CMSC 447 ⧫ Software Engineering 1 ⧫ Professor Druffel



Test Report

# Crisis Response Ticket System for Shawn Davis by Team Cloud

**Team Members:** David Foster, Jack McGrann, Julie Nau, Patrick Wheeler

**Last updated:** November 28th, 2020

*Crisis Response Ticket System*

Testing Report

**Table of Contents**

[**Crisis Response Ticket System for Shawn Davis by Team Cloud**](#_heading=h.x1mm5bfjk8x3) **1**

[**Introduction**](#_heading=h.gjdgxs) **3**

[Purpose of This Document](#_heading=h.30j0zll) 3

[References](#_heading=h.1fob9te) 3

[**Testing Process**](#_heading=h.3znysh7) **4**

[Description](#_heading=h.2et92p0) 4

[Impressions of the Process](#_heading=h.tyjcwt) 4

[Testing Sessions](#_heading=h.3dy6vkm) 5

[**Test Results**](#_heading=h.1t3h5sf) **5**

[Test Cases](#_heading=h.4d34og8) 6

[Test Results - Analysis & Metrics](#_heading=h.2s8eyo1) 7

[*Successful Test Cases (Pass/Fail)*](#_heading=h.17dp8vu) *7*

[*Unable to Execute (Blocked) Cases*](#_heading=h.3rdcrjn) *7*

[*Defects - Failed Test Cases (Resolved and Open)*](#_heading=h.26in1rg) *8*

[Defects Summary](#_heading=h.msow4l24s3gq) 8

[Defect Details](#_heading=h.5blgvkbmk5nz) 10

[**Appendix A – Agreement Between Customer and Contractor**](#_heading=h.lnxbz9) **11**

[**Appendix B – Team Review Sign-off**](#_heading=h.35nkun2) **11**

[**Appendix C – Document Contributions**](#_heading=h.1ksv4uv) **12**

[**Appendix D – Test Case Details (pdf)**](#_heading=h.44sinio) **12**

**Document Versioning Control**

|  |  |  |  |
| --- | --- | --- | --- |
| Version Number | Date | Changes from Previous Version | Author |
| 0.2 | 11/25 | Original prototype document | Team Cloud |
| 0.5 | 11/27 | Added some content | Team Cloud |
| 0.7 | 11/28 | Refined content | Team Cloud |
| 1.0 | 11/29 | Finalized document | Team Cloud |

# Introduction

## Purpose of This Document

This document is a Test Report written to facilitate customer review of the finished software. By examining the cases explored and examined in this document, the reader will be able to accurately determine the capabilities and deficiencies of the product. First the testing process is explained, followed by the results of that process, and concluding with analysis of the results.

## References

Druffel, K. (2020, October 27). 447-10-27-peer\_review\_process. Retrieved November 4, 2020, from <https://umbc-my.sharepoint.com/:w:/g/personal/kdruffel_umbc_edu/EfEEJH3ee_pMsYq7Q8TYcscBig-nRx6Z6WJdg4aKHrFQEg?e=QmOBDA>

Druffel, K. (2020, October 27). 447-10-27-Implementation-Phase. Retrieved November 4, 2020, from <https://umbc-my.sharepoint.com/:p:/g/personal/kdruffel_umbc_edu/ESF8sTEz_LNOnoUoq_jGpQoBWEZ4qag3vVX6t9rU9GZBew?e=YyUmBL>

Druffel, K. (2020, October 29). 447-10-29-VV-Peer-Review. Retrieved November 4, 2020, from <https://umbc-my.sharepoint.com/:p:/g/personal/kdruffel_umbc_edu/EQyq5wmOoFpDs_YInbms0VEBus6NuOuyysZMpWHfC54uoQ?e=330IQD>

Druffel, K. (2020, September). 2-ProjectDescription-CalltoMission. Retrieved November 4, 2020, from <https://umbc-my.sharepoint.com/:w:/g/personal/kdruffel_umbc_edu/EYVFEiSwyVJHv5VKzjr2OJUBkPo4klqrugJTDvI-cusDCA?e=SbTP27>

