Admas University

Department Of Computer Science

*Co*urse:-Advanced Datbase System(Adbms)

Proposal For Hotel Management System

GROUP MEMBERS

1. Abdu Ali........................ ....ADMA/1898/19
2. Denekew Aschalew……...............……..ADMA/2248/19
3. Getachew Lemma............ ....ADMA/2634/19
4. Mulatu Admasu............ ....ADMA/2533/19
5. Tadesse Worku.............. ....ADMA/2070/19

Submitted to Instructor:- SHIWAKENA G.
Date:-5/26/23

Table of Contents

1. Introduction

Some business websites like (www.businessdictionary.com) defines hotel as the following: **Hotel** is a commercial establishment providing, lodging, meals and other guest services. In general, to be called a hotel, an establishment must have a minimum of six letting bedrooms, at least three of which must have attached private bathroom facilities. Although hotels are classified into 'Star' categories (1-Star to 5-Star), there is no standard method of assigning these ratings, and compliance with customary requirements is voluntary. A US hotel with a certain rating, for example, it may look very different from a European or Asian hotel with the same rating, and would provide a different level of amenities, range of facilities, and quality of service.

A **Hotel Information Management System (HIMS)** is an automated system for the hotel management; it thus creates standards in hotels according to customer needs, satisfying their general comfort

1.1. Objective

The project has both general and specific objectives here are the general objective and then Specific objectives

1.2. General objective

The main objective of this project is to design and develop hotel information management system for BlueSky hotel. That can straight forward and seek to ensure we run a professional, profitable and ethical company, building relationships with customers, suppliers and investors, driving business at BlueSky hotel and developing the business as a whole. It is also adopt best commercial practice and ethical standards dealing which clientele, suppliers of goods and services and other contacts

1.3. Specific objective

- Reviewing how the current system works and operates.
- Investigating how the existing system is operating.
- Design a new system that can overcome the problem of the current system
- Identify functional and non functional requirements for the new systems.
- Develop and implement the new system.
- Forward recommendation about system implementation, direction and guidelines

1.4. Scope of the system

Although, design concepts, outputs and other components of the Project can be used for different purpose, the implementation of this project is manage data and human resource management. The scope of the project mainly deals with providing insert data, Update Guest Information, and human resource parts like add employee terminate employee and update employee and etc….

2. Problem Statement

2.1. The current system

Bluesky hotel is an organization that was established in the area. People reserve hotels for their day to day activities, for this reason data handling and record keeping for customers is very difficult today because, the hotel has kept the data with paper records in filing cabinets so it can’t allow employee of the hotel to retrieve customer information from the manual. Now BlueSky hotel has its own database, It can easily organize the hotel's, customer's and employee's information and saves hours and time.

2.2. Problems of the existing system:

The existing (manual) system of BlueSky hotel has the following limitations

- Existing System is manual management System
- Records are managed manually so it is very hard to maintain the records systematically
- There is no data security in the present system, so any user can view and change the data anyway.
- Difficult to generate report
- Storage problems (Update, Search, Delete, Edit), these are very time taking
- No support in decision-making

2.3. Data collection

It is method of data collection for both functional requirement and non functional requirement.

- Interview the hotel manger and employees of the hotel
- Questionnaires from customer of the hotel

2.3.1. Tool selection

Programming languages, tools and techniques will used for the development of the system:

- MySql server

3. The proposed system

After studying the existing system and identifying the limitations, we designed a new system that can provide and ensure the following features:

- Provide easy access to clients (customers) and employee information.
- Satisfy customer's needs and demands.
- Ensure fast and accurate record keeping mechanisms.
- Provide system simplicity, reliability, fast, user friendly and reliable work.
- Provide very Secured data keeping (unauthorized users can't access the system)
- Make a system easy to modify the existing data and customer's information (Search, Update, Delete)

3.1. Functional Requirements

Functional requirements describe the relations between the system and the user or the environment. Here the 'relations' means the direct or indirect interactions between the user and the system. Hence the system has the following stakeholders with different requirements

3.1.1. System Administrator

- ***RQ1***, The system allows the System Administrator to add room.
- ***RQ2***, The system allows the System Administrator to delete room.
- ***RQ3***, The system allows the System Administrator to update room.
- ***RQ4***, The system allows the System Administrator to add employee.
- ***RQ5***, The system allows the System Administrator to terminate employee.
- ***RQ6***, The system allows the System Administrator to update employee.
- ***RQ7***, The system allows the System Administrator to update hotel

3.1.2. Manager

- ***RQ1,*** The system allows the Manger to all system Administration
- ***RQ2,*** The system allows the Manger to permit leave.
- ***RQ3,*** The system allows the Manger to generate report.
- ***RQ4,*** The system allows the Manager to authorize payroll.

