Capstone Project Phase B

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)

September 17, 2024

Table of Contents

Abstract **3**

1 Introduction **4**
1.1 Problem Statement . . . 4
1.2 Existing Solutions . . . 4
1.3 Proposed Solution . . . 4
1.4 Stakeholders . . . 5

2 Background and Related Work **6**
2.1 Tools and Methods for Trip Planning Applications . . . 6
2.1.1 Existing tools . . . 6

3 Expected Achievements **9**
3.1 Objectives . . . 9
3.2 Success Criteria . . . 9
3.3 Challenges . . . 10
3.3.1 Image Storage . . . 10
3.3.2 Live Chat Integration . . . 10
3.3.3 Secure Password Storage . . . 11

4 Engineering Process **12**
4.1 Process . . . 12
4.2 product . . . 12
4.3 Use Case Diagrams . . . 12
4.4 Class Diagram . . . 13
4.5 Package Diagram . . . 14
4.6 Activity Diagram . . . 16
4.6.1 Write a Concept . . . 16
4.6.2 Promote a Concept . . . 17
4.6.3 Publish a Trip . . . 18
4.6.4 Write a Review . . . 18
4.7 MongoDB . . . 19
4.7.1 Database Schema . . . 19
4.8 Evaluation / Verification Plan . . . 21
4.8.1 Unit Testing . . . 21
4.9 Manual Testing . . . 25
4.9.1 Test Scenarios and Results . . . 25
4.9.2 Scenario 2: Trip Plan Creation and Cancellation . . . 25

4.9.3 Scenario 3: Create and Promote a Concept . . . 26
4.9.4 Scenario 4: Forum Discussion Engagement . . . 26
4.9.5 Scenario 5: Write a Review for a Completed Trip . . . 27

5 User Manual **28**
5.1 Logo . . . 28
5.2 Home page . . . 28
5.3 Login page . . . 28
5.4 Sign up page . . . 29
5.5 About Us page . . . 29
5.6 Navbar links according to the user type . . . 30
5.6.1 Traveler . . . 30
5.6.2 Tour operator . . . 30
5.6.3 Admin . . . 30
5.7 Trips page (home user page) . . . 30
5.8 Forum pop-up . . . 30
5.9 Trip Plan page . . . 31
5.9.1 Dates section . . . 31
5.9.2 Trip registration . . . 33
5.9.3 Successfully purchase . . . 33
5.10 My Booking page . . . 33
5.10.1 Write a Review . . . 34
5.10.2 Cancel Booking . . . 34
5.11 My Profile page . . . 35
5.11.1 Delete Account . . . 35
5.12 Concept page . . . 36
5.12.1 Promote a Concept . . . 36
5.13 Create a Trip . . . 37
5.14 My Trips page . . . 37
5.14.1 Review . . . 38
5.15 Manage User page . . . 38
5.16 Customer Review page . . . 39
5.17 Manage Trips page . . . 39
5.17.1 Trips that wait to be published . . . 39
5.17.2 Trips that have dates waiting for confirmation . . . 40

6 Maintenance Guide **41**
6.1 Software Architecture Diagram . . . 41
6.2 Architecture Overview . . . 41
6.3 Components of the Architecture . . . 42
6.3.1 React Client . . . 42
6.3.2 Node.js & Express Backend . . . 42
6.3.3 MongoDB Database . . . 42
6.3.4 Socket.IO for Real-Time Communication . . . 42
6.3.5 Imgur Integration . . . 42

7 GIT **43**

References **44**

Abstract

Traveler's Hub is a comprehensive trip planning platform designed to simplify the process for both travelers and tour operators. The platform's core functionality revolves around providing seamless trip planning tools, enabling users to search for, explore, and book trips. Tour operators can create and manage Adventure Canvases—general travel packages outlining potential activities, destinations, and costs. These canvases serve as the foundation for Trip Plans, which are detailed trip implementations with specific dates, itineraries, and pricing, available for booking.

A unique feature of Traveler's Hub is the concept creation tool, where regular users can propose ideas for trips. These concepts can be reviewed and promoted by tour operators into full Adventure Canvases, thus fostering a community-driven aspect of trip planning. This encourages travelers to participate in the creative process, contributing to a diverse range of trip options.

Traveler's Hub also includes forum discussions where travelers and operators can engage in conversations, share experiences, and collaborate on trip-related topics. This feature enhances the sense of community and offers real-time feedback opportunities.

The platform is built using React for the frontend, Node.js with Express for the backend, and MongoDB for database management. It integrates Imgur for efficient image handling and ensures secure password storage. Traveler's Hub offers a holistic and flexible solution to meet the needs of modern travelers and tour operators alike.

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

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.

Advantages:

- It allows offline access, so users can check their plans without an internet connection.
- The interface is simple and intuitive, making it easy to use even for non-tech-savvy users.

Disadvantages:

- Many of its advanced features, such as flight alerts and real-time updates, are only available with a paid subscription.
- Users need to rely on email forwarding, which can feel outdated compared to more modern methods of syncing information.
- The app doesn't offer much flexibility for customizing your itinerary.

Google Travel

Google Travel 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.

Advantages:

- Excellent integration with Google services like Gmail and Google Maps makes it easy to organize trip details automatically.
- Provides highly personalized recommendations based on your preferences and past behavior.
- Offline access to trip plans is available, making it convenient when traveling without a reliable internet connection.

Disadvantages:

- It lacks strong collaboration features, meaning you can't easily share your itinerary with others or plan a trip with friends.
- Since it's heavily dependent on the Google ecosystem, it works best if you already use other Google products, which may not suit everyone.

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.

Advantages:

- It has a very comprehensive database of attractions, activities, and accommodations, giving users plenty of options to choose from.
- Collaboration features allow users to share itineraries with friends and plan trips together, making it great for group travel.
- The platform helps users estimate trip costs, which is useful for budgeting your vacation.

Disadvantages:

- The interface can be a bit overwhelming for first-time users, as there are so many options and features.
- Sometimes the platform recommends activities that aren't relevant or nearby, making the suggestions less personalized.

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.

Advantages:

- Perfect for road trips, as it helps users discover hidden gems and points of interest along their driving route.

- Offers detailed information about attractions, restaurants, and hotels, allowing users to plan a comprehensive road trip.
- The platform allows you to customize your route, making it flexible for spontaneous changes during the journey.

Disadvantages:

- The service is primarily focused on the U.S. and Canada, which limits its usefulness for international road trips.
- Some of the more advanced route-planning features are locked behind a paid subscription, which may not be ideal for casual users.

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.
6. **Concept Creation:** Allow travelers to propose new travel ideas by submitting trip concepts, which can be reviewed and promoted by tour operators.
7. **Concept Promotion:** Enable tour operators to review and promote user-submitted concepts into full trips, making them available for booking.

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, Forum Discussions, and Concept Creation and Promotion, 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.

5. **Concept Workflow:** Successfully implement the concept workflow, allowing travelers to submit trip ideas and tour operators to promote these concepts into bookable trips.

3.3 Challenges

3.3.1 Image Storage

Challenge:

One challenge we faced was determining where to store images, such as user profile pictures and trip images. Initially, we saved images directly in our database by converting them to base64 strings. The main problem with that is that when we store images as long strings it led to slow upload and retrieval times, and in some cases, fetching multiple images would result in errors.

Solution:

To overcome this challenge, we integrated with Imgur for image storage. We created an account on Imgur and stored the images there. In our database, we saved only the URLs pointing to the images on Imgur. We used a POST request with the access token provided by Imgur to upload the images. This solution significantly improved the speed of saving and displaying images and eliminated the errors we encountered when handling large numbers of images.

3.3.2 Live Chat Integration

Challenge:

We wanted to enable users to communicate with each other in real-time, so we built a forum for each trip. The challenge was to update the chat in real time, ensuring that connected users wouldn't need to refresh the page to see new comments or updates.

Solution:

To address this, we used the Socket.io framework. Socket.io allows the server to push updates to the client as soon as they occur, ensuring that the forum is updated in real-time whenever there is a new comment, or when users interact with elements like the like/dislike button. This approach provided a seamless and interactive experience for users participating in the forum.

3.3.3 Secure Password Storage

Challenge:

We couldn't save the user password as it is due to privacy and security concerns.

Solution:

We implemented a secure password storage mechanism using a hash function combined with a random salt. When a user creates or updates their password, a random salt is generated and combined with the password before being hashed using a slow hash function (bcrypt). This ensures the security of users' passwords.

Chapter 4

Engineering Process

4.1 Process

In Phase B of the Traveler's Hub project, we moved from design to development. We implemented the platform using the MERN stack (MongoDB, Express, React, Node.js), which provided a unified technology base across the entire application, ensuring better performance and easier maintenance.

The frontend was built with React, focusing on creating a responsive, user-friendly interface. Core features such as the *Adventure Canvas* and *Trip Plan* were developed to allow seamless trip management and booking. Additionally, the *Concept Creation* tool enabled users to propose their own trip ideas, which tour operators can review and promote into full trips.

On the backend, Node.js and Express were used to handle API requests and manage data flow. The decision to use MongoDB allowed for a flexible database schema, which was essential for managing dynamic trip data and user interactions. The backend also facilitated secure user authentication using bcrypt for password hashing, along with JWT-based session management.

Real-time communication was a key component of the implementation, achieved through the integration of Socket.IO. This allowed travelers and tour operators to engage in real-time conversations within the platform's forum, ensuring a smooth and interactive experience without requiring page reloads.

