Lab 2 – Wise Traveler Product Specification

Sean Baker

Old Dominion University

CS 411W

Professor Sanobar

Version 2

March 28, 2025

Table of Contents

Table Of Figures

List of Tables

1. Introduction

This Software Requirements Specification (SRS) document officially declares the *Wise Traveler* web application's functional and non-functional requirements. It serves as a reference for system architects, testers, developers, and project managers to create a precise understanding of system behavior, constraints, and functionalities.

This SRS provides a structured approach to the system's design, implementation, and verification, specifying the key components that need to be employed to develop and deploy the application successfully. It specifies the technical scope, including the features implemented in the initial prototype and how they translate to the final product. In addition, it specifies performance, security, and usability requirements to ensure the system meets technical and user requirements.

1.1 Purpose

Wise Traveler is a web application designed to simplify the complexities of trip planning while encouraging exploration beyond mainstream tourist destinations. Many travelers face challenges when preparing for trips, including navigating fragmented travel resources, ensuring personal safety, and gaining meaningful cultural insights. Wise Traveler consolidates essential travel information into a single, AI-powered platform that provides real-time safety alerts, customized recommendations, and cultural guidance. This approach allows users to confidently explore new destinations, making their travel experiences safer, more enriching, and more immersive.

Unlike conventional travel apps that focus primarily on booking accommodations and transportation, Wise Traveler aims to provide a deeper level of engagement by helping users make informed decisions about their destinations., Wise Traveler empowers users to travel more responsibly and enjoy experiences beyond typical tourist attractions.

For travelers, Wise Traveler offers several key features. The Destination Search allows users to find information about a location by entering the city name or enabling location services. The Travel Advisory Feed provides real-time updates on health risks, security concerns, and weather conditions. Travelers can use the Cultural Guide to learn about local customs, etiquette, and laws to ensure respectful interactions. The Itinerary Planner enables users to create personalized travel schedules based on recommendations and must-see attractions.

A standout feature of Wise Traveler is its Interactive Chat, an AI-powered assistant that provides real-time responses to traveler inquiries. Users can ask about local attractions, safety updates, or cultural customs and receive instant guidance. The chat system also connects travelers with local businesses, tourism offices, and transport providers, allowing seamless communication and personalized recommendations.

1.2 Scope

The Wise Traveler software product is an intelligent travel planning application designed to enhance travel experiences by providing reliable, personalized information tailored to each traveler's needs. This comprehensive travel companion application assists users in planning, organizing, and optimizing their travel experiences.

Wise Traveler enables users to generate customized travel itineraries based on their preferences, destinations, and constraints. The system processes natural language queries about travel destinations and creates detailed itineraries, including recommended attractions, activities, and logical visit sequences. The application incorporates route optimization functionality, allowing travelers to efficiently navigate between multiple destinations by analyzing geographical proximity and creating logical travel sequences. Users can visualize these optimized routes through integrated mapping services, which display destinations, routes, and points of interest. Wise Traveler includes user profile management capabilities, enabling account creation, authentication, and profile customization for a personalized experience.

The application scope explicitly excludes several features and capabilities. Wise Traveler does not provide real-time flight tracking, booking, or reservation services for accommodations, transportation, or activities. The system does not process payments or conduct financial transactions. While the application provides travel information, it does not offer comprehensive travel insurance advice or legal consultation regarding visa requirements and travel regulations. The current implementation does not include offline functionality, requiring an internet connection for all features. Additionally, the application does not incorporate real-time communication with travel agents or local guides.

