WhereToGo Trip Advisor Report

1. Briefly describe what the project accomplished.

The project "Where to go" is built with SQL, python flask, and other supporting language and technical components. The user interface on the website enables users to view and search suitable destinations with preferences and check the features of their predetermined destination. Those user-friendly functions and enjoyable user experience are realized by our design of SQL queries, comprehensively searching for the raw data, carefully filtering and logically merging databases, innovative design of UI, and advanced SQL functions. For each aspect, we will illustrate it thoroughly in the following sections.

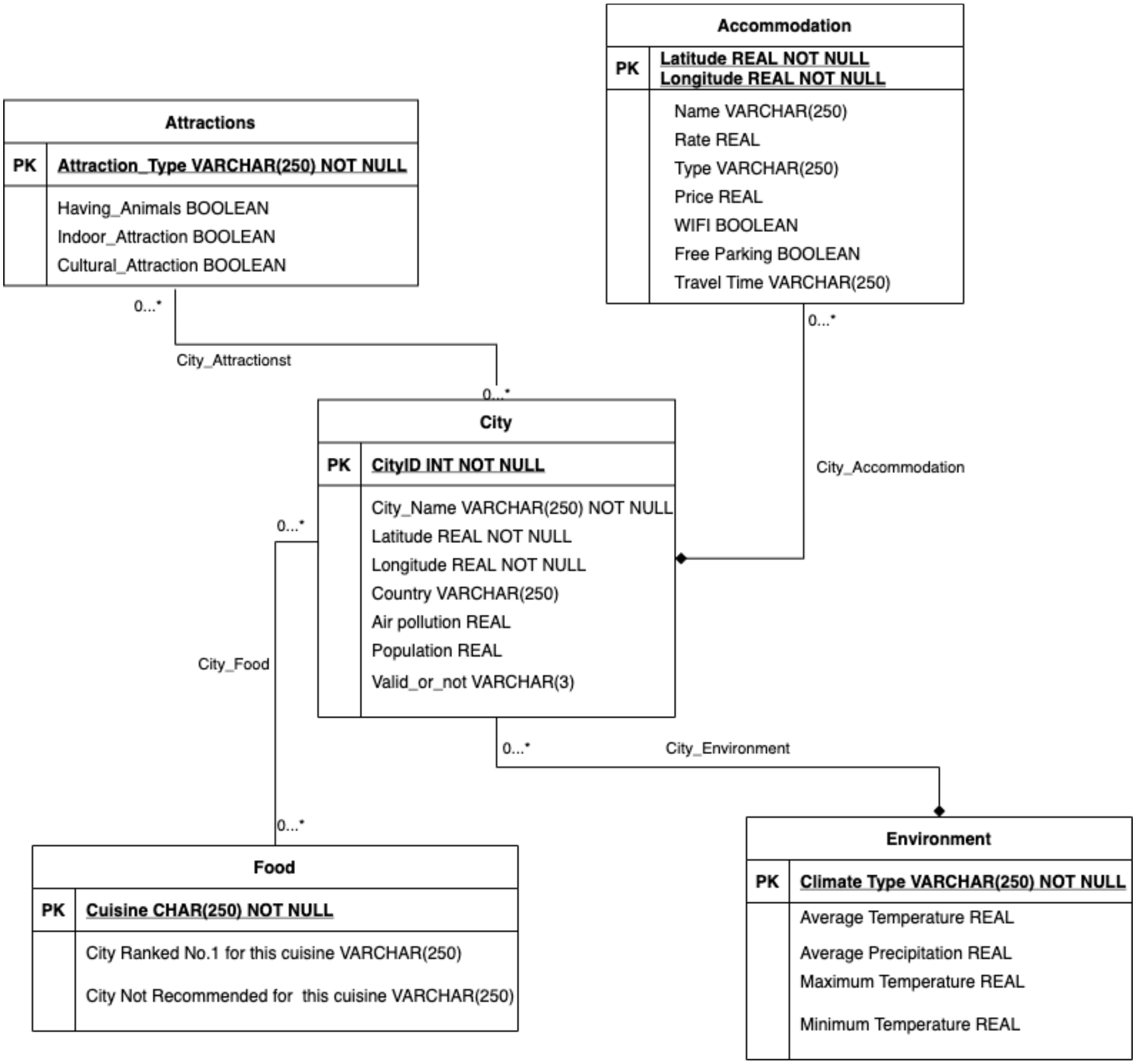
2. Discuss the usefulness of your project, i.e. what real problem you solved.

