



Updated Phase 1: Conception

Project: Build a Data Mart in SQL

BSC in Computer Science

Instructor: Musharaf Doger

Topic: Chatbot for Booking a Hotel Room

By

SK Sahil

9213036

Table of Contents

1. INTRODUCTION 3

2. ROLES 3

3. ACTIONS 3

4. DATA AND FUNCTIONS 4

5. ERM (ENTITY RELATIONSHIP MODEL): 5

5. SUMMARY..... 12

6. BIBLIOGRAPHY 14

1. Introduction

Since it establishes the framework for the database's functionality and structure, database modelling is an essential step within the database's design procedure. During this stage, every mistake or omission could have a negative impact on the implementation and possibly make the database worthless. The roles, actions, data, as well as functions that are the main emphasis of this document's requirements specification are for an Airbnb project.

2. Roles

In the Within the Airbnb system, the following positions have been identified:

Hosts: Those that put their homes for rent are known as hosts.

Guests: People who reserve lodging at properties.

Administrators: Administrators are users who oversee the platform, resolve conflicts, and carry out upkeep duties.

Support Staff: Individuals who answer questions and offer assistance to customers.

3. Actions

Every role in the system carries out particular tasks:

Hosts:

- Enumerate attributes
- Control reservations
- Update the property's information.
- Engage in conversation with visitors.

Guests:

- Look for real estate
- Make reservations
- Post evaluations
- Engage in conversation with the hosts.

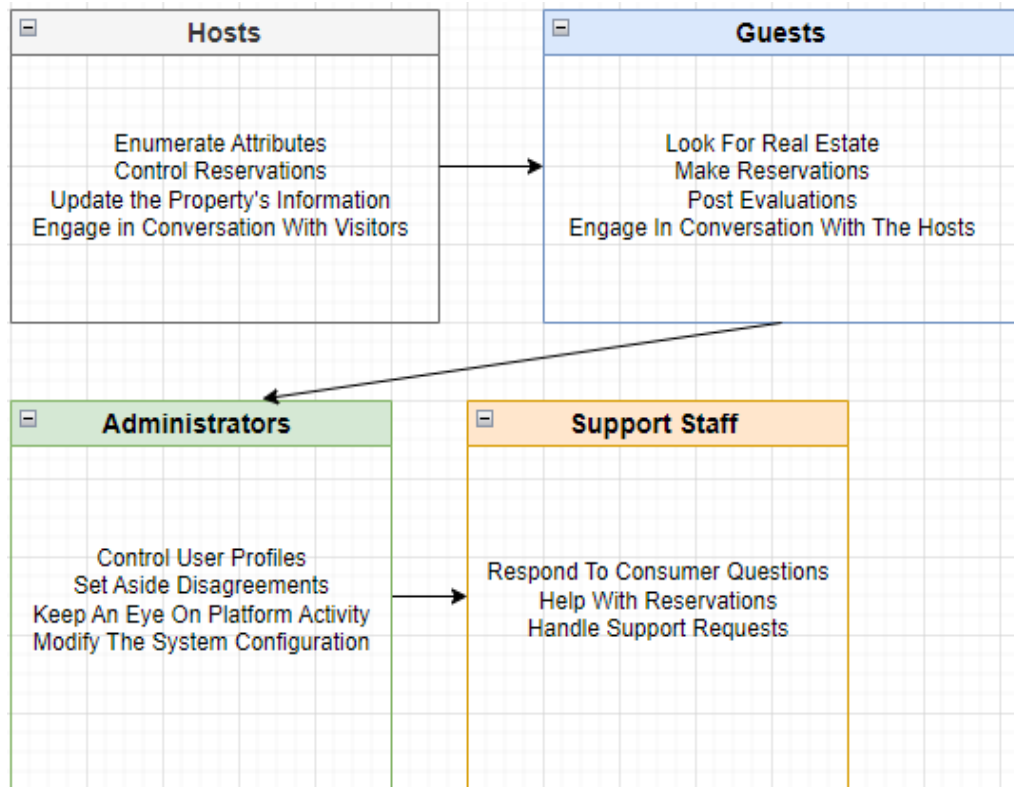
Administrators:

- Control user profiles
- Set aside disagreements
- Keep an eye on platform activity

- Modify the system configuration.

Support Staff:

- Respond to consumer questions
- Help with reservations
- Handle support requests.