1.3 Definitions, Acronyms, and Abbreviations

AI-Assisted Trips	Travel plans are enhanced by AI models that suggest optimal routes and recommendations based on real-time data.
Amazon Aurora	A cloud-based relational database service used for scalable and secure data management.
API (Application Programming Interface)	A set of protocols that allow software applications to communicate and exchange data.
Authentication Module	A system that verifies user identities and ensures secure access to the platform.
Backend	The application's server-side handles data processing, API requests, and database interactions.
Cloud-Based Infrastructure	Remote hosting services like AWS provide scalable storage and computing resources.
Continuous Deployment (CD)	A software development practice where updates are automatically tested and deployed to production.
Continuous Integration (CI)	A practice where code changes are frequently tested and merged to maintain software stability.
CORS (Cross-Origin Resource Sharing)	A security feature that allows web applications to make requests to different domains.
Custom Trips	User-defined travel routes were created with Google Places API, and AI-generated recommendations were generated.
Data Integrity	Ensuring stored data remains accurate, consistent, and secure from corruption or unauthorized changes.
Database	A structured data collection stores user profiles, travel preferences, and historical data.
Dynamic Mapping	Real-time visualization of locations, routes, and travel points using interactive maps.
Frontend	The user-facing part of the Wise Traveler application is where interactions take place.
Google Maps API	A service used to provide location-based information, route planning, and real-time traffic updates.
Google Places API	An API that provides detailed information about places, including reviews, photos, and contact details.
Health Risks Analysis	AI-assisted assessment of potential health hazards in travel destinations.
IDE (Integrated Development Environment)	A software application that provides tools for coding, debugging, and testing.
Itinerary Generation	The AI-powered process of creating personalized travel plans based on user preferences.

Latency	There is a delay in processing and transmitting data across the network.
Load Testing	A testing method that evaluates system performance under high user traffic conditions.
Middleware	Software that facilitates communication between the front end and back end of an application.
MFCD (Model-Function-Class Diagram)	A visual representation of the interaction between different components of Wise Traveler.
Node.js	A JavaScript runtime that allows backend development to handle API requests and server logic.
OpenAI API	A machine-learning-based service used for generating travel insights, recommendations, and itinerary planning.
Periodic Traffic Congestion	Traffic patterns that occur at regular intervals, such as rush hours.
Pre-Travel Analysis	Analyzing traffic, weather, and safety data before a trip begins.
Prototype	A scaled-down version of the final application used to test core functionalities.
Real-Time Alerts	Instant notifications that inform users about changes in travel conditions, safety concerns, or local events.
Recommendation System	AI-powered suggestions for destinations, attractions, and travel routes based on user preferences.
REST API (Representational State Transfer API)	A set of web services that allow communication between different systems.
Return on Investment (ROI)	The financial gains from the Wise Traveler software compared to its development and operational costs.
Scalability	The system can handle increased user traffic and data load without performance issues.
Security Measures	Protocols like authentication, encryption, and data validation to protect user information.
Simulation Console	A tool used for testing system functionalities under simulated real-world conditions.
Software Deployment	The process of launching and updating Wise Traveler's features and services.
SQL (Structured Query Language)	A programming language used for managing relational databases like MySQL.
Travel Data Collection	Gathering user travel behavior, preferences, and real-time location data.
Traffic Avoidance	AI-generated alternative route suggestions to minimize congestion and delays.
Trip Planning Module	The component of Wise Traveler enables users to create, modify, and optimize travel itineraries.

1.4 References

PennState Global. (n.d.). *Travel & Safety Tips to Know Before You Go.* Retrieved from Penn State Global: https://global.psu.edu/article/travel-safety-tips-know-you-go

Centers for Disease Control and Prevention, (2024, November 30). *Survival Guide to Safe and Healthy Travel*. Retrieved from Centers for Disease Control and Prevention, www.cdc.gov/travel/page/survival-guide.

Davies, W. (2023. , June 14). *10 Useful International Travel Tips for First-Time Travelers.* Retrieved from Booking.Com 2023 Sustainable Travel Report. : https://www.gooverseas.com/blog/best-international-travel-tipsfor-first-timetraveler

Documentation, A. C. (n.d.).

Facebook. (2023). (n.d.). *Developer Guide.* Retrieved from React.js: https://reactjs.org/docs/gettingstarted.html

García, R. &. (2022). Automating Travel Planning Using AI Algorithms: A Case Study. *Journal of Computational Tourism*, 10(2), 23–38. .

John, H. (n.d.). *Best Practices for Traveler Safety.".* Retrieved from Travel Gov: https://travel.state.gov/content/travel/en/international-travel/before-you-go/aboutour-newproducts/Best-Practices-for-Traveler-Safety.html.

Johnson, J. (2022,, June, 9). *Security Risks of Traveling Internationally: The Five Greatest Safety Concerns of Going Abroad.* Retrieved from Jensen Hughes, www.jensenhughes.com/insights/the-five-greatest-risks-to-your-security-whentravellinginternationally.

