Software Quality Assurance Plan
Organizational Alert System

For

CS 895 MSE Project

Department of Computer Science

Kansas State University

Submitted to

Dr. Mitch Neilsen

Dr. Torben Amtoft

Dr. Scott Deloach

Submitted by

Angela Hall

10/28/18

1. Purpose

The purpose of this document is to layout the processes, responsibilities, documentation, and oversight for the project that will ensure the project's quality.

2. References

1. Code Conventions for the Java Programming Language. (n.d.). Retrieved from https://www.oracle.com/technetwork/java/codeconvtoc-136057.html
2. IEEE standard for Software Quality Assurance Planning
3. IEEE guide for Software Quality Assurance Planning
4. Kuppa, K. (n.d.). Airline Reservation System [Scholarly project]. Retrieved from http://people.cs.ksu.edu/~kaavya/Vision Document_MSE_Phase I.pdf
5. Tuck, T. Software Quality Assurance Plan for MVC Reporting Website [Scholarly project]. Retrieved from http://thaddeustuckmastersproject.azurewebsites.net/Content/Documents/Software%20Quality%20Assurance%20Plan%20v1.0.pdf
6. Nehl, B. Multiagent Control of Traffic Signals [scholarly project]. Retrieved from http://people.cs.ksu.edu/~bnehl/nehl_inception_sqap.pdf

3. Management

3.1 Management Organization

3.1.1 Supervisory Committee

- Dr. Mitch Neilsen
- Dr. Torben Amtoft
- Dr. Scott Deloach

3.1.2 Major Professor

- Dr. Mitch Neilsen

3.1.3 Developer

- Angela Hall

3.1.4 Technical Inspectors

- Richard Waliser
- Thaddeus Tuck

3.2 Responsibilities

3.2.1 Supervisory Committee

The supervisory committee will review each phase of the project which will be presented to them at the culmination of each milestone. They will offer feedback and suggestions during the presentation that will steer the iterative advancement of the project deliverables.

3.2.2 Major Professor

The major professor will supervise and review the progress of the developer.

3.3.3 Developer

The developer will be responsible for the project process, design, documentation, and code.

3.3.4 Formal Technical Inspectors

The Formal Technical Inspectors will be provided a checklist based on the system requirements and will review the project to ensure that it meets the specifications. The inspectors will provide the completed inspection with a report or letter of approval.

3.4 Tasks

All project tasks are documented in the Project Plan document. At the end of Phase I the Project Plan will be reviewed by the Supervisory Committee and any changes needed will implemented in Phase II.

4. Documentation

4.1 Phase 1 Documentation

Time Log
Vision Document
Project Plan
Software Quality Assurance Plan
Presentation 1

4.2 Phase 2 Documentation

Time Log
Action Items from Phase 1 Presentation
Updated Vision Document
Updated Project Plan
Formal Requirements Specification
Architectural Design
Test Plan
Formal Technical Inspection Checklist
Executable Architectural Prototype
Presentation 2

4.3 Phase 3 Documentation

Time Log
Action Items from Phase 2 Presentation
User Manual
Component Design
Source Code
Project Evaluation
References
Formal Technical Inspection Letters
Presentation 3

5. Standards, Practices, Conventions and Metrics

5.1 Documentation Standard

All project documentation will use the IEEE standards for documents.

5.2 Coding Standard
All code will be written using the [1] Coding Conventions for the Java Programming Language.

5.3 Metrics
The COCOMO II model was used in the time and cost estimations for this project.

6. Reviews and Audits
The supervisory committee will review all documentation at the end of each phase.

7. Test and Problem Reporting
The developer will create a Test Plan as part of the Phase II documentation process. The test plan will list the tests and expected results. Test results, bugs, and resolutions will be documented in the test plan.

8. Tools, Techniques, and Methodologies
The following toolchain will be used for this project:

- Eclipse 4.8 Photon will be used for coding, testing, and debugging.
- Java SDK 1.8
- WindowBuilder Java Libraries
- JUnit Testing
- Git versioning control system

9. Code and Media Control
Code versioning and backup will be controlled with the Git open source versioning control system and will be stored online at https://github.com/ifpthenq/headsup.

Documentation versioning will be manually controlled and all versions will be stored online at http://www.angiehall.com/mse/index.php

10. Risk Management
If any risks are identified, the developer will communicate the risk and remediation strategy to the major professor. The major professor will advise the developer if they feel further remediation is needed.