**ABSTRACT**

**HMS** is a type of properly management system that facilitates the management of hotel operations and functions; main operations such as front office, sales, planning, and accounting. The efficiency of any hotel is determined by the method used to obtain and prevent information from customers' personal data for use in the hotel's various services. It has been a complex and difficult operation to manage their outcome revealed, particularly when information flow is consistent. The details of hotel are store into the Hotel tables respective with all tables. Each entity (Customers, Services, Booking, Rooms, Hotel ) contains primary key and unique keys. The entity Services, Booking has binded with Hotel, Rooms entities with foreign keys. The project, Hotel Management System is a web-based application that allows the hotel manager to handle all hotel activities online. The hotel manager is a very busy person and does not have the time to sit and manage the entire activities manually on paper. This application gives him the power and flexibility to manage the entire system from a single online system. Hotel management project provides room booking, staff management and other necessary hotel management features.

**1. INTRODUCTION**

**1.1 Introduction to the topic**

The project Online Hotel Management System is a web-based application that allows the Hotel Manager & Owner to handle all hotel activities online easily and safely. Using Interactive GUI anyone can easily learn to use the complete system.

Using this Hotel Manager doesn’t have to sit and manage the entire activities on paper. And at the same time Owner of the Hotel will feel comfortable keeping a check on the hotel easily from anywhere around the world. This System will give them power and flexibility to manage the entire system from a single online portal.

Hotel Management System provides room booking, staff management, and bill generation features. The system will be so simple and attractive which will make the customer comfortable to use and choose their ideal room. The system allows the Owner to check the Progress of the hotel from interactive Graphs and he will be notified of each new change made in System.

The system allows the manager to keep track of available rooms in the system and even maintain staff details like their hours worked and salary. Customers can view and book an available room online and the system will automatically generate the bill according to the number of days the type of room is booked.

**Modules of many Hotels Software are:**

* Reservations includes pre-booking, and booking functions.
* Profiles for classifying what guest, privileges, and memberships.
* Groups and blocks for large groups of guests with different privileges.
* Rate and inventory control monitoring for rooms, amenities and materials.
* Administration assigns the access control for each hotel department, and other staffs.
* Reporting displays custom reports for various outputs in the automated hotel operations.

**1.2 Problem Statement**

The aim of hotel management system is to handle all aspects of the hotel's information and booking system.

The Hotel Management System, referred to as HMS, is an application that will help users better utilize rooms used by employees and other guests. HMS helps users manage guest flows by affording them the ability to easily check guests in, check them out, and generate stay reports, among other things.

The project, Hotel Management System is a web-based application that allows the hotel manager to handle all hotel activities online. The hotel manager is a very busy person and does not have the time to sit and manage the entire activities manually on paper. This application gives him the power and flexibility to manage the entire system from a single online

system. Hotel management project provides room booking, staff management and other necessary hotel management features.

The system allows the manager to post available rooms in the system. Customers can view and book room online. Admin has the power of either approving or disapproving the customer’s booking request. Other hotel services can also be viewed by the customers and can book them too. The system is hence useful for both customers and managers to portable manage the hotel activities. Sometime it happens that the rooms get booked soon when one visits the place therefore user can make advance booking using this system. It saves user time in searching a room. The system is useful as it calculates an exact cost of rooms for requested number of days. It saves organization resources and expenses. This system is effective and saves time and cost of users.

The next we talk about the rooms of the hotel. There are different types of rooms like standard rooms, guest rooms, single bed rooms, double bed rooms, VIP rooms etc. The cost of the rooms will differ according to the type of the rooms.

Modules of many Hotels Software are:

* Reservations includes pre-booking, and booking functions.
* Profiles for classifying what guest, privileges, and memberships.
* Groups and blocks for large groups of guests with different privileges.
* Rate and inventory control monitoring for rooms, amenities and materials.
* Administration assigns the access control for each hotel department, and other staffs.
* Reporting displays custom reports for various outputs in the automated hotel operations.

**1.3 Objectives**

**GENERAL OBJECTIVES**

**1.** The basic objective of **HOTEL MANAGEMENT SYSTEM** is to generalize and simplify the monthly or day to day activities of Hotel like Room activities.

**2**. The activities include Check in of New Customer, check out of customer, assigning a room according to customer requirement, and finally compute the bill etc.

**3.** The main purpose of hotel management is to ensure a constant influx of visitors and guests to the hotel throughout the year.

