CSI2132 – Database I

Winter 2020

Project

Announced: January 26th 2020

Submission of 1st deliverable: February 10th 2020

Submission of 2nd deliverable: March 29th 2020

Project presentation: March 30th 2020

Project Overview:

The aim of this project is to gain experience in the design of an enterprise relational database as well as its operation and maintenance using PostgreSQL DBMS. The application domain considered for this project is a real-world online travel enterprise that arranges accommodation or lodging for privately owned properties. You are required to develop, as part of a new information systems project, a new database. A vague description and minimum requirements of the enterprise are presented under the application description section. Please note that extensions to the minimal requirements identified are highly encouraged. A number of enterprises have been suggested to start you off, however, you are allowed to choose any other enterprise that fits the application description. Before you kick off, you will need to visit the website of your selected enterprise in order to gain the domain knowledge required to create a precise design and concrete implementation of the interfaces or applications that will use the database. To design the database, you will need to properly identify all the entities, attributes, relationships and constraints. The project should be carried out by a group of at least 2 (+ 1) students to ensure project completion before the submission deadlines.

Application Description:

The considered enterprise application is an Online Travel Marketplace offering short-term lodging, primarily homestays or tourism experiences. The marketplace provides an excellent platform to search and book a vacation rental while property owners increase their income. Options can range from various locations within the city to unusual places, beaches and historic sites. The enterprise does not own any of the property listings (i.e., only act as brokers, receiving commissions for each booking). To give you a starting point, here is a list of the enterprises you can model but other choices are allowed: Airbnb, TripAdvisor, Trivago, Expedia, Booking.com, VrBO, Homeaway and Tripinn. These enterprises need databases to track various information “not limited to” all types of rentals including apartments, luxury resorts, family run homes, and even reviews from previous users. Various user interfaces and several applications that run on their own access the database to create listings, check availability, analyze the data, etc. Here are a few minimal requirements these companies need you to consider (which can be extended to fit your selected enterprise):

Host: Every property host needs to be tracked. The host details such as the address (house number, street, city, and province), host's name (first, middle, and last name), and email addresses and phone numbers are also required. Note that on some platforms, a registered host can also be a guest (i.e., they can post accommodation as well as book accommodation), e.g., Airbnb.

Guests: A guest should be able to book more than one property. They want you to capture details such as their ids, address, name, email address and phone numbers.

Rental Agreement: A rental agreement is needed for a property. Each such rental agreement should be identified in the database as well as the signing, start and end dates of the rental agreement.

Property: Hosts should be able to list their properties stating the address and property type (e.g., apartment, bed & breakfast, unique homes, and vacation homes, etc.), room type (unique space, private or shared room), accommodates, amenities, bathrooms, bedrooms, beds & type), available date, location, price, etc. Other attributes discovered from your selected enterprise can also be added.

Branches: The enterprise is organized into branches, represented by the country in which the properties are located (e.g., Airbnb is present in 190 countries, therefore, it has 190 branches). Every branch has employees allocated with a branch manager. Each branch details need to be captured by the database.

Employees: Every employee needs to be identified by the basic information, position and salary. Note that the properties in each country the enterprise resides are run by the employees who are managed by a branch manager, who is also an employee.

Pricing (or Renter Rate): Pricing is determined by the hosts. They provide the prices and other details for their rental or event listings, such as the allowed number of guests, home type, rules, and amenities, etc. Each property has its own pricing, but all properties in the same class are priced the same.

Payment: Hosts are able to collect payments for the accommodations online. You are required to keep track of the host that accepted the payment, the type of payment (i.e. cash, check, credit card, direct debit), the amount of the payment, payment status (i.e., completed (cash), approved (debit or credit card), or pending).

Reviews: The enterprise also needs to keep track of the reviews from the guests. Information that needs to be stored include ratings, communications, cleanliness and value.

Project Requirements:

E-R Model

Construct an E-R diagram representing the conceptual design of the database. Since there are many variations of the original version (which represented relationships as diamonds), you can use other forms you prefer, e.g., IDEFIX, IE CROW's foot model, etc. At a minimum, you must include all the entities and relationship sets implied, and they should not be assumed as completely identified under the project requirements. Be sure to identify the primary keys, relationship cardinalities, etc. You can make your design complete by studying the website of your selected online travel marketplace enterprise.

Relational Model

After constructing your E-R Diagram, you need to translate it into the relational database design. Define the necessary constraints that will ensure the correctness of the database to be created according to your Relational model. These are primary keys, referential integrity constraints, domain constraints and user-defined constraints. Implement the relations in PostgreSQL and make sure you create indices and constraints as appropriate. Whenever you discover flaws that require changes to your E-R Diagram, make sure these changes are captured in your relational model.

Populate Relations

Your database needs to be populated with data after creating it. Data can be sourced online from sites such as open data, Kaggle, etc. You can also decide to write a program to automatically populate your tables or randomly generate the data. Adding enough data will make your queries interesting.

Queries:

You need to run a number of test queries to see that you have loaded your database in the way you intended. Here is a list of queries the enterprise will be interested in:

1. Give the details of all the *guests* who rented properties. Please display the columns as *guest name, rental type, rental price*, *signing date*, *branch, payment type* and *payment status*. Sort by the *payment type in ascending order and signing date in descending order*.
2. Create a view named *GuestListView* that gives the details of all the guests. Please, sort the guests by the *branch id* and then by *guest id*.
3. Display the details of the cheapest (completed) rental.
4. List all the properties rented and sort based on the *branch id* and *review rating*.
5. Find the properties that are already listed but not yet rented. Please, avoid duplications.
6. List all the details of all properties rented on the 10th day of any month. Ensure to insert dates in your table that correspond in order to run your query.
7. List all the managers and the employees with salary greater than or equal to $*15000* by their *ids, names, branch ids*, *branch names and salary*. Sort by *manager id* and then by *employee id*.
8. Consider creating a simple bill for a guest stating the *property type*, *host, address, amount paid* and *payment type*.
9. Update the phone number of a guest.
10. Create and test a user-defined function named *FirstNameFirst* that combines two attributes of the guest named *firstName* and *lastName* into a concatenated value named *fullName* [e.g., *James* and *Brown* will be combined to read *James Brown*].

Interfaces:

Several users will need to make use of the database and each will require special application during access. (Since it is not a web application development course, a simple command line interface will suffice but you can develop an application if you are able to).

- Database administrator will need to use SQL either through the command line or SQL Developer.
- Hosts and guests need a web interface to upload their listings and to check availability and rates respectively.

- Branch employees need a lookup application that allows quick access to property listings in order to know the occupancy rate and organize the operations of the properties.

Project Deliverables:

1st Deliverable (due February 10th 23:30):

Please submit a report that includes the following:

1. (20%) E-R diagram.
2. (10%) Relational schema (please do not print out your data).
3. (10%) The constraints that you have defined. Please include a brief justification for each constraint.

2nd Deliverable (due March 29th 23:30):

Please submit a .zip file that includes the following:

1. A report that includes the following:
    a. (10%) The DBMS and the programming languages that you have used in your implementation of the application.
    b. (10%) A list with the DDLs used to create your database
    c. (5%) Specific steps to guide someone to install your applications.
2. (15%) Your SQL code that supports all the functionalities in your application
3. (20%) The code necessary to implement all the user interfaces. (The use of basic interfaces is acceptable but you are encouraged to design more sophisticated ones if you can).

Please upload your deliverables on Brightspace by the due dates stated at the beginning of this document. Each group will be allocated a 20-minute slot to present their project on the 30th of March 2020 (to be confirmed). Please come with your application installed and running on your laptop for your project presentation.