Foster, D. McGrann, J. Nau, J. Wheeler, P. (2020, October 4). System Requirement Specification. Retrieved November 4, 2020 from <https://docs.google.com/document/d/1AmYUFUAu0lfXBrhUMcs8OTf3v3z33YLIvxKWJdgAxdk/edit>

Foster, D. McGrann, J. Nau, J. Wheeler, P. (2020, October 25). System Design Document. Retrieved November 4, 2020 from <https://docs.google.com/document/d/1Q4s01lPTVFuPf3dLJaQZkk2kOyCFJNJCad9Ifspoz4A/edit>

Foster, D. McGrann, J. Nau, J. Wheeler, P. (2020, October 25). User Interface Design Document. Retrieved November 4, 2020 from <https://docs.google.com/document/d/1soKtDNpGoPIUKSBBHk-8vfZ29GVELDdiWIaKQDhTPog/edit>

Mae, F. (1996, May 29). The Hunter Group, Inc. PeopleSoft Implementation Development Standards. Retrieved November 4, 2020, from <https://umbc-my.sharepoint.com/personal/kdruffel_umbc_edu/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fkdruffel%5Fumbc%5Fedu%2FDocuments%2FUMBC%2DCMSC447%2F2020b%2DFall%2DKDruffel%2F20%2DBBDocuments%2F3%2DDocument%2DTemplates%2FFNMA%5FDEVSTNDS%2D1996%2Epdf&parent=%2Fpersonal%2Fkdruffel%5Fumbc%5Fedu%2FDocuments%2FUMBC%2DCMSC447%2F2020b%2DFall%2DKDruffel%2F20%2DBBDocuments%2F3%2DDocument%2DTemplates&originalPath=aHR0cHM6Ly91bWJjLW15LnNoYXJlcG9pbnQuY29tLzpiOi9nL3BlcnNvbmFsL2tkcnVmZmVsX3VtYmNfZWR1L0VhZ2E2UmlkcllSSXBvanZoMkNlalNrQlNrbUlOZk9KQmZobk1McHBMZmpIS0E_cnRpbWU9c0hjTUx2Q0kyRWc>

# Testing Process

Our testing process involved a series of steps. First we established the test cases and a baseline. The baseline used a version of our application we knew was stable and wouldn’t be vulnerable to external factors like a faulty connection to the server or old drivers.

Once we established the various cases on the stable machine, we methodically ran through each case in batches, where each batch focused on a specific element such as the login process, ticket updating, or map use. We compiled the results of these tests which you will now see in the attached spreadsheet.

## Description

First, the team worked together to discuss and develop as many test conditions as necessary to properly test the implementation of the system. Then, once each component was fully implemented and ready for testing, the team was notified and the test cases were executed with the corresponding inputs as defined in the attached excel spreadsheet.

After this, components that still did not satisfy the testing conditions were addressed and fixed to the best of each team members ability. The system was reset to the state it was in before the test cases were executed, and the test cases were re-run, and the corresponding results were recorded, and readdressed if needed.

## Impressions of the Process

For a lot of the components, the testing process was a valuable way to identify errors in checking the validity of user inputs. However, for many of the components of the system, the test cases were conducted iteratively throughout the implementation phase, and new test cases had to be created for components that were already fully functioning. It also felt like a very repetitive process from the code inspection process since we had to test very similar functionality and implementation using similar test cases.

The process of having to document these test cases also takes time away from team members to make much needed improvements on their implementations. Also having this documentation with this much required detail takes time away from developing our final presentation which is due just a few days after this is due.

The best unit was the SQL setup and update operations. The simple and modular nature of this unit makes it the least prone to flaws, as it is very easy to test and verify the integrity of results through using MySQL workbench, which shows the live results of the SQL database.

The worst unit was the HTML views for the users and the business logic connecting them to the SQL server. This is because the flaws can be difficult to pin down as they involve a lot of messy code which can be time-consuming to comb through.

