BRAUDE
College of Engineering, karmiel

Capstone Project Phase A

Traveler's Hub

Streamlining Trip Planning for Travelers and Organizers

Project Number
24-1-D-4

Supervisor: Dr. Zeev Barzily

Shiran Giladi (Shiran.Giladi@e.braude.ac.il)
Lev Velitsky (lev.velitsky@e.braude.ac.il)

May 12, 2024

Table of Contents

Abstract

We designed Traveler's Hub to redefine the complexities of trip planning, establishing a unified platform for travelers and tour organizers to streamline their planning processes. This document delves into the existing trip planning tools, identifying their shortcomings and demonstrating how Traveler's Hub offers a superior, integrated solution. Utilizing a robust technology stack, including React and Flask, the application embraces a client-server architecture to facilitate real-time data handling and user interaction. Special attention is given to the user interface, ensuring a smooth and intuitive user experience across various functionalities such as trip searches, itinerary management, and community engagement in forums. The development strategy incorporates rigorous unit and integration testing to verify the application's functionality and user experience. It ensures Traveler's Hub meets and exceeds its users' needs by providing a comprehensive, all-in-one travel planning resource.

Chapter 1

Introduction

Traveling is a delightful yet complex endeavor that requires meticulous planning and coordination. Whether you're a seasoned traveler or a tour operator, organizing a trip involves numerous details, such as choosing destinations, planning activities, and managing costs. However, the current process is often fragmented, with limited tools for seamless trip planning. This gap presents an opportunity for a comprehensive solution that simplifies and enhances the trip planning experience.

1.1 Problem Statement

The current trip planning process is fragmented, involving multiple platforms and tools that do not seamlessly integrate. This leads to inefficiencies as travelers and tour operators struggle to organize and communicate trip details effectively. Existing solutions lack a cohesive approach, often focusing on specific aspects of trip planning rather than providing a comprehensive platform.

1.2 Existing Solutions

Various websites and applications offer trip planning tools, but they often focus on singular aspects such as booking flights or accommodations. While these tools serve specific needs, they lack integration, making creating and managing comprehensive travel plans challenging. Additionally, the lack of community engagement features limits collaboration and idea-sharing among travelers and tour operators.

1.3 Proposed Solution

Traveler's Hub aims to revolutionize trip planning by providing a unified platform that integrates all aspects of the planning process. By combining features such as personalized profiles, Adventure Canvases, Concepts, and forums, Traveler's Hub offers a solution for creating, sharing, and engaging with travel plans. This comprehensive approach simplifies the planning process and fosters community and user collaboration.

1.4 Stakeholders

Traveler's Hub's stakeholders include individual travelers, tour operators, and the travel industry. The platform offers individual travelers a user-friendly interface for planning trips and discovering new destinations. Tour operators benefit from a platform that simplifies the creation and promotion of travel packages. Overall, the travel industry benefits from increased efficiency and collaboration, enhancing travel experiences for everyone involved.

Chapter 2

Background and Related Work

2.1 Tools and Methods for Trip Planning Applications

Trip-planning applications have become increasingly popular, offering users convenience and efficiency in organizing their travel itineraries. In this literature survey, we explore existing tools and methods relevant to the development and functionality of Traveler's Hub, a web application designed to streamline the trip-planning process for both customers and tour organizers.

2.1.1 Existing tools

1. **TripIt:** TripIt is a popular travel organization app that helps users create detailed itineraries by simply forwarding their travel confirmation emails. It organizes flights, accommodations, and activities into a single itinerary, which can be accessed online or offline.

2. **Google Travel:** Google Trips offers a comprehensive trip planning experience by offering users recommendations for activities, restaurants, and attractions based on location and preferences. It also allows users to organize their itinerary and access it offline.

3. **TripHobo:** TripHobo is a trip planning platform that enables users to create detailed itineraries by selecting activities, attractions, and accommodations from a vast database. It also offers collaboration features, allowing users to share their plans with others.

