VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Machhe, Belagavi, Karnataka 590018

DBMS LABORATORY WITH MINI-PROJECT REPORT

On

HOTEL RESERVATION SYSTEM

Submitted in partial fulfillment of the requirement
for the award of the degree of

Bachelor of Engineering
in
Information Science & Engineering
by

Pooja S (1BG19IS031)

Vidyayāmruthamashnuthe

B.N.M. Institute of Technology

An Autonomous Institution under VTU, Approved by AICTE

Department of Information Science and Engineering
2021 – 2022

B.N.M. Institute of Technology

An Autonomous Institution under VTU

DEPARTMENT OF INFORMATION SCIENCE & ENGINEERING

BHAGEERATHI BAI NARAYANA RAO MAANAY EDUCATIONAL INSTITUTIONS
BANGALORE

Vidyayāmruthamashnuthe

CERTIFICATE

Certified that the Mini-project entitled **Hotel Reservation System** is carried out by **Ms. Pooja S** USN **1BG19IS031** the bonafide student of **B.N.M Institute of Technology** in partial fulfillment for the award of **Bachelor of Engineering** in **Information Science & Engineering** of the **Visvesvaraya Technological University**, Belagavi during the year 2021-2022. It is certified that all corrections / suggestions indicated for Internal Assessment have been incorporated in the report deposited in the department library. The mini-project report has been approved as it satisfies the academic requirements in respect of mini-project prescribed for the said Degree.

Mrs. Mona
Asst. Prof., Dept. of ISE
BNMIT

Dr. Shashikala
Prof & Head, Dept. of ISE
BNMIT

Name & Signature of the Examiners with date:

1.

2.

Table of Contents

Table of Contents

Chapter 1

Introduction

The Hotel reservation system helps the Hotel manager to manage the hotel more effectively and efficiently by computerizing room reservation , billing and inventory control.

1.1 Objective

The objective of this project is to build a web application that can catalogue customer information, menu, manage orders, reserve room and facilitate customers to place orders, reserve rooms provide a holistic view of the current rooms available and in the process, learn and apply the database concepts that are a part of the syllabus.

1.2 Scope of the Project

The aim of our project is to manage and organize the data of a hotel. Since there is a lot of data generated in a given hour, manager is provided with one-stop database to access a particular database quickly and accurately as per the requirement. The access of data is made more convenient for the customers.

1.3 Motivation

We often experience a situation where we have an overload of data. In setting of a, there are numerous entities and attributes one needs to keep track of to run the hotel in a hassle-free manner.

1.4 Requirement Analysis

The Following are the Admin's Requirements:

1.Create, Read, Update and Delete Customer Information.

2.Create, Read, Update and Delete Bill Information.

3.Create, Read, Update and Delete Order Information.

4.Create, Read, Update and Delete Menu.

5.Create, Read, Update and Delete Cashier Information.

6. Allow Customers to place orders.

7.Allow Customers to make payments.

8.An Organizer Dashboard.

9.A Customer Dashboard.

10. Logout and Login functionalities.

The following are the Customer/User Requirements:

1.The user must the give the Username and Password for login credentials.

2.The user should have access to internet and a computer.

3. The user can login and logout at any time.

4. The user can view their orders placed in the cart section in their respective Dashboards.

5.The user can view a list of the waiters and cashiers in their Dashboard.

6.The user can view the restaurant description by clicking on the Restaurant.

Chapter 2

Methodology

2.1 Entity-Relationship Diagram

An entity–relationship model (ER model) describes inter-related things of interest in a specific domain of knowledge. An ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types. An ERD shows the relationships of entity sets stored in a database. An entity in this context is an object, a component of data. An entity set is a collection of similar entities. These entities can have attributes that define its properties. ER Diagrams are most often used to design or debug relational databases in the fields of software engineering, business information systems, education and research ER Diagram for Event Collaboration System consists of 5 regular entities. ER diagrams are related to data structure diagrams (DSDs), which focus on the relationships of elements within entities instead of relationships between entities themselves. An ER diagram consists of the following features

- ➢ Entities
- ➢ Relations
- ➢ Attributes
- ➢ Cardinality Ratios

The regular entities used in Event Collaboration system are

1. CONTACT
2. LOGIN
3. NEWSLETTER LOG
4. PAYMENT
5. ROOM
6. ROOM BOOKING

Description of Entities

- CONTACT - This is the table that holds names and details of thecustomers/users who access the contents of the database

- LOGIN - This is the table in the database that records and stores all the login details
- PAYMENT - This is the table that records the payment details.
- ROOM - This is the table that holds the different types of rooms available in the hotel
- ROOM BOOKING – This is the table that holds details of the booked rooms.

.Their attributes are

- Customer contact (id, fname, lname,phone, address, email)
- Login (id, user_name, password)
- Room (type, bedding, place,room_id)
- Room book (Title, country, nationality, meal, N_room ,T_room ,bedding ,cout date)
- News lettering (id, Title, subject, news)
- Payment (Room rent, bed rent, meals amt, grand total)

The relations between various entities in event collaboration system are

- GENERATES - This is a fully participating binary relationship between payment and bill. The cardinality ratio is 1:n
- RESERVATION - This is a fully participating binary relationship between customer and payment. The cardinality ratio is M:n
- CHECK IN AND OUT - This is a fully participating binary relationship between customer and room.The cardinality ratio is 1:n
- OWN - This is a fully participating binary relationship between room and hotel. The cardinality ratio is m:n
- SERVICES - This is a fully participating binary relationship between customer and food. The cardinality ratio is m:n
- WORKS - This is a fully participating binary relationship between hotel and employee. The cardinality ratio is m

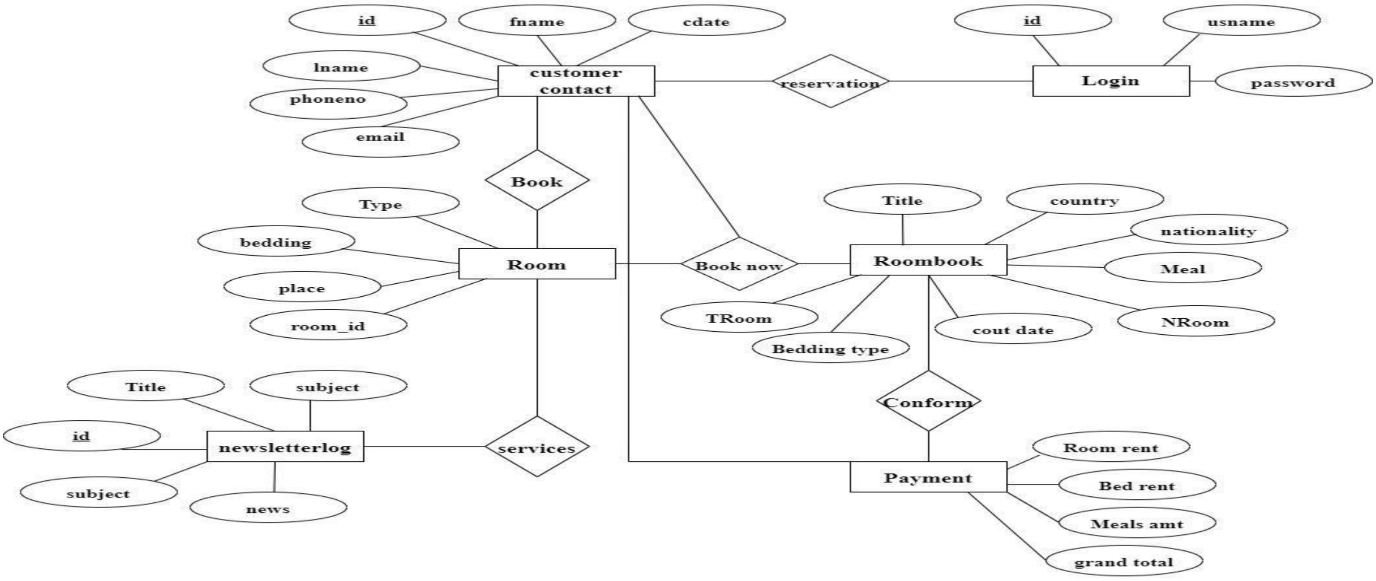


Figure 2.1 E R Diagram

Fig 2.1 shows all the entities and the relationships between each of them. The ER diagram is basically used to show the entities and their attributes in a graphical representation.

2.2 ER to Relational Mapping

ER-to-Relational Mapping Algorithm Step 1: Mapping of Regular Entity TypesStep 2: Mapping of Weak Entity Types Step 3: Mapping of Binary 1:1 Relation

Step 4: Mapping of Binary 1: N Relationship Types Step 5: Mapping of Binary M: N Relationship Types

Step 6: Mapping of Multivalued attributes

Step 7: Mapping of N-ary Relationship Types

- Step 1: Mapping of Regular Entity Types

1.1 : For each regular (strong) entity type E in the ER schema, create a relation R that includes all the simple attributes of E.

1.2 : Choose one of the key attributes of E as the primary key for R.

1.3 : If the chosen key of E is composite, the set of simple attributes that form it will together form the primary key of R.

- Step 2: Mapping of Weak Entity Types

2.1 : For each weak entity type W in the ER schema with owner entity type E, create a relation R & include all simple attributes (or simple components of composite attributes) of W as attributes of R.

2.2 relation R & include all simple attributes (or simple components of composite attributes) of W as attributes of R.

2.3 : Also, include as foreign key attributes of R the primary key attribute(s) of the relation(s) that correspond to the owner entity type(s).

2.4 : The primary key of R is the combination of the primary key(s) of the owner(s) and the partial key of the weak entity type W, if any.

- Step 3: Mapping of Binary 1:1 Relation

For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R. There are three possible approaches:

3.1 : Foreign Key approach: Choose one of the relations-say S-and include a foreign key in S the primary key of T. It is better to choose an entity type with totalparticipation in R in the role of S.

3.2 : Merged relation option: An alternate mapping of a 1:1 relationship type is possible by merging the two entity types and the relationship into a single relation. This may be appropriate when both participations are total.

3.3 : Cross-reference or relationship relation option: The third alternative is to set up a third relation R for the purpose of cross-referencing the primary keys of the two relations S and T representing the entity types.

- Step 4: Mapping of Binary 1: N Relationship Types

4.1 : For each regular binary 1: N relationship type R, identify the relation S that represent the participating entity type at the N-side of the relationship type.

4.2 : For each regular binary 1: N relationship type R, identify the relation S that represent the participating entity type at the N-side of the relationship type

4.3 : Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R.

4.4 : Include any simple attributes of the 1: N relation type as attributes of S.

- Step 5: Mapping of Binary M: N Relationship Types

5.1 : For each regular binary M: N relationship type R, create a new relation S to represent R.

5.2 : Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S.

4.5 :

4.6 Include as foreign key in S the primary key of the relation T that represents the other entity type participating in R.

4.7 : Include any simple attributes of the 1: N relation type as attributes of S.

- Step 5: Mapping of Binary M: N Relationship Types

5.3 : For each regular binary M: N relationship type R, create a new relation S to represent R.

5.4 : Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S.

5.5 : Also include any simple attributes of the M: N relationship type (or simple components of composite attributes) as attributes of S.

- Step 6: Mapping of Multivalued attributes

6.1 : For each multivalued attribute A, create a new relation R.

6.2 : This relation R will include an attribute corresponding to A, plus the primary key attribute K-as a foreign key in R-of the relation that represents the entity type of relationship type that has A as an attribute.

6.3 : The primary key of R is the combination of A and K. If the multivalued attribute is composite, we include its simple components.

- Step 7: Mapping of N-ary Relationship Types

7.1 : For each n-ary relationship type R, where n>2, create a new relationship S torepresent R.

7.2 : Include as foreign key attributes in S the primary keys of the relations thatrepresent the participating entity types.

7.3 : Also include any simple attributes of the n-ary relationship type (or simple components of composite attributes) as attributes of S.

Correspondence between ER and Relational Models

- Entity type “Entity” relation
- 1:1 or 1: N relationship type Foreign key (or “relationship” relation)
- M: N relationship type “Relationship” relation and two foreign keys
- n-ary relationship type “Relationship” relation and n foreign keys

- Simple attribute • Composite attribute Set of simple component attributes
- Multi-valued attribute Relation and foreign key
- Value set Domain
- Key attribute Primary(or secondary) key.

2.3 Relational Database Schema

A relational database schema is the tables, columns and relationships that make up a relational database. A relational database schema helps admins and users to organize and understand the structure of a database. This is particularly useful when designing a new database, modifying an existing database to support more functionality, or building integration between databases.

The schema diagram for Restaurant Management System consists of attributes, key attributes such as foreign key and primary key. Each table has a primary key which is a not null type of data and is fixed and cannot be modified further once entered. Database schema is a basic outline of the database of a database management system which defines the common elements of a database and schema makes it easier to implement the operations efficiently on the relations and other attributes. The Schema for the Hotel Management System is shown with all the key attributes and their references using rows and arrows.

