



CS492 - SENIOR DESIGN PROJECT II

Travimize: Optimized Travel, Personalized Experience

FINAL REPORT

18.05.2023

Supervisor: Selim Aksoy

Innovation Expert: Perin Ünal

Team ID - T2330

Doğa Ece Ersoy - 21902487

Doruk Kantarcıođlu - 21902319

Emre Erdal - 21901597

Esra Genç - 21901962

Kerem Erdal - 21901596

1. Introduction	2
1.1. Purpose of the System	3
1.2. Design Goals	3
1.1.1. Usability	3
1.1.2. Robustness	3
1.1.3. Security	4
1.1.4. Recovery	4
1.1.5. Scalability	4
1.1.6. Flexibility	4
1.1.7. Reliability	4
1.1.8. Marketability	4
1.1.9. Pseudo Requirements	4
1.3. Definitions, Acronyms, and Abbreviations	5
1.4. Overview	6
2. Current Software Architecture	6
3. Proposed Software Architecture	6
3.1. Overview	6
3.2. Subsystem Decomposition	7
3.3. Hardware/Software Mapping	9
3.4. Persistent Data Management	9
3.5. Access Control and Security	10
4. Subsystem Services	10
4.1. UI Layer	10
4.2. Controller Layer	11
4.3. Service Layer	11
4.4. Repository Layer	12
4.5. MySQL Database	13
5. Test Cases	14
5.1 Functional Test Cases	14
5.2 Non Functional Test Cases	52
6. Consideration of Various Factors in Engineering Design	71
7. Teamwork Details	73
7.1. Contributing and Functioning Effectively on the Team	73
7.2. Helping Creating a Collaborative and Inclusive Environment	74
7.3. Taking Lead Role and Sharing Leadership on the Team	74
9. References	74

1. Introduction

Travimize is an application which tries to facilitate people's lives while making plans for their holidays. First of all, it tries to apply the optimization process for the journeys by allowing the person to visit the places they want at the shortest time. The user can first select his/her interests, the places he/she wants to visit during his holiday, and traveling options like walking, public transport, etc. By observing the opening, closing times, distance between two locations, and the choices of the user, the algorithm will provide the optimal route for the journey. By the way, users would also be able to get informed about the distance between two locations and which transportation would be fastest that would help their selection process. The application would just propose a schedule which can further be changed by the person according to his/her own desires, other plans or some schedule changes. Along with these scheduling plans, the user will be able to specify some price, eating, accommodation related choices. Accordingly, hotels and restaurants will be proposed by the application and the user can also be directed to associated websites for hotel reservations. Moreover, the user will be able to rate their plans or specifically places and can write comments which would provide feedback not only for the other customers but also to program admins. These feedback questions would be about the time spent in a specific location, if that place is open during the visit, and then future plans will be created by taking these into consideration. With the help of these evaluations, a user going to a specific location would be firstly informed about the high rated plans and then he/she may have a chance to select these previously liked plans by making some changes or applying them straightforwardly. Finally, the users would also be able to see their previous plans and may repeat or personalize it when they visit the same place again. As a result, Travimize helps people to optimize their holidays and personalize the schedules by specifying their interests at the beginning or changing them according to their desires at the end.

The type of innovation that is planned to be implemented for Travimize is service. As Travimize aims to bring a new perspective to traveling and ease the process of vacation planning, it is included in the category of service innovation that prioritizes customers and their experience. Travimize is also open to performance and functionality enhancements according to the feedback coming from its customers. The application is a radical innovation as it combines accommodation, food, transportation, and tourism industries. [1].

1.1. Purpose of the System

The Travimize application is a travel planning tool that is created to provide an optimized and personalized travel planning experience for its users. The application achieves this goal by providing the users with a comprehensive and user-friendly user interface. The main objective of the application is to enable users to make the most of their holiday time, create lifelong memories, and discover new and intriguing destinations based on their interests and preferences.

Travimize application aims to reduce the stress and uncertainty often associated with travel planning, and simplify the travel planning process, enabling users to create efficient travel schedules based on their interests and choices by streamlining the planning process. The application provides users with relevant information on transportation options, opening and closing times, and other factors that impact their travel plans through data-driven insights and advanced algorithms.

1.2. Design Goals

1.1.1. Usability

- The user interface should be easy to use or learn for maintaining maximum convenience and user friendliness for the target audience.
- Each component of the user interface should have a meaningful and clear name.
- The user interface should offer a guide on navigating the website for users.
- All screens must be independent of one another, with the exception of pop-ups, to prevent users from having to navigate backward through the application.
- The user should be able to log in to his or her account from up to 4 personal devices.

1.1.2. Robustness

- The database should be created in such a way that it can easily respond to the requests from the application.
- The database should respond to an input that is prone to errors without any issues.

1.1.3. Security

- All of the personal data of the users (name, mail, phone number, etc.) should be protected by the application.

1.1.4. Recovery

- Regular database backups should be done at weekly intervals. Data should be saved in the event of a system failure.

1.1.5. Scalability

- The application should handle up to 10000 users logging in to the system without crashing.

1.1.6. Flexibility

- The application should be flexible enough to accommodate changes in user preferences or requirements.
- The application should be flexible enough to adapt to updates to third-party APIs and data sources.

1.1.7. Reliability

- The application should minimize downtime. It should be ideally 0, however downtime up to 40 minutes per month is allowed.
- The application should ensure data consistency.
- The application should handle errors gracefully.

1.1.8. Marketability

- The design of the application should be visually appealing and aligned with the target audience's preferences.
- Travimize should be compatible with a variety of devices and operating systems, including mobile devices, tablets, and desktops.
- The marketing strategy should be well planned and implemented to reach at least 20000 target audience through various channels, including social media, online advertising, and word-of-mouth.

1.1.9. Pseudo Requirements

- Spring Framework will be used for the backend development of the application.
- React will be used for the frontend development of the application.
- MySQL as a relational database will be used to store application information.

1.3. Definitions, Acronyms, and Abbreviations

UI	Abbreviation for user interface.
HTTP	Abbreviation for The Hypertext Transfer Protocol and is used to load web pages using hypertext links.
API	Abbreviation for Application Programming Interface and is a set of mechanisms that specifies how software components should interact with each other.
JDBC	Abbreviation for Java Database Connectivity and is a Java API that enables Java applications to interact with databases.
RDBMS	Abbreviation for Relational Database Management System.
MySQL	MySQL is a free, open-source relational database management system.
Google OR-Tools	Google OR-Tools is a commercial optimization software package developed by Google.
TwilioEmailAPI	Email API by Twilio.
Spring Boot Framework	Spring Boot is a popular open-source framework for building enterprise-grade Java applications.
Bearer token	A Bearer token is a type of access token that is used in authentication and authorization protocols to grant access to protected resources.
BCrypt algorithm	BCrypt is a cryptographic hash function that is commonly used for password hashing and verification.
HS512 algorithm	HS512 is a type of cryptographic algorithm used for digital signatures and message authentication.

Table 0: Definitions, acronyms, and abbreviations

1.4. Overview

Travimize is a web application designed to help people plan their holidays by optimizing and personalizing their traveling plans. The application allows users to select their interests, places to visit, and transportation options. The application then uses an algorithm to provide the optimal route for the trip. Users can also specify their price, eating, and accommodation related choices, and the application proposes hotels and restaurants accordingly. Moreover, the application allows users to rate their plans and write comments, providing feedback for other customers and system admins. Travimize is categorized as a service innovation that prioritizes customers and their experience, and it is a radical innovation as it combines accommodation, food, transportation, and tourism industries.

2. Current Software Architecture

In terms of travel planning, there are several existing applications such as TripAdvisor [2], TripIt [3] and Inspirock [4]. TripAdvisor allows users to customize their trip by saving the hotels, restaurants, activities or tours that catch their attention. They can either schedule their saved items or leave them unscheduled. TripIt chooses travel-related mails from your inbox such as ticket or booking confirmation and creates the plan automatically. Rather than attractions or activities, TripIt mostly focuses on planning flights, hotels, car rentals, etc. Among these applications, Inspirock is the one that has more similarities with Travimize. According to entered destinations and dates, Inspirock comes up with plans for each day that include landmarks, parks or museums to visit for the specified destinations. It also redirects to different websites for hotel options. However, unlike Travimize, it does not consider the money range, sleep hours, walking or food preferences of the users before creating the plan. Most importantly, Travimize differs from all of the existing applications as none of them tries to achieve creating the most optimized travel plan according to the users' preferences.

3. Proposed Software Architecture

3.1. Overview

Travimize is a web application composed of a presentable, convenient user interface serving the purposes of the client; and a service-oriented back-end server to

utilize and present data. In addition to the client-server architecture, the architecture is also layered. To elaborate, each layer (UI, controller, service, repository) can only access the lower layer to ensure proper application flow, reduced dependency, and encapsulation (also known as opaque layering). The entity-related retrieval/writes take place in the repository layer. Business logic involving operations such as travel plan construction, plan modification, place management and user management will take place in the service layer. Additionally, the service layer will utilize external modules such as Google OR-Tools and TwilioEmailAPI. The layer that will communicate with the client and handle all the request-response cycles is the controller. On top of the controller, the UI layer will present the responses of the back-end server to the user with a user-friendly interface.

3.2. Subsystem Decomposition

As stated above, there are four layers composed of UI, controller, service and repository layers, where only the repository layer has access to the database to execute entity-related reads/writes. All layers have separate components inside themselves. Instead of a monolithic architecture, microservices and service-oriented architecture patterns are utilized. Service layer also utilizes external components to send emails and provide optimal travel plans etc.