**SPECIFIC OBJECTIVES**

**1.** The hotel management system provides the quality service to the end user.

**2.** This report shows all essential data for one working day: revenue, room occupancy statistics, and payments.

**3.** It benefits the visiting guests through marketing initiatives.

**2. DATABASE DESIGN**

**2.1 List of Attributes, entities and relationship**

**The relational database schema for hotel management database is as follows**

1.Hotel (H\_id, H\_name,H\_email, H\_address, H\_contact)

2.Rooms (R\_type, R\_number, R\_floor)

3.Facilities\_available (F\_id , F\_name, F\_cost)

4.Booking (B\_id , B\_date, B\_day, R\_type)

5.Payment (P\_id, P\_type, P\_transno)

6.Customer (cid , cname , address, mobile number, gender, DOB, age, email-id)

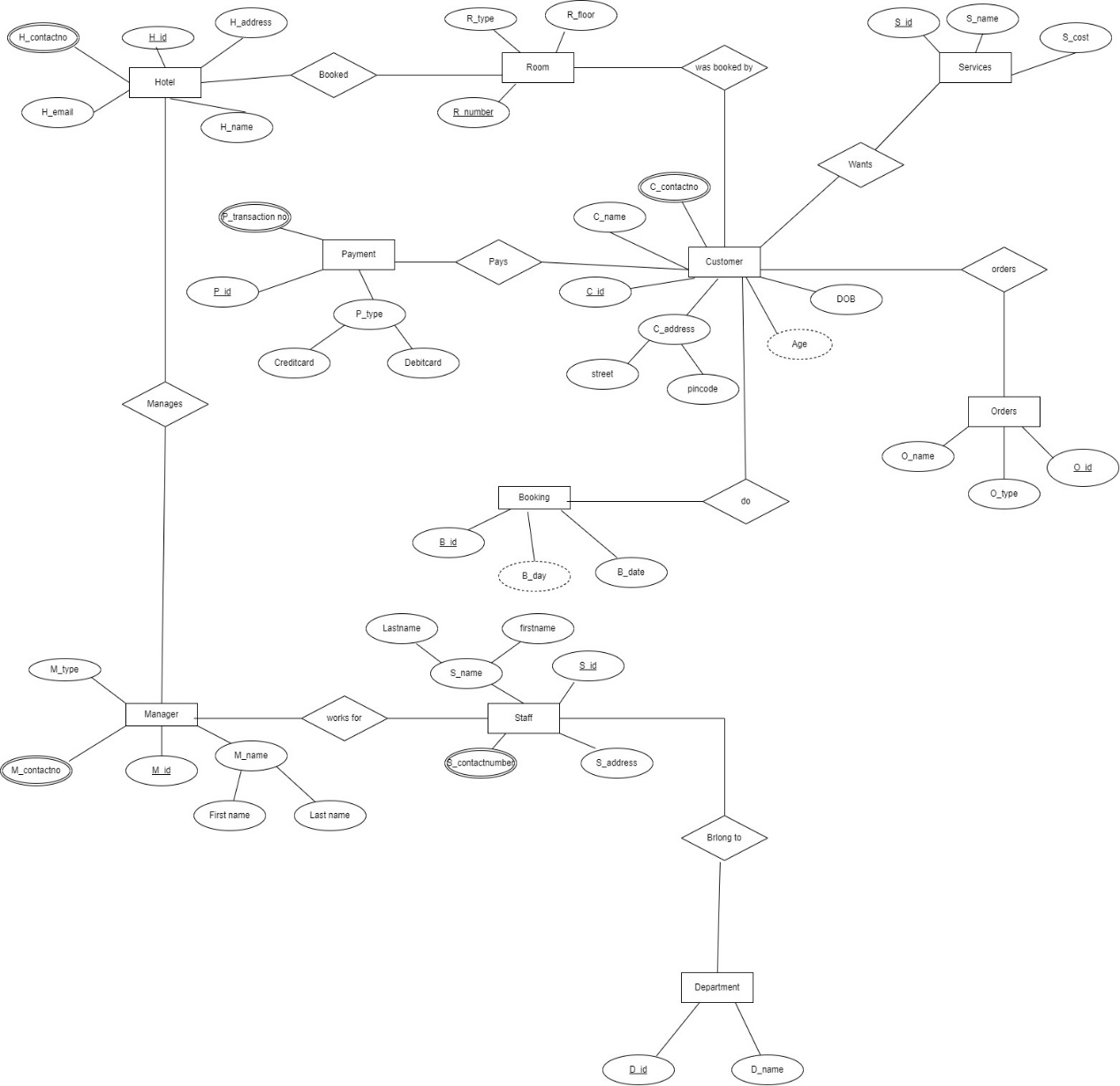
7. Department (DID ,DName)

8.Manager (M\_id,M\_Name,DName,M\_address,M\_mobile)

9.Staff (sid, sname, DName, snumber, saddress, sgender

10.Orders (O\_ID,O\_Name,O\_Type,O\_Quantity)

**2.2 E-R DIAGRAM**



**3. RELATIONAL MODEL**

**3.1 Database Languages**

● A DBMS has appropriate languages and interfaces to express database queries and updates. ● Database languages can be used to read, store and update the data in the database.

