**Software Quality Assurance Plan**

For Multiagent Control of Traffic Signals

Version 1.0

Submitted in partial fulfillment of the requirements of the degree of MSE

Bryan Nehl

CIS 895 – MSE Project

Kansas State University

Table of Contents

[1 Purpose 3](#_Toc304393605)

[2 References 3](#_Toc304393606)

[3 Management 3](#_Toc304393607)

[3.1 Management Organization 3](#_Toc304393608)

[3.1.1 Supervisory Committee 3](#_Toc304393609)

[3.1.2 Major Professor 3](#_Toc304393610)

[3.1.3 Developer 3](#_Toc304393611)

[3.1.4 Technical Inspectors 3](#_Toc304393612)

[3.2 Tasks 3](#_Toc304393613)

[3.3 Roles and Responsibilities 3](#_Toc304393614)

[3.3.1 Supervisory Committee 3](#_Toc304393615)

[3.3.2 Major Professor 3](#_Toc304393616)

[3.3.3 Developer 4](#_Toc304393617)

[3.3.4 Technical Inspectors 4](#_Toc304393618)

[4 Documentation 4](#_Toc304393619)

[4.1 Purpose 4](#_Toc304393620)

[4.2 Minimum Documentation Requirement 4](#_Toc304393621)

[5 Standards, Practices, Conventions and Metrics 4](#_Toc304393622)

[6 Reviews and audits 4](#_Toc304393623)

[7 Test 4](#_Toc304393624)

[8 Problem reporting and corrective action 4](#_Toc304393625)

[9 Tools, techniques, and methodologies 5](#_Toc304393626)

[9.1 Tools 5](#_Toc304393627)

[9.2 Techniques 5](#_Toc304393628)

[9.3 Methodologies 5](#_Toc304393629)

[10 Code control, media control and supplier control 5](#_Toc304393630)

[11 Record collection, maintenance and retention 5](#_Toc304393631)

[12 Risk management 5](#_Toc304393632)

# Purpose

This document serves as the Software Quality Assurance (SQA) plan for the Multiagent Control of Traffic Signals system. This project is the Master of Software Engineering final project for Bryan Nehl.

# References

1. Bryan Nehl. “Multiagent Control of Traffic Signals Project Plan 1.0”, http://people.cis.ksu.edu/~bnehl/repos/macts.git/
2. IEEE Std. 730-1998, IEEE Standard for Software Quality Assurance Plans, IEEE 1998.
3. IEEE Std. 730.1-1995 IEEE Guide for Software Quality Assurance Planning, IEEE, 1995.
4. Kyle Hill. “GMoDS-based Runtime Agent Role Interpreter SQA Plan 1.0”. http://people.cis.ksu.edu/~kylhill/phase\_1/sqa\_plan.pdf.

# Management

## Management Organization

### Supervisory Committee

* Dr. Scott DeLoach
* Dr. Gurdip Singh
* Dr. David Gustafson

### Major Professor

* Dr. Scott DeLoach

### Developer

* Bryan Nehl

### Technical Inspectors

* Inspector 1 (TBD)
* Inspector 2 (TBD)

## Tasks

The project plan[1] details the major tasks that are to be completed.

## Roles and Responsibilities

### Supervisory Committee

asf

### Major Professor

asdf

### Developer

asdf

### Technical Inspectors

adf

# Documentation

Asdf

## Purpose

Asdf

## Minimum Documentation Requirement

Phase 1

Phase 2

Phase 3

# Standards, Practices, Conventions and Metrics

Asdf

# Reviews and audits

Asdf

# Test

Asdf

# Problem reporting and corrective action

When the developer becomes aware of a problem they will notify the major professor of the issue, the impact or severity of the issue and the mitigation or remediation plan. The developer may discover the problem themselves or have it communicated to them by a reviewer. The developer will maintain a risk/defect log. The risk log will include: identification number, type (risk or defect) who identified the risk, short title, description of the risk, mitigation or remediation used, responsible reviewer, date when identified, date developer completed and date when accepted/approved.

The major professor will advise the developer of any discernible issues and suggest potential approaches to mitigation or remediation. Communication will primarily occur between presentations and via email.

The supervisory committee will identify issues for the developer to address as well. They will primarily communicate during the presentations and with follow-up email. The supervisory committee will review and approve mitigation and remediation actions as subsequent presentations.

Technical inspectors will review the system and provide detailed feedback of potential problem areas. Reviewers will provide feedback via email or written response.

# Tools, techniques, and methodologies

## Tools

* Python 2.7 will be used for implementing the agents and tests.
* RabbitMQ is a message queue platform which is how the agents communicate.
* MongoDB is a document store database which is where agents store settings and knowledge base information.
* PyMongo is a Python interface library for MongoDB.
* pika is a Python interface library for RabbitMQ.
* Unittest is a Python module for doing unit testing.
* PyChecker is a Python module for checking code for typical errors.
* Coverage is a Python module for evaluating test coverage.
* Simulation for Urban Mobility is an environment for running traffic simulations.

## Techniques

asdf

## Methodologies

adf

# Code control, media control and supplier control

The developer will also make use of a repository that is on the development machine. The source code and other project documents will be kept in a git repository at http://people.cis.ksu.edu/~bnehl/repos/macts.git/. The local repository will be pushed to the web based repository at least once a week. All milestones will be tagged in the repository.

Supplier control is not applicable to this project.

# Record collection, maintenance and retention

Technical reviewer documentation will be transformed into a PDF format and included in a feedback folder.

PDF versions of all documents will be created prior to each presentation. The PDF versions will be inside the portfolio folder. The developer will also create a zip file for each presentation which will include all project materials (documents, source code and diagrams). The zip file will be made available for download on the http://people.cis.ksu.edu/~bnehl/ web page.

At a minimum the git repository and the website will be maintained for the duration of the project. At the completion of the project, the repository and website files will stay in place for an indefinite period of time. At the request of the Major Professor or the supervisory committee the materials may be moved to a department website.

# Risk management

The developer will be responsible for identifying, and mitigating risk. When a new risk item is identified it along with the remediation strategy will be communicated to the major professor.

The major professor will review and advise the developer regarding the risks and mitigation strategies. The major professor will also let the developer know if they identify additional potential risks.