Software Testing Report

<2810ICT, Part B>

William Crane, Zak Cobham-Davis, Christopher

Table of Contents

[1.0 Unit Tests 3](#_Toc49779837)

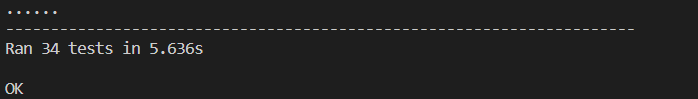
[2.0 Coverage Report 4](#_Toc49779838)

[3.0 Requirements Acceptance Testing 5](#_Toc49779839)

# Unit Tests

| **No** | **Test Case** | **Expected Results** | **Actual Results** |
| --- | --- | --- | --- |
| **1.0** | **Dataset Tests** |  |  |
| 1.1 | test\_loadCSVData | Tests if the dataset loadcsvdata method can accept an invalid path. | Tests the dataset loadcsvdata method and whether it can accept an invalid path. |
| 1.2 | test\_getData | Tests if the program can handle a request to return the dataset even if it is not in memory. | Tests the program to handle a request to return the dataset even if it is not in memory. |
| **2.0** | **ReportTest** |  |  |
| 2.1 | test\_getReportDataEmpty | To Test the program to determine whether the report Data is Empty within the Data frame. | Tests the program to determine whether the report Data is Empty within the Data frame. |
| 2.2 | test\_getReportDataFull | To Test whether a Data frame has been filled with Report Data. | Tests whether a Data frame has been filled with Report Data. |
| 2.3 | test\_generatePlot | To Test the generatePlot method without data and no trend function | Tests generatePlot method without data and no trend function |
| 2.4 | test\_generatePlotIsTrend | To Test the generatePlot method without data and trend function | Tests generatePlot method without data and trend function |
| 2.5 | test\_generatePlotWithValues | To Test the generatePlot method with data and no trend function | Tests generatePlot method with data and no trend function |
| 2.6 | test\_generatePlotIsTrendWithValues | To Test the generatePlot method with data and trend function | Tests generatePlot method with data and trend function |
| 2.7 | test\_reset | To Test the reset method for clearing the dataframe | Tests reset method for clearing the dataframe |
| **3.0** | **AlgorithmAllOffenceTest** |  |  |
| 3.1 | test\_allOffenceIncorrectTypeStartDate | Test incorrect type on start date input for allOffence method in Algorithm class | Tests incorrect type on start date input for allOffence method in Algorithm class |
| 3.2 | test\_allOffenceIncorrectTypeEndDate | Test incorrect type on end date input for allOffence method in Algorithm class | Tests incorrect type on end date input for allOffence method in Algorithm class |
| 3.3 | test\_allOffenceIncorrectTypeIsMobile | Test incorrect type on isMobile input for allOffence method in Algorithm class | Tests incorrect type on isMobile input for allOffence method in Algorithm class |
| **4.0** | **AlgorithmDistributionTest** |  |  |
| 4.1 | test\_distributionIncorrectTypeStartDate | Test incorrect type on start date input for distribution method in Algorithm class | Tests incorrect type on start date input for distribution method in Algorithm class |
| 4.2 | test\_distributionIncorrectTypeEndDate | Test incorrect type on end date input for distribution method in Algorithm class | Tests incorrect type on end date input for distribution method in Algorithm class |
| 4.3 | test\_distributionIncorrectTypeIsMobile | Test incorrect type on isMobile input for distribution method in Algorithm class | Tests incorrect type on isMobile input for distribution method in Algorithm class |
| **5.0** | **AlgorithmInvolvingRadCam** |  |  |
| 5.1 | test\_involveRadCamInvalidTypeStartDate | Test incorrect type on start date input for involveRadCam method in Algorithm class | Tests incorrect type on start date input for involveRadCam method in Algorithm class |
| 5.2 | test\_involveRadCamInvalidTypeEndDate | Test incorrect type on end date input for involveRadCam method in Algorithm class | Tests incorrect type on end date input for involveRadCam method in Algorithm class |
| 5.3 | test\_involveRadCamInvalidTypeIsMobile | Test incorrect type on isMobile input for involveRadCam method in Algorithm class | Tests incorrect type on isMobile input for involveRadCam method in Algorithm class |
| **6.0** | **AlgorithmSingleOffenceTrend** |  |  |
| 6.1 | test\_singleOffenceTrendInvalidTypeStartDate | Test incorrect type on start date input for singleOffenceTrend method in Algorithm class | Tests incorrect type on start date input for singleOffenceTrend method in Algorithm class |
| 6.2 | test\_singleOffenceTrendInvalidTypeEndDate | Test incorrect type on end date input for singleOffenceTrend method in Algorithm class | Tests incorrect type on end date input for singleOffenceTrend method in Algorithm class |
| 6.3 | test\_singleOffenceTrendInvalidTypeIsMobile | Test incorrect type on isMobile input for singleOffenceTrend method in Algorithm class | Tests incorrect type on isMobile input for singleOffenceTrend method in Algorithm class |
| 6.4 | test\_singleOffenceTrendInvalidTypeOffenceCode | Test incorrect type on offence code input for singleOffenceTrend method in Algorithm class | Tests incorrect type on offence code input for singleOffenceTrend method in Algorithm class |
| **7.0** | **ControllerTest** |  |  |
| 7.1 | test\_generateAllOffenceIsMobile | Test for generating all offence involving mobile report | Tests for generating all offence involving mobile report |
| 7.2 | test\_generateAllOffenceNotMobile | Test for generating all offence report | Tesst for generating all offence report |
| 7.3 | test\_generateRadCamIsMobile | Test for generating offences involving radar/cameras and involving mobile phones report | Tests for generating offences involving radar/cameras and involving mobile phones report |
| 7.4 | test\_generateRadCamNotMobile | Test for generating offences involving radar/cameras report | Tests for generating offences involving radar/cameras report |
| 7.5 | test\_generateDistIsMobile | Test for generating distribution of offences involving mobile report | Tests for generating distribution of offences involving mobile report |
| 7.6 | test\_generateDistNotMobile | Test for generating distribution of offences report | Tests for generating distribution of offences report |
| 7.7 | test\_generateTrendIsMobileOffenceCode | Test for generating trend of single offence involving mobiles report | Tests for generating trend of single offence involving mobiles report |
| 7.8 | test\_generateTrendIsMobileNoOffenceCode | Test for generating trend of all offences report | Tests for generating trend of all offences report |
| 7.9 | test\_generateTrendNotMobileOffenceTrend | Test for generating trend of one offence report | Tests for generating trend of one offence report |
| 7.10 | test\_generateTrendNotMobileNoOffenceCode | Test for generating trend of all offences involving mobile phone report | Tests for generating trend of all offences involving mobile phone report |
| 7.11 | test\_incorrectTypeReportID | Test for invalid input of report ID type | Tests for invalid input of report ID type |
| 7.12 | test\_outOfRangeReportID | Test for invalid input of report ID range | Tests for invalid input of report ID range |