Kimberli, K. (2024, September 27). *The Most Helpful 27 Tips for First-Time Travelers*. Retrieved from Worldpackers, https://www.worldpackers.com/articles/first-time-travelers.

Lisa, A. (2020, January 14). *Strange Laws to Be Aware of in the Most Popular Countries for Tourists*. Retrieved from Stacker: https://www.stacker.com/travel/strange-laws-be-awaremost-popular-countries-tourists.

Ng, A. (2022). *Machine Learning Yearning.* Retrieved from deeplearning.ai

Node.js Foundation. . (2023). *OpenAI. (2023). OpenAI.* Retrieved from Node.js Documentation: https://nodejs.org/en/docs

Office of Financial Management, (2024, October). *"Travel Alerts."*. Retrieved from Office of Financial Management, https://www.commerce.gov/ofm/offices/office-financial-reporting-policy-internalcontrols-andtravel/travel-management/travel-alerts.

Smith, P. &. (2023). *Cultural Immersion in the 21st Century: A Guide for Global Explorers.* World Travel Publishing.

Statista Research Department. (2024, June 11). *Number of United States Residents Traveling Overseas from 2002 to 2023: Overseas Tourism US 2023*. Retrieved from Statista: https://www.statista.com/statistics/214774/number-of-outbound-tourists-from-the-us/.

Swanson, M. (2023, July 27). *This Is How Much Time Travelers Spend Researching Before Booking a Trip.* Retrieved from TravelWires: https://www.travelwires.com/this-is-how-muchtime-travelers-spend-researchingbefore-booking-a-trip

Turner, M. (2019, September 17). *Stats: 53% of US Adults Say Cultural Immersion is Essential to Travel.* Retrieved from Travel Agent Central, www.travelagentcentral.com/yourbusiness/stats-53-u-s-adults-say-cultural immersion-essentialtotravel

UN Tourism, . (n.d.). *UN Tourism: Bringing the World Closer.* Retrieved from UN Tourism World Tourism Barometer, https://www.unwto.org/un-tourism-world-tourism-barometer-data.

UNWTO. (n.d.). *World Tourism Barometer*. Retrieved from https://www.unwto.org

Woolf, M. (2024, October 15). Falling for the Hype? How to Evade Tourist Traps [New Study]. pp. 23–26. Retrieved from PhotoAiD.

World Health Organization. (2024, 6 12). *World Health Organization*. Retrieved from https://www.who.int/

World Health Organization. . (2024, 6 12). *World Health Organization*. Retrieved from https://www.who.int/

1.5 Overview

This product specification provides the Wise Traveler prototype's hardware and software configuration, interfaces, features, and capabilities. The following sections will describe each feature and its requirements for the prototype's implementation.

