Software Quality Assurance Plan

For MVC Reporting Website Framework

(MVC RWF)

Version 1.0

Submitted in partial fulfillment of the Masters of Software Engineering Degree.

Thaddeus Thomas Tuck

CIS 895 – MSE Project

Department of Computing and Information Sciences

Kansas State University

Committee Members

Dr. Daniel Andresen

Dr. Mitchell Nielsen

Dr. William Hsu

Change Log

Version #	Changed By	Release Date	Change Description
1.0	Thaddeus Tuck	3/25/2018	Initial Release

Table of Contents

1. Purpose

The purpose of this document is to define a process to ensure that the developed software for the project will be of a high quality. This document lists the tools and techniques that will be used to develop the application and provide a list of deliverables for each phase.

2. Reference

[1] Vision Document
[2] Project Plan
[3] IEEE Standard for Software Quality Assurance Planning
[4] IEEE Guide for Software Quality Assurance Planning

3. Management

3.1 Organization

Supervisory Committee

- Dr. Daniel Andresen
- Dr. Mitchell Nielsen
- Dr. William Hsu

Major Professor

- Dr. Daniel Andresen

Developer

- Thaddeus Thomas Tuck

3.2 Tasks

All tasks for the projects are recorded in detail in the Project Plan document. The Project Plan document will be updated with the suggestions of the supervisory committee after the first presentation with subsequent updates to the Gantt chart as necessary for schedule changes.

3.3 Responsibilities

3.3.1 Supervisory Committee

The primary responsibility of the committee members is to attend all three project presentations and provide feedback to the developer based on the project progress.

3.3.2 Major Professor

The major professor holds two responsibilities; to act as a supervisory committee member and to supervise and evaluate the project with the developer on a regular basis.

3.3.3 Developer

The primary responsibility of the developer is to produce the product within the required time frame while fulfilling the requirements and specifications of the provided project documentation. In addition, the developer must maintain the documentation for the project and maintain an open dialog with the major professor about project feedback and requirements.

4. Documentation

All documents related to the project can be found at http://thaddeustuckmastersproject.azurewebsites.net/.

5. Standards, Practices, Conventions, and Metrics

5.1 Documentation Standards

IEEE standards will be followed for all applicable documentation throughout the project.

5.2 Coding Standards

For the code written in C#, C# naming conventions and best practices will be followed when available and where applicable.

5.3 Metrics

COCOMO II will be used to estimate the project cost in terms of time and effort.

6. Reviews and Audits

At the end of each phase, the documentation, source code, and the executable product will be evaluated by the supervisory committee.

7. Testing

Details about testing procedures and the expected results will be listed in the Test Plan. Since a major component of this project is separation of concerns unit testing is possible and will be completed for each layer and component of the MVC Reporting Framework. Once the system is built it is possible to use Google Chrome's debugging tools to do performance analysis. It is the responsibility of the developer to correct any discovered bugs and then devise testing for the resolution and ensure that the bug, resolution, and test will be documented in the Test Plan.

8. Problems Reporting and Corrective Actions

All problems detected during the development of the framework will be recorded in the Software Problem Report spreadsheet. Each problem detected will be recorded by defining the problem description, the time consumed to fix the problem, and the actions that were taken to correct the issue. If there are any problems that were not solved they will be brought to the attention of the major professor and discussed.

9. Tools, Technologies, and Methodologies

The following tools are used for coding, testing, and documenting the reports:

Visual Studio 2017 – IDE for software development
C# - Language the software will be developed in
Razor – Language for C#, HTML, CSS, and JavaScript Integration

Azure – Platform for hosting the website for testing
Microsoft Word – for documentation of development
Microsoft Excel – for risk and problem report tracking and time logs
Gantt Project – for drawing the Gantt chart (project planning)
Visual Paradigm – for software design development
Azure and ASP.NET MVC – for developing the project webpage
Visual Studio Testing Framework – for unit testing the components of the framework
Selenium C# Testing – for integration testing of the produced view to verify the browser renders correctly

10. Code and Media Control

All source code developed will be controlled using Visual Studio Team Management services. All source code is maintained on the developer's personal computer.

All documents will be maintained on the developer's personal computer stored on Google Drive with file names corresponding to the version numbers and dates on which there were created. Each document includes a change log and all documents are posted to the project's webpage: http://thaddeustuckmastersproject.azurewebsites.net/

11. Risk Management

Software risks will be logged in a Software Risk Reporting and Mitigation spreadsheet as well discussed with the major professor during meeting hours.

12. Deliverables

The list below presents the deliverables at each phase of the project:

Phase I

- Vision Document
- Project Plan
- Prototype Demonstration
- Software Quality Assurance Plan
- Presentation

Phase II

- Vision Document
- Project Plan
- Software Requirements Specification
- Architecture Design
- Test Plan
- Software Risk Reporting and Mitigation Document
- Technical Inspection Checklist
- Executable Architecture Prototype
- Action Items
- Presentation

Phase III

- Component Design
- Source Code
- Executable Project
- User Manual
- Project Evaluation
- Software Problem Reports
- Presentation