4. **Roadtrippers:** Roadtrippers is a tool specifically designed for planning road trips, allowing users to discover attractions along their route and add them to their itinerary. It also provides information on hotels, restaurants, and other points of interest.

2.2 Literature Review

The literature surrounding trip planning applications emphasizes the importance of user-centered design, personalized recommendations, and seamless integration of travel ser-

vices. One study by Wang et al. (2016) [3] discusses the impact of smartphones on travel, highlighting the need for travel applications to adapt to the new era of smartphone use. This aligns with the key features of Traveler's Hub, which aims to provide personalized travel plans tailored to each user's preferences while considering their digital interactions.

Additionally, tour organizers play a crucial role in the trip planning process. They specialize in crafting unique travel experiences and can benefit from tools that help them streamline their services and connect with travelers. Research by Tanti and Buhalis (2017) [2] examines how digital connectivity impacts tour organizers' ability to deliver quality travel experiences, emphasizing the importance of technology in enhancing their offerings. This underscores the value of Traveler's Hub in providing tour organizers with a platform to efficiently plan trips, maintain organizational efficiency, and foster connections with travelers.

Moreover, the importance of user-generated content and social interactions in influencing travel decisions is evident in the literature. Xiang et al. (2010) [4] explored the role of social media in trip planning, emphasizing the significance of user reviews and recommendations. Similarly, Gursoy and Umbreit (2004) [1] found that user-generated content, such as traveler reviews, significantly impacts the decision-making process of potential travelers. This underscores the value of the Travel Guides functionality in Traveler's Hub, which allows users to create and share reviews, engage in forum discussions, and contribute to the overall travel planning experience.

Chapter 3

Expected Achievements

3.1 Objectives

The following objectives are aimed at delivering a comprehensive and user-friendly travel planning application:

1. **Trip Planning Interface:** Develop a user-friendly interface for customers (travelers) to search for trips and for tour organizers to plan and manage trips efficiently.
2. **Trip Planner Tool:** Create a Trip Planner tool that allows tour organizers to create detailed travel plans, including trip names, start and end dates, notes, activities, and budgets.
3. **Trip Search:** Enabling users to search for trips based on their preferences using filters like dates, budget, and more.
4. **Travel Guides:** Enabling trip organizers to include a travel guide for each trip featuring safety tips, recommended equipment, weather information, and more.
5. **Forum Discussions:** Enable users to engage in forum discussions related to travel, opening chats for queries and responses.

3.2 Success Criteria

The following criteria will measure the success of this project:

1. **Project Completion:** Successfully developed and presented a fully functional version of the Traveler's Hub application, as outlined in the project objectives and expected achievements.
2. **Feature Implementation:** Achieve the full implementation of all planned features, including the Trip Planner Tool, Trip Search, Travel Guides, and Forum Discussions, as detailed in the project specifications.
3. **Technical Proficiency:** Demonstrate technical proficiency and problem-solving skills by successfully integrating technologies such as React.js, Flask, and other tools specified in the project design. This includes the application's functionality, performance, and security measures.

4. **Trip Support:** Ensure that the system supports various trips by accommodating various preferences and requirements, enhancing the user experience, and expanding the application's utility.

3.3 Challenges

While developing Traveler's Hub, there are several challenges that we expect to address:

1. **Cross-Browser Compatibility:** Ensuring Traveler's Hub functions correctly across various web browsers like Chrome, Firefox, Safari, and Edge is challenging because each browser has its own rendering engine, potentially leading to display inconsistencies. We'll use fonts and styles compatible with all browsers to address this.

2. **Mobile Responsiveness:** To accommodate the growing use of mobile devices for internet browsing, Traveler's Hub prioritized mobile responsiveness in its design. The application will be designed to adapt to different screen sizes and resolutions seamlessly. Testing will also be conducted on various mobile devices to ensure the application's responsiveness.

