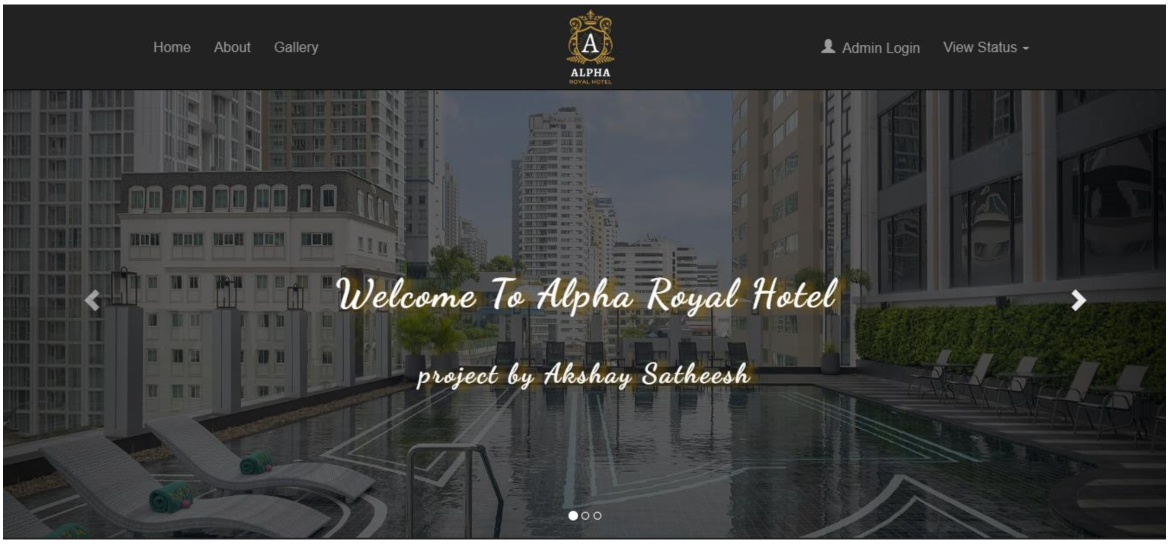
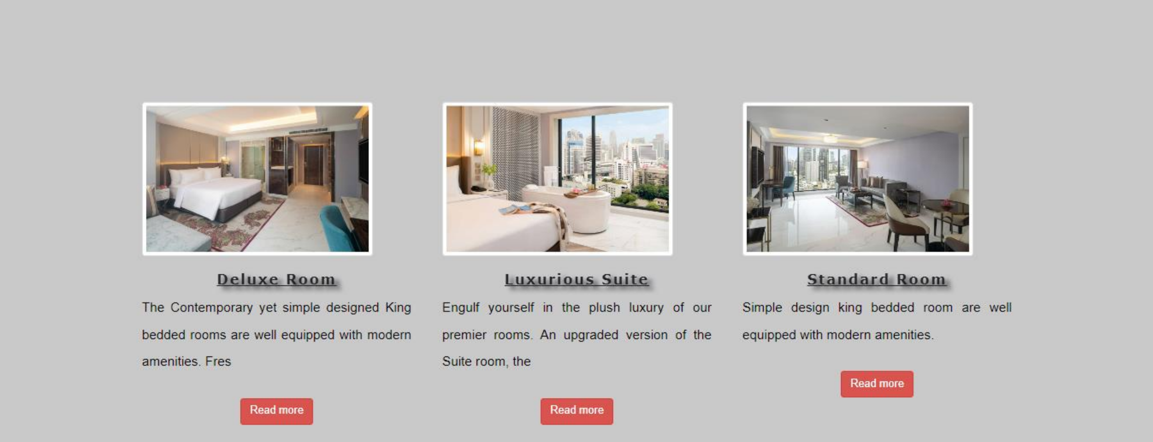


Figure 5. 1 index page(header)



Deluxe Room

The Contemporary yet simple designed King bedded rooms are well equipped with modern amenities. Fres

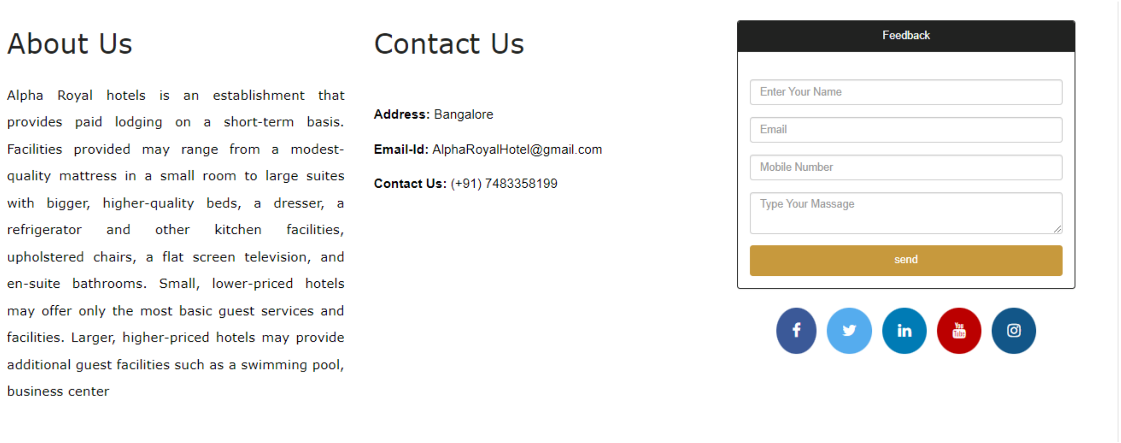
Luxurious Suite

Engulf yourself in the plush luxury of our premier rooms. An upgraded version of the Suite room, the

Standard Room

Simple design king bedded room are well equipped with modern amenities.

