Database Systems Project Proposal

Travel Agency Data Management System

20/10/2021

Project Group No: 10

Ahmet Salman, 21901004

Muhammed Can Küçükaslan, 21901779

Mustafa Yasir Altunhan, 21802530

Güven Gergerli, 21803393

Course Instructor: Özgür Ulusoy

Table of Contents

Introduction

This report will outline the features and functionalities and why we need them for the Travel Agency Data Management project. The aim of the project, constraints, and limitations will also be touched upon.

This proposal will be divided into 4 parts

- Project Description
- Requirement (Functional and Non-functional)
- Limitations
- E-R model of our database

In the project description segment, the desirable behavior of the Travel Agency Data Management will be discussed such as what users will be able to do, what activities can be planned, canceled, postponed, etc.

In the requirements section, the functional and non-functional aspects will be discussed. Functional refers to what the system is supposed to do, while non-functional aspects refer to how that system is supposed to achieve the functional part. Functional requirements can be analyzed using user input (logging in, making a reservation, etc.). Non-functional requirements are mostly about back-end implementation (maintaining sessions, tokens, managing the database, etc.).

In the limitation part, as the name implies, the limitation and constraints of our system will be discussed in depth.

And finally, an E-R diagram will be constructed to reflect our database design

Project Description

The Travel Agency Management System aims to keep the data available and maintain fine and efficient queries in the process.

The system will allow the user to store, retrieve, and modify data relating to tours, tour guides, hotel reservations, and other things. An authorization system will also be implemented such that it gives the employees the authority to change reservations and assign tour guides, while customers only have the authority to manage their own reservations and book a tour for themselves.

The database will support different types of users. The major ones are customers, tour guides, and employees. A customer can book a hotel room and reserve a tour if they wanted, while tour guides can accept or reject tour offers. Finally, Employees have the authority to change the reservation plans of a customer and assign certain tour guides to tours.

Each one of those users is identifiable by a unique ID.

For the tours and hotel reservation, tours can be uniquely identified by the tour guide ID and start time of the tour, a combination of these 2 attributes will always return a unique tuple. Reservations can also be uniquely identified by the customer ID and their stay date.

The final segment is about commenting, a customer can comment on their experience during the tour, likewise, a tour guide can comment on a tour after it has been concluded.

Why Do We Use a Database

The Travel Agency Management System is strictly dedicated to storing and managing data about customers, tour guides, and employees in the context of assigning roles to each of them. The reason for choosing this database is that the only information being stored relates to the 3 types of users we have, this will offer efficient queries. Our database removes the concern of space and time complexity because it does not include any irrelevant data. When fully implemented, our system will be suitable for usage for larger-scale Travel Agency companies.

How Do We Use a Database

The database will be used to store relevant information about the users, managers, tours, and tour guides along with any other information that is necessary for the development of a functional Travel Agency Management System. Queries will be executed against the database in order to retrieve necessary information so they can be displayed correctly in the frontend. However, the session information of each user won't be saved in the database, tokens will be used to maintain sessions between the client and the web app. Moreover, the password credentials of the users won't be saved in clear text, but rather we will implement hashing algorithms and hash salt which will be added to the user password in order to store it safely.

Functional Requirements

Customer

- Customers can manage hotel room reservations
  A customer can book a hotel room in a hotel. He/she can cancel or postpone/change an existing booking. He/she also can see the list of his/her previous bookings.

- Customers can manage reservations for tours
  A customer can make a reservation for a desired tour in the list. He/she can cancel or postpone/change an existing reservation. He/she can see the list of old reservations. He can choose extra activities in the tour.

- Customer can give feedback on the tour and the tour guide
  A customer can rate and leave comments on the tour and the corresponding tour guide.
- Customer can see the ratings and comments of tours and tour guides.

Hotel Employee

- An employee can manage booking for customers
  A hotel employee can book a hotel on behalf of a customer. He/she can cancel or postpone/change an existing booking.

Travel Agency Employee (Receptionist)

- Employees can make offerings to guides for tours.

- Employees can manage tour reservations for customer
  An employee can make a reservation or cancel a reservation.

Travel Agency Administrator (tour creator)

- Administrators can manage tours
  An administrator creates new tours or cancels existing ones. He/she can divide tours to subsections each can be assigned to a tour guide.

Guides

- Accept or decline an assigned tour (i.e. tour offerings) of a TA Receptionist

- Can rate and comment on the tour

- Feedbacks on previous works will be listed on the tour page

Tour

- A tour consists of 1+ days
  - Each day is associated with sightseeing places that will be visited
  - There might be extra activities on tours

- A tour can have subsections with different guides (for large organizations)

- Feedbacks on previous versions of a tour will be listed on the tour page including both customer and guide feedback.