3. **Secure Password Storage:** Implementing a secure method for storing user passwords is a key challenge in developing Traveler's Hub. We are utilizing random salts and a slow hash function to address this. When users create or update their passwords, we combine a random salt with the password before hashing it using a slow hash function. This ensures that even if password hashes are compromised, they cannot be easily cracked without knowledge of the salts, enhancing the security of user accounts.

Chapter 4

Process

In the initial phase of the Traveler's Hub project, we focused on comprehensive planning and documentation. We began by conducting a detailed review of existing trip-planning applications. This research helped us identify successful functionalities and common pitfalls, which informed the unique features and capabilities that Traveler's Hub aims to offer. The insights gained from this research formed the basis for defining the project's requirements and system specifications.

Concurrently, we developed preliminary system designs, exploring how best to integrate React for the front end and Flask for the back end to ensure seamless interaction and scalability. These theoretical models and sketches were essential for visualizing the architecture and preparing for the technical challenges ahead.

As we unfolded the planning, we encountered significant challenges, particularly ensuring that the selected technologies were compatible and could scale according to the system's needs. We addressed this by simulating interactions and conducting theoretical stress tests on the system architecture. Another challenge was designing an intuitive user interface that could handle the complex functionalities required by Traveler's Hub. To tackle this, we engaged in problem-solving sessions and created interface mock-ups, which allowed us to iterate on design concepts effectively.

We also crafted a roadmap for the subsequent development phase during this phase, prioritizing features and establishing timelines. This roadmap was crucial for structuring the project's progression and ensuring that all critical aspects were methodically addressed.

This methodical approach in the planning phase was designed to align with academic standards while providing a robust blueprint for the practical implementation of Traveler's Hub, ensuring that the transition into the development phase will be smooth and well-informed.

Chapter 5

Product

5.1 Software Architecture

The software architecture of Traveler's Hub is designed to support a scalable, maintainable, and user-friendly web application. Our architecture utilizes a client-server model to effectively separate concerns, allowing for the independent development and testing of client-side and server-side logic.

5.1.1 Frontend

The user interface is built using React. Known for its efficient rendering and dynamic component updates, React helps us create an interactive and responsive user experience.

5.1.2 Backend

We use Flask, a lightweight Python web framework, to manage server-side operations. Flask is ideal for setting up RESTful APIs that handle requests efficiently and are easy to integrate with our front end.

5.1.3 Database

For data storage, MongoDB is employed due to its robustness, ability to handle complex queries, and vast data management, which are crucial for our trip planning functionalities.

5.1.4 Communication

Interaction between the front end and back end is facilitated through HTTP/HTTPS protocols, with JSON as the primary data exchange format, ensuring lightweight data flow and system compatibility.

5.1.5 Image Handling

Images like user avatars or trip photos are handled locally to maintain simplicity and control. Upon upload, images are processed directly by the Flask application, including resizing and formatting, before being stored locally on the server. This allows the system to run independently of external services and ensures that all data remains within

the controlled environment, suitable for both development and deployment on a single machine or local network.

5.2 Technology Stack

The technology stack for Traveler's Hub is selected to support comprehensive functionality while ensuring ease of use and development:

5.2.1 Frontend

React: JavaScript library for building interactive user interfaces, known for its component-based architecture and efficient rendering.

5.2.2 Backend

Flask: Manages backend logic and API endpoints, offering a straightforward way to handle web requests.
Python: The primary language for backend development, favored for its readability and extensive library support.

5.2.3 Database

MongoDB: Our database choice for reliable storage and complex data handling capabilities.

5.2.4 Development Tools

Git: For version control, supporting collaborative development and feature tracking.
Docker: For creating isolated environments, ensuring that the application can be deployed and run consistently across various local setups.

This setup ensures that Traveler's Hub can operate entirely within a local environment, supporting all required features without external dependencies, and is ideal for academic settings where simplicity and self-containment are key.

5.3 Use Case Diagrams

This section presents the use case diagram, which outlines user interaction and system functionalities.