Figure 5. 2 index page(2)



About Us

Alpha Royal hotels is an establishment that provides paid lodging on a short-term basis. Facilities provided may range from a modest-quality mattress in a small room to large suites with bigger, higher-quality beds, a dresser, a refrigerator and other kitchen facilities, upholstered chairs, a flat screen television, and en-suite bathrooms. Small, lower-priced hotels may offer only the most basic guest services and facilities. Larger, higher-priced hotels may provide additional guest facilities such as a swimming pool, business center

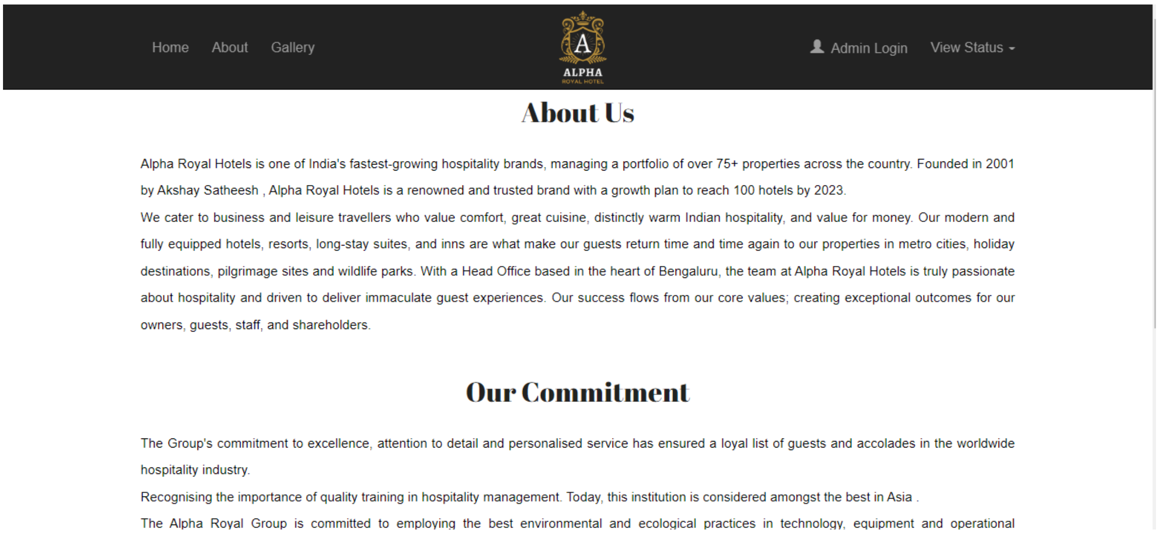
Contact Us

Address: Bangalore

Email-Id: AlphaRoyalHotel@gmail.com

Contact Us: (+91) 7483358199

Figure 5. 3 footer



About Us

Alpha Royal Hotels is one of India's fastest-growing hospitality brands, managing a portfolio of over 75+ properties across the country. Founded in 2001 by Akshay Satheesh , Alpha Royal Hotels is a renowned and trusted brand with a growth plan to reach 100 hotels by 2023.

We cater to business and leisure travellers who value comfort, great cuisine, distinctly warm Indian hospitality, and value for money. Our modern and fully equipped hotels, resorts, long-stay suites, and inns are what make our guests return time and time again to our properties in metro cities, holiday destinations, pilgrimage sites and wildlife parks. With a Head Office based in the heart of Bengaluru, the team at Alpha Royal Hotels is truly passionate about hospitality and driven to deliver immaculate guest experiences. Our success flows from our core values; creating exceptional outcomes for our owners, guests, staff, and shareholders.

Our Commitment

The Group's commitment to excellence, attention to detail and personalised service has ensured a loyal list of guests and accolades in the worldwide hospitality industry.

Recognising the importance of quality training in hospitality management. Today, this institution is considered amongst the best in Asia .

The Alpha Royal Group is committed to employing the best environmental and ecological practices in technology, equipment and operational