3.1.3. Employee

- ***RQ1,*** The system allows the Employee to reserve room.
- ***RQ2,*** The system allows the Employee to cancel reserved room.
- ***RQ3,*** The system allows the Employee to update reserved room.
- ***RQ4,*** The system allows the Employee to search for available room.
- ***RQ5,*** The system allows the Employee to search customer information

3.2. System Requirement

The hardware and software component of a computer system that is required to install and use software efficiently. If your computer system does not meet the system requirements then the software may not work correctly after installation. System requirements for operating systems will be hardware components, while other application software will list both hardware and operating system requirements.

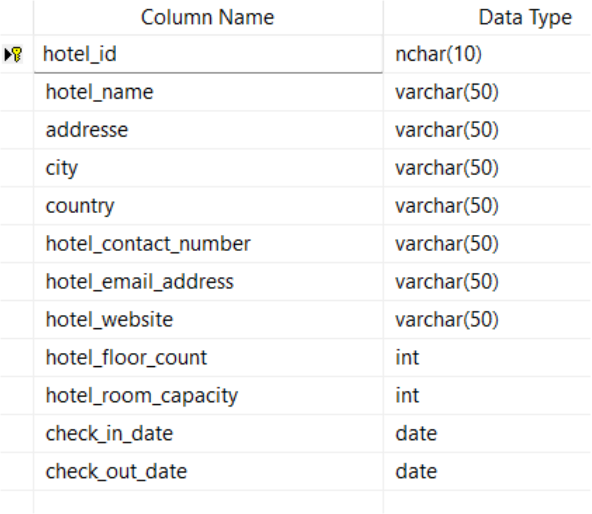
4. Conceptual Database Design

4.1. Define Tables: Design the tables, columns and relationships

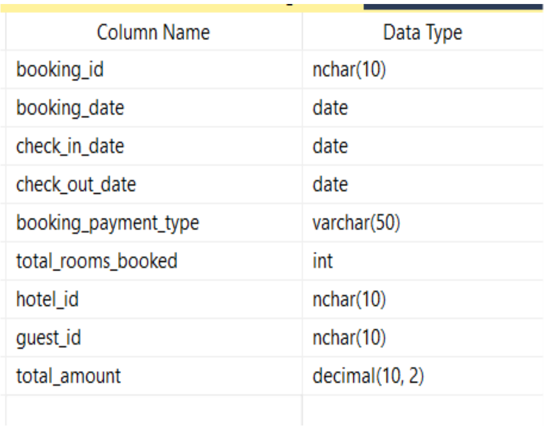
We started with designing the structure of the database. We drew the ERR diagram on a paper, noting down all the tables required. We designed each table with columns and attributes respectively. I have designed the below tables:

Hotel table consists of data related to the hotel. The primary key is hotel_id.

hotel_id which has one-to-many relationship with the employee,guest,booking,room and hotel_service table.



Column Name	Data Type
🔑 hotel_id	nchar(10)
hotel_name	varchar(50)
addresse	varchar(50)
city	varchar(50)
country	varchar(50)
hotel_contact_number	varchar(50)
hotel_email_address	varchar(50)
hotel_website	varchar(50)
hotel_floor_count	int
hotel_room_capacity	int
check_in_date	date
check_out_date	date



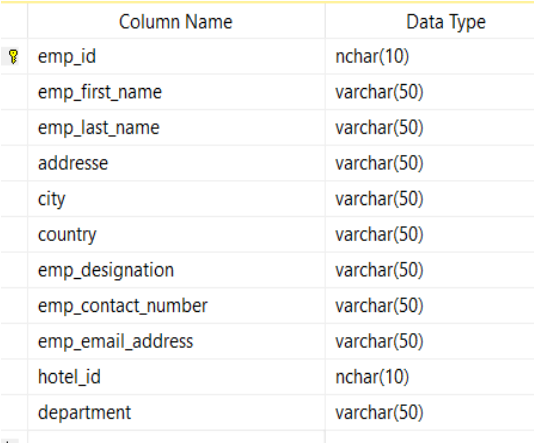
Column Name	Data Type
booking_id	nchar(10)
booking_date	date
check_in_date	date
check_out_date	date
booking_payment_type	varchar(50)
total_rooms_booked	int
hotel_id	nchar(10)
guest_id	nchar(10)
total_amount	decimal(10, 2)

bookings table contains data about the booking made for rooms. The primary key for this table is booking_id.