4.2 product

4.3 Use Case Diagrams

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

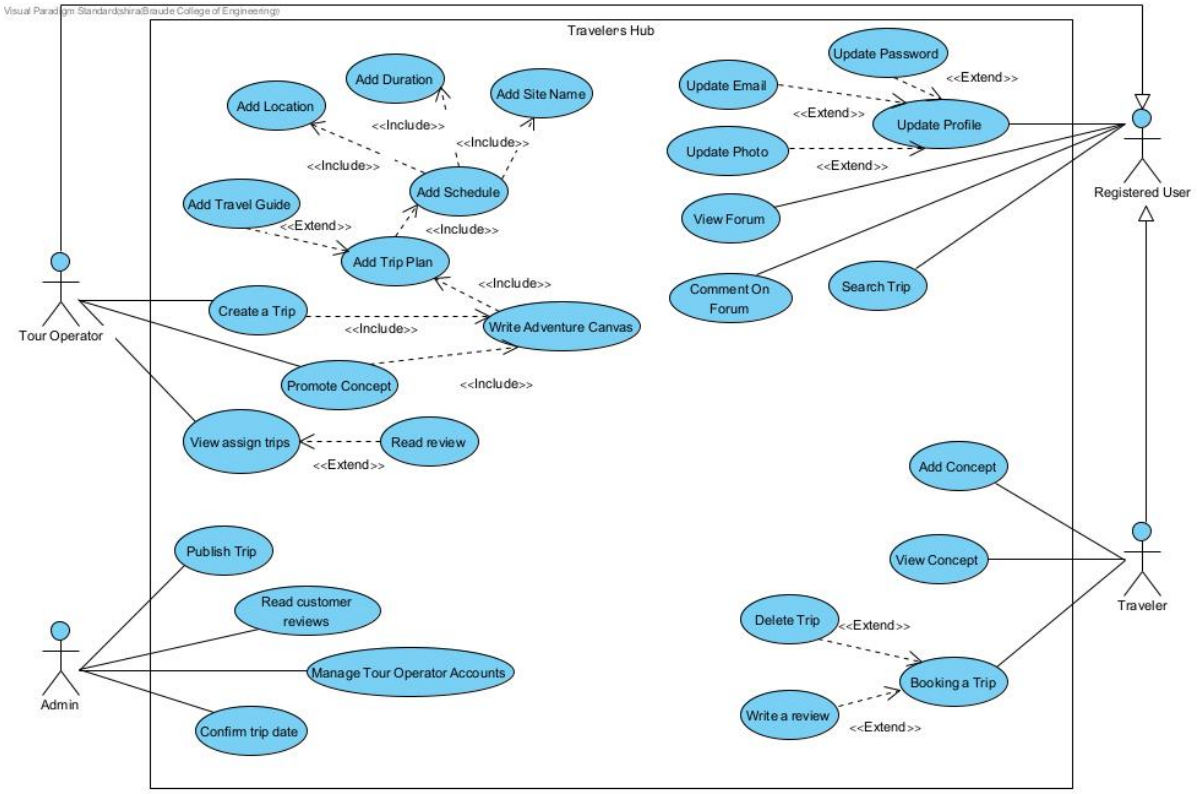


Figure 4.1: Use Case Diagram of Traveler's Hub

4.4 Class Diagram

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

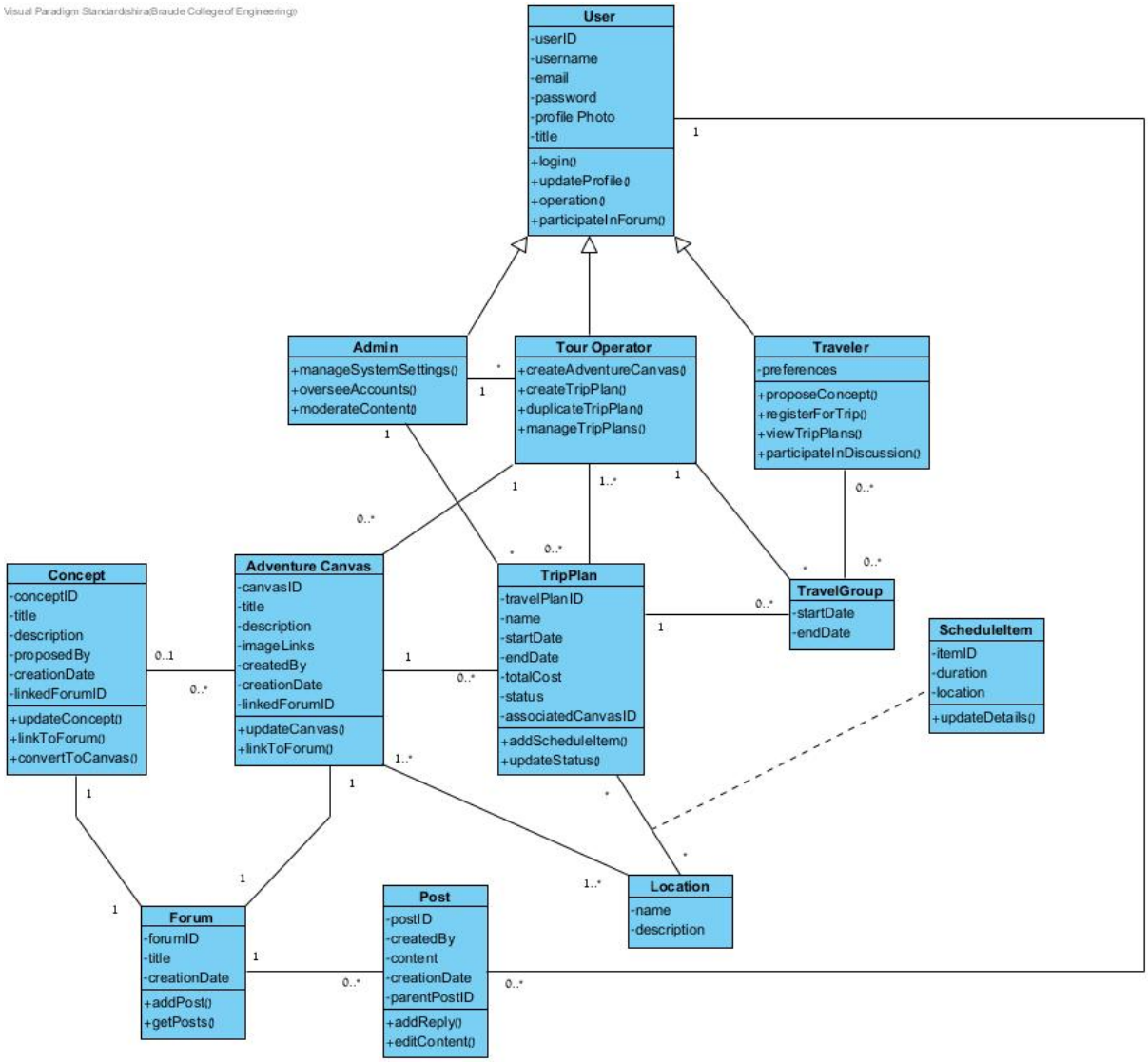


Figure 4.2: Class Diagram of Traveler's Hub

4.5 Package Diagram

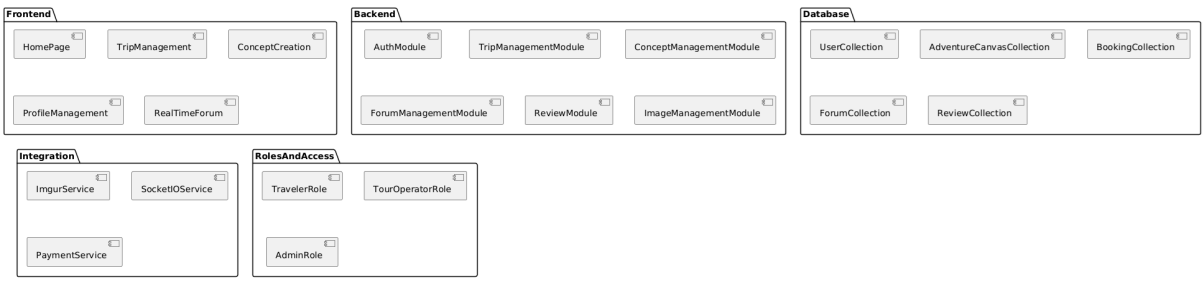


Figure 4.3: Package Diagram of Traveler's Hub

The package diagram organizes the Traveler's Hub system into five main parts. Each package contains related components, making the system easier to manage.

- **Frontend**
  - *HomePage*: Main page with navigation.
  - *TripManagement*: Search and book trips.
  - *ConceptCreation*: Travelers propose new trip ideas.
  - *ProfileManagement*: Manage user info and trip history.
  - *RealTimeForum*: Real-time chat for users.
- **Backend**
  - *AuthModule*: Handles login, sign-up, and security (JWT).
  - *TripManagementModule*: Manages trip creation and bookings.
  - *ConceptManagementModule*: Handles trip ideas and promotions.
  - *ForumManagementModule*: Manages chat features.
  - *ReviewModule*: Manages user reviews.
  - *ImageManagementModule*: Integrates with Imgur for images.
- **Database**
  - *UserCollection*: Stores user data.
  - *AdventureCanvasCollection*: Stores trip details.
  - *BookingCollection*: Tracks trip bookings.
  - *ForumCollection*: Stores chat messages.
  - *ReviewCollection*: Stores user reviews.
- **Integration**
  - *ImgurService*: Image hosting.
  - *SocketIOService*: Real-time communication.
  - *PaymentService*: Future payment handling.
- **Roles and Access**
  - *TravelerRole*: Users who book trips and write reviews.
  - *TourOperatorRole*: Manage trips and promote concepts.
  - *AdminRole*: Full system control.

4.6 Activity Diagram

4.6.1 Write a Concept

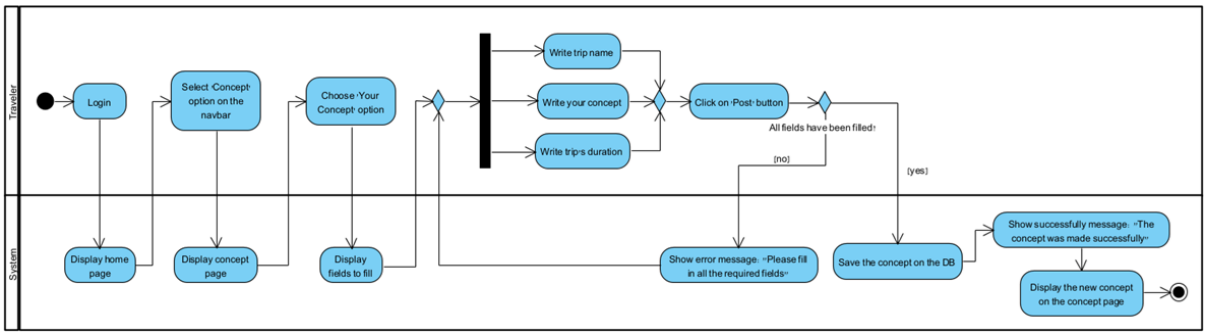


Figure 4.4: Activity Diagram for Write a Concept

1. Traveler logs into the system.
2. The system displays the home page.
3. The user navigates to the 'Concept' page.
4. The system displays the concept page.
5. The user clicks on the 'Your Concept' option.
6. The system displays fields to fill.
7. The user enters the trip name, their concept for the trip, and the duration.
8. The user clicks on the 'Post' button.
9. If the user doesn't fill all the fields, the system will show an error message.
10. Otherwise, the system saves the new concept in the database.
11. The system shows a successful message.
12. The system displays the new concept on the page.

4.6.2 Promote a Concept

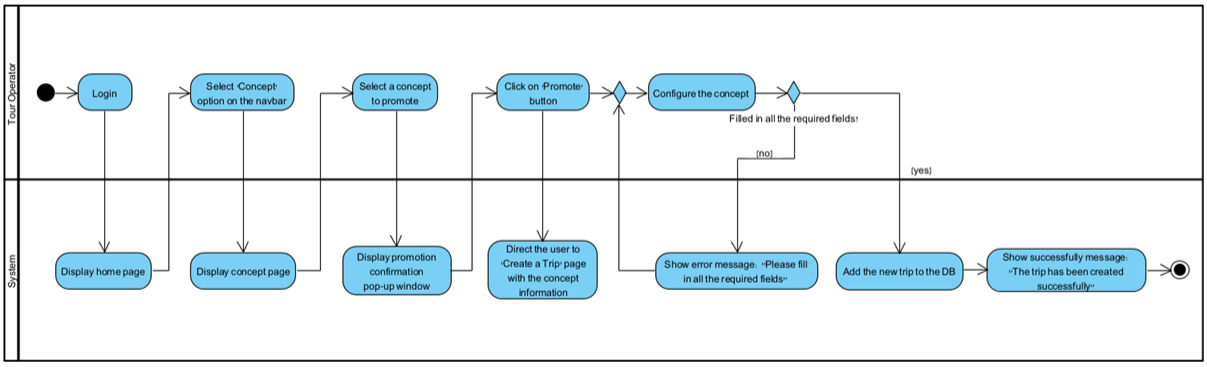


Figure 4.5: Activity Diagram for User Sign-Up

1. Tour operator logs into the system.
2. The system displays the home page.
3. The user navigates to the 'Concept' page.
4. The system displays the concept page.
5. The user selects a concept to promote.
6. The system displays a promotion confirmation pop-up window.
7. The user clicks on the 'Promote' button.
8. The system directs the user to the 'Create a Trip' page with the concept information.
9. The user enters all related fields for the concept to complete the trip plan.
10. If the user doesn't enter all the required fields, the system shows an error message.
11. Otherwise, the system adds the new trip to the database.
12. The system shows a successful message.

