

CSD202 PROJECT

Aastha Dogra 1910110464

Aditi Rai 1910110029

Shambhavi Sud 1910110365

Objective: Application for Hotel Reservation at Holiday Destinations

Introduction and Working: Booking a hotel-room is a must-do job for anybody planning a trip to another place. Our application is something similar to the destination based Hotel reservation website called Trivago. Because of the fast speed of the digital world, most manual systems have been replaced by computer systems. Anything is at the touch of a button, from settling bills to buying clothing – there are no lines to wait in. Following the latest trend, our Application was created to link users with hotels in a potential destination. The customer must merely choose an alternative from the available choices, check his selection, and they may even cancel the reservation as per convenience. The aim of any program is to minimise, and in some situations remove, the difficulties that customers experience when dealing with manual processes. The programme in question has been optimised to prevent data entry errors. Invalid data is either entirely barred from entry or generates a code error as a result. All of this means that a client would not need any technological or programming skills in order to access this website – a concept that was kept in mind when creating this website, and steps were taken to achieve this goal.

First Webpage is the Home page, where the user enters details regarding their Location, Check-In Date, Check-Out Date, and the Room-Type from the list of options. Existing data matching the options selected by the user is displayed. User views a list of hotels, the various attributes like number of people the room can accommodate, WiFi, Parking availability, Pet-Friendliness and books the most suitable Hotel. A room at the same hotel can be booked multiple times, as per needs. On the page displaying the Hotel Room, the user may at the same time, add the room to their favourites as well as leave Rating and Comments.

On going to the Profile page the user can view their bookings, their bill per booking , their favourite hotels and the list of hotels they have reviewed. For cancelling their bookings, the user may click the go-to-room button and suitably cancel their bookings as well.

Features:

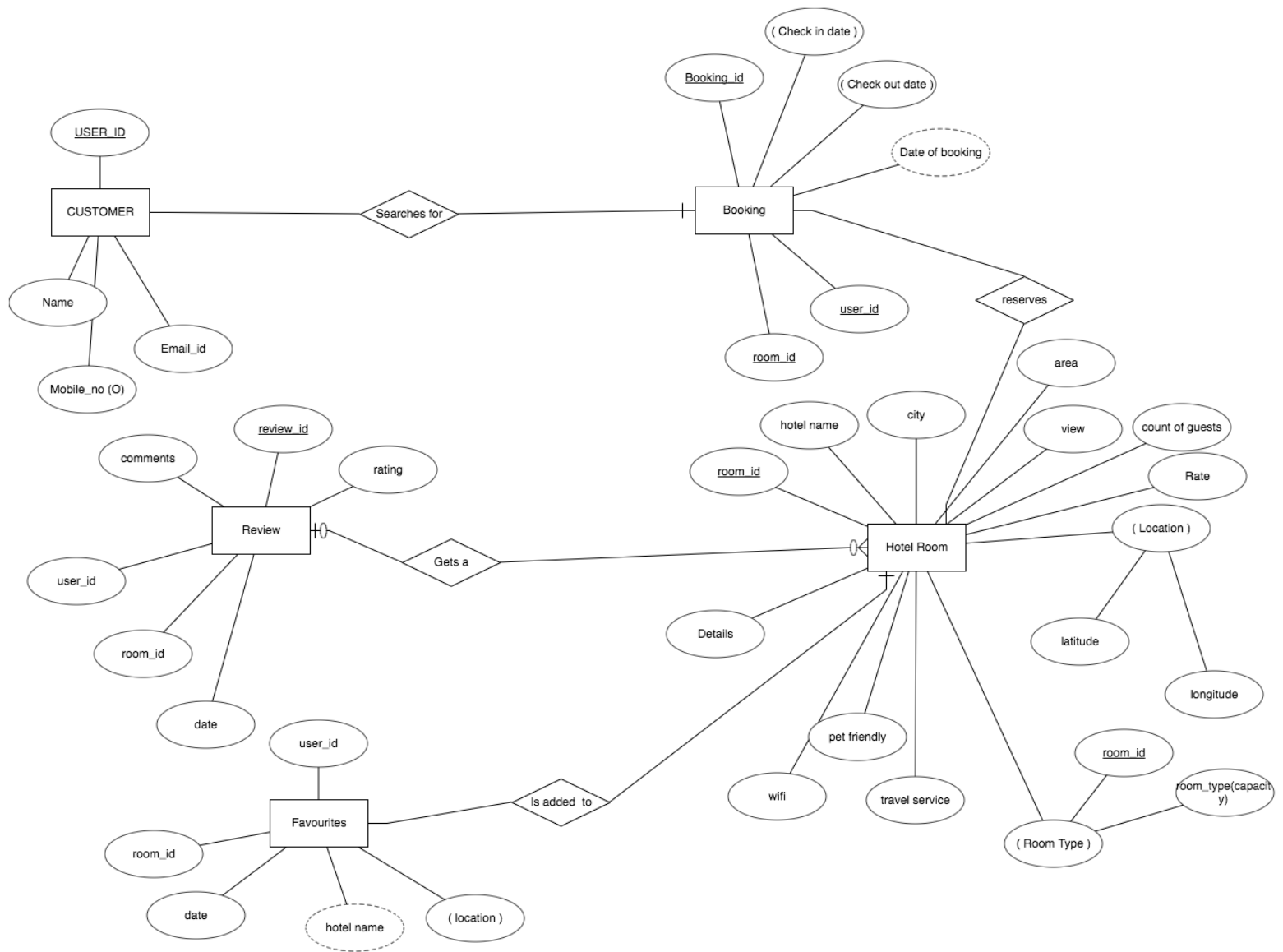
1. Booking: Making suitable bookings as per user's needs and preferences.
2. Cancellation : Being able to Cancel the booking in case of emergency.
3. Location: While viewing a hotel, the GUI also displays the real-time location of the hotel in Map(terrain) and Satellite View as per the option selected.
4. Rating & Review : The User has an option to give a star rating and review comments.
5. Add to Favourites: On clicking the heart-button the user may add the hotel to their list of favorite hotels.
6. Profile: The profile section of the user contains details of the rooms booked, their billing , the list of their favourite hotels and the list of their ratings of hotels they reviewed.