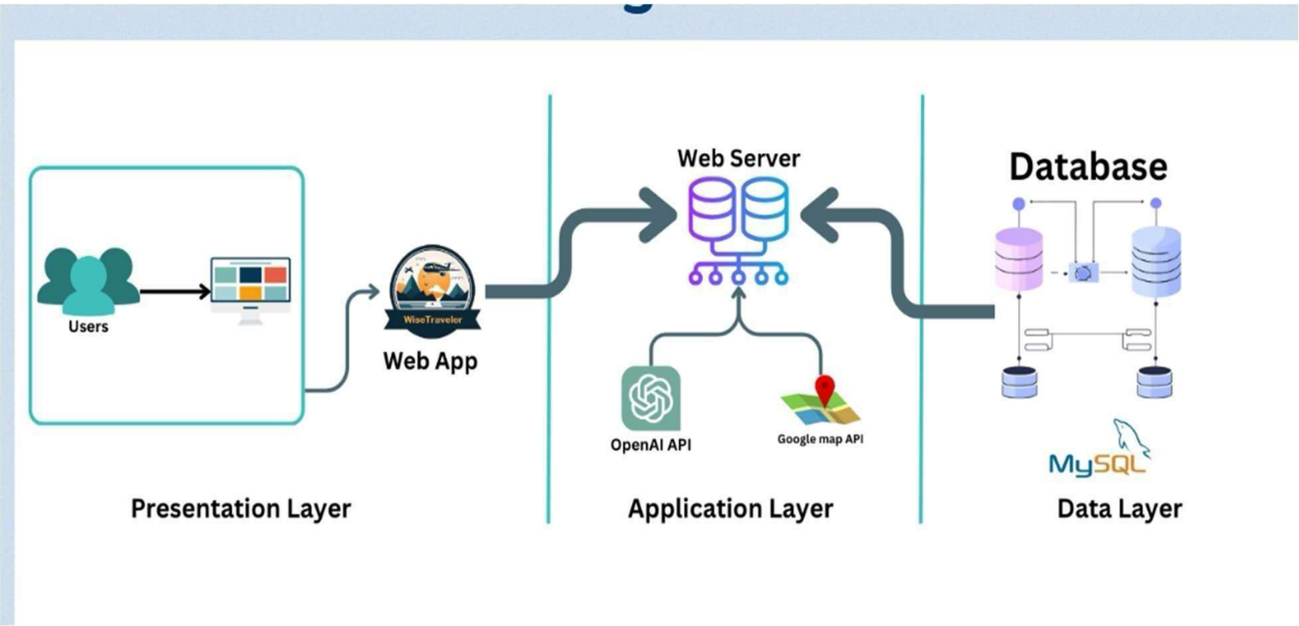
2. Overall Description

The Wise Traveler prototype will be developed as a web application with a database hosted on Supabase and deployed on Vercel. It will feature two distinct interfaces: one optimized for smartphone screens to accommodate travelers seeking real-time travel insights and another for tablet use, allowing business owners and service providers to manage their listings and engage with tourists. While the prototype will include most of the core functionalities expected in the full version, some features will be partially implemented or simulated, ensuring a streamlined proof of concept for testing and validation before full deployment.

2.1 Product Perspective

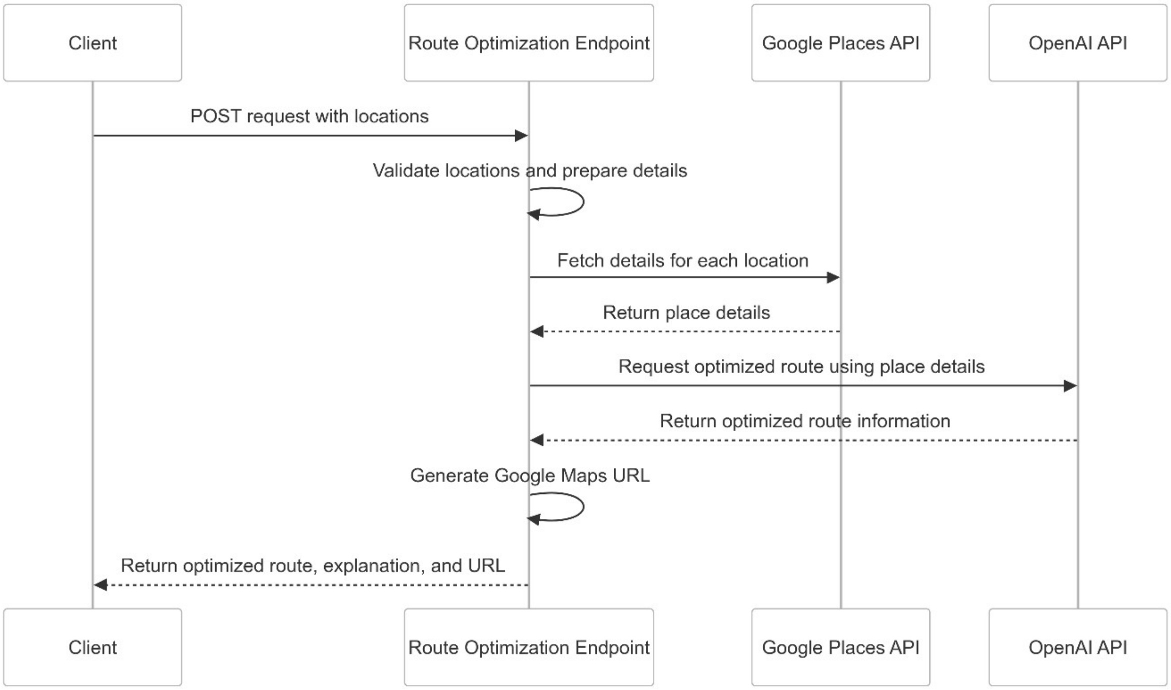
The Wise Traveler prototype is a simplified version designed to test core functionalities before deployment. The system is designed to simulate real-world travel scenarios, ensuring that AI-driven recommendations, travel insights, and user interactions function as expected. By integrating Next.js for the frontend, Node.js for the backend, Supabase (PostgreSQL) as the database, and AI-powered APIs such as OpenAI and Google Places, the prototype offers a seamless and intelligent travel experience.