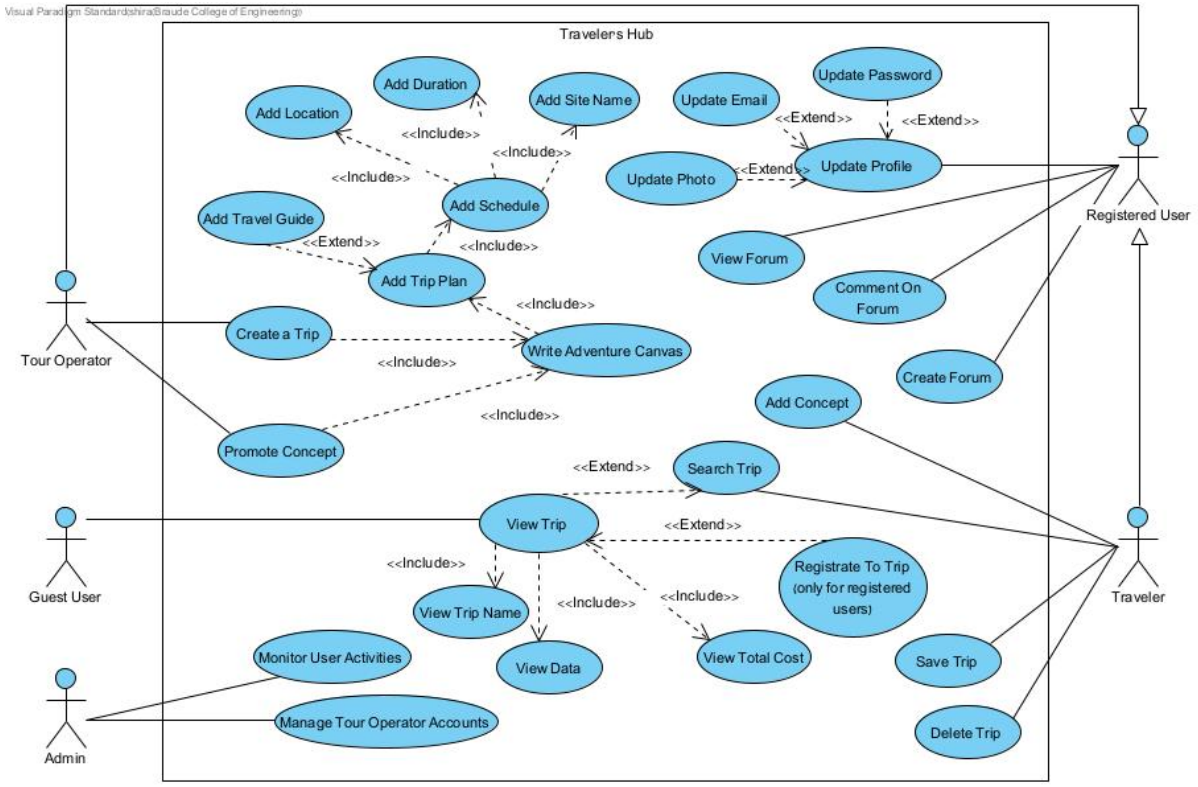

Figure 5.1: Use Case Diagram of Traveler's Hub

5.4 Class Diagram

This section presents the class diagram, which illustrates the system's structure, including classes, their attributes, methods, and object relationships.

