Solace

Test Strategy

Revision History

Date	Version	Author	Description
13th March 2022	1	Sudais Baig	The first basic version of the test strategy to be used for this project.

Table of Contents

1. Scope

This section will define the scope of the Test Strategy for Solace. This document shall be completed and used by the project test team to guide how testing will be managed for this project. The test effort will be prioritized and executed based on the project priorities as defined in the Project Plan and Requirements Specification. This is a living document that maybe refined as the project progresses. All the team members shall review the document before submission and the instructor – Abbas Akhtar – will approve the final version of the Test Strategy document.

2. Test Approach

The testing approach for this release shall be done in a fashion that will accommodate the current functionality in Solace.

Testing will cover functionality testing for Solace changes through the use of the test interface. This will validate base functions of the new code as it relates to the standard model of presentation for data and user entered data.

Process of testing:

The testing process comprises of the following steps.

1) Test Software Requirements
2) Test Software Design – tests both external and internal design primarily through verification techniques.
3) Build Phase Testing – test all of the build
4) Execute and Record Result
5) Test what user believes software should perform, as against what documented requirements state software should perform.
6) Report Test Results

Testing levels:

We will use four levels of testing: unit testing, integration testing, system testing, and acceptance testing. The purpose of testing levels is to make software testing systematic and easily identify all possible test cases at a particular level.

1) **Unit Testing**: checks if software components are fulfilling functionalities or not.
2) **Integration Testing**: checks the data flow from one module to other modules.
3) **System Testing**: evaluates both functional and non-functional needs for the testing.
4) **Acceptance Testing**: checks the requirements of a specification or contract are met as per its delivery.

Roles and responsibilities of each team member:

- QA team – Sudais Baig, Bilal Fayaz
  Will be responsible for planning the entire testing process, check the availability of resources, preparing status report of the testing activities.
- Test team – Hafsa Irfan, Aliza Lakhani
  Will be responsible for implementing the test process, update latest test results and tools, completing test product documentation, conducting test as per the test standard and analyse the results.

Types of Testing:

- Load testing: Load testing gives confidence in the system & its reliability and performance. This will helps identify the bottlenecks in the system under heavy user stress. We will use open source load testing tools
- Security testing: We will use Security Testing is to identify the threats in the system and measure its potential vulnerabilities, so the threats can be encountered and the system does not stop functioning or cannot be exploited.
- Performance testing: this will determine whether their software meets speed, scalability and stability requirements under expected workloads.
- Black box testing
- White box testing

Testing approach & automation tool if applicable:

We will mainly use manual testing for this project, testing every module traditionally as we go along. And then finally performing integration testing after integrating everything. But we also plan move to automated testing when needed. For instance, in case of load testing, we plan to use tools like Selenium.

Adding new defects, re-testing, Defect triage, Regression Testing and test sign off:

- After fixing any bugs, we will use **retesting** to check specific test cases that were found with bug/s in the final execution.
- After adding any new features, we will use **regression testing** to check whether a code change has not unfavourably disturbed current features & functions of Solace
- After going through rigorous testing of our product solace, and after getting everything verified by the instructor, we will formally declare the completion of testing by **test sign off.**

3. Test Environment

Following are the key requirements of the test environment

- System and applications
- Test data
- Database server
- Front-end running environment
- Browser
- Network
- Documentation required like reference documents/configuration guides/installation guides/ user manuals

Setup required for each environment

Network setup as per the test requirement. It includes

- Internet setup
- LAN Wi-Fi setup
- Private network setup

Test PC setup

- Since our project is a web-based application, we will use web testing, and will need to set up different browsers for different testers.

Define backup of test data and restore strategy

We plan to create multiple backups stored both online and offline.

- We will use our OneDrive provided by Habib University to store backups online.
- Backups will be made manually after every update to the code.
- Offline backups will be stored on every developer's system.
- In case of a disaster, the most recent backup of the test data would be restored from OneDrive.

4. Testing Tools

We plan to use the following testing and test management tools for our project

- Selenium: As Selenium is compatible with quite a lot of programming languages, testing frameworks, browsers and operating systems that’s why we are using selenium.
- Apache JMeter: A load testing tool, Apache JMeter is available in an open-source format, it is highly accessible to software businesses of all sizes. The testing tool is extremely versatile, despite being open-source. It is compatible with several web and networking protocols such as HTTP, HTTPS, FTP, LDAP, SOAP, and TCP that’s why we will be using this. This tool can simulate up to 10,000 users.
- Qase : Qase is a TestOps system that allows you to manage every aspect of QA in a single collaborative workspace, arming you with tools for test management, test execution and results reporting, extensive analytical capabilities, requirement and defect management, and integrations with 3rd party project management tools as well as automated testing frameworks. That''s why we will be using Qase.

5. Release Control

The release management plan will have different steps,

- Plan release: A robust release plan will help our team stay on track and ensure standards and requirements are properly met. The release management method we will be using is the systems development life cycle.
- Build Release: With the release plan finalized, we will start designing and building the product for release. This is the actual “development” of the product based on the requirements outlined in the release plan.
- User acceptance testing: This is used to identify bugs and fix these bugs.
- Prepare Release: To put the finishing touches on the product taking into account things and bugs found during user acceptance testing.
- Deploy Release: Time to release the product.

6. Risk Analysis

Risk Analysis in project management is a sequence of processes to identify the factors that may affect a project's success.

- Bad Timing. Anyone from the team may suffer from this.
- Poor Code Quality. Code may malfunction.
- Technical Risks. Anyone may face technical difficulties for instance, their laptop may get damaged or they may face internet issues.
- Poor Productivity. Lagging behind in work. Struggling to meet deadlines.
- Poor Management and communication between team members.
- Unpredictable External Risks.

A plan to mitigate the risks also a contingency plan

- Plan everything from the beginning. We will discuss how to go about the sprint before the beginning of every sprint. Set strict deadlines and hope to achieve them.
- Test every aspect of the code rigorously. Make sure no errors or bugs are present before moving forward.
- Technical risks are inevitable. The best we may do is to cover for each other, if one team member faces any technical difficulties, others will come to help however they can.
- Again, we will aim to achieve and meet the deadlines as best as we can but in case one lags behind in their work, the other team members will help them out.

7. Review and Approvals

- All these activities are reviewed and signed off by the team and the instructor.
- Summary of review changes should be traced at the beginning of the document along with approved date, name, and comment.