4.6.3 Publish a Trip

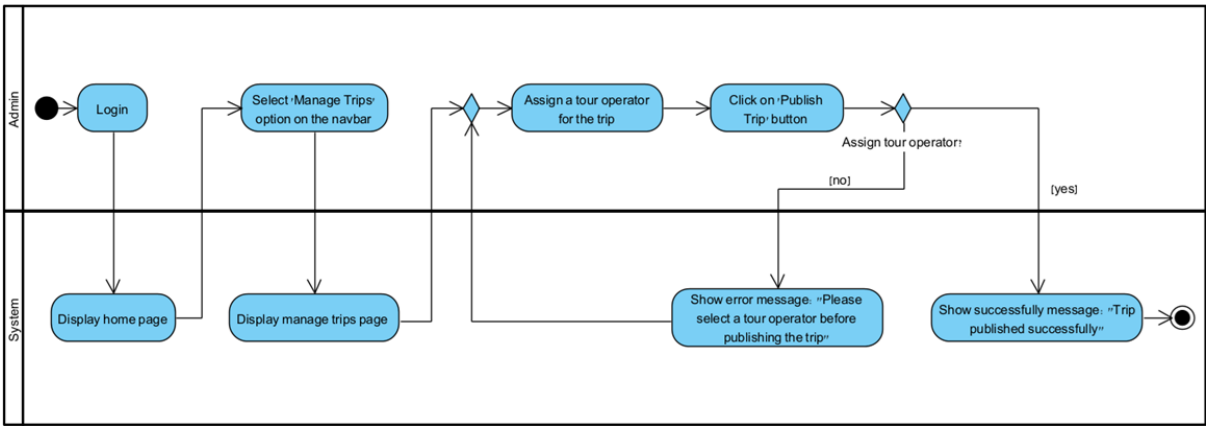


Figure 4.6: Activity Diagram for Publish a Trip

1. The admin logs into the system.
2. The system displays the home page.
3. The user navigates to the 'Manage Trips' page.
4. The system displays the manage trips page.
5. The system displays a list of all the trips that are waiting to be published.
6. The user chooses a trip and assigns a tour operator for that trip.
7. The user clicks on the 'Publish Trip' button.
8. If the user doesn't assign any tour operator, the system will show an error message.
9. Otherwise, the system shows a successful message.

4.6.4 Write a Review

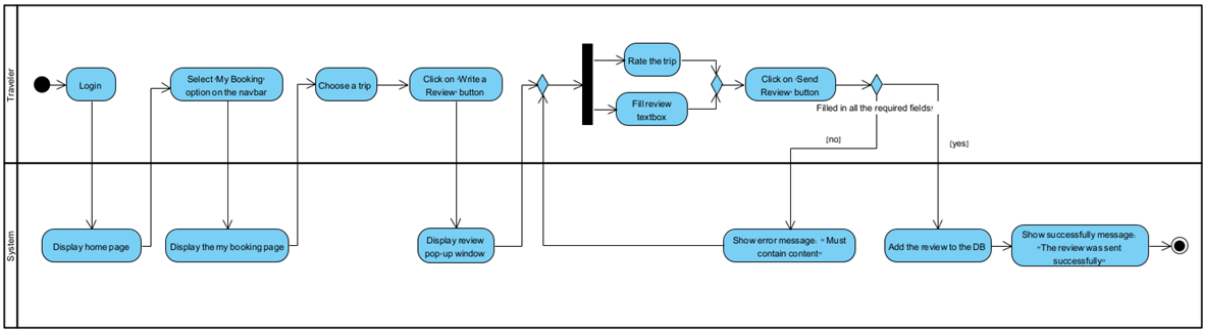


Figure 4.7: Activity Diagram for Write a Review

1. Traveler logs into the system.
2. The system displays the home page.
3. The user navigates to the 'My Booking' page.
4. The system displays the My Booking page.
5. The user chooses a trip.
6. The user clicks on the 'Write a Review' button.
7. The system displays a review pop-up window.
8. The user rates the trip from 0 to 5.
9. The user writes the review for the trip.
10. The user clicks on the 'Send Review' button.
11. If the user doesn't enter all the required fields, the system will show an error message.
12. Otherwise, the system adds the review to the database.
13. The system shows a successful message.

4.7 MongoDB

MongoDB's flexibility, scalability, and seamless integration with JavaScript make it a great fit for building websites with Node.js and React. It lets developers work within a unified technology stack, easily adapt to changing data models. Specifically for our website, MongoDB's real-time data handling capabilities and ability to manage dynamic user profiles, support complex trip organization, and enable live chat make it the ideal database for building a responsive, scalable, and feature-rich web application.

4.7.1 Database Schema

Every object in our database contains:

Field Name	**Type**
_id	ObjectId
createdAt	Date
updateAt	Date

Table 4.1: General fields for all objects in the database.

Collections details:

* *duration* is in days.

** *cost* is in US dollars (USD).

Collection Name	Field Name	Type
adventurecanvas	images	Array[String]
	tripName	String
	duration	Number
	description	String
	cost	Number
	isPublished	Boolean
	assignTourOperator	String

Table 4.2: Schema for the adventurecanvas collection.

Collection Name	Field Name	Type
booking	tripName	String
	duration	Number
	startDate	String
	endDate	String
	member	Number
	totalCost	Number
	payment	paymentSchema
	adventureCanvas_id	ObjectId
	user_id	ObjectId
	date_id	ObjectId

Table 4.3: Schema for the booking collection.

Field Name	Type
method	String
cardHolder	String
cardNumber	Number
expirationDate	String
cvv	Number

Table 4.4: paymentSchema type

Collection Name	Field Name	Type
concept	tripName	String
	description	String
	duration	Number
	likes	Array[String]

Table 4.5: Schema for the concept collection.

Collection Name	Field Name	Type
forum	comments	Array[commentSchema]
	adventureCanvas_id	ObjectId

Table 4.6: Schema for the forum collection.

Field Name	Type
email	String
message	String
likes	Array[String]
dislikes	Array[String]

Table 4.7: commentSchema type.

Collection Name	Field Name	Type
review	travelerEmail	String
	rating	Number
	content	String
	tourOperator_email	String
	adventureCanvas_id	ObjectId

Table 4.8: Schema for the review collection.

Collection Name	Field Name	Type
tripPlan	dates	Array[dateSchema]
	scheduled	Array[scheduledItemSchema]
	travelGuide	String
	adventureCanvas_id	ObjectId

Table 4.9: Schema for the tripPlan collection.

Field Name	Type
starting	String
ending	String
currBooking	Number
limitBooking	Number
executionConfirmed	Boolean

Table 4.10: dateSchema type.

Field Name	Type
location	String
day	Number
description	String

Table 4.11: scheduledItemSchema type.

Collection Name	Field Name	Type
user	username	String
	password	String
	email	String
	photo	String
	userType	String

Table 4.12: Schema for the user collection.

4.8 Evaluation / Verification Plan

4.8.1 Unit Testing

Component	Test Case	Expected Result
Adventure Canvas Controller	Get all published adventure canvases	Return all published adventure canvases with tour operator details
Adventure Canvas Controller	Get all published adventure canvases (none found)	Return 404 if no adventure canvases are found
Adventure Canvas Controller	Create a new trip	Create a new trip with adventure canvas, trip plan, and forum
Adventure Canvas Controller	Create a new trip (no file provided)	Return 400 if no file is provided
Adventure Canvas Controller	Get trip plan by adventure canvas ID	Return a trip plan for a given adventure canvas ID
Adventure Canvas Controller	Get trip plan by adventure canvas ID (not found)	Return 404 if trip plan is not found
Adventure Canvas Controller	Get adventure canvas by ID	Return an adventure canvas with tour operator details
Adventure Canvas Controller	Get adventure canvas by ID (not found)	Return 404 if adventure canvas is not found
Adventure Canvas Controller	Get tour operator trips	Return all trips for a tour operator
Adventure Canvas Controller	Get tour operator trips (none found)	Return 404 if no trips are found for the tour operator
Booking Controller	Get bookings by user ID	Return bookings for a user
Booking Controller	Save booking details	Save a new booking and update TripPlan
Booking Controller	Save booking details (TripPlan not found)	Return 404 if TripPlan is not found
Booking Controller	Save booking details (Date not found)	Return 404 if date is not found in TripPlan
Booking Controller	Delete booking	Delete a booking and update TripPlan
Booking Controller	Delete booking (TripPlan not found)	Return 404 if TripPlan is not found
Booking Controller	Delete booking (Date not found)	Return 404 if date is not found in TripPlan
Booking Controller	Delete booking (Booking not found)	Return 404 if Booking is not found

Component	Test Case	Expected Result
Concept Controller	Get concepts	Return all concepts sorted by updatedAt in descending order
Concept Controller	Post concept	Create a new concept
Concept Controller	Post concept (creation fails)	Return 404 if concept creation fails
Concept Controller	Update like to concept (add like)	Add a like to a concept
Concept Controller	Update like to concept (remove like)	Remove a like from a concept
Concept Controller	Delete concept	Delete a concept
Concept Controller	Delete concept (not found)	Return 404 if concept is not found
Forum Controller	Get forum content	Return forum content with commenter details
Forum Controller	Get forum content (not found)	Return 404 if forum is not found
Forum Controller	Add message	Add a new message to the forum
Forum Controller	Update like/dislike (like)	Update likes for a comment
Forum Controller	Update like/dislike (dislike)	Update dislikes for a comment
Manage Trips Controller	Get unpublished adventure canvases	Return unpublished adventure canvases
Manage Trips Controller	Get unpublished adventure canvases (none found)	Return 404 if no unpublished canvases are found
Manage Trips Controller	Publish trip	Publish a trip
Manage Trips Controller	Publish trip (canvas not found)	Return 400 if adventure canvas is not found
Manage Trips Controller	Get execution unconfirmed trip plans	Return unconfirmed trip plans
Manage Trips Controller	Get execution unconfirmed trip plans (none found)	Return 404 if no unconfirmed trip plans are found
Manage Trips Controller	Update confirmed date	Update execution confirmed status of a date
Manage Trips Controller	Update confirmed date (trip plan not found)	Return 404 if trip plan is not found
Manage Trips Controller	Get all tour operators	Return all tour operators

