Kathmandu University
Department of Computer Science and Engineering
Dhulikhel, Kavre

A Project Report
on
"Travel Recommendation System"

[Code No.: COMP 202]

(For partial fulfillment of Year II / Semester I in Computer Engineering)

Submitted by
Prajaya Shrestha (038002-24)

Submitted to
Sagar Acharya
Department of Computer Science and Engineering

February 18, 2026

Bona fide Certificate

This project work on

“Travel Recommendation System”

is the bona fide work of

“

Prajaya Shrestha (038002-24)

”

who carried out the project work under my supervision.

Project Supervisor

Sagar Acharya
Lecturer
Department of Computer Science and Engineering

Acknowledgements

There were many helping hands that helped us on my mini project journey. First of all, this project would not have been possible without the support and guidance of my teacher Mr. Sagar Acharya.

I also thank the Department of Computer Science and Engineering, Kathmandu University for giving me this great opportunity for showcasing and expanding our knowledge.

Abstract

This project is travel recommendation system that suggests similar travel destination across Nepal according to the user's preferences. The system was designed to address the difficulty users face while selecting suitable travel locations from a large number of available options.
The project was carried out to simplify the process of choosing the best travel destinations that fits the user's taste. It is recommended that similar future projects to implement features such as user feedback mechanisms, real time data and many more. The system showed promising results in providing personalized travel recommendations.

Keywords: *Travel Recommendation System, Qt, Data Structures and Algorithm,C++*

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Tourism has rapidly adopted digital technologies such as Artificial Intelligence and Machine Learning to improve how travelers search, plan, and experience destinations. Modern tourism platforms now use recommendation systems, chatbots, and interactive tools to provide faster and more intelligent guidance. Recent developments mainly focus on data-driven personalization to improve user satisfaction and decision-making.

However, most previous tourism systems simply suggest places based on a general context, such as location or popularity. They do not ask about user preferences, interests, or travel styles. As a result, travelers often receive recommendations that do not match what they truly want, reducing satisfaction and engagement with the platform.

1.2 Objectives

The objectives of this project are listed as follows:

- To design and develop platform for Nepal that provides personalized travel recommendations.
- To suggest places that fit the interests of each user.
- To use different DSA topics into use.

1.3 Motivation and Significance

Nepal is a country of stunning landscapes, rich culture, and diverse experiences, attracting millions of tourists each year. From the towering Himalayas to rivers, ancient temples, and unique cuisines, the country offers many attractions. However, despite these options, travelers often struggle to plan their trips effectively. Most travel guides and online platforms provide general information and do not consider individual preferences or interests, making it hard for tourists to enjoy a truly personalized experience. This project aims to simplify this very problem.

Chapter 2

Related Works

2.1 Review of Existing Works

The field of tourism technology and travel recommendation systems has seen significant growth in recent years. Several platforms have been developed to assist travelers in discovering destinations and planning trips. Antaranga aims to provide similar services with enhanced features such as interest based clustering, analysis of user preferences and behavior, and personalized recommendations to help travelers make informed choices and plan better experiences.

2.1.1 Trip Turbo

TripTurbo (2025)Travel Recommendation Platform provides a structured repository of travel options with fields for destination, price range, travel dates, accommodation type, and user search preferences. The system supports basic analytics for booking trends and popular routes, illustrating standard practices for designing scalable travel and booking platforms.

Strengths:

- User-friendly interface for browsing destinations and travel deals.
- Provides basic recommendations based on popular routes and user searches.
- Integrates booking options for flights and hotels.

Limitations:

- Relies mainly on static filters and general popularity trends.
- Does not use advanced machine learning for deep personalization.

2.1.2 Kayak

Kayak (2025) Travel Recommendation Platform provides a structured repository of travel options with fields for destination, price range, travel dates, transportation mode, and accommodation type. The system supports basic analytics for flight and hotel trends, helping users compare options and find deals efficiently.

