**Test Report**

|  |  |
| --- | --- |
| UMGC Fall 2020 | Benjamin Fetterman, Benjamin Murray, Hanim Danur, James Cornelius, Robert Lee  SWEN 670 |

Project Plan Approvals

|  |  |  |
| --- | --- | --- |
| Name | Signature | Date |
| Approved by:  Dr. Mir Assadullah |  |  |
| Approved by:  "Stakeholder" |  |  |
| Approved by:  "Project Manager" | Robert Lee | 11/3/2020 |

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| Revision | Date | Name | Description of Change(s) |
| 1.0 | 11/03/2020 | Robert Lee | Initial Release |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[1.0 Introduction 5](#_Toc55247379)

[1.1 Purpose 5](#_Toc55247380)

[1.2 Intended Audience 5](#_Toc55247381)

[1.3 Technical Project Stakeholders 5](#_Toc55247382)

[2.0 Overview of Test Results 5](#_Toc55247383)

[3.0 Test Summary 6](#_Toc55247384)

[3.1 Detailed Test Results 6](#_Toc55247385)

[4.0 Subsystem Test 7](#_Toc55247386)

[4.1 Chatbot Greeting 7](#_Toc55247387)

[4.2 Chatbot Zoning/Permit Links 8](#_Toc55247388)

[4.3 Chatbot Application Links 9](#_Toc55247389)

[4.4 Chatbot Retrieve Regs 10](#_Toc55247390)

[4.5 Chatbot Retrieve Standards 11](#_Toc55247391)

[4.6 Chatbot Retrieve Permits Applicable to Zoning Statutes 13](#_Toc55247392)

[4.7 Chatbot GIS Map Layer Integration 15](#_Toc55247393)

[4.8 Chatbot Address Requirements 17](#_Toc55247394)

[4.9 Address Validation 19](#_Toc55247395)

[4.10 Chatbot Other Information 20](#_Toc55247396)

[4.11 Chatbot Help 21](#_Toc55247397)

[4.12 Retrieve Permit Unit Test 22](#_Toc55247398)

[5.0 Test Execution Findings 23](#_Toc55247399)

[5.1 Metrics 23](#_Toc55247400)

[5.2 SonarQube Overview 23](#_Toc55247401)

[5.3 Selenium Overview 24](#_Toc55247402)

[5.4 JUnit Overview 24](#_Toc55247404)

[6.0 Exit Criteria 25](#_Toc55247406)

[Appendix A: Acronyms and Abbreviations 26](#_Toc55247407)

[Appendix B: References 27](#_Toc55247408)

List of Tables

[Table 1 Stakeholders 5](#_Toc55247363)

[Table 2 Testing Responsibilities 6](#_Toc55247364)

[Table 3 Schedule for Test Tasks 6](#_Toc55247365)

[Table 4 Test Report 7](#_Toc55247366)

[Table 5 Test-1.1 Chatbot Greeting 7](#_Toc55247367)

[Table 6 Test-2.1 Chatbot Zoning/Permit Links 8](#_Toc55247368)

[Table 7 Test-2.2 Chatbot Application Links 9](#_Toc55247369)

[Table 8 Test-2.3 Chatbot Retrieve Regs 10](#_Toc55247370)

[Table 9 Test-2.4 Chatbot Retrieve Standards 11](#_Toc55247371)

[Table 10 Test-3.1 Chatbot Retrieve Permits Applicable to Zoning Statues 13](#_Toc55247372)

[Table 11 Test-3.2 Chatbot GIS Map Layer Integration 15](#_Toc55247373)

[Table 12 Test-4.1 Chatbot Address Requirements 17](#_Toc55247374)

[Table 13 Test 4.2 Address Validation 19](#_Toc55247375)

[Table 14 Test-5.2 Chatbot Help 21](#_Toc55247376)

[Table 15 Overall Results of Selenium Tests 24](#_Toc55247377)

[Table 16 Overall Results of JUnit Unit Tests 25](#_Toc55247378)

List of Figures

[Figure 1 SonarQube Report 23](#_Toc55238569)

1. Introduction

This document contains information pertaining to the testing activities of the UMGC chatbot application. The document contains the overview of test results, test summary, subsystem tests, and test execution findings.

Purpose

The purpose of this document is to explain and detail the variety of tests and their outcomes that were undertaken to ensure the viability of the UMGC chatbot application. This document details the results and findings of the tests.