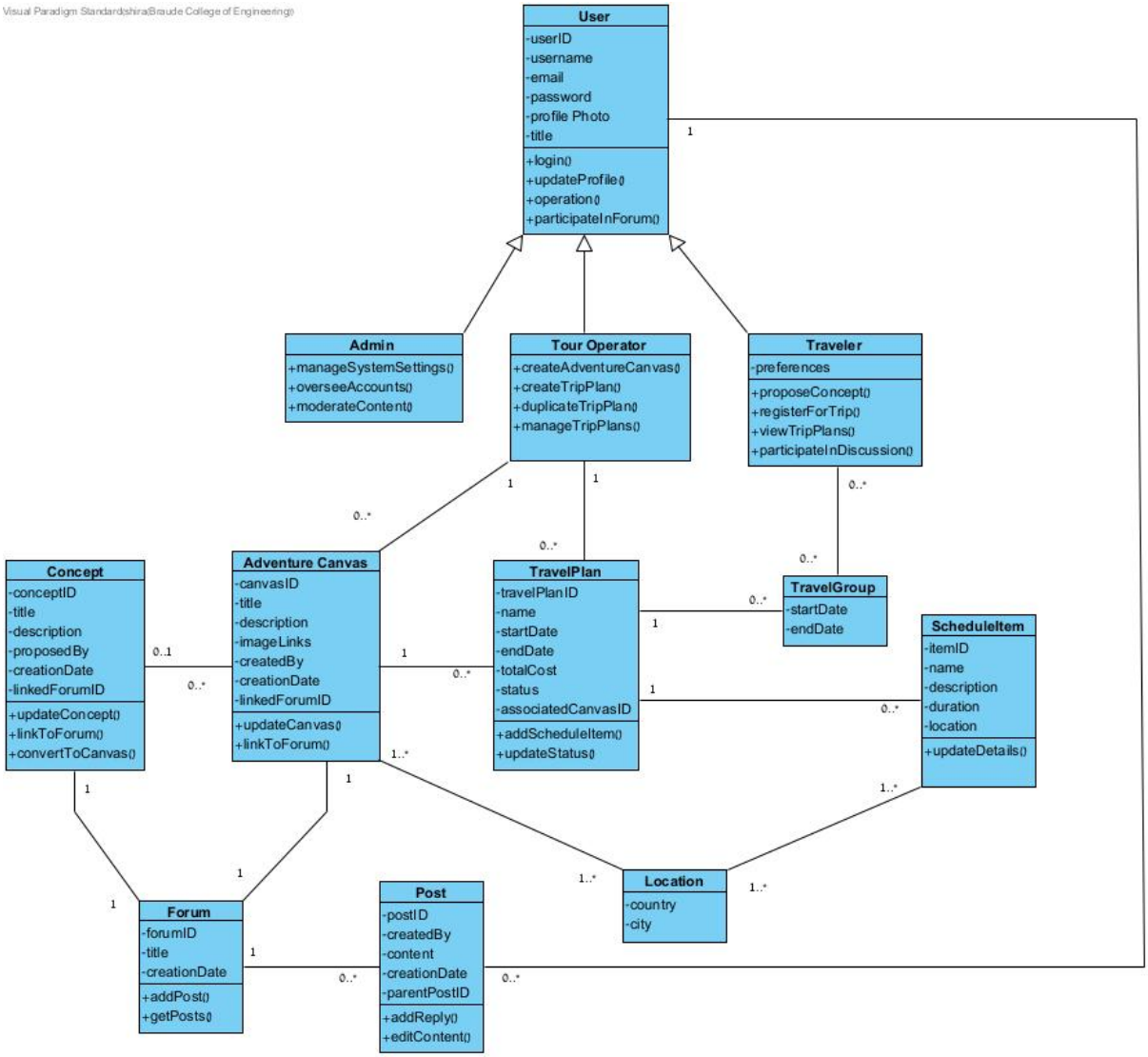

Figure 5.2: Class Diagram of Traveler's Hub

5.5 Activity Diagram

5.5.1 User SignUp

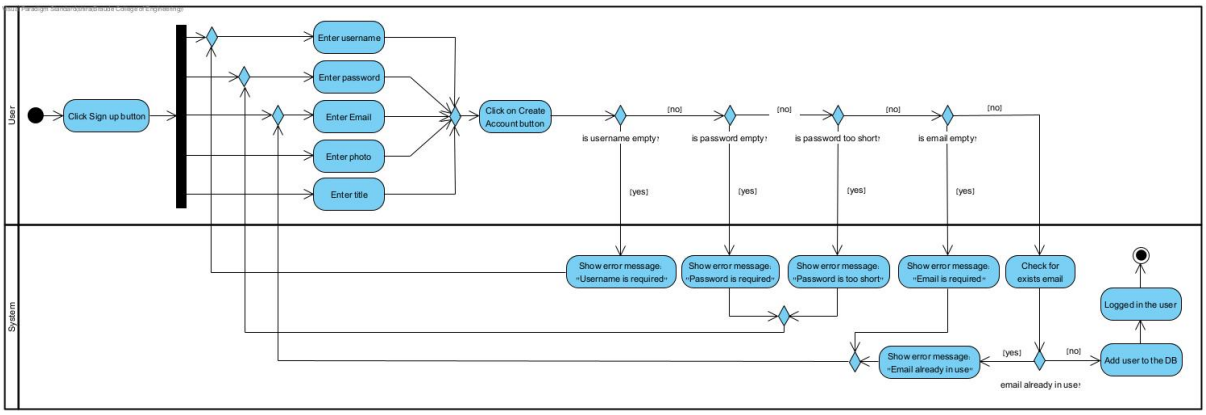

Figure 5.3: Activity Diagram for User Sign-Up

When a user decides to sign up for the system, they start by clicking the "Sign up" button. They are then prompted to fill in their username, password, email, photo, and title. Upon completing the form, the user clicks the "Create Account" button to submit their information.

The system then begins a series of checks to ensure that all entered information meets the necessary criteria: it first checks if the username has been entered. If not, an error message is displayed. If a username is provided, it checks if the password field is empty. An empty password field triggers an error message. If a password is entered, the system checks the length of the password and displays an error if it's too short.

Next, the system verifies if the email field is filled. If it's empty, an error message alerts the user. If an email is entered, the system checks to see if it is already used. If the email is unique, the user is successfully added to the database and logged into the system. If the email already exists, an error message informs the user.

This structured process ensures that all user information is valid and each user's account is unique, enhancing security and user experience.

5.6 Functional Requirements

Number	Category	Description	Priority (1-5)
1	General	Integration with MongoDB	1
2	General	Help section on website pages	4
3	Guest User	Sign up to the website	1
4	Registered User	Log in to the website	1
5	Registered User	Log out from the website	1
6	Registered User	Delete user account	3
7	Registered User	View user profile	2
8	Registered User	Update user profile details	3
9	Registered User	View forms	3
10	Registered User	Create forms	2
11	Registered User	Comment on forms	3