Strengths:

- Powerful search and filtering tools for flights, hotels, and rental cars.
- Provides recommendations based on price, availability, and travel dates.
- Integrates booking and comparison options in a single platform.

Limitations:

- Primarily focuses on price and availability rather than user-specific interests.
- Offers limited personalization beyond general search preferences.
- Does not analyze long-term user behavior for tailored recommendations.

2.1.3 Spatial clustering in tourism recommendation

Li et al. proposed spatial clustering techniques for personalized travel recommendation systems Li et al. (2024). Their approach grouped tourist destinations based on geographical attributes and user preference data to improve recommendation accuracy. The system demonstrated how spatial data mining can enhance tourism recommendations through intelligent grouping of locations.

Strengths:

- Improved personalization using spatial clustering.
- Effective grouping of destinations based on geographic features.
- Enhanced recommendation relevance.

Limitations:

- Requires large datasets for effective clustering.
- Involves complex spatial data processing.

2.1.4 Hybrid tourism recommendation

Goel and Rizvi developed a hybrid tourism recommendation model integrating content-based and collaborative filtering techniques Goel and Rizvi (2024). Their model achieved improved recommendation accuracy using user interaction data.

Strengths:

- Hybrid filtering improves recommendation precision.
- Utilizes user behavior for personalization.

Limitations:

- Depends heavily on sufficient user data.
- Cold-start problem for new users.

Chapter 3

Design and Implementation

3.1 System Overview

The Travel Recommendation Mini Project is a desktop-based application developed using C++ and the Qt framework. The main objective of the system is to allow users to explore travel destinations through a graphical interface.

The system consists of three main components:

- Graphical User Interface (GUI)
- Location Data Module
- Recommendation Logic

The GUI interacts with the backend logic to display destinations and related information. User inputs are processed and corresponding results are shown on the interface.

3.2 System Requirement Specifications

3.2.1 Software Requirements

- Programming Language: C++
- Framework: Qt 6
- IDE: Qt Creator
- Version Control: Git and GitHub
- Operating System: Windows

Functional Requirements

- Display list of travel destinations
- Show destination details
- Allow user navigation between screens
- Provide basic recommendations

Non-Functional Requirements

- Easy to use interface
- Fast response time
- Reliable execution
- Simple and clean design

3.2.2 Hardware Requirements

- Minimum 4GB RAM
- Intel/AMD processor
- Windows-based computer

3.3 System Design

3.3.1 Architectural Design

The application follows a simple modular architecture. The GUI layer is separated from the logic layer. Qt widgets handle user interaction while C++ classes manage data processing.

3.3.2 Module or Component Design

Major modules include:

- Main Window Module – handles user interface
- Location Module – stores destination data
- Recommendation Module – processes recommendations

3.3.3 Database or Data Design

The project uses internally stored datasets in C++ structures. Each location contains fields such as name, description, and category.

3.4 Implementation Details

3.4.1 Software Implementation

Qt widgets, signals, and slots were used to build the interface. C++ classes were created to manage destinations and logic. The frontend and backend were connected through Qt's event system.

3.4.2 Hardware Implementation

Not applicable as this is a software-based project.

3.5 Algorithms and Flowcharts

The main algorithm follows these steps:

1. Start application
2. Load travel data
3. Display destinations
4. Accept user input
5. Show recommendation

Flowcharts were used to visualize user navigation and recommendation flow.

3.6 Tools and Technologies Used

- C++
- Qt Framework
- Qt Creator
- Git
- GitHub

3.7 Summary

This chapter explained the design and implementation of the Travel Recommendation Mini Project. The system was developed using C++ and Qt to create a user-friendly desktop application. Modular design was adopted to separate GUI and logic components. The project demonstrates practical use of object-oriented programming and GUI development concepts. The system can be further improved by adding databases, maps, and advanced recommendation features.

Chapter 4

Results and Discussion

This chapter presents the results obtained from the implementation of the Travel Recommendation Mini Project developed using C++ and Qt. It discusses the completed features, system performance, challenges faced, and overall effectiveness of the application.