Intended Audience

The intended audience for this guide are the developers and other project stakeholders who undertake supporting the UMGC chatbot application. This document will make them aware of tests that must be passed in order for the UMGC chatbot to function properly. If any change are made in the future, these tests should be run in order to ensure that such changes do not break the functionality of the UMGC City Chatbot.

Technical Project Stakeholders

Table 1 Stakeholders

|  |  |  |
| --- | --- | --- |
| Name | E-mail address | Role |
| Professor Assadullah | mir.assadullah@faculty.umgc.edu | Stakeholder |
| Robert Lee | rlee97@student.umgc.edu | Project Manager |
| Benjamin Fetterman | bfetterman2@student.umgc.edu | Developer |
| Benjamin Murray | bmurray19@student.umgc.edu | Developer |
| Hanim Danur | hdanur@student.umgc.edu | Developer |
| James Cornelius | jcornelius10@student.umgc.edu | Developer |
| Glenn Goodlett | ggoodlett1@student.umgc.edu | DevOps |
| Dustin Emerson | demerson2@student.umgc.edu | DevOps |

1. Overview of Test Results

The test for the City Chatbot application were performed as per the Test Plan and covered the entirety of the SRS that was established at the beginning of the project. The following roles were assigned to team members in accordance with the testing plan Additionally the below schedule was used to guide the test through the testing cycle to ensure all functional test were captured in testing. In addition to the outlined plan below, all JUnit test were complete and reviewed at the time of code submission to its respective repository.

Table 2 Testing Responsibilities

| **Position** | **Team Member** | **Responsibilities** |
| --- | --- | --- |
| **Testing Lead** | Rob Lee | Writing and executing functional test. Signs off on pass-fail and severity criteria. Submits reports. |
| **Quality Lead** | Ben Murray | Witnessing and validating testing. Confirming pass-fail criteria. Determines status of testing and resumption criteria. |
| **Testing Team** | Benjamin Fetterman  Hanim Danur  James Cornelius  Ben Murray  Robert Lee | Responsible for conducting testing. |

Table 3 Schedule for Test Tasks

|  |  |  |
| --- | --- | --- |
| **Due Date** | **Task** | **Comment** |
| 09/29/2020 | Test plan generated | Submit to project stakeholder for approval. |
| 09/29/2020 | Testing specifications documented and delivered to testing team | Included in this document |
| 10/09/2020 | Testing environment set up | Start with unit testing for developers and then system integration tests. |
| 10/20/2020 | Milestone 3 Deliverable | Test plan updates and interim test reports delivered for Milestone 3 |
| 10/28/2020 | Complete specified tests | Complete tests by the end of the final development sprint. |
| 11/03/2020 | Defect analysis and test summary report generated | Use severity list in section 7 of this document rating impact of defects |
| 11/03/2020 | Test report delivered | Submit to project stakeholder for approval. |

1. Test Summary

Detailed Test Results

The following test were run to confirm the functional validity of the chatbot application and are identified in the below table with an indication of their status. All test passed and fulfilled the requirements established for this project. As such, this concludes testing for this project with no major bugs or defects found at this time.

Table 4 Test Report

| **Test ID** | **Test Name** | **Functional Requirement ID** |  |
| --- | --- | --- | --- |
| **Test-1.1** | Chatbot Greeting | CH-2 | **PASSED** |
| **Test-2.1** | Chatbot Zoning/Permit Links | CH-3 | **PASSED** |
| **Test-2.2** | Chatbot Application Links | CH-4 | **PASSED** |
| **Test-2.3** | Chatbot Retrieve Regs | CH-5 | **PASSED** |
| **Test-2.4** | Chatbot Retrieve Standards | CH-6 | **PASSED** |
| **Test-3.1** | Chatbot Retrieve Permits Applicable to Zoning Statues | CH-7 | **PASSED** |
| **Test-3.2** | Chatbot GIS Map Layer Integration | CH-8 | **PASSED** |
| **Test-4.1** | Chatbot Address Requirements | CH-9 | **PASSED** |
| **Test-4.2** | Address Validation | CH-10 | **PASSED** |
| **Test-5.1** | Chatbot Other Information | CH-11 | **PASSED** |
| **Test-5.2** | Chatbot Help | CH-13 | **PASSED** |
| **Test-6.1** | Retrieve Permit | CH-7 | **PASSED** |

1. Subsystem Test

All test for this section, with the exception of the specified JUnit test, were complete using Selenium testing plugin. The testing was conducted by the Testing Lead and Verified by the Quality Lead once the final code release had been completed on 10/27/2020. Testing was conducted and verified on 10/28/2020 in accordance with the above schedule.