**1. Data Definition Language**

● DDL stands for Data Definition Language. It is used to define database structure or pattern.

● It is used to create schema, tables, indexes, constraints, etc. in the database.

● Using the DDL statements, you can create the skeleton of the database.

● Data definition language is used to store the information of metadata like the number of tables and schemas, their names, indexes, columns in each table, constraints, etc.

**Here are some tasks that come under DDL:**

● **Create:** It is used to create objects in the database.

**Syntax:** CREATE TABLE table\_name (column1 datatype, column 2 datatype, column 3 datatype, .... );

● **Alter:** It is used to alter the structure of the database.

**Syntax:** ALTER TABLE table\_name MODIFY COLUMN column\_name datatype;

● **Drop:** It is used to delete objects from the database. Syntax: DROP TABLE table\_name;

● **Truncate:** It is used to remove all records from a table.

**Syntax:** TRUNCATE TABLE table\_name;

● **Rename:** It is used to rename an object.

**Syntax:** ALTER TABLE old\_table\_name RENAME TO new\_table\_name;

● **Comment:** It is used to comment on the data dictionary.

**Syntax:** Single line comments start with -- and multi-line comments starts with /\* and ends with \*/

**2. Data Manipulation Language:**

DML stands for Data Manipulation Language. It is used for accessing and manipulating data in a database. It handles user requests.

**Here are some tasks that come under DML:**

● **Select:** It is used to retrieve data from a database. Syntax: SELECT column1, column2, … FROM table\_name;

● **Insert:** It is used to insert data into a table. Syntax: INSERT INTO table\_name (column1, column2, column3, ...) VALUES (value1, value2, value3, ...);

● **Update:** It is used to update existing data within a table. Syntax: UPDATE table\_name SET column1 = value1, column2 = value2, ... WHERE condition;

● **Delete:** It is used to delete all records from a table. Syntax: DELETE FROM table\_name WHERE condition;

● **Merge:** It performs UPSERT operation, i.e., insert or update operations. Syntax: MERGE target\_table USING source\_table ON merge\_condition WHEN MATCHED THEN update\_statement WHEN NOT MATCHED THEN insert\_statement WHEN NOT MATCHED BY SOURCE THEN DELETE;

● **Call:** It is used to call a structured query language or a Java subprogram. Syntax: CALL procedure name (parameter [param1, param2, ...])

**3.2 Table Description**

**1. Hotel:** This table consists of hotel name, hotel id, address of the hotel. It will give the details about the hotel.

**Constraint:** Hotel id will be unique in this table.

**2. Rooms:** This table contains the information about the rooms and its availability. There will be different types of rooms like Single bed, double bed, standard rooms and VIP rooms. The customer can able to book the rooms of his/her choice. According to the floor each room can be identified by the room number. This table also contains the details of the room like room number, room type etc.

**Constraint:** Room number will be the primary key which is unique for every room.

**3. Facilities\_available:** This table contains the information about all the facilities available in the hotel room. The facilities available are like customers landline, Wi-Fi, laundry and tv.

**Constraint:** Facilities should be available for every room.

**4. Booking:** This table contains all the information about the booking details of a customer like booking the rooms they want. For booking, the customer should have a valid email-id and mobile number.

**Constraint:** Booking id is unique for every customer as per their order.

**5. Payment:** This table contains the payment details like payment id, payment type, transaction number, payer name. The payment type will be of credit card, debit card, net banking.

**Constraint:** Payment id will be unique for each payment.

**6. Customer:** This table gives the information about customer name, id, address, mobile number, email id, gender, DOB and number of customers available in the hotel and what type of rooms they booked.

**Constraint:** customer id will be the primary key in this table.

**7. DEPARTMENT:** This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.

**Constraint:** Department name will be unique for each department

**8. Manager:** This table gives the information about manager name, id, address, mobile number, email id, DOB and which department the manager belongs to, like whether she/he belongs to food management, hotel management, service management and security management.

**Constraint:** Gender should be M or F only. Manager name and department must exist.

**9. Staff:** This table gives the information about whether the staff is working staff or non-working staff. For every department there is staff like as in food department, security department, management department and service department. According to their shift they will be working in the hotel.