4.1 Implemented Features

The following key features were successfully implemented:

- Graphical User Interface (GUI) for interacting with the system
- Display of travel destinations with basic details
- Navigation between different screens
- Simple recommendation logic based on predefined data
- Modular backend structure using C++ classes

These features allow users to explore destinations easily through a desktop interface while demonstrating core concepts of GUI programming and data handling.

4.2 Results and Performance Analysis

The application runs smoothly on a standard Windows system without noticeable delays. User inputs such as button clicks and navigation actions are processed instantly.

The system correctly loads travel datasets and displays them on the interface. Screenshot outputs of the main window and recommendation screens were obtained and verified during testing.

Overall performance is satisfactory for a mini project, as the dataset size is moderate and processing requirements are minimal.

4.3 Comparison with Objectives or Existing Systems

The original objectives of the project were:

- Build a desktop application using Qt
- Implement basic travel recommendation logic
- Apply data structures and modular design

- Create a usable frontend

All primary objectives were achieved. Compared to professional travel applications, this project offers limited features; however, it successfully demonstrates fundamental concepts such as GUI development, C++ programming, and basic recommendation logic.

Due to time constraints, advanced features like online APIs, real-time maps, and databases were not implemented.

4.4 Challenges and Limitations

Several challenges were encountered during development:

- Initial difficulty in configuring Qt and build files
- Integrating frontend with backend logic
- Managing GitHub repository and executable deployment
- Limited dataset and simple recommendation approach

These challenges were addressed through debugging, restructuring code, and incremental testing. Some limitations remain, including lack of database support and advanced filtering options.

4.5 Discussion

The project successfully demonstrates how a C++ application can be combined with Qt to create a functional GUI-based system. Although the recommendation logic is basic, the system meets academic requirements and provides a solid foundation for future improvements.

The results show that modular programming and GUI integration were achieved effectively. This mini project helped strengthen understanding of data structures, object-oriented programming, and software development workflow.

Future enhancements could include database integration, map services, user authentication, and smarter recommendation algorithms.

Chapter 5

Conclusion and Future Works

This project successfully implemented a desktop-based Travel Recommendation System using C++ and the Qt framework. The primary objective of developing a GUI-based application that demonstrates core programming concepts such as data structures, modular design, and frontend–backend integration was achieved.

The system allows users to view travel destinations, receive basic recommendations, and interact through a graphical interface. Throughout the development process, practical knowledge of Qt Creator, C++ programming, GitHub version control, and application deployment was gained. The project also helped strengthen understanding of software development workflow and debugging techniques.

Although the recommendation logic is simple, the application meets the academic goals of the mini project and provides a foundation for future expansion.

5.1 Limitations

Despite achieving the main objectives, the system has several limitations:

- The dataset is small and manually defined.
- Recommendations are based on basic logic without advanced filtering or personalization.
- No database integration; all data is stored in memory.
- The application was tested only on a local Windows environment.
- Performance optimization and scalability were not explored due to time constraints.

These limitations are mainly due to limited development time and the academic scope of the project.

5.2 Future Enhancements

Several improvements can be made in future versions of the system:

- Integration of a database for storing destinations and user preferences.
- Implementation of advanced recommendation algorithms.
- Addition of user login and profile management.
- Inclusion of maps and real-time location services.
- Expansion of the dataset and support for online APIs.

- Cross-platform deployment for Linux and macOS.

These enhancements would make the system more practical for real-world use while improving scalability and user experience.

References

Goel, S. and Rizvi, S. (2024). Travel recommendation system using content and collaborative filtering. *Journal of Mechanical and Construction Engineering*, 4(2):1–8.

Kayak (2025). Kayak travel search and recommendation platform. `https://www.kayak.com`.

Li, C., Cui, J., and Li, Y. (2024). Research on system development of spatial clustering in tourism recommendation. *Applied Mathematics and Nonlinear Sciences*.