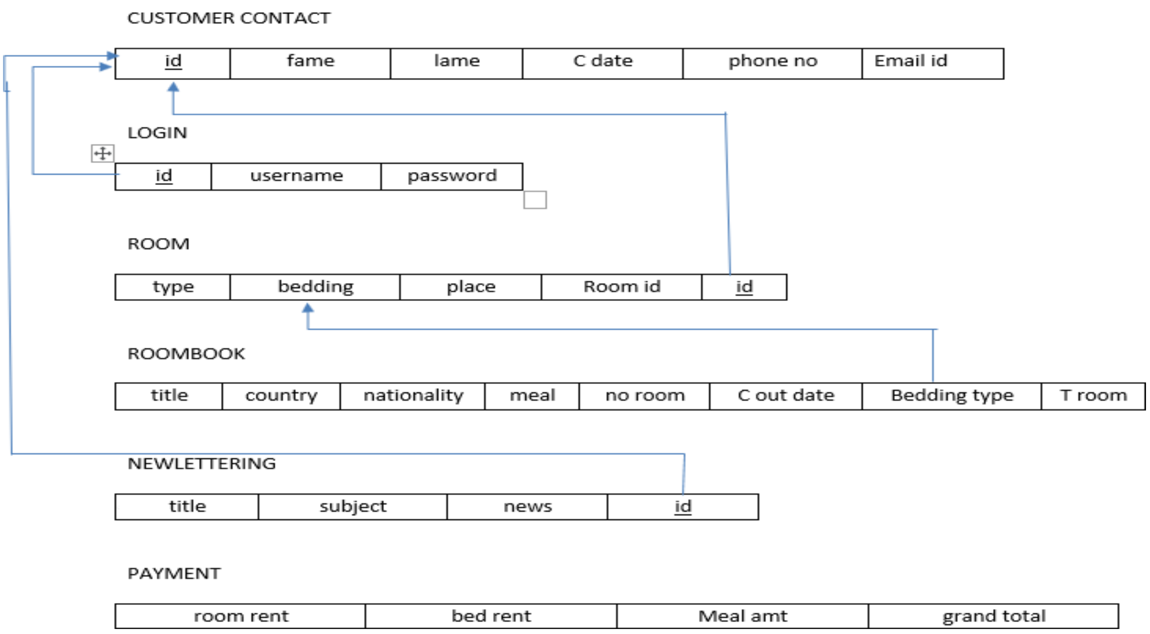


Figure 2.3 Relational Database System

- Fig 2.3 shows the relations (tables) used in Restaurant Management System. The underlined attributes are the primary keys of that table and the references for the foreign keys are shown using arrows.

2.4 Relational Model

The relational model (RM) for database management is an approach to managing data using a structure and language consistent with first-order predicate logic. The Relational model contains all entities and their relations with other entities. It also contains all relations having (m: n) cardinality. Relational data model is the primary data model, which is used widely around the world for data storage and processing.

This data model shows the foreign key and primary key as PK and FK and the reference table or tuple for the foreign key is specified by arrows. This makes it much easier to understand the structure of the schema more than the schema diagram. The relational model (RM) for database management is an approach to managing data using a structure and language consistent with first-order predicate logic.

2.5 Normalised Relational Schema

Normalization involves arranging attributes in relations based on dependencies between attributes, ensuring that the dependencies are properly enforced by database integrity constraints. Normalization is accomplished by applying some formal rules either by a process of synthesis or decomposition. Synthesis creates a normalized database design based on a known set of dependencies. Decomposition takes an existing (insufficiently normalized) database design and improves it based on the known set of dependencies. Normalization involves arranging attributes in relations based on dependencies between attributes, ensuring that the dependencies are properly enforced by database integrity constraints. Normalization is accomplished by applying some formal rules either by a process of synthesis or decomposition.

1 NF - First normal form

First normal form (1NF) is a property of a relation in a relational database. A relation is in first normal form if and only if the domain of each attribute contains only atomic (indivisible) values, and the value of each attribute contains only a single value from that domain. First normal form enforces these criteria:

- Eliminate repeating groups in individual tables.
- Create a separate table for each set of related
- Identify each set of related data with a primary key

The attributes which are in 1NF in Restaurant Management System are

- **Customer contact** - id, fname, lname,phone, address, email)
- **Login** – id , user_name , password
- **Room** - type, bedding, place, room_id
- **Room book** (Title, country, nationality, meal, N_room , T_room ,bedding ,cout date)
- **News lettering** - id, Title, subject, news
- **Payment** - Room rent, bed rent, meals amt, grand total

2NF: Second normal form

Second normal form (2NF) is a normal form used in database normalization. A relation that is in first normal form (1NF) must meet additional criteria if it is to qualify for second normal form. Specifically: a relation is in 2NF if it is in 1NF and no non-prime attribute is dependent on any proper subset of any candidate key of the relation. A non-prime attribute of a relation is an attribute that is not a part of any candidate key of the relation.

The attributes which are in 2NF in Hotel Management System are

- **Customer contact** - id, fname, lname,phone, address, email)
- **Login** – id , user_name , password
- **Room** - type, bedding, place, room_id
- **Room book** (Title, country, nationality, meal, N_room , T_room ,bedding ,cout date)
- **News lettering** - id, Title, subject, news
- **Payment** - Room rent, bed rent, meals amt, grand total

3NF: Third normal form

Third normal form(3NF) is a normal form that is used in normalizing a database design to reduce the duplication of data and ensure referential integrity by ensuring that (1) the entity is in second normal form, and (2) all the attributes in a table are determined only by the candidate keys of that relation and not by any non-prime attributes. 3NF was designed to improve database processing while minimizing storage costs. 3NF data modelling was ideal for online transaction processing (OLTP) applications with heavy order entry type of needs. The Mapped relations or tables are a combination of two or more tables which consists of one or more primary keys of different tables. In this project, the mapped tables mainly consist of primary keys of one or more tables.

The attributes which are in 3NF in Restaurant Management System are

- **Customer contact** - id, fname, lname,phone, address, email)
- **Login** – id , user_name , password
- **Room** - type, bedding, place, room_id
- **Room book** (Title, country, nationality, meal, N_room , T_room ,bedding ,cout date)
- **News lettering** - id, Title, subject, news
- **Payment** - Room rent, bed rent, meals amt, grand total

2.6 Key Attributes

Table 2.6 Key Attributes

Entities	**Primary Key**	**Foreign Key**
Customer Contact	id	-
Login	-	id
Room	type	-
Room book	title	**Bedding**
News Lettering	-	Id, title
Payment	-	id

Table 2.6 shows primary keys and foreign keys in the database.

- PRIMARY KEY

Primary key is a set of one or more fields/columns of a table that uniquely identify a record in database table. It cannot accept null, duplicate values. Only one Candidate Key can be Primary Key. Restaurant Management System contains primary keys – manager_id, restaurant_id, ID_no, order_no, item_no, customer_id,bill_no, cashier_id, id.

- FOREIGN KEY

Foreign Key is a field in database table that is Primary key in another table. It canaccept multiple null, duplicate values. The foreign key in Restaurant ManagementSystem are ID_no, manager_id, restaurant_id, bill_no, order_no,customer_id.

- UNIQUE KEY

Unique key is a set of one or more fields/columns of a table that uniquely identifya record in database table. It is like Primary key but it can accept only one null value and it cannot have duplicate values.

- NOT NULL

The NOT NULL constraint enforces a column to NOT accept NULL values. This enforces a field

This enforces a field to always contain a value, which means that insertion of a new record, or updating of a record without adding a value to this field cannot be done. A field which has unique values is, essentially, a key. However, a key is used to uniquely identify a row in a table, while an index is used to sort, or group, the rowsin the table. A key should not change once it has been initially set, as it might be referenced to elsewhere in the database. Restaurant Management System contains not null constraint attributes.

- THE DEFAULT SQL CONSTRAINT

  At times, to insert a value into a column when no other value is provided can be done automatically. That's where the DEFAULT SQL constraint comes in. The constraint can be used to define a column's default value. This can be a handy way to add the current date and time to a column or to avoid having to use NULL values. Restaurant Management System contains default constraint attributes.

- CHECK CONSTRAINT

  A check constraint is a type of integrity constraint in SQL which specifies a requirement that must be met by each row in a database table. The constraint must be a predicate. It can refer to a single column, or multiple columns of the table.

Chapter 3

System Requirement and Specifications

3.1 User Requirements

User Interface:

- Front-end software: HTML, CSS, JAVASCRIPT, BOOTSTRAP
- Back-end software: MySQL 8.0 CE WORKBENCH

3.2 Software Requirements

1. Operating systems: Windows 10 Home
2. Database Management Systems: MySQL 8.0. CE WORKBENCH
3. Web Server: Node.js 12.0
4. Compatible Browser: Google Chrome
5. Text editor: Visual Studio Code, Atom

3.3 Hardware Requirements

a. Processor: intel® core TM i5-6500 CPU @ 3.20 GHz 3.20 GHz

b. Installed memory (RAM): 8.00 GB (7.88 GB usable)

c. System type: 64-bit operating system, x64-based processor

d. Pen and touch: No pen or touch input is available for this display

3.4 Functional Requirements

- The system shows a list of cards of rooms. Each card contains a

picture of the room, below which the price is displayed

- The customer will be able to add the desired room to the cart.
- After the customer books the room, the payment page will be display
- The admin will be able to add all necessary information about the staff and also be able to add and delete room in the list.

3.5 Non-Functional Requirements

Performance Requirements - The system must be interactive and the delays involved must be less. There should be very less delay in displaying the rooms in the booking page and it should be visible to the admin. The customer should be able to add and remove items from the cart without experiencing any delay.

Safety Requirements – The system should be user friendly. It should not cause any safety violations.

Security Requirements – There is a need for an encrypted login authentication for admin and customer since sensitive information and inventory should be protected. Information transmission should be securely transmitted without any changes in information in order to avoid disturbances in orders and billing.

3.6 Tools and Languages used

3.6.1 Tools

The following lists the various software, frameworks and editors that will be used for the project.

- **Visual Studio Code [V 2019] -** Visual Studio Code is a free source-code editor made by Microsoft for Windows, Linux and macOS. Features include support for debugging, syntax highlighting, intelligent code completion, snippets, code refactoring, and embedded Git. Users can change the theme, keyboard shortcuts, preferences, and install extensions that add additional functionality.
- **MySQL Workbench -** MySQL Workbench is a visual database design tool that integrates SQL development, administration, database design, creation and maintenance into a single integrated development environment for the MySQL database system. It is the

- successor to DB Designer 4 from fabFORCE.net, and replaces the previous package of software, MySQL GUI Tools Bundle.

- **XAMPP** - XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, Maria DB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components

XAMPP, it makes transitioning from a local test server to a live server possible. XAMPP's ease of deployment means WAMP or LAMP stack can be installed quickly and simply on an operating system by a developer, with the advantage that common add- in applications such as Word Press and Joomla! can also be installed with similar ease using Bitnami. Though it is a heavy app for most of the operating systems even when owing to its less size it takes a load on the processor speed

- Draw IO - Draw io is free online diagram software for making flowcharts, process diagrams, org charts, UML, ER and network diagrams. It is used to create schema diagrams and other type of UML diagrams, export diagrams as PNG

3.6.2 Languages

HTML - Hypertext Mark-up Language (HTML) is the standard mark-up language for creating webpages and web applications. With Cascading Style Sheets (CSS) and JavaScript it forms a triad of cornerstone technologies for the World Wide Web.HTML elements are the building blocks of HTML pages. The definition of HTML is Hypertext Mark-up Language:

- Hypertext is the method by which you move around on the web-by clicking on special text called hyperlink which bring you to the next page. The fact that it is hyper just means it is not Restaurant Management System Dept. of ISE, BNMIT 2019-2020 Page 17 linear-i.e., you can go to any place on the Internet whenever you want by clicking on links.

- Mark-up is what HTML tags do to the text inside them. They mark it as a certain type of text (italicised text, for instance).

- HTML is a Language, as it has code-words and syntax like any language. With HTML constructs, images and other objects, such as interactive forms, may be embedded into the rendered page. It provides a means to create structured documents by denotin

- structural semantics for text such as headings, paragraphs, lists, links, quotes and other items.

- HTML elements are delineated by tags, written using angle brackets. Tags such as

<img /> and <input /> introduce content into the page directly. Others 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 embed programs written in a scripted language such as JavaScript which affect the behaviour and content of the web page.

