HOTEL MANAGEMENT SYSTEM

- D HARSHA VARDHAN
  14CS01033
- Y HARSHA PRIYATAM
  14CS01040

1. Abstract
2. Introduction and requirement analysis
3. System analysis and proposed design
4. Conceptual design
5. Software requirement specification
6. Implementation
7. Features and security

ABSTRACT

- Project titled “HOTEL MANAGEMENT SYSTEM” is a web based application to serve the management needs of a hotel/cuisine.
- It has several features for customers which includes
  - Rooms booking
  - Food ordering etc
- It also have features for management of employees etc.
- This application is build on basing a relational database.

Requirement analysis

- Disadvantages of manual system
  - Large storage space
  - Human and computational errors
  - Poorly generated records
  - Poor communication
  - Difficulty in data analysis
- To enable online booking via the internet.
- Ensure efficient and reliable communication within the hotel.
- Avoid data entry errors by use of input masks.
- Enforce security measures to avoid unauthorized access to guest records.
- Enable fast and easy retrieval of guest records and data for fast reference activities.

Proposed system

- A internet application which enables the customers and management to
  - Check availability of the room without having to come to hotel
  - Book slots for rooms online
  - Book food, services, cabs.. Etc after lodging in to hotel
- Advantages:
  - Entire system is completely synchronous, ensured consistency
  - Data integrity
  - Security aspects
  - Faster communication

Design process

- A conceptual design to entire problem of managing a hotel.
- Figuring the data storage requirements for process.
- Identifying the entities of storage.
- Specifying the entities and fields.
- Figuring all the actions or processes to achieve the required functionalities.
- Relating the data and actions to make a Entity-Relationship model.

Software requirements

- A relational database to implement the conceptual design of the data storage requirement of the application
- MySql is a freeware ,light weight and efficient relational database so, we choose it
- Graphical user interface (GUI) can be made in many ways like a command line application, stand-alone os application, web application
- Web apps are easy to develop, available to everyone by use of internet, environment independent so, we choose this way.
- Middleware of a dynamic web app can be developed using programming languages like python, php, Node.js framework, Go, Ruby, Java, C# etc..
- Java is a robust PL with support for web apps by having servlet class and JSP

Complete Technical requirements

- Interface: HTML and CSS
- client side: JavaScript
- server: Apache tomcat 8
- server side: JAVA, servlet/JSP
- DBMS: MySql 5.7