TripTurbo (2025). Tripturbo travel recommendation platform. `https://www.tripturbo.com`.

Appendix A

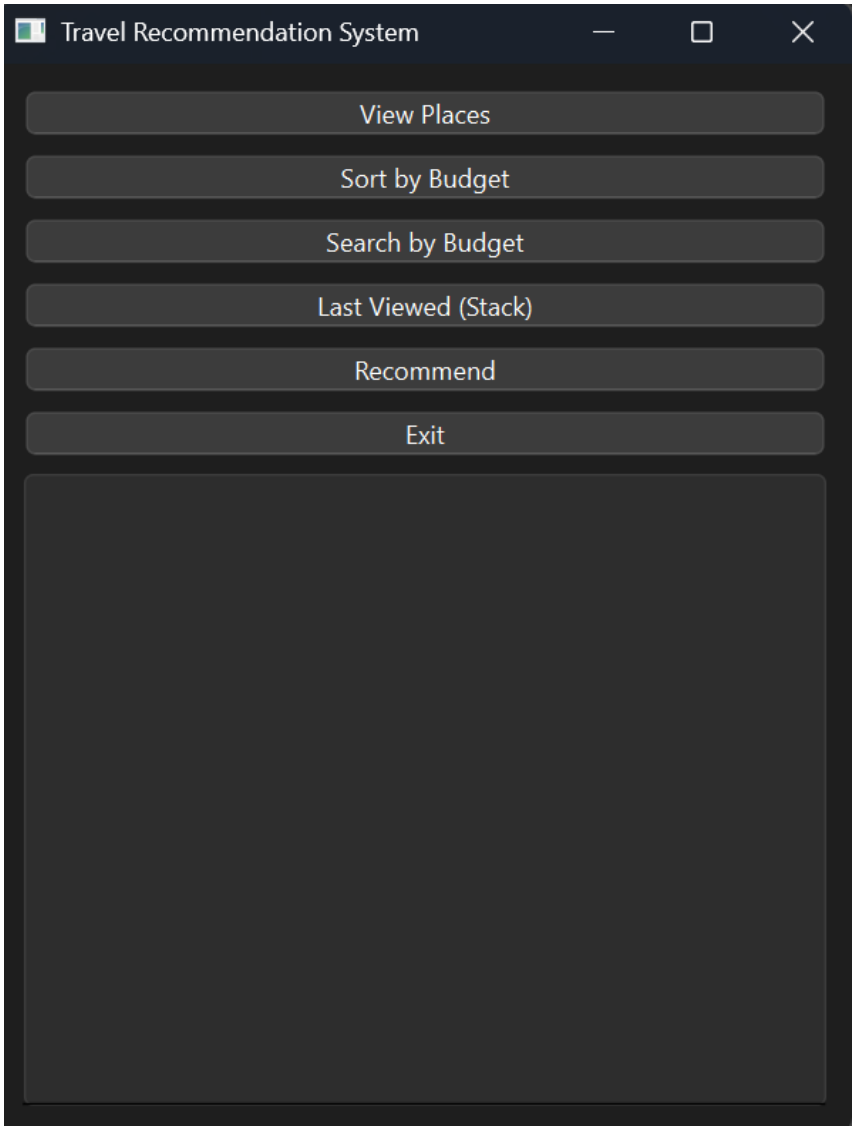

Figure 5.1: Main page

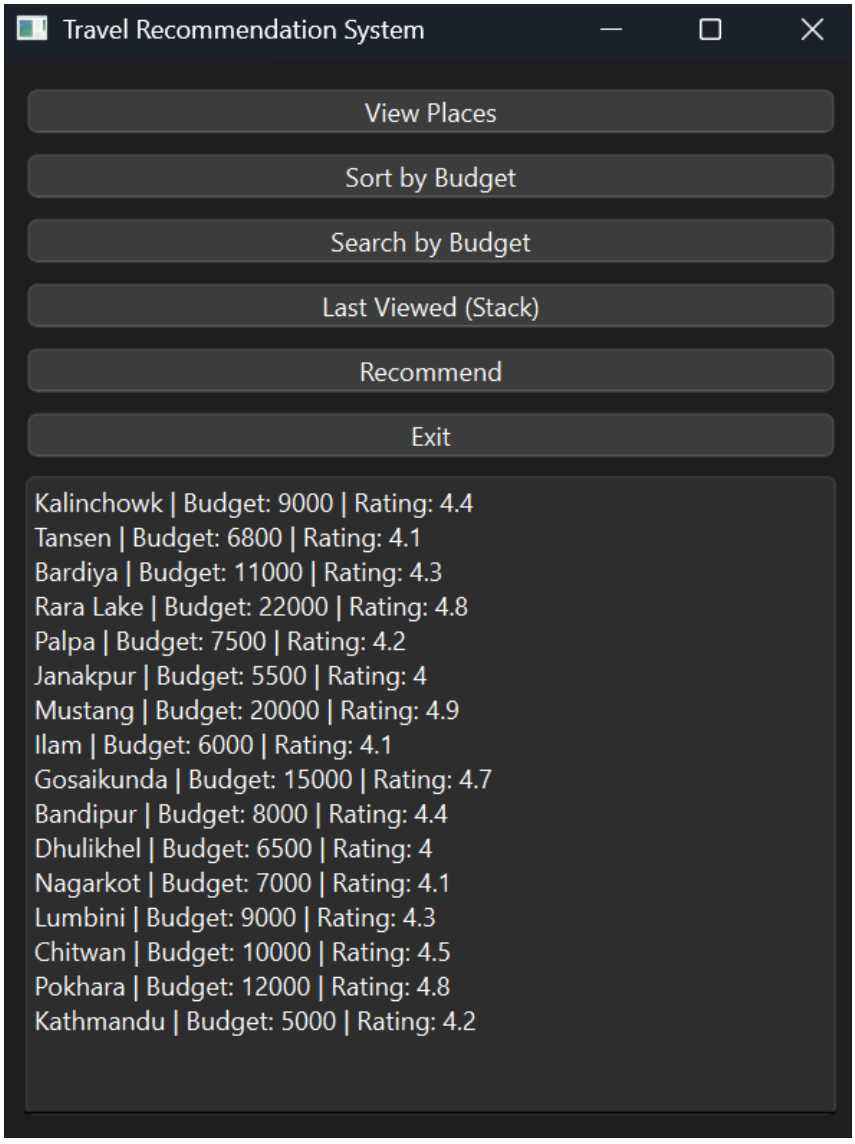