Figure 5. 4 about us page

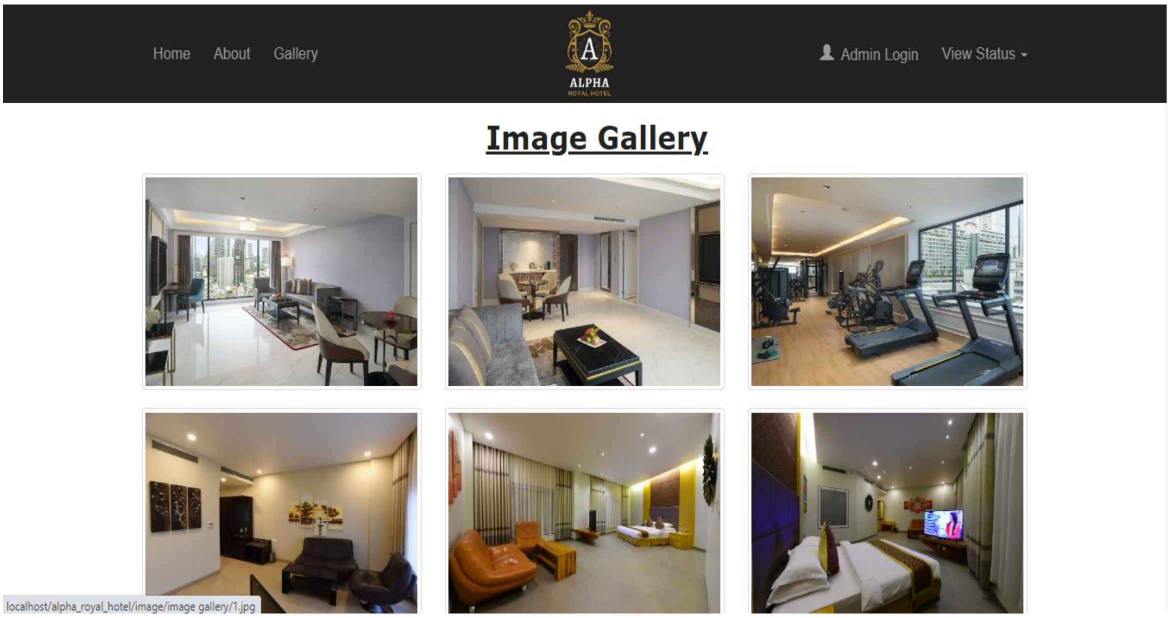


Figure 5. 5 image gallery page

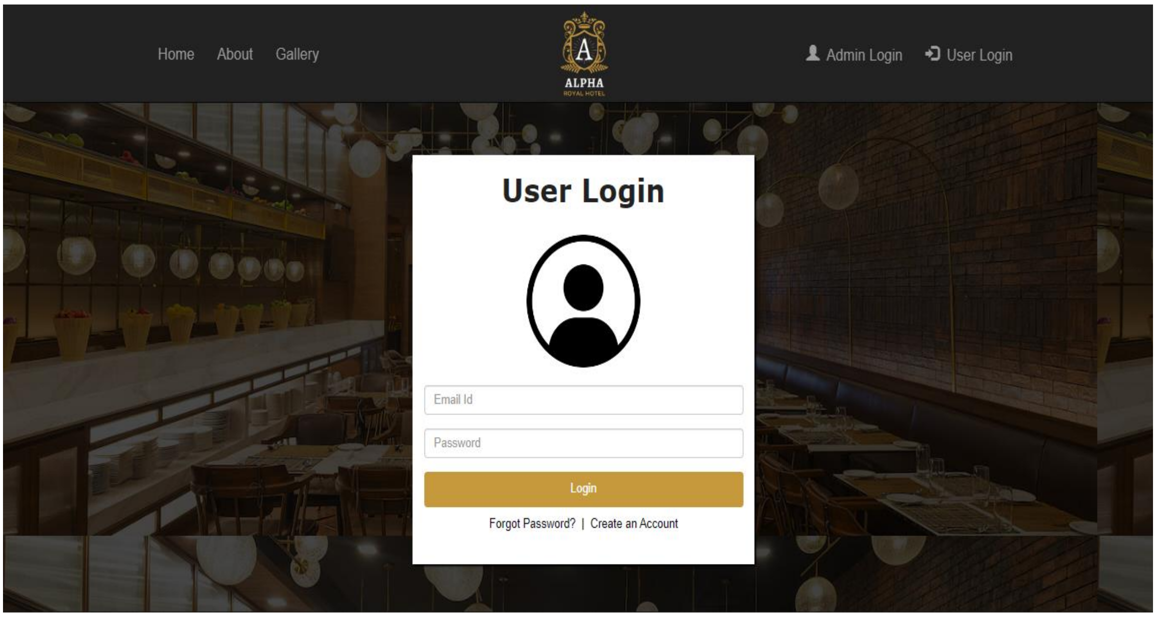


Figure 5. 6 user login page

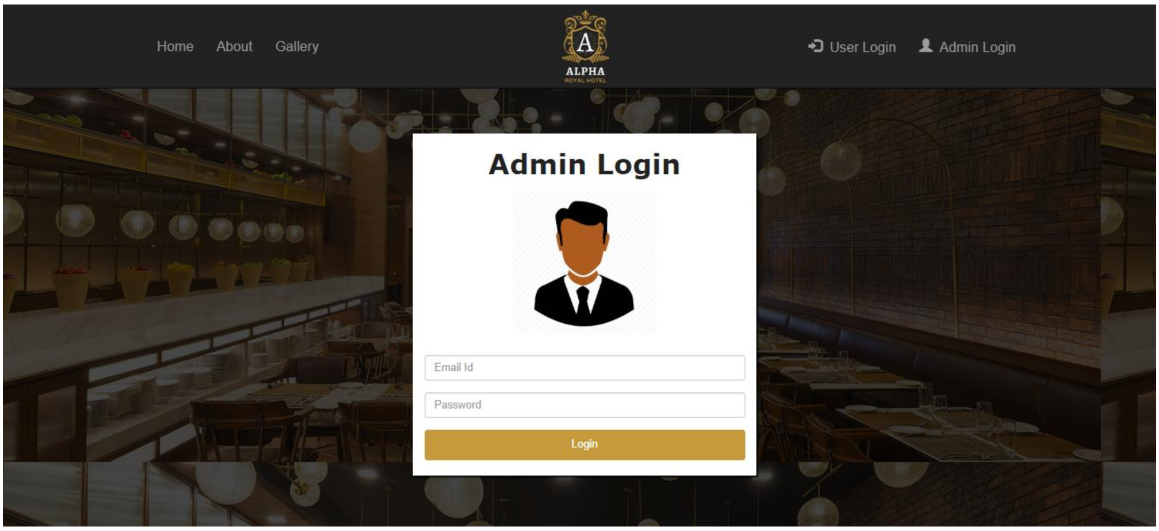
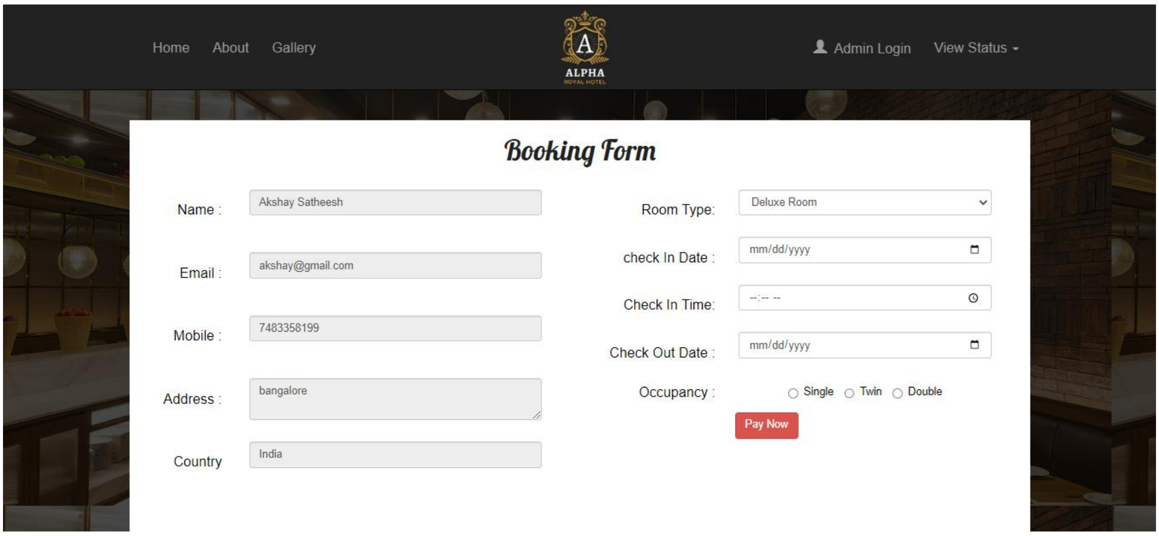
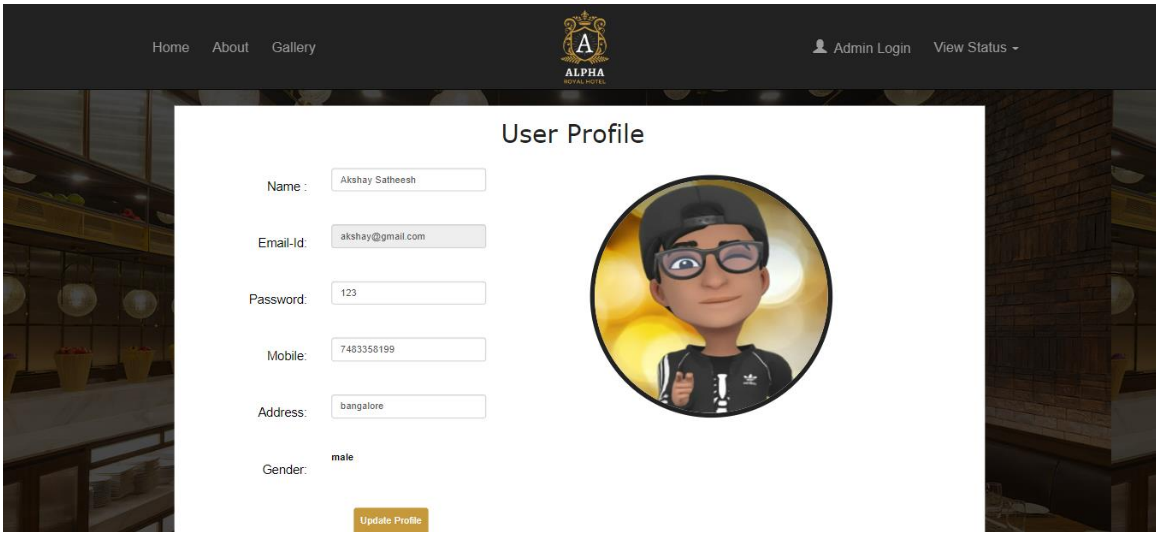


Figure 5. 7 admin login page



Name :	Akshay Satheesh	Room Type:	Deluxe Room
Email :	akshay@gmail.com	check In Date :	mm/dd/yyyy
Mobile :	7483358199	Check In Time:	--:-- --
Address :	bangalore	Check Out Date :	mm/dd/yyyy
Country	India	Occupancy :	○ Single ○ Twin ○ Double