Chatbot Greeting

Table 5 Test-1.1 Chatbot Greeting

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-1.1 |
| **Test Name** | Chatbot Greeting |
| **Description** | This functional requirement was to ensure that the chatbot would greet the user with each opening and ensure they are aware of the uses for the chatbot. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Chatbot Zoning/Permit Links

Table 6 Test-2.1 Chatbot Zoning/Permit Links

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-2.1 |
| **Test Name** | Chatbot Zoning/Permit Links |
| **Description** | This functional requirement ensures that a link is returned to the user as requested when the correct information is given. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Chatbot Application Links

Table 7 Test-2.2 Chatbot Application Links

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-2.2 |
| **Test Name** | Chatbot Application Links |
| **Description** | This functional requirement ensures that a link is returned to the user as requested when the correct information is given. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Chatbot Retrieve Regs

Table 8 Test-2.3 Chatbot Retrieve Regs

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-2.3 |
| **Test Name** | Chatbot Retrieve Regs |
| **Description** | This functional requirement ensures that a link is returned to the user as requested when the correct information is given. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Chatbot Retrieve Standards

Table 9 Test-2.4 Chatbot Retrieve Standards

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-2.4 |
| **Test Name** | Chatbot Retrieve Standards |
| **Description** | This functional requirement ensures that a link is returned to the user as requested when the correct information is given. |
| **Pictures** | Single standard:    More than one standard: |
| **Selenium Test** | **PASSED** |

Chatbot Retrieve Permits Applicable to Zoning Statutes

Table 10 Test-3.1 Chatbot Retrieve Permits Applicable to Zoning Statues

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-3.1 |
| **Test Name** | Chatbot Retrieve Permits Applicable to Zoning Statues |
| **Description** | This test is confirmed by searching for development standards with an address and confirming the address returns the correct zone standards. |
| **Pictures** | Chatbot:    Returned Zone Standard:  <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART2ZODIALLAUSZOECST_CH17.26SPPUZODI_17.26.040SPPUDIGEDEST>    Confirmation of city Zone: |
| **Selenium Test** | **NA** |

Chatbot GIS Map Layer Integration

Table 11 Test-3.2 Chatbot GIS Map Layer Integration

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-3.2 |
| **Test Name** | Chatbot GIS Map Layer Integration |
| **Description** | This test is confirmed by searching for development standards with an address and confirming the address returns the correct zone standards. |
| **Pictures** | Chatbot:    Returned Zone Standard:  <https://library.municode.com/ca/pasadena/codes/code_of_ordinances?nodeId=TIT17_ZONING_CODE_ART2ZODIALLAUSZOECST_CH17.26SPPUZODI_17.26.040SPPUDIGEDEST>    Confirmation of city Zone: |
| **Selenium Test** | **NA** |

Chatbot Address Requirements

Table 12 Test-4.1 Chatbot Address Requirements

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-4.1 |
| **Test Name** | Chatbot Address Requirements |
| **Description** | This functional requirement is used to ensure the chatbot request an address for each transaction so accurate information can be returned to the user. |
| **Pictures** | Permit address required    Development standards address required |
| **Selenium Test** | **PASSED** |

Address Validation

Table 13 Test 4.2 Address Validation

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-4.2 |
| **Test Name** | Address Validation |
| **Description** | This functional requirement is used to determine if a city address is recognized by the chatbot in order to process request. Additionally, it test to ensure a bad address prompts the user to try again. |
| **Pictures** | Bad Address Verification    Good Address Verification |
| **Selenium Test** | **PASSED** |

Chatbot Other Information

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-4.2 |
| **Test Name** | Chatbot Other Information |
| **Description** | This functional requirement is used to provide the user with an additional place to look for information if the Chatbot is unable to answer their question. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Chatbot Help

Table 14 Test-5.2 Chatbot Help

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-5.2 |
| **Test Name** | Chatbot Help |
| **Description** | The functional requirement should return the proper contact information for a user that request contact information. |
| **Pictures** |  |
| **Selenium Test** | **PASSED** |

Retrieve Permit Unit Test

This is one of the unit test that has been developed for the REST API. It is representative of all of the unit tests. The other unit tests can be provided upon request.