Figure 5.2: View Places

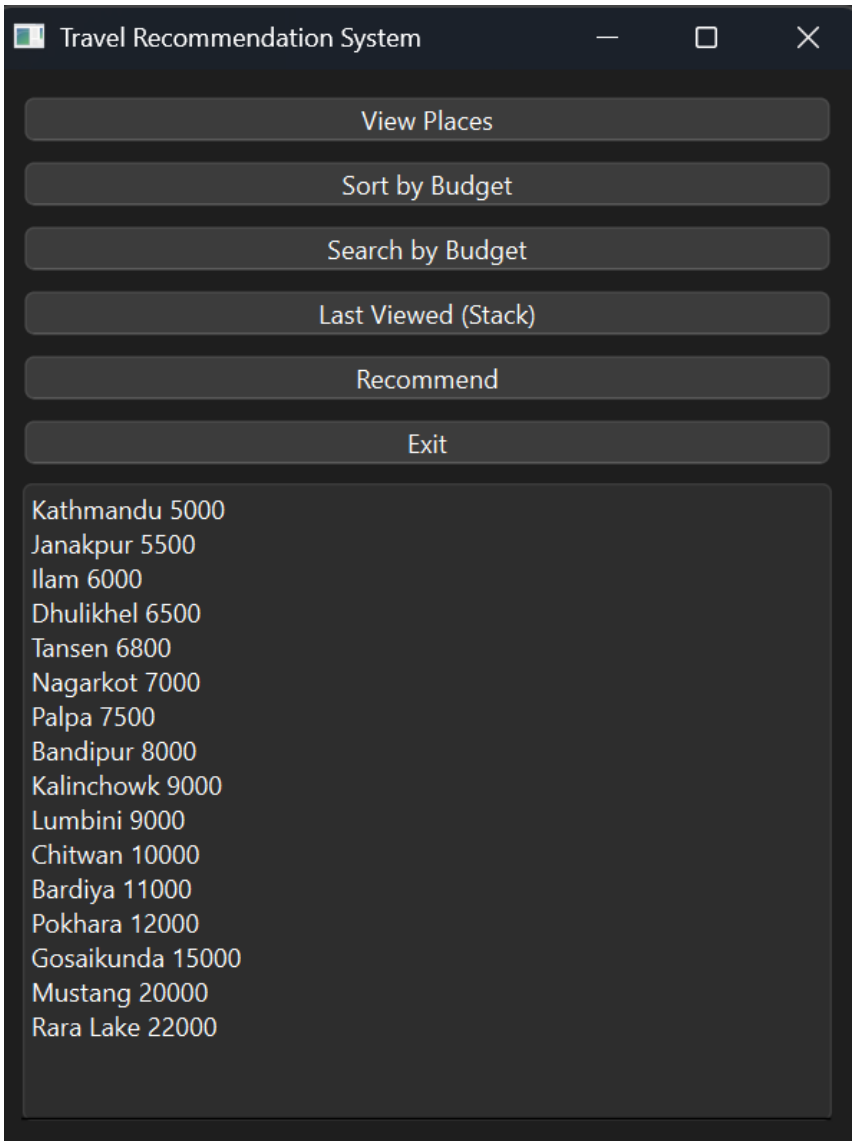

Figure 5.3: Sort by budget

