

Founding AI Engineer Assignment

Personalised Itinerary Generation

Objective:

We are building a travel companion app that provides personalised travel itineraries based on a user's preferences, behaviours, and travel history. The goal of this assignment is to assess your ability to design an intelligent system that generates personalised travel itineraries using user data (expenses, vibe, reviews, Instagram photos, travel mates, etc.) and to evaluate your proficiency in AI, backend engineering (Node.js), and AWS architecture (RDS, Beanstalk, etc.).

The key goal of this task is to demonstrate how you'd build a scalable system capable of generating AI-driven itineraries for different users and destinations, with a focus on destination "X" and some complexity around how the system selects "X".

Problem Statement:

You are given a dataset of users with the following information:

- **Expenses:** Their past travel expenses (categories, budgets, etc.).
- **Vibe:** Personal preferences like adventure, relaxation, socializing, etc.
- **Reviews:** Reviews of destinations they have visited.
- **Instagram Photos:** Popular photos from their social media.
- **Travel Mates' Vibe:** The general vibe and preferences of the people they usually travel with.

Task:

Build a system that will:

1. **Select a destination (X)** for a given user based on this data.
2. **Generate a personalised itinerary for this destination (X)** that takes into account:
 - User's vibe and preferences.
 - User's past travel history and expenses.
 - Social context: How the vibe of their usual travel companions influences the trip.
 - Reviews and popular spots from Instagram data related to that user.

Additional Complexity (optional but encouraged):

- **Contextual Decision for X:** Introduce a secondary layer of complexity in how the destination is chosen. For example, evaluate trends, social media influence, or seasonality to help determine the best destination.
-

Assignment Guidelines:

1. AI Component:

- Design and implement a model or algorithm that suggests a travel itinerary based on the data points provided. The approach should incorporate user preferences (vibe, reviews, etc.) and adapt the itinerary accordingly.
- Consider how the vibe of the user's travel mates might influence their decision-making and modify the itinerary.
- You are encouraged to implement a simple recommendation algorithm or a more complex AI/ML solution.

2. Backend Component:

- Implement a Node.js API that will serve the generated itinerary.
- The API should take as input: user information (and potentially travel mate data) and return the final itinerary with travel recommendations.
- Integrate with AWS (RDS, Beanstalk) to store user information and retrieve past trip data.

3. Cloud Infrastructure:

- Use AWS services (RDS for database storage, Beanstalk for deployment) to ensure the system can scale as the number of users grows.
 - The system should be designed in a way that could eventually serve millions of users.
-

Instructions:

1. **Dataset:** You can mock or create your own dataset for this task. Assume you have data related to user profiles, past trips, expenses, travel mates, and reviews (text or images).
 - You are provided with a CSV file that contains historical data of users, including the following fields:
 - i. **user_id:** Unique identifier for each user.
 - ii. **expenses:** Average spending per day during trips.
 - iii. **vibe:** The general travel vibe preferred by the user (e.g., Adventure, Relaxation, Luxury, Budget, etc.).
 - iv. **reviews:** Textual reviews left by the user for past trips.
 - v. **instagram_photos:** Links to a few Instagram photos shared by the user (you can assume that you have metadata from these photos like locations, categories, and themes).

- vi. **previous_destinations**: List of previous destinations visited by the user.
 - vii. **favorite_activities**: List of preferred activities (e.g., hiking, shopping, fine dining).
 - You can simulate or generate additional data if required
- 2. **Technology**: You must use Node.js and AWS (preferably RDS, Beanstalk) in your solution. For the AI model, any open-source libraries or custom algorithms can be used.
- 3. **Tasks**:
 - **Data Preprocessing**:
 - i. Clean and preprocess the data. Handle missing or inconsistent entries. You may perform tasks like text processing for reviews or Instagram metadata analysis.
 - **User Segmentation**:
 - i. Use unsupervised learning (e.g., clustering) or any other relevant approach to segment users based on their preferences, past behaviors, and reviews.
 - **Recommendation Engine**:
 - Build a recommendation engine that generates a personalized itinerary for a new user for Destination = "X". This could include places to visit, restaurants, activities, etc.
 - The recommendations should align with the user's preferences based on their cluster/segment and past behavior.
 - **Itinerary Generation**:
 - i. Design a system that produces a full itinerary for a trip to Destination = "X." This should include day-wise suggestions, along with approximate costs for each day based on the user's spending habits.
 - **Evaluation**:
 - i. Provide an approach on how you would evaluate the performance of your recommendation engine. Think about metrics such as user satisfaction, relevance, or novelty.
- 4. **Documentation**: Provide clear documentation that explains:
 - How the destination is selected (X) for a user.
 - How the itinerary is generated and personalised.
 - What AI/ML techniques or algorithms you used.
 - How the backend API is designed and how it integrates with AWS services.
- 5. **Testing**: Ensure that the solution is tested and includes a test case for different users with varying preferences.
- 6. **Timeframe**: The assignment is designed to be completed in 5 days. **Deadline 10th October.**

Deliverables:

1. **Code Repository:**
 - A GitHub repository with your complete solution, including backend code, AI model/algorithm, and AWS configuration (or mock setup if you don't have an AWS account).
 - A clear **README.md** detailing how to set up, run, and test the solution.
2. **Technical Documentation:**
 - A technical brief describing the AI approach used, the backend API design, and how you structured the AWS components for scalability.
 - Explain any assumptions you made or constraints you encountered.
3. **Optional but Encouraged:**
 - Discuss potential optimizations for the AI model and how you'd enhance it with more real-world data or additional complexity.
 - Provide any metrics you think would be valuable in measuring the success of the generated itineraries (e.g., satisfaction, engagement).

Evaluation Criteria:

- **AI/ML Approach:** How effectively is the user's data leveraged to personalize the travel itinerary?
- **System Design:** Is the solution designed to scale and integrate effectively with AWS?
- **Code Quality:** Clean, modular code with clear documentation.
- **Long-term Thinking:** Demonstration of how this could evolve into a long-term product feature.
- **Problem-Solving Skills:** How well you navigate through the complexity of data and personalization.

We are looking for individuals with a strong problem-solving mindset and a willingness to build robust, scalable systems that go beyond quick hacks. This assignment is designed to give you a taste of the work you'd be doing as a founding AI engineer at Saathi.

Good luck!