Test Plan for FloTorch.ai

Introduction

This test plan outlines the scope, objectives, approach, resources, and schedule for testing the FloTorch.ai application. The purpose of this testing effort is to validate the functionality, usability, performance, and reliability of FloTorch.ai, ensuring it meets user expectations and business requirements.

---

1. Scope

The testing will focus on the following aspects of FloTorch.ai:

- Evaluate the automated installation and uninstallation processes for the application and its prerequisites on AWS, ensuring seamless setup and removal.
- Conduct tests across multiple permutations of user parameters, validating the functionality of each parameter by selecting and testing different values to ensure they work correctly within valid ranges.
- Assess the user interface functionality, including login, project creation, experiment configuration, and monitoring. Verify that file upload and download features operate as expected, along with the save and upload configuration functionalities.
- Test the execution of tasks initiated from the user interface, ensuring outputs align with expected question-and-answer results.
- Validate experiment evaluation metrics for accuracy and reliability in presenting results.
- Examine the performance and reliability of both AWS infrastructure and application components under varying load conditions to ensure scalability, responsiveness, and resilience

---

2. Objectives

- Validate the correct installation and setup of the application on AWS.
- Ensure smooth functionality of the user interface for project creation and experiment management.
- Verify the accuracy and completeness of experiment evaluation metrics.
- Test the robustness of AWS infrastructure components under expected loads.
- Ensure proper cleanup of resources during the uninstallation process.

---

3. Test Approach

The testing approach will combine manual and automated testing methods.

1. **Manual Testing**: For UI functionality, user workflows, and evaluation metrics validation.
2. **Automated Testing**: For infrastructure deployment, teardown, and backend API workflows using scripts.

---

4. Test Deliverables

- Test Plan Document
- Test Cases Document
- Test Execution Report
- Final Test Summary Report

---

5. Test Environment

- **Hosting Platform:** AWS App Runner
- **Front-end Framework:** Nuxt 3, Vue 3
- **Back-end Framework:** FastAPI
- **Database:** DynamoDB

Test Data

- Sample knowledge base and ground truth datasets for RAG experiments.
- Configuration files (JSON) for indexing and retrieval strategies.

---

6. Test Schedule

Phase	Activity
Test Planning	Create test plan and test cases
Test Environment Setup	Deploy AWS infrastructure and install tools
Test Execution	Perform functional, performance, and stress tests
Defect Reporting	Log and track defects
Test Closure	Prepare test summary and finalize reports

7. Test Cases

Category:

- Installation and Setup
- User Interface
- Experiment Management
- Evaluation Metrics
- Uninstallation

---

8. Defect Management

Defects will be logged in a tracking tool with the following details:

- Defect ID
- Description
- Defect Raised by

---

9. Risks and Mitigation

Risk	Impact	Mitigation
Incorrect AWS permissions or setup	Installation failure	Validate permissions before testing.
Large datasets causing performance lag	Delayed experiment runs	Use sample datasets for testing.
Script or tool incompatibility	Deployment errors	Ensure prerequisites are installed.

---

10. Approval

This test plan has been reviewed and approved by the following stakeholder:

- **Project Manager**

Conclusion

This test plan provides a comprehensive approach to validate FloTorch.ai. It ensures a thorough assessment of installation, functionality, and performance to deliver a reliable product.