Test Scenario (TS) 1: Check if the restrictions on sign up functionality are working. The only restrictions on sing up fields are in e-mail and password. Motive: Avoid sing up with invalid data (Test name in Visual Studio: **SingUp**)

* Test Case (TC) 1: Check system behavior when invalid email is entered;
* TC 2: Check system behavior when invalid or valid password is entered;

TS 2: Check if sign in and log out functionalities are working. Motive: Avoid that the system accepts login with invalid password, username or both, and when logged, assert log out functionality. (Test name in Visual Studio: **SignInSignOut**)

* TC 1: Check system behavior when valid email and invalid password is entered;
* TC 2: Check system behavior when invalid email and valid password is entered;
* TC 3: Check system behavior when valid email and password is entered;
* TC 4: Check system behavior when log out is made.

TS 3: Check home page links and images. Motive: There might be some broken images or links that can be detected and avoid a negative user experience. (Test name in Visual Studio: **CheckHomepage**)

* TC 1: Check if all links are working (through status code);
* TC 2: Check if any image is broken;

TS 4: Check if Topics page is working as supposed. Motive: when a popular topic is chosen, the url associated may not be referring to the topic. Also, ensures that the star button is not available without login, and the topics title are sorted correctly by default. (Test name in Visual Studio: **CheckTopicsAndTrendings**)

* TC 1: When clicking a popular topic, check system behavior (the URL should contain the popular topic, as well as the title of the page);
* TC 2: Among the featured topics, check system behavior when clicking the Star button (check if it asks to log in);
* TC 3: Topic titles seem to be sorted in an ascending order. Check if the page present this behavior correctly.

TS 5: Check if search engine is working correctly. Motive: the search engine may not be working as expected with strange inputs. In addition, it is checked if the sorting criteria are working as expected. (Test name in Visual Studio: **CheckSearchEngine**)

* TC 1: Enter only special characters that should not return any results;
* TC 2: Enter only numbers and only letters;
* TC 3: Enter only ‘%’, that should not redirect to 404 ERROR;
* TC 4: Enter alphanumeric characters and special characters;
* TC 5: Enter string  in search box with spaces (before string, after string and in between);
* TC 6: Try a input that should return results (like a vogal);
* TC 7: For the TC 6 returned results, check if sorting criteria is working;
* TC 8: When selecting a certain language, check if the results are presented as expected.

TS 6: Check if Trending page is working as supposed. Motive: trending page presents different kinds of filters that must be tested, and adding a trending page through star button functionality is tested when logged (Test name in Visual Studio: **CheckTrending**)

* TC 1: Check if filters are working (programming language and date);
* TC 2: Try to click a Star button;
* TC 3: Go to “Your Stars” menu, and check if the element added is present;
* TC 4: Unstar the previous element, refresh the page, and check if the element is no longer there.

TS 7: Assert if Pricing plans is working as expected. Motive: the testimonials may not belong to the right companies, so this association must be tested. (Test name in Visual Studio: **CheckPricingPlan**)

* TC 1: Assert if the message presented by other companies refer to them.

TS 8: Assert if Appearance menu is working as expected. Motive: assert if changing theme or default emoji are working as expected. (Test name in Visual Studio: **CheckAppearance**)

* TC 1: Assert if, when changing the theme, the page changes the theme color;
* TC 2: Assert if preferred default emoji is changed when saving another default.

TS 9: Assert if repository creation if working as expected. Motive: assert if repository basic operations are working (add, edit and delete), because they are one of the main components of GitHub. (Test name in Visual Studio: **CheckRepoOperations**)

* TC 1: Assert if a repository can be created;
* TC 2: Check if a repository with the same name cannot be created;
* TC 3: Change the repository name and check if it was correctly changed. Ensure that the same name cannot be accepted as a change;
* TC 4: Delete the repository and check if it was successful.

TS 10: Assert if profile data is working as expected. Try different data in each filter to check if one field may have the value of another. Motive: When changing the profile data, there may not be an update of information. Also, the association between the input in the edited profile data and the output of the current profile data are matched, to ensure that they are correctly associated. (Test name in Visual Studio: **ProfileData**)

* TC 1: Enter only special characters in data fields;
* TC 2: Enter only numbers and only letters;
* TC 3: Enter alphanumeric characters and special characters;
* TC 4: Enter string  in search box with spaces (before string, after string and in between);
* TC 5: Check if its correctly displayed in profile description;