Figure 5. 8 booking form page



User Profile

Name : Akshay Satheesh

Email-Id: akshay@gmail.com

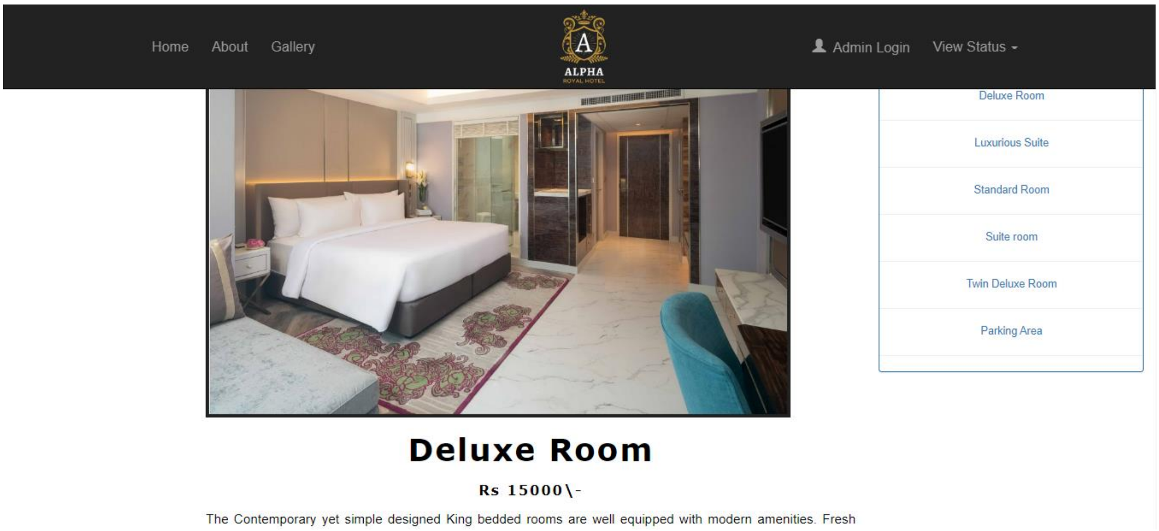
Password: 123

Mobile: 7483358199

Address: bangalore

Gender: male

Figure 5. 9 user profile page

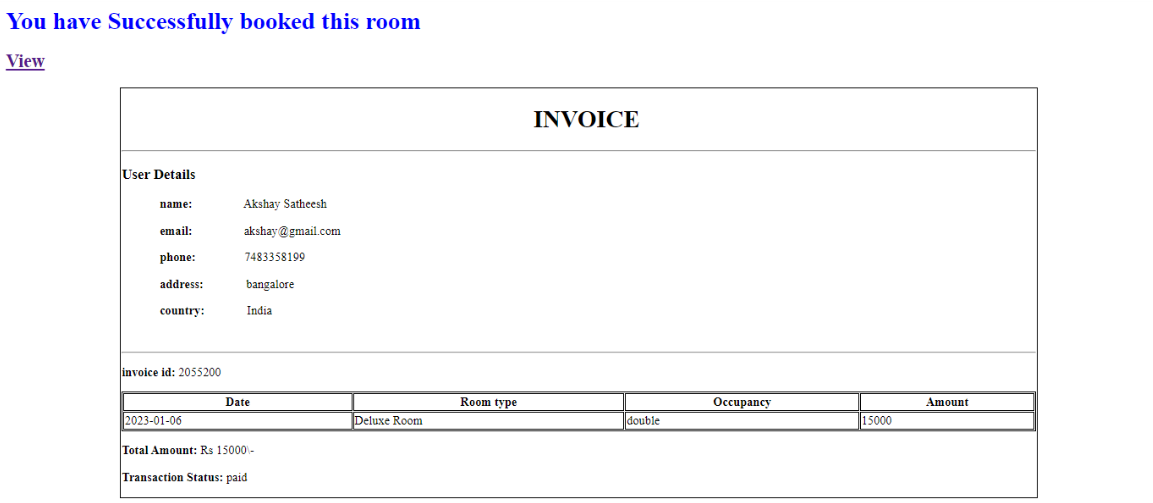


Deluxe Room

Rs 15000\-

The Contemporary yet simple designed King bedded rooms are well equipped with modern amenities. Fresh