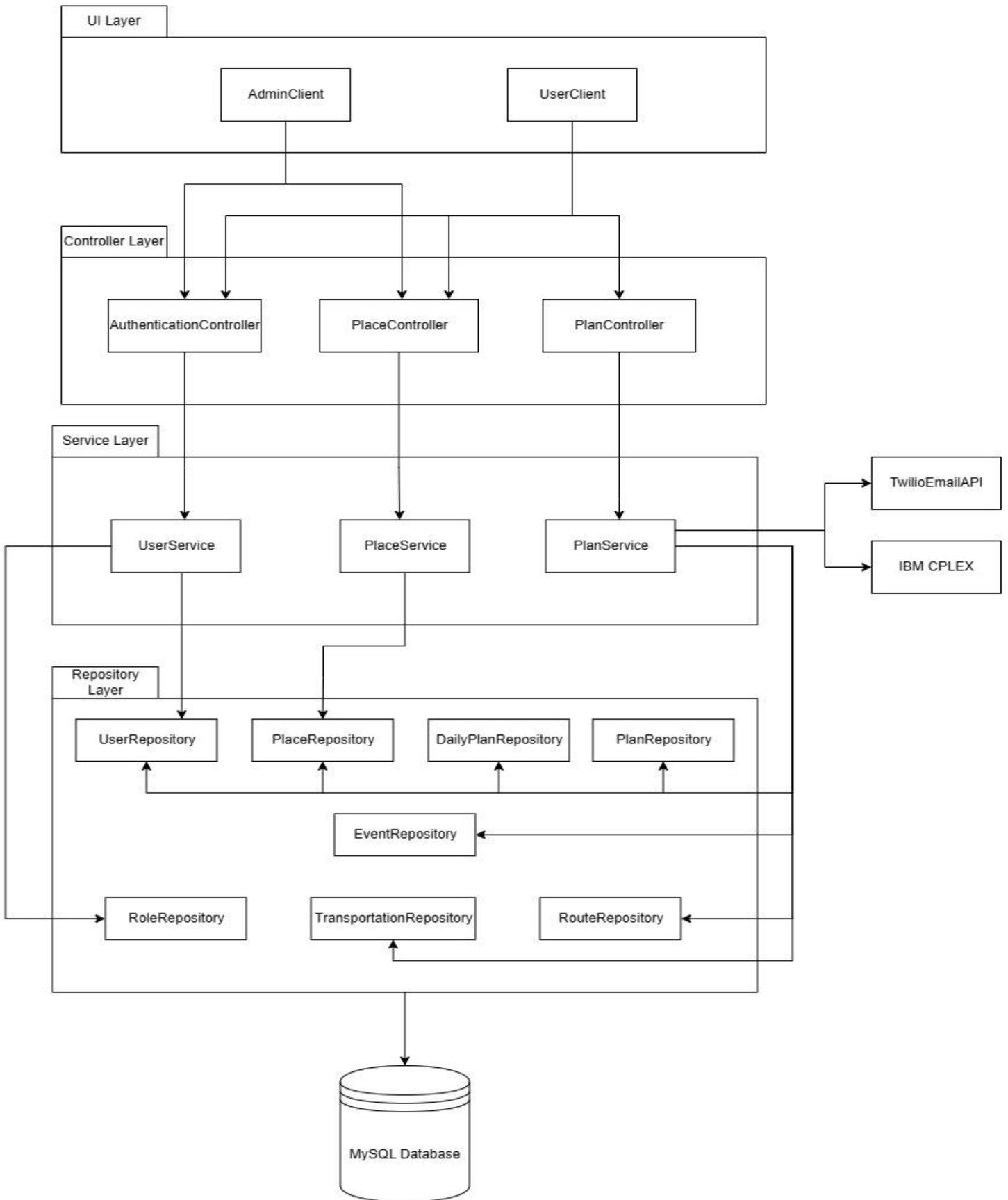


Figure 1: Subsystem Decomposition Diagram

3.3. Hardware/Software Mapping

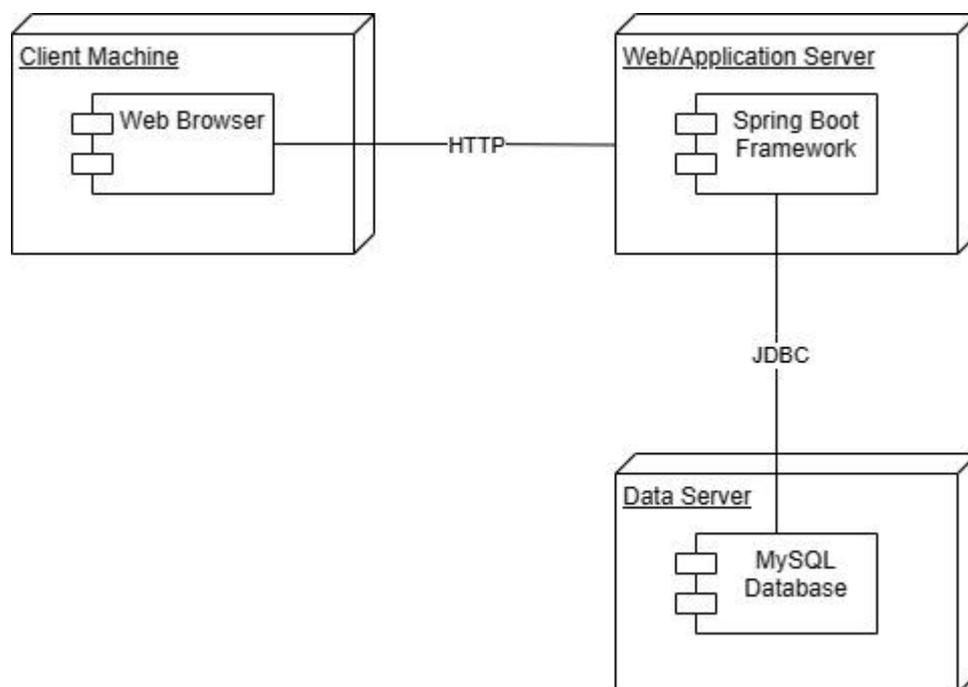


Figure 2: Hardware/Software Mapping Diagram

The hardware/software mapping of the project is simple in terms of number of components and interactions. The client will interact with the web server via HTTP requests/responses. Before returning responses to the client, the web server will interact with the database for retrievals/modifications. The connection between the client and the server will be through HTTP requests. The web server and the database will utilize the in-built JDBC driver of Spring Boot framework. MySQL is utilized for the underlying database.

3.4. Persistent Data Management

The most important type of data to reside in the database is of course user data, consisting of the information supplied during the registration process (username, hashed password, email address) and additionally the plans that they have created (places, start/end dates, time intervals for events). These two types of data are the most sensitive regarding the end users, hence they are mentioned in this article.

3.5. Access Control and Security

In the Travimize application, users (normal/admin) are authenticated via role-based authentication, with users only being able to access their own resources. Since travel plans are important pieces of personal data, our aim before designing the infrastructural systems was to first develop a secure-first API. Super-user functionalities such as adding a place are only exclusive to users with the admin role (e.g. admin user).

Every request except the login/register requests need to be authenticated with a Bearer token in order to ensure resource protection. Upon registering, the user's credentials are saved into the database, the password hashed with the BCrypt algorithm. The password hashing ensures ultimate data privacy in the worst case of a data breach, and also ensures an industry-standard on application data security. After a successful login, the response contains the Bearer token to be utilized by the authenticated user's requests to the web server. The token is signed with the HS512 algorithm, and the payload only contains the username, issued at field and expires at field. The username from the payload is utilized to check the resource integrity of the request. Expired tokens will be obsolete, hence making the request invalid.

4. Subsystem Services

4.1. UI Layer

4.1.1. AdminClient

AdminClient package consists of the front-end of the pages where only admins interact with. These pages include AdminHome and AdminAddPlace which help the admins of the system to navigate the homepage and add a new place to the database.

4.1.2. UserClient

UserClient package consists of the front-end of the pages where there is no need for an admin authentication. These pages include simple pages like homepage, setting page or the user's profile page, as well as pages that enable users to create a traveling plan or modify their existing plans.

4.2. Controller Layer

4.2.1. AuthenticationController

The `AuthenticationController` class includes authentication related response logics such as login and register. This class is used for both the users and admins (`UserClient` and `AdminClient`). This controller takes the information for the response from the `UserService`.

4.2.2. PlaceController

The `PlaceController` class includes creating and viewing the places. This class is used for both the users and admins (`UserClient` and `AdminClient`). The admins use the system for viewing and creating the places and the users use the system for only viewing places. This controller takes the information for the response from the `PlaceService`.

4.2.3. PlanController

The `PlanController` class includes creating and viewing the plans, editing and creating the events together with viewing, and viewing the daily plans. This class is used for only the users (`UserClient`). The users use the system for the creating, editing and viewing the events, and plans. This controller takes the information for the response from the `PlanService`.

4.3. Service Layer

4.3.1. UserService

This class is used by the `AuthenticationController` class and deals with the authentication related business logics such as login and register. By the `authenticate` and `register` functions, the registration and logging in to the system is done by the appropriate roles. The `UserService` takes the necessary information for the business logic from the `UserRepository` and `RoleRepository`.

4.3.2. PlaceService

This class is used by the `PlaceController` class and deals with creating and viewing the places. By the `create` and `get` functions, the admin can create and view the plans, and the user can just view the plans. The `PlaceService` takes the necessary information for the business logic from the `PlaceRepository`.

4.3.3. PlanService

This class is used by the PlanController class and deals with creating and viewing the plans, editing and creating the events together with viewing, and viewing the daily plans. The users use the system for the creating, editing and viewing the events, and plans by the functions such as create, edit, view and delete. This service also uses TwilioEmailAPI as an external package for the notifications that will be sent automatically to the users by mail system. Also, the PlanService uses one more external package named as Google OR-Tools for the optimization problem solved to create the plan. The PlanService takes the necessary information for the business logic from the PlaceRepository, UserRepository, PlanRepository, DailyPlanRepository, EventRepository, TransportationRepository, and RouteRepository.

4.4. Repository Layer

4.4.1. UserRepository

UserRepository is used to get the user related information from the database. This information is managed and utilized on the service layer with UserService.

4.4.2. PlaceRepository

PlaceRepository is used to get the information related to the attractions from the database. This information is managed and utilized on the service layer with PlanService and PlaceService.

4.4.3. DailyPlanRepository

DailyPlanRepository is used to get the information related to the one day of traveling plans created by the users from the database. This information is managed and utilized on the service layer with PlanService.

4.4.4. PlanRepository

PlanRepository is used to get the information related to the traveling plans created by the users from the database. This information is managed and utilized on the service layer with PlanService.

4.4.5. EventRepository

EventRepository is used to get the information related to the events that are planned to occur in one time slot of the traveling plans created by the users

from the database. This information is managed and utilized on the service layer with PlanService.

4.4.6. RoleRepository

RoleRepository is used to get the information about the authentication role of the user who is logged in to the system from the database. This information is managed and utilized on the service layer with UserService.

4.4.7. TransportationRepository

TransportationRepository is used to get the information related to the transportation options and their durations and content for every traveling plan that is created by the users from the database. This information is managed and utilized on the service layer with PlanService.

4.4.8. RouteRepository

RouteRepository is used to get information related to the locations of every event in the traveling plans, and the route that specifies how to travel between these locations from the database. This information is managed and utilized on the service layer with PlanService.

4.5. MySQL Database

MySQL is chosen as the RDBMS that will be used for the Travimize application. All the user, place, traveling plan, transportation related information that is needed by the Travimize application is kept in a MySQL database. This information is received from the application with the help of JDBC.

5. Test Cases

5.1 Functional Test Cases

Test ID	Test 1
---------	--------

Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Signup Button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Signup button redirects to the Signup page ● Check if this button can be clicked when any of the username or password fields in login page is filled ● Check if the Signup button is clickable by mouse ● Check if the Signup button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to the Signup button on the main login page redirects to Signup page that contains the form for signing up to application ● This button can be clicked and serves the intended functionality even if any of the fields in the login page are filled ● Signup button can be clicked both by mouse and touchpad.
Priority	Major

Table 1: Test case table 1

Test ID	Test 2
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Email Field on Signup Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the email field by clicking on it ● Check if the user can type in the email field ● Check if the user can paste to the email field ● Check if the email entered by the user is different from the registered ones ● Check if the entered email is missing @ or domain ● Check if the error message is displayed when an invalid email is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the email field ● If the email entered by the user is valid, it should be accepted without any errors ● If the email entered by the user is already registered, the appropriate error message should be displayed ● If the email entered by the user is missing @ or domain, the appropriate error message should be displayed ● If the email entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 2: Test case table 2

Test ID	Test 3
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Password Field on Signup Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the password field by clicking on it ● Check if the user can type in the password field ● Check if the user can paste to the password field ● Check if the entered password is masked in the form of asterisks ● Check if the entered password meets the length requirements ● Check if the password field accepts lowercase letters ● Check if the password field accepts uppercase letters ● Check if the password field accepts numbers ● Check if the password field accepts special characters ● Check if the password field accepts spaces ● Check if the error message is displayed when an invalid password is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the password field ● The password entered by the user should be displayed in the form of asterisks ● If the password entered by the user is valid, it should be accepted without any errors ● If the password entered by the user does not meet the length requirements, the appropriate error message should be displayed ● If the password entered by the user does not meet the character requirements, the appropriate error message should be displayed ● If the password entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 3: Test case table 3

Test ID	Test 4
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Full Name Field on Signup Page
	<ul style="list-style-type: none"> ● Check if the user can access the full name field by clicking on it

