AI Trip Creator Project: Testing Policy Document

The Rolling Capstones

October 1, 2024

1 Overview

This testing policy outlines the strategies, procedures, and responsibilities for ensuring the quality and functionality of the AI Trip Creator project. The testing process aims to validate that all components, from user interface elements to backend services, function as intended, meet performance standards, and provide a smooth user experience.

2 Objectives

- Ensure the core features of the AI Trip Creator operate without defects.
- Validate data integrity between the frontend (ReactJS) and backend (Firestore or relevant database).
- Test user experience (UX) flows, including itinerary creation, editing, and deletion.
- Test personalization features, such as home page greetings, user reminders, and user-specific content (itineraries, interests).
- Identify potential performance bottlenecks, security vulnerabilities, and UI/UX issues.
- Ensure integration between modules (Flights, Analytics, Itineraries) functions seamlessly.

3 Testing Scope

The following functionalities will be tested:

- **Itinerary Creation & Editing**: Ensuring users can create and edit itineraries with correct data persistence in Firestore.
- **Itinerary Deletion**: Verifying that itineraries are deleted correctly and that the `deleteItineraryFromFirestore()` function works without errors.
- **Personalized Home Page**: Testing that users are greeted by name and shown the correct personalized content based on their activities.

- **User Profile Features**: Testing profile editing, preferences, and analytics dashboard interactions.
- **Unique Itinerary Photos**: Ensuring that each itinerary card displays a unique, relevant image.
- **User Behavior Analytics**: Ensuring proper data collection and display of user behavior on the profile page.
- **UI Responsiveness**: Ensuring the interface works across different devices (mobile, tablet, desktop).

4 Testing Types

4.1 Unit Testing

Test individual components, methods, and functions, such as:

- `deleteItineraryFromFirestore()`
- Profile editing form handlers.

4.2 Integration Testing

Ensure modules work together, including the interaction between the frontend and backend.

- Itinerary data persistence with Firestore.
- Interaction between the Analytics and Flights components.

4.3 End-to-End Testing (E2E)

Simulate real user journeys to ensure the system works as expected from start to finish. Test scenarios:

- Creating, editing, and deleting itineraries.
- Viewing personalized content on the home page.
- Modifying profile preferences and seeing the changes reflected in recommendations.

4.4 Usability Testing

Collect feedback on ease of use, interface clarity, and navigation flow.

4.5 Performance Testing

Measure response times, system load, and stress test the platform, particularly during operations like fetching itineraries or large data loads in the analytics dashboard.

4.6 Security Testing

Ensure protection against unauthorized data access, cross-site scripting (XSS), and proper handling of sensitive information like user preferences.

5 Test Environment

- **Test Data**: Use mock data for flights, itineraries, and user profiles.
- **Devices**: Test across a range of devices (desktop, tablet, mobile) to ensure responsiveness.
- **Browser Compatibility**: Test in the latest versions of Chrome, Firefox, Safari, and Edge.

6 Tools and Frameworks

6.1 Frontend Testing

- **Jest** and **React Testing Library** for unit testing React components.
- **Cypress** for end-to-end testing.

6.2 Backend Testing

- **Firebase Emulator** for testing Firestore queries and data updates.

6.3 Performance Testing

- **Lighthouse** for assessing page speed, responsiveness, and best practices.

6.4 Security Testing

- Use tools like **OWASP ZAP** or **Burp Suite** to identify vulnerabilities.

7 Test Plan

7.1 Development Phase

Test individual components (unit testing) as they are developed, focusing on the core functionalities of itinerary creation and profile personalization.

7.2 Pre-Release Phase

Conduct integration and E2E testing. Prioritize critical user flows, including itinerary creation and deletion, profile editing, and personalized content display.

7.3 Post-Release Phase

Implement continuous testing for new features and bug fixes, utilizing automated tests where possible.

8 Reporting & Resolution Process

8.1 Bug Reporting

Use a ticketing system (e.g., Jira, GitHub Issues) to log any issues found during testing. Each bug report should include a description, steps to reproduce, expected vs. actual behavior, and priority level.