Figure 5. 10 rooms page



You have Successfully booked this room

View

INVOICE

User Details

name: Akshay Satheesh

email: akshay@gmail.com

phone: 7483358199

address: bangalore

country: India

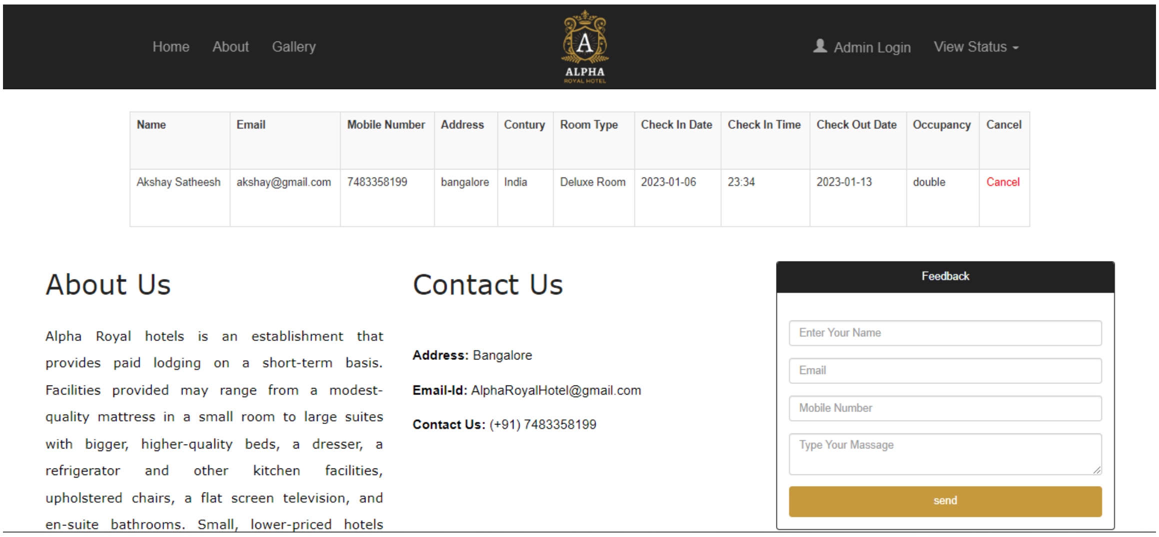
invoice id: 2055200

Date	Room type	Occupancy	Amount
2023-01-06	Deluxe Room	double	15000

Total Amount: Rs 15000\-

Transaction Status: paid

Figure 5. 11 invoice page



Name	Email	Mobile Number	Address	Contury	Room Type	Check In Date	Check In Time	Check Out Date	Occupancy	Cancel
Akshay Satheesh	akshay@gmail.com	7483358199	bangalore	India	Deluxe Room	2023-01-06	23:34	2023-01-13	double	Cancel

About Us

Alpha Royal hotels is an establishment that provides paid lodging on a short-term basis. Facilities provided may range from a modest-quality mattress in a small room to large suites with bigger, higher-quality beds, a dresser, a refrigerator and other kitchen facilities, upholstered chairs, a flat screen television, and en-suite bathrooms. Small, lower-priced hotels