The table has the following foreign keys: hotel_id and guest_id

hotel_id which has a many-to-one relationship with the hotel table.

guest_id which has a many-to-one relationship with the guests table.



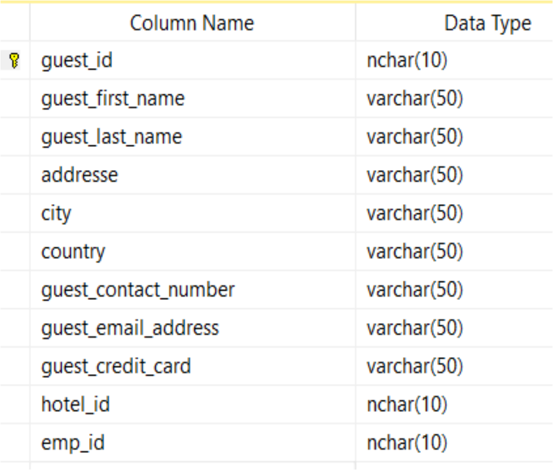
	Column Name	Data Type
🔑	emp_id	nchar(10)
	emp_first_name	varchar(50)
	emp_last_name	varchar(50)
	addresse	varchar(50)
	city	varchar(50)
	country	varchar(50)
	emp_designation	varchar(50)
	emp_contact_number	varchar(50)
	emp_email_address	varchar(50)
	hotel_id	nchar(10)
	department	varchar(50)

Employees table consists of data related to the employees. The primary key is employee_id. There are one foreign keys, hotel_id.

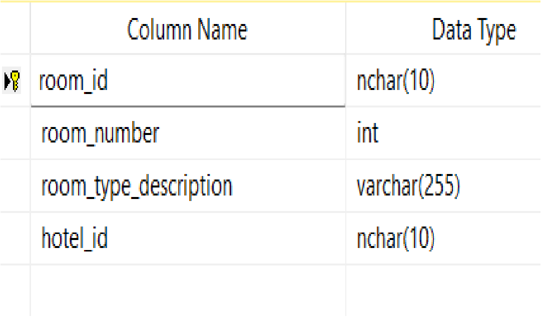
hotel_id that denotes many-to-one relationship with the hotel table.

Guests table has the data about the guests that check in to the hotel.

The primary key of this table is guest_id. there is two foreign key in this table, hotel_id and Emp_id that has many-to-one relationship with the hotel and employee table.



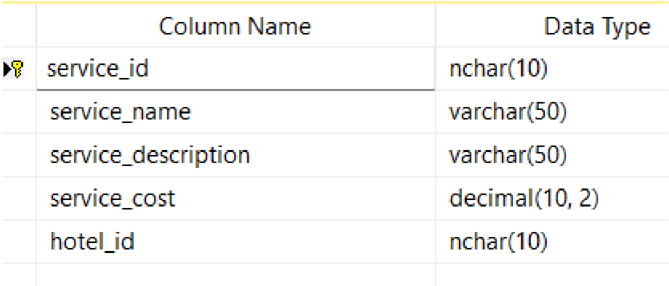
	Column Name	Data Type
🔑	guest_id	nchar(10)
	guest_first_name	varchar(50)
	guest_last_name	varchar(50)
	addresse	varchar(50)
	city	varchar(50)
	country	varchar(50)
	guest_contact_number	varchar(50)
	guest_email_address	varchar(50)
	guest_credit_card	varchar(50)
	hotel_id	nchar(10)
	emp_id	nchar(10)



	Column Name	Data Type
🔑	room_id	nchar(10)
	room_number	int
	room_type_description	varchar(255)
	hotel_id	nchar(10)

Room table has the data about the Room that check in to the hotel.

The primary key of this table is room_id. there is one foreign key in this table, hotel_id that has many-to-one relationship with the hotel table.