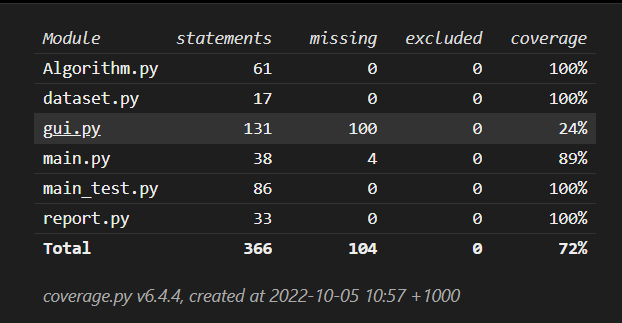
These unit tests all pass with the final version of the program



# Coverage Report

Unit testing of the Graphical User Interface (GUI) Design and Data Display has helped to ensure that the functionality has been provided as a means of verification for refactoring efforts, before development began coverage of the unit testing was broken up into 5 critical coverage criteria function coverage, statement coverage, path coverage, branch and decision coverage and condition coverage. Moreover, these criteria were all considered in the development of the unit tests whether that was conditional coverage that evaluated True or False conditions which can be seen in Unit Tests No 2.4 or 2.5 or Functions that could be called from the source code, and executed at least once (Functional Coverage), something that can be viewed in 2.3 or 2.6. Furthermore, the testing covers seven aspects of the program the Data Set, Reports, Offence Function, Distribution Function, Radar and Camera Function, Single Offence Trends and controller tests. This testing covers all the major files and functions within the Program and reviews a plethora of potential errors that could arise within the program, all varying in type of coverage based on requirement.

We have included a copy of the output coverage report. This shows full coverage in all files and functions with exception to those referring to the GUI. These were not appropriate to test as they are generating visual elements and not testable data.



# Requirements Acceptance Testing

| **Software  Requirement No** | **Test** | **Implemented (Full /Partial/ None)** | **Test Results (Pass/ Fail)** | **Comments (for partial implementation or failed test results)** |
| --- | --- | --- | --- | --- |
| **REQ1** | The main UI of the system shall display a drop-down to select the required feature, a checkbox to limit the results to offences involving mobile phone usage, date inputs for start and end date, a textbox for text entry where applicable, and a submit button. | Implemented Full | Pass | An Offence Code Text Box was added in to support the functions |
| **REQ2** | The system shall only accept numerical input from the user where applicable. | Implemented Full | Pass | Date Range |
| **REQ3** | The main screen of the system shall display the name of the program and a list of group members. | Implemented Full | Pass |  |
| **REQ4** | The submit button on the main screen of the system shall be disabled until the user inputs for the feature are valid. | Implemented Full | N/A | By default, the system already includes a set of valid inputs and it is impossible to enter invalid data achieving this requirement automatically |
| **REQ5** | Upon clicking/pressing the submit button on the main screen of the system, a results screen shall display. | Implemented Full | Pass |  |
| **REQ6** | The results screens shall be independent from the system’s main screen | Implemented Full | Pass | The Tkinter Package offers this within its display functionality |
| **REQ7** | Upon clicking the exit button on the results screen, the results window should close leaving the main screen only | Implemented Full | Pass | The Tkinter Package offers this within its display functionality |
| **REQ8** | Results screens which display a list of information on penalties shall display an interactive scrollable panel where the information is displayed. | Implemented Full | Pass | The Tkinter Package offers this within its display functionality |
| **REQ9** | Results screens that display charts shall include buttons for interacting with the appearance of the chart. | Implemented Full | Pass | The Tkinter Package offers this within its display functionality |
| **REQ10** | The “Change Display” button on the results screen shall toggle the displayed graph on the results screen between bar- and pie-chart. | Implemented Full | Pass | The Tkinter Package offers this within its display functionality |
| **REQ11** | Results screens which display trend over time for penalties shall display a bar chart. | Implemented Full | Pass |  |
| **REQ12** | The start and end date inputs will be filled with the dates of the earliest and latest offences recorded respectively by default. | Implemented Full | Pass |  |