**Constraint:** The department should be specific for the staff.

**10. Orders:** This table gives the information about order name, order id, order type, order quantity and in which room the order to be served.

**Constraint:** customer id and room number should exist in this table.

**3.3 Relational Database Schema**

1. **Hotel**

**Column Name Data Type Remarks**

HID int primary key,

Hname varchar (30) not null,

Hemail varchar (30) not null,

Address varchar (30) null,

Mobile varchar (10) null

**2. Rooms**

**Column Name Data Type Remarks**

R\_type char (1) not null check (R\_type='A' or R\_type='N'),

R\_number int primary key,

R\_floor int null

**3. Facilities Available**

**Column Name Data Type Remarks**

F\_id int primary key,

F\_name varchar (20) not null,

F\_cost money not null

**4. Booking**

**Column Name Data Type Remarks**

B\_id int primary key,

B\_date date not null,

R\_number int not null foreign key references Room(R\_number)

**5. Customer**

**Column Name Data Type Remarks**

CID int primary key,

CName varchar (20) not null,

Cmobile bigint not null,

age bigint null,

crating bigint not null

**6. Department**

**Column Name Data Type Remarks**

DID int primary key,

Dname varchar (20) not null

**7. Manager**

**Column Name Data Type Remarks**

M\_id int not null,

M\_name varchar (20) not null,

DID int not null foreign key references Depart (DID),

M\_gender char (1) not null check(M\_gender='M' or M\_gender='F'),

M\_Address varchar (30) null,

M\_Mobile varchar (10) null,

HID int not null foreign key references hotel (HID)

**8. Staff**

**Column Name Data Type Remarks**

s\_sid int primary key,

s\_name varchar (30) not null,

DID int not null foreign key references Depart (DID),

s\_mobile varchar (10) null

**3.4 Relational Queries**

/\* create a table hotel\*/

create table hotel

(

HID int primary key,

Hname varchar (30) not null,

Hemail varchar (30) not null,

Address varchar (30) null,

Mobile varchar (10) null

)

insert into hotel values(101,'novotel','novotel@gmail.com','vizag',9998887771)

insert into hotel values(102,'regency','regency@gmail.com','chennai',9998887772)

insert into hotel values(103,'carlton','carlton@gmail.com','tiruapthi',9998887773)

insert into hotel values(104,'taj','taj@gmail.com','mumbai',9998887774)

insert into hotel values(105,'savoy','savoy@gmail.com','chennai',9998887775)

select \* from hotel

**Output:**

**HID** **Hname**  **Hemail**  **Address Mobile**

101 novotel novotel@gmail.com vizag 9998887771

102 regency [regency@gmail.com](mailto:regency@gmail.com) chennai 9998887772

103 carlton [carlton@gmail.com](mailto:carlton@gmail.com) tiruapthi 9998887773

104 taj taj@gmail.com mumbai 9998887774

105 savoy savoy@gmail.com chennai 9998887775

106 capitol capitol@gmail.com bangalore 9998887776

107 hoteldupleix hoteldupleix@gmail.com pondicherry 9998887777

108 tajmalabar [tajmalabar@gmail.com](mailto:tajmalabar@gmail.com) kerala 9998887778

109 fortheritage [fortheritage@gmail.com](mailto:fortheritage@gmail.com) kerala 9998887779

110 aman aman@gmail.com rajasthan 9998887710

111 baghvan baghvan@gmail.com pune 9998887711

112 banjaar banjaar@gmail.com jammu 9998887712

113 ramji ramji@gmail.com calcutta 9998887713

114 diphlu diphlu@gmail.com bhopal 9998887714

115 granddragon granddragon@gmail.com lucknow 9998887715

/\*create a table Rooms\*/

create table Room

(

R\_type char (1) not null check(R\_type='A' or R\_type='N'),

R\_number int primary key,

R\_floor int null

)

insert into Room values('A',10,1)

insert into Room values('A',12,1)

insert into Room values('A',13,2)

insert into Room values('N',23,2)

insert into Room values('N',24,2)

insert into Room values('N',27,5)

select \* from Room

**Output:**

**R\_type** **R\_number**  **R\_floor**

A 10 1

A 12 1

A 13 2

A 17 4

A 19 3

A 20 3

N 23 2

N 24 2

N 26 5

N 27 5

N 30 4

N 31 3

N 34 1

N 37 2

A 40 2

/\*create a table Facilities Available\*/

create table Facilities\_available

(

F\_id int primary key,

F\_name varchar(20) not null,

F\_cost money not null

)

insert into Facilities\_available values(201,'food',2000)

