SOFTWARE TEST PLAN (STP)

Monday website

Nidaa AbuSaleh

TABLE OF CONTENTS

INTRODUCTION

1. Purpose:

The purpose of this Software Test Plan (STP) is to outline the approach, scope, resources, and schedule for conducting QA automation tests on the Monday.com website. The STP aims to ensure that the website meets the specified quality standards and that any defects are identified and addressed effectively.

2.Project Overview:

The project involves the automation of QA tests for the Monday.com website, a popular work operating system used by teams to manage their work and projects. The automation tests will cover various aspects of the website, including functionality, usability, and performance, to ensure a high level of quality and reliability.

3.Website Overview:

The Monday.com website serves as a central hub for teams to collaborate, plan, and execute their work. It offers features such as task management, project tracking, team communication, and more. The website is accessible via web browsers on desktop and mobile devices, making it essential to ensure compatibility and functionality across different platforms.

Test Strategies

1.Test Objectives:

Functional Testing: Validate critical functionalities of the Monday.com website, including task creation, assignment, and status tracking, to ensure they meet specified requirements.
Compatibility Testing: Verify the website's cross-browser (Chrome, Firefox, Edge)compatibility to guarantee a consistent user experience across different platforms.

Usability Testing: Validate the website's usability and accessibility to ensure it meets the needs of diverse user groups, including those with disabilities.

2. Feature to be tested:

Module Name	**Applicable Roles**	**Description**
Login	User	Verify successful and failed login attempts with various credentials and connected accounts.
Boards	User	Create customizable boards to organize tasks, projects, and processes.
Deal Management	User	Create, update, and manage deals, including pipeline progression.
Search and Filter	User	● Use powerful search and filter options to quickly find information. ● Refine views based on specific criteria to focus on relevant data.
File Sharing:	User	● Attach files directly to items for easy reference and collaboration. ● Collaborate on documents and files within the platform.
User Management	User	Create, delete, and manage user accounts with appropriate permissions.
Data Import and Export	User	Import and export data for various purposes.
Automation and Workflow	User	Test workflow automation and custom rules.
Security	User	Ensure user authentication, authorization, secure access, and data encryption.

3.Test Assumptions:

The Monday.com website is stable and accessible for testing purposes.

Test environments accurately reflect the production environment in terms of configuration and usage.

Test data is available and can be manipulated to simulate various scenarios.

4.Test Approach:

Automation: Employ automated testing tools, such as Selenium WebDriver with Python, to increase test efficiency and coverage.

Comprehensive Testing: Utilize a combination of functional, compatibility, performance, security, and usability testing to ensure comprehensive test coverage.

Risk-Based Testing: Prioritize testing efforts using a risk-based approach to focus on high-impact and high-likelihood scenarios.

5.Data Approach:

Synthetic Data: Utilize synthetic data to simulate real-world scenarios and test edge cases effectively.

Test Data Management: Employ test data management tools to generate and manage test data efficiently and securely.

Data Privacy: Ensure compliance with data protection regulations and maintain data privacy throughout the testing process.

6.Levels of Testing:

Unit Testing: Validate individual components and modules to ensure they function as intended.

Integration Testing: Verify the integration of different components to ensure seamless communication.

System Testing: Validate the entire system to ensure all components work together harmoniously.

Acceptance Testing: Test the website with end users to validate its functionality, usability, and compliance with requirements.

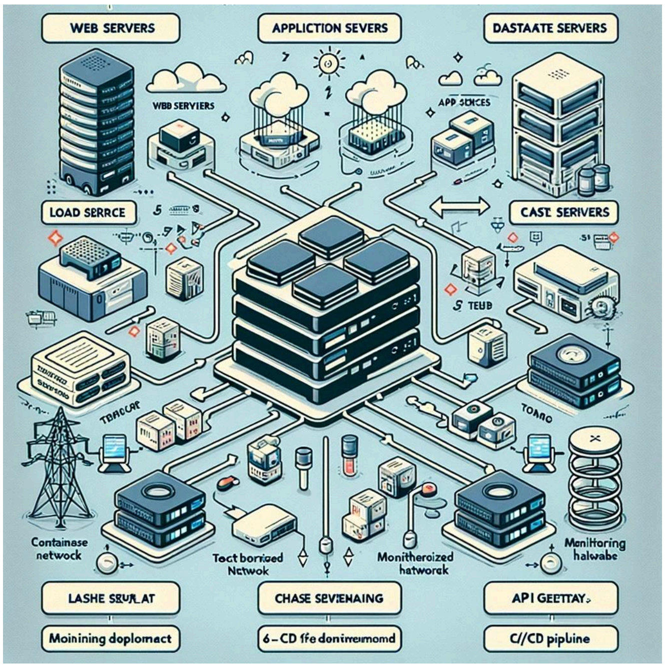
Execution Strategy

1.Test Execution:

Automation Framework Selection: Utilize a robust and scalable automation framework, such as Selenium WebDriver with Python, to automate test cases efficiently.
Continuous Integration (CI): Integrate automated tests into the CI/CD pipeline to ensure that tests are run automatically with each code commit, enabling early detection of defects.

2.Test Environment:

The Test Environment should be setup as figure below:



Multiple Test Environments: Utilize multiple test environments (e.g., development, staging, production) to simulate real-world conditions and ensure that tests are conducted in a controlled and isolated environment.
Environment Configuration Management: Ensure that test environments are configured identically to the production environment, to guarantee accurate test results and minimize environment-related issues.

