

**The A Team**

**Image Processing Tool for**

**Leidenfrost-Ratchet Systems**

**Test Plan Document for Version 2.0 (Final Draft)**

May 4, 2015

**Authored By:**

Sanan Aamir

Romando Garcia

Anne Lam

James Rowe

Hieu Tran

Table of Contents

1. Introduction 2
   1. Scope of Test Plan Document 2
   2. Product Overview 2
2. Overview of Document 2

2.1 Test Strategies 2-3

2.2.The Testing Process. 3

2.3.Testing Criteria. 3

2.3.1. Unit Testing. 3-4

2.3.2. Interface Testing. 4

2.3.3. Integration Testing. 4-5

2.3.4. System Testing 5

1. Test Schedule 5
2. Resources 5
   1. Personnel 5
   2. Hardware 6
   3. Software 6
3. Test Cases Appendix A, 5
4. Glossary 6
5. References 6

Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version Number** | **Date** | **Contributor** | **Description** |
| V1.0 | 4/12/2015 | Author: Hieu Tran  Reviewers: Rest of  A-Team | First draft of test plan |
| V2.0 | 4/28/2015 | Authors: Anne Lam and  Hieu Tran | Make revisions and add test  cases |

**1. Introduction**

This document details the test plan for the testing phase of the software development life cycle. This document sets the scope of various tests to be conducted, the activities to be completed, the general resources required, and the process to be used to test the final product before release.

1.1 Scope of Test Plan Document

The scope consists of what we intend to test and nothing more. For our purposes and our time constraints, we will concentrate on unit, interface, system testing approaches.

1.2 Product Overview

The current purpose of the software is to efficiently track a drop of liquid and record measurements through images gathered from a high speed camera as the drop falls from an injection needle and travels along a ratchet surface. In this version, Version 2.0, we set out to also obtain net measurements as well as droplet volume, improve the user interface, and the processing time.

**2. Overview of Document**

In total, this document specifies the strategies and plans for testing the Image Processing Tool Version 2.0.