insert into Facilities\_available values(202,'internet',1000)

insert into Facilities\_available values(203,'books',5000)

insert into Facilities\_available values(204,'computer',1500)

insert into Facilities\_available values(205,'laundry',500)

select \* from Facilities\_available

**Output:**

**F\_id** **F\_name** **F\_cost**

201 food 2000.00

202 internet 1000.00

203 books 5000.00

204 computer 1500.00

205 laundry 500.00

206 meeting facilities 2000.00

207 car parking 700.00

208 catering 1000.00

209 water purifier 5000.00

210 house keeping 2500.00

/\* create a table Booking\*/

create table Booking

(

B\_id int primary key,

B\_date date not null,

R\_number int not null foreign key references Room(R\_number)

insert into Booking values (301,'2022-01-15','10')

insert into Booking values (302,'2022-01-22','12')

insert into Booking values (303,'2022-03-07','13')

insert into Booking values (304,'2022-02-10','17')

insert into Booking values (305,'2022-04-11','19')

**Output:**

**B\_id** **B\_date** **R\_number**

301 2022-01-15 10

302 2022-01-22 12

303 2022-03-07 13

304 2022-02-10 17

305 2022-04-11 19

306 2022-05-17 20

307 2022-06-29 23

308 2022-01-05 24

309 2022-03-25 27

310 2022-02-03 26

311 2022-04-13 30

312 2022-05-23 31

313 2022-01-01 34

314 2022-04-04 37

315 2022-02-02 40

/\* create a table customers \*/

create table customers

(

CID int primary key,

CName varchar (20) not null,

Cmobile bigint not null,

age bigint null,

crating bigint not null

)

insert into customers values (401,'hima',9997776661,18,7)

insert into customers values (402,'harshitha',9997776662,19,7)

insert into customers values (403,'janu',9997776663,20,7)

insert into customers values (404,'poojitha',9997776664,18,6)

insert into customers values (405,'sreekar',9997776665,21,8)

insert into customers values (406,'rithu',9997776666,30,9)

select \* from customers

**Output:**

**CID** **CName** **Cmobile** **age**  **crating**

401 hima 9997776661 18 7

402 harshitha 9997776662 19 7

403 janu 9997776663 20 7

404 poojitha 9997776664 18 6

405 sreekar 9997776665 21 8

406 rithu 9997776666 30 9

407 abhi 9997776667 40 10

408 rohan 9997776668 37 10

409 arjun 9997776669 12 7

410 shetal 9997776610 10 6

411 rehman 9997776611 54 5

412 priya 9997776612 88 8

413 nitya 9997776613 78 10

414 jayanth 9997776614 68 9

415 rujith 9997776615 70 9

416 reshma 9786543211 26 9

417 deepak 7382171766 27 8

/\*create table Depart \*/

create table Depart

(

DID int primary key,

Dname varchar (20) not null

)

insert into Depart values(1,'food')

insert into Depart values (2,'housekeeping')

insert into Depart values(3,'security')

insert into Depart values(4,'service')

insert into Depart values(5,'management')

select \* from Depart

**Output:**

**DID** **Dname**

1 Hotel

2 Food

3 Security

4 Service

5 management

/\* create table manger\*/

Create table Manager

(

M\_id int not null,

M\_name varchar (20) not null,

DID int not null foreign key references Depart (DID),

M\_gender char (1) not null check(M\_gender='M' or M\_gender='F'),

M\_Address varchar (30) null,

M\_Mobile varchar (10) null,

HID int not null foreign key references hotel (HID)

)

insert into manager values(601,'shriya',1,'F','tirupathi',9998886661,101)

insert into manager values(602,'keishna',2,'M','chennai',9998886662,102)

insert into manager values(603,'meera',3,'F','kerala',9998886663,103)

insert into manager values(604,'tarun',4,'M','chennai',9998886664,104)

insert into manager values(605,'jansi',5,'F','tirupathi',9998886665,105)