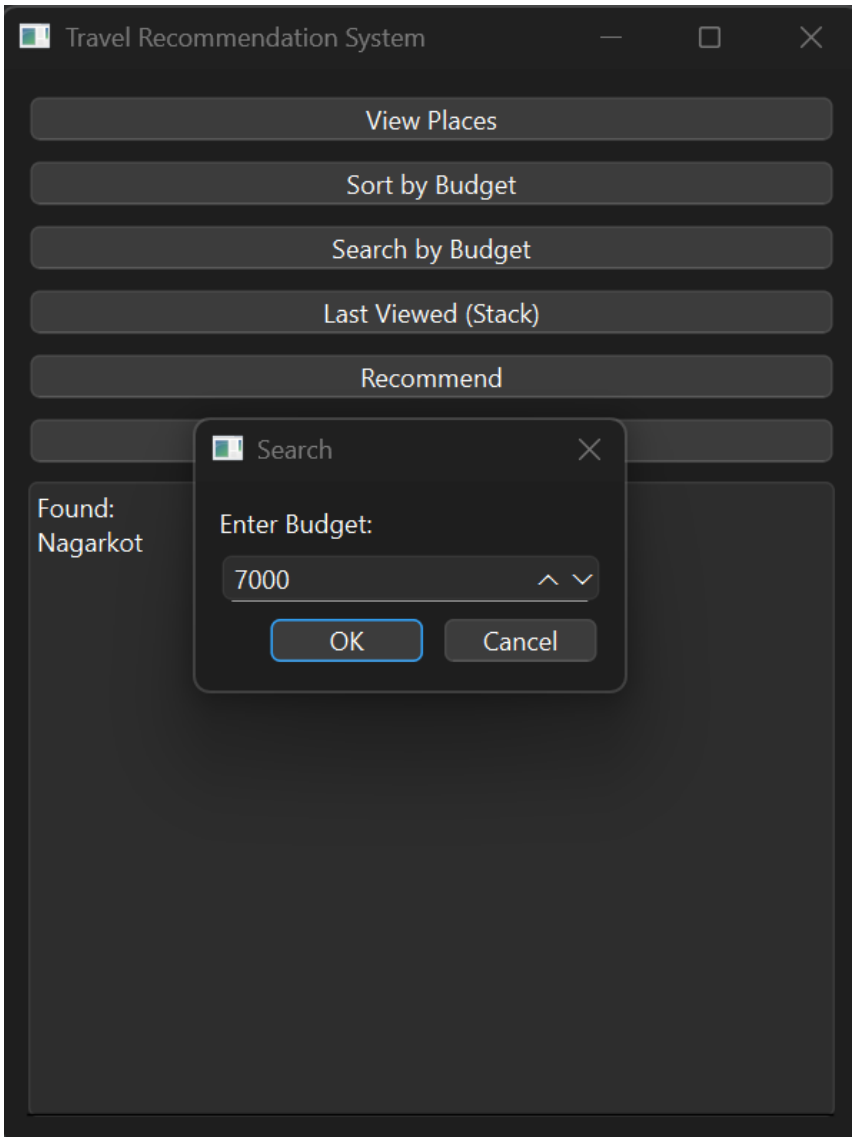

Figure 5.4: Search by budget

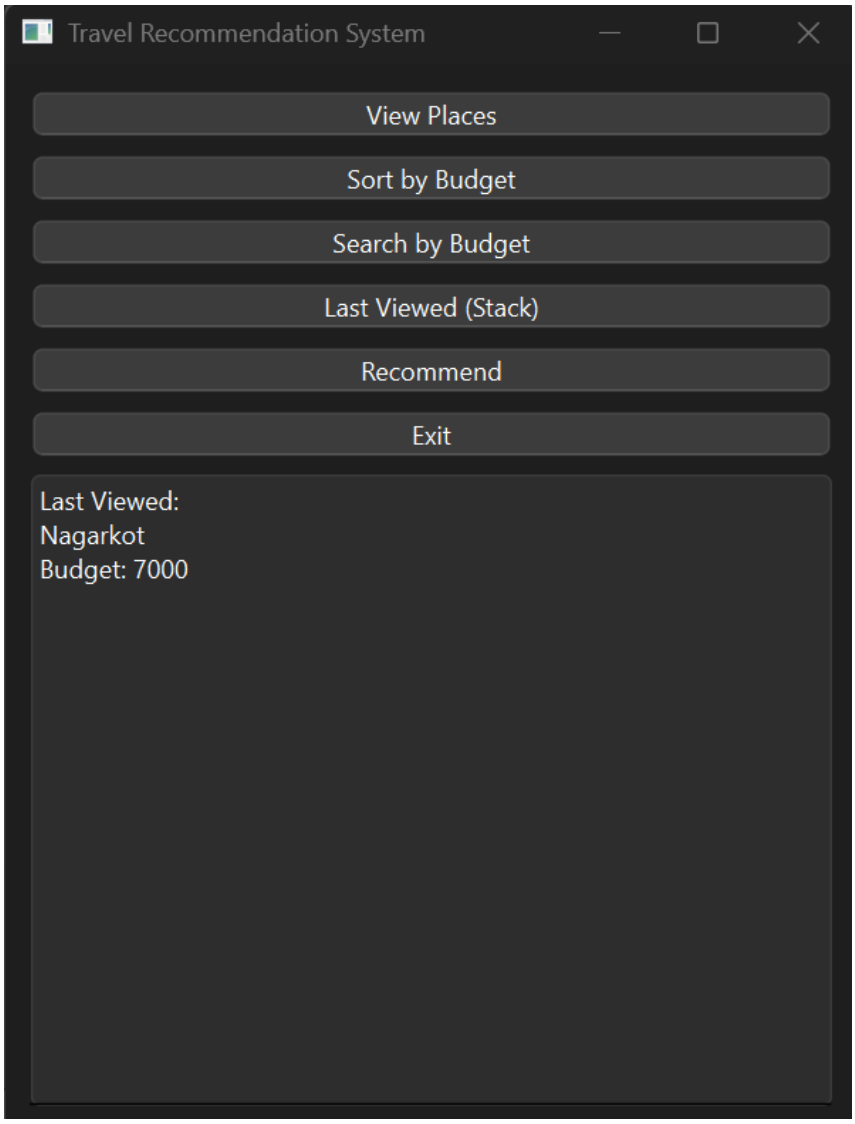