Limitations:

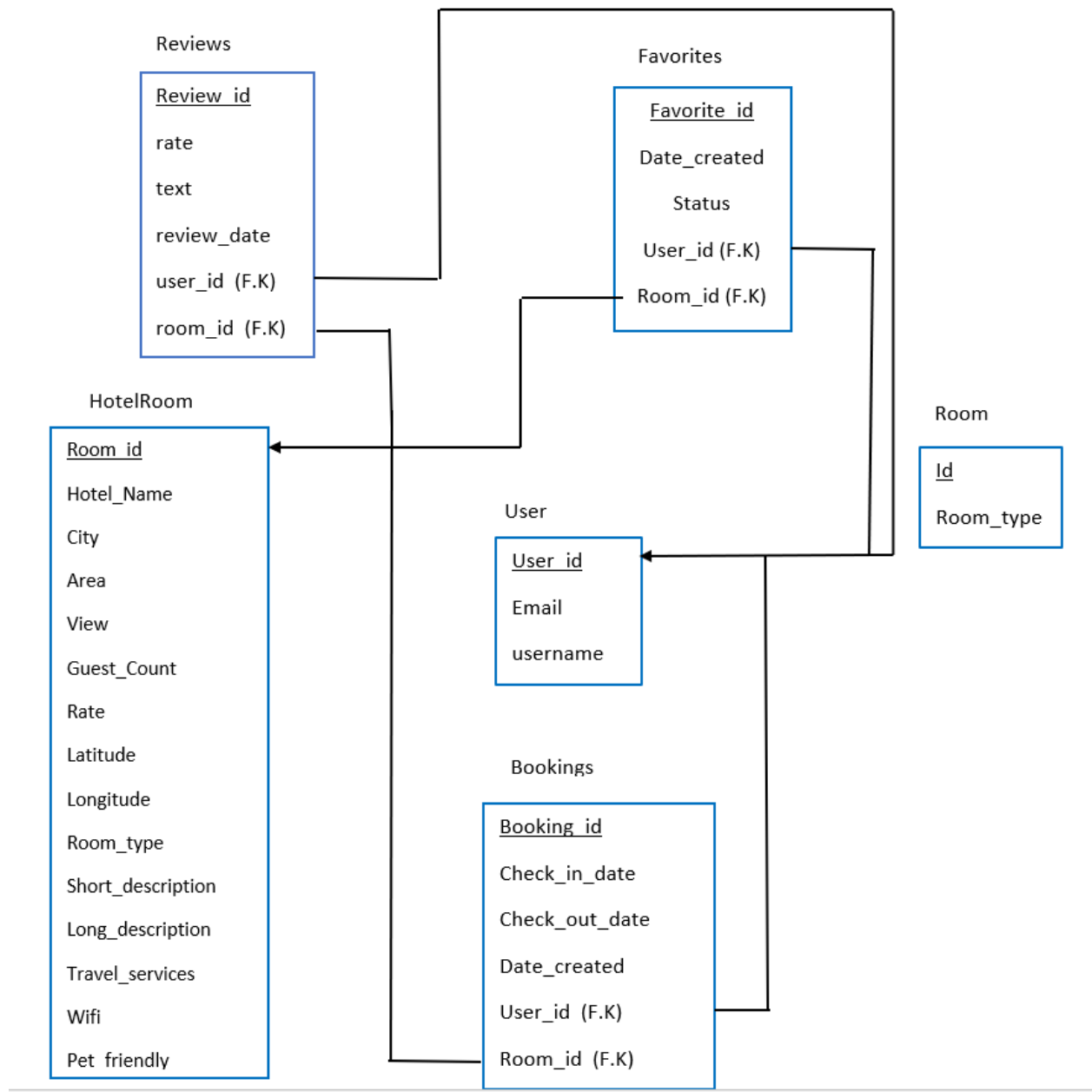
1. In terms of frontend , the project lacks an admin page and accounts for each hotel owner. The option of adding , deleting users, hotels etc is integrated within code, but we haven't created a full fledged website containing admin accounts to delete/ add various features or to be able to manage the Employee side of the website. Even though the database could be made, the part of implementing front -end functionalities took time to achieve. Making it in use only from user's perspective
2. The webpage lacks the ability to create a user account , the application is such that it caters to one user on whose computer the project is being run. Hence one can't create multiple profiles from the same computer.

This takes away the ability to make a payment gateway, or deleting the user account.

3. Doesn't contain Customer Chat support service in case the customer faces a problem, like a real-time hotel website.
4. Fake reviews cannot be moderated.
5. Does not include Privacy policy or a way to avoid user data breach



ER Diagram



SCHEMA DIAGRAM

TABLES

The database has 6 tables:

1. Bookings: Contains essential information regarding the booking made by user
2. Favorites: record of all the hotels the user pressed the heart icon on as a reference for really liking the hotel services
3. Reviews: Contains information about the rating and review given by the user, this is also displayed on the user profile.
4. Hotel Room : Consists details like services offered , location, room type being offered by the hotel.
