PostgreSQL

Account: ym2991

Password: ym2991

URL of web application: <http://35.243.171.181:8111/>

Description of proposal implementation:

* **Itinerary Planning**: Users can create new itineraries and view existing ones.
* **Destination Discovery**: Users can search for specific destinations by typing in the search box. Details including related flights, accommodations, and activities will be provided.

This part is slightly different from original proposal that users utilize search function instead of directly clicking on destinations.

* **Interactive Features**: Users can rate and review itineraries they have booked.
* **Recommendation System**: Users can get recommendations on the flights, accommodations, and activities based on the user’s itinerary plan (destination, budget & duration)

Things interesting about the web pages:

* **Recommendation**: We require users to input destination, budget, and duration and give the recommended flights, accommodations, and activities. The destination input is used to retrieve flights to the destination, and accommodation and activities located in the destination. Then we calculate the total price of flights, duration times the rate of accommodation, and activities, which is compared with the user’s budget. Therefore, we can give the matched flights, accommodations, and activities to the user. The Recommendation page is particularly interesting because it offers a highly personalized querying service that involves several dimensions of data. It's not merely about matching static database entries but about synthesizing multiple dynamic factors such as pricing, user preferences, and date-specific availability.
* **Rate and review**: We require users to input their traveler id, itinerary id, rate, and review in the page, and they can also view the existing reviews. Traveler id and itinerary id are used to check if the user indeed booked the itinerary through existence in the table Books. When it is confirmed, we will insert the new rate and review into our database, and users can view the rate and review they created. The requirement for a traveler ID and itinerary ID as inputs to leave a review ensures that feedback is authentic and tied to real experiences. The verification step against the Books table to confirm that a traveler has indeed booked the itinerary is an excellent example of data integrity enforcement. It prevents fraudulent reviews and maintains the reliability of the feedback system. From a database operations perspective, dynamically inserting reviews and instantly displaying them back to the user without page refreshes requires a responsive database setup.