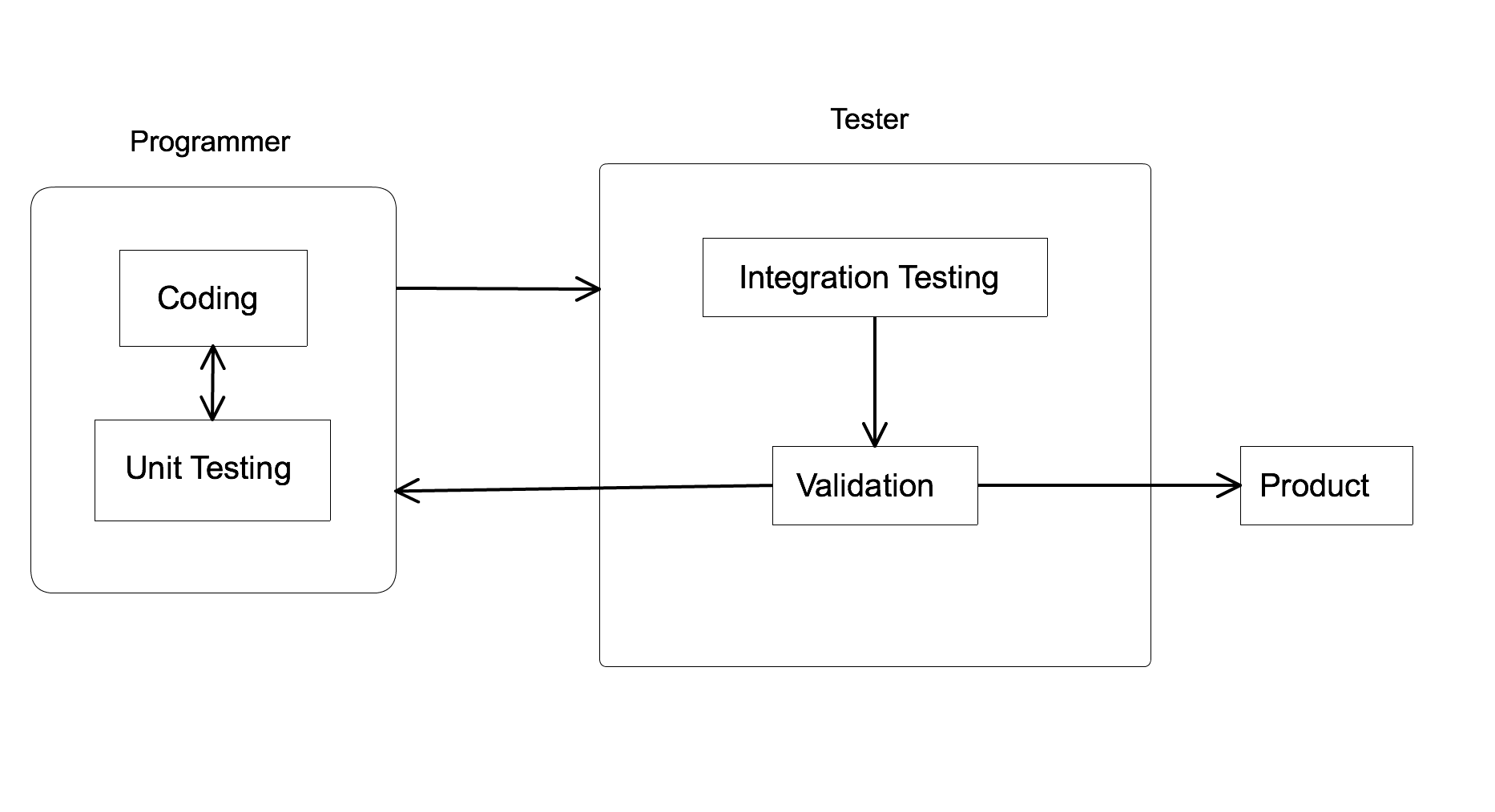
2.1 Test Strategies

The following testing strategies will be employed in order to efficiently track issues and resolve them before release.

* **Unit Testing** – This uses the "white box" method of testing to find any potential faults in the code of each class. Since processing images brings a level of complexity, unit tests will only be written to test calculation/algorithm logic. Only the programmers perform unit testing. Anne Lam, Sanan Aamir, Romando Garcia, and James Rowe will be responsible for unit testing on their individual code contributions.
* **Interface Testing** – This uses the "black box" method of testing to review functionality of the user interface. Test cases are required to check every possible scenario and outcome resulting from interactions with the user interface. Hieu Tran will be responsible for interface testing.
* **System Testing –** This also uses the "black box" method of testing to check for each input the user may enter and ensure the expected results occur. Hieu Tran will primarily be responsible for system testing.

2.2 The Testing Process

The flow chart in Figure 1 shows the process of the our test plan.



Ini Interface testing

Ini Integration & System testing

Figure 1. Test Plan Flow Chart

2.3 Testing Criteria

This section discusses the testing criteria for each testing strategy (unit, interface, and system) we plan to use during the testing phase of the software development cycle.

2.3.1 Unit Testing

“At agreed upon regularly scheduled intervals, the system’s developmental progress will be subject to peer review. If necessary, the testing schedule will correspond to the completion of a system module. The objective of these reviews is to ensure correctness and to test the functional integrity within each individual module. Issues to consider are matching of parameters, arguments, relative attributes, I/O interface and memory management. Symbolic execution will be tested utilizing ‘basic path testing’ also known as ‘white box’ testing.” (Sommerville, pg 448)

**Entrance Criteria** - At least one module should be coded and tested to schedule a formal review meeting. As the group meets every Monday and Wednesday in a week, the development team will try to code at least one module by every Monday and Wednesday so that it can be formally reviewed in the next group meeting.

**Exit Criteria** – All modules should pass their unit tests created by the programmer. No more than 2 defects are found within the modules after review and revisions.

2.3.2 Interface Testing

Interface Testing is performed to evaluate whether the user interface of the Image Processing Tool passes data correctly and behaves as expected by the user. It is to verify if all the interactions between functional modules are working properly and errors are handled properly.