**OUTPUT:**

**M\_id** **M\_name** **DID** **M\_gender** **M\_Address** **M\_Mobile**  **HID**

601 shriya 1 F tirupathi 9998886661 101

602 keishna 2 M chennai 9998886662 102

603 meera 3 F kerala 9998886663 103

604 tarun 4 M chennai 9998886664 104

607 divya 1 F mumbai 9998886667 107

608 pradeepthi 2 F hyderabad 9998886668 108

609 reshma 3 F vijayawada 9998886669 109

610 deepak 4 M kerala 9998886610 110

612 padma 1 F Bhopal 9998886612 112

613 karthik 2 M vizag 9998886613 113

614 neha 3 F tirupathi 9998886614 114

615 yeshwanth 4 M bangalore 9998886615 115

/\*create table staff \*/

create table staff

(

s\_sid int primary key,

s\_name varchar (30) not null,

DID int not null foreign key references Depart(DID) ,

s\_mobile varchar (10) null

)

insert into staff values(501,'vicky',1,8889997771)

insert into staff values(502,'ram',2,8889997772)

insert into staff values(503,'kiran',3,8889997773)

insert into staff values(504,'sunny',4,8889997774)

insert into staff values(505,'vijay',5,8889997775)

select \* from staff

**Output:**

**s\_sid** **s\_name** **DID** **s\_mobile**

501 vicky 1 8889997771

502 ram 2 8889997772

503 kiran 3 8889997773

504 sunny 4 8889997774

505 vijay 5 8889997775

506 mahi 1 8889997776

507 ravi 2 8889997777

508 arpitha 3 8889997778

509 ramya 4 8889997779

510 siri 5 8889997710

511 adhi 1 8889997711

512 keerthi 2 8889997712

513 anshu 3 8889997713

514 akrythi 4 8889997714

515 ashok 5 8889997715

**SQL QUERIES:**

**Query1: update the address of a hotel Manager whose id is 614.**

update manager set M\_Address='vijayawada' where M\_id=614

**/\*output:**

**M\_id M\_name DID M\_gender M\_Address M\_Mobile HID**

601 shriya 1 F tirupathi 9998886661 101

602 keishna 2 M chennai 9998886662 102

603 meera 3 F kerala 9998886663 103

604 tarun 4 M chennai 9998886664 104

605 jansi 5 F tirupathi 9998886665 105

606 rupa 5 F tirupathi 9998886666 106

607 divya 1 F mumbai 9998886667 107

608 pradeepthi 2 F hyderabad 9998886668 108

609 reshma 3 F vijayawada 9998886669 109

610 deepak 4 M kerala 9998886610 110

611 raj 5 M chennai 9998886611 111

612 padma 1 F Bhopal 9998886612 112

613 karthik 2 M vizag 9998886613 113

614 neha 3 F vijayawada 9998886614 114

615 yeshwanth 4 M bangalore 9998886615 115 **\*/**

**Query2: Insert a coloumn name called Blood group with data type of varchar (10) and constrain as null in customer table,manager table and staff table.**

alter table customer add BloodGroup varchar (10) null

alter table manager add BloodGroup varchar (10) null

alter table staff add BloodGroup varchar (10) null

**/\*output:**

**s\_sid s\_name DID s\_mobile BloodGroup**

501 vicky 1 8889997771 NULL

502 ram 2 8889997772 NULL

503 kiran 3 8889997773 NULL

504 sunny 4 8889997774 NULL

505 vijay 5 8889997775 NULL

506 mahi 1 8889997776 NULL

**CID CName Cmobile age crating BloodGroup**

401 hima 9997776661 18 7 NULL

402 harshitha 9997776662 19 7 NULL

403 janu 9997776663 20 7 NULL

404 poojitha 9997776664 18 6 NULL

405 sreekar 9997776665 21 8 NULL

**M\_id M\_name DID M\_gender M\_Addres M\_Mobile HID BloodGroup**

601 shriya 1 F tirupathi 9998886661 101 NULL

60 keishna 2 M chennai 9998886662 102 NULL

603 meera 3 F kerala 9998886663 103 NULL

60 tarun 4 M chennai 9998886664 104 NULL

605 jansi 5 F tirupathi 9998886665 105 NULL

606 rupa 5 F tirupathi 9998886666 106 NULL **\*/**

**Query3: Display the total cost of all facities which are avilable.**

selectsum(F\_cost)fromFacilities\_available

**/\*output**

**(Nocolumnname)**

21200.00 **\*/**

**Query4:Display the details of customer who has given rating more than 7.**

select \* from customer where crating>7

**/\* output:**

**CName Cmobile age crating CID BloodGroup**

sreekar 9997776665 21 8 405 NULL

rithu 9997776666 30 9 406 NULL

abhi 9997776667 40 10 407 NULL

rohan 9997776668 37 10 408 NULL

priya 9997776612 88 8 412 NULL

nitya 9997776613 78 10 413 NULL

jayanth 9997776614 68 9 414 NULL

rujith 9997776615 70 9 415 NULL **\*/**

**Query5: Create view as hotel\_manager whose address is tirupathi and update view and drop the view.**

create view hotel\_manager from Manager as

select M\_id, M\_name, M\_gender

from Manager where address='tirupathi'

select \* from hotel\_manager

**/\* output:**

**M\_id M\_name M\_gender**

601 shriya F

605 jansi F

606 rupa F

update manager set M\_name='dhanya' where M\_id=605

**/\* output:**

**M\_id M\_name M\_gender**

601 shriya F

605 dhanya F

606 rupa F

drop view hotel\_manager

**/\* Output:** Commands completed successfully**\*/**

**Query6: Find the total no:of customers whose age is greater than 60 and rating=9**

select count(CID) from customer where age>60 and crating=9

**/\* output:**

**(No column name)**

2  **\*/**

**Query7: Find the ID's of customer whose contact is null.**

select CID from customer where contact=null.