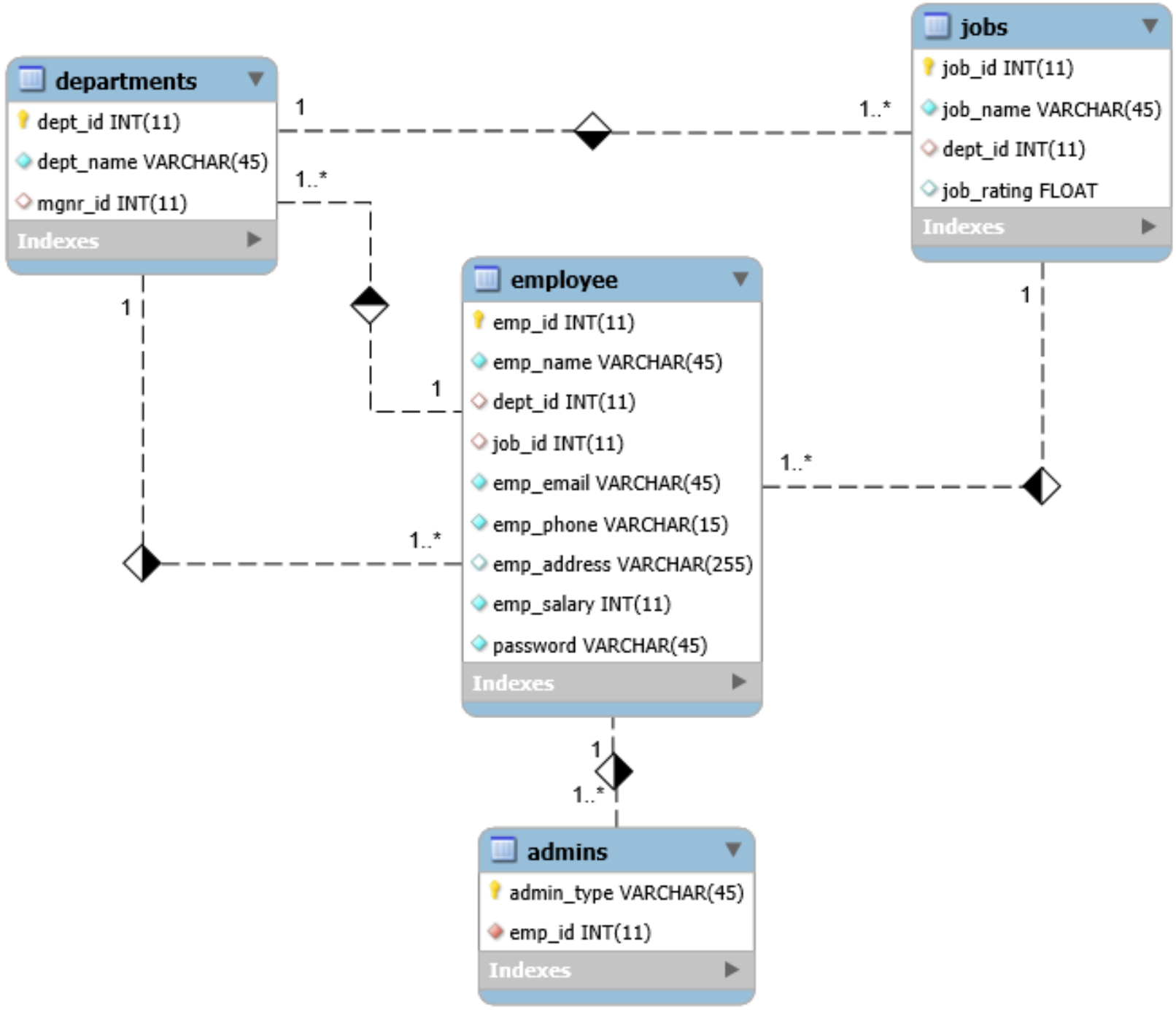
Implementation

- *Customer*
  1. can check availability and book/cancel hotel rooms.
  2. can order food & beverages after check-in in to the booked room.
  3. can book services like cabs, laundry etc.
  4. can complaint and give feedback on all the services.
- *Managers*
  1. can receive the orders from customers boarded in room.
  2. Improve the quality of services according to the feedback received.
- *HR and Sales managers*
  1. Monitor all the activities in the hotel.
  2. Can add/remove employees in the hotel in any department.
  3. Change the heads of departments.
  4. View the reports based on analysis of data, feedback of services and implement changes to improve hotel sales.