## Testing Sessions

In this section, we will give a log of when, where, and how each testing session took place. We will present, in tabular format, the date, method of communication used, time started and ended, participants, and which use case was covered using the Test# from the attached excel spreadsheet.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Date | Method of Communication | Time Span | Participants | Test Case |
| 11/15 | Discord | 3-4 | David F  Jack M | WD-1  WD-2 |
| 11/18 | Google Meet | 3-4 | David F  Jack M | L-1, L-2, L-3 + L-4  M-1 + M-2 |
| 11/29 | Discord | 3-6 | David F  Jack M  Patrick W  Julie N | CCO-1 + CCO-2  OC-1  MM-1  NFR-1:3 |

# Test Results

The purpose of this section is to document the results of all the Testing Sessions laid out in section 2.3. It goes into detail as to how the test cases were constructed and documented, and why these methods were chosen by the team. Then it goes over the results of every one of these testing sessions, and then statistical analysis is performed on the resulting pass/fail data. It also goes into detail as to which components are blocked and which components still have defects and whether or not the components with remaining defects have been resolved or not. Statistical analysis is then performed on the components that still have defects based on the severity of those defects, and a summary of the defects are given for any that require further detail.

## Test Cases

Test cases are defined for each module, user view, and non-functional requirement as they are described in the SDD, UIDD, and SRS respectively. The testing information is organized in a shared spreadsheet. This was convenient for a tester to independently record the outcome of a given test, and provide updates if necessary. Additionally, spreadsheet platforms provide powerful tools for analyzing datasets, their charting and graphing capabilities were of great use.

Using the fields in the table provided below, we have defined tests to assess the functionality of each component under a variety of circumstances a potential user may encounter. The majority of test cases are used to assess the application’s data validation, error handling, database and server communication, and page routing. Specific details of the tests are provided in Appendix D.

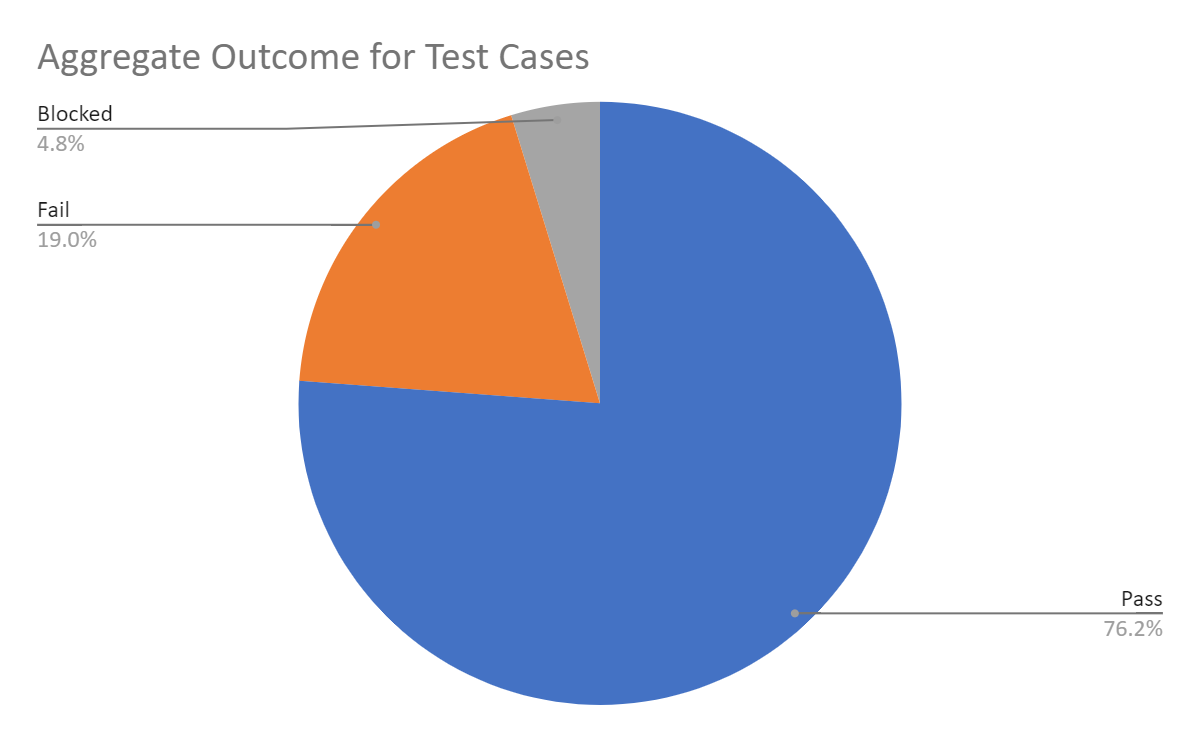
|  |  |
| --- | --- |
| Item | Description |
| Component | The part of the system being tested |
| Test # (ID) | A unique test identification number |
| Test Case Title | A short, descriptive title for the test case |
| Test Steps | An ordered list of steps that must be taken to execute a given test case |
| Test Data/Input Values | The values to be input |
| Expected Result | The result expected for passing the test |
| Actual Result | The actual results from the test |
| Status (Pass, Fail, or Blocked) | Status of whether the test passed or failed to get the expected results |
| Severity | How critical is a failed test to the project? Examples: Critical, Major, and Minor |
| Tester & Date Tested | Testing information (who and when) |
| Comments | Notes from the tester (optional) |