The tourism industry has entered the rising trend with the pandemic trend going weaker and the increasing vaccine-taking rate. Research conducted by Hong Kong City University shows that the world's travel destinations are focused on only 39 cities and many participants expressed their difficult time in choosing destinations and ended up randomly choosing a famous city. To solve this problem, we help people to combine their preferences and minimize the range of potential choices. With entering the preference via our website, the potential cities meeting certain requirements are shown so that users are equipped with a helpful tool to make a meaningful decision.

3. Discuss the data in your database.

The features include Accommodations, Environments, Cuisine, Attractions of a city together with its own characteristics. The point we'd like to explain further is the importance of the 'population' feature in our database. According to statistical facts of normal distribution and Covid-19 infection rate model by John Hopkins, the rate is positively related to population size and density. Hence, we provide this attribute for users to make decisions based on their attitudes toward COVID-19.

4. Include your updated ER/UML Diagram.



5. Include your final database schema (DDLs) and index design analysis.

Relational Schema:

City(City_ID:INT [PK], City_Name: VARCHAR(250), Latitude:REAL[FK], Longitude: REAL [FK], Country: VARCHAR(250), Airpollution: REAL, Population: REAL, Valid_or_not: VARCHAR(3))
Accommodation(Longitude: Real [PK] Latitude Real: [PK], Name: VARCHAR(250),: Rate: REAL, Type: VARCHAR(250), Price: REAL, WIFI: BOOLEAN, Free Parking BOOLEAN, Travel time: VarCHAR(250)
Food(Cuisine: VARCHAR(250) [PK], City Ranked No.1 for the Cuisine: VARCHAR(250), City Not Recommended for the Cuisine: VARCHAR(250)
Environment(Climate Type: VARCHAR(250) [PK], Average Temperature: REAL, Average Precipitation: REAL, Maximum Temperature: REAL, Minimum Temperature: REAL)

Attraction(Attraction_Type: VARCHAR(250)[PK], Having_Animals: BOOLEAN, Indoor_Attraction: BOOLEAN, Cultural_Attraction: BOOLEAN)
City_Attractions(CityID: INT [PK][FK to City.CityID], Attraction_Type: VARCHAR(250) [PK][FK to Attractions.Attraction_Type])
City_Food(CityID: INT [PK][FK to City.CityID], Cuisine_Type: VARCHAR(250)[PK][FK to Food.Cuisine_Type])
City_Environment(CityID: INT[PK][FK to City.CityID], Climate_Type: VARCHAR(250)[FK to Environment.Climate_Type])
City_Accommodation(Latitude: REAL[PK][FK to Accomodation.Latitude], Longitude: REAL[PK][FK to Accomodation.Longitude], CityID: INT[FK to City.CityID])

The index analysis was completed at previous stage and we used the default index design in MySQL considering the good performance based on the previous analysis.

6. Briefly discuss where you collected data and how you did it (if crawling is automated, explain how and what tools were used).

The data are collected from reliable academic websites and official statistics websites. Some missing data are filled with the random forest machine learning model, with existing non-empty data as training data. The update of our database with the provided function will also be signed by procedure and validated by our team. Hence, the precise database can succeed to a certain extent.

7. Briefly discuss your application design and the features involved. Clearly list the functionality of your application (feature specs).

As stated previously, the goal of the “Where To Go” application is to provide users a platform where they can look for their travel destination with their own selected preference. Like most recommendation websites, the website starts with a table listing the most popular travel cities in the world along with their location, population, country, etc. Below the table, user exploration space (e.g., search bar) and other ideas/ inspiration for users’ reference can be found. For example, for users who are in love with nature, there is a one-click button that recommends the tropical cities where attractions with animals can be found. Aside from this, through a one-click button, the users can have access to the accommodation conditions of each city in terms of ratings, WIFI availability, etc.

Specific features include creating, read, update and delete in our application. Over the city table, users can add their cities to the table if not listed or delete the cities which are not their cakes. A special feature involved in our application is that users can update the city population over the table if any out-of-dated data is found. However, an attribute that identifies the validation of the population is also provided. With this feature, we would like to improve our database by using up-to-date information from the user end without losing any credibility. The search/ read function is realized through a search bar where users can search for cities and get city information. Other features including travel ideas, evaluation of local accommodations, etc. can also be found.

8. Include your advanced database program. Discuss your choice for the advanced database program and how it is suitable for your application. For example, if you choose transaction+trigger, explain how this choice is suitable for your application, the isolation level of your transaction, etc.

Stored procedure and trigger are selected for our application.