Component	Test Case	Expected Result
Manage Trips Controller	Get all tour operators (none found)	Return 404 if no tour operators are found
Manage Trips Controller	Delete trip	Delete a trip and associated data
Manage Trips Controller	Delete trip (associated data not found)	Return 404 if any associated data is not found
Profile Controller	Update username	Update the username successfully
Profile Controller	Update username (user not found)	Return 400 if user is not found
Profile Controller	Update password	Update the password successfully
Profile Controller	Update password (user not found)	Return 400 if user is not found
Profile Controller	Update photo	Update the profile photo successfully
Profile Controller	Update photo (no file provided)	Return 400 if no file is provided
Profile Controller	Delete photo	Delete the profile photo successfully
Profile Controller	Delete photo (user not found)	Return 404 if user is not found
Profile Controller	Delete user account	Delete the user account and associated bookings successfully
Profile Controller	Delete user account (user not found)	Return 404 if user is not found
Review Controller	Add review	Add a review successfully
Review Controller	Add review (tour operator not found)	Return 404 if tour operator is not found
Review Controller	Get review by adventure canvas ID	Return reviews with user details
Review Controller	Get review by adventure canvas ID (no reviews)	Return 404 if no reviews are found
Review Controller	Get review by tour operator email	Return reviews with user details
Review Controller	Get review by tour operator email (no reviews)	Return 404 if no reviews are found
User Authentication Controller	Login user	Login a user successfully
User Authentication Controller	Login user (invalid credentials)	Return 400 if login fails

Component	Test Case	Expected Result
User Authentication Controller	Signup user (traveler)	Signup a traveler successfully
User Authentication Controller	Signup user (tour operator)	Signup a tour operator successfully
User Authentication Controller	Signup user (existing email)	Return 400 if signup fails due to existing email

4.9 Manual Testing

Manual testing involved simulating the user experience to ensure the application performed as expected. The focus was on verifying core functionality, user flows, and overall intuitiveness of the Traveler's Hub platform.

4.9.1 Test Scenarios and Results

Scenario 1: Login and Logout Process

Test Description: The user logs into the system using valid credentials and then logs out of the application.

Steps:

1. Open Traveler's Hub in a web browser.
2. Enter valid email and password on the login page.
3. Click the "Login" button.
4. After logging in, click the "Logout" button.

Expected Result: The user is successfully logged in and redirected to the home page. Upon logging out, the user is redirected to the login page.

Actual Result: The login and logout process worked as expected. No issues were encountered.

4.9.2 Scenario 2: Trip Plan Creation and Cancellation

Test Description: The user creates a new trip plan, and then the admin publishes that trip.

Steps:

1. Log into the system as a tour operator.
2. Navigate to the "Create a Trip" page.
3. Fill in trip details such as trip name, duration, description, and cost.
4. Add scheduled items and dates.

5. Click the "Create" button to submit the trip.
6. Logout the system.
7. Log into the system as an admin.
8. Navigate to the "Manage Trips" page. and click the "Cancel" button for the newly created trip.
9. Assign tour operator for the trip.
10. Click on "Publish" button.

Expected Result: The trip is successfully created and published and can be seen in the "Trip" page.

Actual Result: The trip was successfully created and published. No issues were found in the process.

4.9.3 Scenario 3: Create and Promote a Concept

Test Description: A traveler creates a new concept for a trip, and a tour operator promotes the concept to a full trip.

Steps:

1. Log into the system as a traveler.
2. Navigate to the "Concept" page.
3. Click the "Create Concept" button and enter the required information (trip name, description, duration).
4. Submit the concept.
5. Log out and log in as a tour operator.
6. Navigate to the "Concept" page, find the newly created concept, and click the "Promote" button.
7. Complete the trip details and publish it as a full trip.

Expected Result: The concept is created successfully by the traveler. The tour operator can view the concept, promote it, and publish it as a trip.

Actual Result: The concept creation and promotion process worked seamlessly. No issues were found.

4.9.4 Scenario 4: Forum Discussion Engagement

Test Description: The user participates in a forum discussion related to a trip, posting messages and interacting with other users.

Steps:

1. Log into the system as a traveler.
2. Navigate to the "Trips" page and select a trip.

3. Enter the forum for the selected trip.

4. Post a new message and engage with other users by liking/disliking messages.

Expected Result: The user is able to post messages in the forum, and real-time updates are visible without refreshing the page. The like/dislike functionality works as expected.

Actual Result: The forum functionality was fully operational, with real-time updates visible and no errors encountered in message posting or interaction.

4.9.5 Scenario 5: Write a Review for a Completed Trip

Test Description: After completing a trip, the user submits a review for the trip.

Steps:

1. Log into the system as a traveler.

2. Navigate to the "My Bookings" page.

3. Select a completed trip and click the "Write a Review" button.

4. Enter a review, rate the trip, and click "Submit."

Expected Result: The review is successfully saved in the database, and the user receives a confirmation message.

Actual Result: The review was successfully submitted, and a confirmation message was displayed.

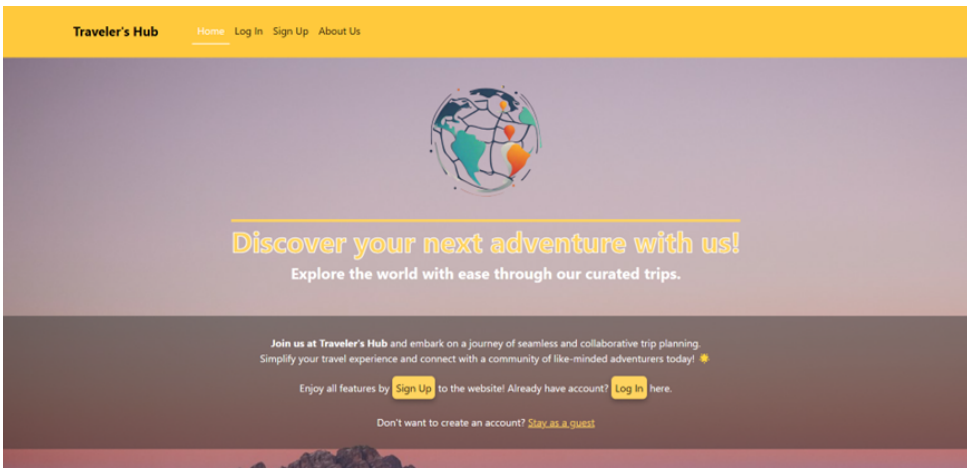
Chapter 5

User Manual

5.1 Logo

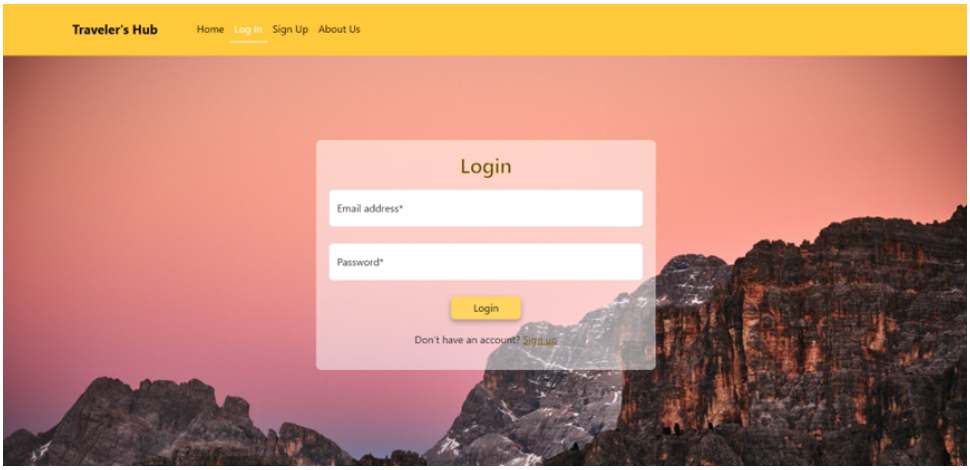
5.2 Home page

In the home page, users can get information about the website and see some of the features that the website offers.



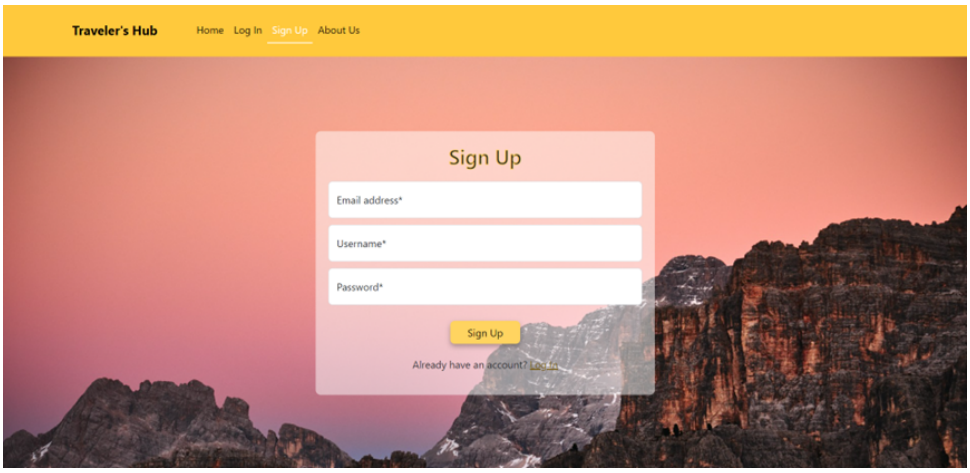
5.3 Login page

Users that already have an account can log in to the website by entering their email address and password.



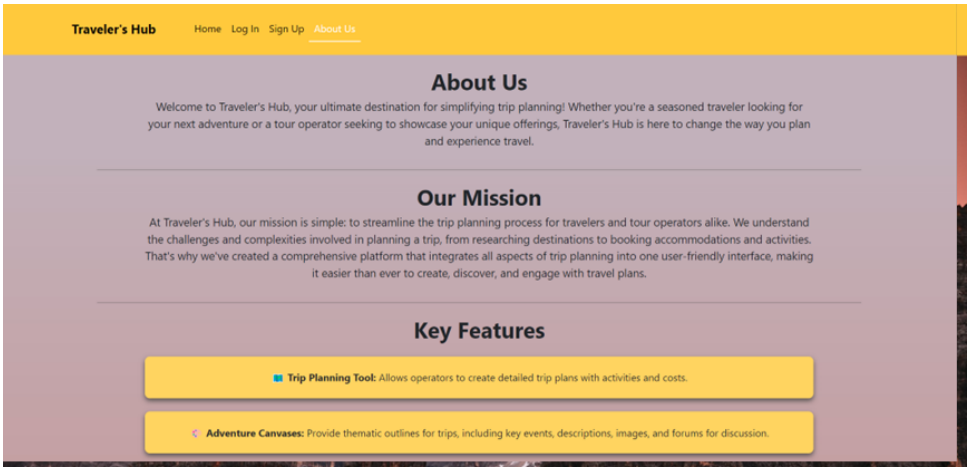
5.4 Sign up page

Users that don't have an account, can create one, by entering email address, username and password



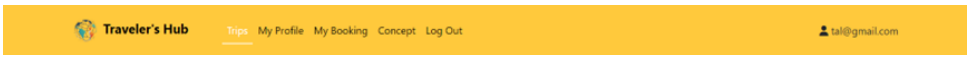
5.5 About Us page

Show information about the goal of the website and present more features.