Procedure	<ul style="list-style-type: none"> ● Check if the user can type in the full name field ● Check if the user can paste to the full name field ● Check if the user is allowed to type only alphabetical characters in the full name field ● Check if the error message is displayed when an invalid full name is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the full name field ● If the full name entered by the user is valid, it should be accepted without any errors ● If the full name entered by the user includes characters that are not alphabetical, the appropriate error message should be displayed ● If the full name entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 4: Test case table 4

Test ID	Test 5
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Confirm Password Field on Signup Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the confirm password field by clicking on it ● Check if the user can type in the confirm password field ● Check if the user can paste to the confirm password field ● Check if the entered password is matched with the one typed in the password field ● Check if the error message is displayed when an invalid password is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the confirm password field ● The password entered by the user should be displayed in the form of asterisks ● If the password entered by the user is valid, it should be accepted without any errors ● If the password entered by the user does not match with the one typed in the password field, the appropriate error message should be displayed ● If the password entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 5: Test case table 5

Test ID	Test 6
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Signup Button on Signup Page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Signup button redirects to the main Login page ● Check if this button can be clicked when any of the email or password fields in login page is not filled ● Check if the Signup button is clickable by mouse ● Check if the Signup button is clickable by keyboard ● Check if the Signup button can be clicked by pressing Enter button on the keyboard ● Check if verification email is sent to the user after he/she clicks to the Signup button ● Check if the user is added to database before email verification is completed ● Check if appropriate message is displayed if there is problem with any of the fields ● Check if appropriate message is displayed if the verification mail is sent to complete signup process
Expected results	<ul style="list-style-type: none"> ● Clicking to this button redirects the user to Login page ● This button can be clicked by mouse or touchpad when all the fields are filled with required conditions ● If any of the fields do not satisfy the required conditions, appropriate error message will be displayed ● The signup button can also be clicked by pressing Enter on the keyboard ● Verification email is sent to user if all the fields satisfy required conditions and the user is added to “Pending Users” database table before email verification is completed ● A message indicating the verification mail is sent is displayed to the user after clicking to the Signup button
Priority	Critical

Table 6: Test case table 6

Test ID	Test 7
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Email Verification for SignUp
	<ul style="list-style-type: none"> ● Check if accepting this verification enables adding that user to the Users database ● Check if the user is directed to Login page after approving the

Procedure	<p>verification</p> <ul style="list-style-type: none"> ● Check if the user can Login to the application after being directed to Login page with the determined credentials ● Check if the message for verification is sent to the provided email ● Check if appropriate message is sent for verification of account creation ● Check if this verification email does not go to Spam folder in the email
Expected results	<ul style="list-style-type: none"> ● The email verification message is sent to the mail provided during signup ● Clicking to provided link in the email makes the users added into the database table of “Real Users” after removing them from “Pending Users” table ● The user is directed to login page after completing verification and can enter the application by using the provided credentials during Signup ● The verification email sent to the users go to Inbox folder instead of Spam
Priority	Major

Table 7: Test case table 7

Test ID	Test 8
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Login button on Signup page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Login button redirects to the Login page ● Check if this button can be clicked when any of the fields is empty ● Check if this button can be clicked when any of the fields does not satisfy the required conditions ● Check if this button can be clicked when all of the fields for Sign Up are filled ● Check if the user is added to database if Login button is clicked before clicking to SignUp button ● Check if the email verification is sent to the user when Login button is clicked instead of SignUp button ● Check if the Login button is clickable by mouse ● Check if the Login button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to the Login button at Signup page redirects the user to the Login page independent from the conditions about the fields at this page

	<ul style="list-style-type: none"> ● If the information of any new user is entered in Sign Up page, it should not be added to database if Login button is pressed ● If the information of any new user is entered in Sign Up page, email verification should not be sent if Login button is pressed ● Login button is clickable by mouse or touchpad
Priority	Minor

Table 8: Test case table 8

Test ID	Test 9
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Password Field on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the password field by clicking on it ● Check if the user can type in the password field ● Check if the user can paste to the password field ● Check if the entered password is masked in the form of asterisks ● Check if the entered password meets the length requirements ● Check if the password field accepts lowercase letters ● Check if the password field accepts uppercase letters ● Check if the password field accepts numbers ● Check if the password field accepts special characters ● Check if the password field accepts spaces ● Check if the error message is displayed when an invalid password is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the password field ● The password entered by the user should be displayed in the form of asterisks ● If the password entered by the user is valid, it should be accepted without any errors ● If the password entered by the user does not meet the length requirements, the appropriate error message should be displayed ● If the password entered by the user does not meet the character requirements, the appropriate error message should be displayed ● If the password entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 9: Test case table 9

Test ID	Test 10
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Username Field on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the username field by clicking on it ● Check if the user can type in the username field ● Check if the user can paste to the username field ● Check if the username entered by the user is already registered ● Check if the error message is displayed when an invalid username is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the username field ● If the username entered by the user is valid, it should be accepted without any errors ● If the username entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 10: Test case table 10

Test ID	Test 11
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Login Button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Login button redirects to the home page if such an account exists with given username and password ● Check if appropriate error message is displayed when any problem exists with username or password fields ● Check if the appropriate error message is displayed if no account exists with given username and password ● Check if the appropriate error message is displayed if email verification is not completed ● Check if the appropriate message is displayed if login is accepted ● Check if the Login button is clickable by mouse ● Check if the Login button is clickable by keyboard ● Check if the Login button can be clicked by pressing Enter on keyboard
	<ul style="list-style-type: none"> ● If an account exists with the written credentials (username and password), clicking to the Login button displays an appropriate message indicating successful login and the user

Expected results	<p>is directed to his/her home page.</p> <ul style="list-style-type: none"> ● If any problem exists with the requirements of username and password fields or there is no account with written credentials, an appropriate error message is displayed to the user. ● An appropriate message is displayed after clicking the Login button if the account with written credentials has not completed email verification yet. ● Login button can be clicked by mouse, touchpad, or pressing Enter on the keypad and appropriate functionality is observed.
Priority	Major

Table 11: Test case table 11

Test ID	Test 12
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Forgot Password button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Forgot Password button redirects to the Forgot Password page ● Check if the Forgot Password button is clickable by mouse ● Check if the Forgot Password button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Forgot Password button directs the user to the Forgot Password page ● User can click to Forgot Password button both by mouse or using the keyboard
Priority	Major

Table 12: Test case table 12

Test ID	Test 13
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Email Field on Forgot Password Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the email field by clicking on it ● Check if the user can type in the email field ● Check if the user can paste to the email field ● Check if the email entered by the user is already registered ● Check if the entered email is missing @ or domain ● Check if the error message is displayed when an invalid email is entered

Expected results	<ul style="list-style-type: none"> ● User can type/paste to the email field ● If the email entered by the user is valid, it should be accepted without any errors ● If the email entered by the user is not registered, the appropriate error message should be displayed ● If the email entered by the user is missing @ or domain, the appropriate error message should be displayed ● If the email entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 13: Test case table 13

Test ID	Test 14
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Submit Button on Forgot Password Page
Procedure	<ul style="list-style-type: none"> ● Check if the reset link is sent to the entered email address after clicking to the Submit button ● Check if Resend button appears after clicking to the Submit button ● Check if the Submit button is clickable by mouse ● Check if the Submit button is clickable by keyboard ● Check if the Submit button can be clicked by pressing Enter on keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Submit button should send a reset link to the entered email address ● Resend button should appear after clicking to the Submit button ● User can click to Submit button by mouse, using the keyboard or pressing enter
Priority	Major

Table 14: Test case table 14

Test ID	Test 15
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Resend Button on Forgot Password Page
	<ul style="list-style-type: none"> ● Check if the reset link is sent to the entered email address after clicking to the Resend button

Procedure	<ul style="list-style-type: none"> ● Check if the Resend button is clickable by mouse ● Check if the Resend button is clickable by keyboard ● Check if the Resend button can be clicked by pressing Enter on keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Resend button should send a reset link to the entered email address ● User can click to Resend button by mouse, using the keyboard or pressing enter
Priority	Major

Table 15: Test case table 15

Test ID	Test 16
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Add New Place Button on Home Page for Admins
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Add New Place button redirects to the page with the form for adding a new place ● Check if the Add New Place button is clickable by mouse ● Check if the Add New Place button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Add New Place button directs the user to the page with the form for adding a new place ● User can click to Add New Place button both by mouse or using the keyboard
Priority	Major

Table 16: Test case table 16

Test ID	Test 17
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Name Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the name field by clicking on it ● Check if the user can type in the name field ● Check if the user can paste to the name field ● Check if the user is allowed to type only alphabetical characters in the name field ● Check if the error message is displayed when an invalid name is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the name field

	<ul style="list-style-type: none"> ● If the name entered by the user is valid, it should be accepted without any errors ● If the name entered by the user includes characters that are not alphabetical, the appropriate error message should be displayed ● If the name entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 17: Test case table 17

Test ID	Test 18
Test Type	Functional Testing (Component Testing)
Title	Test Cases for City Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the city field by clicking on it ● Check if the user can type in the city field ● Check if the user can paste to the city field ● Check if the user is allowed to type only alphabetical characters in the city field ● Check if the error message is displayed when an invalid city is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the city field ● If the city entered by the user is valid, it should be accepted without any errors ● If the city entered by the user includes characters that are not alphabetical, the appropriate error message should be displayed ● If the city entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 18: Test case table 18

Test ID	Test 19
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Opening Hour Field on Form Page for Adding a New Place
	<ul style="list-style-type: none"> ● Check if the user can access the opening hour field by clicking on it

Procedure	<ul style="list-style-type: none"> ● Check if the user can type in the opening hour field ● Check if the user can paste to the opening hour field ● Check if the user is allowed to type only numeric characters in the opening hour field ● Check if the error message is displayed when an invalid opening hour is entered ● Check if the entered opening hour is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the opening hour field ● If the opening hour entered by the user is valid, it should be accepted without any errors ● If the opening hour entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the opening hour entered by the user is not in the specified range, the appropriate error message should be displayed ● If the opening hour entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 19: Test case table 19

Test ID	Test 20
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Closing Hour Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the closing hour field by clicking on it ● Check if the user can type in the closing hour field ● Check if the user can paste to the closing hour field ● Check if the user is allowed to type only numeric characters in the closing hour field ● Check if the error message is displayed when an invalid closing hour is entered ● Check if the entered closing hour is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the closing hour field ● If the closing hour entered by the user is valid, it should be accepted without any errors ● If the closing hour entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the closing hour entered by the user is not in the specified range, the appropriate error message should be displayed ● If the closing hour entered by the user is invalid, the

	appropriate error message should be displayed
Priority	Major

Table 20: Test case table 20

Test ID	Test 21
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Opening Minute Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the opening minute field by clicking on it ● Check if the user can type in the opening minute field ● Check if the user can paste to the opening minute field ● Check if the user is allowed to type only numeric characters in the opening minute field ● Check if the error message is displayed when an invalid opening minute is entered ● Check if the entered opening minute is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the opening minute field ● If the opening minute entered by the user is valid, it should be accepted without any errors ● If the opening minute entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the opening minute entered by the user is not in the specified range, the appropriate error message should be displayed ● If the opening minute entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 21: Test case table 21

Test ID	Test 22
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Closing Minute Field on Form Page for Adding a New Place
	<ul style="list-style-type: none"> ● Check if the user can access the closing minute field by clicking on it