3.Test Data Management:

Data Generation: Generate test data using automated tools to simulate real-world scenarios and test edge cases, ensuring comprehensive test coverage.
Data Reusability: Reuse test data across different test scenarios to reduce duplication and improve efficiency, ensuring that tests are executed with consistent and reliable data.

4.Test Execution Process:

Test Prioritization: Prioritize test cases based on risk and criticality, using techniques like risk-based testing, to focus testing efforts on high-impact areas and ensure that critical functionalities are thoroughly tested.
Test Execution Flow: Define the flow of test execution, including preconditions, test steps, and expected outcomes, using tools like TestRail, to ensure consistent test execution and facilitate test case reusability.
Defect Reporting: Report defects found during test execution using a standardized format, such as JIRA, to ensure that they are properly tracked, prioritized, and addressed, minimizing the impact on the project timeline and quality.

5.Test Coverage:

Coverage Analysis: Analyze test coverage reports to identify areas of the code that are not adequately covered by tests, ensuring that all critical functionalities are tested and validated.

TEST CRITERIA

- Suspension Criteria:

If the team members report that there are 20% of test cases failed, suspend testing until the development team fixes all the failed cases.

The tests are tests that constitute a better and higher quality user experience

- Exit Criteria:

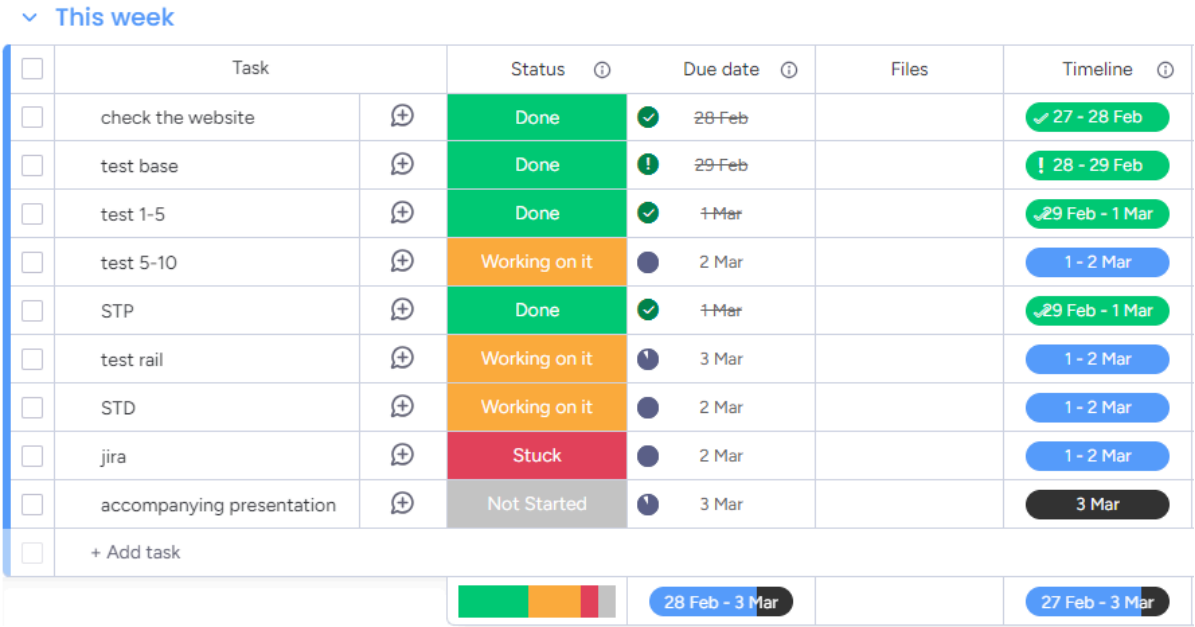
Specifies the criteria that denote a successful completion of a test phase

- Run rate is mandatory to be 100% unless a clear reason is given.
- Pass rate is 80%, achieving the pass rate is mandatory.

Human Resource

Member	**Tasks**
Project Manager	Oversee project, define goals, secure resources for website testing.
Test Team Lead	Lead team in test strategy, execution, defect reporting. Coordinate with outsourced teams. Verify and assess Test Approach.
Test Engineers	Develop and execute test cases, collaborate with developers for comprehensive coverage.
SQA Analysts	Oversee quality assurance, ensure alignment with requirements, provide continuous improvement feedback.

SCHEDULE & ESTIMATION



This week

□	Task		Status		Due date	Files	Timeline
□	check the website	⊕	Done	✓	~~28 Feb~~		✓ 27 - 28 Feb
□	test base	⊕	Done	!	~~29 Feb~~		! 28 - 29 Feb
□	test 1-5	⊕	Done	✓	~~1 Mar~~		✓ 29 Feb - 1 Mar
□	test 5-10	⊕	Working on it	●	2 Mar		1 - 2 Mar
□	STP	⊕	Done	✓	~~1 Mar~~		✓ 29 Feb - 1 Mar
□	test rail	⊕	Working on it	●	3 Mar		1 - 2 Mar
□	STD	⊕	Working on it	●	2 Mar		1 - 2 Mar
□	jira	⊕	Stuck	●	2 Mar		1 - 2 Mar
□	accompanying presentation	⊕	Not Started	●	3 Mar		3 Mar
□	+ Add task						
			🟩🟧🟥⬜		28 Feb - 3 Mar		27 Feb - 3 Mar