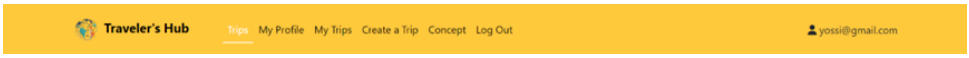


5.6 Navbar links according to the user type

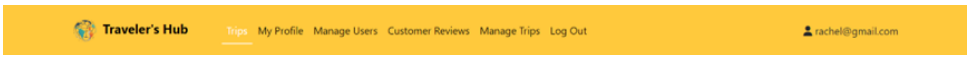
5.6.1 Traveler



5.6.2 Tour operator

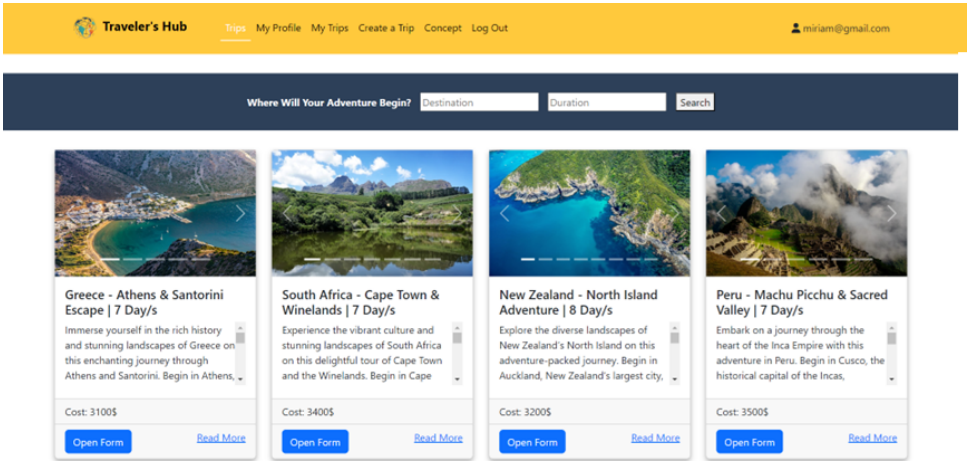


5.6.3 Admin



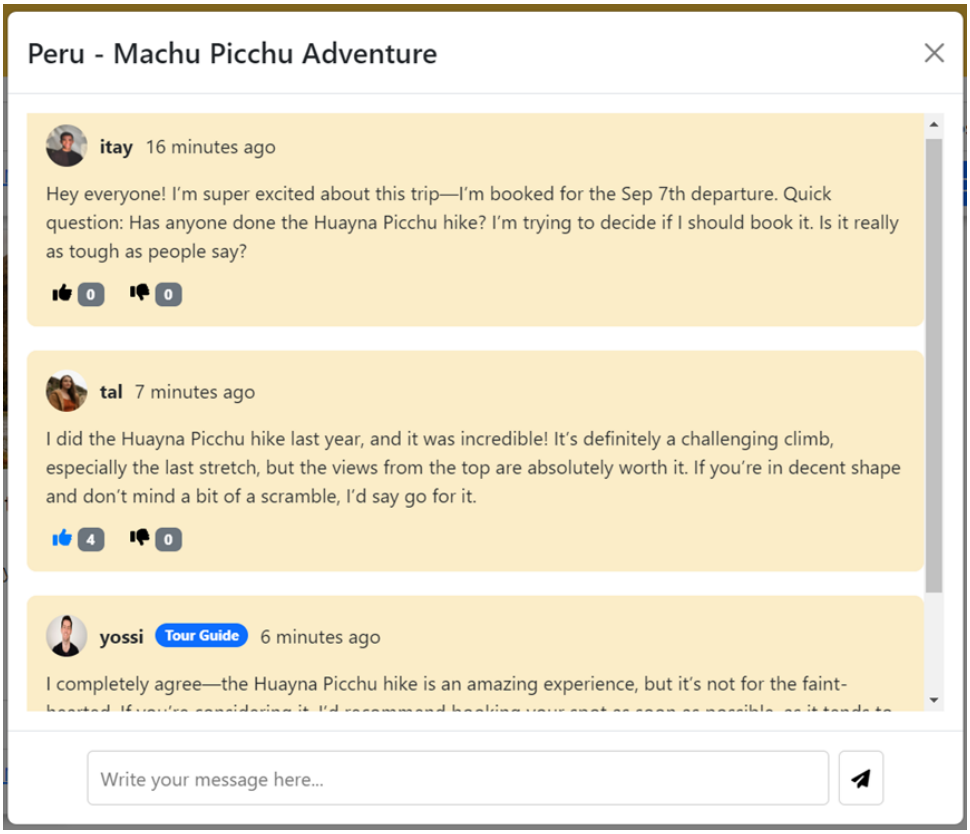
5.7 Trips page (home user page)

Here the users can see all the organized trips that the company offers. For each trip the user can see the trip name, duration, information and cost. To see more details on the trip, the user can click on 'Read More' link. In addition, each trip has a forum where they can talk with all the users. The user can search for a specific trip by using the search bar.



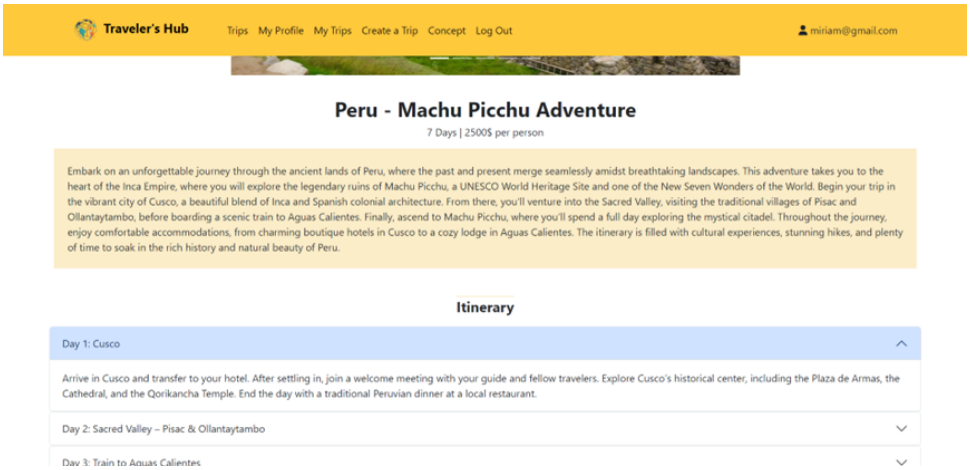
5.8 Forum pop-up

When users click on the 'Forum' button, they can see all the messages that have been sent by the users. They can write their own messages and send them too, by clicking on the 'paper plane' icon.



5.9 Trip Plan page

After clicking on the 'Read More' the user is directed to this page. Here the user can see all details on the trip plan (trip name, duration, cost, adventure canvas, itinerary, dates, the tour operator and the travel guide).



5.9.1 Dates section

In this section, the user can see the starting and ending date for the trip, as well as the status* of the date. The user can book a trip on a specific date by clicking on the 'Booking' button.

There are 3 statuses:

- **available** – The user can book a trip.
- **unavailable** – The date of the trip has already begun, therefore the user cannot book the trip on that date.
- **fully booked** – The trip on this date has reached the booking limit, therefore the user cannot book the trip on that date.

Tour operator's user:

- Tour operators can also see the number of travelers who have booked that trip and check if the admin has confirmed the trip.

Admin's user:

- Admins have the option to confirm each date in the trip by clicking on the 'Confirm' button. The user can undo this action by clicking the button again.

Dates

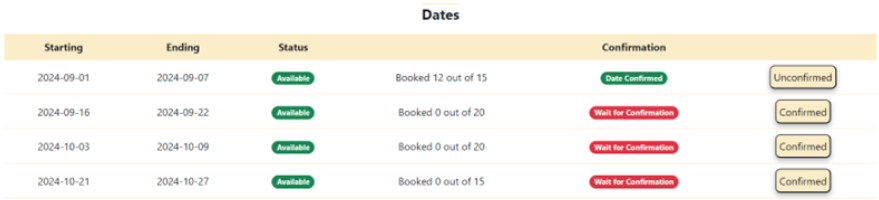
Starting	Ending	Status	
2024-09-01	2024-09-07	Available	Booking
2024-09-16	2024-09-22	Available	Booking
2024-10-03	2024-10-09	Available	Booking
2024-10-21	2024-10-27	Available	Booking

Figure 5.1: Dates Traveler

Dates

Starting	Ending	Status		Confirmation
2024-09-01	2024-09-07	Available	Booked 12 out of 15	Date Confirmed
2024-09-16	2024-09-22	Available	Booked 0 out of 20	Wait for Confirmation
2024-10-03	2024-10-09	Available	Booked 0 out of 20	Wait for Confirmation
2024-10-21	2024-10-27	Available	Booked 0 out of 15	Wait for Confirmation

Figure 5.2: Dates Tour Operator



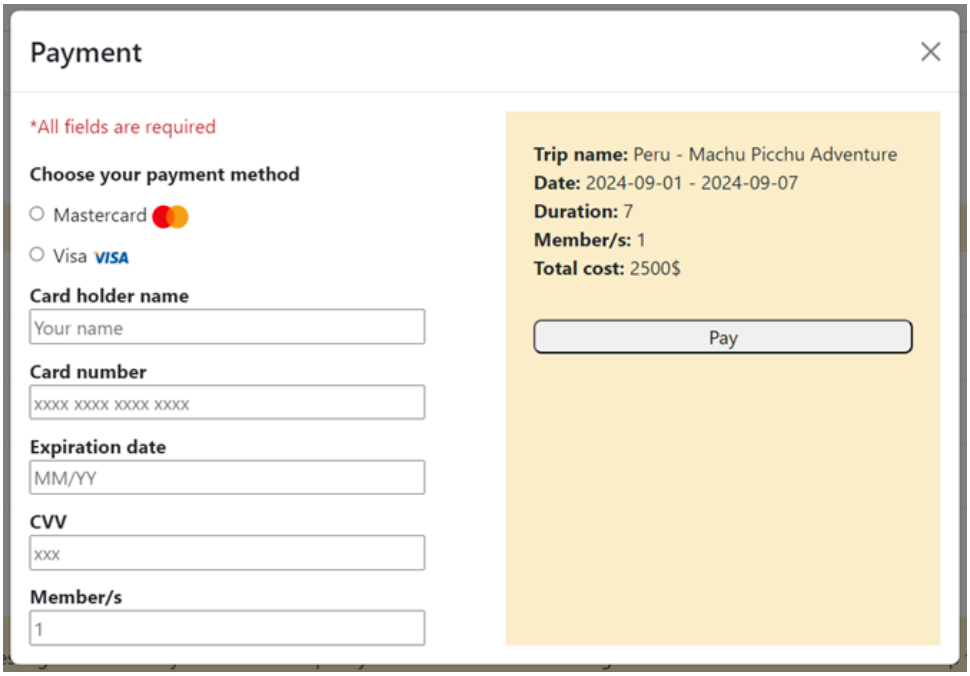
Dates

Starting	Ending	Status		Confirmation	
2024-09-01	2024-09-07	Available	Booked 12 out of 15	Date Confirmed	Unconfirmed
2024-09-16	2024-09-22	Available	Booked 0 out of 20	Wait for Confirmation	Confirmed
2024-10-03	2024-10-09	Available	Booked 0 out of 20	Wait for Confirmation	Confirmed
2024-10-21	2024-10-27	Available	Booked 0 out of 15	Wait for Confirmation	Confirmed

