Trek

Requirements

Last Update 2024-07-01

Version 1.0

Table of Contents

Document Information

Revision History

Date	Version	Status	Prepared by	Comments
2024-05-25	1.0	Approved	Trekkers	
2024-07-01	2.0	Approved	Trekkers	

Approval

Role	Name	Signature / Initial	Date
Developer	Jacob Zhu	JZ	2024-07-01
Developer	Justin Lieu	JL	2024-07-01
Developer	Kevin Xu	KX	2024-07-01
Developer	Matthew Kang	MK	2024-07-01
Developer	William Xiao	WX	2024-07-01

Introduction

This document serves as the Requirements Documentation for the Trek project, developed by Group 4: Trekkers, as part of CPSC 455 at the University of British Columbia. Following the Terms of Reference, this document provides a detailed outline of the functional and non-functional requirements necessary to build Trek, a personalized trip-planning web application.

Trek aims to leverage the latest advancements in ML/AI technologies to offer a seamless and intuitive trip-planning experience. This document captures the specific needs and expectations of the end-users, as well as the technical specifications required to meet these needs. The requirements detailed herein will guide the development process, ensuring that all aspects of the project align with the envisioned goals and deliverables.

Objectives

This document serves for the following objectives:

1. Define clear and comprehensive functional requirements that describe the behavior and interactions of Trek.
2. Specify non-functional requirements that address performance, usability, reliability, and other critical quality attributes.
3. Provide a foundation for the development, testing, and deployment phases, ensuring all stakeholders have a shared understanding of the project scope and objectives.

Definitions

Functional Requirements - These requirements define specific behavior or functions of the system. They describe what the system should do, such as features, operations, and interactions.

Non-Functional Requirements - These requirements specify criteria that judge the operation of a system, rather than specific behaviors. They include performance, usability, reliability, and security requirements.

MVP (Minimum Viable Product) - Essential features that must be implemented for the system to be functional and deliverable.

Standard - Features that add significant value to the product but are not critical for the initial release. These are implemented after the MVP is completed.

Stretch - Additional features that would be nice to have but are not essential. These are considered only if time and resources permit after completing MVP and Standard requirements

Initial Project Requirements

The initial project requirements were released on May 25th 2024.

Minimal Viable Product Requirements

The system (Trek) shall handle the following functionalities:

- Allows the user to create, view, and manage trips, as CRUD operations
- Allows the user to create and manage their own profile, as CRUD operations
- Provides a user-specific interface including list of trips the user is associated with
- Provides a trip creation interface for the users to create a new trip
- Allows a non-authenticated user to access the trip creation interface with most of its features
- Provides an access control interface for a user-created trip with sharing functionalities
- Provide a sign-in interface to a non-authenticated user when saving a user-created trip

The system (Trek) shall support the following non-functional requirements:

- Uses MongoDB to manage user information and support CRUD operations
- Uses MongoDB to store user-created trips

Standard Requirements

The system (Trek) shall handle the following functionalities:

- Allows multiple users to collaborate on a single trip, with real-time live collaboration
- Allows the user to share a trip via a URL
- Allows the user to search and explore destinations of a city

The system (Trek) shall support the following non-functional requirements:

- Uses internal APIs to aggregate trip information
- Uses external APIs (such as Google Maps Platform API) to obtain information regarding a location

Stretch Goals

The system (Trek) shall handle the following functionalities:

- Provides ML/AI suggested recommendation to the user regarding a trip
- Provides transportation options and associated costs to the user regarding a trip
- Provides weather information to the user regarding a trip
- Allows the user to interact with the system to automatically create an AI-generated itinerary

The system (Trek) shall support the following non-functional requirements:

- Uses external LLM API to implement and integrate LLM wrapper to the trip planning experience.

Refined Requirements

High-Level Goals

The high-level goals of the Trek project are:

User-Centric Design: Create an intuitive and user-friendly interface that simplifies the trip-planning process.

Personalization: Utilize ML/AI technologies to offer personalized trip recommendations and itineraries based on user preferences and behaviors.

Scalability: Ensure the application can handle a growing number of users and data without compromising performance.