**/\* output:**

**CID**

**\*/**

**Query8: Find the age and names of customers whose name begin and ends with p and a and having atleast three characters.**

select c.CName, c.age from customers c where CName like 'p\_%a'

**/\* output:**

**CName age**

Poojitha 18

priya 88 **\*/**

**Query9: Find the ID's and names of customers who have given rating more than 8.**

select c.CID, c.CName from customers c where rating>8

**/\* output:**

**CID CName**

406 rithu

407 abhi

408 rohan

413 nitya

414 jayanth

415 rujith **\*/**

**Query10: Display the names of facilties that are available.**

select F\_name from Facilities\_available

**/\* output:**

**F\_name**

food

internet

books

computer

laundry

meeting facilities

car parking

catering

water purifier

house keeping **\*/**

**Query11: Display the total count of staff members.**

select count(s\_sid) as staff\_count from staff

**/\* output:**

**staff\_count**

15 **\*/**

**Query12: Display the names and mobile number of male managers.**

select m.M\_name, m.M\_Mobile from manager m where M\_gender='F'

**/\*output:**

**M\_name M\_Mobile**

keishna 9998886662

tarun 9998886664

deepak 9998886610

raj 9998886611

karthik 9998886613

yeshwanth 9998886615 **\*/**

**Query13: Display the names of departments using order by statement**.

select Dname from Depart order by Dname asc

**/\* output:**

**Dname**

food

house keeping

management

security

service  **\*/**

**Query14: Find the details of the customer whose age is between 20 and 30.**

select \* from customers where age between 20 and 30

**/\* output:**

**CID CName Cmobile age crating**

403 janu 9997776663 20 7

405 sreekar 9997776665 21 8

406 rithu 9997776666 30 9 **\*/**

**Query15: find the ID and name of managers who belong to department 5 and tirupathi.**

select M\_id, M\_name from manager where DID=5 and M\_Address='tirupathi'

**/\* output:**

**M\_id M\_name**

605 jansi

606 rupa **\*/**

**Query16: Find the total number of hotels that are in this city.**

SELECT COUNT(HID) AS num\_hotels FROM Hotel;

/\* output:

num\_hotels

15 **\*/**

**Query17: Update the contact of a particular staff.**

update staff set s\_mobile='7672044500' where s\_sid=513

**/\* output:**

**s\_sname DID s\_mobile s\_sid**

vicky 1 8889997771 501

ram 2 8889997772 502

kiran 3 8889997773 503

sunny 4 8889997774 504

vijay 5 8889997775 505

mahi 1 8889997776 506

ravi 2 8889997777 507

arpitha 3 8889997778 508

ramya 4 8889997779 509

siri 5 8889997710 510

adhi 1 8889997711 511

keerthi 2 8889997712 512

anshu 3 7672044500 513

akrythi 4 8889997714 514

ashok 5 8889997715 515 \*/

**Query18: Insert a column name called R\_cost with datatype as money and constraint as null in rom table and insert values in it.**

alter table Room add R\_cost money null

update Room set R\_cost=3000 where R\_number=10 or R\_number=12 or R\_number=13

update Room set R\_cost=3250 where R\_number=17 or R\_number=19 or R\_number=20

update Room set R\_cost=3500 where R\_number=23 or R\_number=24 or R\_number=26

update Room set R\_cost=3750 where R\_number=27 or R\_number=30 or R\_number=31

update Room set R\_cost=4000 where R\_number=34 or R\_number=37 or R\_number=40

**/\* output:**

**R\_type R\_number R\_floor R\_cost**

A 10 1 3000.00

A 12 1 3000.00

A 13 2 3000.00

A 17 4 3250.00

A 19 3 3250.00

A 20 3 3250.00

N 23 2 3500.00

N 24 2 3500.00

N 26 5 3500.00

N 27 5 3750.00

N 30 4 3750.00

N 31 3 3750.00

N 34 1 4000.00

N 37 2 4000.00

A 40 2 4000.00 \*/

**Query19: Find the name of the hotel whose name starts with b and having atleast five characters.**

select Hname from hotel where Hname like 'b\_%'

