Documentation

**Approach Overview:**

The recommendation system implemented is a KMeans-based clustering model designed to segment users based on their travel preferences and behaviors, and provide city recommendations accordingly.

Key Features:

1. Vibe
2. Expenses
3. Previous Destination
4. Favorite City

Steps taken to train Model

1. Data Preprocessing – Encoding and Standard scaling is done on the features vibe and expenses.
2. Training – Kmeans Algi is trained on a combined feature set of encoded user vibes and scaled expenses to segment users into clusters.

Model Approach

1. When a new user makes a request, the system determines which cluster this user belongs to by comparing the new user's preferences to the pre-trained clusters.
2. Based on the cluster the user is assigned to, the system recommends the most suitable city, places to visit, restaurants, and activities.

**Backend Design**

Overview:

The backend API is designed using FastAPI, a modern, fast (high-performance) web framework for building APIs with Python 3.7+. The API serves the recommendation system and provides endpoints for generating recommendations based on user inputs.

Endpoints:

1. /recommend:

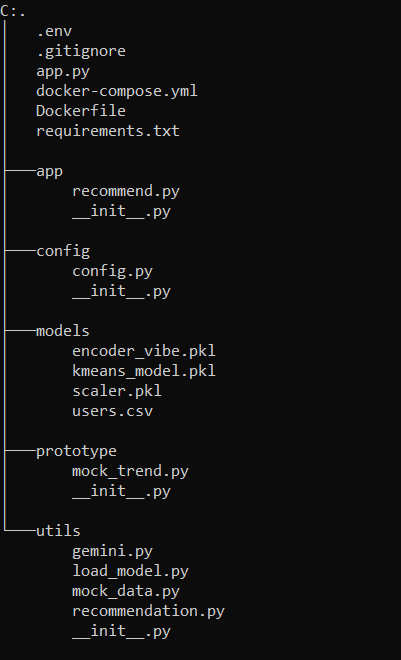
* Method : POST
* Desc : This endpoint takes the user's travel preferences (vibe, expenses, and season) and returns the most appropriate city and its itinerary (places to visit, restaurants, and activities).
* Required Params :
  + Vibe
  + Expenses
  + Season
* Response :
* Itenary : Consists of personalized itenary.

Architecture

Components:

* FastAPI – Serves as backend
* Kmeans Model – ML model for the application
* Recommendation engine – Logic to recommends based on the user cluster

Backend Structure :



**Potential optimizations**

Potential Optimizations for the AI Model

To improve the recommendation system, especially with more real-world data or added complexity, we can explore several strategies:

1. Incorporating Additional Features

To enhance the clustering accuracy and recommendation quality, we can introduce more features that capture user preferences and behaviors more comprehensively:

**Social Media & Search Data**: Gathering and analyzing data from users' travel-related social media posts or search queries can reveal implicit preferences

**Group Size & Age Demographics**: Users traveling solo versus in a group or with family would likely have different preferences for accommodations and activities. Age demographics could also influence preferences for more adventurous or relaxing itineraries.

**Improving ML Model**

1. Hyper Tuning – Using different methods to optimize number of clusters based on real world data.
2. Hybrid Model – Using Content based and collaborative filtering.
3. Feed back loop – Allow user to rate recommendations.’’

**Success Metrics**

1. Session Duration – Track how long a user stayed engaged.
2. CTR – Check how many times user interacted with recommended Itenary.

**Satisfaction Metrics**

1. Rating/Feedback – Let user Rate itenary.
2. Retention Rate – Check how many time user return to the platform after first interaction.