Procedure	<ul style="list-style-type: none"> ● Check if the user can type in the closing minute field ● Check if the user can paste to the closing minute field ● Check if the user is allowed to type only numeric characters in the closing minute field ● Check if the error message is displayed when an invalid closing minute is entered ● Check if the entered closing minute is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the closing minute field ● If the closing minute entered by the user is valid, it should be accepted without any errors ● If the closing minute entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the closing minute entered by the user is not in the specified range, the appropriate error message should be displayed ● If the closing hour entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 22: Test case table 22

Test ID	Test 23
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Average Time Spent Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the average time spent field by clicking on it ● Check if the user can type in the average time spent field ● Check if the user can paste to the average time spent field ● Check if the user is allowed to type only numeric characters in the average time spent field ● Check if the error message is displayed when an invalid average time spent value is entered ● Check if the entered average time spent value is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the average time spent field ● If the average time spent value entered by the user is valid, it should be accepted without any errors ● If the average time spent value entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the average time spent value entered by the user is not in the specified range, the appropriate error message should be

	<p>displayed</p> <ul style="list-style-type: none"> ● If the average time spent value entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 23: Test case table 23

Test ID	Test 24
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Latitude Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the latitude field by clicking on it ● Check if the user can type in the latitude field ● Check if the user can paste to the latitude field ● Check if the user is allowed to type only numeric characters in the latitude field ● Check if the error message is displayed when an invalid latitude is entered ● Check if the entered latitude is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the latitude field ● If the latitude entered by the user is valid, it should be accepted without any errors ● If the latitude entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the latitude entered by the user is not in the specified range, the appropriate error message should be displayed ● If the latitude entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 24: Test case table 24

Test ID	Test 25
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Longitude Field on Form Page for Adding a New Place
	<ul style="list-style-type: none"> ● Check if the user can access the longitude field by clicking on it ● Check if the user can type in the longitude field

Procedure	<ul style="list-style-type: none"> ● Check if the user can paste to the longitude field ● Check if the user is allowed to type only numeric characters in the longitude field ● Check if the error message is displayed when an invalid longitude is entered ● Check if the entered longitude is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the longitude field ● If the longitude entered by the user is valid, it should be accepted without any errors ● If the longitude entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the longitude entered by the user is not in the specified range, the appropriate error message should be displayed ● If the longitude entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 25: Test case table 25

Test ID	Test 26
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Average Rating Field on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the average rating field by clicking on it ● Check if the user can type in the average rating field ● Check if the user can paste to the average rating field ● Check if the user is allowed to type only numeric characters in the average rating field ● Check if the error message is displayed when an invalid average rating value is entered ● Check if the entered average rating value is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the average rating field ● If the average rating value entered by the user is valid, it should be accepted without any errors ● If the average rating value entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the average rating value entered by the user is not in the specified range, the appropriate error message should be displayed

	<ul style="list-style-type: none"> ● If the average rating value entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 26: Test case table 26

Test ID	Test 27
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Create Button on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Create button redirects to the Home page for admins ● Check if the Create button is clickable by mouse ● Check if the Create button is clickable by keyboard ● Check if appropriate error message is displayed when any problem exists with the fields of the form ● Check if the specified place is added to the list of existing places on the admin's Home page after clicking the Create button.
Expected results	<ul style="list-style-type: none"> ● Clicking to Create button directs the user to the Home page for admins ● User can click to Create button both by mouse or using the keyboard ● If there are invalid entries in any of the fields of the form, the appropriate error message should be displayed ● Specified place should be added to the list of existing places on the admin's Home page
Priority	Major

Table 27: Test case table 27

Test ID	Test 28
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Cancel Button on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Cancel button redirects to the Home page for admins ● Check if the Cancel button is clickable by mouse ● Check if the Cancel button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Cancel button directs the user to the Home page for admins

	<ul style="list-style-type: none"> User can click to Cancel button both by mouse or using the keyboard
Priority	Major

Table 28: Test case table 28

Test ID	Test 29
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Place Objects on Home Page for Admins
Procedure	<ul style="list-style-type: none"> Check if clicking to the place objects redirects to the page for editing the place information Check if the place objects are clickable by mouse Check if the place objects are clickable by keyboard
Expected results	<ul style="list-style-type: none"> Clicking to place objects direct the user to the page for editing the place information User can click to place objects both by mouse or using the keyboard
Priority	Major

Table 29: Test case table 29

Test ID	Test 30
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Name Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> Check if the user can access the name field by clicking on it Check if the user can type in the name field Check if the user can paste to the name field Check if the error message is displayed when an invalid name is entered
Expected results	<ul style="list-style-type: none"> User can type/paste to the name field If the name entered by the user is valid, it should be accepted without any errors If the name entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 30: Test case table 30

Test ID	Test 31
Test Type	Functional Testing (Component Testing)
Title	Test Cases for City Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the city field by clicking on it ● Check if the user can type in the city field ● Check if the user can paste to the city field ● Check if the error message is displayed when an invalid city is entered
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the city field ● If the city entered by the user is valid, it should be accepted without any errors ● If the city entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 31: Test case table 31

Test ID	Test 32
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Opening Time Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the opening time field by clicking on it ● Check if the user can type in the opening time field ● Check if the user can paste to the opening time field ● Check if the error message is displayed when an invalid opening time is entered ● Check if the entered opening time is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the opening time field ● If the opening time entered by the user is valid, it should be accepted without any errors ● If the opening time entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the opening time entered by the user is not in the specified range, the appropriate error message should be displayed ● If the opening time entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 32: Test case table 32

Test ID	Test 33
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Closing Time Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the closing time field by clicking on it ● Check if the user can type in the closing time field ● Check if the user can paste to the closing time field ● Check if the error message is displayed when an invalid closing time is entered ● Check if the entered closing time is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the closing time field ● If the closing time entered by the user is valid, it should be accepted without any errors ● If the closing time entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the closing time entered by the user is not in the specified range, the appropriate error message should be displayed ● If the closing time entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 33: Test case table 33

Test ID	Test 34
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Average Rating Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the average rating field by clicking on it ● Check if the user can type in the average rating field ● Check if the user can paste to the average rating field ● Check if the user is allowed to type only numeric characters in the average rating field ● Check if the error message is displayed when an invalid average rating value is entered ● Check if the entered average rating value is in the specified

	range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the average rating field ● If the average rating value entered by the user is valid, it should be accepted without any errors ● If the average rating value entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the average rating value entered by the user is not in the specified range, the appropriate error message should be displayed ● If the average rating value entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 34: Test case table 34

Test ID	Test 35
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Latitude Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the latitude field by clicking on it ● Check if the user can type in the latitude field ● Check if the user can paste to the latitude field ● Check if the user is allowed to type only numeric characters in the latitude field ● Check if the error message is displayed when an invalid latitude is entered ● Check if the entered latitude is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the latitude field ● If the latitude entered by the user is valid, it should be accepted without any errors ● If the latitude entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the latitude entered by the user is not in the specified range, the appropriate error message should be displayed ● If the latitude entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 35: Test case table 35

Test ID	Test 36
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Longitude Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the longitude field by clicking on it ● Check if the user can type in the longitude field ● Check if the user can paste to the longitude field ● Check if the user is allowed to type only numeric characters in the longitude field ● Check if the error message is displayed when an invalid longitude is entered ● Check if the entered longitude is in the specified range
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the longitude field ● If the longitude entered by the user is valid, it should be accepted without any errors ● If the longitude entered by the user includes characters that are not numeric, the appropriate error message should be displayed ● If the longitude entered by the user is not in the specified range, the appropriate error message should be displayed ● If the longitude entered by the user is invalid, the appropriate error message should be displayed
Priority	Major

Table 36: Test case table 36

Test ID	Test 37
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Change Button on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Change button redirects to the Home page for admins ● Check if the Change button is clickable by mouse ● Check if the Change button is clickable by keyboard ● Check if appropriate error message is displayed when any problem exists with the fields of the form ● Check if the information of the specified place is updated on the list of existing places on the admin's Home page after clicking the Change button.
Expected results	<ul style="list-style-type: none"> ● Clicking to Change button directs the user to the Home page

	<p>for admins</p> <ul style="list-style-type: none"> ● User can click to Change button both by mouse or using the keyboard ● If there are invalid entries in any of the fields of the form, the appropriate error message should be displayed ● The information of the specified place should be updated on the list of existing places on the admin's Home page
Priority	Major

Table 37: Test case table 37

Test ID	Test 38
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Cancel Button on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Cancel button redirects to the Home page for admins ● Check if the Cancel button is clickable by mouse ● Check if the Cancel button is clickable by keyboard
Expected results	<ul style="list-style-type: none"> ● Clicking to Cancel button directs the user to the Home page for admins ● User can click to Cancel button both by mouse or using the keyboard
Priority	Major

Table 38: Test case table 38

Test ID	Test 39
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Home Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Home button redirects to the Home page that displays the current travel plans of the user ● Check if the Home button is clickable by mouse ● Check if the Home button is clickable by keyboard ● Check if the mouse hovers on the Home button
Expected results	<ul style="list-style-type: none"> ● Clicking to Create button directs the user to the Home page that displays the current travel plans of the user ● User can click to Home button both by mouse or using the

	keyboard <ul style="list-style-type: none"> • The mouse hovers on the Home button
Priority	Major

Table 39: Test case table 39

Test ID	Test 40
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Favorites Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> • Check if clicking to the Favorites button redirects to the page that lists the favorite travel plans of the user • Check if the Favorites button is clickable by mouse • Check if the Favorites button is clickable by keyboard • Check if the mouse hovers on the Favorites button
Expected results	<ul style="list-style-type: none"> • Clicking to Favorites button directs the user to the page that lists the favorite travel plans of the user • User can click to Favorites button both by mouse or using the keyboard • The mouse hovers on the Favorites button
Priority	Major

Table 40: Test case table 40

Test ID	Test 41
Test Type	Functional Testing (Integration)
Title	Test Cases for Favorite Plans (Addition & Removal)
Procedure	<ul style="list-style-type: none"> • Check if the user can click to star button using the mouse to add a plan to favorites • Check if the user can click to star button using the touchpad to add a plan to favorites • Check if the user can click to previously selected star button using the mouse to remove a plan from the favorites • Check if the user can click to previously selected star button using the mouse to remove a plan from the favorites • Check if the plan is added to the favorites database table when the unselected star button is clicked • Check if the plan is removed from the favorites database table when the unselected star button is clicked • Check if the plan is added to the Favorites page when the

	<ul style="list-style-type: none"> unselected star button is clicked ● Check if the plan is removed from the Favorites page when the unselected star button is clicked ● Check if the user is redirected to Homepage after changing the favorites status of the plan
Expected results	<ul style="list-style-type: none"> ● The user can click to the star button near the plan using the mouse or touchpad to add a plan to favorites ● If a plan is previously selected as favorite by the user, the user can click to star button once more to remove it from the favorites list using the mouse or touchpad ● The plan would be inserted into the favorites database table and would be displayed in the Favorites page of the application if it is selected as a favorite plan by clicking the associated star button. ● The plan would be removed from the favorites database table and would not be displayed in the Favorites page of the application if the star button is clicked on a plan which is previously added to favorites ● The user is redirected to Homepage after changing the favorites status of the plan
Priority	Major

Table 41: Test case table 41

Test ID	Test 42
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for History Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the History button redirects to the page that lists the previous travel plans of the user ● Check if the History button is clickable by mouse ● Check if the History button is clickable by keyboard ● Check if the mouse hovers on the History button
Expected results	<ul style="list-style-type: none"> ● Clicking to History button directs the user to the page that lists the previous travel plans of the user ● User can click to History button both by mouse or using the keyboard ● The mouse hovers on the History button
Priority	Major