**Entrance Criteria** –Interface control functions and event handlers must all be established.

**Exit Criteria** – Interface testing exits when the tester verifies all interface test cases have passed and produced the correct expected output.

2.3.3 Integration Testing

“This test proves that all areas of the system interface with each other correctly and that there are no gaps in the data flow. The final integration test proves that system works as an integrated unit when all the fixes are complete. The actual testing method used for this phase will be the ‘black box’ method, (Sommerville, pg 443). ‘Bottom-up testing strategy’ will be followed throughout the integration-testing phase” (Sommerville, pg 453).

**Entrance Criteria** – At least 2 high priority modules must have passed their unit tests and been thoroughly reviewed.

**Exit Criteria –** All high priority errors from integration tests are fixed and tested.

2.3.4 System Testing

System testing is a black box testing technique performed to verify all expected inputs result in the system providing the correct output. The functionality of the Image Processing Tool will be tested from the user perspective.

**Entrance Criteria –** Entry criteria will start when the interface test phase has been completed.

**Exit Criteria –** When the tester has verifies all the system test cases has pass and produced the correct expected output.

**3. Test Schedule**

For the testing phase, the following schedule will apply, with room for flexibility as needed:

Unit testing: 3 days

Interface testing: 2 days

Integration testing: 3 days

System testing: 2 days

**4. Resources**

This section list the resources required to execute the test plan including the team members and the hardware and software requirements.

4.1 Personnel

The A-Team consists of the following team members:

Sanan Aamir – Tester/Programmer

Romando Garcia – Tester/Programmer

Anne Lam – Tester/Programmer

James Rowe – Tester/Programmer

Hieu Tran - Tester

4.2 Hardware

The testing phase will require at least one PC with the following specifications:

32 - bit architecture

at least 1GB of RAM

at least 1GHz of Processor Speed

4.3 Software

The testing phase will also require the following software:

Microsoft Visual Studio 10 or higher

Microsoft .NET 2.0 or higher

Windows 7 or higher

**5. Test Cases**

The test cases for the interface and system testing are listed in a table found in Appendix A.

**6. Glossary**

|  |  |
| --- | --- |
| **Term** | **Definition** |
| **Leidenfrost-Ratchet System** | A system involving a ratchet surface heated to a fluid's Leidenfrost point will allow a droplet of that fluid to spontaneous accelerate along that surface, even if it means traveling up slope. |
| **Droplet** | A very small drop of a liquid. |
| **Black Box** | A method of software testing that examines the functionality of an application without peering into its internal structures or workings. |
| **White Box** | A method of testing software that tests internal structures or workings of an application |
| **Unit Test** | A software testing method by units of source code, sets of one or more computer program modules together with associated control data, usage procedures, and operating procedures, are tested to determine whether they are fit for use. |

**7. References**

This document was completed with guidance from the Mavis and Scors test plan document, Test Plan Power Point slides written by Catherine Stringfellow, Professor at Midwestern State University, and a Software Engineering book written by Sommerville.

