*INstitute Security Management*

Test report

Version <

<

VERSION HISTORY

[Provide information on how the development and distribution of the Test Report was controlled and tracked. Use the table below to provide the version number, the author implementing the version, the date of the version, the name of the person approving the version, the date that particular version was approved, and a brief description of the reason for creating the revised version.]

|  |  |  |
| --- | --- | --- |
| **Version #** | **Implemented**  **By** | **Revision**  **Date** |
| 1.0 | *12* | *<05/04/16>* |
|  |  |  |
|  |  |  |
|  |  |  |

***Note to the Author***

[This document is a template of a **Test Report** document for a project. The template includes instructions to the author, example text, and fields that should be replaced with the values specific to the project.

* Blue italicized text enclosed in square brackets ([text]) provides instructions to the document author, or describes the intent, assumptions and context for content included in this document.
* Blue italicized text enclosed in angle brackets (<text>) indicates a field that should be replaced with information specific to a particular project.
* Text and tables in black are provided as examples of wording and formats that may be used or modified as appropriate to a specific project. These are offered only as suggestions to assist in developing project documents; they are not mandatory formats.

When using this template for your project document, it is recommended that you follow these steps:

1. Replace all text enclosed in angle brackets (e.g., <Project Name>) with the correct field values. These angle brackets appear in both the body of the document and in headers and footers.
2. Modify example text as appropriate to the specific project.
3. To add any new sections to the document, ensure that the appropriate header and body text styles are maintained. Styles used for the Section Headings are Heading 1, Heading 2 and Heading 3. Style used for boilerplate text is Body Text.
4. To update the Table of Contents, right-click and select “Update field” and choose the option- “Update entire table”
5. Before submission of the first draft of this document, delete this “Notes to the Author” page and all instructions to the author, which appear throughout the document as blue italicized text enclosed in square brackets.]

**Table of Contents**

[1.0 Introduction 5](#_Toc445795901)

[1.1 Purpose 5](#_Toc445795902)

[2.0 Test PLAN 5](#_Toc445795903)

[3.0 Test Assessment 5](#_Toc445795904)

[4.0 Test Results 5](#_Toc445795905)

[4.1 Unit/Module/System Testing 5](#_Toc445795906)

[4.2 System Testing 6](#_Toc445795907)

[4.3 User Acceptance Testing 6](#_Toc445795908)

[4.4 Regression Testing 7](#_Toc445795909)

[4.5 Performance Testing 7](#_Toc445795910)

[4.6 <Type of Test> 8](#_Toc445795911)

[5.0 Variances 8](#_Toc445795912)

[6.0 Test Instances 8](#_Toc445795913)

[6.1 Resolved Test Incidents 8](#_Toc445795914)

[6.2 Unresolved Test Incidents 8](#_Toc445795915)

[7.0 Recommendations 9](#_Toc445795916)

[APPENDIX A: REFERENCES 10](#_Toc445795917)

[APPENDIX B: KEY TERMS 11](#_Toc445795918)

# 1.0 Introduction

## 1.1 Purpose

This Institute Security Management Test Report provides a summary of the results of test performed as outlined within this document.

# 2.0 Test PLAN

The Framework used for testing is GoogleTest Framework developed by google for testing C/C++ projects. This framework is open source and is hosted [**here**](https://github.com/google/googletest).

The tests performed here are in competence with the test provided in document itself. The googletest project repository is cloned into the source root of project under test. An external build file (Makefile) is developed to reduce the redundant work of compiling whole project from scratch, thus it records what are the changes that we've done in code and compiles only the modified part of the project.

The test written include unit testing, As the code is simple we did manual as well as automatic testing on the software. Input for most test cases were manually feed into a file or were written in test\_ism.cc file. The output then was compared to expected results. Also source code was checked for any coding errors.

test\_ism.cc contains all the unit test with proper assertions and test cases.

# 3.0 Test Assessment

Test assessment was thoroughly done. Source code was traversed manually to search for any errors present. Most of the test cases are checked for errors. Furthermore test cases can be tested for expected result to get all failure cases.

# 4.0 Test Results

[Summarize the test results. Include a detailed description of any deviations from the original test plan, design, test case, or expected results. Include any issues or bugs discovered during the test.]

## Unit/Module/System Testing

Unit, module, and system integration testing activities were performed during the development of the system build or release.