## Test Results - Analysis & Metrics

### Successful Test Cases (Pass/Fail)

|  |  |  |  |
| --- | --- | --- | --- |
| **Test Cases Planned** | **Test Cases Executed** | **Test Cases Passed** | **Test Cases Failed** |
| 23 | 22 | 18 | 4 |

76.2% of planned test cases were passed, while 19.0% were failed; one case could not be executed due to restrictions with external services, meaning that of all executed test cases, 81.8% were passed and 18.2% were failed. In all successful test cases, the actual result of the test was identical to the expected result.



### Unable to Execute (Blocked) Cases

Although the vast majority of our tests were able to be successfully executed, we encountered a blocked test case with regard to the non-functional requirement of server availability. Our application is currently dependent on Amazon Web Services for database functionality, and therefore the availability of our application is dependent on the availability of AWS, over which this development team has no control. Because of this, we were unable to confirm or deny that this requirement will be consistently met.

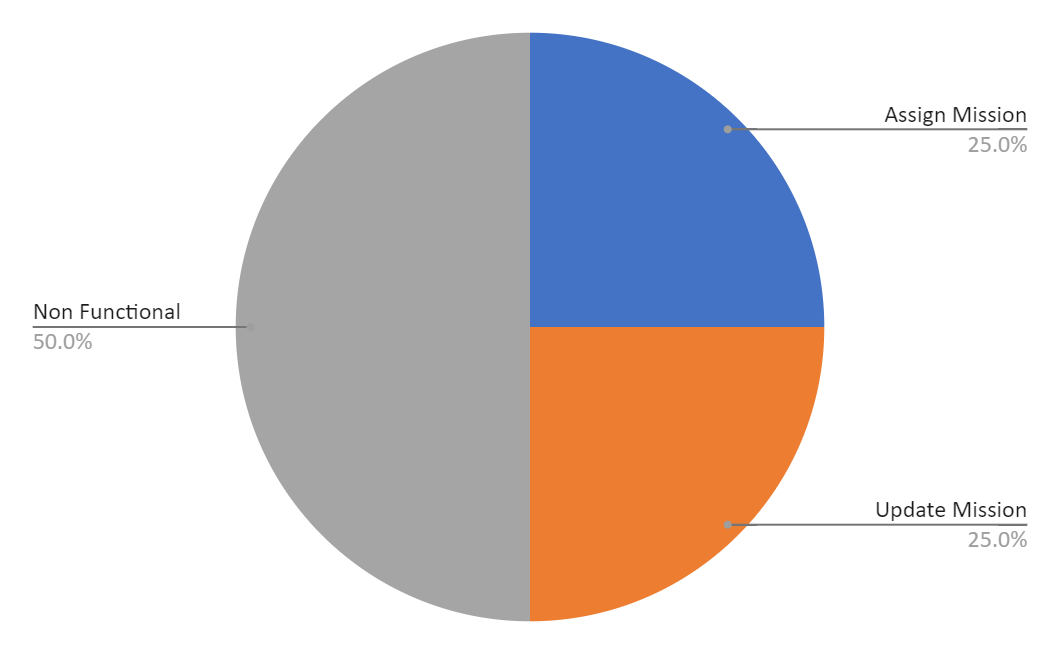
### Defects - Failed Test Cases (Resolved and Open)

