

Test Report: Glass Breakage analysis Program (GlassBR)

Vajiheh Motamer

December 27, 2018

1 Revision History

Date	Version	Notes
12/20/18	1.0	Initial draft based on initial SystemVnV-Plan
12/25/18	1.1	Make changes based on SystemVnVPlan changes

2 Symbols, Abbreviations and Acronyms

The symbols, abbreviations, and acronyms used in this document include those defined in the table below, as well as any defined in the tables found in Section 2.3 of the Software Requirements Specification (SRS) document

symbol	description
MIS	Module Interface Specification
MG	Module Guide
TC	Test Case
VnV	Verification and Validation

Contents

1	Revision History	i
2	Symbols, Abbreviations and Acronyms	ii
3	Functional Requirements Evaluation	1
4	Nonfunctional Requirements Evaluation	2
4.1	Reusability	2
4.2	Understandability	2
4.3	Correctness	2
5	Comparison to Existing Implementation	2
6	Unit Testing	2
7	Changes Due to Testing	2
8	Automated Testing	3
9	Trace to Requirements	3

List of Tables

1	Traceability Matrix Showing the Connections Between Requirements and Test Cases	3
---	---	---

List of Figures

This document outlines the results of testing for GlassBR. Section 3 reports on the Test Cases (TCs) for functional requirements and Section 4 reports on the tests for non-functional requirements, all of which are described in the System Verification and Validation (VnV) Plan for this project. Section ?? compares this implementation of GlassBR to the original implementation. Section 6 reports on the results of the unit tests, which are described in the Unit VnV Plan for this project. Section 7 comments on changes to the project that came as a result of the testing. Section ?? describes how the tests were implemented with an automated testing framework. Sections 9 and ?? show the traceability between test cases and requirements and modules. Supporting documents and other resources, such as the VnVPlans and original implementation, can be found on [the GitHub repository for this project](#).

3 Functional Requirements Evaluation

The System VnV Plan described TC1 - TC7 for testing the requirement for verifying different forms of valid user inputs. Besides, these test cases verified GlassBR for the special values that glass is safe or not. All of these tests passed.

The System VnV Plan described TC8 - TC24 for testing the requirement for verifying that inputs meet physical constraints. The test cases TC21 and TC22 and TC23 and TC24 passed. However, TC25 and TC26 failed at the first time, then they passed. I think it would be better equation should be removed from SRS “W” and “SD” bands.

The final functional requirement tests in the System VnV Plan, TC27 - TC29, were for verifying the requirements for verifying and delivering output. These tests all passed.

The System V&V Plan can be found at: <https://github.com/smiths/caseStudies/tree/master/CaseStudies/glass/docs/VnVPlan/SystVnVPlan/SystVnVPlan.pdf>.

4 Nonfunctional Requirements Evaluation

4.1 Reusability

TC³¹ in the System VnV Plan is a test for gaining confidence in the reusability of GlassBR. The alternative and new implementation(units has been considered as meter) was written based off of the old implementation which the units have been considered as “mm”, which accepted by GlassBR. The tests passed, showing that all of the other modules were reusable in this alternative version of GlassBR.

4.2 Understandability

Since reusability are related to understandability, the success of the tests for those requirements gives confidence in the understandability of GlassBR.

4.3 Correctness

The success of the system tests for the calculations of GlassBR, specifically TC¹ - TC²⁶, gives confidence in the correctness of GlassBR.

5 Comparison to Existing Implementation

This section is not applicable for GlassBR.

6 Unit Testing

The Unit VnV Plan described TC¹ - TC⁴², all of which were implemented in the unit testing framework and passed, with the exception of TC³², which failed on the first run,

7 Changes Due to Testing

The test cases for missing inputs in the System V&V Plan initially failed because there was a bug in the condition that checked for missing inputs

and was producing an incorrect output. Upon modifying the condition, the correct output was produced and the tests have passed.

The failure of TC32 revealed that the implementation had a bug in the implementation. This prompted a change to the FuncADT module to properly handle this case.

8 Automated Testing

Every test, with the exception of the tests for non-functional requirements, was automated so that the tests could be executed with a simple command.

9 Trace to Requirements

Tables ?? and ?? were taken directly from the System VnV Plan document for this project. It shows that the tests effectively cover every Requirement (R) and Non-Functional Requirement (NFR) from the SRS. Unless otherwise specified, the test cases referenced in the table refer to tests described in the System VnV Plan.

	R1	R2	R3	R4	R5	R6
TC1 - TC7	X					
TC1 - TC7		X				
TC8 - TC24			X			
TC1 - TC7				X		
TC1 - TC7					X	
TC27 - TC29						X

Table 1: Traceability Matrix Showing the Connections Between Requirements and Test Cases