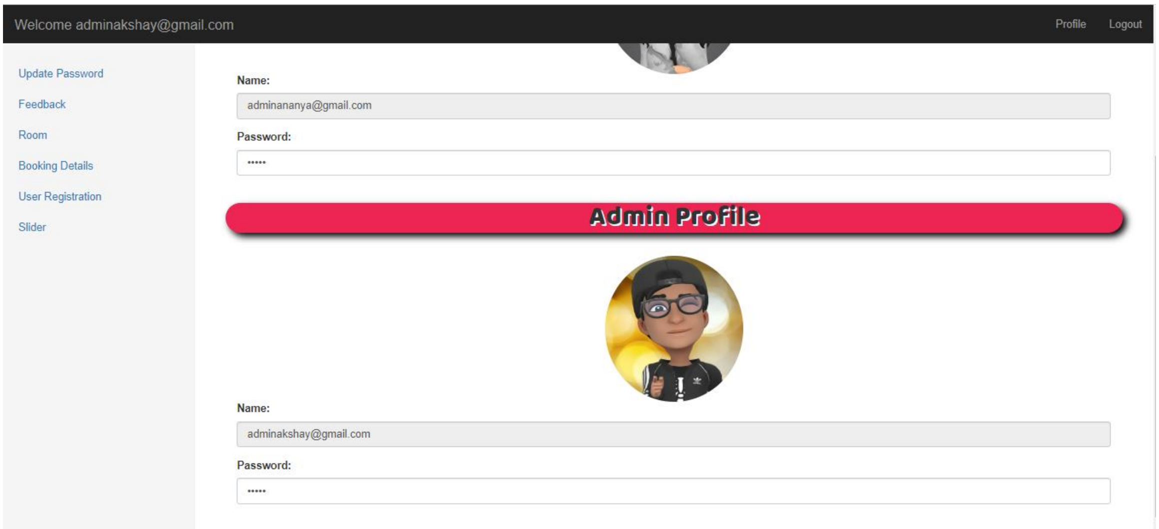
Contact Us

Address: Bangalore

Email-Id: AlphaRoyalHotel@gmail.com

Contact Us: (+91) 7483358199

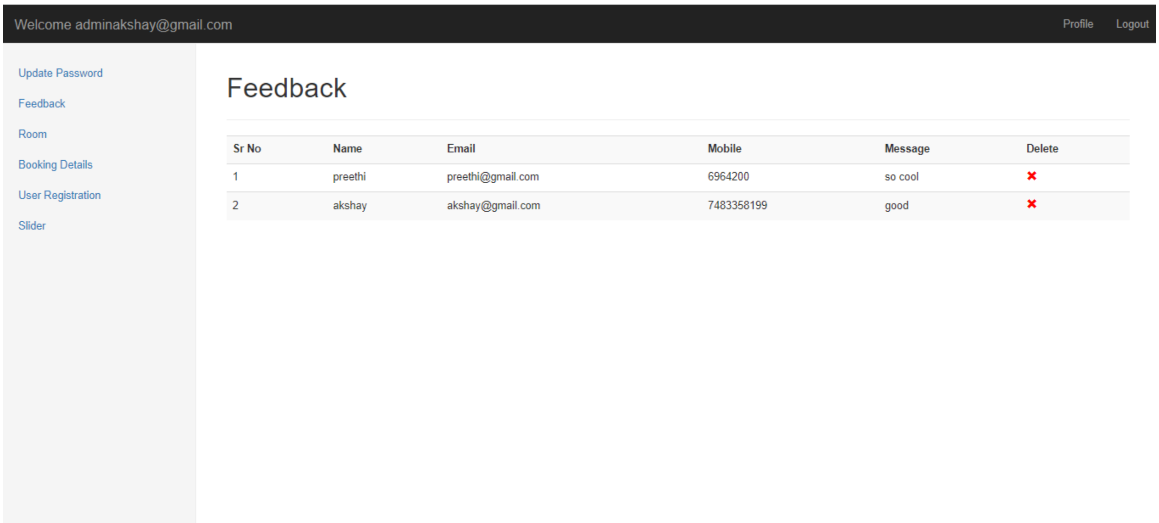
Figure 5. 12 booked room details page



Name:
adminananya@gmail.com
Password:
•••••

Name:
adminakshay@gmail.com
Password:
•••••

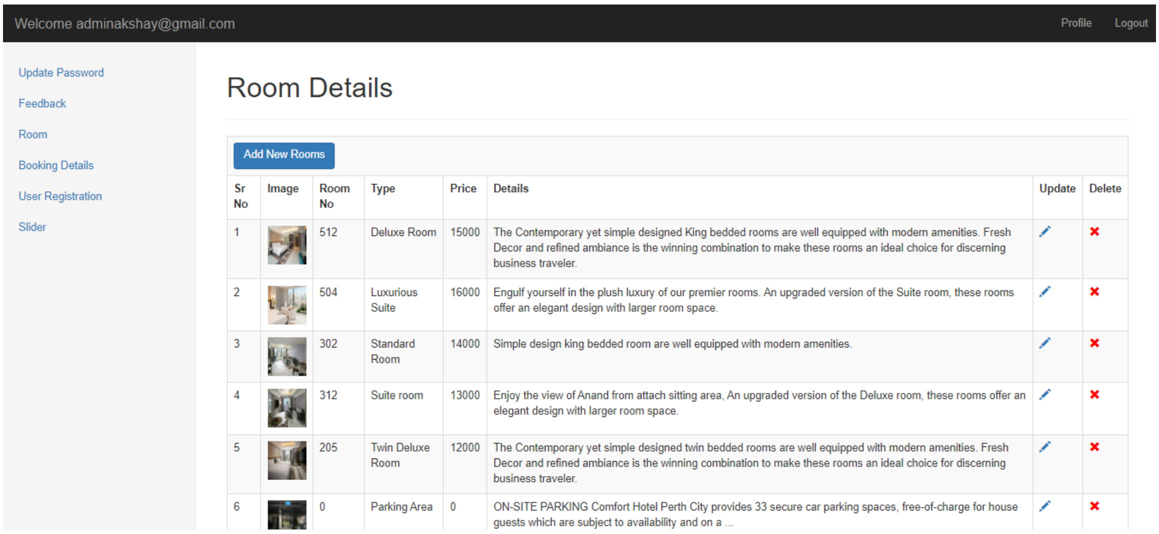
Figure 5. 13 admin dashboard



Feedback

Sr No	Name	Email	Mobile	Message	Delete
1	preethi	preethi@gmail.com	6964200	so cool	✖
2	akshay	akshay@gmail.com	7483358199	good	✖

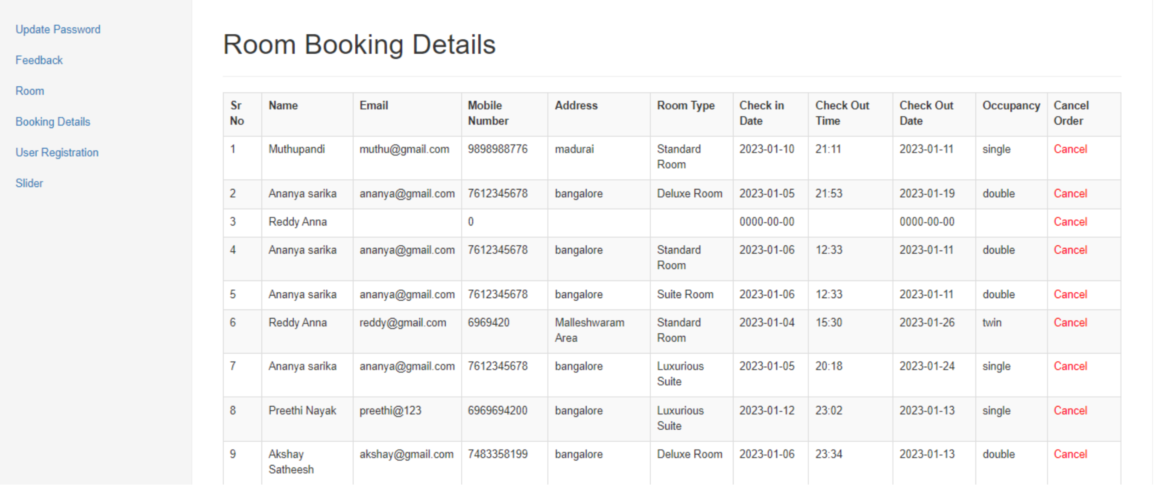
Figure 5. 14 admin edit(feedback)



