Software Test Report (STR)

Software It Counts (SWIC)

CMSC 447

Updated on November 20, 2015

Table of Contents

[1 Scope 3](#_Toc432634202)

[1.1 Identification 3](#_Toc432634203)

[1.2 System overview 3](#_Toc432634204)

[1.3 Document overview 3](#_Toc432634205)

[2 Referenced documents 3](#_Toc432634206)

[3 Overview of test results 4](#_Toc432634207)

[3.1 Overall assessment of the software tested 4](#_Toc432634208)

[3.2 Impact of test environment 4](#_Toc432634209)

[3.3 Recommended improvements 4](#_Toc432634210)

[4 Detailed test results 4](#_Toc432634211)

[4.1 (Project-unique identifier of a test) 4](#_Toc432634212)

[4.1.1 Summary of test results 4](#_Toc432634213)

[4.1.2 Problems encountered 5](#_Toc432634214)

[4.1.3 Deviations from test cases/procedures 5](#_Toc432634215)

[5 Test log 5](#_Toc432634216)

[6 Notes 6](#_Toc432634217)

[A. Appendixes 6](#_Toc432634218)

1. Scope

This section shall be divided into the following paragraphs.

* 1. Identification

For this project, we will be working with the web application, Parable of the Polygons. The software will simulate segregation levels between three different shapes, allowing them to move to different locations on the board in an attempt to be happy. The website will be programmed in HTML while the application will be programmed in JavaScript. The repository is located at https://github.com/Kirkas1/polygons<https://github.com/Kirkas1/polygons>, and branches off the source code at [https://github.com/dncnmcdougall/polygonshttps://github.com/dncnmcdougall/polygons](https://github.com/dncnmcdougall/polygons).https://github.com/Kirkas1/polygons/tree/gh-pages/documents

* 1. System overview

This project has tasked us with inserting a 3rd polygon, a red circle, into the game, allowing for relations between 3 different shapes to be shown. In addition, two new algorithms will be created. One such algorithm will be based around movement when based on the happiness levels of single polygons, henceforth known as the “happiness algorithm.” The second algorithm will be based around the happiness of single polygons, as well as the happiness of those in the 8 squares adjacent to them, henceforth known as the “collective happiness algorithm.”

The following document will address the test results produced by the system (in fulfillment of the requirements given by the customer Russ Cain). The subsequent test report for the project will thoroughly detail the test outcomes carried out by the group (SWIC). This document will also be uploaded to the project’s repository at https://github.com/Kirkas1/polygons <https://github.com/Kirkas1/polygons>, which has the history of system development thoroughly summarized<https://github.com/Kirkas1/polygons/tree/gh-pages/documents>.

* 1. Document overview

This document will assess the testing of the system and the results of each test (based on requirements for the project, as previously discussed by customer). The overview of the testing shall include an overall assessment of the software tested and recommended improvements. It will also discuss each test and result given by the STD, any problems encountered, and any deviations from the test case. The test results and handling shall be carried out by the group (SWIC), and the findings shall be traced back to the requirements and then published.

1. Referenced documents
2. Parable of Polygons, Revised Apr 18, 2015, <http://ncase.me/polygons/><https://github.com/ncase/polygons><http://ncase.me/polygons/>, Vi Hart and Nicky Case
3. Parable of Polygons Source Code, Revised Oct 25, 2015, [https://github.com/ncase/polygonshttps://github.com/ncase/polygons](https://github.com/ncase/polygons), Vi Hart and Nicky Case
4. Polygons, Revised Dec 9, 2014, <https://github.com/dncnmcdougall/polygons><https://github.com/ncase/polygons>, Duncan McDougall
5. Overview of test results

This section shall be divided into the following paragraphs to provide an overview of test results.

* 1. Overall assessment of the software tested

The tests able to be performed were successful except. The Red Circle Test was completed without flaws. The preliminary tests for the Happiness algorithms were complete given by the fact that they are selectable options on the website and had full/partial implementation. However they could not be completed in full as the algorithms have bugs that prevent the algorithms from working 100% of the time. The Customized Slider Test will be conducted when that phase of development is complete.

* 1. Impact of test environment

All tests were done on machines with similar environments to that which they will be run on with the exception of adding the radio buttons. That test was done on a computer running Linux, but the radio buttons proved to be functional on Windows as well.