**/\* output:**

**Hname**

baghvan

banjaar **\*/**

**Query20: Display the average of all the facilities costs.**

select avg(F\_cost) as Avg\_fac\_cost from Facilities\_available

**/\* output:**

**Avg\_fac\_cost**

2120.00 **\*/**

**Query21: Find the mobile number of staff whose name ends with i.**

select s\_mobile from staff where s\_name like '%i'

**/\* output:**

**s\_mobile**

8889997776

8889997777

8889997710

8889997711

8889997712

8889997714 **\*/**

**Query22: create a view from customers which includes CID, CName and mobile number of customers of age >=30.**

create view customer\_details as

select CID, CName, Cmobile

from customers where age>=30

select \* from customer\_details

**/\* output:**

**CID CName Cmobile**

406 rithu 9997776666

407 abhi 9997776667

408 rohan 9997776668

411 rehman 9997776611

412 priya 9997776612

413 nitya 9997776613

414 jayanth 9997776614

415 rujith 9997776615  **\*/**

**Query23: Find the booking id of customer who booked the room 19.**

select b.B\_id from Booking b, Room r where b.R\_number=r.R\_number and r.R\_number=19

**/\* output:**

**B\_id**

305 **\*/**

**Query24: Display the top five cid's of customers.**

select top(5) CID from customer

**/\* output:**

**CID**

101

102

103

104

105 **\*/**

**Query25: Find the ID's and count of ID's of customers whose age is greater than 25 ( use group by statement).**

select CID, count(CID) as count from customers

where age>25 group by CID

**/\* output:**

**CID count**

406 1

407 1

408 1

411 1

412 1

413 1

414 1

415 1

416 1

417 1  **\*/**

**Query 26: Find names of manager of hotel.**

select M\_name from manager m, hotel h where h.HID=m.HID

**/\*output:**

**M\_name**

shriya

keishna

meera

tarun

divya

pradeepthi

reshma

deepak

padma

karthik

neha **\*/**

**Query 27: Find the staff who belongs to food department**

select s\_name from staff s,Depart D where D.Dname='food' and s.DID=D.DID

**/\*output:**

**s\_name**

ram

ravi

keerthi **\*/**

**Query 28: Display the names of mangers who belongs to security and house keeping department. ( use UNION)**

select m.M\_name from manager m,Depart D,hotel h where D.Dname='security'

and m.DID=D.DID and m.HID=h.HID

UNION

select m.M\_name from manager m,Depart D,hotel h where D.Dname='house keeping'

and m.DID=D.DID and m.HID=h.HID

**/\*output:**

**M\_name**

meera

neha

reshma **\*/**

**Query 29: Find the booking id who booked roomtype as ac.**

select b.B\_id from Room r,Booking b where b.R\_number=r.R\_number and r.R\_type='A'

**/\*output:**

**B\_id**

301

302

303

304

305

306

315**\*/**

**Query 30: Find cid, cname and from customer whose age is minimum**

select CID,CName,age from customers where age=(select min(age) from customers)

**/\*output:**

**CID CName age**

410 shetal 10 **\*/**

**Query31: Display the total no of departments**

Select count (DID) from Depart

**/\* output:**

**(No column name)**

5  **\*/**

**4.  CONCLUSION AND FUTURE WORK**

**4.1 Conclusion**

This project is designed to meet the requirements of Online Hotel Management.

It has been developed in Visual Basic keeping in mind the specifications of the system.

This project have been designed in a great user friendly and largely providing attribute interface and tp meet the requirement of a Hotel Management System.. The project is working with quick response and same level of security is also maintained. The Oracle database is maintained and frontend with Visual Basic.

In this designing system, I have used an ER Diagrams; overall this report teaches us the essential skill like;

-Understanding the Oracle database handling and query processing.

-Using system analysis and design techniques like ER diagram and Data modeling in designing the system.

**4.2 Future Work**

The Hotel management is one of the highly job oriented field; it covers a wide range of services including food service, accommodation and catering. The major job fields in the hospitality sector include Hotels, resorts, fast food chains, restaurants, etc. The main purpose of hotel management is to ensure a constant influx of visitors and guests to the hotel throughout the year, showcase the wide variety of services and USPs and how it benefits the visiting guests through marketing initiatives.

In the near future automation will be a great part of hotel management industry too: Speak to order platforms will be trendy by which a virtual assistant will manage everyday mundane tasks, such as taking simple food orders and explain all the queries a customer has.