

Travel Guide Application

Index

1. Entity Relationship Description and Application Basic Structure

1.1. Entities

- Client
- Destination
- Questionnaire
- Suggestion

1.2. Relationships

- Client – Questionnaire (Fills Out)
- Questionnaire – Suggestion (Generates)
- Destination – Suggestion (Is Suggested As)

1.3. Working Mechanism

2. Entity Relationship Diagram

3. Source Code Relational Schema

4. Project Members and Github Repository

1. Entity Relationship Description and Application Basic Structure

1.1. Entities

- Client
 - Represents each user who registers on the application.
 - **Attributes:** client_id (PK), first_name, last_name, email, password
- Destination
 - Represents each travel destination.
 - **Attributes:** destination_id (PK), name, average_cost, best_travel_time, average_weather
- Questionnaire
 - Represents each set of answers provided by the user.
 - **Attributes:** questionnaire_id (PK), client_id (FK), budget, weather_preference, travel_start_date, travel_end_date
- Suggestion
 - Represents travel destination suggestions for each user.
 - **Attributes:** suggestion_id (PK), questionnaire_id (FK), destination_id (FK), match_score

Travel Guide Application

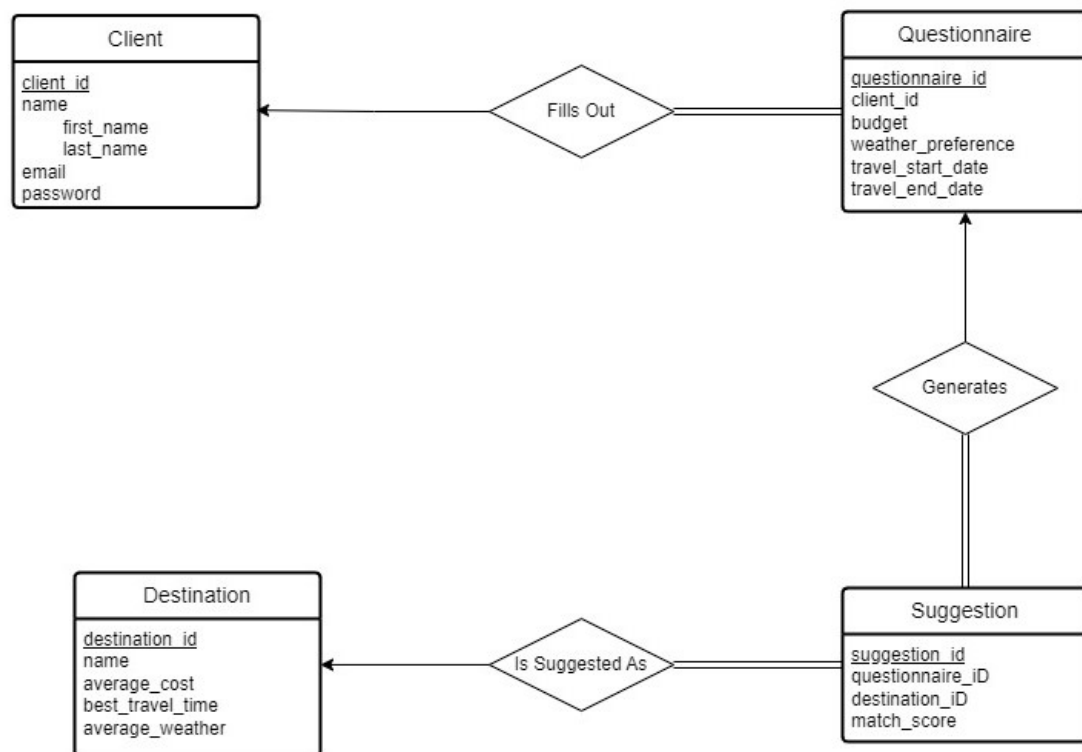
1.2. Relationships

- Client – Questionnaire (Fills Out)
 - One-to-Many from *Client* to *Questionnaire*: one user can fill out multiple questionnaires over time. Each *Questionnaire* entry corresponds to exactly one user.
- Questionnaire – Suggestion (Generates)
 - One-to-Many from *Questionnaire* to *Suggestion*: based on a single questionnaire, multiple travel suggestions can be generated. Each suggestion corresponds to exactly one questionnaire.
- Destination – Suggestion (Is Suggested As)
 - One-to-Many from *Destination* to *Suggestion*: one destination can be suggested in multiple suggestions, but each suggestion points to exactly one destination.

1.3. Working Mechanism

- Users register and login.
- Users fill out a questionnaire detailing their travel preferences.
- Our application will process this data and compare it with the attributes of various travel destinations. Based on matching criteria (like budget, preferred weather, and travel dates), the application generates a score for each destination for the specific user. The destinations with the highest scores will be suggested to the user, which will be saved as *Suggestion* entities linked to the specific *Questionnaire* entity.

2. Entity Relationship Diagram



Travel Guide Application

3. Source Code Relational Schema

```
CREATE TABLE client (  
    client_id SERIAL PRIMARY KEY,  
    first_name VARCHAR(255) NOT NULL,  
    last_name VARCHAR(255) NOT NULL,  
    email VARCHAR(255) NOT NULL UNIQUE,  
    password VARCHAR(255) NOT NULL  
);  
  
CREATE TYPE temperature_preference AS ENUM ('Freezing', 'Cold', 'Cool', 'Mild',  
'Warm', 'Hot', 'Very Hot');  
  
CREATE TABLE questionnaire (  
    questionnaire_id SERIAL PRIMARY KEY,  
    client_id INT NOT NULL,  
    budget DECIMAL(10, 2),  
    weather_preference temperature_preference,  
    travel_start_date DATE,  
    travel_end_date DATE,  
    FOREIGN KEY (client_id) REFERENCES client(client_id),  
    CHECK (travel_end_date > travel_start_date)  
);  
  
CREATE TABLE destination (  
    destination_id SERIAL PRIMARY KEY,  
    name VARCHAR(255) NOT NULL,  
    average_cost DECIMAL(10, 2), -- average cost per day  
    best_travel_time VARCHAR(255), -- Stores lists like "January,February,March"  
    average_weather temperature_preference  
);  
  
CREATE TABLE suggestion (  
    suggestion_id SERIAL PRIMARY KEY,  
    questionnaire_id INT NOT NULL,  
    destination_id INT NOT NULL,  
    match_score DECIMAL(5, 2),  
    FOREIGN KEY (questionnaire_id) REFERENCES questionnaire(questionnaire_id),  
    FOREIGN KEY (destination_id) REFERENCES destination(destination_id)  
);
```

4. Project Members and Github Repository

- Mohamad Dib Fares (A20482852)
- HAMZA Taheir BU OBIAD (A20500711)
- Mehal Gosalia (A20484633)
- Rajdeep Singh Konthoujam (A20493036)
- Github repository: <https://github.com/Moody162/Travel-Guide/tree/main>