Test Plan for System Analysis

Mohammad Hesham Elsayed, Mark Mounir, Omar Hosny, Zeyad Abdelnasser Elzayaty
Supervised by:
Dr. Sahar Abdelrahman
Dr. Maha Sayed

May 21, 2024

Table 1: Document version history

Version	Date	Reason for Change
1.0	17-Mar-2024	Test Plan First version is defined.
1.1	17-May-2024	Test Scenarios "Select tour" and "Login" have been updated.
1.2	18-May-2024	Test Scenarios "Select transportation" and "Select Hotel" have been added.

Contents

1 Introduction

1.1 Purpose

The purpose of this document is to outline the testing strategy and procedures for verifying the functionality of the "Select tour", "Login", "Select Transportation", "Select Hotel" features in the application. It provides a structured approach to ensure that the features meet the specified requirements, functions correctly, and maintains expected behavior across different scenarios. Additionally, this document serves as a reference for stakeholders involved in the testing process, including developers, testers, and project managers, to understand the scope, objectives, and resources required for testing the "Select tour", "Login", "Select Transportation", "Select Hotel" functionality effectively.

2 Test Scenario "Select tour"

2.1 User Actions and Behaviors

- **User Interaction**:
  - Users navigate to the tours page or tour listing where the "Select tour" functionality is available.
  - They may be logged in or not logged in to the system.
- **Actions**:
  - Logged-In User: Clicks on the "Book" button/link associated with a specific tour.
  - Not Logged-In User: Attempts to "Book" to the cart without logging in.
- **Behaviors**:
  - Successful Addition: The tour is added to the user's cart, and a confirmation message is displayed.
  - Existing tour: If the tour is already in the cart, the system prevents duplication and notifies the user.
  - User Not Logged In: If the user is not logged in, the system prompts the user to log in or register before adding tour to the cart.

2.2 Technical Aspects

- **Frontend Interaction**:
  - The "Book" button triggers an HTTP request to the backend.
  - The request includes the tour ID and user session information (if logged in).
- **Backend Processing**:

  - The backend controller (`TourController`) handles the request.
  - It retrieves the necessary data from repositories (e.g., `UserRepository`, `TourRepository`) to validate the request.
  - The system checks if the tour is already in the user's cart and performs the necessary actions.

- **Data Persistence**:
  - The system updates the user's cart in the database (if applicable) to reflect the addition of the tour.

- **Response**:
  - The system sends an appropriate response to the frontend indicating the success or failure of the operation.

2.3 Scenarios of System Failure

- **Invalid tour ID**:
  - If the provided tour ID is invalid or does not exist, the system should handle the error gracefully and notify the user.

- **Database Connection Failure**:
  - If there is a failure in connecting to the database or updating the cart, the system should log the error and display a generic error message to the user.

- **Session Management Issues**:
  - If there are issues with user session management (e.g., session timeout, session invalidation), the system should redirect the user to the login page or prompt them to re-authenticate.

- **Concurrency Issues**:
  - If multiple users attempt to add the same tour to their carts simultaneously, the system should handle concurrency issues to prevent data inconsistency or race conditions.

2.4 References

- Use Cases in Software Requirements Specification (SRS)
- Sequence Diagrams in Software Design Document (SDD)

2.5 Test Cases

Test Cases for the scenario mention in section 2 shown in Table 2

Table 2: Test Cases for "Select Tour" Scenario

Test Case ID	Test Case Description	Functional Requirement Code	Test Data	Expected Result
TC01	Add New tour to Cart	FR001	Valid tour ID, Logged-in user session	Tour added to tour successfully
TC03	Add Tour to Cart (User Not Logged In)	FR001	Valid tour ID, No user session	User not logged in

3 Test Scenario "Login"

In this test scenario, we focus on testing the functionality of the "Login" feature in the application. The purpose of this feature is to allow users to authenticate and gain access to the system.

3.1 Possible Users' Actions and Behaviors

Users interact with the application by accessing the "Login" page, entering their credentials (such as username and password), and submitting the form. They may also navigate away from the page without logging in.

3.2 Technical Aspects of the Requirement

The backend of the application should handle login submissions by validating the input data, authenticating the user, and establishing a user session. It should also handle error conditions, such as invalid credentials or system errors, gracefully.

3.3 Possible Scenarios of System Failure

- **Invalid Credentials:** If the user submits incorrect or incomplete login credentials, the system should provide appropriate error messages and prompt the user to correct the input.
- **System Error:** In case of a failure while processing the login request (e.g., server issues or database connection problems), the system should handle the error gracefully and notify the user about the issue.
- **Session Timeout:** If the user's session expires while attempting to log in, the system should prompt them to log in again. Any previously entered credentials should be preserved.