Table 42: Test case table 42

Test ID	Test 43
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Notifications Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Notifications button redirects to the page that displays the notification Wall of the user ● Check if the Notifications button is clickable by mouse ● Check if the Notifications button is clickable by keyboard ● Check if the mouse hovers on the Notifications button
Expected results	<ul style="list-style-type: none"> ● Clicking to Notifications button directs the user to the page that displays the notification Wall of the user ● User can click to Notifications button both by mouse or using the keyboard ● The mouse hovers on the Notifications button
Priority	Major

Table 43: Test case table 43

Test ID	Test 44
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Suggestions Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Suggestions button redirects to the page that lists the recommended travel plans ● Check if the Suggestions button is clickable by mouse ● Check if the Suggestions button is clickable by keyboard ● Check if the mouse hovers on the Suggestions button
Expected results	<ul style="list-style-type: none"> ● Clicking to Suggestions button directs the user to the page that lists the recommended travel plans ● User can click to Suggestions button both by mouse or using the keyboard ● The mouse hovers on the Suggestions button
Priority	Major

Table 44: Test case table 44

Test ID	Test 45
Test Type	Functional Testing (Integration Testing)

Title	Test Cases for Settings Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Settings button redirects to the Settings page of the user ● Check if the Settings button is clickable by mouse ● Check if the Settings button is clickable by keyboard ● Check if the mouse hovers on the Settings button
Expected results	<ul style="list-style-type: none"> ● Clicking to Settings button directs the user to the Settings page of the user ● User can click to Settings button both by mouse or using the keyboard ● The mouse hovers on the Settings button
Priority	Major

Table 45: Test case table 45

Test ID	Test 46
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Profile Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Profile button redirects to the Profile page of the user ● Check if the Profile button is clickable by mouse ● Check if the Profile button is clickable by keyboard ● Check if the mouse hovers on the Profile button
Expected results	<ul style="list-style-type: none"> ● Clicking to Profile button directs the user to the Profile page of the user ● User can click to Profile button both by mouse or using the keyboard ● The mouse hovers on the Profile button
Priority	Major

Table 46: Test case table 46

Test ID	Test 47
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Change Profile Photo on Profile Page
	<ul style="list-style-type: none"> ● Check if local folders can be observed for photo selection by clicking to Change button ● Check if the photo can be selected from appropriate folder by

<p>Procedure</p>	<p>clicking on it by mouse</p> <ul style="list-style-type: none"> ● Check if the photo can be selected from appropriate folder by clicking on it by keyboard ● Check if the photo can be selected from appropriate folder by selecting it using the touchpad ● Check if more than one photos can be selected at the same time for photo selection ● Check if various sized photos can be selected by the user ● Check if Change Profile Photo button is clickable by mouse ● Check if Change Profile Photo is clickable by keyboard ● Check if appropriate message is displayed after selecting the photo from the folders ● Check if Select button can be clicked without selecting any photos ● Check if .JPEG or .PNG typed photos are accepted ● Check if appropriate message is displayed or it is accepted if another type of file is selected except a .jpeg or .png ● Check if profile photo is updated after selecting the photo ● Check if the selected photo enables an update at the associated field in the database ● Check if the user is directed back to Profile Page after selecting the photo from the file explorer
<p>Expected results</p>	<ul style="list-style-type: none"> ● The user can observe the local folders after clicking the Change button and can select the desired photo from the appropriate folder by clicking with mouse, keyboard or touchpad. ● Just one photo can be selected at a time, if more than one photo is selected, an error message will be displayed. ● Various sized photos can be selected by the user, which can be resized by the system to visualize profile photo ● After selection of photos and successful completion of photo selection, a message will be displayed. ● The select button cannot be pressed without selecting any photos from the folders ● .JPEG and .PNG type photos are accepted for selection ● Profile photo is updated in the database and Profile page after selection of the photo from the local folders
<p>Priority</p>	<p>Major</p>

Table 47: Test case table 47

<p>Test ID</p>	<p>Test 48</p>
<p>Test Type</p>	<p>Functional Testing (Component Testing)</p>
<p>Title</p>	<p>Test Cases for Username and Full Name display on Profile Page</p>

Procedure	<ul style="list-style-type: none"> ● Check if the username field corresponds to the username determined by the user in the SignUp process ● Check if the name field corresponds to the full name determined by the user in the SignUp process ● Check if the username and full name are correctly displayed for various lengths ● Check if the username can be correctly displayed if it contains numbers or non letter characters
Expected results	<ul style="list-style-type: none"> ● Username displayed in the profile page is the same one with the username given during sign up. ● Full name displayed in the profile page is the same one with the full name given during sign up.
Priority	Minor

Table 48: Test case table 48

Test ID	Test 49
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Create New Plan Button on Home Page
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Create New Plan button redirects to the page that displays the form for creating a new plan ● Check if the Create New Plan button is clickable by mouse ● Check if the Create New Plan button is clickable by keyboard ● Check if the mouse hovers on the Create New Plan button
Expected results	<ul style="list-style-type: none"> ● Clicking to Create New Plan button directs the user to the Settings page of the user ● User can click to Create New Plan button both by mouse or using the keyboard ● The mouse hovers on the Create New Plan button
Priority	Major

Table 49: Test case table 49

Test ID	Test 50
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Plan Name Field on Create Plan Page
	<ul style="list-style-type: none"> ● Check if the user can access the plan name field by clicking on it ● Check if the user can type in the plan name field

Procedure	<ul style="list-style-type: none"> ● Check if the user can paste to the plan name field ● Check if the user is allowed to type both alphabetical and numerical characters in the name field ● Check if the error message is displayed when the plan name exceeds 50 characters ● Check if error message is displayed when the user has a past travel plan with same name
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the name field ● If the name entered by the user is valid, it should be accepted without any errors ● If the name entered by the user exceeds the 50 character limit, the appropriate error message should be displayed ● If the name entered by the user is used before for a previous travel plan, the appropriate error message should be displayed ● Both numerical and alphabetical characters should be accepted in the name
Priority	Major

Table 50: Test case table 50

Test ID	Test 51
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Start Date Field on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the calendar for date selection or write on the field by clicking on it ● Check if the user can move between the months of the calendar by using the arrows at the top of the calendar ● Check if the user can move between the days by using forward and back arrow buttons on the keyboard ● Check if the user can select dates earlier than the plan creation day ● Check if the selected date in the calendar is displayed correctly on the field after clicking the date on the calendar ● Check if the user can type the date in the Start Date field in the form of 01/01/2023 instead of selecting from the calendar ● Check if the user can paste to the start date field in the appropriate form ● Check if the user is allowed to type alphabetical characters in the date field ● Check if the error message is displayed when the date is entered in an invalid format ● Check if the error message is displayed when the user enters a previous date in correct format

Expected results	<ul style="list-style-type: none"> ● The user can access the calendar for date selection by clicking on the field or he/she can write on the field in the form of 01/01/2023 ● At the calendar, the dates before plan creation day are not allowed to be selected and the user can move between months to create future plans ● When the user enters the date in incorrect format or writes a date before the current date, appropriate error messages are displayed accordingly. ● The user can move between the days at the calendar by pressing forward and back arrows on the keyboard ● User can type/paste on the start date field ● User cannot enter alphabetical characters on the start date field
Priority	Major

Table 51: Test case table 51

Test ID	Test 52
Test Type	Functional Testing (Component Testing)
Title	Test Cases for End Date Field on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the calendar for date selection or write on the field by clicking on it ● Check if the user can move between the months of the calendar by using the arrows at the top of the calendar ● Check if the user can move between the days by using forward and back arrow buttons on the keyboard ● Check if the user can select dates earlier than the plan creation day ● Check if the user can select dates earlier than the plan start date if it is selected prior to end date ● Check if the user can select dates that would exceed maximum plan length if start date is selected previously ● Check if the selected date in the calendar is displayed correctly on the field after clicking the date on the calendar ● Check if the user can type the date in the End Date field in the form of 01/01/2023 instead of selecting from the calendar ● Check if the user can paste to the end date field in the appropriate form ● Check if the user is allowed to type alphabetical characters in the date field ● Check if the error message is displayed when the date is entered in an invalid format ● Check if the error message is displayed when the user enters a previous date in correct format

	<ul style="list-style-type: none"> ● Check if the error message is displayed when the user enters a prior date to the plan start date after start date is determined ● Check if the error message is displayed when the plan length exceeds the maximum length for plan creation
Expected results	<ul style="list-style-type: none"> ● The user can access the calendar for date selection by clicking on the field or he/she can write on the field in the form of 01/01/2023 ● At the calendar, the dates before plan creation day are not allowed to be selected and the user can move between months to create future plans ● At the calendar, if the start date is firstly selected, the dates before the start date are not allowed to be selected ● At the calendar, if the start date is firstly selected, the dates which result in a longer plan than maximum plan length are not allowed to be selected. ● When the user enters the date in incorrect format or the written dates are against the conditions indicated above, appropriate error messages are displayed accordingly. ● The user can move between the days at the calendar by pressing forward and back arrows on the keyboard ● User can type/paste on the end date field ● User cannot enter alphabetical characters on the end date field
Priority	Major

Table 52: Test case table 52

Test ID	Test 53
Test Type	Functional Testing (Component Testing)
Title	Test Cases for City Name Field on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the city name field by clicking on it ● Check if the user can type in the city name field ● Check if the user can paste to the city name field ● Check if the user is allowed to type only alphabetical characters in the city name field ● Check if the error message is displayed when an invalid city name is entered ● Check if the possible city lists are displayed below the field according to entered characters ● Check if the user can select the city name from the list displayed below by clicking on the city name or using the touchpad ● Check if the user can move between the city names displayed in the list by using arrows in the keyboard

	<ul style="list-style-type: none"> ● Check if the user can select the city name where the selection box is on by pressing the Enter button
Expected results	<ul style="list-style-type: none"> ● User can type/paste to the city name field ● If the city name entered by the user is valid, it should be accepted without any errors ● User should not be allowed to enter numerical characters on the city name field ● If the city name entered to the field is invalid, appropriate error message should be displayed ● When the user starts entering characters, the possible city options starting with entered characters should be presented in a list below the field. ● The user can move the selection box between the city names at the list by pressing “up-bottom” or “back-forward” arrows on the keyboard. ● The user can select the city name from the list below by clicking on it, selecting it with touchpad, or pressing Enter when the selection box is on it.
Priority	Major

Table 53: Test case table 53

Test ID	Test 54
Test Type	Functional Testing (Component Testing)
Title	Test Cases for “Move Out Time” Field on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the “Move Out Time” field by clicking on it ● Check if the user can type in the “Move Out Time” field ● Check if the user can paste to the “Move Out Time” field ● Check if the user is allowed to type alphabetical characters on the “Move Out Time” field ● Check if the time selection panel appears below the field after clicking on it ● Check if the user can move between hours and minutes by using the arrows on the keyboard ● Check if the user can select the time by pressing Enter when the selection box is on some time (hours-minutes) ● Check if the user can type the “Move Out Time” in the format like 23:30 instead of selecting from the time selection panel. ● Check if the error message is displayed when the time is entered in an invalid format ● Check if a message is displayed when the time is not meaningful (like before 7 am and after 8 pm) ● Check if the error message is displayed when the entered time

	is invalid (hours greater than 24 or minutes greater than 60)
Expected results	<ul style="list-style-type: none"> ● The user can access the “Move Out Time” field by clicking on it and the time selection panel also appears which would allow more practical selection ● The user can type/paste the move out time on the field ● The user is not allowed to write alphabetical characters on that field and error message will be displayed accordingly ● In the time selection panel, the user can move between hours and minutes using the arrows in the keyboard, and complete selection by clicking on hour/minutes or moving selection box to intended time and pressing Enter at the end ● Appropriate error message is displayed if the time is entered in a non-acceptable format or an impossible time is entered (like hours greater than 24 or minutes greater than 60) ● A message will be displayed about the difficulty of traveling if the “Move Out Time” is entered before 7 am or later than 20 am.
Priority	Major