Figure 1Wise Traveler Prototype Major Functional Component Diagram



The Wise Traveler system integrates AI-driven route optimization to provide travelers with efficient and personalized itineraries. Users submit a list of destinations, which the Route Optimization Endpoint validates before retrieving essential details from the Google Places API, such as coordinates, operating hours, and reviews. The system then leverages the OpenAI API to analyze and optimize the route based on travel time, user preferences, real-time traffic, safety considerations, and interests. Once the optimized route is generated, the system creates a Google Maps URL, allowing seamless navigation. This AI-powered process ensures smarter, safer, and more efficient travel, offering users a customized itinerary that saves time and enhances their overall experience. By integrating real-time data, intelligent recommendations, and seamless mobile-friendly navigation, Wise Traveler helps users explore destinations effortlessly.

Figure 2 API Route Details


Client
Route Optimization Endpoint
Google Places API
OpenAI API
POST request with locations
Validate locations and prepare details
Fetch details for each location
Return place details
Request optimized route using place details
Return optimized route information
Generate Google Maps URL
Return optimized route, explanation, and URL
Client
Route Optimization Endpoint
Google Places API
OpenAI API

The frontend, built using Next.js and hosted on Vercel, provides an intuitive and responsive interface that allows travelers to input their preferences, receive AI-generated travel recommendations, and access location-based insights. Designed for mobile-first usability, the interface ensures users can easily navigate through destination searches, travel advisories, itinerary planning, and real-time safety updates. Prototype Functional Description

2.2 Product Functions

The Wise Traveler prototype will utilize hardware, software, and user interfaces to support its cross-platform functionality, ensuring a seamless experience for travelers and business owners.

Table 1 Table 1 Wise Traveler Feature Description and Prototype Implementation

Feature	Prototype	Real World Product
User Interface	Basic, Limited Functionality	Full Design and Function
Data	Static Data from Sources	Live Data from API
Calendar	Limited, Manual Entry	Robust and Integrated
Map	Basic Location Points	Location and Travel Time
AI Assistance	Fully Functional	Fully Functional
Safety Concerns	Partially Functional	Fully Functional
Health Risks	Partially Functional	Fully Functional

The features for travelers that will be partially implemented in the Wise Traveler prototype include AI-powered recommendations, Interactive Chat, and Itinerary Planner. The AI-powered recommendations will generate personalized travel suggestions based on user preferences, but real-time machine-learning adaptation will be limited. The Interactive Chat will be fully functional for answering general travel inquiries and providing cultural insights; however, direct integration with local businesses and tourism offices will be simulated. The

Itinerary Planner will allow users to create and manage their travel plans, but automated syncing with real-time transportation schedules will not be included.

Additionally, the Email Notification System will be implemented with travel advisory alerts and itinerary reminders, but real-time transportation updates, such as bus departure alerts, will use simulated data. The Google Maps integration will include location-based search and navigation but will not support live business updates. These limitations ensure the prototype effectively demonstrates core functionalities while allowing further development in the full product release.

2.2.1 Hardware Interfaces

The application will be compatible with operating systems, including Mac, Windows, and Linux, for desktop users and iOS and Android for mobile users. While optimized for smartphones (for travelers), it will also be accessible via standard web browsers.