4. Data and Functions

The following information and features are needed for the Airbnb system:

Guest Functions:

- **Search and Browse Listings:** Filter and sort listings based on criteria.
- **Make Reservations:** Initiate reservation process, provide check-in and check-out dates, process payments securely.
- **View Reservation History:** Access past reservations.
- **Leave Reviews:** Submit reviews and ratings for past stays.
- **Manage Profile Information:** Update personal details and preferences.

Host Functions:

- **Create and Manage Listings:** Add new listings, edit or deactivate existing listings, provide detailed listing information.

- **Respond to Reservation Requests:** View and respond to reservation requests, confirm or decline reservations based on availability.
- **View Income Reports:** Access reports summarizing yearly income from reservations.
- **Manage Profile Information:** Update personal and listing details.

Employee Functions:

- **Add, Edit, or Delete Employee Records:** Manage employee information and records.
- **Manage Employee Roles and Permissions:** Assign roles and permissions to users, modify roles and permissions for existing users.
- **View Employee-Related Reports:** Access reports related to employee performance and roles.

Admin Functions:

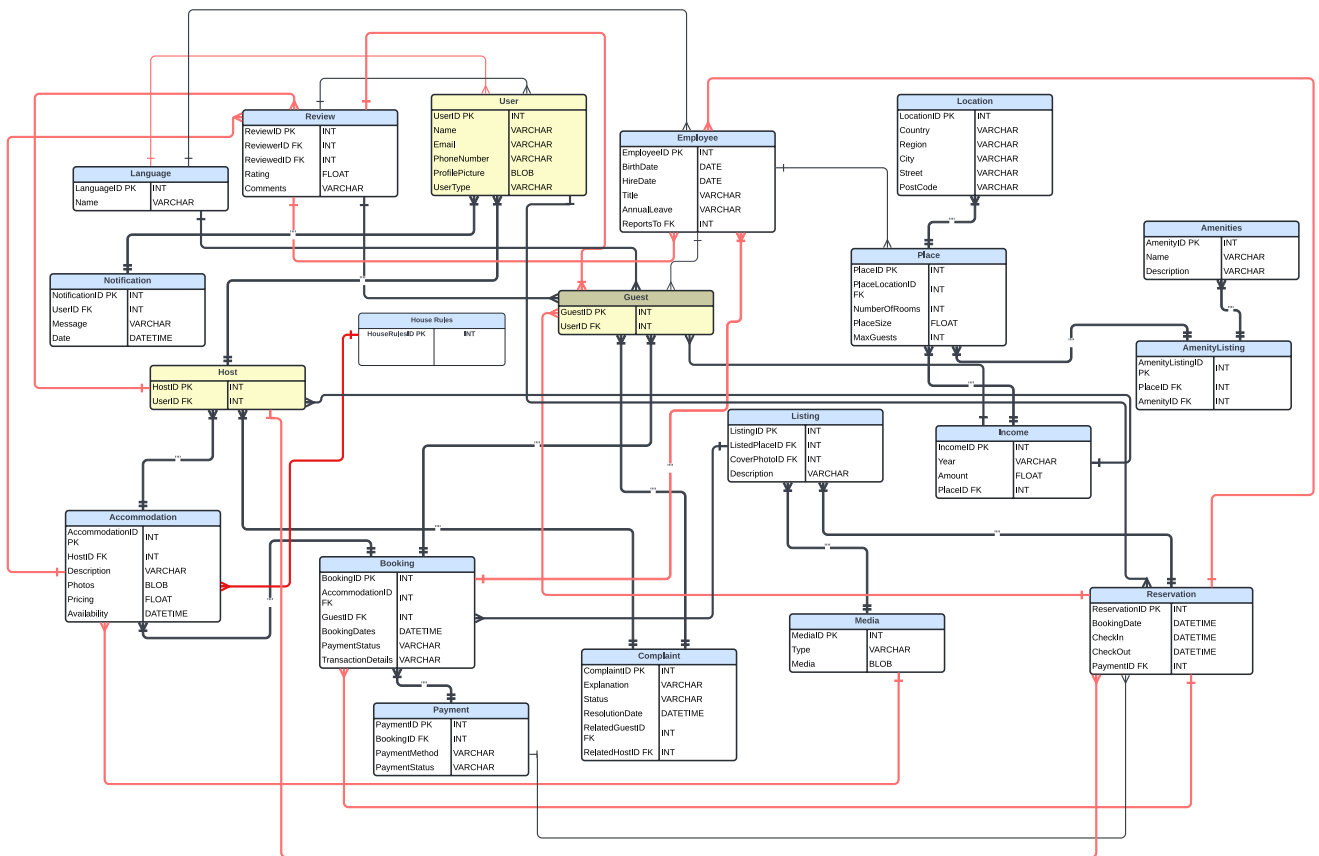
- **Manage User Accounts:** Create, update, and delete user accounts.
- **Configure System Settings:** Manage global settings for system operation.
- **Generate System-Wide Reports:** Access reports for data analysis and monitoring.
- **Resolve and Manage Complaints:** Handle complaints submitted by guests and hosts.