Reliability: Develop a robust and reliable system that users can depend on for accurate and up-to-date information.

Integration: Seamlessly integrate with external APIs (e.g., Tripadvisor, weather services) to provide comprehensive travel information.

Impact on Different People and Stakeholders

The development and deployment of Trek will have varying impacts on different groups of people and stakeholders:

End-Users: Travelers looking for an easy and efficient way to plan their trips will benefit from a personalized, all-in-one trip-planning tool. They will experience increased convenience and satisfaction from using Trek to organize their travel plans.

Development Team: Members of Group 4: Trekkers will gain hands-on experience in software development, ML/AI integration, and project management, enhancing their technical skills and teamwork capabilities.

Future Developers: Clear documentation and a scalable design will make it easier for future developers to understand, maintain, and enhance the application, promoting long-term sustainability.

Instructors and Teaching Assistants: Ian McLean, Firas Moosvi, and Teaching Assistants will have the opportunity to guide and mentor the team, ensuring the project meets educational objectives and providing valuable feedback.

Broader Academic Community: The project serves as a practical example of applying theoretical knowledge to real-world problems, contributing to the academic discourse on software development and ML/AI applications.

By addressing the needs and expectations of these stakeholders, Trek aims to deliver a meaningful and impactful solution that stands out in the realm of trip-planning applications.

In Scope

The following functionalities and features are within the scope of the Trek project:

User Profile Management: Users can create, view, edit, and delete their profiles, which include personal information and travel preferences.

Trip Planning Interface: Users can create, modify, and manage trip itineraries, including destinations, accommodations, and activities.

Personalized Recommendations: Leveraging ML/AI technologies to provide users with personalized travel recommendations based on their preferences and past behaviors.

Interactive Map: Display trip details and routes on an interactive map to aid in visualization and transportation planning.

Budget Management: Tools for users to manage their trip budgets, including cost estimations for transportation, lodging, and activities.

Chatbot Assistance (stretch): An integrated chatbot to help users with trip planning tasks, providing suggestions and making changes based on user input.

Integration with External APIs: Using APIs from services like Tripadvisor for destination ratings, weather information, and other travel-related data.

User Feedback: Collect and manage user feedback to improve the application’s functionality and user experience.

Out of Scope

The following functionalities are not included in the scope of the Trek project:

Administrative Interfaces: There will be no dedicated interfaces for system administrators to manage the application.

Advanced User Analytics: Comprehensive analytics and reporting features for tracking user behavior and application usage beyond basic metrics.

Full-Scale Deployment: Deployment and maintenance of the application in a live, production environment able to handle 150+ concurrent users or other conditions beyond the scope of the course project.

Third-Party Advertising: Integration of third-party advertising or monetization features.

Constraints

Budget: The project budget is limited to $100, affecting the use of paid APIs and cloud services.

Timeline: The project must be completed within the timeframe of the CPSC 455 course, with specific deadlines for deliverables and presentations.

Technical Expertise: The team's skills and experience will influence the choice of technologies and the complexity of implemented features.

Academic Requirements: The project must meet the academic standards and requirements set by the course instructors and teaching assistants.

Dependencies

Integration & API Access: Access to external APIs such as OpenAI and Google's APIs is crucial for personalization and other functionalities. Each team member will contribute $20 towards this, totaling $100.

Development Tools: Tools and platforms such as React, Node.js, Express, MongoDB, and Auth0 will be utilized for building the application.

Team Collaboration: Effective collaboration and communication among team members are essential for the project's success. This includes regular meetings, use of Slack for communication, and adherence to the development roadmap.

Feedback and Testing: Input from TAs, instructors, and classmates during bi-weekly reviews and beta testing will be vital for refining and improving the application.

Documentation: Thorough documentation of the project's progress, technical design, and user feedback will be maintained to ensure clarity and facilitate future development.

Requirement Categories

The refined requirements can be organized into the following categories:

1. User Profile Management
2. Trip Planning Interface
3. Budget Management
4. Personalized Recommendations
5. Virtual Travel Agent (Stretch)
6. Performance and Scalability
7. Security
8. Usability

Requirements

The up-to-date requirements for Trek are available under the **Trek Requirements Spreadsheet**.