* 1. Recommended improvements
* Fix bugs in the Happiness and Collective Happiness Algorithms. Design and implement the three-sided slider so that it may be properly tested.

1. Detailed test results

This section shall be divided into the following paragraphs to describe the detailed results for each test. Note: The word "test" means a related collection of test cases.

* 1. (Project-unique identifier of a test)

The following test results will be ordered by test: Red Circle Test, Random Algorithm Test, Happiness Algorithm Test, Collective Happiness Algorithm Test, and Customized Slider Test.

### Summary of test results

**Red Circle Test** – All results as expected. A third shape, a red circle, was implemented onto the polygon board with the same functionality as the other shapes. The red circle animates and interacts with the board environment as intended.

**Random Algorithm Test** – All results as expected. The radio buttons accurately default to the “Random” algorithm upon page loading. Likewise, the “Random” algorithm mimics random polygon movement when said polygon is unhappy. The “Random” algorithm was implemented and tested as intended.

**Happiness Algorithm Test** – Problems encountered, infinite loops. The radio buttons accurately list a “Happiness” algorithm that is selectable. The “Happiness” algorithm segregates the polygons more efficiently than the “Random” algorithm. However, during some of the trials, an infinite loop occurs on the last unsatisfied polygon where it moves between two spaces where it is unsatisfied, breaking the program. The “Happiness” algorithm results were not as intended and therefore not a success.

**Collective Happiness Algorithm Test** – Problems encountered, freezes the grid. The radio buttons accurately list a “Collective Happiness” algorithm that is selectable. However, upon selecting “Start Movin'” with “Collective Happiness” selected, the characters on the grid freeze until either another algorithm is selected, or “Stop Movin'” is pressed. The “Collective Happiness” algorithm results were not as intended and therefore not a success.

**Customized Slider Test** – Not applicable, incomplete. The customized slider used to assess polygon bias between shapes (for happiness levels) is currently not implemented on the board interface. There is currently nothing to test which is unexpected, and the slider is still stuck in design. Thus, the results of this test are not available and therefore incomplete.

### Problems encountered

During the testing process for the red circle it was first noted that the circles were not transparent. This occurred in our initial test. However this was a change required to the art assets and not to the code. This was changed once by making the background transparent as opposed to white (which was thought to symbolize transparent at the time. Likewise, trouble implementing both the “Happiness” algorithms occurred via bugs causing the algorithms to not finish the polygon simulation.

### Deviations from test cases/procedures

Formal means of testing the efficiency of the new algorithms against the Random one were planned in the STP. This did not occur since they were not required to be more efficient (see SRS page 4).

1. Test log

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test | Date | Location | Description/Results | Hardware/software configurations | Performers |
| Integrating red circle | November 20, 2015  8 PM-10 PM | UMBC Library, on each team member’s laptop | Integrated a third shape, a red circle, to the sandbox. The red circle acted like the other shapes (e.g. moved if a certain amount of shapes were around it, and shook like the other shapes). | Latest version (as of November 2015) of Fire Fox, Internet Explorer, and Chrome on laptops with Windows 7 or Linux | Team SWIC |
| Adding the selected algorithm radio buttons | November 21, 2015  1 PM | Ian’s laptop | Added three radio buttons for each algorithm (random, happiness, collective happiness) | Latest version (as of November 2015) of Fire Fox, Internet Explorer, and Chrome on laptops with Windows 7 or Linux | Team SWIC |
| Working on Internet Explorer, Fire Fox, and Chrome | November 23, 2015  7 PM | All of Team SWIC’s laptops | Tested to see if the code ran on the browsers listed. Only successful on Firefox. | Latest version (as of November 2015) of Fire Fox, Internet Explorer, and Chrome on laptops with Windows 7 or Linux | Team SWIC |
| Happiness Algorithm | December 7, 2015 12pm | James' Laptop | Selected the Happiness algorithm and ran automatic.js on the bottom of index.html. Failure after discovery of infinite loop. | Latest version (as of December 2015) of Fire Fox, Internet Explorer, and Chrome on laptops with Windows 7 or Linux | Team SWIC |
| Collective Happiness Algorithm | December 7, 2015 12PM | James' Laptop | Selected the Happiness algorithm and ran automatic.js on the bottom of index.html. Failure as it froze the grid until selection of another algorithm. | Latest version (as of December 2015) of Fire Fox, Internet Explorer, and Chrome on laptops with Windows 7 or Linux | Team SWIC |