System Functions:

- **Data Integrity Enforcement:** Enforce referential integrity constraints, validate and sanitize user inputs.
- **Notification Generation:** Generate notifications for users based on events.
- **Security Management:** Implement secure authentication and authorization, monitor and log security-related events.
- **Multilingual Support:** User language preferences, language configuration.
- **Media Management:** Upload and manage multimedia content, ensure proper storage and retrieval of media.

5. ERM (Entity Relationship Model):

Updated ERM



Entity Description:

At least twenty entities, two to three triple relationships, including recursive connections as needed will be included in the Airbnb project's ERM. Every entity will be given appropriate properties, and important attributes will be highlighted. Chen notation will be utilized to specify the cardinality of relationships.

Entities and Attributes:

User

- UserID (INT, Primary Key)
- Name (VARCHAR)
- Email (VARCHAR)
- PhoneNumber (VARCHAR)
- ProfilePicture (BLOB)
- UserType (VARCHAR) - Host/Guest

Host

- HostID (INT, Primary Key)
- UserID (INT, Foreign Key)

Guest

- GuestID (INT, Primary Key)

- UserID (INT, Foreign Key)

Accommodation

- AccommodationID (INT, Primary Key)
- HostID (INT, Foreign Key)
- Description (VARCHAR)
- Photos (BLOB)
- Pricing (FLOAT)
- Availability (DATETIME)

Triple Relationships:

- Booking-Guest-Property: The relationship between reservations made by visitors for properties is represented by the term Booking-Guest-Property.
- Review-Booking-Guest: Indicates the connection between the reviews that visitors leave for the reservations they make.

Recursive Relationships:

- User-Support Ticket: A host or visitor has the ability to create and handle multiple support tickets, as well as multiple tickets from the support staff.

Data Dictionary:

| | | | |
|-------------|----------------|-------------------------------|--------------|
| User | UserID (PK) | Unique identifier for a user. | int |
| | Name | Name of the user. | varchar (45) |
| | Email | Email of the user. | varchar (45) |
| | PhoneNumber | Phone number of the user. | varchar (45) |
| | ProfilePicture | Profile picture of the user. | blob |
| | UserType | Type of the user | varchar (45) |

| | | | |
|-------------|-------------|-------------------------------|-----|
| Host | HostID (PK) | Unique identifier for a host. | int |
|-------------|-------------|-------------------------------|-----|

| | | | |
|--|-------------|--------------------------------------|-----|
| | UserID (FK) | References the id in the user table. | int |
|--|-------------|--------------------------------------|-----|

| | | | |
|--------------|--------------|--------------------------------------|-----|
| Guest | GuestID (PK) | Unique identifier for a guest. | int |
| | UserID (FK) | References the id in the user table. | int |

| | | | |
|----------------------|----------------------|--|--------------|
| Accommodation | AccommodationID (PK) | Unique identifier for a accommodation. | int |
| | HostID (FK) | Reference the id of the host table. | int |
| | Description | Description of the accommodation. | varchar (50) |
| | Photos | Photos of the accommodation. | blob |
| | Pricing | Pricing of the accommodation. | float |
| | Availability | Availability of the accommodation. | datetime |

| | | | |
|----------------|----------------------|---|--------------|
| Booking | BookingID (PK) | Unique identifier for a booking. | int |
| | AccommodationID (FK) | Reference the id in the accommodation. | int |
| | GuestID (FK) | References the id in the booking table. | int |
| | BookingDates | Date of the booking. | datetime |
| | PaymentStatus | Payment status of the booking. | varchar (50) |
| | TransactionDetails | Transaction details of the booking. | Varchar (50) |

