TRIP PLANNER

SOFTWARE REQUIREMENTS SPECIFICATIONS

Group members

Athullya R

Aditya Krishna Vijay

Sidharth M

Rohan Anil Kumar

Contents

Introduction

Purpose

The purpose of this Software Requirements Specification (SRS) document is to provide a comprehensive overview of the functional and non-functional requirements for a software application that generates 4-5 paths for users to visit tourist places within a given visiting area without exceeding their budget or time duration. This document serves as a guide for the developers and stakeholders involved in the development of the software application, ensuring that all requirements are clearly documented and understood. The SRS document outlines the scope and limitations of the software application, as well as the constraints, assumptions, and dependencies that may affect its development and implementation. By providing a clear and concise description of the software requirements, this document ensures that the software application meets the needs and expectations of its users.

Scope

This SRS document describes a software application that generates feasible travel paths for users who want to visit tourist places within a specified budget and time duration. The application will generate 4-5 paths based on the user's inputs of starting point, visiting area, budget, and time duration, including the tourist places to visit, mode of transportation, and accommodations within budget constraints. The web-based application will be accessible through a web browser and designed to be user-friendly, intuitive, and easy to use for users with basic computer literacy. The application will handle multiple user requests simultaneously, generating paths within seconds.

References

This SRS document refers to the following external documents:

1. Algorithms for the weight constrained shortest path problem by Irina Dumitrescu and Natashia Boland
2. Effective Indexing for Approximate Constrained Shortest Path Queries on Large Road Networks by Sibo Wang, Xiaokui Xiao, Yin Yang, Wenqing Lin

Overview

The remaining sections of this document will describe the functional and non-functional requirements for the software application. The requirements will be organized into sections that describe the general requirements, functional requirements, non-functional requirements, constraints, and assumptions and dependencies.

Stakeholder Analysis

Identification of Stakeholders

- Primary Stakeholders

  Traveler, Travel Agencies Tour operators and service providers like accommodation, food and transport. Hotels, Restaurants and other tourism related business are key stake holder in the tourism department.

- Secondary Stakeholders

  Government agencies (Kerala tourism department), non-governmental organizations. NGO's that work in areas such as environmental protection cultural preservation and community development can also be stakeholders in tourism department. Media outlets such as newspapers, Television Station and Travel websites can also be considered secondary stakeholders in the tourism department. They may influence public opinion about a destination and affect the number of Tourists who choose to visit.

Requirements Proposed by Stakeholders

- Distance, distance also affects the accessibility of a destination. Remote locations may be more challenging to reach due to limited transportation options, which may require additional planning and coordination.
- Budget, the cost of transportation, accommodation, and other expenses associated with travelling increases with distance. Long-distance travel may require more extensive planning and budgeting to ensure that costs are manageable.
- Duration, the time required for travelling is a crucial factor for planning trips, and can make travelling organized.

Requirement Gathering Methodologies

- Contacting tour operators

  Experts in the field of travel organizing were connected through various platforms like LinkedIn and email. The priorities, primary factors and precautions to be considered were organized and analyzed.

- Public opinion survey

  In TKMCE, Google forms was circulated to analyze the general public considerations while travelling and their preferred mode of transportation and services.

General Requirements

Product Perspective

The software application will be a standalone web application that can be accessed from any modern web browser. The application will utilize external data sources to gather information about tourist destinations, transportation options, and accommodations.

Product functions

- Login
- Select route type.
- Select destination.
- Select source.
- Select time constraints.
- Select budget constraints.
- Select the ratings preferences of the stay required.
- Output 5-6 optimal routes based on the given conditions.
- Select preferred route to view detailed information about the route.

User Classes and Characteristics

The target audience for this software application is tourists who want to plan a visit to a particular area. The users of this software application will have basic computer literacy and be familiar with using web applications.

Operating Environment

The software application will be designed to work on any modern web browser, including Chrome, Firefox, and Safari. The application will require an internet connection to access external data sources.
In the production environment, the web application should have access to a MongoDB database.

Design Implementation and Constraints

The software application will be designed using HTML, CSS, JavaScript, and any necessary web application frameworks. The application will be implemented using a modern web application server.

External Interface Requirements

User Interfaces

- Login and authentication:

  The user should be able to log into the application using his email and password. Once logged, the user will be greeted with a dashboard that he can use to find the routes to his desired destination.

- User Route Dashboard:

  The user gets the dashboard upon login. The user can then select the route type from a carefully given catalogue of route types.

- User Route Input:

  Upon selecting the route type, the user is greeted with a page that allows him to enter his preferred details for the route. The page asks for the destination and budget constraints.