## 4.2 System Testing

The table below summarizes the results of system testing:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Closed prior to Production Release? *[This column will be filled by the dev team.]* | Comments |
|  |  |  |  |  |  | <Yes> or <No> |  |

*[If the test case failed, list the corresponding Test Incident ID in the Comments column.]*

## User Acceptance Testing

The table below summarizes the test cases employed for user acceptance testing and the test results obtained for each test case:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Closed prior to Production Release? *[This column will be filled by the dev team.]* | Comments |
|  |  |  |  |  |  | <Yes> or <No> |  |

*[If the test case failed, list the corresponding Test Incident ID in the Comments column.]*

## Regression Testing

The table below summarizes the test cases employed for regression testing and the test results obtained for each test case:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Closed prior to Production Release? *[This column will be filled by the dev team.]* | Comments |
|  |  |  |  |  |  | <Yes> or <No> |  |

*[If the test case failed, list the corresponding Test Incident ID in the Comments column.]*

## Performance Testing

The table below summarizes the test cases employed for performance testing and the test results obtained for each test case:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Closed prior to Production Release? *[This will be filled by the dev team.]* | Comments |
|  |  |  |  |  |  | <Yes> or <No> |  |

*[If the test case failed, list the corresponding Test Incident ID in the Comments column.]*

## <Type of Test>

The table below summarizes the test cases employed for <type of test (e.g., unit/ module/ interface testing)> and the test results obtained for each test case:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Test Case ID | Date Tested | Tester | Pass/Fail | Severity of Defect [Low/ Medium/ High] | Summary of Defect | Closed prior to Production Release? *[This will be filled by the dev team.]* | Comments |
|  |  |  |  |  |  | <Yes> or <No> |  |

*[If the test case failed, list the corresponding Test Incident ID in the Comments column.]*

# 5.0 Variances

[Describe any variances between the testing that was planned and the testing that actually occurred. Also, provide an assessment of the manner in which the test environment may be different from the operational environment and the effect of this difference on the test results.]

# 6.0 Test Instances

[Provide a brief description of the unexpected results, problems, or defects that occurred during the testing.]

## Resolved Test Incidents

*[Identify all resolved test incidents and summarize their resolutions. Reference may be made to Test Incident Reports that describe in detail the unexpected results, problems, or defects reported during testing, along with their documented resolutions, which may be included as an appendix to this document.]*

*[This will be filled by the dev team.]*

## Unresolved Test Incidents

[Identify all unresolved test incidents and provide a plan of action for their resolution. Reference may be made to Test Incident Reports that describe in detail the unexpected results, problems, or defects reported during testing, which may be included as an appendix to this document.]

*[This will be filled by the dev team.]*

# 7.0 Recommendations

[Describe what actions are suggested upon completion of this test. Provide any recommended improvements in the design, operation, or future testing of the business product that resulted from the testing being reported. A discussion of each recommendation and its impact on the business product may be provided. If there are no recommendations to report, then simply state as such.]

* Interaction with program is command line based with menu driven as numeric input, instead a proper GUI can be used to drive the software.
* Program should contain multiple files with function declaration, definition, and driver program in different files. Generally the classes are defined using “virtual” modifier to ease testing. Program must be modular for ease of understanding and modification.
* Three global arrays of size 100 are declared at the start of the source code. Declaring global variables is not a good practice it should be avoided.
* No constructor is coded for the vehicle database this results in garbage values in database.
* Error while entering input are not handled properly. If we input a string instead of an integer the program fails.
* Instead of using arrays vectors can be used for vehicle database so as to let the database accommodate more than 100 data.

APPENDIX A: REFERENCES

[Insert the name, version number, description, and physical location of any documents referenced in this document. Add rows to the table as necessary.]

The following table summarizes the documents referenced in this document.

|  |  |  |
| --- | --- | --- |
| **Document Name** | **Version** | **Description** |
| *<Document Name>* | *<Version Number>* | *[Provide brief description of the document]* |

APPENDIX B: KEY TERMS

*[Insert terms and definitions used in this document. Add rows to the table as necessary.]*

The following table provides definitions for terms relevant to this document.

|  |  |
| --- | --- |
| **Term** | **Definition** |
| *[Insert Term]* | *[Provide definition of the term used in this document.]* |
| *[Insert Term]* | *[Provide definition of the term used in this document.]* |
| *[Insert Term]* | *[Provide definition of the term used in this document.]* |