TEST PLAN

DAT Freight & Analytics
https://www.dat.com/

The Header Menu functionality

Table of Contents

1. INTRODUCTION

The Test Plan has been created to communicate the test approach to team members. This document describes approaches and methodologies that will apply to the Headers Menu functional testing for the website https://www.dat.com.

This document has clearly identified sub-tasks of website's module testing, types of testing, what devices, environments, and tools to be used. It includes test cases with expected and actual results, schedule and executing estimated time each of sub-tasks and entire project.

2. SCOPE

The document mainly targets the GUI, Exploratory, Functional Positive, Negative and Ad-hoc testing of the Headers Menu of website, Performance and Security testing of the website.

3. SUB-TASKS

3.1. WEBSITE MANUAL TEST

Header section for a website is very important displaying major functionality of website.

Execute manual testing for the Header Menu based on the next types of testing:
- GUI;
- Exploratory testing;
- Functional Positive testing;
- Functional Negative testing;
- Functional Ad-hoc testing.

GUI Testing:

GUI testing includes testing the UI part. It covers look and feel, error messages, spelling mistakes, GUI guideline violations.

Exploratory Testing:

Exploratory testing will include a type of software testing where Test cases are not created in advance but QA check system on the fly.

Functional Positive testing:

Checks whether an application behaves as expected with positive inputs.

Functional Negative testing:

Invalid data is inserted to compare the output against the given input.

Functional AD-hoc testing:

Includes an informal testing type with an aim to break the system.

For Functional testing execute the next manual tests:

Positive for the following options:
- Verify header section added on the website.
- Verify header section is aligned.
- Verify icons aligned with the text if added on the header.
- Verify icons added should be related to the link text.
- Verify links text should be readable or not.
- Verify the logo of the website added on the header is align or not.
- Verify the same header used on whole website pages.
- Verify header is sticky on the whole page or fixed as per requirements and design.
- Verify all links should be opening on the header or not.

- Verify on click on the link on the header the correct page should be open.

- Verify the home page should be open on clicking on the logo on all pages.

- Verify on mouse hover color change or not for the link added on the header menu.

- Verify selected header links should remain ACTIVE to inform the user about the clicked screen/page.

- Verify search bar is added on the header menu or not.

- Verify header and body section should not look the same. The header and body section must distinguish.

- Verify icon color should also change on mouse hover on the header if icons added.

- Verify dropdown options added on header shown on mouse hover or click.

Environment Support:
- OS: Windows 10/ 64
- Browsers (latest versions): Google Chrome, Firefox Mozilla, Microsoft Edge

Testing Tools used:

Process	Tools
Test case creation	Microsoft Word, Microsoft Excel, JIRA
Test case tracking	Microsoft Word, Microsoft Excel, JIRA
Test case execution	Manual
Test case management	Microsoft Excel, JIRA
Test reporting	Microsoft Excel, JIRA

3.2. WEBSITE AUTOMATION TEST

Automation tests are created by using Selenium Python UnitTest frameworks based on Functional manual test cases and executes to check functionality of Header Menu.

Environment Support:
- OS: Windows 10/ 64
- Browsers (latest versions): Google Chrome, Firefox Mozilla, Microsoft Edge

Testing Tools used:

Process	Tools
Test case creation	PyCharm, Selenium
Test case tracking	PyCharm, Browsers
Test case execution	Automation
Test case management	PyCharm
Test reporting	HTML reports, Allure reports

3.3. WEBSITE API TEST

API tests execute to determine whether the APIs that are developed meet expectations when it comes to functionality, performance, reliability, and security for website.

Environment Support:
- OS: Windows 10/ 64
- Browsers (latest versions): Google Chrome (DevTools-Postman Interceptor)

Testing Tools used:

Process	Tools
Test case creation	Postman, Chrome DevTools-Postman Interceptor
Test case tracking	Postman
Test case execution	Automation
Test case management	Postman
Test reporting	

3.4. WEBSITE AUTOMATION PPERFORMANCE TEST

Performance automation tests execute for measures the speed, responsiveness, and stability of the tested website. Tests execute in incognito environments.

After testing reports were generate.

Testing Tools used:

Process	Tools
Test case creation	Google Lighthouse, GTMetrix, BrowserStack-SpeedLab (OSXCatalina_Chrome110/ Windows10_Chrome110/ iPhone12Pro_Safari14)
Test case tracking	Google Lighthouse, GTMetrix, BrowserStack-SpeedLab
Test case execution	Automation
Test case management	
Test reporting	txt -, html -, pdf - files

3.5. WEBSITE AUTOMATION SECURITY TEST

Security automation tests execute to reveal potential flaws or weaknesses of software and website. Testing focuses on whether the application is designed and configured correctly.

Tests execute in incognito environments.

After testing reports were generate.

Testing Tools used:

Process	Tools
Test case creation	Mozilla Observatory, PentestTools-WebsiteScanner, ImmuniWeb
Test case tracking	Mozilla Observatory, PentestTools-WebsiteScanner, ImmuniWeb
Test case execution	Automation
Test case management	
Test reporting	txt -, html -, pdf - files

4.   TESTS SCHEDULE AND ESTIMATED TIME

Sub-Task Name	Start	Est. time (hours)	Finish
Manual Testing		1	
Automation Testing		16	
API Testing		20	
Performance testing		4	
Security testing		2	

Test Execution
QA   - Olga Vorontsova
Test phase   -
Status   -

APPROVALS:

	Project Manager	QA Lead
Signature		