2.2.2 Software Interfaces

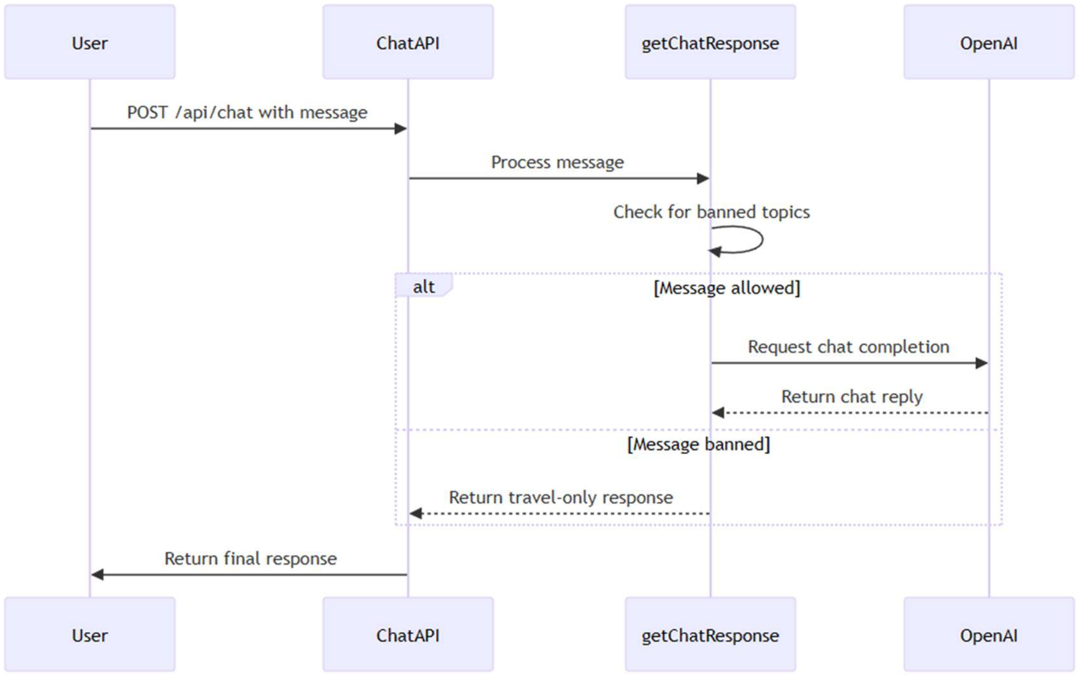
Wise Traveler is hosted on Vercel, which provides a scalable, serverless environment optimized for Next.js applications. The application architecture follows a hybrid static and server rendering model, ensuring high performance and responsiveness.

The backend uses Next.js (version 15) and organizes API routes into two directories. The **`/app/API/`** directory handles new API features and services, while the **`/pages/API/`** directory manages user authentication and legacy API functionality.

User data, including profiles, travel history, and preferences, is managed through Supabase, a PostgreSQL-based backend-as-a-service. Authentication is handled via Supabase Auth, ensuring secure access control and efficient data management.

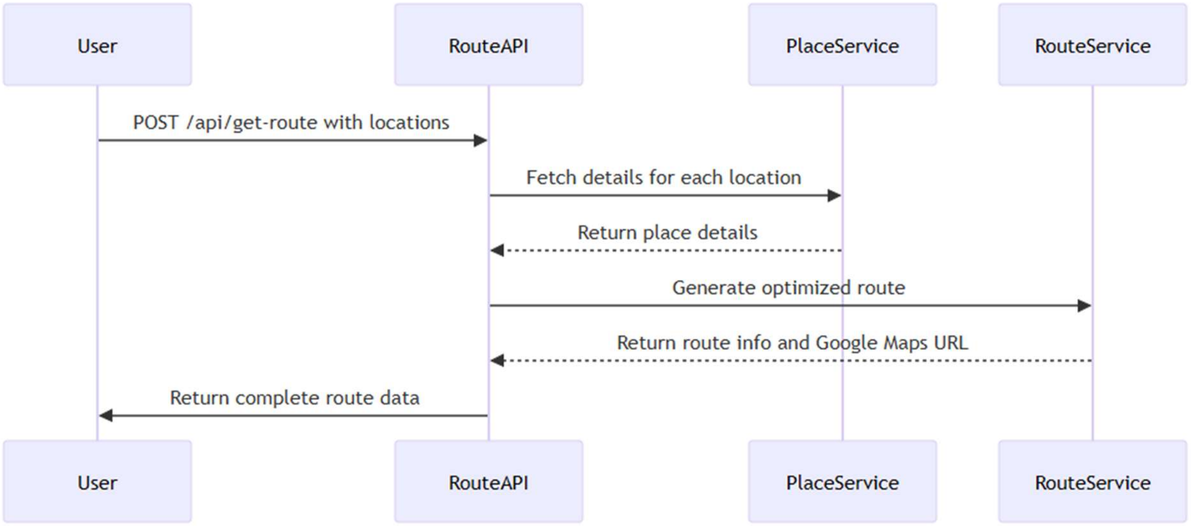
The application integrates OpenAI API (version 4) to provide AI-driven travel recommendations based on user preferences. AI-powered functionality is implemented through multiple API routes, including `/API/chat` for AI chat responses, `/API/generate-itinerary` for personalized itinerary generation and `/API/get-route` for optimized travel routes.