CSS – Cascading Style Sheets

- Inclusion of CSS defines the look and layout of content. HTML consists of a series of short codes typed into text-file by the site author-these are the tags. The text is then saved as a html file, and viewed through a browser, like Internet Explorer or Netscape Navigator. This browser reads the file and translates the text into a visible fo

JavaScript

- JavaScript often abbreviated as JS, is a programming language that conforms to the ECMA Script specification. JavaScript is high-level, often just-in-time compiled, and multi-paradigm. It has curly-bracket syntax, dynamic typing, prototype-based object-orientation, and first-class functions.

- Alongside HTML and CSS, JavaScript is one of the core technologies of the World Wide Web. JavaScript enables interactive web pages and is an essential part of web applications. The vast majority of websites use it for client-side page behaviour, and all major web browsers have a dedicated JavaScript engine to execute it.

As a multi-paradigm language, JavaScript supports event-driven, functional, and imperative programming styles. It has application programming interfaces (APIs) for working with text, dates, regular expressions, standarard.

PHP

- PHP is a general-purpose scripting language especially suited to web development. It was originally created by Danish-Canadian programmer Rasmus Lerdorf in 1994. The PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Preprocessor.

- PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of an HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control. Arbitrary PHP code can also be interpreted and executed via command-line interface (CLI).

- There are a lot more things to do with PHP:

○ User can generate dynamic pages and files.

○ User can create, open, read, write and close files on the server

○ User can collect data from the web form such as user information, email, credit card info and much more.

MySQL

MySQL is a fast, easy-to-use RDBMS being used for many small and big businesses. MySQL is developed, marketed and supported by MySQL AB, which is Swedish company.

- MySQL is becoming so popular because of many good reasons-
- MySQL is released under an open-source license, which makes it free to use.
- MySQL is a very powerful program. It handles a large subset of the functionality of the most expensive and powerful database packages

MySQL works on many operating systems and with many languages including PHP, PERL, C, C++, JAVA, and so on

Chapter 4

System Design and Development

4.1 Architectural Design

Requirements of the software should be transformed into an architecture that describes the software's top-level structure and identifies its components. This is accomplished through architectural design (also called system design), which acts as a preliminary blueprint from which software can be developed. IEEE architectural design as the process of defining a collection of hardware and software components and their interface to establish the framework for the development of a computer system. This framework is established by examining the software requirement document and designing a model for providing implementation details. These details are used to specify the components of the system along with their inputs, outputs, functions, and the interaction between them. An architectural design performs several functions.

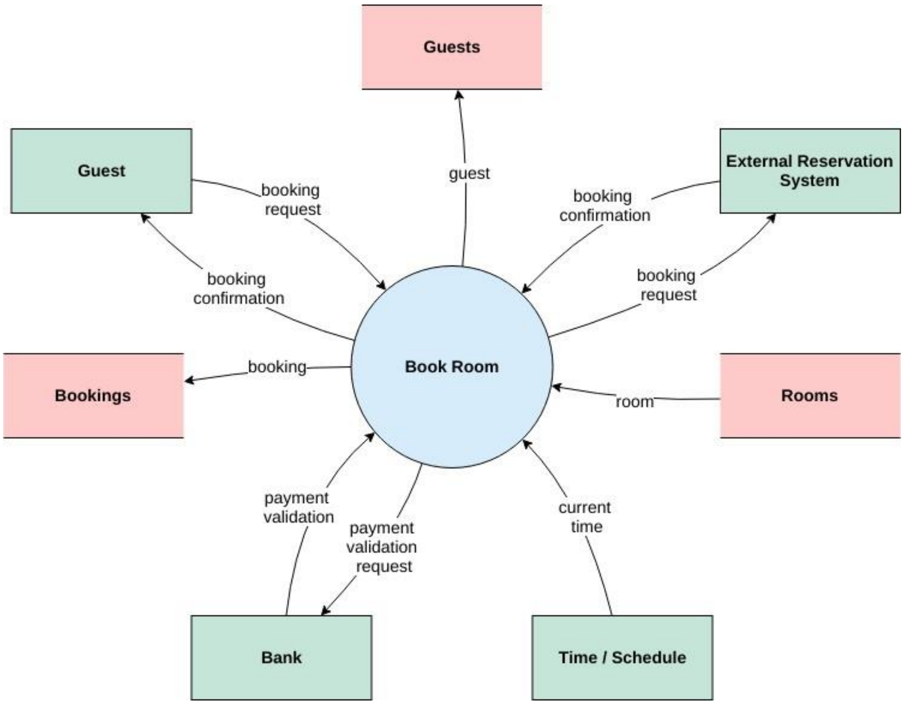


Fig 4.1: Architectural Design

Chapter 5

Implementation

Back End:

From requirements collected and analysed, the conceptual schema for the database is created using the entity-relationship (ER) diagram. Back End is used to create the table and insert the values in the respective tables in the database management system. The car rental system back end focuses on application data access, database administration, data transformation and backup methods.

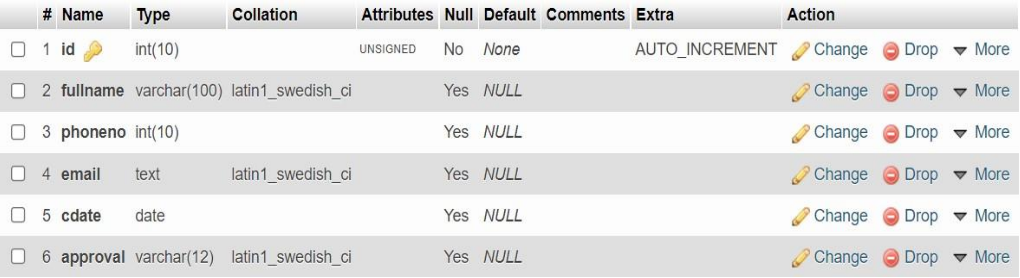
5.1 List of Modules

- **Contact-** This entity specifies the list of users
- **Login -** This entity specifies the list of reservations done by users
- **Newsletter log -** This entity specifies the list of payments done by users.
- **Room -** This entity specifies the list of types of rooms .
- **Roombook-** This entity specifies the list of items present in the menu.

5.2 Module Description

Create Table Statements:

```
CREATE TABLE `contact` (
`id` int(10) unsigned NOT NULL,
  `fullname` varchar(100) DEFAULT NULL,
  `phoneno` int(10) DEFAULT NULL,
  `email` text,
  `cdate` date DEFAULT NULL,
  `approval` varchar(12) DEFAULT NULL
);
```

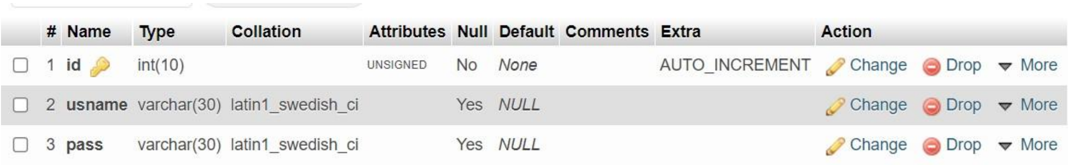


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	id	int(10)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
□	2	fullname	varchar(100)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
□	3	phoneno	int(10)			Yes	*NULL*			Change Drop More
□	4	email	text	latin1_swedish_ci		Yes	*NULL*			Change Drop More
□	5	cdate	date			Yes	*NULL*			Change Drop More
□	6	approval	varchar(12)	latin1_swedish_ci		Yes	*NULL*			Change Drop More

Figure 5.1: Contact table

In the figure 5.1, the table for contact is created which includes id, fullname , phoneno, email, cdate and approval as attributes

```
CREATE TABLE IF NOT EXISTS `login` (
`id` int(10) unsigned NOT NULL,
  `usname` varchar(30) DEFAULT NULL,
  `pass` varchar(30) DEFAULT NULL
);
```

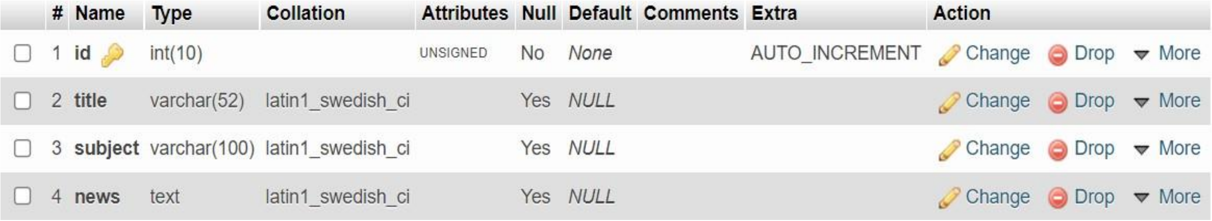


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	id	int(10)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change Drop More
□	2	usname	varchar(30)	latin1_swedish_ci		Yes	*NULL*			Change Drop More
□	3	pass	varchar(30)	latin1_swedish_ci		Yes	*NULL*			Change Drop More

Figure 5.2: Login table

In the figure 5.2, the table for login is created which includes id username and pass as attributes

```
CREATE TABLE IF NOT EXISTS `newsletterlog` (
`id` int(10) unsigned NOT NULL,
  `title` varchar(52) DEFAULT NULL,
  `subject` varchar(100) DEFAULT NULL,
  `news` text
);
```

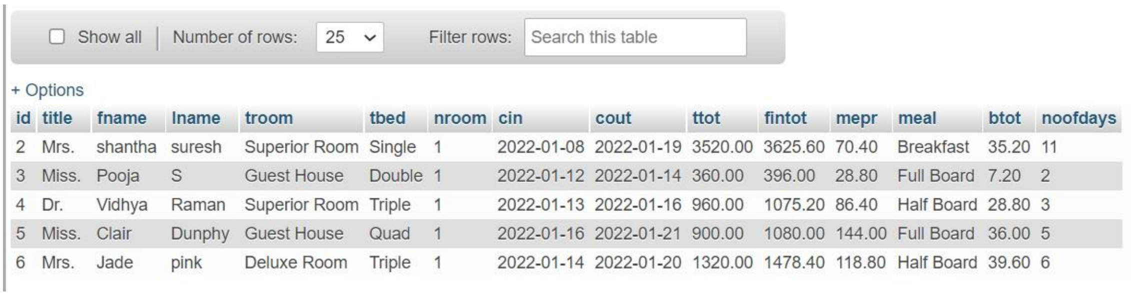


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
□	1	**id**	int(10)		UNSIGNED	No	*None*		AUTO_INCREMENT	Change	Drop	More
□	2	**title**	varchar(52)	latin1_swedish_ci		Yes	*NULL*			Change	Drop	More
□	3	**subject**	varchar(100)	latin1_swedish_ci		Yes	*NULL*			Change	Drop	More
□	4	**news**	text	latin1_swedish_ci		Yes	*NULL*			Change	Drop	More

Figure 5.3 :Newsletter table

In the figure 5.3, the table for contact is created which includes id, title, subject and news as attributes

```
CREATE TABLE IF NOT EXISTS `payment` (
  `id` int(11) DEFAULT NULL,
  `title` varchar(5) DEFAULT NULL,
  `fname` varchar(30) DEFAULT NUL
`lname` varchar(30) DEFAULT NULL,
  `troom` varchar(30) DEFAULT NULL,
  `tbed` varchar(30) DEFAULT NULL,
  `nroom` int(11) DEFAULT NULL,
  `cin` date DEFAULT NULL,
  `cout` date DEFAULT NULL,
  `ttot` double(8,2) DEFAULT NULL,
  `fintot` double(8,2) DEFAULT NULL,
  `mepr` double(8,2) DEFAULT NULL,
  `meal` varchar(30) DEFAULT NULL,
  `btot` double(8,2) DEFAULT NULL,
  `noofdays` int(11) DEFAULT NULL
);
```