Room Details

Sr No	Image	Room No	Type	Price	Details	Update	Delete
1		512	Deluxe Room	15000	The Contemporary yet simple designed King bedded rooms are well equipped with modern amenities. Fresh Decor and refined ambiance is the winning combination to make these rooms an ideal choice for discerning business traveler.	✎	✖
2		504	Luxurious Suite	16000	Engulf yourself in the plush luxury of our premier rooms. An upgraded version of the Suite room, these rooms offer an elegant design with larger room space.	✎	✖
3		302	Standard Room	14000	Simple design king bedded room are well equipped with modern amenities.	✎	✖
4		312	Suite room	13000	Enjoy the view of Anand from attach sitting area, An upgraded version of the Deluxe room, these rooms offer an elegant design with larger room space.	✎	✖
5		205	Twin Deluxe Room	12000	The Contemporary yet simple designed twin bedded rooms are well equipped with modern amenities. Fresh Decor and refined ambiance is the winning combination to make these rooms an ideal choice for discerning business traveler.	✎	✖
6		0	Parking Area	0	ON-SITE PARKING Comfort Hotel Perth City provides 33 secure car parking spaces, free-of-charge for house guests which are subject to availability and on a ...	✎	✖

Figure 5. 15 admin edit(room details)



Room Booking Details

Sr No	Name	Email	Mobile Number	Address	Room Type	Check in Date	Check Out Time	Check Out Date	Occupancy	Cancel Order
1	Muthupandi	muthu@gmail.com	9898988776	madurai	Standard Room	2023-01-10	21:11	2023-01-11	single	Cancel
2	Ananya sarika	ananya@gmail.com	7612345678	bangalore	Deluxe Room	2023-01-05	21:53	2023-01-19	double	Cancel
3	Reddy Anna		0			0000-00-00		0000-00-00		Cancel
4	Ananya sarika	ananya@gmail.com	7612345678	bangalore	Standard Room	2023-01-06	12:33	2023-01-11	double	Cancel
5	Ananya sarika	ananya@gmail.com	7612345678	bangalore	Suite Room	2023-01-06	12:33	2023-01-11	double	Cancel
6	Reddy Anna	reddy@gmail.com	6969420	Malleshwaram Area	Standard Room	2023-01-04	15:30	2023-01-26	twin	Cancel
7	Ananya sarika	ananya@gmail.com	7612345678	bangalore	Luxurious Suite	2023-01-05	20:18	2023-01-24	single	Cancel
8	Preethi Nayak	preethi@123	6969694200	bangalore	Luxurious Suite	2023-01-12	23:02	2023-01-13	single	Cancel
9	Akshay Satheesh	akshay@gmail.com	7483358199	bangalore	Deluxe Room	2023-01-06	23:34	2023-01-13	double	Cancel

Figure 5. 16 admin edit(room booking details)

CONCLUSIONS AND FUTUTRE ENHANCEMENTS

Conclusion

The Online Hotel Booking System provides an environment for users to book hotel rooms, perform booking activities, and manage personal account at Alpha Royal Hotel with a Web browser. The system uses Apache Web Server running on Windows platform. The database server is MySQL. To implement the system, the developer has used JavaScript, HTML, and PHP. All dynamic contents are handled by PHP. Persistent data are saved in the database. Online Hotel Booking System is a user-friendly and easy-to-use system of a Web-based application. Everyone who knows how to use a Web browser can register and then "" login to book a room, change booking details, cancel booking, and view or modify personal profile online. It is easy and fast td make a reservation. There can still be improvements for the Online Hotel Booking System.

Future Enhancement

The possible improvements that can be made for the Online Hotel Room Reservation System include: I can make the graphical user interface friendlier and more functional in the next development. The Online Hotel Room Reservation System aims to provide a user-friendly interface and more functions for real world hotels. But there is still some room for improvements. For example, I can change the settings and functions of some options in the Web pages to make them more professional and artistic. I can also use more pop-up windows so that users can choose the value from them directly. In this way the users can avoid many possible mistakes caused by inappropriate input. -. In future improvements, the Online Hotel Room Reservation System can offer more services such as car rental, flight ticket purchase, and the vacation package advising. These services have been offered already on some real world online booking systems. More hotels will add these services on their online systems. In this way, people can make all their requests at once no matter they are business trip arrangement, shopping, travel, or vacation.

REFERENCES/BIBLIOGRAPHY

[1] IEEE Std. 830-1998 IEEE Recommended Practice of Software Requirements Specifications.

[2] Gareth Downes-Powel1, Tim Green, and Bruno Mairlot. "Dreamweaver MX-PHP Web Development", First Edition, Wiley Publishing Inc, July 2002.

[3] Paul DuBois and Michael Widenius. "MySQL", First Edition, New Riders Publishing, December 1999.

[4] Kevin Yank. "Build Your Own Database Driven Website Using PHP & MySQL", First Edition, SitePoint Pty Ltd, September 2001.

[5] Larry Ullman. "PHP and MySQL for Dynamic Web Sites: Visual QuickPro Guide", First Edition, Peachpit Press, March 2003.

[6] Martin Fowler and Kendall Scott. "UML Distilled - A brief guide to the standard object modeling language", Second Edition, Addison-Wesley, July 2001.

[7] Hugh E. Williams and David Lane. "Web Database Applications with PHP & MySQL", First Edition, O'Reilly & Associates, Inc., March 2000.

[8] Rasmus Lerdorf and Kevin Tatore. "Programming PHP", First Edition, O'Reilly & Associates, Inc., March 2002.

[9] Shelly Cashman Woods. "HTML Complete Concepts and Techniques", Second Edition, Thomson Course Technology, 2002.

[10] Jennifer Niederst. "Learning Web Design", First Edition, O'Reilly & Associates, Inc., 2001.

[11] Tim Converse and Joyce Park, "PHP 4 Bible", IDG Books worldwide, Inc., 2000.