Figure 5.3: Dates Admin

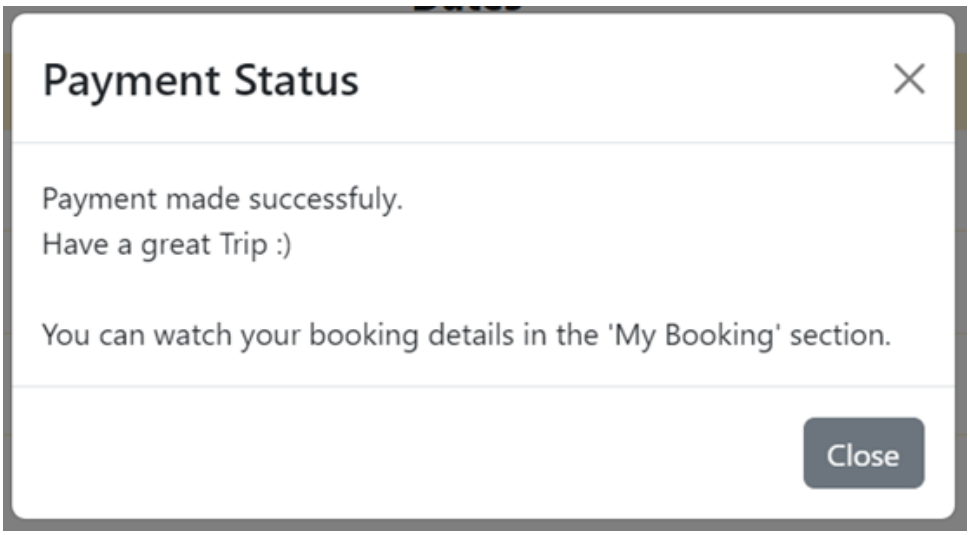
5.9.2 Trip registration

When the user clicks on the 'Booking' button, a pop-up window will be displayed. On the left side, the user needs to enter all the required fields (payment details and member). On the right side, the user can see a summary of the purchase.



5.9.3 Successfully purchase

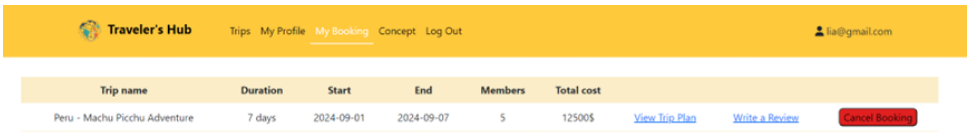
After clicking on the 'pay' button the user will see a successful message for the purchase and an option to go to the 'My Booking' page.



5.10 My Booking page

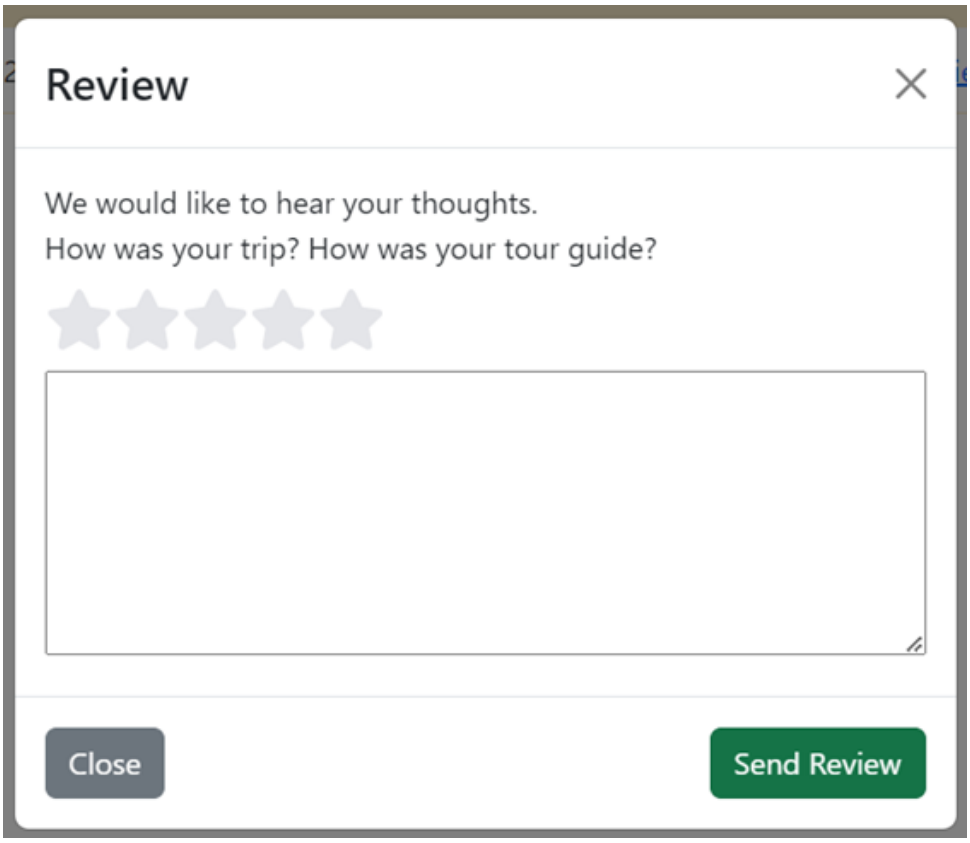
The user can see all the booking made. For each booking, the user can see the trip name, duration, start and end date, the members amounts and the total cost. In addition, the

user can click on 'View Trip Plan' to see all the trip details; by doing that, the user will be directed to the 'Trip Plan' page. The user can also write a review on the trip, by clicking on 'Write a Review'. To cancel a booking the user needs to click on the 'cancel booking' button.



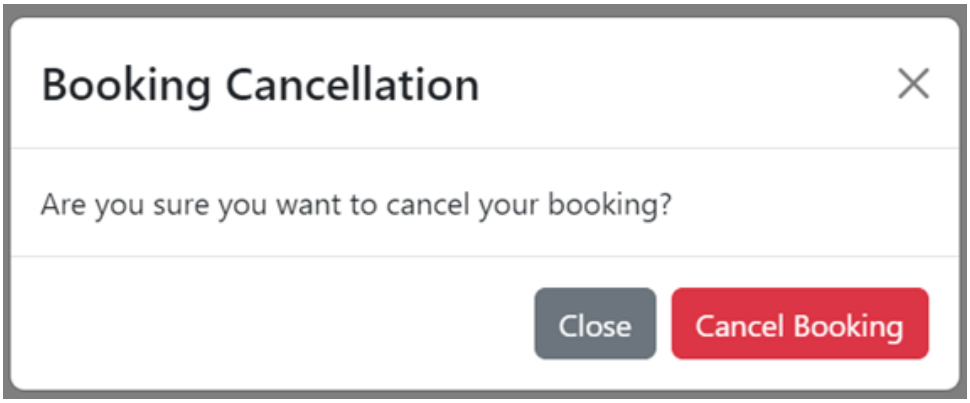
5.10.1 Write a Review

After clicking on 'Write a Review' a pop-up window will be displayed. Here the user can rate the trip from 0 to 5 and can write the review's content.



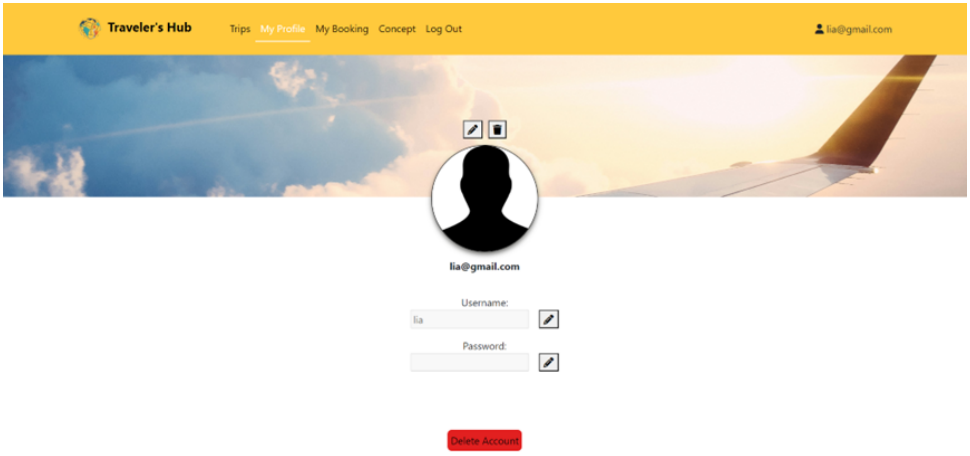
5.10.2 Cancel Booking

After clicking on 'Cancel Booking' a booking cancellation pop-up window will be displayed to confirm that the user wants to cancel the booking.



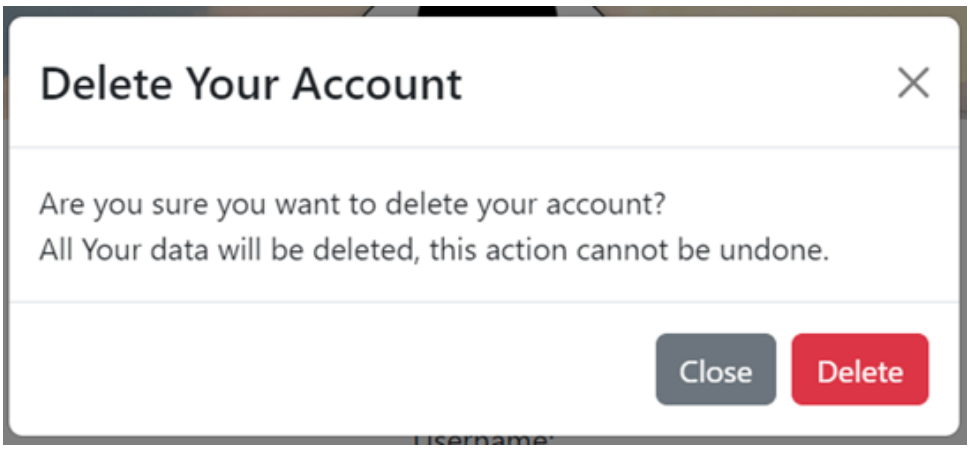
5.11 My Profile page

The user can update and delete the profile image. The user can also update the username and the password. To delete the account, the user can click on the 'Delete Account' button.