- Route Results:

  This page returns a bunch of routes optimized according to the users' inputs. The user can then proceed to select between these routes to get the detailed view of the route.

- Route detailed view:

  This page details the route that the user has selected so that the user can plan the trip according to the given route.

Hardware Interfaces

- Computer system: This includes the hardware components used for running the handwriting recognition software and performing the evaluation. This can include processors, memory, storage, and other components like keyboard, mouse, touch screen and a display.
- Network connection: A network connection is required to upload and download files, access online resources, and communicate with other users or systems.

Software Interfaces

- Graphical User Interface (GUI): The graphical user interface of the software is the primary interface for users to interact with the software. It includes features like menus, icons, buttons, and other graphical elements that allow users to navigate and interact with the software.
- Database Interface: The database interface allows the software to interact with a database or data storage system to store and retrieve data related to the student answers, grading criteria, and reports.
- Google Map Interface: Google map interface allows us to get the destinations, accommodations, Food and drink, Services, Attraction. This includes Hotels, Campgrounds, Restaurants, Coffee, Fast Food, Groceries, Rest Stop, gas, EV charging, Bank, Things to do, Events Parks, Museum.

- External API's: Used for getting restaurant prices, nearby locations, hotel booking prices and all the things that use the transaction of money.

Functional Requirements

Input Requirements

The software application will require the following input from the user:

- Starting point: the user's current location
- Visiting area: the area that the user wants to visit.
- Budget : the maximum amount of money the user is willing to spend, including transportation and accommodation. Time duration: the amount of time the user has to visit the area

Output Requirements

The software application will generate the following output for the user: 4-5 different paths for the user to visit the tourist places in the visiting area. The cost of each path, including transportation and accommodations. The time required to complete each path.

Processing Requirements

The software application will perform the following processing steps:

- Retrieve information about tourist destinations, transportation options, and accommodations from external data sources.
- Calculate the distance and time required to travel between each destination.
- Generate multiple paths based on the user's input, ensuring that each path does not exceed the user's budget or time duration.

Data Requirements

The software application will require the following data:

- Geographic coordinates for each tourist destination
- Price information for transportation options and accommodations
- Time required to travel between each destination.
- Distance between each destination

Nonfunctional Requirements

Performance Requirements

The software application must be able to generate 4 different paths for the user to visit the tourist places in the visiting area in 5 seconds. The application must be able to handle at least 10,000 simultaneous requests.

Security Requirements

The software application must ensure that user data is protected from unauthorized access. The application must implement proper authentication and authorization mechanisms to ensure that only authorized users can access the system.

Usability Requirements

The software application must be easy to use and navigate for users with basic computer literacy. The application must provide clear and concise instructions to guide the user through the process of entering their input and reviewing the generated paths.

Maintainability Requirements

The software application must be designed and implemented in a modular and extensible manner to enable easy maintenance and updates. The application must be well-documented to facilitate future development and maintenance efforts.

Constraints

Budget Constraints

The software application must ensure that the generated paths do not exceed the user's specified budget, including transportation and accommodations.

Time Constraints

The software application must ensure that the generated paths can be completed within the user's specified time duration.

Data Source Constraints

The software application must rely on external data sources for information about tourist destinations, transportation options, and accommodations.

Assumptions and Dependencies

Assumptions

The software application assumes that the external data sources for tourist destinations, transportation options, and accommodations will provide accurate and up-to-date information.

Dependencies

The software application is dependent on a modern web application server and web application frameworks to implement the required functionality. The application is also dependent on external data sources for information about tourist destinations, transportation options, and accommodations.

Literature Review

- Algorithms for the weight constrained shortest path problem by Irina Dumitrescu and Natashia Boland

  In the document a directed graph is given whose arcs have an associated cost, and associated weight, the weight constrained shortest path problem (WCSPP) consists of ending a least-cost path between two specified nodes, such that the total weight along the path is less than a specified value.

- Effective Indexing for Approximate Constrained Shortest Path Queries on Large Road Networks by Sibo Wang, Xiaokui Xiao, Yin Yang, Wenqing Lin’

  In a constrained shortest path (CSP) query, each edge in the road network is associated with both a length and a cost. Given an origin s, a destination t, and a cost constraint θ, the goal is to find the shortest path from s to t whose total cost does not exceed θ.

Conclusion

The software application is designed to help users plan their travels in a given area, while taking into consideration the user's budget, time duration, and visiting the area. It will provide the user with 4-5 paths to visit tourist places. in the visiting area, while ensuring that the user does not exceed them budget or time duration. It is designed using modern web technologies, with scalability, maintainability, and performance in mind, and will rely on external data sources and APIs for tourist place information, transportation options, and prices.