This test scenario ensures that the "Login" feature functions correctly and meets the requirements specified in the Software Requirements Specification (SRS) and Sequence diagrams in the Software Design Document (SDD).

3.4 Test Cases

Test Cases for the scenario mentioned in section 3 are shown in Table 3.

Table 3: Test Cases for "Login" Scenario

Test Case ID	Test Case Description	Functional Requirement Code	Test Data	Expected Result
TC03	Submitting Valid Credentials	FR02	Valid username and password, user session active	User authenticated, access granted, user session established
TC04	Submitting Invalid Credentials	FR02	Invalid username or password, user session active	Error message displayed, user prompted to correct input

4 Test Scenario "Select Transportation"

This scenario involves users interacting with the "Select Transportation" feature, where users can choose the plane ticket they need to travel to their tour via a form submission. The technical aspects include verifying form validation, session handling, and associating the transportation details with the correct user. Possible system failures include issues with form validation, session management, and database interactions.

4.1 Possible Users' Actions and Behaviors

Users interact with the application by accessing the "Travel" page, where they can view all available plane tickets. Users select a suitable ticket, choose the travel date, and proceed to checkout. They may also navigate away from the page without completing the selection.

4.2 Technical Aspects of the Requirement

The backend of the application should handle ticket selection and form submissions by validating the input data, associating the transportation details with the logged-in user, and storing the transportation details in the database. It should also handle error conditions, such as invalid input data or database errors, gracefully.

4.3 Possible Scenarios of System Failure

- **Invalid Input Data:** If the user submits invalid or incomplete data during ticket selection or checkout, the system should provide appropriate error messages and prompt the user to correct the input.
- **Database Error:** In case of a failure while saving the transportation details to the database (e.g., database connection issues or constraints violation), the system should handle the error gracefully and notify the user about the issue.

- **Session Timeout:** If the user's session expires while selecting a ticket or during checkout, the system should redirect them to the login page and prompt them to log in again. Any data entered in the form prior to the session timeout should be preserved.

4.4 Test Cases for "Select Transportation" Scenario

Table 4: Test Cases for "Select Transportation" Scenario

Test Case ID	Test Case Description	Functional Requirement Code	Test Data	Expected Result
TC05	Selecting Valid Ticket	FR003	Valid ticket selection, logged-in user session	Transportation details saved successfully, transportation associated with user
TC06	Submitting Invalid Data	FR003	Invalid ticket selection or date, logged-in user session	Error message displayed, user prompted to correct input
TC07	Session Timeout	FR003	Valid ticket selection, session timeout	User redirected to login page, data entered preserved

5 Test Scenario "Select Hotel"

This scenario involves users interacting with the "Select Hotel" feature, where users can choose a hotel for their tour via the Hotel pages. The technical aspects include verifying form validation, session handling, and associating the hotel selection with the correct user. Possible system failures include issues with form validation, session management, and database interactions.

5.1 Possible Users' Actions and Behaviors

Users interact with the application by accessing the "Hotel" pages, where they can view all available hotels. Users select a suitable hotel, choose the check-in and check-out dates, and proceed to checkout. They may also navigate away from the page without completing the selection.

5.2 Technical Aspects of the Requirement

The backend of the application should handle hotel selection and form submissions by validating the input data, associating the hotel details with the logged-in user, and storing the hotel details in the database. It should also handle error conditions, such as invalid input data or database errors, gracefully.

5.3 Possible Scenarios of System Failure

- **Invalid Input Data:** If the user submits invalid or incomplete data during hotel selection or checkout, the system should provide appropriate error messages and prompt the user to correct the input.
- **Database Error:** In case of a failure while saving the hotel details to the database (e.g., database connection issues or constraints violation), the system should handle the error gracefully and notify the user about the issue.
- **Session Timeout:** If the user's session expires while selecting a hotel or during checkout, the system should redirect them to the login page and prompt them to log in again. Any data entered in the form prior to the session timeout should be preserved.

5.4 Test Cases for "Select Hotel" Scenario

Table 5: Test Cases for "Select Hotel" Scenario

Test Case ID	Test Case Description	Functional Requirement Code	Test Data	Expected Result
TC05	Selecting Valid Hotel	FR004	Valid hotel selection, logged-in user session	Hotel details saved successfully, hotel associated with user
TC06	Submitting Invalid Data	FR004	Invalid hotel selection or dates, logged-in user session	Error message displayed, user prompted to correct input
TC07	Session Timeout	FR004	Valid hotel selection, session timeout	User redirected to login page, data entered preserved