5.11.1 Delete Account

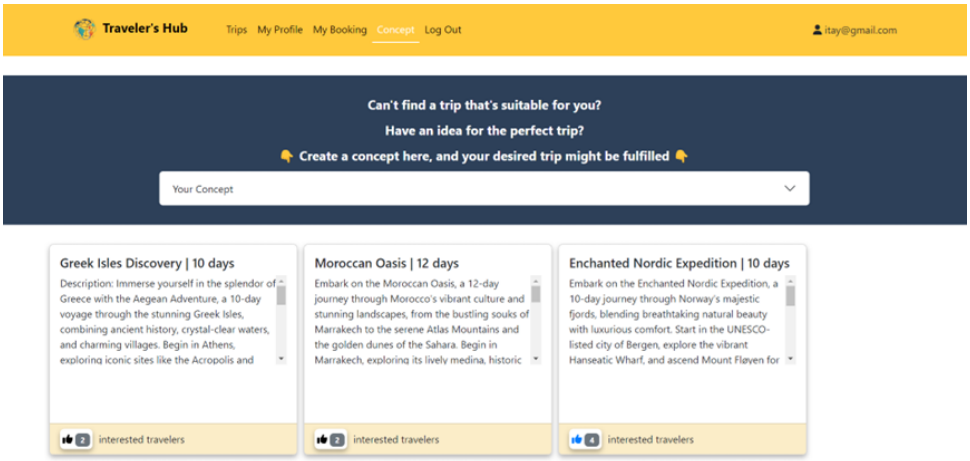
After clicking on the 'Delete Account' a pop-up window will be displayed to confirm that the user wants to delete the account.



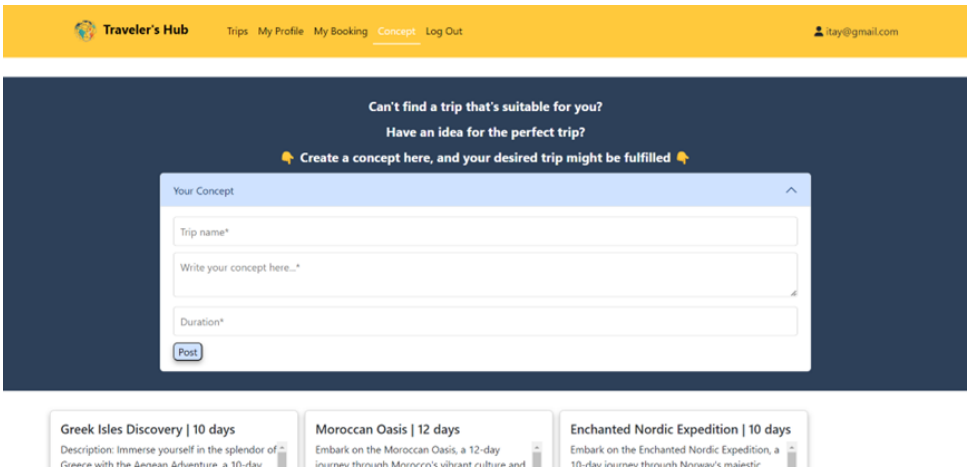
5.12 Concept page

The user can write a concept for a trip by filling in all the required fields and clicking on the 'Post' button. The user can see all the concepts that are made by all the users. For each concept, the user can see all the details for that concept and see the amount of users that are interested in the trip concept. If the user is interested in that concept trip, they can click on the like button icon. The user can undo this by clicking the like button again. **Tour Operator's** user can also promote and delete a concept.

Before clicking on 'Your Concept'

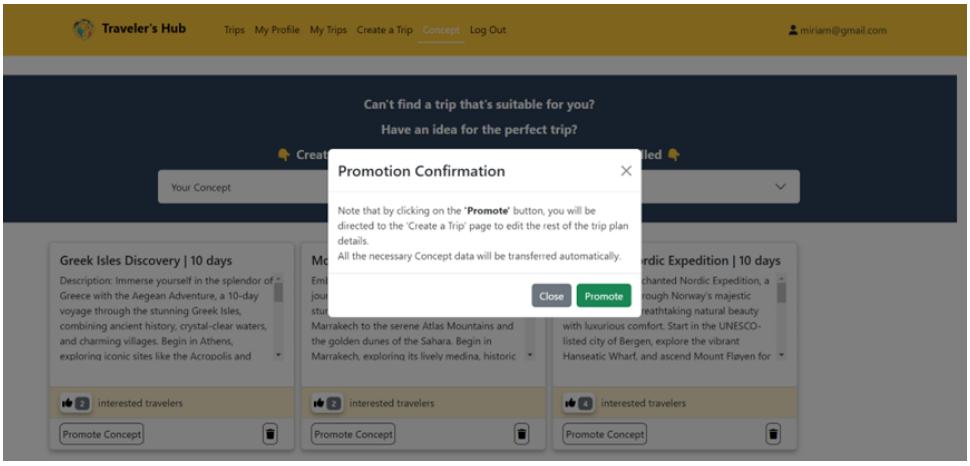


After clicking on 'Your Concept'



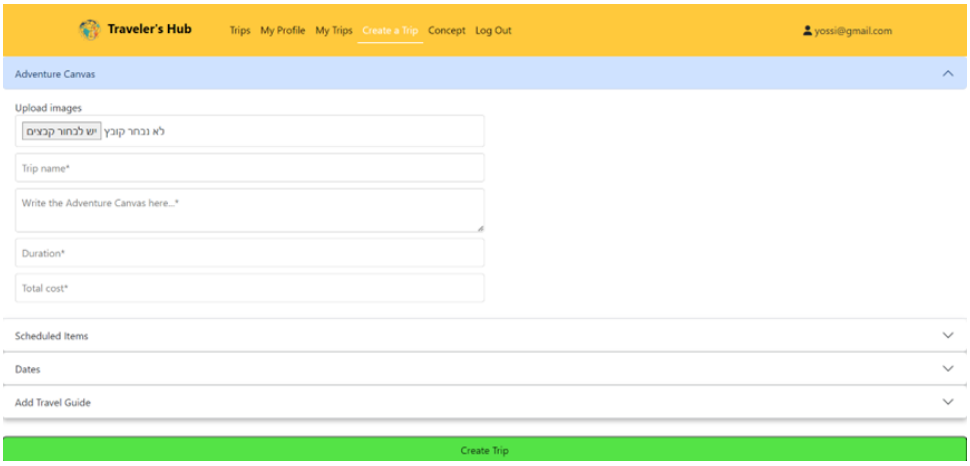
5.12.1 Promote a Concept

After clicking on 'Promote Concept' a promotion confirmation pop-up window will be displayed. By clicking on the 'Promote' button, the user will be directed to the 'Create a Trip' page.



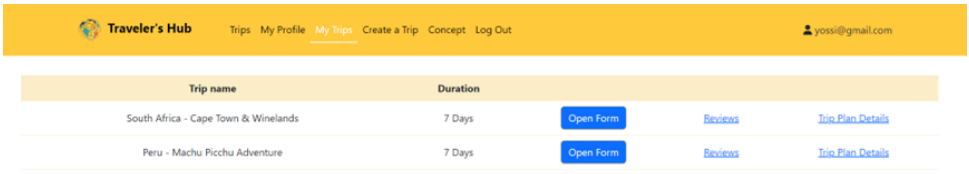
5.13 Create a Trip

Here the user can create a trip by filling in all the following sections: Adventure Canvas, Scheduled Item, Dates and Travel Guide (optional).
Adventure Canvas section – the user needs to upload images, enter the trip name, the Adventure Canvas (an overview information about the trip), duration and the cost (per person).
Scheduled Item section – the user needs to enter for each scheduled item the site name / location, the day and the description. By clicking on the 'Add' button the user will see the scheduled item on the right side of the section as a list.
Dates section – the user needs to enter for each date the starting and ending date and the limit of the travelers' group on that date. By clicking on the 'Add' button the user will see the dates on the right side of the section as a list.
Travel Guide section – the user can enter specific instructions/recommendations/tips for that trip.



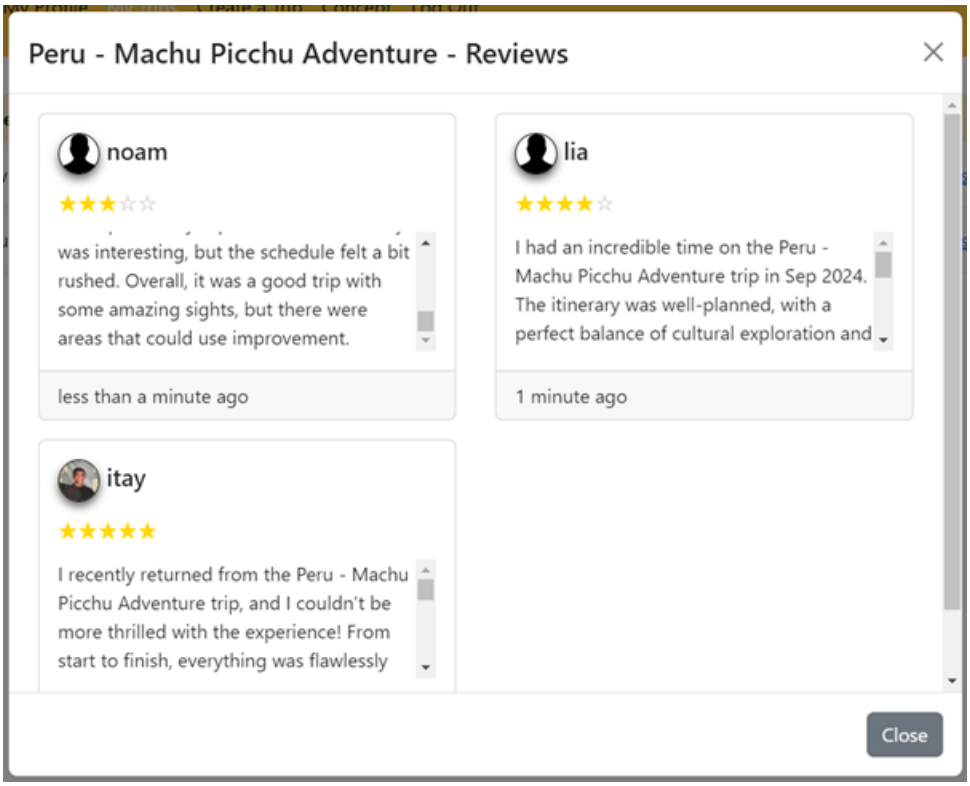
5.14 My Trips page

Here, the user sees all the trips they were assigned to. For each trip, the user can see the trip name, the duration, the forum discussions, the review and the trip plan details. By clicking on the trip plan details link, the user will be directed to the trip plan page.