8.2 Resolution & Retesting

Developers will resolve reported issues, and once resolved, the features will be retested to ensure the fix is effective.

9 Roles & Responsibilities

9.1 Testers

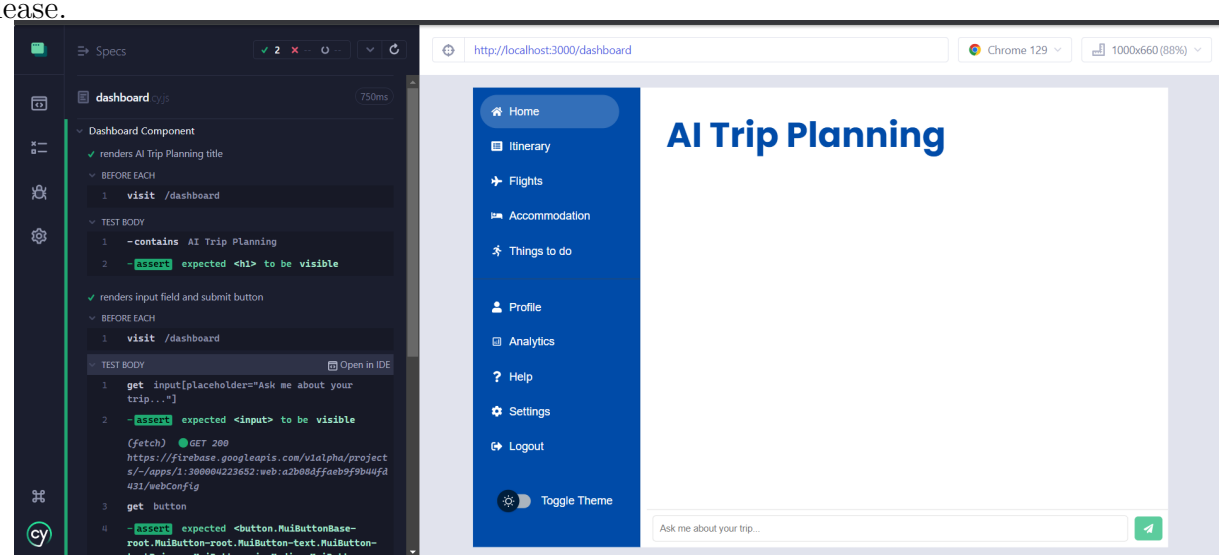
Responsible for running tests, documenting results, and raising bugs. Testers will also provide feedback on usability.