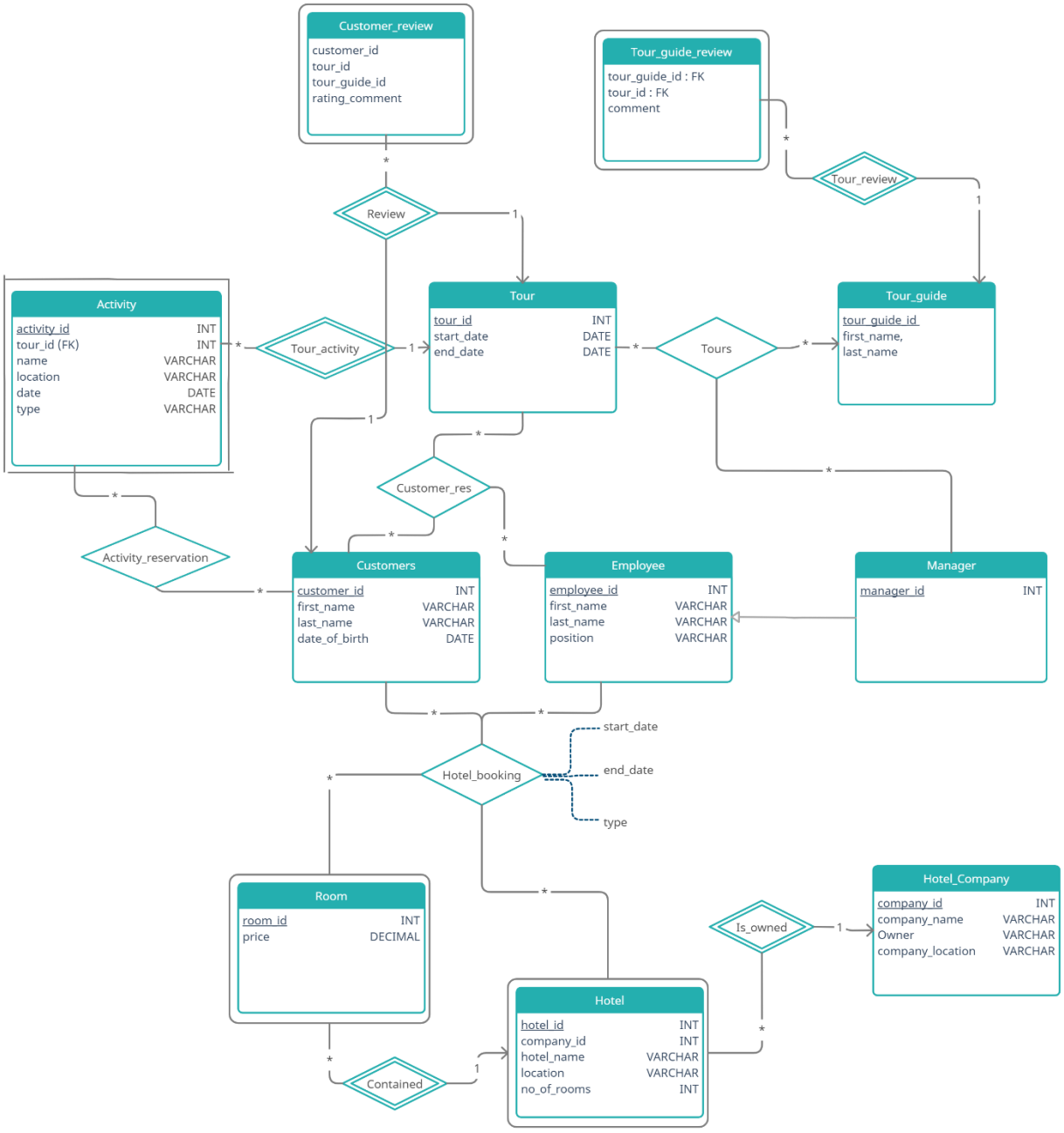
Non-Functional Requirements

- The query will execute in less than 5 seconds
- The web page will display the query result in less than 10 second
- In case of app failure, no data will be lost
- The system will ensure coherence during database manipulation
- The system will be able to handle requests from at least 100 users simultaneously
- The system will allow for at least 1000 users to enroll and execute queries quickly

Limitations:

- A tour guide can only be present in one tour at a time (no overlapping times for the same tour guide)
- A customer cannot book a hotel or reserve a tour guide if they are not signed in
- Managers cannot assign tour guides to tours that have already passed
- Tour guides cannot comment on a tour they have not managed
- Customers cannot comment on a tour they have not attended
- Guides can only accept/reject tour guides assigned to them
- Tour guides cannot accept requests that have passed already
- Customers can comment on a tour guide of a tour they attended
- A customer can only reserve extra activities

E/R Model



Project Web Page

https://muhammedcankucukaslan.github.io/CS-353/