□ Show all | Number of rows: 25 Filter rows: Search this table

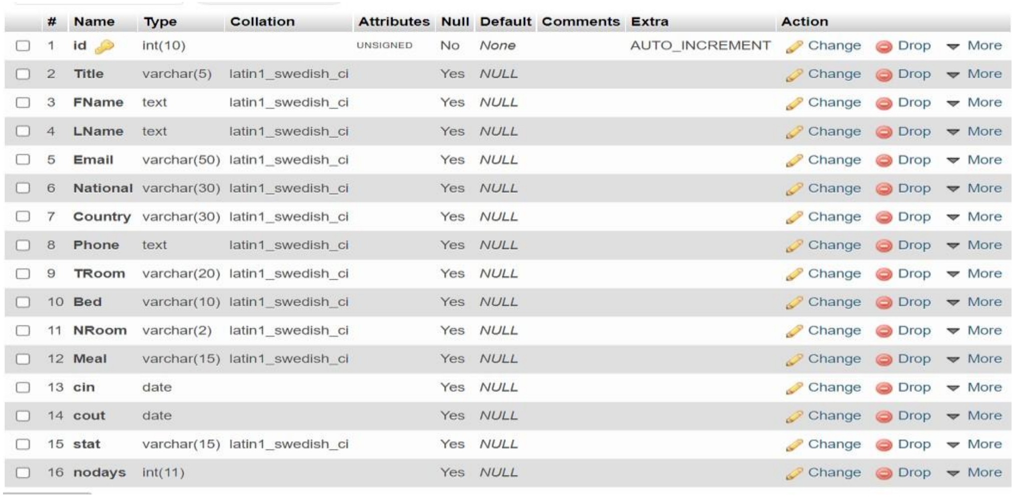
+ Options

id	title	fname	lname	troom	tbed	nroom	cin	cout	ttot	fintot	mepr	meal	btot	noofdays
2	Mrs.	shantha	suresh	Superior Room	Single	1	2022-01-08	2022-01-19	3520.00	3625.60	70.40	Breakfast	35.20	11
3	Miss.	Pooja	S	Guest House	Double	1	2022-01-12	2022-01-14	360.00	396.00	28.80	Full Board	7.20	2
4	Dr.	Vidhya	Raman	Superior Room	Triple	1	2022-01-13	2022-01-16	960.00	1075.20	86.40	Half Board	28.80	3
5	Miss.	Clair	Dunphy	Guest House	Quad	1	2022-01-16	2022-01-21	900.00	1080.00	144.00	Full Board	36.00	5
6	Mrs.	Jade	pink	Deluxe Room	Triple	1	2022-01-14	2022-01-20	1320.00	1478.40	118.80	Half Board	39.60	6

Figure 5.4:Payment table

In the figure 5.4, the table for contact is created which includes id, title, fname, lname, troom, tbed, nroon, cin, cout, ttot, fintot, mepr, meal, btot and noofdays as attributes.

```
CREATE TABLE IF NOT EXISTS `room` (
`id` int(10) unsigned NOT NULL,
  `type` varchar(15) DEFAULT NULL,
  `bedding` varchar(10) DEFAULT NULL,
  `place` varchar(10) DEFAULT NULL,
  `cusid` int(11) DEFAULT NULL
);
```

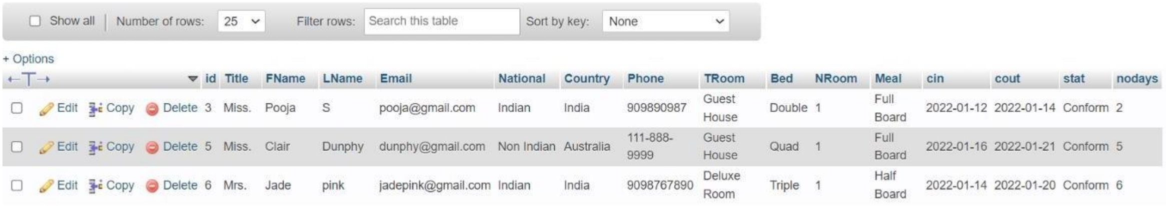


	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action		
□	1	id	int(10)		UNSIGNED	No	None		AUTO_INCREMENT	Change	Drop	More
□	2	Title	varchar(5)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	3	FName	text	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	4	LName	text	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	5	Email	varchar(50)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	6	National	varchar(30)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	7	Country	varchar(30)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	8	Phone	text	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	9	TRoom	varchar(20)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	10	Bed	varchar(10)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	11	NRoom	varchar(2)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	12	Meal	varchar(15)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	13	cin	date			Yes	NULL			Change	Drop	More
□	14	cout	date			Yes	NULL			Change	Drop	More
□	15	stat	varchar(15)	latin1_swedish_ci		Yes	NULL			Change	Drop	More
□	16	nodays	int(11)			Yes	NULL			Change	Drop	More

Figure 5.5 :Room table

In the figure 5.5, the table for contact is created which includes id, title, fname, lname, email, national, country, phone, troom, tbed, nroon, cin, cout, ttot, fintot, mepr, meal, stat and nodays as attributes.

```
CREATE TABLE IF NOT EXISTS `roombook` (
`id` int(10) unsigned NOT NULL,
 `Title` varchar(5) DEFAULT NULL,
 `FName` text,
  `LName` text,
  `Email` varchar(50) DEFAULT NULL,
  `National` varchar(30) DEFAULT NULL,
  `Country` varchar(30) DEFAULT NULL,
  `Phone` text,
  `TRoom` varchar(20) DEFAULT NULL,
  `Bed` varchar(10) DEFAULT NULL,
  `NRoom` varchar(2) DEFAULT NULL,
  `Meal` varchar(15) DEFAULT NULL,
  `cin` date DEFAULT NULL,
  `cout` date DEFAULT NULL,
  `stat` varchar(15) DEFAULT NULL,
  `nodays` int(11) DEFAULT NULL
) ;
```



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

				id	Title	FName	LName	Email	National	Country	Phone	TRoom	Bed	NRoom	Meal	cin	cout	stat	nodays
☐	Edit	Copy	Delete	3	Miss.	Pooja	S	pooja@gmail.com	Indian	India	909890987	Guest House	Double	1	Full Board	2022-01-12	2022-01-14	Conform	2
☐	Edit	Copy	Delete	5	Miss.	Clair	Dunphy	dunphy@gmail.com	Non Indian	Australia	111-888-9999	Guest House	Quad	1	Full Board	2022-01-16	2022-01-21	Conform	5
☐	Edit	Copy	Delete	6	Mrs.	Jade	pink	jadepink@gmail.com	Indian	India	9098767890	Deluxe Room	Triple	1	Half Board	2022-01-14	2022-01-20	Conform	6

Figure 5.6:Room booking

In the figure 5.6, the table for contact is created which includes id, title, fname, lname, email, national, country, phone, troom, tbed, nroon, cin, cout, ttot, fintot, mepr, meal, stat and nodays as attributes.

HEADER PAGE:

```
        <html>
        <head>
        <meta charset="utf-8">
        <title>Details of Book key</title>
        <style>
                /* reset */

*
{border: 0;

box-sizing: content-box;
        color: inherit;
        font-family: inherit;
        font-size: inherit;
        font-style: inherit;
        font-weight: inherit;
        line-height: inherit;
        list-style: none;
        margin: 0;
        padding: 0;
        text-decoration: none;
        vertical-align: top;
}
/* content editable */
*[contenteditable] { border-radius: 0.25em; min-width: 1em; outline: 0; }
*[contenteditable] { cursor: pointer; }
*[contenteditable]:hover, *[contenteditable]:focus, td:hover *[contenteditable], td:focus
*[contenteditable], img.hover { background: #DEF; box-shadow: 0 0 1em 0.5em #DEF;
}
span[contenteditable] { display: inline-block; }
```

```
          /* heading */
h1 { font: bold 100% sans-serif; letter-spacing: 0.5em; text-align: center; text-transform: uppercase; }
/* table */
table { font-size: 75%; table-layout: fixed; width: 100%; }
table { border-collapse: separate; border-spacing: 2px; }
th, td { border-width: 1px; padding: 0.5em; position: relative; text-align: left; }
th, td { border-radius: 0.25em; border-style: solid; }
th { background: #EEE; border-color: #BBB; }
td { border-color: #DDD; }
/* page */
html { font: 16px/1 'Open Sans', sans-serif; overflow: auto; padding: 0.5in; }
html { background: #999; cursor: default; }
body { box-sizing: border-box; height: 11in; margin: 0 auto; overflow: hidden; padding: 0.5in; width: 8.5in
body { background: #FFF; border-radius: 1px; box-shadow: 0 0 1in -0.25in rgba(0, 0, 0, 0.5); }
/* header */
header { margin: 0 0 3em; }
header:after { clear: both; content: ""; display: table; }
header h1 { background: #000; border-radius: 0.25em; color: #FFF; margin: 0 0 1em; padding: 0.5em 0; }
header address { float: left; font-size: 75%; font-style: normal; line-height: 1.25; margin: 0 1em 1em 0; }
header address p { margin: 0 0 0.25em; }
header span, header img { display : block; float: right; }
header span { margin: 0 0 1em 1em; max-height: 25%; max-width: 60%; position: relative; }
header img { max-height: 100%; max-width: 100%; }
header input { cursor: pointer; -ms-filter:"progid:DXImageTransform.Microsoft.Alpha(Opacity=0)"; height: 100%; left: 0; opacity: 0; position: absolute; top: 0; width: 100%;
/* article */
article, article address, table.meta, table.inventory { margin: 0 0 3em; }
article:after { clear: both; content: ""; display: table; }
article h1 { clip: rect(0 0 0 0); position: absolute; }
article address { float: left; font-size: 125%; font-weight: bold; }
```

```
    /* table meta & balance */
    table.meta, table.balance { float: right; width: 36%; }
}

table.meta:after, table.balance:after { clear: both; content: ""; display: table; }
/* table meta */
table.meta th { width: 40%; }
table.meta td { width: 60%; }
/* table items */
table.inventory { clear: both; width: 100%; }
table.inventory th { font-weight: bold; text-align: center; }
table.inventory td:nth-child(1) { width: 26%; }
table.inventory td:nth-child(2) { width: 38%; }
table.inventory td:nth-child(3) { text-align: right; width: 12%; }
table.inventory td:nth-child(4) { text-align: right; width: 12%; }
table.inventory td:nth-child(5) { text-align: right; width: 12%; }

/* table balance */
table.balance th, table.balance td { width: 50%; }
table.balance td { text-align: right; }
/* aside */
aside h1 { border: none; border-width: 0 0 1px; margin: 0 0 1em; }
aside h1 { border-color: #999; border-bottom-style: solid; }

/* javascript */
.add, .cut
{
        border-width: 1px;
        display: block;
        font-size: .8rem;
        padding: 0.25em 0.5em;
        float: left;
        text-align: center;
        width: 0.6em;
}
```

```
.add, .cut
{
background: #9AF;
        box-shadow: 0 1px 2px rgba(0,0,0,0.2);
        background-image: -moz-linear-gradient(#00ADEE 5%, #0078A5 100%);
        background-image: -webkit-linear-gradient(#00ADEE 5%, #0078A5 100%);
        border-radius: 0.5em;
        border-color: #0076A3;
        color: #FFF;
        cursor: pointer;
        font-weight: bold;
        text-shadow: 0 -1px 2px rgba(0,0,0,0.333);
}
.add { margin: -2.5em 0 0; }
.add:hover { background: #00ADEE;

.cut { opacity: 0; position: absolute; top: 0; left: -1.5em; }
.cut { -webkit-transition: opacity 100ms ease-in; }
tr:hover .cut { opacity: 1; }
@media print {
    * { -webkit-print-color-adjust: exact; }
    html { background: none; padding: 0; }
    body { box-shadow: none; margin: 0; }
    span:empty { display: none; }
    .add, .cut { display: none; }
}
@page { margin: 0; }
</style>
</head>
<body>
<?php
ob_start();
include ('db.php');
$pid = $_GET['sid'];
```

```
$sql ="select * from roombook where id = '$pid' ";
$re = mysqli_query($con,$sql);

while($row=mysqli_fetch_array($re))
{
$id = $row['id'];
$title =  $row['Title'];
$Fname = $row['FName'];
$lname = $row['LName'];
$email = $row['Email'];
$National = $row['National'];
$country = $row['Country'];
$phone = $row['Phone'];
$room_type = $row['TRoom'];
$Bed_type = $row['Bed'];

                    //$Noof_room = $row['Nroom']
$meal_type = $row['Meal'];
$cin_date = $row['cin'];
$cout_date = $row['cout'];
$nodays = $row['nodays'];
}
?>
<header>
<h1>Information of Guest</h1>
<address >
<p>SUN RISE HOTEL,</p>
<p>UBY CITY  br>BANGALORE,<br>INDIA</p>
<p> +91 9678978930</p>
</address>
<span><img alt="" src="assets/img/sun.png"></span>
</header>
<article>
```

```
        <address >

<p><br></p>
<p>Coustomer Name : - <?php echo $title.$Fname." ".$lname;?><br></p>
</address>
<table class="meta">
<tr>
<th><span >Customer ID</span></th>
<td><span ><?php echo $id; ?></span></td>
</tr>
<tr>
<th><span >Check in Date</span></th>
<td><span ><?php echo $cin_date; ?> </span></td>
</tr>
<tr>
<the><span >Check out Date</span></th>
 <td><span ><?php echo $cout_date; ?> </span></td>
</tr>
</table>
<table >
<tr>
<td>Customer phone : - <?php echo $phone; ?> </td>
<td>Customer email : - <?php echo $email; ?> </td>
</tr>
<tr>
<td>Customer Country : - <?php echo $country; ?> </td>
<td>Customer National : - <?php echo $National; ?> </td>
</tr>
</table>
<br>
<br>
<table class="inventory">
<thead>
<tr>
```

```
<th><span >Item</span></th>
<th><span >No of Days</span></th>
</tr>

</thead>
<tbody>
<tr>
<td><span ><?php echo $room_type; ?></span></td>
<td><span ><?php echo $nodays; ?> </span></td>
</tr>
<tr>
<td><span ><?php echo $Bed_type; ?> Bed </span></td>
                    <td><span ><?php echo $nodays; ?></span></td>
                     </tr>
                <tr>
<td><span ><?php echo $meal_type; ?> </span></td>

<td><span ><?php echo $nodays; ?></span></td>
</tr></tbody>
</table>
</article>
<aside>
<h1><span >Contact us</span></h1>
<div >
<p align="center">Email :- info@sunrise.com || Web :- www.sunrise.com || Phone :- +94 65 222 44 55 </p>
</div>
</aside>
</body>
</html>
<?php
ob_end_flush();
?>
```