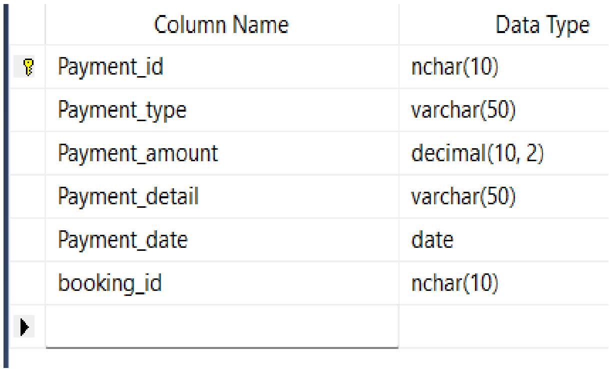
	Column Name	Data Type
▶🔑	service_id	nchar(10)
	service_name	varchar(50)
	service_description	varchar(50)
	service_cost	decimal(10, 2)
	hotel_id	nchar(10)

Hotel_Services table has the data about the hotel_services that check in to the hotel.

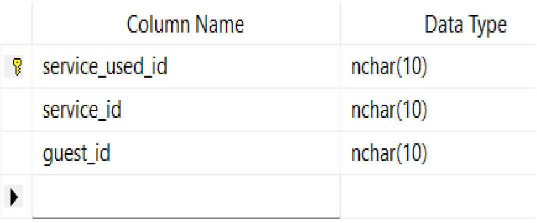
The primary key of this table is service_id. there is one foreign key in this table, hotel_id that has many-to-one relationship with the hotel table.

Payment table has the data about the payment that check in to the booking.

The primary key of this table is payment_id. there is one foreign key in this table, booking_id that has many-to-one relationship with the booking table.



	Column Name	Data Type
🔑	Payment_id	nchar(10)
	Payment_type	varchar(50)
	Payment_amount	decimal(10, 2)
	Payment_detail	varchar(50)
	Payment_date	date
	booking_id	nchar(10)
▶		



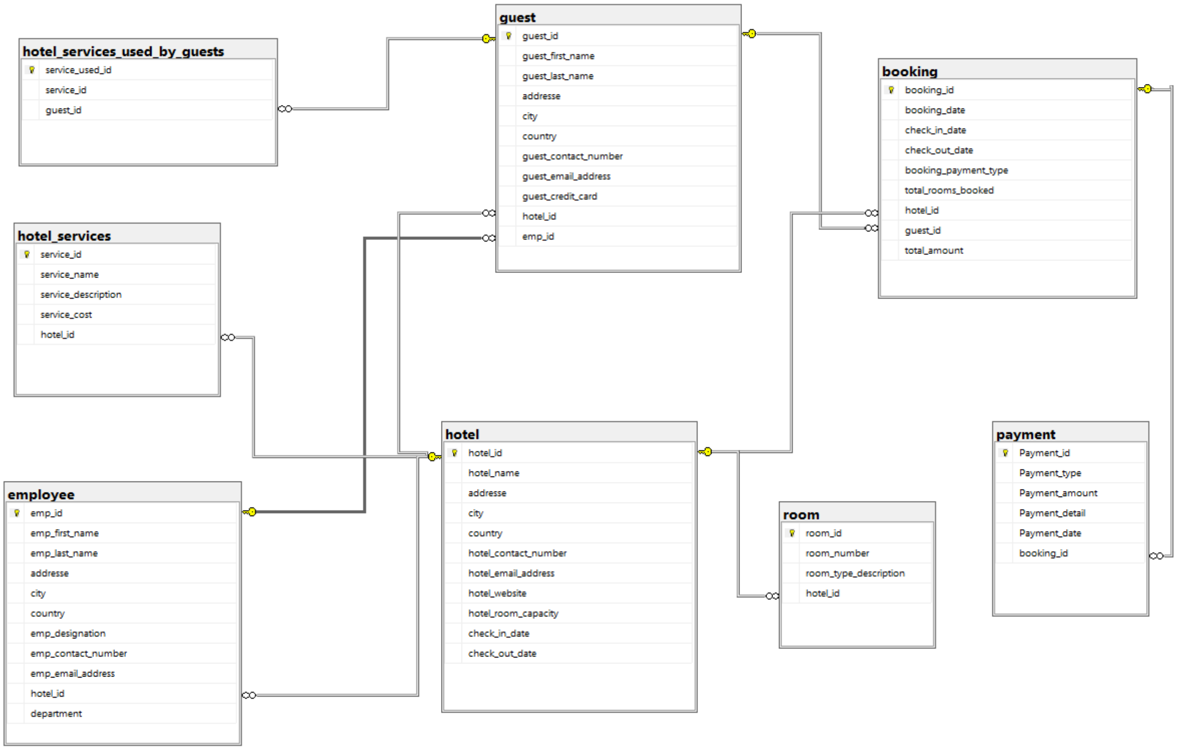
	Column Name	Data Type
🔑	service_used_id	nchar(10)
	service_id	nchar(10)
	guest_id	nchar(10)
▶		