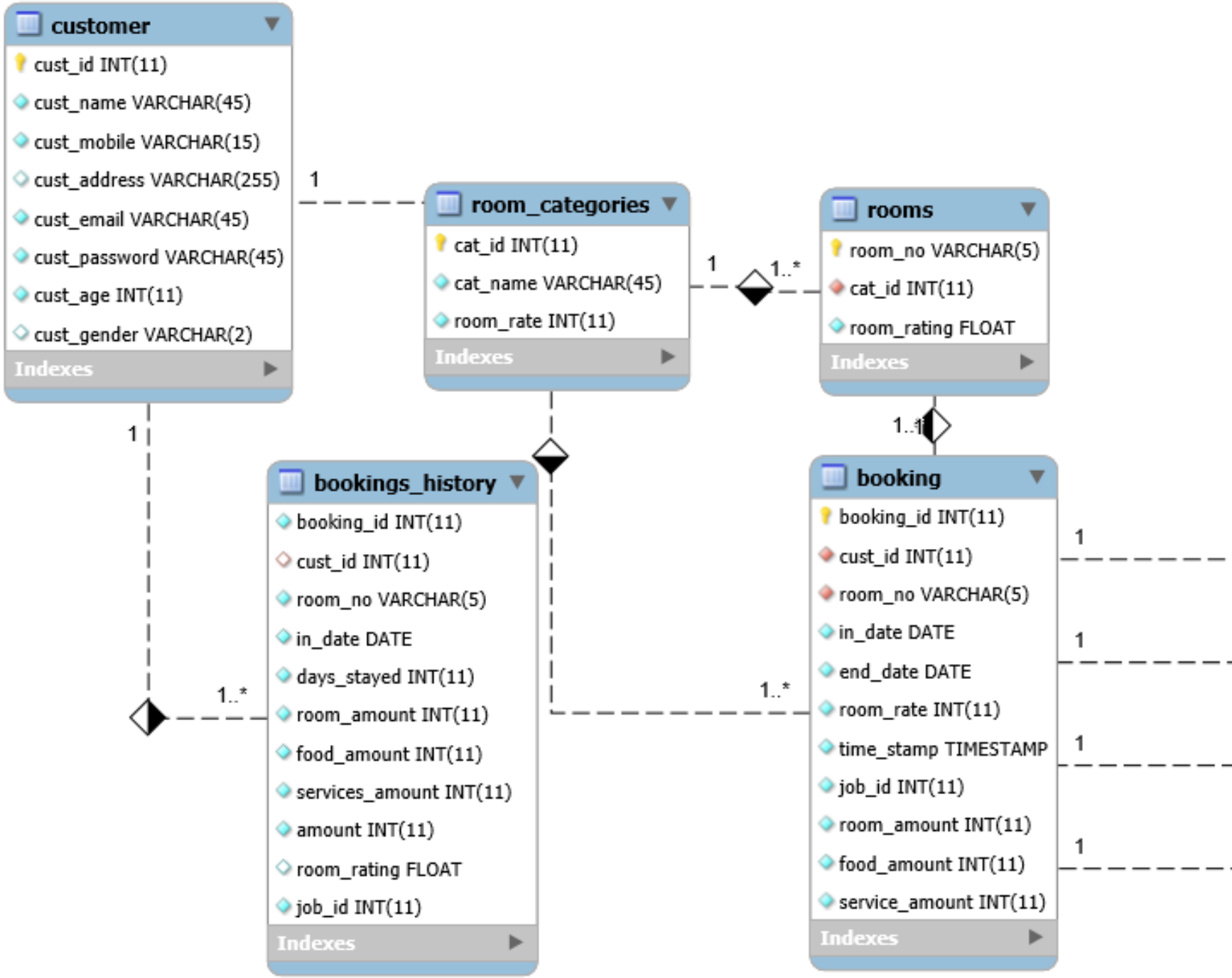
Data entities

- **Database**:
  contains 7 parts
  1. HR
  2. Profile and credentials
  3. Rooms and bookings
  4. Food and bookings
  5. Services
  6. Complaints

