Testing

This page outlines the different types of testing conducting to verify the user stories implemented.

Table of contents

Since, the first sprint consisted of user stories focused on changing the user interface, we decided to only focus on User Acceptance Testing. It was conducted by the clients to ensure that the implementations meet their requirements.

Acceptance Testing

The document displays a table outlining the user acceptance tests conducted for the user stories implemented so far. Each user story has an acceptance criteria which was collected from the clients/users and during the tests, the expected outcome was noted. If the expected outcome satisfies all the criteria, the test is a pass else it is a fail.

Version 1.0

Story ID	Tag	Acceptance Criteria	Expected Outcome	Pass /Fail
01	US_01_ADD_MODES	• Student users should be able to view Insert and Search as modes for the Binary Search Tree algorithm to differentiate between the two functionalities.	• Labels in the instructions can be viewed as "Insert Mode" and "Search Mode". • Instructions containing steps to be followed for each mode can be viewed.	PASS
02	US_02_LABEL_SPEED_SLIDER	• Student users should be able to view a label named "Speed" next to the speed slider to be able to understand its functionality.	• A label "SPEED" can be viewed on the left of the slider.	PASS
03	US_03_ADD_PROGRESS_BAR	• Student users should be able to view the progress bar next to the play button. • Student users should be able to view the progress of the algorithm in % after clicking the play button. • The progress bar must return to 0% every time a new set of parameters are loaded by the user.	• The progress bar is visible next to the play button. • It loads the progress in % after the play button is clicked. • The bar resets i.e. it returns to 0% after a new set of input parameters are inserted by the user.	PASS
06	US_06_CLOSE_NESTED_BLOCKS	• Student users should not need to collapse a nested block when the respective parent block is collapsed. • The nested blocks must collapse automatically when the parent block is collapsed by the user.	• The nested blocks collapse automatically when a parent block is collapsed. • The user is expected to manually expand the nested blocks again.	PASS
17	US_17_CHANGE_LABELS_FOR_VIEWS	• Student users should be able to view the labels as "Array view" and "Tree view" for the Heapsort algorithm.	• Tree view and Array view are visible for users to view when using the Heapsort algorithm	PASS
18	US_18_LABEL_GRAPH_SIZE	• Student users should be able to clearly identify the buttons for increasing and decreasing the graph size and use them to edit the graph size.	• The button to increase the graph size is labelled as "increase graph size". • The button to decrease the graph size is labelled as "decrease graph size".	PASS
19	US_19_CHANGE_LOAD_BUTTON	• Student users should be able to locate which button can build the graph based on the matrix in control panel easily.	• The button "LOAD" is renamed as "BUILD GRAPH" for easy identification.	PASS
20	US_20_ADD_RESET_BUTTON	• Student users should be able to update the graph if changes are made to the matrix.	• If there exists a built graph, the "BUILD GRAPH" label is automatically renamed as "UPDATE". • The "UPDATE" button can be used to load the graph with updated values in the matrix.	PASS