| | | | |
|---------------|---------------|---------------------------------|-----|
| Review | ReviewID (PK) | Unique identifier for a review. | int |
|---------------|---------------|---------------------------------|-----|

| | | | |
|--|-----------------|--|--------------|
| | ReviewerID (FK) | References the id in the reviewer table. | int |
| | ReviewedID (FK) | References the id in the reviewed table. | int |
| | Rating | Rating is given in the review. | float |
| | Comments | Comments provided in the review. | varchar (50) |

| | | | |
|----------------|----------------|------------------------------------|--------------|
| Payment | PaymentID (PK) | Unique identifier for a payment. | int |
| | BookingID (FK) | Booking id of the payment. | int |
| | PaymentMethod | For payment which method are using | varchar (50) |
| | PaymentStatus | Payment status | varchar (50) |

| | | | |
|--------------------|-------------------|---|-----|
| House Rules | HouseRulesID (PK) | Unique identifier for a House Rules ID. | int |
|--------------------|-------------------|---|-----|

| | | | |
|---------------------|---------------------|--|-------------|
| Notification | NotificationID (PK) | Unique identifier for a notification. | int |
| | UserID (FK) | References the id in the notification table. | int |
| | Message | Message of the notification. | varchar(50) |
| | Date | Date of the notification. | datetime |

| | | | |
|------------------|------------------|--|--------------|
| Complaint | ComplaintID (PK) | Unique identifier for a complaint. | int |
| | Explanation | Detailed explanation of the complaint. | varchar(250) |
| | Status | Current status of the complaint | varchar(50) |

| | | | |
|--|------------------------|---|----------|
| | ResolutionDate | References the date in the resolution date table. | datetime |
| | RelatedGuestID (FK) | References the id in the guest table. | int |
| | RelatedHostID (FK) | References the id in the host table. | int |

| | | | |
|-----------------|-----------------|-----------------------------------|--------------|
| Location | LocationID (PK) | Unique identifier for a location. | int |
| | Country | Country name. | varchar (50) |
| | Region | Region or state name. | varchar (50) |
| | City | City name. | varchar (50) |
| | Street | Street name. | varchar (50) |
| | PostCode | Postal Code. | varchar (50) |

| | | | |
|-----------------|-----------------|---------------------------------|-------------|
| Language | LanguageID (PK) | Unique identifier for language. | int |
| | Name | Name of the language. | varchar(50) |

| | | | |
|-----------------|-----------------|------------------------------------|--------------|
| Employee | EmployeeID (PK) | Unique identifier for an employee. | int |
| | BirthDate | Date of birth of an employee. | date |
| | HireDate | Date when the employee was hired. | date |
| | Title | Job title of the employee. | varchar (50) |
| | AnnualLeave | Annual leave details. | varchar (50) |
| | ReportsTo (FK) | Supervisor or manager. | int |

| | | | |
|---------------|---------------|---------------------------------------|--------------|
| Income | IncomeID (PK) | Unique identifier for income. | int |
| | Year | Year to which the income corresponds. | varchar (50) |
| | Amount | Amount of income. | float |
| | PlaceID (FK) | Reference the id in the place table. | int |

| | | | |
|--------------|----------------------|--|-------|
| Place | PlaceID PK) | Unique identifier for a place. | int |
| | PlaceLocationID (FK) | References the id in the location table. | int |
| | NumberOfRooms | Number of rooms in the place | float |
| | PlaceSize | Place size capacity | int |
| | MaxGuests | Maximum guest capacity | int |

| | | | |
|--------------------|--------------------|--|----------|
| Reservation | ReservationID (PK) | Unique identifier for a reservation. | int |
| | BookingDate | Booking date of the reservation. | datetime |
| | CheckIn | Date and time of check-in. | datetime |
| | CheckOut | Date and time of check-out. | datetime |
| | PaymentID (FK) | Reference the id of the payment table. | int |

| | | | |
|--------------|--------------|--|--------------|
| Media | MediaID (PK) | Unique identifier for media. | int |
| | Type | Type of media. | varchar (50) |
| | Media | Binary data or reference to media content. | blob |

| | | | |
|----------------|--------------------|---|-----|
| Listing | ListingID (PK) | Unique identifier for listing. | int |
| | ListedPlaceID (FK) | Reference the id of the listed place table. | int |