Figure 5.5: Last viewed

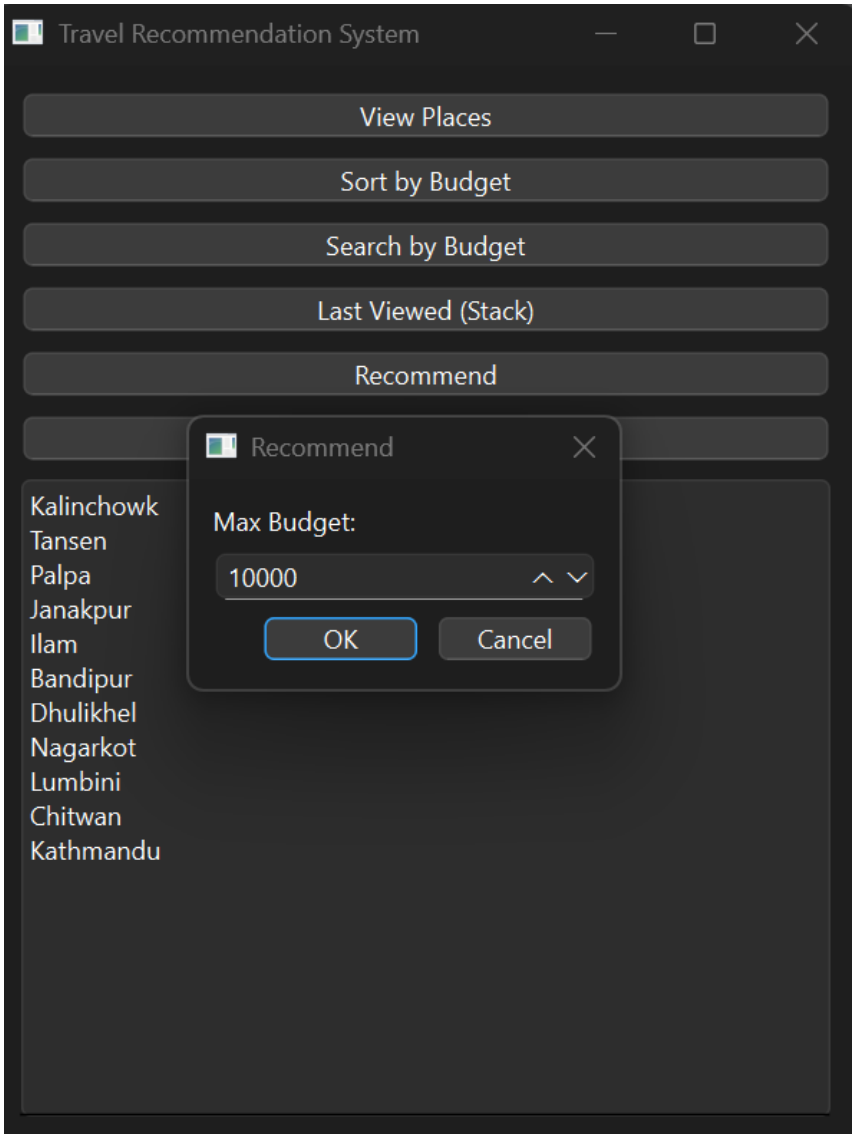

Figure 5.6: Recommend