Table 54: Test case table 54

Test ID	Test 55
Test Type	Functional Testing (Component Testing)
Title	Test Cases for “Come Back Time” Field on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can access the “Come Back Time” field by clicking on it ● Check if the user can type in the “Come Back Time” field ● Check if the user can paste to the “Come Back Time” field ● Check if the user is allowed to type alphabetical characters on the “Come Back Time” field ● Check if the time selection panel appears below the field after clicking on it ● Check if the user can move between hours and minutes by using the arrows on the keyboard ● Check if the user can select the time by pressing Enter when the selection box is on some time (hours-minutes) ● Check if the user can type the “Come Back Time” in the format like 23:30 instead of selecting from the time selection panel. ● Check if the error message is displayed when the time is entered in an invalid format ● Check if a message is displayed when the time is not meaningful (like before 11 am and after 3 am) ● Check if the error message is displayed when the entered time

	<p>is invalid (hours greater than 24 or minutes greater than 60)</p> <ul style="list-style-type: none"> ● Check if the error message is displayed when “Come Back Time” is before “Move Out Time”
Expected results	<ul style="list-style-type: none"> ● The user can access the “Come Back Time” field by clicking on it and the time selection panel also appears which would allow more practical selection ● The user can type/paste the come back time on the field ● The user is not allowed to write alphabetical characters on that field and error message will be displayed accordingly ● In the time selection panel, the user can move between hours and minutes using the arrows in the keyboard, and complete selection by clicking on hour/minutes or moving selection box to intended time and pressing Enter at the end ● Appropriate error message is displayed if the time is entered in a non-acceptable format or an impossible time is entered (like hours greater than 24 or minutes greater than 60) ● A message will be displayed about the difficulty of traveling if the “Come Back Time” is entered before 11 am or later than 3 am. ● An error message would be displayed if “Come Back Time” is before the entered “Move Out Time”
Priority	Major

Table 55: Test case table 55

Test ID	Test 56
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Attraction Type Buttons on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can select the attraction types of interest by clicking the box to the left of it by using a mouse or touchpad. ● Check if the user can pass this part without selecting any attractions ● Check if the user can select multiple types of attractions in this part ● Check if the user can move between the checkboxes using the arrows in the keyboard ● Check if clicking to a selected box using mouse or touchpad makes it unselected
Expected results	<ul style="list-style-type: none"> ● User can select the attraction types of interest by clicking the box on the left of its name using the mouse or touchpad ● The user can complete plan creation without selecting any attraction type or selecting multiple number of attractions ● The user can move between the selection boxes using the

	<p>“right-left” arrows in the keyboard.</p> <ul style="list-style-type: none"> ● Clicking to a selected attraction type box using mouse or touchpad makes it unselected
Priority	Major

Table 56: Test case table 56

Test ID	Test 57
Test Type	Functional Testing (Component Testing)
Title	Test Cases for Food Type Buttons on Create Plan Page
Procedure	<ul style="list-style-type: none"> ● Check if the user can select the food types of interest by clicking the box to the left of it by using a mouse or touchpad. ● Check if the user can pass this part without selecting any food types ● Check if the user can select multiple types of foods in this part ● Check if the user can move between the checkboxes using the arrows in the keyboard ● Check if clicking to a selected box using mouse or touchpad makes it unselected
Expected results	<ul style="list-style-type: none"> ● User can select the food types of interest by clicking the box on the left of its name using the mouse or touchpad ● The user can complete plan creation without selecting any food type or selecting multiple number of food ● The user can move between the selection boxes using the “right-left” arrows in the keyboard. ● Clicking to a selected food type box using mouse or touchpad makes it unselected
Priority	Major

Table 57: Test case table 57

Test ID	Test 58
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Create Plan Button
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Create button redirects to the Plans page after creating the travel plan with specified constraints ● Check if the Create button is clickable by mouse ● Check if the Create button is clickable by keyboard

	<ul style="list-style-type: none"> ● Check if appropriate error message is displayed when any problem exists with the fields of the plan creation form ● Check if the specified plan is added to the list of existing plans on the user's My Plans page after clicking the Create button.
Expected results	<ul style="list-style-type: none"> ● Clicking to Create button directs the user to the My Plans page of the user ● User can click to Create button both by mouse or using the keyboard ● If there are invalid entries in any of the fields of the form, the appropriate error message should be displayed ● Specified place should be added to the list of existing plans on the user's My Plans page
Priority	Major

Table 58: Test case table 58

Test ID	Test 59
Test Type	Functional Testing (Integration)
Title	Test Cases for Delete Plan Button
Procedure	<ul style="list-style-type: none"> ● Check if the user can click delete plan button using the mouse ● Check if the user can click delete plan button using the touchpad ● Check if the plan is deleted from My Plans list when the button is clicked ● Check if the plan is removed from the database when the button is clicked ● Check if the user is redirected to My Plans page after clicking to delete button ● Check if the user is asked in a panel about approving this deletion request ● Check if the deletion of the plan from My Plans list takes place after clicking to "Yes" button in the approval panel ● Check if the plan is removed from database when "Yes" button is clicked at the approval panel ● Check if "Yes" and "No" in the approval panel are clickable by mouse ● Check if "Yes" and "No" in the approval panel are clickable by touchpad ● Check if the deletion of the plan from My Plans list takes place after clicking to "No" button in the approval panel ● Check if the plan is removed from database when "No" button is clicked at the approval panel

Expected results	<ul style="list-style-type: none"> ● The user can click delete plan button using the mouse or touchpad to see the approval panel ● No operations take place in database and My Plans list before clicking to “Yes” in approval panel ● User is redirected back to My Plans page after deleting the plan to see the new list of his/her plans ● Clicking the “Yes” button should result in removal of the plan instance from the list of the user’s plans in the database and it should not be displayed in the My Plans list of the user. ● “Yes” and “No” buttons on the approval panel for plan deletion requests are clickable by mouse and touchpad.
Priority	Major

Table 59: Test case table 59

Test ID	Test 60
Test Type	Functional Testing (Integration Testing)
Title	Test Cases for Logout button
Procedure	<ul style="list-style-type: none"> ● Check if clicking to the Logout button redirects to the Login page ● Check if this button can be clicked at the Home page ● Check if this button can be clicked at the Profile page ● Check if this button can be clicked at Favorites page ● Check if this button can be clicked at the History page ● Check if this button can be clicked at the Suggestions page ● Check if this button can be clicked at the Notifications page ● Check if this button can be clicked at the Settings page ● Check if the user remains logged in to the website after clicking logout button and then opening website from a new window ● Check if the Login button is clickable by mouse ● Check if the Login button is clickable by keyboard ● Check if appropriate message is displayed after clicking to Logout button
Expected results	<ul style="list-style-type: none"> ● Clicking to Logout button directs the user to the Login page ● Logout button can be clicked while the user is at any of the pages ● User cannot access his/her account from another window from the same browser after logging out from one of the windows ● User can click to Logout button both by mouse or using the keyboard ● Appropriate message is displayed to the user indicating that the user has been logged out from the system

Priority	Major
-----------------	-------

Table 60: Test case table 60

5.2 Non Functional Test Cases

Test ID	Test 61
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Signup Button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the Signup button is aligned properly and is readable on all browsers ● Check if the Signup button has a different color (bold) than other fields to take the user's interest ● Check if the Signup button is aligned on the top of other items to be filled
Expected results	<ul style="list-style-type: none"> ● SignUp button at the Login page is on the top of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Signup button is aligned the same and readable on different browsers.
Priority	Minor

Table 61: Test case table 61

Test ID	Test 62
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Fields on Signup Page
Procedure	<ul style="list-style-type: none"> ● Check if the full name field is aligned properly on all browsers ● Check if the username field is aligned properly on all browsers ● Check if the email field is aligned properly on all browsers ● Check if the password field is aligned properly on all browsers ● Check if the confirm password field is aligned properly on all browsers ● Check if the fields have lower opacity in terms of color in

	order to display that they are typeable
Expected results	<ul style="list-style-type: none"> • The fields of the Signup page are aligned properly on all browsers • The fields have lower opacity in order to display that they are typeable
Priority	Minor

Table 62: Test case table 62

Test ID	Test 63
Test Type	Nonfunctional Testing (Usability & Security & Performance)
Title	Test Cases for Signup Button on Signup Page
Procedure	<p>Usability</p> <ul style="list-style-type: none"> • Check if the Signup button is aligned properly on all browsers • Check if the Signup button is in larger punto than other fields, and written in uppercase letters to take the user’s interest • Check if the Signup button is located below all the fields to be filled <p>Security</p> <ul style="list-style-type: none"> • Check if the password of the user is encrypted before adding to the database • Check if the account details of the user like username and password are stored in cache or not <p>Performance</p> <ul style="list-style-type: none"> • Check if the user is added to “Pending User” database in a few seconds after clicking to SignUp button
Expected results	<ul style="list-style-type: none"> • The Signup button is larger than other fields in that page, written uppercase and is readable • Signup button is aligned properly on all browsers at the bottom of the fields to be filled for registration • After adding the user to “Pending User” database, the password is not stored in database as it is, instead it is in the encrypted form • The account details of the user, which are taken in that page are not stored in cache after clicking to SignUp button • The user is added to “Pending User” database in a few seconds after clicking to Signup button before completing the email verification
Priority	Critical

Table 63: Test case table 63

Test ID	Test 64
Test Type	Nonfunctional Testing (Usability & Performance)
Title	Test Cases for Email Verification for SignUp
Procedure	<p>Usability</p> <ul style="list-style-type: none"> ● Check if the mail sent for email verification is clear and provides an appropriate description to the user. ● Check if the provided link in the mail directs the user to Login page after completing the verification of the email <p>Performance</p> <ul style="list-style-type: none"> ● The verification email is sent to the provided email address in at most 5 minutes after completion of the SignUp process ● The verification of the email and addition of the user to real database must be completed in a few seconds after clicking to verification link
Expected results	<ul style="list-style-type: none"> ● Email sent to the user has clear explanations, which would direct the user to click to the verification link. ● After clicking to the link in the email, the user's email verification would be completed and directed to Login page ● The verification email is sent to the user in a few minutes (at most 5) after completion of SignUp steps. ● The user would be directly added to database and verification will be completed in a few seconds after clicking to verification link
Priority	Major

Table 64: Test case table 64

Test ID	Test 65
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Login button on Signup page
Procedure	<ul style="list-style-type: none"> ● Check if the Login button is aligned properly on all browsers ● Check if the Login button is placed at the top of other fields to ensure user can realize it firstly ● Check if the Login button is readable and written in different colors than other fields
Expected results	<ul style="list-style-type: none"> ● The Login button is different than other fields in the page with its larger size ● The Login button is placed at the top of all the fields to be filled for sign up and has proper alignment on all browsers
Priority	Minor

Table 65: Test case table 65

Test ID	Test 66
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Fields on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the username field is aligned properly on all browsers ● Check if the password field is aligned properly on all browsers ● Check if the fields have lower opacity in terms of color in order to display that they are typeable
Expected results	<ul style="list-style-type: none"> ● Username field and password field are aligned properly on all browsers ● The fields have lower opacity in order to display that they are typeable
Priority	Minor