INDEX PAGE:

```
<?php
 session_start();
 if(isset($_SESSION["user"]))
 {
   header("location:home.php");
 }
 ?>
<!DOCTYPE html>
<html >
<head>
  <meta charset="UTF-8">
  <title>SUN RISE ADMIN</title>
      <link rel="stylesheet" href="css/style.css">
</head>
<body>
  <div id="clouds">
            <div class="cloud x1"></div>
            <!-- Time for multiple clouds to dance around -->
            <div class="cloud x2"></div>
            <div class="cloud x3"></div>
            <div class="cloud x4"></div>
            <div class="cloud x5"></div>
</div>
 <div class="container">
      <div id="login">
        <form method="post"

          <fieldset class="clearfix">
            <p><span class="fontawesome-user"></span><input type="text" name="user"
value="Username" onBlur="if(this.value == '') this.value = 'Username'" onFocus="if(this.value
== 'Username') this.value = ''" required></p> <!-- JS because of IE support; better:
placeholder="Username" -->
<p><span class="fontawesome-lock"></span><input type="password" name="pass"
value="Password" onBlur="if(this.value == '') this.value = 'Password'" onFocus="if(this.value ==
'Password') this.value = ''" required></p> <!-- JS because of IE support; better:
placeholder="Password" --> <p><input type="submit" name="sub" value="Login"></p>
</fieldset>
        </form>
      </div> <!-- end login -->
    </div>

<div class="bottom"> <h3><a href="../index.php">SUN RISE HOMEPAGE</a></h3></div>
</body>
</html>
<?php
   include('db.php');
   if($_SERVER["REQUEST_METHOD"] == "POST") {
      // username and password sent from form
      $myusername = mysqli_real_escape_string($con,$_POST['user']);
      $mypassword = mysqli_real_escape_string($con,$_POST['pass']);
      $sql = "SELECT id FROM login WHERE usname = '$myusername' and pass =
'$mypassword'";
      $result = mysqli_query($con,$sql);
      $row = mysqli_fetch_array($result,MYSQLI_ASSOC);
    $active = $row['active'];
    $count = mysqli_num_rows($result);



// If result matched $myusername and $mypassword, table row must be 1 row
    if($count == 1) {
       $_SESSION['user'] = $myusername;
       header("location: home.php");
    }else {
       echo '<script>alert("Your Login Name or Password is invalid") </script>' ;
    }
  }
?>
```

SIGN UP AND LOGIN:

```
<?php
include('db.php')
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
      <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>RESERVATION SUNRISE HOTEL</title>
      <!-- Bootstrap Styles-->
    <link href="assets/css/bootstrap.css" rel="stylesheet" />
     <!-- FontAwesome Styles-->
    <link href="assets/css/font-awesome.css" rel="stylesheet" />
       <!-- Custom Styles-->
    <link href="assets/css/custom-styles.css" rel="stylesheet" />
     <!-- Google Fonts-->
   <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet' type='text/css' />
   </head>
                    <body>
     <div id="wrapper">
      <nav class="navbar-default navbar-side" role="navigation">
      <div class="sidebar-collapse">
      <ul class="nav" id="main-menu">
       <li>
       <a href="../index.php"><i class="fa fa-home"></i> Homepage</a>
        </li>
                 </ul>
                    </div>
                 </nav>
                 <div id="page-wrapper" >
                    <div id="page-inner">
                  <div class="row">
                          <div class="col-md-12">

          <h1 class="page-header">
                              RESERVATION <small></small>
                           </h1>
                        </div>
                     </div>
                    <div class="row">
                       <div class="col-md-5 col-sm-5">
```

```
  <div class="panel panel-primary">
                    <div class="panel-heading">
                        PERSONAL INFORMATION


                    </div>
                    <div class="panel-body">
                      <form name="form" method="post">
                        <div class="form-group">
                                    <label>Title*</label>
                                    <select name="title" class="form-control" required >
                                <option value selected ></option>
                                       <option value="Dr.">Dr.</option>
                                       <option value="Miss.">Miss.</option>
                                       <option value="Mr.">Mr.</option>
                                       <option value="Mrs.">Mrs.</option>
                                <option value="Prof.">Prof.</option>
                                <option value="Rev .">Rev .</option>
                                <option value="Rev . Fr">Rev . Fr .</option>
                             </select>
                         </div>
                       <div class="form-group">
                          <label>First Name</label>
                           <input name="fname" class="form-control" required>
                                 </div> <div class="form-group">
                                    <label>Last Name</label>
                                    <input name="lname" class="form-control" required>
                                     </div> <div class="form-group">
                                    <label>Email</label>
                                    <input name="email" type="email" class="form-control" required>
                          </div><div class="form-group">
                                    <label>Nationality*</label>
                                    <label class="radio-inline">
                                       <input type="radio" name="nation" value="Indian " checked="">
Indian







</label>
                                    <label class="radio-inline">
                                       <input type="radio" name="nation" value="Non Indian ">Non
Indian
                                    </label>
                           </div>
                      <?php

$countries = array("India");
?>
<div class="form-group">
 <label>Passport Country*</label>
<select name="country" class="form-control" required>
            <option value selected ></option> <?php
                                foreach($countries as $key => $value):
echo '<option value="'.$value.'">'.$value.'</option>'; //close your tags!!endforeach;
```

```
?> </select>
    </div>
    <div class="form-group">
    <label>Phone Number</label>
      <input name="phone" type ="text" class="form-control" required>
    </div>
    </div>
    </div>
      </div> <div class="row">
                <div class="col-md-6 col-sm-6">
                    <div class="panel panel-primary">
    <div class="panel-heading">
```

LOG OUT:

```
<?php
session_start();
unset($_SESSION["user"]);
header("location:index.php");
?>
```

USER FRONTEND PAGES:

- **HOTEL PAGE:**

```
<?php
session_start();
if(!isset($_SESSION["user"]))
{
 header("location:index.php");
}
?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Administrator          </title>
    <!-- Bootstrap Styles-->
    <link href="assets/css/bootstrap.css" rel="stylesheet" />
    <!-- FontAwesome Styles-->
    <link href="assets/css/font-awesome.css" rel="stylesheet" />
    <!-- Morris Chart Styles-->
    <link href="assets/js/morris/morris-0.4.3.min.css" rel="stylesheet" />
    <!-- Custom Styles-->
    <link href="assets/css/custom-styles.css" rel="stylesheet" />
    <!-- Google Fonts-->
    <link href='http://fonts.googleapis.com/css?family=Open+Sans' rel='stylesheet'
type='text/css' />
</head>
<body>
    <div id="wrapper">
        <nav class="navbar navbar-default top-navbar" role="navigation">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle" data-toggle="collapse" data-
target=".sidebar-collapse">
   <span class="icon-bar"></span>
```

```
          <span class="icon-bar"></span>
        </button>
        <a class="navbar-brand" href="home.php"> <?php echo $_SESSION["user"]; ?>
</a>
      </div> <ul class="nav navbar-top-links navbar-right">
        <li class="dropdown">
          <a class="dropdown-toggle" data-toggle="dropdown" href="#" aria-
expanded="false">
            <i class="fa fa-user fa-fw"></i> <i class="fa fa-caret-down"></i>
          </a>
          <ul class="dropdown-menu dropdown-user">
            <li><a href="usersetting.php"><i class="fa fa-user fa-fw"></i> User
Profile</a>
            </li>
            <li><a href="settings.php"><i class="fa fa-gear fa-fw"></i> Settings</a>
            </li>
            <li class="divider"></li>
            <li><a href="logout.php"><i class="fa fa-sign-out fa-fw"></i> Logout</a>
            </li>
          </ul>
          <!-- /.dropdown-user -->
        </li>
        <!-- /.dropdown -->
      </ul>
    </nav>
    <!--/. NAV TOP  -->
    <nav class="navbar-default navbar-side" role="navigation">
      <div class="sidebar-collapse">
        <ul class="nav" id="main-menu">
          <li><a class="active-menu" href="home.php"><i class="fa fa-dashboard"></i>
Status</a>
          </li>
          <li>
      <a href="messages.php"><i class="fa fa-desktop"></i> News Letters</a>
          </li>
    <li>
     <a href="roombook.php"><i class="fa fa-bar-chart-o"></i> Room Booking</a>
          </li>
          <li>
```

```
<a href="payment.php"><i class="fa fa-qrcode"></i> Payment</a>
                    </li>
                    <li>
                        <a href="profit.php"><i class="fa fa-qrcode"></i> Profit</a>
                    </li>
                    <li>
    <a href="logout.php"><i class="fa fa-sign-out fa-fw"></i> Logout</a>
                    </li>
                   </ul>
```

ADMIN FRONTEND PAGES:

- **ROOMBOOK PAGE:**

```
<?php
session_start();
if(!isset($_SESSION["user"]))
{
 header("location:index.php");
}
?>

<?php
if(!isset($_GET["rid"]))
{ header("location:index.php");
}
else {$curdate=date("Y/m/d");
        include ('db.php');
$id = $_GET['rid'];
$sql ="Select * from roombook where id = '$id'";
$re = mysqli_query($con,$sql);
while($row=mysqli_fetch_array($re))
{$title = $row['Title'];
$fname = $row['FName'];
$lname = $row['LName'];
$email = $row['Email'];
$nat = $row['National'];

$country = $row['Country'];
$Phone = $row['Phone'];
troom = $row['TRoom'];
$nroom = $row['NRoom'];
$bed = $row['Bed'];
$non = $row['NRoom'];
$meal = $row['Meal'];
$cin = $row['cin'];
$cout = $row['cout'];
$sta = $row['stat'];
$days = $row['nodays'];
}
}
<th>Name</th>
<th><?php echo $title.$fname.$lname; ?> </th>
</tr><tr>
  <th>Email</th><th><?php echo $email; ?> </th>
      </tr><tr>
    <th>Nationality </th>
```

```
<th><?php echo $nat; ?></th>
  </tr><tr>

  <th><?php echo $country; ?></th>
  </tr><tr><th>Phone No </th>
         <th><?php echo $Phone; ?></th>
                                  </tr><tr>
                                  <th>Type Of the Room </th>
                                  <th><?php echo $troom; ?></th>
                               </tr><tr>
                                  <th>No Of the Room </th>
                                  <th><?php echo $nroom; ?></th>
                               </tr>
                           <tr>
                                  <th>Meal Plan </th>
                                  <th><?php echo $meal; ?></th>
                               </tr><tr>
                                  <th>Bedding </th>
                                  <th><?php echo $bed; ?></th>
                               </tr><tr>
                                  <th>Check-in Date </th>
                                  <th><?php echo $cin; ?></th>
                               </tr><tr>
                                  <th>Check-out Date</th><th><?php echo $cout; ?></th>

                               </tr><tr>
                                  <th>No of days</th>
                                  <th><?php echo $days; ?></th>

                               </tr><tr>
                                  <th>Status Level</th>
                                  <th><?php echo $sta; ?></th>
                               </tr
                         </table>
                      </div>
                   </div>
                   <div class="panel-footer">
                      <form method="post">

                           <div class="form-group">
                                     <label>Select the Conformation</label>
                                                                  <select name="conf"class="form-
  control">                                   <option value selected> </option>
        <option value="Conform">Conform</option>
  </select></div>
  <input type="submit" name="co" value="Conform" class="btn btn-success">
  </form></div>
                </div></div>
  <?php$rsql ="select * from room";$rre= mysqli_query($con,$rsql);
        $r =0 ;
  $sc =0;
  $gh = 0;
  $sr = 0;
  $dr = 0;
  while($rrow=mysqli_fetch_array($rre))
  {$r = $r + 1;
  $p = $rrow['place'];
  if($s=="Superior Room" )
{
        $sc = $sc+ 1;
        }

if($s=="Guest House")

{$gh = $gh + 1;
}
if($s=="Single Room" )
{$sr = $sr + 1;
}if($s=="Deluxe Room" )
{$dr = $dr + 1;
}
}?>
<?php
$csql ="select * from payment";
$cre= mysqli_query($con,$csql);
$cr =0 ;$csc =0;
$cgh = 0;
$csr = 0;
$cdr = 0;
while($crow=mysqli_fetch_array($cre))
{$cr = $cr + 1;
$cs = $crow['troom'];
if($cs=="Superior Room" )
{$csc = $csc + 1;
}
if($cs=="Guest House" )
{$cgh = $cgh + 1;
}if($cs=="Single Room" )
$csr = $csr + 1;
        }
        if($cs=="Deluxe Room" )
        {$cdr = $cdr + 1;}
```