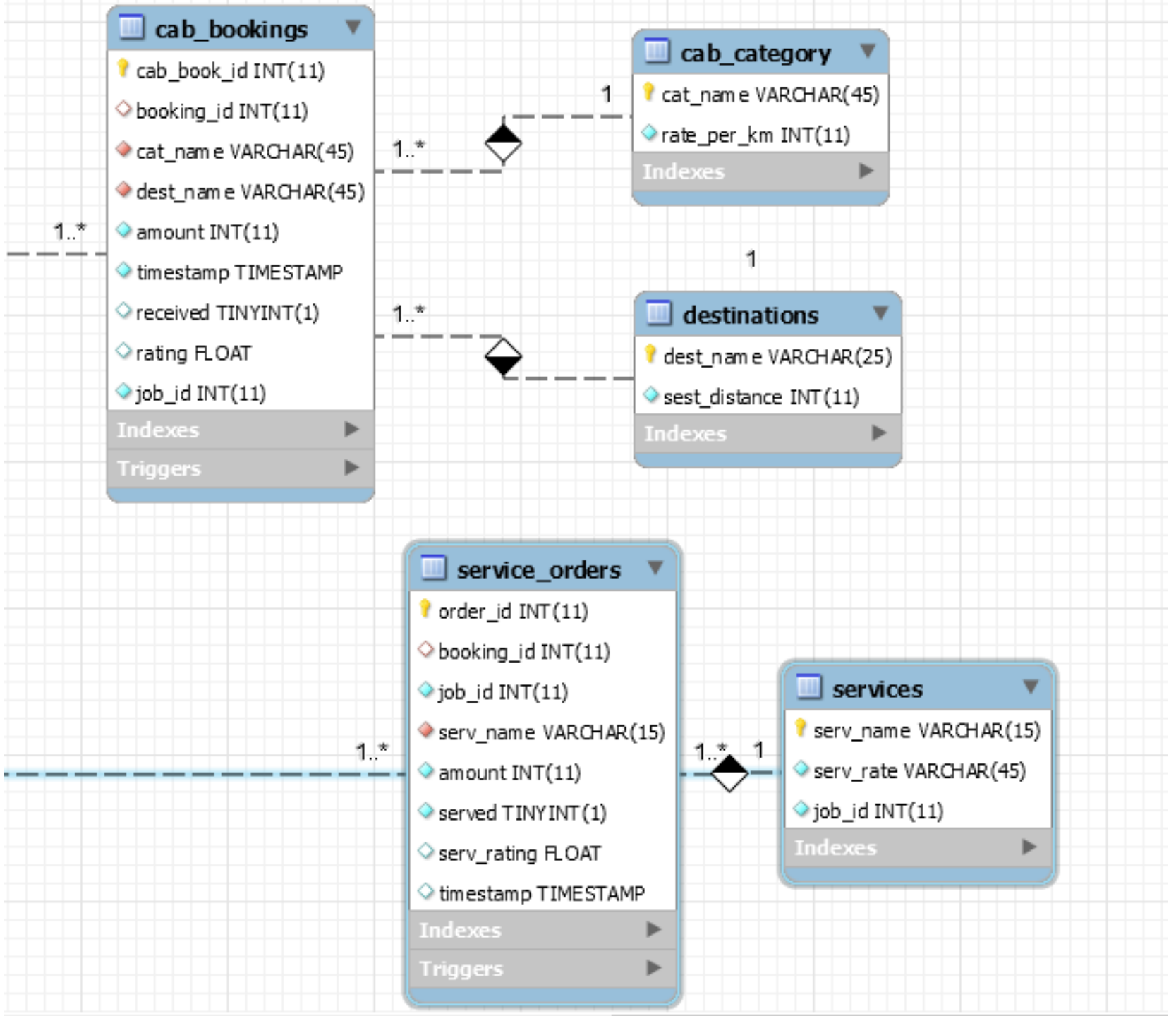
HR


departments
dept_id INT(11)
dept_name VARCHAR(45)
mgnr_id INT(11)
Indexes
jobs
job_id INT(11)
job_name VARCHAR(45)
dept_id INT(11)
job_rating FLOAT
Indexes
employee
emp_id INT(11)
emp_name VARCHAR(45)
dept_id INT(11)
job_id INT(11)
emp_email VARCHAR(45)
emp_phone VARCHAR(15)
emp_address VARCHAR(255)
emp_salary INT(11)
password VARCHAR(45)
Indexes
admins
admin_type VARCHAR(45)
emp_id INT(11)
Indexes
1
1..*