Table 66: Test case table 66

Test ID	Test 67
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Login Button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the Login button is aligned properly and is readable on all browsers ● Check if the Login button is greater than all other buttons and written uppercase to be catchy ● Check if the Login button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Login button at the Login page is at the bottom of all other fields and has a greater dimension than other fields/buttons on the page and written uppercase to take user's interests ● Login button is aligned the same and readable on different browsers.
Priority	Minor

Table 67: Test case table 67

Test ID	Test 68
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Forgot Password button on Login Page
Procedure	<ul style="list-style-type: none"> ● Check if the Forgot Password button is aligned properly and is readable on all browsers ● Check if the Forgot Password button has a different color (bold) than other fields to take the user's interest ● Check if the Forgot Password button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Forgot Password button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Forgot Password button is aligned the same and readable on different browsers.
Priority	Minor

Table 68: Test case table 68

Test ID	Test 69
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Resend Button on Forgot Password Page
Procedure	<ul style="list-style-type: none"> ● Check if the Resend button is aligned properly and is readable on all browsers ● Check if the Resend button has a different color (bold) than other fields to take the user's interest ● Check if the Resend button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Resend button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Resend button is aligned the same and readable on different browsers.
Priority	Minor

Table 69: Test case table 69

Test ID	Test 70
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Submit Button on Forgot Password Page
Procedure	<ul style="list-style-type: none"> ● Check if the Submit button is aligned properly and is readable on all browsers ● Check if the Submit button has a different color (bold) than other fields to take the user's interest ● Check if the Submit button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Submit button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Submit button is aligned the same and readable on different browsers.
Priority	Minor

Table 70: Test case table 70

Test ID	Test 71
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Email Field on Forgot Password Page
Procedure	<ul style="list-style-type: none"> ● Check if the email field is aligned properly on all browsers ● Check if the fields have lower opacity in terms of color in order to display that they are typeable
Expected results	<ul style="list-style-type: none"> ● Email field is aligned properly on all browsers ● Email field has lower opacity in order to display that they are typeable
Priority	Minor

Table 71: Test case table 71

Test ID	Test 72
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Profile Button on Navigation Bar
	<ul style="list-style-type: none"> ● Check if the Profile button is aligned properly and is readable on all browsers

Procedure	<ul style="list-style-type: none"> ● Check if the Profile button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Profile button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Profile button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Profile button is aligned properly and is readable on different browsers.
Priority	Minor

Table 72: Test case table 72

Test ID	Test 73
Test Type	Nonfunctional Testing (Usability & Security & Performance)
Title	Test Cases for Change Profile Photo on Profile Page
Procedure	<p>Usability</p> <ul style="list-style-type: none"> ● Check if the Change button is aligned properly on all browsers ● Check if the profile photo is displayed appropriately on all browsers ● Check if profile photo is scaled proportionally to the actually selected photo <p>Security</p> <ul style="list-style-type: none"> ● Check if all the photos in the local folders of the user is saved in any databases of the application ● Check if the photos in the local folders of the user are stored in cache <p>Performance</p> <ul style="list-style-type: none"> ● Check if the profile photo in the Profile page of the user is directly updated in a few seconds after selecting the photo from local directories. ● Check if the profile photo is added to the appropriate field in the database in a few seconds after selecting the photo from local directories.
Expected results	<ul style="list-style-type: none"> ● Change button is displayed properly in all browsers by being aligned to the profile photo and located below it ● Profile photo is displayed clearly as selected by the user and has a proportional scaling to the actual photo ● The photos in the local folders of the user should not be stored in any databases of the application or the cache to ensure the protection of privacy ● The profile photo should be updated at database and Profile page in at most 5 seconds after selecting the associated photo from the local directories

Priority	Critical
-----------------	----------

Table 73: Test case table 73

Test ID	Test 74
Test Type	Nonfunctional Testing (Usability Testing)
Title	Test Cases for Username and Full Name display on Profile Page
Procedure	<ul style="list-style-type: none"> ● Check if the Username field on the profile page is aligned properly on all browsers ● Check if the Full Name field on the profile page is aligned properly on all browsers ● Check if the Username field on the profile page is readable with appropriate fonts and colors ● Check if the Full Name field on the profile page is readable with appropriate fonts and colors
Expected results	<ul style="list-style-type: none"> ● Username field is aligned to the profile photo and located below it. This fashion remains the same in different browsers ● Name field is aligned to the profile photo and located below it. This fashion remains the same in different browsers ● Username field is easily readable with suitable tones of blue and adequate fonts ● Full name field is easily readable with suitable tones of blue and adequate fonts
Priority	Minor

Table 74: Test case table 74

Test ID	Test 75
Test Type	Nonfunctional Testing (Usability & Performance)
Title	Test Cases for Create Plan Button
Procedure	<p>Usability</p> <ul style="list-style-type: none"> ● Check if the Create button is aligned properly at the bottom of the fields to be filled on all browsers ● Check if Create button has greater punto than other fields to be more effective <p>Performance</p> <ul style="list-style-type: none"> ● Check if the plan is created at most in 1-2 minutes after clicking to Create button for the plan creation request ● Check if the created plan exists on My Plans page just after plan creation, which is at most 2 minutes after clicking to

	Create button
Expected results	<ul style="list-style-type: none"> ● Create button has larger dimensions than other fields in the plan creation page and is aligned properly at the bottom of other fields on all browsers. ● Travel plan is created at most in 1-2 minutes after clicking to Create button for the plan creation request ● The travel plan appears on My Plans page just after the creation of it.
Priority	Major

Table 75: Test case table 75

Test ID	Test 76
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Fields on Plan Creation Form
Procedure	<ul style="list-style-type: none"> ● Check if the food type selection boxes are properly aligned on all browsers ● Check if the food type selection boxes are large enough for easily selecting them with mouse ● Check if the attraction type selection boxes are properly aligned on all browsers ● Check if the attraction type selection boxes are large enough for easily selecting them with mouse ● Check if the “Move Out Time” field is properly aligned on all browsers ● Check if the time selection panel for “Move Out Time” is flexible for usage by being aligned below the field. ● Check if the “Come Back Time” field is properly aligned on all browsers ● Check if the time selection panel for “Come Back Time” is flexible for usage by being aligned below the field. ● Check if the city selection field is properly aligned on all browsers ● Check if the city selection options for the field are clearly displayed for usage by being aligned below the field. ● Check if start date field is aligned properly on all browsers ● Check if the calendar appearing in start date selection part is easy for usage by being aligned above the date field ● Check if end date field is aligned properly on all browsers ● Check if the calendar appearing in end date selection part is easy for usage by being aligned above the date field ● Check if the plan name field is aligned properly on all browsers

Expected results	<ul style="list-style-type: none"> • All the fields and selection boxes in the plan creation form are aligned at the same column properly and are readable in different browsers. • The calendars for start and end date selection are aligned on the top of date fields without blocking any other fields for more usable presentation. • The time selection boxes are aligned below the “Move Out” and “Come Back” time fields. • The options for city name are aligned below the city name field and has a scrollbar on the right side for easily moving between city names
Priority	Minor

Table 76: Test case table 76

Test ID	Test 77
Test Type	Nonfunctional Testing (Performance & Usability)
Title	Test Cases for Delete Plan Button
Procedure	<p>Performance</p> <ul style="list-style-type: none"> • Check if the plan is removed from associated database table in a few seconds after approving the Delete Plan request • Check if the plan appears in My Plans list in a few seconds after creating the plan deletion request <p>Usability</p> <ul style="list-style-type: none"> • Check if the Delete Plan button is on the right corner of the plan header and aligned properly on all browsers • Check if the approval panel seems on the top of Delete Plan button and is aligned properly on all browsers • Check if the “Yes” and “No” buttons on the approval panel are easily readable and have green and red colors respectively • Check if the “Yes” and “No” buttons on the approval panel are aligned properly on all browsers.
Expected results	<ul style="list-style-type: none"> • The plan is removed from the database table and it does not appear in My Plans list in a few seconds after creating the plan deletion request • Delete Plan button is on the right side of plan header and aligned properly on different browsers • Approval panel is located at the top of “Delete Plan” button and it is aligned properly on different browsers along with the “Yes” and “No” buttons on that page
Priority	Major

Table 77: Test case table 77

Test ID	Test 78
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Home Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if the Home button is aligned properly and is readable on all browsers ● Check if the Home button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Home button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Home button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Home button is aligned properly and is readable on different browsers.
Priority	Minor

Table 78: Test case table 78

Test ID	Test 79
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for History Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if the History button is aligned properly and is readable on all browsers ● Check if the History button has the same color with the other buttons on the navigation bar to create harmony ● Check if the History button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● History button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● History button is aligned properly and is readable on different browsers.
Priority	Minor

Table 79: Test case table 79

Test ID	Test 80
Test Type	Nonfunctional Testing (Usability)

Title	Test Cases for Suggestions Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if the Suggestions button is aligned properly and is readable on all browsers ● Check if the Suggestions button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Suggestions button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Suggestions button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Suggestions button is aligned properly and is readable on different browsers.
Priority	Minor

Table 80: Test case table 80

Test ID	Test 81
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Settings Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if the Settings button is aligned properly and is readable on all browsers ● Check if the Settings button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Settings button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Settings button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Settings button is aligned properly and is readable on different browsers.
Priority	Minor

Table 81: Test case table 81

Test ID	Test 82
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Notifications Button on Navigation Bar
	<ul style="list-style-type: none"> ● Check if the Notifications button is aligned properly and is readable on all browsers

Procedure	<ul style="list-style-type: none"> ● Check if the Notifications button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Notifications button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Notifications button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Notifications button is aligned properly and is readable on different browsers.
Priority	Minor

Table 82: Test case table 82

Test ID	Test 83
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Cancel Button on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the Cancel button is aligned properly and is readable on all browsers ● Check if the Cancel button has a different color (bold) than other fields to take the user's interest ● Check if the Cancel button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Cancel button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Cancel button is aligned the same and readable on different browsers.
Priority	Minor

Table 83: Test case table 83

Test ID	Test 84
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Change Button on Page for Editing the Place Information
	<ul style="list-style-type: none"> ● Check if the Change button is aligned properly and is readable on all browsers ● Check if the Change button has a different color (bold) than

Procedure	<p>other fields to take the user's interest</p> <ul style="list-style-type: none"> ● Check if the Change button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Change button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Change button is aligned the same and readable on different browsers.
Priority	Minor

Table 84: Test case table 84

Test ID	Test 85
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Create Button on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the Create button is aligned properly and is readable on all browsers ● Check if the Create button has a different color (bold) than other fields to take the user's interest ● Check if the Create button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Create button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Create button is aligned the same and readable on different browsers.
Priority	Minor

Table 85: Test case table 85

Test ID	Test 86
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Cancel Button on Form Page for Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the Cancel button is aligned properly and is readable on all browsers ● Check if the Cancel button has a different color (bold) than other fields to take the user's interest ● Check if the Cancel button is aligned at the bottom of other items to be filled

Expected results	<ul style="list-style-type: none"> ● Cancel button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Cancel button is aligned the same and readable on different browsers.
Priority	Minor

Table 86: Test case table 86

Test ID	Test 87
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Add New Place Button on Home Page for Admins
Procedure	<ul style="list-style-type: none"> ● Check if the Add New Place button is aligned properly and is readable on all browsers ● Check if the Add New Place button has a different color (bold) than other fields to take the user's interest ● Check if the Add New Place button is aligned at the bottom of other items to be filled
Expected results	<ul style="list-style-type: none"> ● Add New Place button is at the bottom of all other fields and has a bolder color than other fields/buttons on the page to take user's interests ● Cancel button is aligned the same and readable on different browsers.
Priority	Minor