Figure 3 Chat API Route



For mapping and geolocation services, *Wise Traveler* relies on Google Maps API to provide location-based data and interactive maps. The Google Maps library facilitates API requests for geolocation and route optimization, while the React Google Maps library enables interactive map rendering.

Figure 4 Map API Route


User
RouteAPI
PlaceService
RouteService
POST /api/get-route with locations
Fetch details for each location
Return place details
Generate optimized route
Return route info and Google Maps URL
Return complete route data
User
RouteAPI
PlaceService
RouteService

2.2.3 User Interfaces

The *Wise Traveler* web application is accessible on desktop and mobile devices. It features an interface developed using React (version 19), which allows for dynamic rendering and optimizes performance. The styling is done with styled-components, ensuring a modular and maintainable user interface.

The traveler dashboard enables users to manage their profiles, itineraries, and travel history. An AI chat assistant provides real-time travel insights, while the itinerary planner allows users to create and modify their travel plans. The route optimization feature also helps users find the best travel routes based on their preferences. The interactive search functionality allows users to search for destinations and receive AI-driven recommendations.

2.2.4 Communications Protocols and Interfaces

The *Wise Traveler* prototype will rely on HTTP and HTTPS protocols to ensure secure communication between the frontend, backend, and external APIs. All API interactions follow RESTful principles with JSON as the primary data exchange format. For real-time interactions, the prototype will implement WebSockets to enable instant updates for AI-powered chat responses, Live travel advisories, and safety alerts. To maintain secure authentication and session management, the system will use JWT (JSON Web Tokens) for user authentication, ensuring encrypted and tamper-proof login sessions. While SMTP will be integrated for email notifications (such as itinerary changes, travel alerts, and promotions), this feature is currently planned for future iterations. It will not be implemented in the initial prototype. Error handling will include retry mechanisms for failed API calls and fallback responses in case of service disruptions from third-party APIs like OpenAI and Google Maps.

2.3 Constraints

- **API Rate Limits** – The system is restricted by the request limits imposed by OpenAI's API for AI-generated recommendations and Google Maps API for location services.
- **Security Compliance** – All user authentication must be handled using JWT (JSON Web Tokens), and sensitive data must be transmitted securely via TLS 1.3 encryption.
- **Platform & Hosting**—The application must be deployed on Vercel's cloud infrastructure; self-hosted deployments will not be supported.
- **Internet Dependency** – The system requires constant internet access; offline access to travel advisories, itineraries, and AI-generated recommendations will not be available in the prototype.

2.4 Assumptions and Dependencies

2.4.1 Assumptions

- The following assumptions are made for the development and operation of the Wise Traveler prototype:
- Users will have a stable internet connection while using the application, as offline functionality is not supported.
- Mobile users will have GPS/location services enabled to receive AI-driven recommendations and travel advisories.
- The system's APIs will be available and functional, including OpenAI API for AI chat and Google Maps API for navigation.

2.4.2 Dependencies

- **Vercel Hosting** – The application is deployed on Vercel, and downtime on this platform will affect system availability.
- **Supabase Database** – User authentication and data storage rely on Supabase (PostgreSQL).
- **Third-Party APIs** – The platform requires OpenAI API for AI-powered travel suggestions and Google Maps API for location-based services.
- **JWT Authentication**—JWT (JSON Web Tokens) is used for authentication, secure login, and user session management.