Hotel_services_used_by_guests table contains info about the services used by the guests. Primary key is service_used_id & two foreign keys,

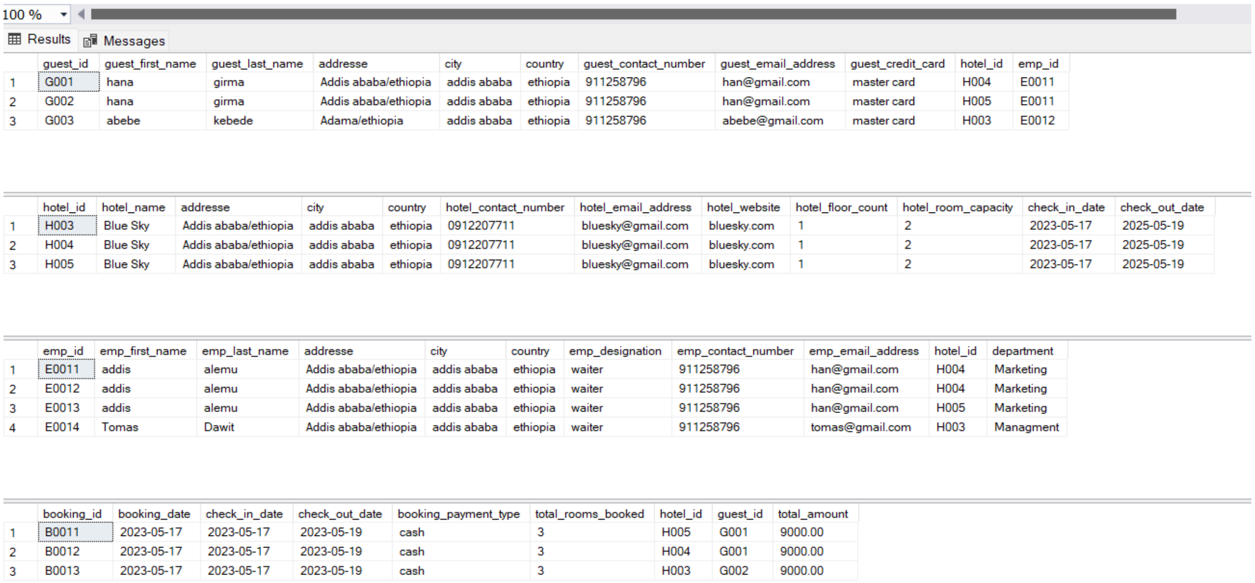
service_id, which relates to hotel_services table & guest_id relates to guest table.

4.2. **Database Designing:** Starting with drawing the ERR diagram

We started with designing the structure of the database. We drew the ERR diagram on a paper, noting down all the tables required. We designed each table with columns and attributes respectively and trying to make an idea about the relationships between tables. We tried to keep the tables in a form where we can reduce the data redundancy and tried to make it accessible in an easy and simple way. The ERR diagram created is as below:



4.3. Relational schema:



	guest_id	guest_first_name	guest_last_name	addresse	city	country	guest_contact_number	guest_email_address	guest_credit_card	hotel_id	emp_id
1	G001	hana	girma	Addis ababa/ethiopia	addis ababa	ethiopia	911258796	han@gmail.com	master card	H004	E0011
2	G002	hana	girma	Addis ababa/ethiopia	addis ababa	ethiopia	911258796	han@gmail.com	master card	H005	E0011
3	G003	abebe	kebede	Adama/ethiopia	addis ababa	ethiopia	911258796	abebe@gmail.com	master card	H003	E0012

	hotel_id	hotel_name	addresse	city	country	hotel_contact_number	hotel_email_address	hotel_website	hotel_floor_count	hotel_room_capacity	check_in_date	check_out_date
1	H003	Blue Sky	Addis ababa/ethiopia	addis ababa	ethiopia	0912207711	bluesky@gmail.com	bluesky.com	1	2	2023-05-17	2025-05-19
2	H004	Blue Sky	Addis ababa/ethiopia	addis ababa	ethiopia	0912207711	bluesky@gmail.com	bluesky.com	1	2	2023-05-17	2025-05-19
3	H005	Blue Sky	Addis ababa/ethiopia	addis ababa	ethiopia	0912207711	bluesky@gmail.com	bluesky.com	1	2	2023-05-17	2025-05-19

	emp_id	emp_first_name	emp_last_name	addresse	city	country	emp_designation	emp_contact_number	emp_email_address	hotel_id	department
1	E0011	addis	alemu	Addis ababa/ethiopia	addis ababa	ethiopia	waiter	911258796	han@gmail.com	H004	Marketing
2	E0012	addis	alemu	Addis ababa/ethiopia	addis ababa	ethiopia	waiter	911258796	han@gmail.com	H004	Marketing
3	E0013	addis	alemu	Addis ababa/ethiopia	addis ababa	ethiopia	waiter	911258796	han@gmail.com	H005	Marketing
4	E0014	Tomas	Dawit	Addis ababa/ethiopia	addis ababa	ethiopia	waiter	911258796	tomas@gmail.com	H003	Managment

	booking_id	booking_date	check_in_date	check_out_date	booking_payment_type	total_rooms_booked	hotel_id	guest_id	total_amount
1	B0011	2023-05-17	2023-05-17	2023-05-19	cash	3	H005	G001	9000.00
2	B0012	2023-05-17	2023-05-17	2023-05-19	cash	3	H004	G001	9000.00
3	B0013	2023-05-17	2023-05-17	2023-05-19	cash	3	H003	G002	9000.00

4.4. Normalization:

Tables should be normalized till 3rd norm Database normalization is the process of restructuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. Generally, if a database is normalized until third normal form then it is considered to be normalized. We tried to normalize the database until third normal form.

5. SQL Programming that include

- SQL Constraints (unique , check , identity element)

```
-- Add UNIQUE Constraint
CREATE TABLE [dbo].[room](
        [room_id] [nchar](10)PRIMARY KEY NOT NULL,
        [room_number] [int] NOT NULL UNIQUE, -- Add UNIQUE Constraint
        [room_type_description] [varchar](255) NULL,
        [hotel_id] [nchar](10) NULL,
        );


-- Add check Constraint
CREATE TABLE [dbo].[hotel_services](
        [service_id] [nchar](10)PRIMARY KEY NOT NULL,
        [service_name] [varchar](50) NULL,
        [service_description] [varchar](50) NULL,
        [service_cost] [decimal](10, 2) CHECK ([service_cost] >=1 ),
        [hotel_id] [nchar](10) NULL,
            );
```

5.1. Stored procedure

```
--Stored Procedure With One Parameter used to Cascaded delete from hotel,guest,booking and employee table

ALTER PROCEDURE [dbo].[cascadedelete] @hotel_id nchar(10)
AS
  Delete from hotel
  where hotel_id = @hotel_id;


EXEC cascadedelete @hotel_id ='';
```

5.2. SQL Select Queries

```
-- How many rooms are booked in a particular hotel on a given date?
       SELECT SUM(total_rooms_booked) AS 'Total Rooms Booked'        -- sum of total 
rooms
       FROM booking
       WHERE booking_date LIKE '2023-05-17' AND hotel_id = 'H001';

       -- How many books has a customer made in one year?
       SELECT count(*) AS 'Total Bookings'              -- count of total bookings
       FROM booking
       WHERE YEAR(booking_date) = 2023 AND guest_id ='G001';         -- bookings in Year 
2023 by guest Jane with id 1
```

6. **S**ummary**:**

This was an attempt to create a database management system for hotel where a HotelMS can easily manage the hotel, rooms, bookings, guests, employees, services, etc. and other things as well, easily and quickly. Overall, it is huge area and we tried to cover few of the parts of it. Thank you!