| | | | |
|--|-------------------|--|--------------|
| | CoverPhotoID (FK) | Reference the id of the covered photo table. | int |
| | Description | Description of the listing. | varchar (50) |

| | | | |
|-----------------------|-----------------------|---|-----|
| AmenityListing | AmenityListingID (PK) | Unique identifier for an amenity listing. | int |
| | PlaceID (FK) | References the id in the place table. | int |
| | AmenityID (FK) | References the id in the amenities table. | int |

| | | | |
|------------------|----------------|-------------------------------------|--------------|
| Amenities | AmenityID (PK) | Unique identifier for an amenities. | int |
| | Name | Name of the amenity. | varchar (50) |
| | Description | Description of the amenity. | varchar (50) |

5. Summary

Abstract:

Building a solid as well as extensible database framework which appropriately represents the roles, activities, information, including functionalities throughout the framework was the goal of this database modelling phase for Airbnb. Through a comprehensive requirements analysis and the development of an elaborate ERM, we guarantee that the database will function effectively and efficiently to support Airbnb's operations. Utilizing the Chen technique during relationship requirements along with building a vocabulary of data to specify parameters along with information types are the methods selected for this step.

Procedure:

- Requirement Analysis: Roles, activities, and data requirements were identified using requirement analysis.
- Entity Identification: Specified organizations as well as their characteristics are identified as entities.
- Relationship Mapping: Triple as well as recursion connections are examples of established relationships.

- Cardinality Specification: To specify cardinalities, use Chen notation.
- Data Dictionary Creation: Created a data dictionary with properties and data types documented.

Conceptual Phase:

Introduction:

The idea behind this effort is to provide an extensive information template for Airbnb that takes into account the requirements of the system and guarantees a solid framework. The objective is to offer a framework that facilitates correct relationship representation within the system and effective data management.

Aim of the Work:

The goal is to create an all-encompassing ER model that supports Airbnb's capabilities and complies with its operational criteria.

Ideas and Sustainability:

The first concepts concentrate on enumerating all relevant entities and relationships in order to guarantee that the model can manage the anticipated volume and complexity of data. A organized system which improves efficiency is a plus, but maintaining triple along with recursion connections may be difficult at times.

Concept:

In order to implement the idea, a thorough ER model must be created, relationships, objects, as well as characteristics must be specified, and the requirements specification must be followed exactly.

Methodology and Tools:

The selected methodology makes use of Chen notation regarding relationship requirements together with ER modeling tools. Software for ER diagramming and feedback sessions are tools for improving the model.

Plan and Procedure:

The strategy calls for:

- Analyzing requirements is being done.
- Defining characteristics and entities.
- Defining cardinalities and charting relationships.
- Making a dictionary of data.
- Making use of suggestions for improvement.

6. Bibliography

Admin (2023). Comprehensive Guide to Entity-Relationship (ER) Modelling: With Example [Online] Visual Paradigm Blog. Available at: <https://blog.visual-paradigm.com/comprehensive-guide-to-entity-relationship-er-modeling-with-example/> (Accessed: 2024)

Anon(2023). What are the 7 Phases of Database Design? – Database Town. [Online] Available at: <https://databasetown.com/what-are-the-7-phases-of-database-design/> (Accessed: 2024)

Rodin, D. (2023). Chen ER Diagram - Entity-Relationship Diagram in Chen Notation - Software Ideas Modeler. [Online] www.softwareideas.net. Available at: <https://www.softwareideas.net/chen-er-diagram-erd> (Accessed: 2024)

Sheng, X. (2023). Metis: Building Airbnb's Next Generation Data Management Platform. [online] The Airbnb Tech Blog. Available at: <https://medium.com/airbnb-engineering/metis-building-airbnbs-next-generation-data-management-platform-d2c5219edf19> (Accessed: 2024)

SQL Easy Tutorial. (2023). what is ER Modeling? A Comprehensive Guide With Practical Examples - SQL Knowledge Center. [online] Available at: <https://www.sql-easy.com/learn/what-is-er-modeling/> (Accessed: 2024)

Visual Paradigm (2019). What is Entity Relationship Diagram(ERD)? [online] Visual-paradigm.com. Available at: <https://www.visual-paradigm.com/guide/data-modeling/what-is-entity-relationship-diagram/> (Accessed: 2024)