Table 15 Test-6.1 Retrieve Permit

| **Title** | **Test Results** |
| --- | --- |
| **Test ID** | Test-6.1 |
| **Test Name** | Retrieve Permit |
| **Description** | The retrieveInformation method in the ProcessRequest should return the correct URL for the given type, action, object, and zone ID. |
| **Pictures** |  |
| **Junit Test** | **PASSED** |

1. Test Execution Findings

Metrics

Test metrics are essential for assessing the functionality of an application in relation to what its requirements are. Metrics can help establish a baseline of where the system is at and reveal areas that need to be worked on. For this project, test metrics come from SonarQube analysis scans and from Selenium test results.

SonarQube Overview

SonarQube is a software application for performing code scans in order to find errors, vulnerabilities, and bad practices in a code base (SonarQube, *n.d.*). The UMGC SWEN 670 DevOps team has set up SonarQube to run a code scan each time that a build for the Spring IO master branch is started. Once a scan is complete, a report is published that contains the metrics on errors, vulnerabilities, and bad practices along with many more categories. Below is a screenshot that displays the results from the final SonarQube scan on the Spring IO master branch.

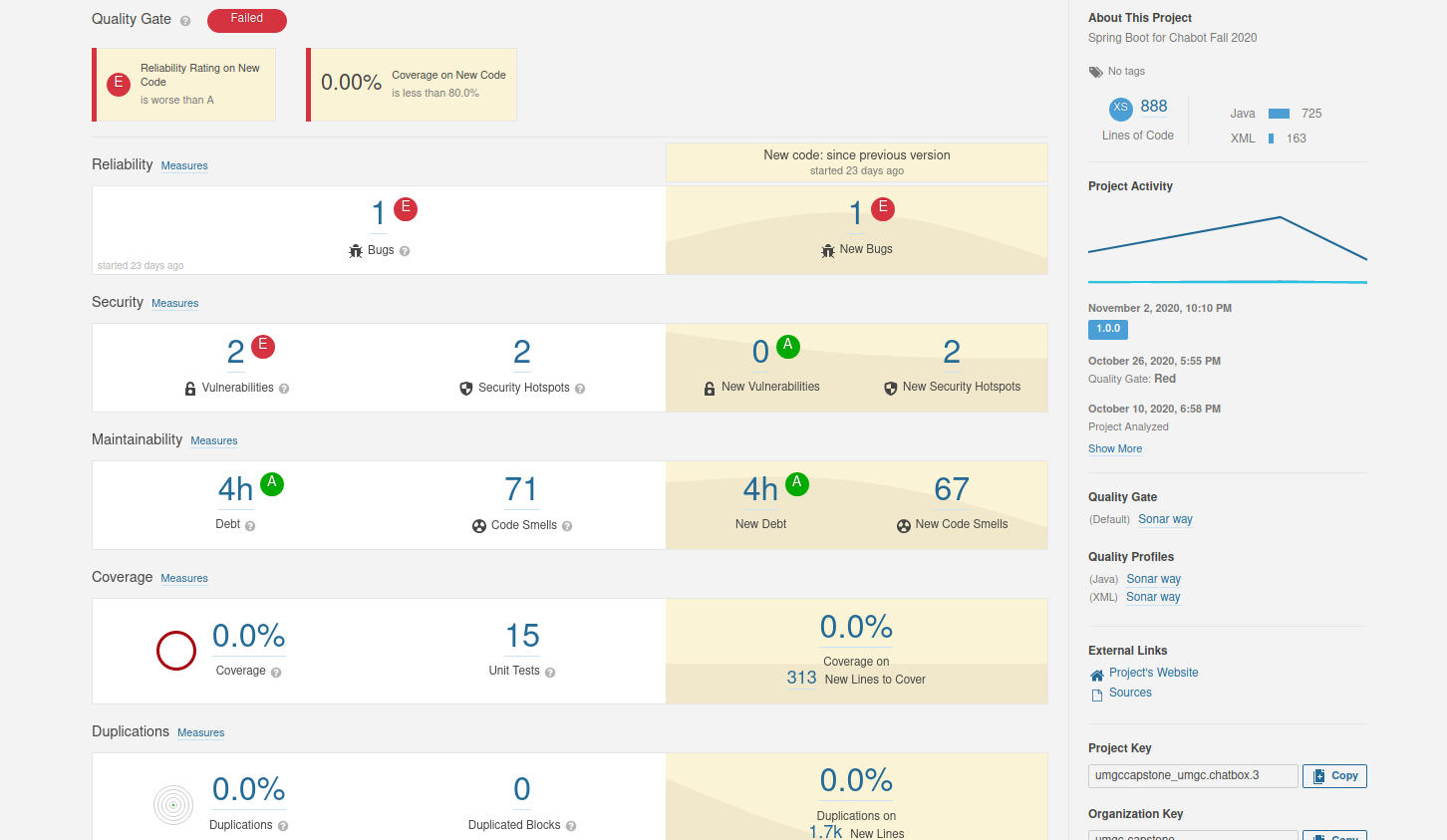


Figure 1 SonarQube Report

### Selenium Overview

Selenium is a collection of free tools for creating automated tests that run on a web browser (Selenium, *n.d.*). The test cases listed in section 4 of this document were created using Selenium IDE. Selenium IDE is a browser extension that enables for quick and easy development of Selenium tests. Below are the overall results from running the Selenium tests.

Table 15 Overall Results of Selenium Tests

|  |  |
| --- | --- |
| **Item** | **Result** |
| Features to be Tested | 11 |
| Features that Passed Testing | 11 |
| Features that Failed Testing | 0 |

There were no test cases that failed and thus no findings to report.

### JUnit Overview

Junit is a testing framework for writing unit tests in Java (JUnit, *n.d.*). Unit tests are small tests that test whether a single piece of code functions as expected (Software Testing Fundamentals, *n.d.*). For this project, unit tests were only created for the Spring IO REST API application. Because unit tests are meant to be small simple tests, the unit tests for this project do not reach out to the MySQL database nor to the MapQuest Geocode API. Instead, the Mockito framework is used to return mock data for any unit tests that need to reach out to either the MySQL database or to the MapQuest Geocode API. Below are the overall results for running the JUnit unit tests.

