**Travel Destination Planner - Documentation**

1. **Problem Statement:**

Planning a trip can lead to confusion. Travelers often feel it overwhelming to choose the right destination or the destination that fulfils majority of their checklist i.e. budget, interests, preferred group type etc.

**Eg. Visiting Thailand with family may not be the good idea**

Currently, majority of people try to search for destinations on Internet platforms like

* Instagram reels
* Youtube vlogs
* Blogs
* Asking for suggestions from Relatives

The goal of the project is to build a simple AI-based travel planner that recommends destinations that suggests them the perfect or the best suitable destination based on their weightage to parameters like more weightage to budget or more weightage to location type.

This approach allows users to give them not the perfect but the most suitable destination based on some questions asked.

1. **Scope**

The simple AI-based Travel Destination Planner focuses on helping users find suitable travel destinations by asking users simple questions. It uses simple rule based logic and scoring to suggest destination based on parameters such as:

* Budget
* Preferred location type
* Type of group
* Month of travel
* Foreign travel
* Visa availability

Features:

* Interactive Q&A to capture user preferences
* Weighted scoring system based on what matters most to the user
* Rule-based recommendations (if-else based logic)
* Highlight destinations based on travel month (e.g., “Best places to visit in August”)

1. **Brief Overview of Existing Work:**

AI-powered Itinerary Tools:

1. Wonderplan.ai

Wonderplan.ai is an AI-powered itinerary generator that asks for user inputs such as travel dates, destination, preferences like (culture, budget). It uses GPT-based models to generate daily plans, activities and routes.

1. Plantrip.io

Plantrip.io is an AI-driven travel tool that auto-generates full itineraries based on user input like destination, preferences, travel, duration and budget. Custom itinerary generation using AI algorithms trained on a large travel database.

However, this websites are the one that plans itineraries and not tailored for use case where the user doesn’t want to plan itinerary and want to get suggestions for destination. Also these websites are not tailored for user priority to parameters and requires initial destination input.

**4. Theoretical Concepts Used for Solving Problem**

**4.1 Rule-Based Reasoning**

The system uses **explicit rules** to match user preferences (like group type, visa requirement, budget, or month) with destination features. For example, if a user prefers a family trip, the system filters destinations supporting "family" in the group\_type field.

**4.2 Weighted Scoring Model**

For the "Smart Suggestion" path, preferences are assigned weights by the user. Each destination is evaluated against these preferences, and a **score is calculated**. The destination with the highest total weighted score is recommended. This approach combines:

* **Multi-Criteria Decision Making (MCDM)** principles
* **Simple additive weighting** method

**4.3 Filtering Algorithms**

The system uses simple yet effective **attribute-based filtering** techniques in various paths like "Suggestions by Budget" and "Suggestions by Month". Based on the user’s input, it filters destinations that satisfy the relevant criteria:

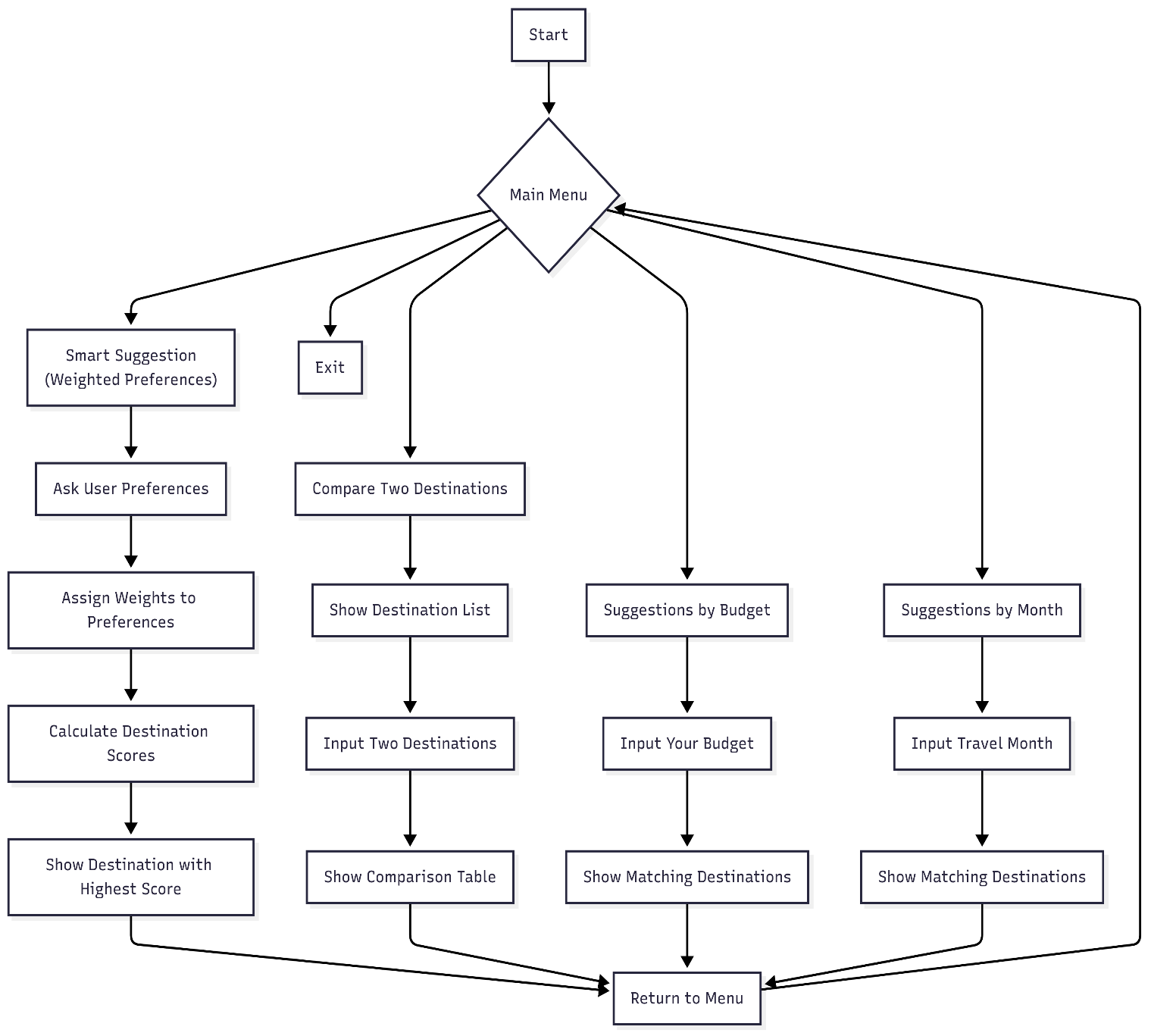
* **Budget Filtering**: Only destinations whose cost\_per\_person range overlaps with the user’s specified budget are selected.
* **Month-Based Filtering**: Destinations are recommended only if the selected travel month appears in their ideal\_months list.
* **Destination Type Filtering**: If the user explicitly prefers only **local** or **foreign** trips, destinations not matching this preference are filtered out.
* **Visa Requirement Filtering**: If a destination requires a visa (visa\_required = True) and the user indicates **they do not have a visa**, such destinations are excluded from suggestions.

This ensures that the system only presents **realistic and personalized** travel options that the user can actually consider.

**4.4 Pairwise Comparison**

When comparing two destinations, the system creates a **comparison table** displaying differences across all fields such as cost, purpose, visa requirement, ideal months, etc. This supports better decision-making using **comparative analysis**.

1. **Flowchart**



1. **Dataset Description**

Your project uses a **manually curated, rule-based dataset** of destinations. Each destination is stored as a Python dictionary with various attributes.

{

"name": "Bali",

"type": "foreign",

"cost\_per\_person": [35000, 50000],

"group\_type": ["solo", "family", "friends"],

"visa\_required": false,

"purpose": ["Beach", "Relaxation", "Luxury"],

"ideal\_months": ["April", "May", "June", "September"],

"not\_ideal\_months": ["January", "February", "March", "October", "November", "December"],

"month\_reasons": {

"April": "Start of dry season, great weather",

"May": "Pleasant and sunny",

"June": "Dry and comfortable",

"September": "End of dry season, ideal for travel",

"January": "Wet season with heavy rains",

"February": "Rainy and humid",

"March": "Wet season continues",

"October": "Start of rainy season",

"November": "Rain increases",

"December": "Peak rainy season"

}

}

**name**: The name of the destination, such as *Bali* or *Manali*.

**type**: Indicates whether the destination is "domestic" (within the country) or "foreign" (international).

**cost\_per\_person**: A list of two integers representing the approximate budget range for one traveler. For example, [35000, 50000] means the trip will cost between ₹35,000 and ₹50,000 per person.

**group\_type**: A list of ideal travel groups for this destination. Common values include "solo", "family", and "friends". This helps suggest destinations based on whether the user is traveling alone or with others.

**visa\_required**: A boolean field indicating whether a visa is needed to visit this destination.

**purpose**: A list of travel purposes or themes that the destination is suitable for, such as "Relaxation", "Luxury", or "Beach".**ideal\_months**: A list of months that are best suited for visiting this destination, based on weather, events, and other factors.

**not\_ideal\_months**: A list of months when visiting the destination is not recommended, usually due to bad weather, off-season, or high costs.