12	Traveler	Search for a trip	1
13	Traveler	View adventure canvas	1
14	Traveler	View trip plan	1
15	Traveler	Register to a trip	2
16	Traveler	Unregister from a trip	2
17	Traveler	Create concept	1
18	Tour Operator	Add adventure canvas	1
19	Tour Operator	Add trip plan	1
20	Tour Operator	Add travel guide	4
21	Tour Operator	Promote concept	1
22	Tour Operator	Open a trip for registration	1
23	Tour Operator	Close a trip for registration	1
24	Admin	Monitor user activities	4
25	Admin	Manage Tour Operator account	1

Table 5.1: Functional requirements for the system

5.7 Non-Functional Requirements

Number	Category	Description	Priority (1-5)
1	Usability	User-friendly interface to use	1
2	Usability	Clear navigation and intuitive interactions	2
3	Usability	Provide error messages and notifications	2
4	Usability	Design that adapts to different screen sizes and devices	3
5	Security	Implement user authentication and authorization	1
6	Security	Encrypt user passwords before storing them in the database	1
7	Compatibility	Compatible with multi web browsers and operating systems	4
8	Performance	Data synchronization with MongoDB	1
9	Performance	Low response times for user interactions	2
10	Maintainability	Design with modularity and code reusability	3
11	Maintainability	Comprehensive documentation of the code	3
12	Ethical	Respect user privacy and data protection rights	1
13	Privacy	Ability to delete accounts and personal data permanently upon request	2

Table 5.2: Non-functional requirements for the system

Ethical Considerations

Respecting user privacy and data protection rights is essential in maintaining trust and legal compliance. It ensures that personal data is only used for its intended purpose and is not shared with third parties without explicit consent. This ethical principle supports the notion that users should control their data, reinforcing their rights to privacy and protection.