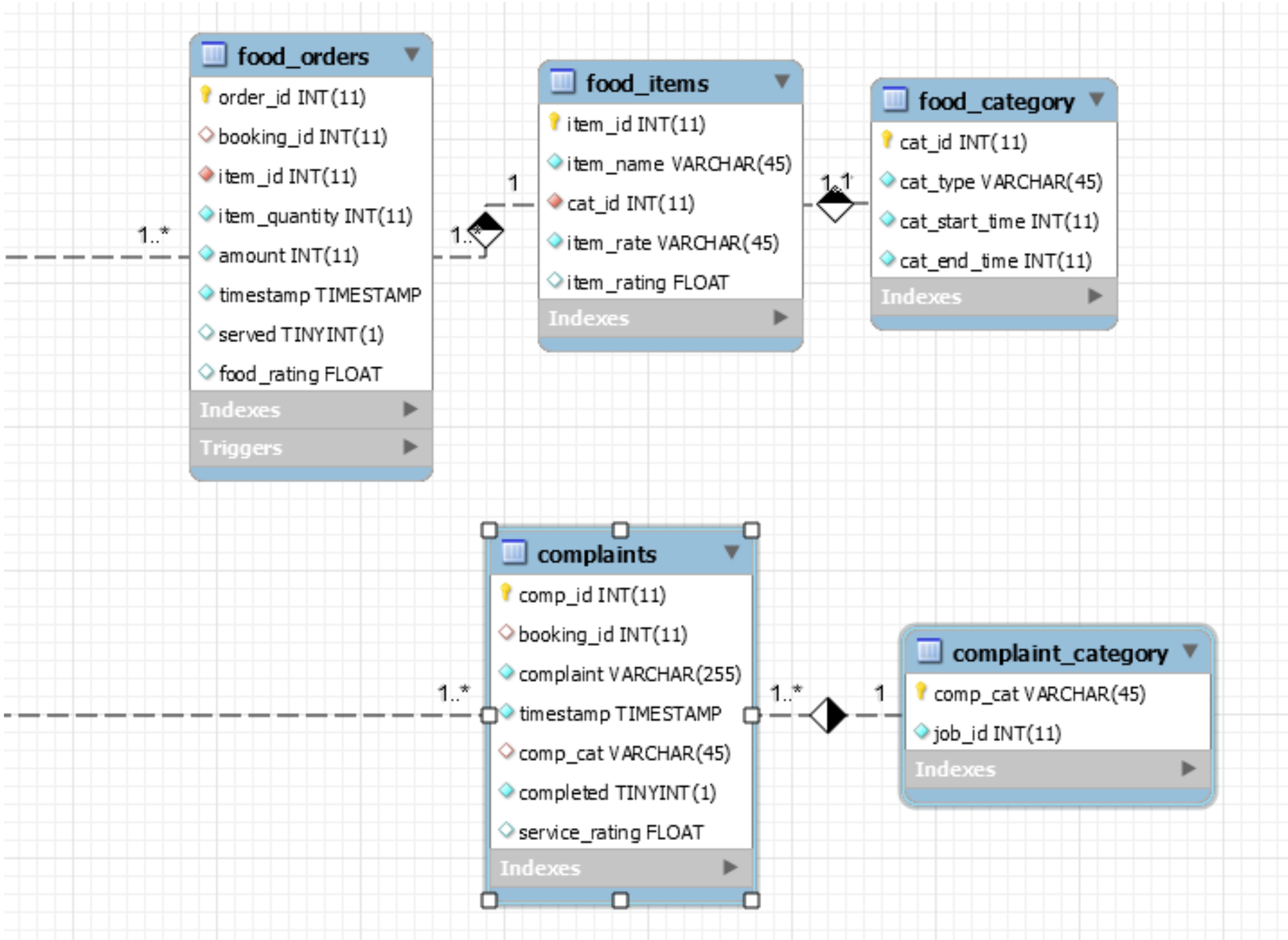
Rooms and Accounts



Services


cab_bookings
cab_book_id INT(11)
booking_id INT(11)
cat_name VARCHAR(45)
dest_name VARCHAR(45)
amount INT(11)
timestamp TIMESTAMP
received TINYINT(1)
rating FLOAT
job_id INT(11)
Indexes
Triggers
cab_category
cat_name VARCHAR(45)
rate_per_km INT(11)
Indexes
destinations
dest_name VARCHAR(25)
sest_distance INT(11)
Indexes
service_orders
order_id INT(11)
booking_id INT(11)
job_id INT(11)
serv_name VARCHAR(15)
amount INT(11)
served TINYINT(1)
serv_rating FLOAT
timestamp TIMESTAMP
Indexes
Triggers
services
serv_name VARCHAR(15)
serv_rate VARCHAR(45)
job_id INT(11)
Indexes
1
1..*

Food and complaints



Triggering events

- Whenever a service, food order, cab booking is completed the amount calculated will be added to the total amount in the accounts section
- Whenever Feedback is given the average is recalculated and stored to the job, food, service, room etc.

Summary

- Views are used to show data in user end for security of the data
- Procedures are used to for booking rooms, services and food