PAYMENT PAGE :

```
<?php
session_start();
if(!isset($_SESSION["user"]))
{
 header("location:index.php");

        }
    ?>
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
      <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>SUNRISE HOTEL</title>
        <!-- Bootstrap Styles-->
    <link href="assets/css/bootstrap.css" rel="stylesheet" />
     <!-- FontAwesome Styles-->
    <link href="assets/css/font-awesome.css" rel="stylesheet" />
     <!-- Morris Chart Styles-->
       <link href="assets/js/dataTables/dataTables.bootstrap.css" rel="stylesheet" />
</head>
<body>
```

```
             <div id="wrapper">
                    <nav class="navbar navbar-default top-navbar" role="navigation">
                      <div class="navbar-header">
                        <button type="button" class="navbar-toggle" data-toggle="collapse" data-
               target=".sidebar-collapse">
                          <span class="sr-only">Toggle navigation</span>
                          <span class="icon-bar"></span>
                          <span class="icon-bar"></span>
                          <span class="icon-bar"></span>
                        </button>
                        <a class="navbar-brand" href="home.php"><?php echo $_SESSION["user"]; ?>
               </a>
                      </div>
                      <ul class="nav navbar-top-links navbar-right">
                        <li class="dropdown">
                          <a class="dropdown-toggle" data-toggle="dropdown" href="#" aria-
               expanded="false">
                            <i class="fa fa-user fa-fw"></i> <i class="fa fa-caret-down"></i>
                          </a>
                          <ul class="dropdown-menu dropdown-user">
                            <li><a href="usersetting.php"><i class="fa fa-user fa-fw"></i> User
               Profile</a>
                </li>
                <li><a href="settings.php"><i class="fa fa-gear fa-fw"></i> Settings</a>
                 </li>
                 <li class="divider"></li>
                  <li><a href="logout.php"><i class="fa fa-sign-out fa-fw"></i> Logout</a>
                   </li>
                  </ul>
                    <!-- /.dropdown-user -->
                     </li>
                     <!-- /.dropdown -->
                      </ul>
                    </nav><div id="page-wrapper" >
                <div id="page-inner">
                          <div class="row">

<div class="col-md-12">

               <h1 class="page-header">

             Payment Details<small> </small>
            </h1>
                         </div>
                       </div>
                        <!-- /. ROW  -->
                      <div class="row">
                       <div class="col-md-12">
                        <!-- Advanced Tables -->
                        <div class="panel panel-default">
                         <div class="panel-body">
                          <div class="table-responsive">
                         <table class="table table-striped table-bordered table-hover" id="dataTables-
               example">
                            <thead>
                            <tr>
                            <th>Name</th>
               <th>Room type</th
```

```
<th>Bed Type</th>
   <th>Check in</th>
<th>Check out</th>
<th>No of Room</th>

<th>Meal Type</th>
<th>Room Rent</th>
<th>Bed Rent</th>
<th>Meals </th>
<th>Gr.Total</th>
<th>Print</th>
</tr>
</thead>
<tbody>
<?php
include ('db.php');
$sql="select * from payment";
$re = mysqli_query($con,$sql);
while($row = mysqli_fetch_array($re))
{$id = $row['id'];
if($id % 2 ==1 )
{
echo"<tr class='gradeC'>
<td>".$row['title']." ".$row['fname']." ".$row['lname']."</td>
<td>".$row['troom']."</td>
<td>".$row['tbed']."</td>
<td>".$row['cin']."</td>
<td>".$row['cout']."</td>
<td>".$row['nroom']."</td>
<td>".$row['meal']."</td>
<td>".$row['ttot']."</td>
<td>".$row['mepr']."</td>
<td>".$row['btot']."</td>
<td>".$row['fintot']."</td>
<td><a href=print.php?pid=".$id ." <button class='btn btn-primary'> <i class='fa fa-print'
></i> Print</button></td>
</tr>";
}
else
{
echo"<tr class='gradeU'>

<td>".$row['title']." ".$row['fname']." ".$row['lname']."</td>
<td>".$row['troom']."</td>
<td>".$row['tbed']."</td>

<td>".$row['cin']."</td>
<td>".$row['cout']."</td>
<td>".$row['nroom']."</td>
<td>".$row['meal']."</td>
<td>".$row['ttot']."</td>
<td>".$row['mepr']."</td>
<td>".$row['btot']."</td>
<td>".$row['fintot']."</td>
<td><a href=print.php?pid=".$id ." <button class='btn btn-primary'> <i class='fa fa-print' ></i>
Print</button></td>
</tr>";
}
}
?>
 </tbody>
  </table>
 </div>
    </div>
</body>
</html>
```

5.3: Queries

1. LOGIN QUERY:

```
$con=mysqli_connect("localhost","root","","hotel");
$check="SELECT * FROM roombook WHERE email = '$_POST[email]'";
rs = mysqli_query($con,$check);
$data = mysqli_fetch_array($rs, MYSQLI_NUM);
if($data[0] > 1) {echo "<script type='text/javascript'> alert('User Already in
Exists')</script>";
}
else{$new ="Not Conform";$newUser="INSERT INTO `roombook`(`Title`, `FName`,
`LName`, `Email`, `National`, `Country`, `Phone`, `TRoom`, `Bed`, `NRoom`, `Meal`, `cin`,
`cout`,`stat`,`nodays`) VALUES
('$_POST[title]','$_POST[fname]','$_POST[lname]','$_POST[email]','$_POST[nation]','$_P
OST[country]','$_POST[phone]','$_POST[troom]','$_POST[bed]','$_POST[nroom]','$_POS
T[meal]','$_POST[cin]','$_POST[cout]','$new',datediff('$_POST[cout]','$_POST[cin]'))";
if (mysqli_query($con,$newUser)) {echo "<script type='text/javascript'> alert('Your
Booking application has been sent')</script>";
}else{ "<script type='text/javascript'> alert('Error adding user in database')</script>";
}

$msg="Your code is correct";
}}?>
```

2. ROOMBOOK QUERY:

```
$check="SELECT * FROM roombook WHERE email = '$_POST[email]'";
$rs = mysqli_query($con,$check);
$data = mysqli_fetch_array($rs, MYSQLI_NUM);
if($data[0] > 1) {
echo "<script type='text/javascript'> alert('User Already in Exists')</script>";
}
else
{
$new ="Not Conform";
$newUser="INSERT INTO `roombook`(`Title`, `FName`, `LName`, `Email`, `National`,
`Country`, `Phone`, `TRoom`, `Bed`, `NRoom`, `Meal`, `cin`, `cout`,`stat`,`nodays`)
VALUES
('$_POST[title]','$_POST[fname]','$_POST[lname]','$_POST[email]','$_POST[nation]','$_P
OST[country]','$_POST[phone]','$_POST[troom]','$_POST[bed]','$_POST[nroom]','$_POS
T[meal]','$_POST[cin]','$_POST[cout]','$new',datediff('$_POST[cout]','$_POST[cin]'))";
if (mysqli_query($con,$newUser))
{
echo "<script type='text/javascript'> alert('Your Booking application has been
sent')</script>";
}
else
{
```

```
echo "<script type='text/javascript'> alert('Error adding user in database')</script>";
}
$msg="Your code is correct";
}
}
?>
```

3. PAYMENT QUERY:

```
$sql="select * from payment";
$re = mysqli_query($con,$sql);
while($row = mysqli_fetch_array($re))
{
$id = $row['id'];
if($id % 2 ==1 ) {
        echo"<tr class='gradeC'><td>".$row['title']." ".$row['fname']." ".$row['lname']."</td>
                <td>".$row['troom']."</td>
                        <td>".$row['tbed']."</td><td>".$row['cin']."</td>
                                <td>".$row['cout']."</td>
                                        <td>".$row['nroom']."</td>
        <td>".$row['meal']."</td><td>".$row['ttot']."</td>
                <td>".$row['mepr']."</td>
                        <td>".$row['btot']."</td>
                                <td>".$row['fintot']."</td>
        <td><a href=print.php?pid=".$id ." <button class='btn btn-primary'> <i class='fa fa-
print' ></i> Print</button></td>
        </tr>";}

else{echo"<tr class='gradeU'>
        <td>".$row['title']." ".$row['fname']." ".$row['lname']."</td>
        <td>".$row['troom']."</td>
                        <td>".$row['tbed']."</td>
        <td>".$row['cin']."</td>
        <td>".$row['cout']."</td><td>".$row['nroom']."</td>
                <td>".$row['meal']."</td>
<td>".$row['ttot']."</td>
                <td>".$row['mepr']."</td>
                        <td>".$row['btot']."</td>
                                <td>".$row['fintot']."</td>
                                        <td><a href=print.php?pid=".$id ." <button class='btn
btn-primary'> <i class='fa fa-print' ></i> Print</button></td>
                                                </tr>";
}
}
```

RESULTS

6.1 Snapshots of the project and description

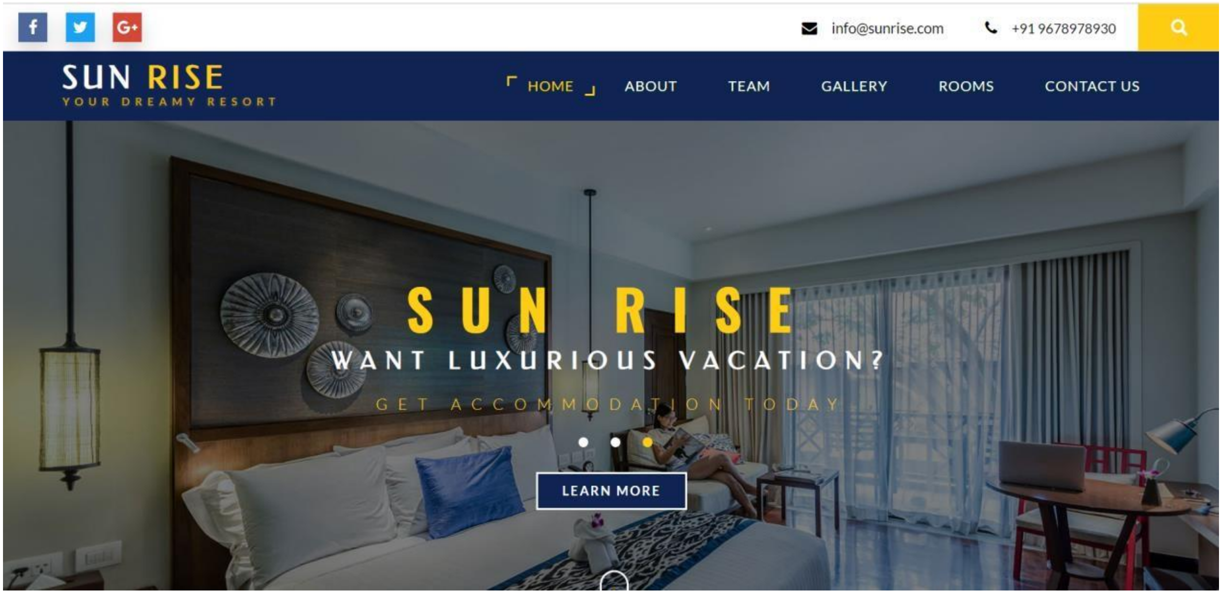
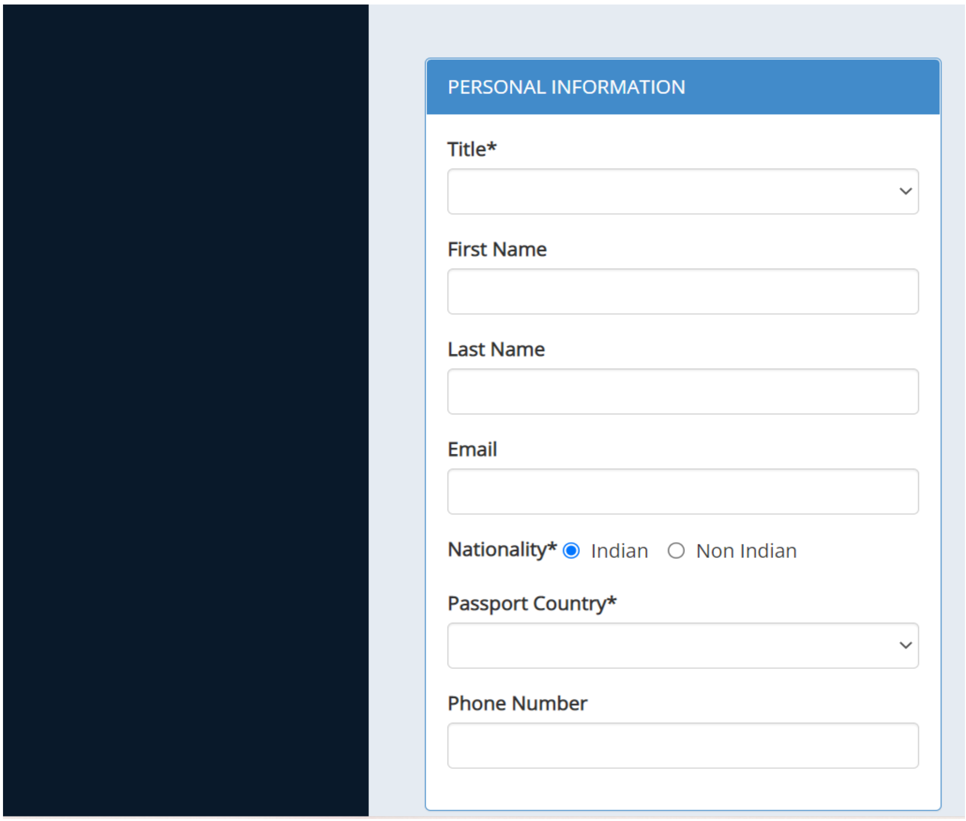


Fig 6.1: HOME PAGE

In the figure 6.1, the home page, the navbar is present from where we can toggle between the tabs.