5. Room_type : Consist of information regarding the type of room and id.
6. User :

TRIGGERS

1. REVIEWDISPLAY : show the list of reviews by the user on profile
2. Show Fav : show the list of hotels added to favorite by the user on profile

PROCEDURES

ShowBooking : shows the existing bookings of the user profile

NORMALIZATION

First Normal Form

The tables are in 1NF since we haven't used any multi-valued attributes in our tables. Hence it need not be brought to 1NF.

Second Normal Form

None of the tables contain composite keys and only a sole primary key is present, and all attributes can be derived from this P.K. Hence it appears that the tables are in 2NF. Hence it need not be brought to 2NF.

Third Normal Form

In order to satisfy 3NF the tables need to belong to 1NF and 2NF and should not contain any transitive dependency. The Hotel_room seems to be in Transitive relationship, since the Bookings depend on the Booking_id and that in turn depends on Room_id. Hence the Hotel_room table is transitive, Hence we split Hotel_room into two tables, Hotel_room and bookings. As Bookings table already exists with all the necessary attributes required by hotel_room table, splitting and linking of two tables suffices.

Conclusion: The project highlights the use of

- System analysis and design techniques like data flow diagrams.
- Database handling and query processing.
- Database Designing.
- User interface design
- Various features that make the application convenient for use.