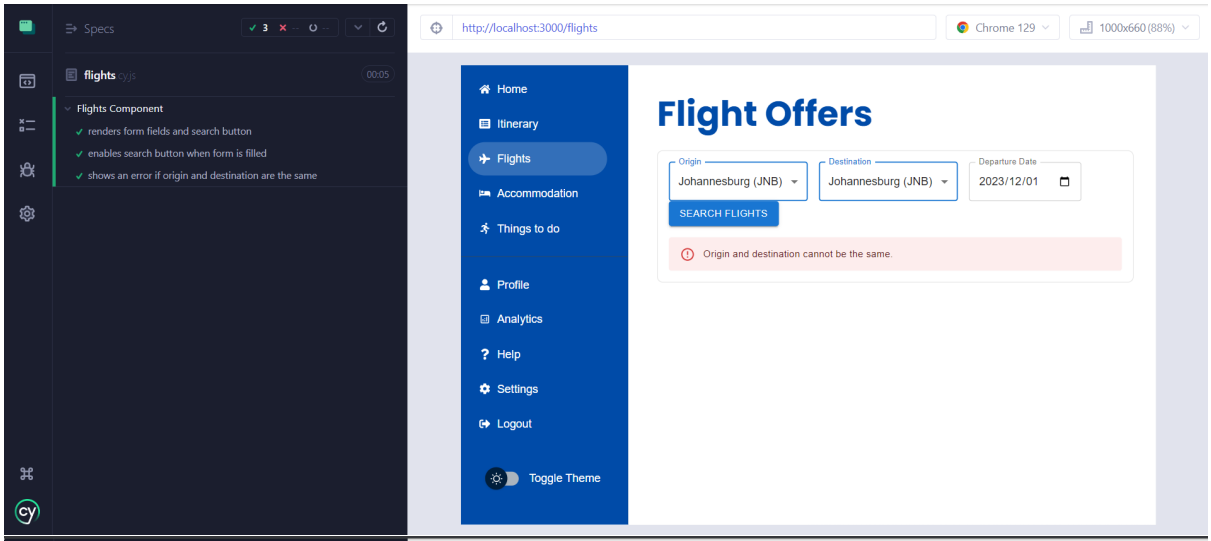
9.2 Developers

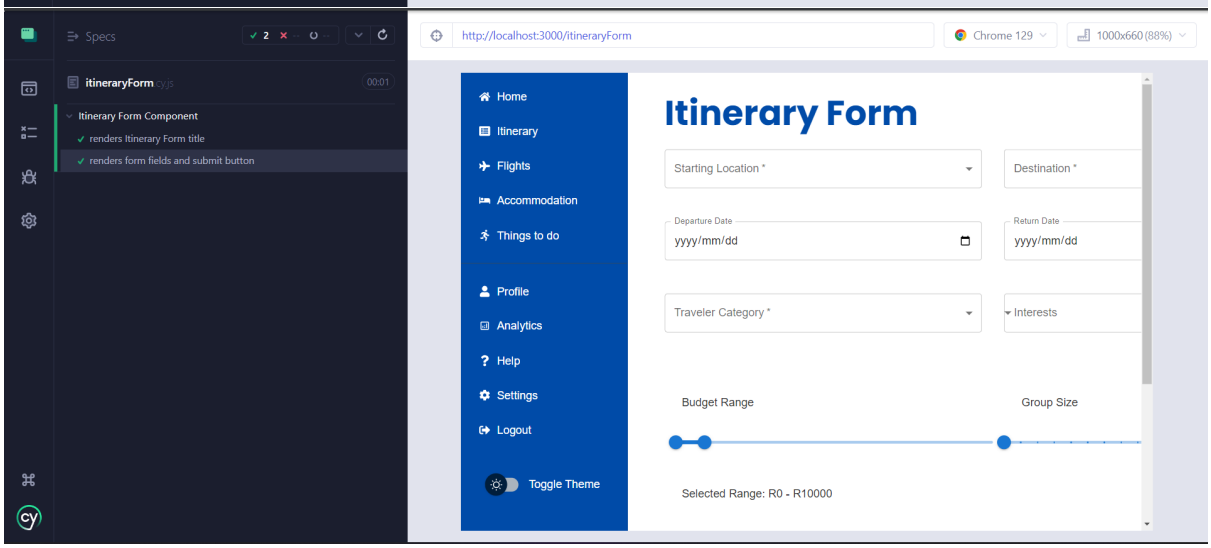
Responsible for writing unit tests for their code and ensuring that issues raised are resolved promptly.

9.3 Project Manager

Ensures adherence to the testing timeline and that all critical functionality is tested before release.




Specs
3
flights.cy.js
00:05
Flights Component
renders form fields and search button
enables search button when form is filled
shows an error if origin and destination are the same
http://localhost:3000/flights
Chrome 129
1000x660 (88%)
Home
Itinerary
Flights
Accommodation
Things to do
Profile
Analytics
Help
Settings
Logout
Toggle Theme
Flight Offers
Origin
Johannesburg (JNB)
Destination
Johannesburg (JNB)
Departure Date
2023/12/01
SEARCH FLIGHTS
Origin and destination cannot be the same.


Specs
2
itineraryForm.cy.js
00:01
Itinerary Form Component
renders Itinerary Form title
renders form fields and submit button
http://localhost:3000/itineraryForm
Chrome 129
1000x660 (88%)
Home
Itinerary
Flights
Accommodation
Things to do
Profile
Analytics
Help
Settings
Logout
Toggle Theme
Itinerary Form
Starting Location *
Destination *
Departure Date
yyyy/mm/dd
Return Date
yyyy/mm/dd
Traveler Category *
Interests
Budget Range
Group Size
Selected Range: R0 - R10000