Test Plan for Date Parser

1. Objective
   1. Purpose

This document describes the plan for testing the Date Parser application. This Test Plan document supports the following objectives:

* Identify the user inputs for the date that should be tested.
* Recommend and describe the testing strategies to be employed.
* List the Environment details and Platforms on which testing will be carried.
  1. Scope

This Test Plan describes the type of tests that will be conducted on Date Parser application. The listing below identifies the types of tests (functional requirements and non-functional requirements) that have been identified as targets for testing.

* Feature Testing
* Automation Testing
* Usability Testing
* Cross Platform Testing.

It is assumed that Date Parser website will not be used by large number of people at a time, so the performance test need not be covered.

Since, the website is not storing any sensitive info such as user credential, payment info, etc., the security testing is also not in scope.

1. Feature Details

The date parser takes in user input and returns a date that matches the intention of the user. The application is to help seniors, which should be easy for them to use, by taking the typed text input.

1. Requirements for Test

The listing below identifies use cases that have been identified as targets for testing. This list represents what will be tested and how will be tested.

* 1. Functional Testing

**Test Objective** -Ensure the functionality of Date parser is working fine. When user enter valid input, the user should get valid formatted date output. If the date is not valid, user should get appropriate message.

* Verify the Date formats.
* Verify the Date separators.
* Verify the Invalid Date scenarios.

**Technique** – Open the website and carry out the user scenarios mentioned in the testcases.

**Completion Criteria** – All the usecases are working as expected and bugs are logged for the defects found in the application.

**Special Consideration** – Verify some of the boundary cases of the date inputs.

**Resources –**

User Requirements - Requirements+for+Date+Parser.pdf

Testcase Repository – Testcases.xls

* 1. Automation Testing

**Test Objective** - Automate all the testcase identified in functional testing and generate a report of passed and failed usecases.

**Technique** – Clone and run the automation as per instructions mentioned in the Readme.md

<https://github.com/madhubalajain/dateparser>

**Completion Criteria** – All usecases are automated.

**Special Consideration** - The automation should be easy to run.

**Resources -**

TestLogs - [TestExecution.logs](https://github.com/madhubalajain/dateparser/blob/master/TestExecution.logs)

TestResults - src/test/resources/testdata/testdata.xlsx

* 1. Usability Testing

**Test Objective** - Verify the website is user friendly to support seniors to use the application and suggest improvements to product owner/Developers.

**Technique** – Verify the application is easy to use by seniors and have sufficient tips.

**Completion Criteria** – All improvements which are logged, should to be reviewed by product owner and developers.

* 1. Cross Platform Testing

**Test Objective** - Verify the website is working fine on supported browsers and operating system.

List of supported browser and platforms:

\*\*The section is expected to be modified based on feedback from product owner.

**Technique –** Run the test suite in Selenium Grid.

**Completion Criteria** – Application is working as expected on all supported browsers and platform.

1. Test Environment –

Build Info -

URL – <https://vast-dawn-73245.herokuapp.com/>

1. Defects Tracking –
2. Test Schedule –
3. Conclusion –

The document details the Test Plan and Test Strategies for testing the Date Parser App.