PERSONAL INFORMATION

Title*

First Name

Last Name

Email

Nationality* ● Indian ○ Non Indian

Passport Country*

Phone Number

Fig 6.2: USER LOGIN PAGE

In the figure 6.2, the login page, user is allowed to login by giving the correct credentials in case of wrong email id or password entered, a alert message will be displayed “Incorrect email id or password”.

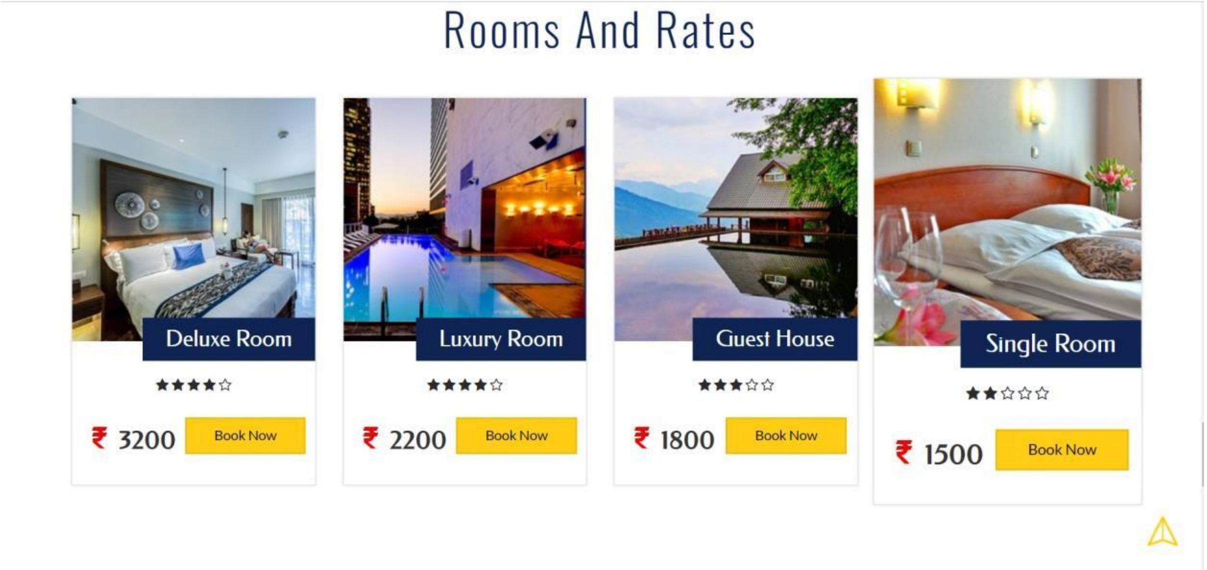


Fig 6.3: ROOM BOOKING PAGE

In the figure 6.3, the Room booking page, the user can select the room of his/her choice by selecting the book now button given below each Room description

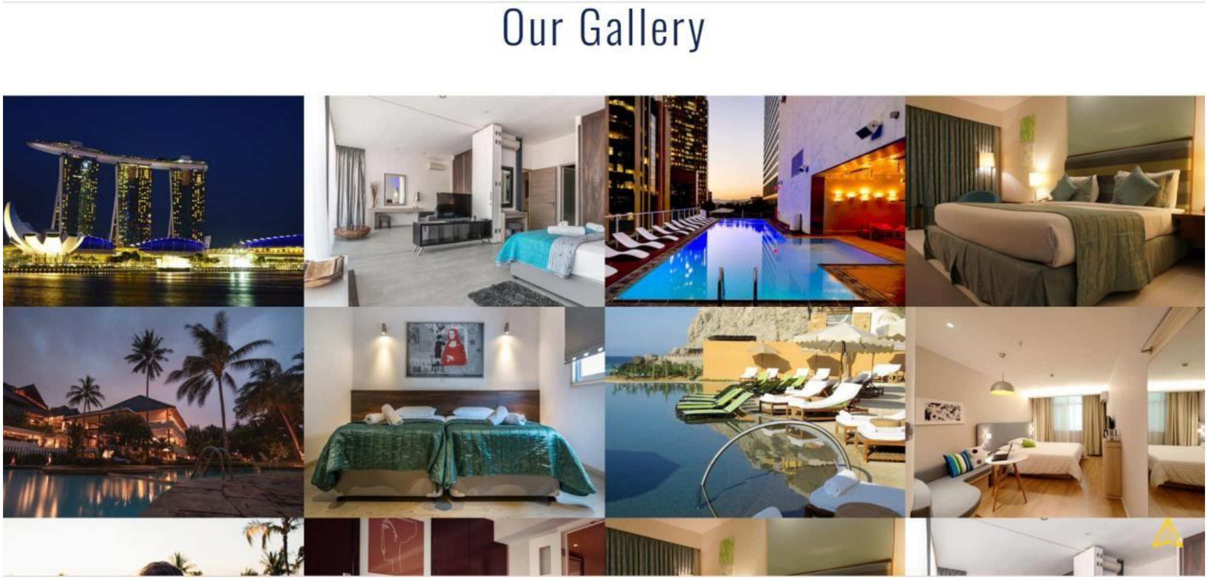
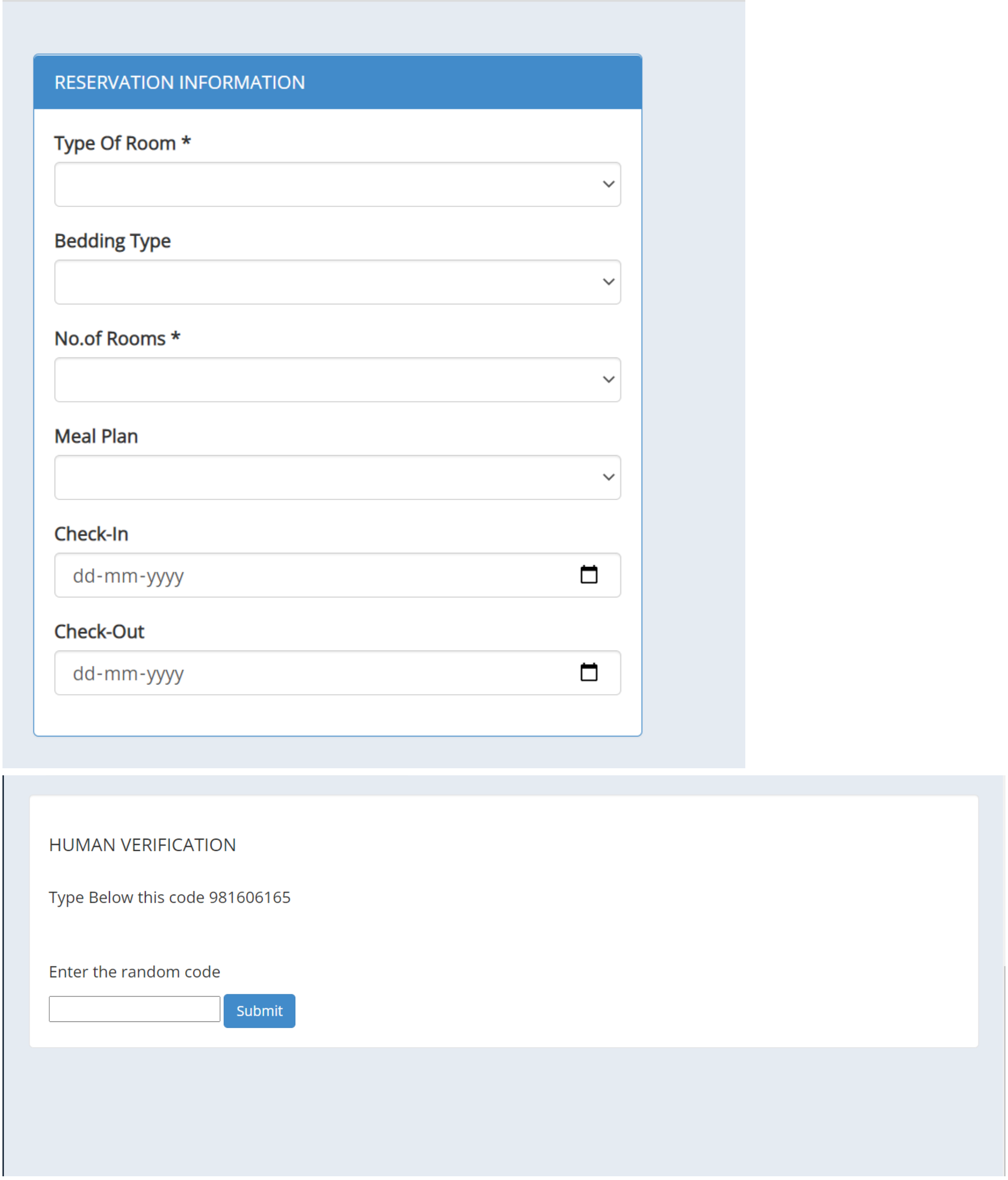


Fig 6.4: GALLERY PAGE

In the figure 6.4, the Gallery page, the user is given the option to choose the room according to his/her choice with respect to the room name and price.



RESERVATION INFORMATION

Type Of Room *

Bedding Type

No.of Rooms *

Meal Plan

Check-In

dd-mm-yyyy

Check-Out

dd-mm-yyyy

HUMAN VERIFICATION

Type Below this code 981606165

Enter the random code

Fig 6.5: RESERVATION PAGE

In figure 6.5, The Reservation page, the user is given the option to Reserve a room based on his/her choice of room.