4 out of 22 executed test cases were failed at the time of testing.

#### Defects Summary

Overall, four defects were identified at this stage of the application’s implementation. Two of these defects, one in the context of properly assigning missions and one in properly updating them, are due to a lack of proper implementation of methods dealing with database manipulation. The remaining two defects are related to certain non-functional requirements which deal with account security.

None of these defects have been rectified at the time of writing this document.



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Assign Mission** | **Update Mission** | **Non Functional** | **Total** |
| **Critical** | 1 | 0 | 0 | 1 |
| **Major** | 0 | 1 | 0 | 1 |
| **Minor** | 0 | 0 | 2 | 2 |
| **Total** | 1 | 1 | 2 | 4 |

#### Defect Details

The first defect, which is associated with assigning tickets to new missions, is dependent on user interface- and database-related methods which have not yet been effectively implemented. The second defect, which is similarly associated with updating the status of existing missions, exists for comparable reasons. These will be resolved upon implementation of the necessary methods.

The last two defects stem from non-functional requirements which have yet to be met, in one case regarding the encapsulation of different user views, preventing one type of user from accessing another type of user’s view, and in the other case regarding general account security, specifically the repeated input of incorrect passwords. As these security measures are not integral to the practical functionality of the application, they are not a priority at this time, but these defects will be repaired with additional scrutiny of the code associated with them and the implementation of further security measures if necessary.

# Appendix A – Agreement Between Customer and Contractor

The acceptance criteria for passing a test means that the test result fulfills the requirement put in place for that element in advance, fulfilling the base functionality to the user satisfaction. Above are listed all the results of the tests that passed and those that didn’t, and all remaining and known defects. By signing this document, each party agrees that the current state of the tested product is ready and acceptable to “go live.”

|  |  |  |
| --- | --- | --- |
| **Typed Name** | **Electronic Signature** | **Date Signed** |
| David F | David Foster | 11/29/2020 |
| Jack M | John McGrann | 11/29/2020 |
| Julie N | Julia Nau | 11/29/2020 |
| Patrick W | Patrick Wheeler | 11/29/2020 |
| Shawn D |  |  |

Space for customer comments:

# Appendix B – Team Review Sign-off

By signing below, all team members have reviewed the document and agree on its content and format. The comment section is to be used for minor unresolved issues.

|  |  |  |  |
| --- | --- | --- | --- |
| **Typed Name** | **Electronic Signature** | **Date Signed** | **Comments** |
| David F | David Foster | 11/29/2020 |  |
| Jack M | John McGrann | 11/29/2020 |  |
| Julie N | Julia Nau | 11/29/2020 |  |
| Patrick W | Patrick Wheeler | 11/29/2020 |  |

# Appendix C – Document Contributions

|  |  |  |
| --- | --- | --- |
| **Team Member** | **Contribution** | **Estimated Percent Work** |
| David F | Into/App, Part 2 | 25% |
| Jack M | All except the non-functional requirements on excel spreadsheet, Section 2, Intro for Section 3 | 25% |
| Julie N | Section 3.2 | 25% |
| Patrick W | Section 3.1 | 25% |

# Appendix D – Test Case Details (pdf)