**Test plan for Trello**

1. **Introduction**

This test plan describes the testing approach and overall framework that will drive the testing of Trello project. The document introduces the scope and identifies the items being tested, the features to be tested, the testing tasks to be performed, and the risks associated with this plan.

1. **Quality Objective**

Main goals of testing for current project to verify next:

* Any user should be able to interact with Trello website (<https://trello.com/>) through the UI
* Any user should be able to interact with Trello website (<https://trello.com/>) through the API

1. **Features to be tested**

Functional,UI and API features of existing project should be covered during testing activities. This include next functionals:

* Registration
* Login
* Token generation
* Board CRUD activities
* Card CRUD activities
* List CRUD activities and etc.

1. **Features not to be tested**

Any third-party integrations with Trello and new features

1. **Testing types**

**5.1 Functional testing**

that is performed based on functional requirements provided from official documentation and verifies the system against them.

5.2 **Regression Testing**

will be performed after new components/features and defect fixes are into the existing system to validate that unchanged components (that were implemented earlier) still work as originally intended

**5.3 Non-Functional** is the testing of a software application or system for its non-functional requirements: the way a system operates, rather than specific behaviors of that system:

* + **Localization** will be performedfor particular target locale and language
  + **UI** technique that verifies graphical user interface including screen validations, all navigations and etc.
  + **Compatibility** conducted on the application to evaluate the application's compatibility within different environments. For desktop suggested next list of browsers/OS

### 

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Desktop | | | | | |
|  | **Chrome 86** | **Chrome 87** | **Safari 14** | **Edge** | **Firefox 84** |
| OSX | X | **X** | X |  |  |
| Windows | X | **X** |  | X | X |

* + **Usability** technique that allow to evaluate how easily system can be used by end users
  + **Accessibility testing** technique will be performed to ensure that system is usable by people with disabilities
  + **Performance**  technique to evaluate behavior of an application under varying loads. It can be performed using testing tools such as HP LoadRunner and Apache JMeter.
  + **Security** will be performed to ensure to check basic vulnerabilities.Given below is the list of security testing tools that can be used for such kind of testing:
  + **Wapiti** is the open-source application that scans web pages and injects the testing data to check for the security lapse. Support the GET and POST HTTP attacks to check next vulnerabilities: file disclosure, database injection, XSS injections and etc.
  + Zed Attack Proxy tool supported on different OS and have next key features: automatic scanner, authentication support, AJAX spiders and so on.

**5.4 Automation testing**

Automated functional testing of Trello is highly recommended because of the following reasons:

* big number of supported browsers and devices that will increase smoke/regression testing scope
* two types of orientations (landscape and portrait) for each mobile device will double smoke/regression testing scope

Automation approach can be applied for smoke, regression, and cross -browsers and cross-platform testing. It will save time and costs for testing of newly implemented features as well.

Test Automation on Trello can be achieved using a combination of the following open-source tools and technologies:

* Selenium WebDriver (.Net/C# libraries);
* Allure or Extent report as a reporting tool;
* SpecFlow or other BDD framework;

1. **Deliverables**

The following are the test deliverables for this project:

* Test plan document
* Functional test case
* Auto-tests
* Bug reports
* Test reports