Table 87: Test case table 87

Test ID	Test 88
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Fields on Adding a New Place
Procedure	<ul style="list-style-type: none"> ● Check if the name field is aligned properly on all browsers ● Check if the city field is aligned properly on all browsers ● Check if the opening hour field is aligned properly on all browsers ● Check if the opening minute field is aligned properly on all browsers ● Check if the closing hour field is aligned properly on all browsers ● Check if the closing hour field is aligned properly on all

	<p>browsers</p> <ul style="list-style-type: none"> ● Check if the latitude field is aligned properly on all browsers ● Check if the longitude field is aligned properly on all browsers ● Check if the average time spent field is aligned properly on all browsers ● Check if the average rating field is aligned properly on all browsers ● Check if the fields have lower opacity in terms of color in order to display that they are typeable
Expected results	<ul style="list-style-type: none"> ● The fields of the Signup page are aligned properly on all browsers ● The fields have lower opacity in order to display that they are typeable
Priority	Minor

Table 88: Test case table 88

Test ID	Test 89
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Longitude Field on Page for Editing the Place Information
Procedure	<ul style="list-style-type: none"> ● Check if the name field is aligned properly on all browsers ● Check if the city field is aligned properly on all browsers ● Check if the opening time field is aligned properly on all browsers ● Check if the closing time field is aligned properly on all browsers ● Check if the latitude field is aligned properly on all browsers ● Check if the longitude field is aligned properly on all browsers ● Check if the average time spent field is aligned properly on all browsers ● Check if the average rating field is aligned properly on all browsers ● Check if the fields have lower opacity in terms of color in order to display that they are typeable
Expected results	<ul style="list-style-type: none"> ● The fields of the Signup page are aligned properly on all browsers ● The fields have lower opacity in order to display that they are typeable
Priority	Minor

Table 89: Test case table 89

Test ID	Test 90
Test Type	Nonfunctional Testing (Usability)
Title	Test Cases for Favorites Button on Navigation Bar
Procedure	<ul style="list-style-type: none"> ● Check if the Favorites button is aligned properly and is readable on all browsers ● Check if the Favorites button has the same color with the other buttons on the navigation bar to create harmony ● Check if the Favorites button is aligned at the same place on each page
Expected results	<ul style="list-style-type: none"> ● Favorites button is aligned at the same place on each page and has the same color with the other buttons to create harmony ● Favorites button is aligned properly and is readable on different browsers.
Priority	Minor

Table 90: Test case table 90

Test ID	Test 91
Test Type	Nonfunctional Testing (Usability & Performance)
Title	Test Cases for Favorite Plans (Addition & Removal)
Procedure	<p>Usability</p> <ul style="list-style-type: none"> ● Check if the star button for addition and removal of a plan from favorites list is located on the right corner of the plan box and aligned properly on all browsers <p>Performance</p> <ul style="list-style-type: none"> ● Check if the plan is removed from the favorites database table in a few seconds if it is removed by the user from the favorites by pressing the previously selected star button ● Check if the plan is added to the favorites database table in a few seconds if it is added by the user to the favorites by pressing the star button ● Check if the plan is removed from the Favorites page in a few seconds after completing the removal request from the favorites ● Check if the plan is added to the Favorites page in a few seconds after completing the addition request to the favorites
Expected results	<ul style="list-style-type: none"> ● The star button for addition/removal from the favorites list is

	<p>aligned on the right side of the plan titles page and it is readable and aligned properly on different browsers</p> <ul style="list-style-type: none"> • The previously clicked star button can be unselected by pressing on it and the database and Favorites page must be updated in a few seconds by the removal of the plan from favorites • The star button can be clicked for adding a plan to Favorites and the associated database table is updated in a few seconds along with the Favorites page of the user.
Priority	Major

Table 91: Test case table 91

Test ID	Test 92
Test Type	Nonfunctional Testing (Usability & Security & Performance)
Title	Test Cases for Logout button
Procedure	<p>Usability</p> <ul style="list-style-type: none"> • Check if the Logout button is aligned properly on all browsers • Check if the Logout button is displayed on the left bottom corner of the screen at all pages <p>Security</p> <ul style="list-style-type: none"> • Check if the user's account remains still open after clicking the Logout button • Check if the user's account information are stored in cache even after clicking the Logout button <p>Performance</p> <ul style="list-style-type: none"> • Check if the user is logged out from his/her account in a few seconds after clicking the Logout button
Expected results	<ul style="list-style-type: none"> • The Logout button is aligned properly on all browsers and located at the left bottom corner of the screen at all pages • The user cannot reach his/her account before logging in once more after clicking Logout button • User's account information is not stored in cache after logging out from the application • The user is logged out from the account in 1-2 seconds after clicking the Logout button
Priority	Critical

Table 92: Test case table 92

Test ID	Test 93
----------------	---------

Test Type	Nonfunctional Testing (Documentation)
Title	Test Cases for User Manual
Procedure	<ul style="list-style-type: none"> ● Check if the user can sign up to the application by using the information provided in the manual ● Check if the user can log in to the application by using the information provided in the manual ● Check if the user can create/delete travel plan by using the information provided in the manual ● Check if the user can add/remove a plan to favorites by using the information provided in the manual ● Check if the user can update profile photo by using the information provided in the manual ● Check if the user can change password by using the information provided in the manual ● Check if the user can view the plans by using the information provided in the manual ● Check if the user can see the mostly preferred routes by using the information provided in the manual ● Check if the user can log out from application by using the information provided in the manual
Expected results	<ul style="list-style-type: none"> ● The user can sign up to the application by using the information provided in the manual ● The user can log in to the application by using the information provided in the manual ● The user can create/delete travel plan by using the information provided in the manual ● The user can add/remove a plan to favorites by using the information provided in the manual ● The user can update profile photo by using the information provided in the manual ● The user can change password by using the information provided in the manual ● The user can view the plans by using the information provided in the manual ● The user can see the mostly preferred routes by using the information provided in the manual ● The user can log out from application by using the information provided in the manual
Priority	Major

Table 93: Test case table 93

6. Consideration of Various Factors in Engineering Design

Several factors that may affect the engineering design and implementation has been thought of and are described below:

- **Public Health**

Travimize takes public health into consideration when creating a personalized travel plan. It does not create a plan where the travelers need to walk more than humanly possible or it does not constrain their sleep under 7-8 hours to fit in more attractions to the plan. Moreover, Travimize will make sure that the travelers eat a healthy amount of meals throughout the day.

- **Public Safety**

Travimize takes public safety very seriously since it is very easy to run into a safety problem when traveling. In order to ensure safety of its users, Travimize will not suggest attractions, restaurants, hotels, parks, or any other place whose safety is suspicious or has no comments to its users. Travimize will also collect feedback from its users after a trip has ended to make sure their trip was safe. It will use this data to create safer routes.

- **Public Welfare**

Travimize is a free to use web application that is planned to be supported by advertisements and commissions from suggested hotels, restaurants or any other institution that profits from this web application. Moreover, Travimize will encourage people to travel and experience new places. Not only the travelers will broaden their minds, local businesses will also benefit from the outcomes of the trips.

- **Global, Cultural and Social Factors**

Travimize does not support global travel yet, it only supports travel plans inside the borders of Turkey. However, in the future, this application is expected to support global travel too. Travimize is not affected by cultural differences since it creates personalized plans based on the constraints the user has given. Moreover, Travimize

will try to make the traveler experience as much local influence as possible. Travimize will also pave the way for its users to socialize and travel together with its “share a route with a friend” feature.

- **Economic Factors**

Travimize will use several APIs from third parties in the implementation. Also the application will use a server and a relational database. These factors will have a cost. Currently this cost is minimal for a small number of users and can be afforded easily. However, in the future, when the application is deployed and used by many, in order for the cost to be afforded, there can be a need for sponsors and possible in-app purchases.

- **Environmental Factors**

Travimize will check the condition of the destinations both before creating and during the travel plan. If a destination is closed, if the fees have changed or if the road to the destination is closed, the application will notify the user so that there will be no trouble for the user. Travimize will also take the weather conditions or the degree of a ramp when it schedules the user for a walk.

	Effect level	Effect
Public health	5	Not threatening the health of the users will be a concern when creating a plan.
Public safety	8	Not suggesting destinations or hotels that might risk the safety of the users will be a concern when creating a plan.
Public welfare	3	Ease of reach and use of the application will be considered when putting a price on the application.
Global factors	1	Global factors will have no effect in the design.

Cultural factors	2	Travimize will try to let users experience all the cultural aspects of the destination location when creating a plan.
Social factors	3	Travimize will let users interact with other users by sharing a route with each other.
Economic factors	7	Cost affordability will be the primary concern when designing and implementing the application.
Environmental factors	5	Environmental factors will be taken into consideration when creating a personalized travel plan.

Table 94: Factors that can affect analysis and design.

7. Teamwork Details

7.1. Contributing and Functioning Effectively on the Team

Teamwork is one of the crucial points for the project. For effective teamwork we divided the project group into smaller teams and maintained the communication between these teams according to the plans of the leaders in the different parts of the projects. To elaborate, at the initial stage, we divided the project into 6 parts and assigned a leader for each of the parts. These leaders were responsible to maintain the communication between the small teams and plan weekly meetings. Apart from the leaders, for the implementation the sub teams are divided to 3, where frontend team members are Doğa Ece Ersoy and Emre Erdal, backend team members are Esra Genç and Doruk Kantarcıoğlu, algorithm team member is Kerem Erdal. Hence, to make effective teamwork, we planned regular meetings and divided the team into smaller subteams.

7.2. Helping Creating a Collaborative and Inclusive Environment

For a collaborative and inclusive environment, we do not only meet via Zoom but we organize some face to face meetings at least once in 2 weeks and try to maintain a stronger bond between the team members. This action creates an inclusive environment where every team member wants to contribute to the work. Moreover, communication is a critical point at this part. For this communication purpose, we are using Whatsapp and Zoom, GitHub for sharing

the codes for the project and Google Drive for sharing the reports. These tools help us to demonstrate teamwork more effectively.

7.3. Taking Lead Role and Sharing Leadership on the Team

To have effective teamwork and a successful product as a result, leadership is a key feature. A leader is necessary to organize meetings easier, check the process clearly and demonstrate effective teamwork. Hence, at the initial stage of the project we decided to divide the project stages into 6 categories and assign the leaders for these categories. In the Project Specifications and Analysis part, the leader was Doğa Ece Ersoy. In the Pre-prototype-implementation study part, the leader was Doruk Kantarcıoğlu. In the Prototype implementation part, the leader was Emre Erdal. In the High Level and Low Level Design part, the leader is Esra Genç. In the complete product implementation where we are currently detailing with, the leader is Kerem Erdal. Lastly, in the Testing part, the leader will be Doğa Ece Ersoy.

8. References

[1] J. Desjardins, “10 types of innovation: The art of discovering a breakthrough product,” *Visual Capitalist*, 02-Jul-2020. [Online]. Available: <https://www.visualcapitalist.com/10-types-of-innovation-the-art-of-discovering-a-breakthrough-product/>. [Accessed: 16-Oct-2022].

[2] “Over a billion reviews & contributions for hotels, attractions, restaurants, and more,” *Tripadvisor*. [Online]. Available: <https://www.tripadvisor.com/>. [Accessed: 13-Nov-2022].

[3] “An easier trip, each time,” *TripIt*. [Online]. Available: <https://www.tripit.com/web>. [Accessed: 13-Nov-2022].

[4] “Trip planner: Plan & manage your vacation itinerary on inspirock • inspirock,” *Inspirock*. [Online]. Available: <https://www.inspirock.com/>. [Accessed: 13-Nov-2022].