5.8 Web Interface

5.8.1 Sign Up Interface

The signup page allows users to create a new account. It prompts users to enter their email address, username, and password. Additionally, users can provide a title and upload a photo for their profile.
For users who prefer not to create an account, there is an option to "Stay as a guest."

However, the tooltip (exclamation mark) explains that this guest mode limits users to only viewing trips and restricts access to other features such as booking, forum discussions, and concept creation.

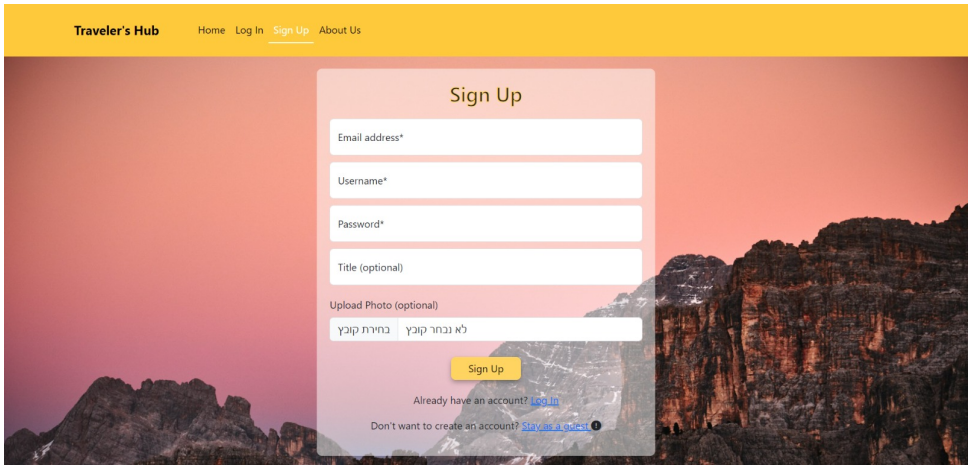

Figure 5.4: Screenshot of the Sign Up Interface

5.8.2 Forum Interface

Users can view and participate in discussions related to an organized trip. The forum includes messages from users and a tour guide, displayed chronologically.
Users can also see if a message is from a tour guide, indicated by a "Tour Guide" badge next to the username.

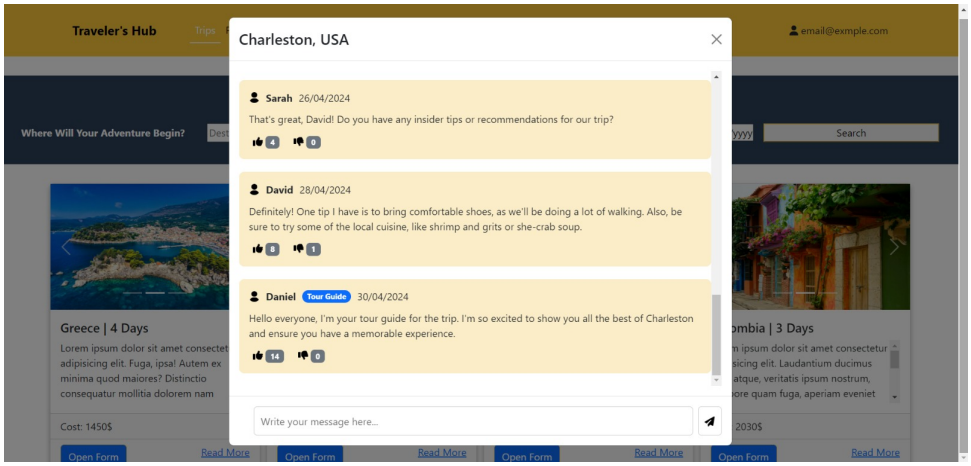

Figure 5.5: Screenshot of the Forum Interface

Chapter 6

Evaluation/Verification Plan

6.1 Unit Testing

Unit testing is essential for verifying that each component of Traveler's Hub operates as expected. These tests focus on individual functionalities, helping identify and resolve bugs early in development.