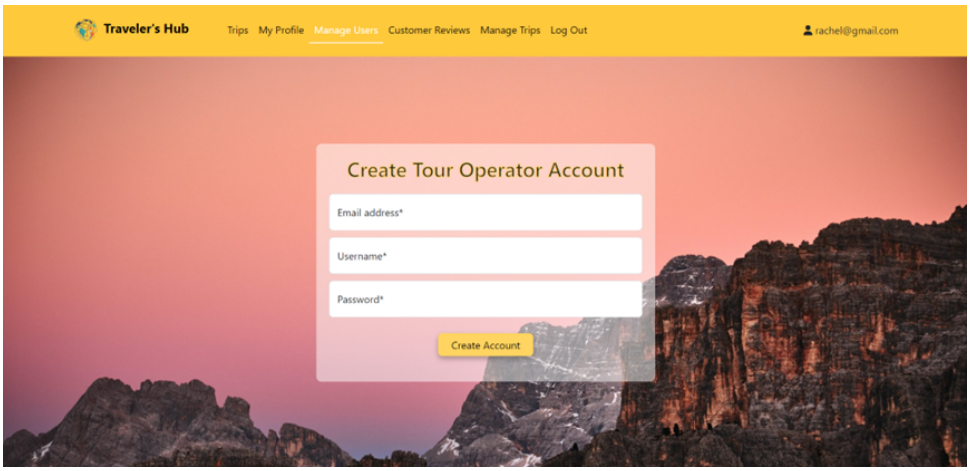
5.14.1 Review

After clicking on the 'Review' link, a pop-up window will be displayed. The user can see all the reviews that were written on that trip.



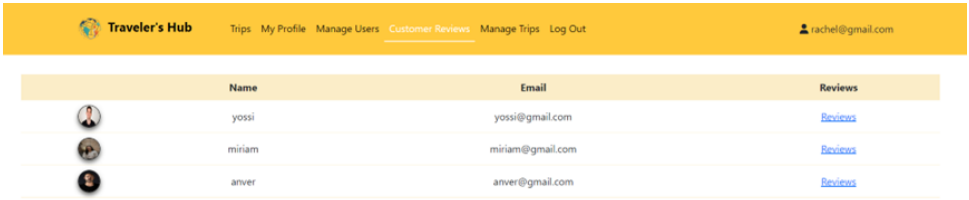
5.15 Manage User page

The user can create a tour operator account.



5.16 Customer Review page

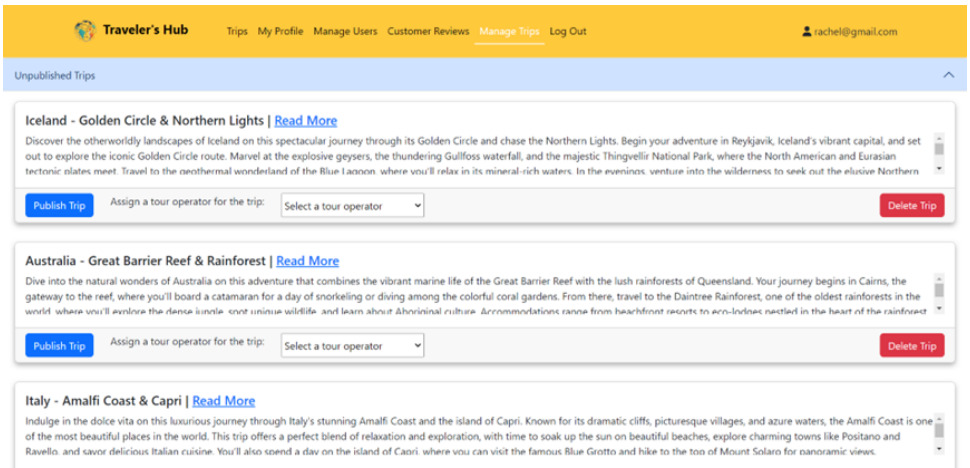
The user can see for each tour operator the reviews that were written on their trips.



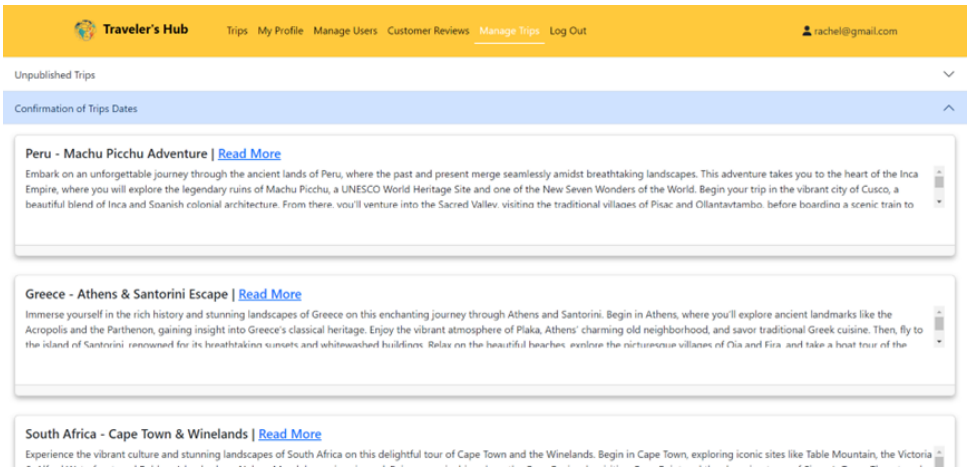
5.17 Manage Trips page

The user can see all the trips that wait to be published (under the Unpublished Trip section) and the trips that need to confirm their dates (under the Confirmation of Trips Dates section).
Unpublished Trip section – for each trip that needs to be published the user can see its details, by clicking on the 'Read More' link and be directed to the 'Trip Plan' page. For each trip the user needs to assign a tour operator from the list, before publishing the trip. In addition, the user can delete a trip by clicking on the 'Delete Trip' button.
Confirmation of Trips Dates section – for each trip on the list there is at least one date that waits for confirmation. By clicking on the 'Read More' link the user will be directed to the 'Trip Plan' page to confirm a date.

5.17.1 Trips that wait to be published



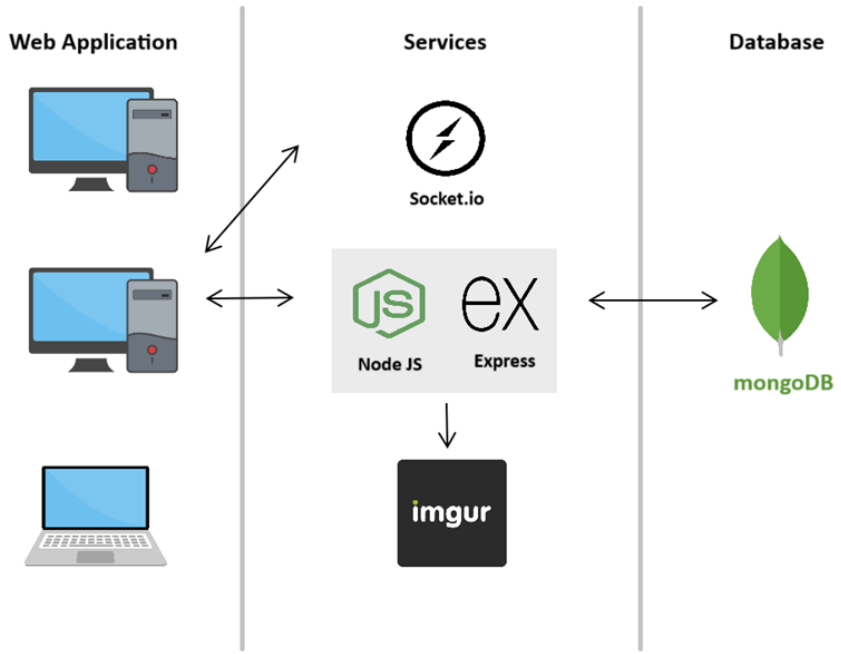
5.17.2 Trips that have dates waiting for confirmation



Chapter 6

Maintenance Guide

6.1 Software Architecture Diagram



6.2 Architecture Overview

The architecture of the application is designed to provide a seamless and dynamic user experience, real-time communication, and efficient data handling and storage. This architecture follows a client-server model, where the frontend is built with React and interacts with a Node.js and Express-based backend. The application also leverages Socket.IO for live chat functionality and integrates with Imgur for external image storage, ensuring that the application is both scalable and responsive.

6.3 Components of the Architecture

6.3.1 React Client

The frontend of the application is developed using React, which handles the user interface and manages the client-side application logic. Users interact with the application through a series of components that facilitate tasks like messaging, image uploads, profile management, and more. The React client communicates with the backend via RESTful APIs and uses Socket.IO to handle real-time messaging, ensuring an interactive and dynamic experience for users.

6.3.2 Node.js & Express Backend

The backend is built on Node.js with Express. It serves as the bridge between the frontend and the database, processing incoming requests, and managing data flow. The backend is responsible for authenticating users, handling user's functionality, and managing data storage. It integrates with Imgur through API requests to upload and retrieve images, which are then referenced in the database. Additionally, the backend uses Socket.IO to maintain persistent, real-time communication channels with the React client, enabling live chat functionality and immediate data updates.

6.3.3 MongoDB Database

MongoDB serves as the primary database for the application, providing a flexible and scalable solution for storing user data, chat history, and image paths. The database is accessed by the backend through Mongoose.

6.3.4 Socket.IO for Real-Time Communication

Socket.IO is integrated into both the frontend and backend, allowing for real-time communication between the client and the server. This is crucial for the live chat feature, where messages need to be instantly relayed to all participants. The use of WebSockets ensures that the chat is responsive and that users receive updates in real-time without needing to refresh the application continuously.

6.3.5 Imgur Integration

For handling image uploads, the backend is integrated with Imgur, an image hosting service. When a user uploads an image, it is sent to the backend, which then forwards the image to Imgur using its API. Upon successful upload, Imgur returns a URL, which is stored in MongoDB as a reference. This approach offloads the storage and serving of images to Imgur, allowing the application to remain lightweight and scalable while providing users with fast and reliable image hosting.

Chapter 7

GIT

Explore the Traveler's Hub on GitHub, https://github.com/VelitskyLev-Coder/TravelerHub

References

1. TripIt: A travel organization app that creates itineraries from forwarded emails.
2. Google Travel: A comprehensive tool offering trip planning, recommendations, and offline access.
3. TripHobo: A trip planning platform offering detailed itineraries and collaboration features.
4. Roadtrippers: A tool for planning road trips, helping users discover attractions and customize their routes.
5. React: A JavaScript library for building user interfaces.
6. Node.js: A JavaScript runtime built on Chrome's V8 JavaScript engine for server-side development.
7. Express: A web application framework for Node.js, designed for building web applications and APIs.
8. MongoDB: A NoSQL database used for storing data in a flexible, JSON-like format.
9. Socket.IO: A JavaScript library for real-time web applications, enabling real-time, bidirectional communication between web clients and servers.
10. Imgur API: A platform for image hosting, used in the project to handle user-uploaded images.
11. Mongoose: An object data modeling (ODM) library for MongoDB and Node.js.
12. PlantUML: An open-source tool used to create UML diagrams from plain text descriptions.
13. Visual Paradigm: A visual modeling tool that supports UML, SysML, ERD, DFD, and other diagramming capabilities.
14. Client–server model: An architectural model for distributing applications between clients and servers.