Table 16 Overall Results of JUnit Unit Tests

|  |  |
| --- | --- |
| **Item** | **Result** |
| Unit Tests to Run | 31 |
| Unit Tests that Passed | 31 |
| Unit Tests that Failed | 0 |

There were no unit tests that failed and thus no findings to report.

1. Exit Criteria

All exit criteria for this test have been met and testing has been concluded. No bugs or defects have been identified or require addition inspection. The testing of the bot has concluded the successful completion of the established test plan and encompasses all requirements for the project as established in the SRS.

1. Acronyms and Abbreviations

|  |  |
| --- | --- |
| Acronym/Abbreviation | Definition/References |
| AI | Artificial Intelligence – An application that aims to mimic human intelligence. |
| API | Application Programming Interface |
| CH | Chatbot |
| CU | Conditional Use Permit |
| DevSecOps | Development, Security and Operations – Group of developers responsible for the deployment and security of an application. |
| ECUP | Expressive Use Permit |
| EPSP | East Pasadena Specific Plan |
| ETL | Extract, transfer, and load |
| FGSP | Fair Oaks/Orange Grove Specific Plan |
| GIS | Geographical Information System – System for working with geographical data. |
| HTTP | Hyper Text Transfer Protocol – A network protocol for specifying how servers and clients communicate with each other. |
| IDE | Integrated Development Environment |
| KML | Keyhold Markup Language – A markup language for visualizing geographical data. |
| LASP | Lincoln Avenue Specific Plan |
| MCUP | Minor Conditional Use Permit |
| MVC | Model-View-Controller |
| REST API | Representation State Transfer – An API for interacting with data. |
| SRS | Software Requirements Specification |
| UI | User Interface – The part of the application that users use to interact with the application. |
| URL | Uniform Resource Locator |
| WSL | Windows Subsystem for Linux |

1. References

JUnit. (*n.d.*). JUnit. Retrieved from <https://junit.org/junit4/>

Selenium. (*n.d.*). The Selenium Browser Automation Project. Retrieved from <https://www.selenium.dev/documentation/en/>

Software Testing Fundamenstals. (*n.d.*). Unit Testing. Retrieved from <https://softwaretestingfundamentals.com/unit-testing/>

SonarQube. (*n.d.*). SonarQube Documentation. Retrieved from <https://docs.sonarqube.org/latest/>