Component	Test Case	Expected Result
User Registration	Register a new user with valid details	Registration successful, user data stored in MongoDB
User Registration	Register a new user with invalid details	Registration failed, error message displayed
User Login	Login with valid details	Login successful, a user navigates to the main dashboard
User Login	Login with invalid details	Login failed, error message displayed
User Profile	Update user profile with valid details	Profile updates successful, changes reflected in the database
User Profile	Attempt to update the profile with invalid details	Update failed, the error message displayed
Trip Planning	Create a trip with valid details	Trip creation successful, trip details stored in the database
Trip Planning	Create a trip with incomplete details	Trip creation failed, error message displayed
Adventure Canvas	Add new adventure canvas with valid details	Canvas addition successful, details stored in the database
Adventure Canvas	Add new adventure canvas without title	Canvas addition failed, error message displayed
Concept Promotion	Promote a valid concept to Adventure Canvas	Promotion successful, concept listed as Adventure Canvas

Component	Test Case	Expected Result
Concept Promotion	Promote an invalid concept	Promotion failed, error message displayed
Forum Interaction	Post a new message in a forum with valid input	Message posted successfully, visible in the forum
Image Handling	Upload a valid image file for user avatar	Image upload successful, avatar updated in user profile
Image Handling	Upload an image with unsupported format	Upload failed, error message displayed

6.2 Integration Testing

Integration testing ensures that different parts of the application work together as expected.

Integration Test Scenarios for Traveler's Hub:

- **Trip Registration Flow:** Log in, navigate to the trip, and register for a trip. *Expected Result:* Successful registration, trip appears in user's itinerary.
- **Profile Update and View:** Update profile, view updated profile. *Expected Result:* Changes reflected correctly across the application.
- **Search and View Trip:** Perform a search, select a trip. *Expected Result:* Trip details displayed correctly.

6.3 System Testing

System testing evaluates the entire system's compliance with the specified requirements.

System Test Cases for Traveler's Hub:

- **Complete System Navigation:** Navigate through all main screens. *Expected Result:* Seamless navigation without errors, correct content display.
- **Cross-browser Compatibility Testing:** Open the application in different browsers. *Expected Result:* Consistent performance and appearance across browsers.
- **Mobile Responsiveness Testing:** Access the site on various mobile devices. *Expected Result:* UI adjusts appropriately for different screen sizes.

6.4 Manual Testing

Manual testing involves user simulations to ensure the user experience is as designed and intuitive.

Manual Test Scenarios for Traveler's Hub:

- **Login and Logout Process:** Log in to the application, then log out. *Expected Result:* Successful login followed by successful logout.
- **Trip Plan Creation and Cancellation:** Create a trip plan, then cancel it. *Expected Result:* Trip plan created and removed without residual data.
- **Access Help Section:** Navigate to and engage with the help section. *Expected Result:* Help content is accessible and informative.

References

[1] Gursoy, D., & Umbreit, W. T. (2004). Tourist information search behavior: Cross-cultural comparison of European Union member states. *International Journal of Hospitality Management*, 23(1), 55-70.

[2] Tanti, A., & Buhalis, D. (2017). The influences and consequences of being digitally connected and/or disconnected from travelers. *Information Technology & Tourism*, 17(1), 121-141.

[3] Wang, D., Xiang, Z., & Fesenmaier, D. R. (2016). Smartphone use in everyday life and travel. *Journal of travel research*, 55(1), 52-63.

[4] Xiang, Z., & Gretzel, U. (2010). Role of social media in online travel information search. *Tourism management*, 31(2), 179-188.

[5] OpenAI GPT-3.5: Used for literature review, the prompt 'write me a two-page literature review on travel trip planning. Remember to include refrerences'.

[6] Claude.AI: Used for literature review, the prompt 'write me a two-page literature review on travel trip planning. Remember to include refrerences'