**Appendix A - Interface and System Test Cases**

**Interface Test Cases**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Cases | Step & Input | Expected Results | What is Tested | Actual Results | Pass or Fail |
| **Load Button** |  |  |  |  |  |
| LB 011  Open File Dialog | Click Load Button > Show Open File Dialog | Open File Dialog Should Appear | User Interface |  |  |
| LB 021 Accept Directory .TIF | Find Image Directory > Open .TIF image folder | Accept .TIF Directory | Open File Dialog |  |  |
| LB 022  Accept Directory .BMP | Find Image Directory > Open .BMP image folder | Accept .BMP Directory | Open File Dialog |  |  |
| LB 023  Reject Single File | Find Image Directory > Open single file | Will not show single file | Open File Dialog |  |  |
| LB 024  Reject Other Directory Extension | Find Image Directory > Open other folder directory | Will not allow user to select other extension | Open File Dialog |  |  |
| LB 031 Show Image | Find Directory > Click Open | Will show Image on UI | User Interface / Open File Dialog |  |  |
| LB 032  Enable Run Button | Find Directory > Click Open | Run Button Will Enable | User Interface |  |  |
| LB 033  Enable Calibrate Button | Find Directory > Click Open | Calibrate Button Will Enable | User Interface |  |  |
| **Initial User Interface** |  |  |  |  |  |
| UI 011 Disable Calibrate Button | Click Image Processing Application | Calibrate Buttons disable | User Interface |  |  |
| UI 012 Disable Run Button | Click Image Processing Application | Run Button is Disable | User Interface |  |  |
| UI 013  Disable Black/White Calibration Drop down | Click Image Processing Application | Black/White Calibration drop down is disable | User Interface |  |  |
| UI 014  Enable Load Button | Click Image Processing Application | Enable Load Button | User Interface |  |  |
| UI 015  Enable Frame Rate Input | Click Image Processing Application | Enable Frame Rate Input | User Interface |  |  |
| UI 016  Enable Base/Needle Height | Click Image Processing Application | Enable Base/Needle Height | User Interface |  |  |
| UI 021  Enable File Menu | Click Image Processing Application | Enable File Menu | User Interface |  |  |
| UI 022  Enable Image Menu | Click Image Processing Application | Enable Image Menu | User Interface |  |  |
| UI 023 Enable About Menu | Click Image Processing Application | Enable About Menu | User Interface |  |  |
| **Frame Rate** |  |  |  |  |  |
| FR 011  Enter Frame Rate | Click Up or Down to select frame rate | Let User Choose Correct Frame Rate | User Interface |  |  |
| FR 011  Enter Frame Rate | Click Down to select negative number | Should stop at zero | User Interface |  |  |
| **Base/Needle Height** |  |  |  |  |  |
| BN 011 Enter Base/Needle Height | Enter negative number Base/Needle Height | Should not let user enter negative number | User Interface |  |  |
| BN 012 Enter Base/Needle Height | Enter letter for Base/Needle Height | Should not let user enter alphabetic | User Interface |  |  |
| BN 013 Enter Base/Needle Height | Enter positive Number for Base/Needle Height | Should let user enter positive number | User Interface |  |  |
| BN 014 Enter Base/Needle Height | Enter decimal number for Base/Needle Height | Should let user enter decimal number | User Interface |  |  |
| **Black/White Calibration** |  |  |  |  |  |
| BW 011 Black/White Calibration | Click Up or Down to select Black/White Calibration | Should let user click up or down | User Interface |  |  |
| BW 012 Black/White Calibration | Click Down to select negative number for Black/White Calibration | Should stop at zero | User Interface |  |  |
| **Run Button** |  |  |  |  |  |
| RB 011 Run Button | Click Run Button | Generate excel | User Interface |  |  |
| RB 012 Run Button | Click Run Button > Save File Dialog > Choose Location > Enter Name | Let user choose location and enter a name | User Interface / Save File Dialog |  |  |
| **File Menu Strip** |  |  |  |  |  |
| FM 011  Open File Dialog | Click Load Button > Show Open File Dialog | Open File Dialog Should Appear | User Interface |  |  |
| FM 021 Accept Directory | Find Image Directory > Open .TIF image folder | Accept .TIF Directory | Open File Dialog |  |  |
| FM 022  Accept Directory | Find Image Directory > Open .BMP image folder | Accept .BMP Directory | Open File Dialog |  |  |
| FM 023  Accept Directory | Find Image Directory > Open single file | Will not show single file | Open File Dialog |  |  |
| FM 024  Accept Directory | Find Image Directory > Open other folder directory | Will not allow user to select other extension | Open File Dialog |  |  |
| FM 031 Show Image | Find Directory > Click Open | Will show Image on UI | User Interface / Open File Dialog |  |  |
| LB 041  Enable Run Button | Find Directory > Click Open | Run Button Will Enable | User Interface |  |  |
| LB 051  Enable Calibrate Button | Find Directory > Click Open | Calibrate Button Will Enable | User Interface |  |  |
| **About Menu Strip** |  |  |  |  |  |
| AB 011 Show Information Form | Click About on the Menu Strip | Open an information regarding about the application | User Interface |  |  |

**System Test Cases**