**month\_reasons**: A dictionary that provides reasons why a month is ideal or not ideal. This supports explainable recommendations by justifying each suggestion to the user.

1. **Time and Space Complexity:**

**Smart Suggestion Algorithm**

* **Time Complexity:** O(N)  
  Where N is the number of destinations. Each destination is evaluated against a fixed number of user preferences (group type, visa requirement, budget, purpose, month). These are constant-time condition checks, so the complexity remains linear.
* **Space Complexity:** O(N)  
  A score is calculated and stored for each destination before sorting or selecting the best one.

**Budget and Month-Based Filtering**

* **Time Complexity:** O(N)  
  Each destination is checked once to see if it matches the user's given budget range or travel month.
* **Space Complexity:** O(K)  
  Where K is the number of destinations that matched the filter and are returned.

**Destination Comparison**

* **Time Complexity:** O(1)  
  Only two destinations are selected and displayed from the existing dataset without iteration.
* **Space Complexity:** O(1)  
  No additional space used beyond those two destination objects.

1. **Test Cases**

|  |
| --- |
| Test Case ID : TC\_01 |
| Feature : Plan a Destination (by priority) |
| Input : |
| - User chooses: Option 1 (Plan a Destination) |
| - Ranks: Budget = 1, Group Type = 2, Purpose = 3 |
| - User Data: Budget = ₹40,000, Group Type = Solo, Purpose = Beach, Month = May, Visa Required = No |
| Expected Output : 1 destination suggestion (e.g., Bali) |
| Actual Output : 1 destination suggested - Bali |
| Result : ✅ Pass |

|  |
| --- |
| Test Case ID : TC\_02 |
| Feature : Plan a Destination (by priority) |
| Input : |
| - User chooses: Option 1 (Plan a Destination) |
| - Ranks: Group Type = 1, Purpose = 2, Budget = 3 |
| - User Data: Budget = ₹18,000, Group Type = Family, Purpose = Nature, Month = December, Visa Required = No |
| Expected Output : 1 destination suggested (e.g., Andaman Islands) |
| Actual Output : 1 destination suggestion - Andaman Islands |
| Result : ✅ Pass |

|  |
| --- |
| Test Case ID : TC\_03 |
| Feature : Plan a Destination (by priority) |
| Input : |
| - User chooses: Option 1 (Plan a Destination) |
| - Ranks: Purpose = 1, Budget = 2, Group Type = 3 |
| - User Data: Budget = ₹5,000, Group Type = Solo, Purpose = Adventure, Month = July, Visa Required = Yes |
|  |
| Expected Output : No destination suggested (due to critical mismatches in top priorities) |
|  |
| Actual Output : 1 destination suggested (Bali) |
| - Suitable for solo travelers |
| - No visa required |
| - ⚠️ Budget issue: Requires approx ₹30,000 extra |
| - ⚠️ Purpose mismatch: Not ideal for adventure |
|  |
| Result : Pass |

|  |
| --- |
| **Test Case ID:** TC\_04 |
| **Feature:** Suggest Destinations by Month |
| **Input:**  User selects: Option 2  Month: December |
| **Expected Output:**  List of ideal December destinations (e.g., Goa, Dubai, Jaipur) |
| **Actual Output:**  Goa, Dubai, Jaipur, Thailand, Andaman Islands |
| **Result:** ✅ Pass |
| **Remarks:** Correct destinations shown based on season; excluded unsuitable places like Ladakh and Manali due to extreme weather. |

|  |
| --- |
| Test Case ID: TC\_05 |
| Feature Name: Compare Two Destinations |
| Input: |
| First Destination: Goa |
| Second Destination: Manali |
|  |
| Expected Output: |
| 📊 Comparison: Goa vs Manali |
| Actual Output: |
| 📊 Comparison: Goa vs Manali |
| 💰 Cost per Person: ₹[9000, 16000] vs ₹[7000, 13000] |
| 🌤 Best Months: November, December, January, February vs March, April, May, September, October |
|  |
| Result: ✅ Pass |
|  |
| Remarks: Output matched all expected values correctly. |
| Visual formatting of the comparison is clear and easy to understand. |

|  |
| --- |
| Test Case ID: TC\_06 |
|  |
| Feature: Destination recommendation based on user's budget input |
| Input: ₹8000 |
| Expected Output: Suggested: Destinations that have travel cost of 8000 or less than that (per person) |
| Actual Output: Suggested: Manali, Jaipur, Sikkim | Not Suggested: Goa, Ladakh, Dubai |
| Pass/Fail: ✅ Pass |
| Remarks: Destinations with cost ≤ ₹8000 shown correctly. App returned to main menu. |