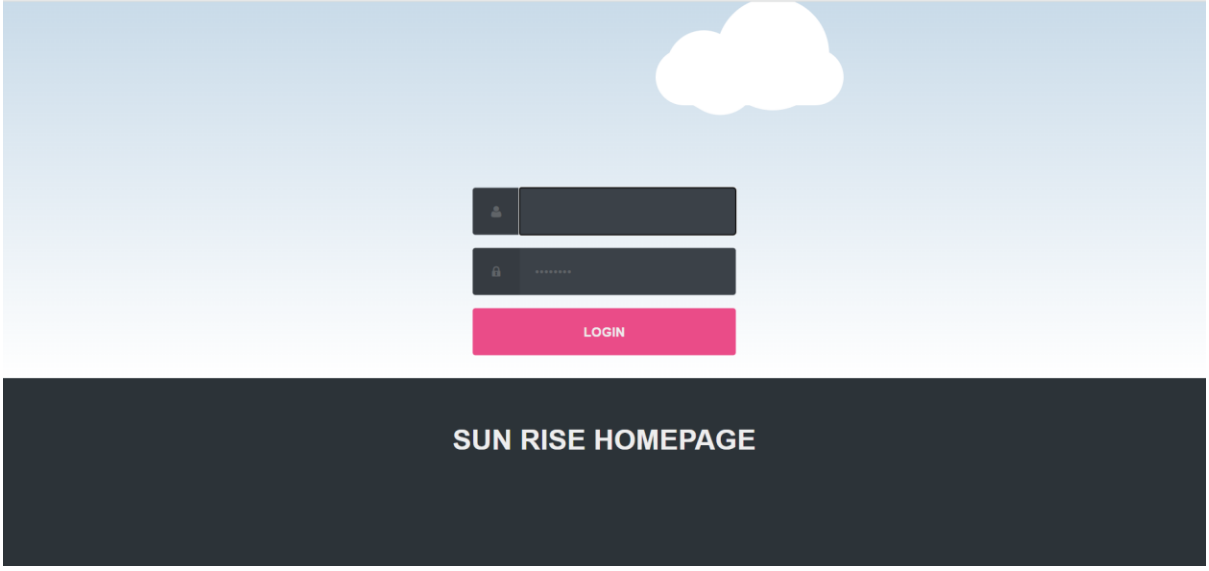
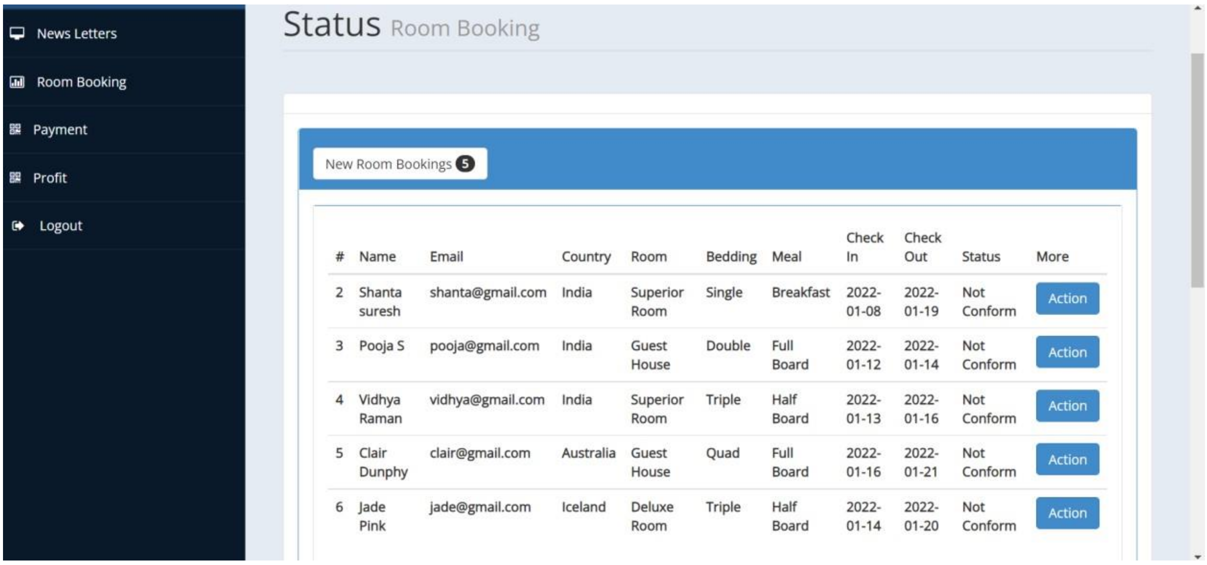


Fig 6.6: ADMIN LOGIN PAGE

In the figure 6.6, the admin login page, the admin can login through the Username and password allotted to him/her



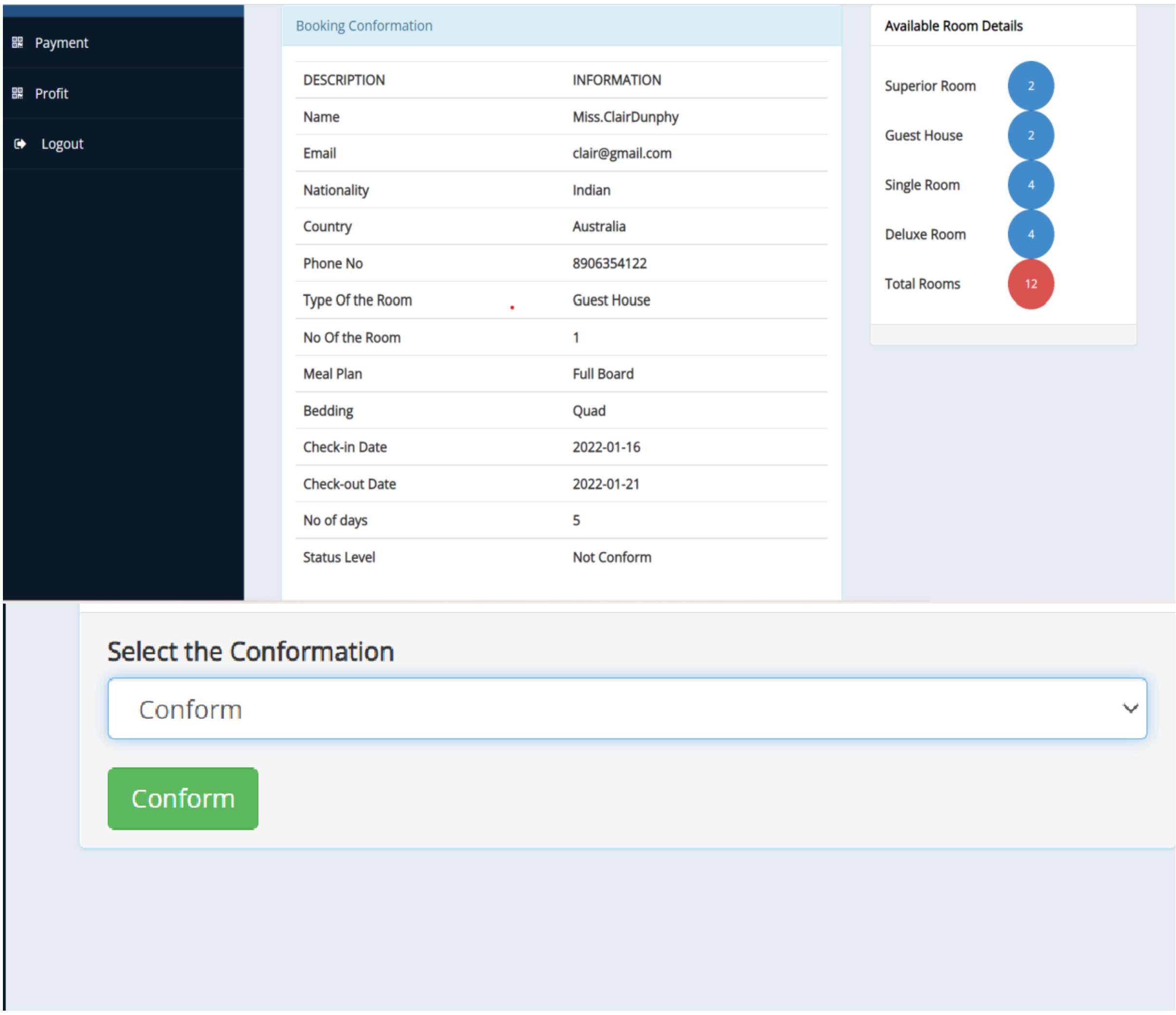
Status Room Booking

New Room Bookings 5

#	Name	Email	Country	Room	Bedding	Meal	Check In	Check Out	Status	More
2	Shanta suresh	shanta@gmail.com	India	Superior Room	Single	Breakfast	2022-01-08	2022-01-19	Not Conform	Action
3	Pooja S	pooja@gmail.com	India	Guest House	Double	Full Board	2022-01-12	2022-01-14	Not Conform	Action
4	Vidhya Raman	vidhya@gmail.com	India	Superior Room	Triple	Half Board	2022-01-13	2022-01-16	Not Conform	Action
5	Clair Dunphy	clair@gmail.com	Australia	Guest House	Quad	Full Board	2022-01-16	2022-01-21	Not Conform	Action
6	Jade Pink	jade@gmail.com	Iceland	Deluxe Room	Triple	Half Board	2022-01-14	2022-01-20	Not Conform	Action

Fig 6.7: STATUS ROOM BOOKING PAGE

In the figure 6.7, the status room booking page, the admin can take see the status of all reservation and can take action of payment to conform them.



Booking Conformation

DESCRIPTION	INFORMATION
Name	Miss.ClairDunphy
Email	clair@gmail.com
Nationality	Indian
Country	Australia
Phone No	8906354122
Type Of the Room	Guest House
No Of the Room	1
Meal Plan	Full Board
Bedding	Quad
Check-in Date	2022-01-16
Check-out Date	2022-01-21
No of days	5
Status Level	Not Conform

Available Room Details

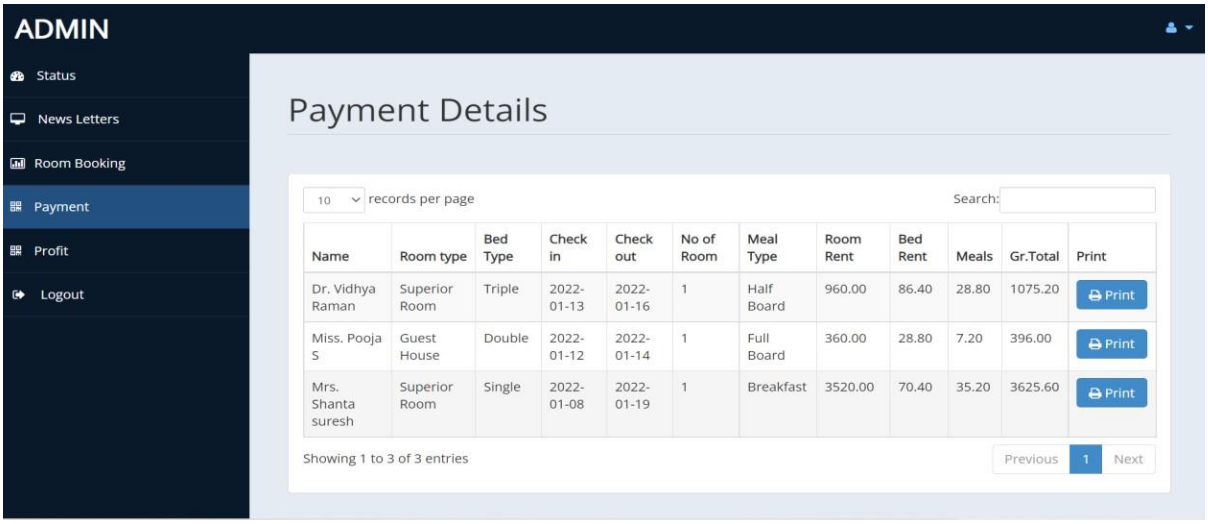
Superior Room	2
Guest House	2
Single Room	4
Deluxe Room	4
Total Rooms	12

Select the Conformation

Conform

Fig 6.8: BOOKING CONFORMATION PAGE

In the figure 6.8, the booking conformation page, the admin can conform the booking done by the user after he receives the payment.



Payment Details

10 records per page

Search:

Name	Room type	Bed Type	Check in	Check out	No of Room	Meal Type	Room Rent	Bed Rent	Meals	Gr.Total	Print
Dr. Vidhya Raman	Superior Room	Triple	2022-01-13	2022-01-16	1	Half Board	960.00	86.40	28.80	1075.20	Print
Miss. Pooja S	Guest House	Double	2022-01-12	2022-01-14	1	Full Board	360.00	28.80	7.20	396.00	Print
Mrs. Shanta suresh	Superior Room	Single	2022-01-08	2022-01-19	1	Breakfast	3520.00	70.40	35.20	3625.60	Print

Showing 1 to 3 of 3 entries

Fig 6.9: PAYMENT DETAILS PAGE

Chapter 7

Conclusion

In this project, hotel management system is created using MY SQL in the backend and HTML, CSS and Java Script in the frontend. And they are connected using PHP and Java Script codes. SQL is used for storing the data in the backend (DBMS). HTML is used for creating frontend pages. PHP code is used to link/connect the backend and frontend together. This system allows the user to register and login into the system, see the types of rooms and choose the room from the home page. The booked room are based on unique id that is generated automatically when a user sign-up. The payment can be made by the user to the cashier. The project has a very vast scope in future. The project is flexible in terms of expansion and has advantages like: 1 In Hotel Management System, the user will be able to choose the desired room and desired dishes along with the price displayed with each room . 2 The web based room management system has offered an advantage to both customer and Hotel Management Company to effectively manage the business and satisfy customer' s needs at the click of a button. 3 Also easy implementation environment. 4 This system therefore helps the admin with the easy maintenance of staff and orders through the portal. 5 This system provides good GUI support

REFERENCES

Textbooks:

- Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7 th Edition, 2017, Pearson.
- Database management systems, Ramakrishnanand Gehrke, 3 rdedition, 2014, Mcgraw hill.

Websites:

- W3Schools Online Web Tutorials
- Bootstrap · The most popular HTML, CSS, and JS library in the world. (getbootstrap.com)