Test plan

Trello board

Prepared by Dmytro Moskalenko

16.03.2023

Introduction

The Trello is one of the most popular visual tools that empowers your team to manage any type of project, workflow, or task tracking. One of the base functionalities of Trello is a board object. Current test plan is built to create full and comprehensive test coverage and test strategy for CRUD (Create/Read/Update/Delete) functionality of current.

Objectives

Objectives

- Create test coverage for CRUD functionality of Trello board
- Create test plan with above test that will identify Back-End and Front-End coverage
- Automate 3 test cases both from back end and front end part
- Provide automation run test report

Tasks

- Create manual test cases that will cover CRUD functionality for Trello Board in checklist view
- Prioritize the test cases and identify witch should be implemented on back-end and witch on front-end
- Create automation framework for REST API using REST Assure
- Automate back-end tests with above framework
- Create automation framework for UI test automation using Selenium framework
- Automate UI tests with above framework
- During automation implement Gherkins layer based on Cucumber framework
- Set up reporting for test run

Scope

- We are going to cover CRUD operation on board as well as CRUD operation on objectives that can be part of the board.
- Positive and negative test coverage should be identified
- Automation should support different browser as well as different operation systems

Testing Strategy

- We require to cover all decision table of Board module as well as all modules that could be part of the board

- We have to include into test scope the boarder values approach for text filed as well as for different capacities of the system (for example maximum amount of object that could be created)
- Data injection tests
- We need to check that system behaves predictable on nonstandard situations, predict corner cases that used can faced with
- UI test will cover scenarios:
    - Requiring UI validation
    - Checking UI elements presents
    - All UI element interaction at least once
    - Scenarios that output Reports/Warnings/Output messages
- API tests will cover scenarios:
    - That require multi-creation of any objects
    - That include interactions that is already covered by UI tests
    - As a part of UI tests for clean-up created objects, setting up pre-requisites
    - Part of smoke integration tests that require quick respond about product quality
    - Test API calls that can be provided with wrong input data
- Will not be covered:
    - Each board related call specification. Expected to be covered on unit testing level.
    - Load/stress testing should be part of additional strategy document
    - Security testing – should be done on higher level of test strategy
    - Premium features

Alpha Testing (Unit Testing)

N/A

System and Integration Testing

- System testing as a testing of whole system from perspective of current scope, it will be implemented as a several automation scenarios that will cover e2e scenarios

Performance and Stress Testing

N/A

User Acceptance Testing

- UAT will be created as a manual/automation test that will cover UAT component testing

Batch Testing

N/A

Automated Regression Testing

N/A

Beta Testing

N/A

Environment requirements

Operation systems

We going to test support of:

- Windows, starting from Windows XB
- Linux (Versions and typed TBD)
- Mac OS (Version TBD)
- Android (Version TBD)

Hardware

- TBD

Feature to be tested

- CRUD operations on Lists
- CRUD operations on Cards
- CRUD operations on Board