Software Testing Report

< Victoria Road Crash Data Software>

Group 52

Saikat Dutta Tanu – s5234269

s5219071 Inho Kim

Table of Contents

[1.0 Unit Tests 3](#_Toc49779837)

[2.0 Coverage Report 4](#_Toc49779838)

[3.0 Requirements Acceptance Testing 5](#_Toc49779839)

# Unit Tests

Delete the RED text and replace with your own

(In this table you fill out details about what unit tests you have done using the unittest module)

| **No** | **Test Case** | **Expected Results** | **Actual Results** |
| --- | --- | --- | --- |
| **1.0** | **wordcount Functions** |  |  |
| 1.1 | Tested file with a wrongname | Exception Handled | Exception Handled |
| 1.2 | Tested with empty input file | Displayed error message and exit | Displayed error message and exit |
| **2.0** | **Histogram Functions** |  |  |
| 2.1 | Empty input dictionary | Displayed error message and exit | Displayed error message and exit |

# Coverage Report

The unit test was done for functions used in this software program. First, the unit test was done for reading the csv file properly. The testing was done to check whether the csv file name, if put wrong, can be detected by the program. It passed the test. For further verification, the unit test was done for all the attributes the csv file is having and the program also passed the test. When wrong file name was provided, the unit test successfully returned the warning message and ignored the input. When the input file name was empty, the program was also able to display error message and exit.

Requirements acceptance testing was observed for four aspects. They are accepting multiple file names as arguments from the command line, displaying an appropriate message if a file does not exist or if a file name is invalid, File name can be a simple file name or include the full path of the file with one or more levels, the program should be able to accept as many levels for each file name as the user wants to input, and this is limited only by the number of levels allowed in Windows (approximately 120). The first three aspects were passed but the 4th one failed.

# Requirements Acceptance Testing

(You will need to fill out the column on the left with the requirements listed in software design documents and the columns on the right with the results of your own testing)

| **Software**  **Requirement No** | **Test** | **Implemented (Full /Partial/ None)** | **Test Results (Pass/ Fail)** | **Comments (for partial implementation or failed test results)** |
| --- | --- | --- | --- | --- |
| 1 | Accept multiple file names as arguments from the command line | Full | Pass |  |
| 2 | Display the details of all valid files | None |  |  |
| 3 | Display an appropriate message if a file does not exist or if a file name is invalid | Full | Pass |  |
| 4 | Display a message if an argument is a directory instead of a file | None |  |  |
| 5 | File name can be a simple file name or include the full path of the file with one or more levels | Full | Pass |  |
| 6 | file names must start with an alphabetical character | None |  |  |
| 7 | Valid file name extensions must be 3 or 4 alphabetical characters preceded by a dot) | None |  |  |
| 8 | Directory/level names must start with an alphabetical character to be considered valid | None |  |  |
| 9 | The program should be able to accept as many levels for each file name as the user wants to input. This is limited only by the number of levels allowed in Windows (approximately 120) | Full | Fail |  |