A stored procedure is used to evaluate local accommodations in each city. The accommodation database is expected to update from time to time (e.g., new accommodation is built and the old one is closed, rating changes, price fluctuations, etc.) and such stored procedure can keep our evaluation up-to-date coincidently without much effort.

The trigger is used when population data is updated through users. As stated above, our application allows users to update population data through their end. A trigger feature is added to identify the validation to avoid users randomly updating unreasonable population data for the city. Application developers are expected to check such updates and validate the information. We view this as a good way to keep interactions with users under the assurance of credibility.

9. List and briefly explain the data flow, i.e. the steps that occur between a user entering the data on the screen and the output that occurs (you can insert a set of screenshots).

There are three places on our website where users can input their data.

- **Add a city.**

  Users can add a city that is not listed in our table by themselves. After entering, a new record with assigned ID and city name will be inserted into our Cities database and also displayed over the UI interface. Other information about this city will be recorded as NULL. The user input is designed to be minimum in this function to keep the credibility of the website.

- **Update city populations.**

  Users can enter a population to any city listed in the table and the city population data will be updated at our Cities database with UI interface display. Along with this update, a trigger will also be activated and the attribute (Valid_or_not) for this city in Cities database will be changed to “NO” which will also be displayed. After this update, no front-end commands can change the “NO” mark for validation, and application developers are expected to validate the information and update corresponding records.

- **Search for a specific city.**

  Users can enter a city name at the search bar and basic information of this city including location, the population will be displayed.

10. Describe one technical challenge that the team encountered. This should be sufficiently detailed such that another future team could use this as helpful advice if they were to start a similar project or where to maintain your project. Say you created a very robust crawler - share your knowledge. You learned how to visualize a graph and make an interactive interface for it - teach all! Know how to minimize time building a mobile app - describe!

The first technical plight we might have was the constitution of the SQL database from CSV. It was a great step from just understanding SQL language and running it in a certain environment with a built database to edit and run it.

To overcome this strait, what we did was to understand the relationships among every single sheet first and employ a traversal operator in python to write CSV in the SQL database for future queries. The sample Python codes and SQL server instructions played an important role in guiding us through the way to build the database.

Secondly, the UI was a hard obstacle for us to realize the application. The connection between the forefront and back end was a problem. Among the three recommendations, we chose python flask because of our familiarity with python.

One of the predicaments was that the results were not shown on the forefront website page. However, the terminal presented that the search was performed and output was obtained. We first thought it may be caused by the empty output lists since there was no content expression of the output. But when we checked the database manually, multiple outputs should be exhibited on the website. Then we turn to work on the methods performing the corresponding query. During the debug process, we noticed that the returning object was not in the correct type to show on the website. Hence, we added a step to transform it into a suitable type for the forefront. Therefore, for future teams, we strongly recommend them to also pay enough attention to the parts of codes which realize the operation of UI-backend connection, especially the types and interface implementation issues.

11. State if everything went according to the initial development plan and proposed specifications, if not - why?!

The project's goal and main technological structures strictly follow the initial plan decided by all of the group members. We design all features and functions to provide users thorough results, convenient usage, and enjoyable user experiences, providing professional travel advice and enhancing the happiness of tourism at the same time.

We also had certain improvements on several aspects of the application based on the initial plan. Instead of purely depending on users' subjective preferences, we also made additional recommendations. First, we added, "Are you a natural guy" to provide a list of potential destinations and their features for travelers who prefer parks with good climates.

In addition to the natural world, we also offer quick recommendations for users who are interested in the cultural universe. The museums can constitute a prosperous window to the internal world of certain cultures. Hence, we add a query realized by GROUP BY to exhibit the countries with more than three of its cities having "museums".

To be concise, we followed our initial plan and added functional features in consideration of user experience and the instruction of project track which could be super helpful for the project.

12. Describe the final division of labor and how well you managed teamwork.

We distributed the total labor wisely and separated development and back-testing to ensure the operation consistency of the project and the precision of the output. [Dividing the whole team into two parts and taking responsibility for test parts developed by the other two members]. Specific labor division followed our arrangement at the beginning of the course.

The only difference was one of our members dropped the course because of the schedule modification. The new member originally from the other group joined us and played one of the roles to complete our project. Because of the joining time, he spent much of his time getting familiar with our content and was mainly joining us with the back-testing part. This arrangement was based on our main idea of separating developments and back-testing.

To ensure the process of development goes well and successfully meets each stage checkpoint, we always have meetings via zoom to check the progress of each member, work on technical issues together, and discuss the new potential features.