Software Quality Assurance Plan

Michigan Treasury Local Government Data Parser

[Adham Abdalla](mailto:adhama@umich.edu),

Hasan Alameh,

[Tanis Daniels-Wanamaker](mailto:tanisd@umich.edu),

[Ahmed Mawari](mailto:afmawari@umich.edu)

Winter 2021

# 

[**1.0 Introduction**](#_s85926fxhvku) **6**

[1.1 Scope and intent of SQA activities](#_t8cl6bn6gbav) 6

[The purpose of the SQA document is to insure a level of quality for delivering the final product to the client. The team wants to make sure we deliver every specification the client provided.](#_4scnw4jcfdc) 6

[1.2 SQA organizational role](#_h7q2u9ipxzd9) 6

[**2.0 SQA Tasks**](#_c0jesz1isvhu) **6**

[**2.1 Task Overview**](#_gx9wic8wznkq) **6**

[2.1.1 Description of SQA task m](#_gnad84forsze) 6

[2.1.2 Work products and documentation](#_2qwwd8fjw7w5) 7

[2.2 Standards, Practices and Conventions (SPC)](#_gsmefn9irx91) 7

[2.3 SQA Resources](#_o8a4dbqzp83) 8

[**3.0 Reviews and Audits**](#_sy42xzl31khf) **8**

[3.1 Generic Review Guidelines](#_wxdpx8mt3fl7) 8

[3.1.1 Conducting a Review](#_dcqjypsmydx9) 8

[3.1.2 Roles and Responsibilities](#_ewbwz0htbb90) 8

[3.1.3 Review work products](#_ol3qylbuwb9c) 8

[3.2 Formal Technical Reviews](#_5f4lcvtf9og1) 9

[3.2.1 System Specification review](#_a9jhp7a7t35d) 9

[3.2.1.1 Description and focus of the System Specification review](#_8o3nhvvp2x6) 9

[3.2.1.2 Timing of the review](#_22vcy9qqzuzf) 9

[3.2.1.3 Work products produced](#_e9wkwmg3uoqt) 9

[3.2.1.4 Review checklist](#_svio1v3aiccu) 9

[3.2.2 Software Project Plan review](#_shtw8d54iatf) 9

[3.2.2.1 Description and focus of the Software Project Plan review](#_4fkzyfagxl02) 9

[3.2.2.2 Timing of the review](#_yrbmq1oe8csx) 9

[3.2.2.3 Work products produced](#_harnm6x069fl) 9

[3.2.2.4 Review checklist](#_eoqmqnzht8gi) 9

[3.2.3 RMMM review](#_drr7o2opdqwz) 10

[3.2.3.1 Description and focus of the RMMM review](#_968lx050op4j) 10

[3.2.3.2 Timing of the review](#_a5lt8ov1u8q3) 10

[3.2.3.3 Work products produced](#_tzvkfd36ru84) 10

[3.2.3.4 Review checklist](#_9iiar1op4jc2) 10

[3.2.4 Requirements reviews (models, specification)](#_7k6vkuafrlwq) 10

[3.2.4.1 Description and focus of the Requirements Specification review](#_r6qw2herpqtw) 10

[3.2.4.2 Timing of the review](#_9pnseg7gjaxp) 10

[3.2.4.3 Work products produced](#_lwlrwp3pm3sr) 10

[3.2.4.4 Review checklist](#_69n814k43gel) 10

[3.2.5 Data Design review](#_u2zv2acr133a) 11

[3.2.5.1 Description and focus of the Data Design review](#_1mbj8vt6rsn) 11

[3.2.5.2 Timing of the review](#_asuy07byujs3) 11

[3.2.5.3 Work products produced](#_7loj0qevj8ur) 11

[3.2.5.4 Review checklist](#_24fq5x4841x0) 11

[3.2.6 Architectural Design review](#_vzv34g4jni5r) 11

[3.2.6.1 Description and focus of the Architectural Design review](#_nxz9jhjnexco) 11

[3.2.6.2 Timing of the review](#_9iqy9u9bw8gp) 11

[3.2.6.3 Work products produced](#_r61llv2eulad) 11

[3.2.6.4 Review checklist](#_bbj6qsfkeq27) 12

[3.2.7 Interface (GUI) Design review](#_fnhl4mo7xid9) 12

[3.2.7.1 Description and focus of the Interface Design review](#_sfq7w5n89ch7) 12

[3.2.7.2 Timing of the review](#_mn49khilx871) 12

[3.2.7.3 Work products produced](#_5otyzvcg4fre) 12

[3.2.7.4 Review checklist](#_xnt691f00cta) 12

[3.2.8 Component Design review(s)](#_u5fg9spql7kn) 12

[3.2.8.1 Description and focus of the Component Design review](#_aojjgr4mxjjd) 12

[3.2.8.2 Timing of the review](#_rzq4lf6uaq58) 12

[3.2.8.3 Work products produced](#_s8ruqcmcbdo6) 12

[3.2.8.4 Review checklist](#_d2znf7u3glob) 12

[3.2.9 Code Reviews](#_vrrm6chip9xo) 13

[3.2.9.1 Description and focus of the Component Design review](#_bja4tuu98610) 13

[3.2.9.2 Timing of the review](#_6i37b4djm7qi) 13

[3.2.9.3 Work products produced](#_zdlsf86alerf) 13

[3.2.9.4 Review checklist](#_hnfcneqkdzq1) 13

[3.2.10 Test Specification review](#_6dosqyubmmtm) 13

[3.2.10.1 Description and focus of the Test Specification review](#_x6pyf3vw4ydo) 13

[3.2.10.2 Timing of the review](#_u8675gmxzpyj) 13

[3.2.10.3 Work products produced](#_cfk6kiz4d9wl) 13

[3.2.10.4 Review checklist](#_mxh2zqduw6j5) 13

[3.2.11 Change control reviews and audits](#_9q9iw72h5px7) 14

[3.3 SQA Audits](#_1w0nfz1r9ilt) 14

[**4.0 Problem Reporting and Corrective Action/Follow-up**](#_oyjm51xmvz0b) **14**

[4.1 Reporting mechanisms](#_rp8ady6ex0k7) 14

[4.2 Responsibilities](#_u95l7wsql41y) 14

[4.3 Data collection and evaluation](#_5rj6kotf3dp0) 14

[4.4 Statistical SQA](#_96qg2gk6fo5s) 14

[**5.0 Software Process Improvement Activities**](#_yoqersvu3l7q) **15**

[5.1 Goal and objectives of SPI](#_wyyetjgiprx2) 15

[5.2 SPI tasks and responsibilities](#_kbpu27kdhnnb) 16

[**6.0 Software Configuration Management Overview**](#_a89jqbss7f4s) **16**

[**7.0 SQA Tools, Techniques, Methods**](#_40xr5878rev) **16**

[**8.0 Appendix**](#_d6llhffwk38c) **16**

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 1.0 Introduction

## 1.1 Scope and intent of SQA activities

## The purpose of the SQA document is to insure a level of quality for delivering the final product to the client. The team wants to make sure we deliver every specification the client provided.

## 1.2 SQA organizational role

Having a small team and multiskilled team members, the roles can easily overlap. However, we recognize the importance of structure in ensuring quality. The team's plan is to have Ahmed Mawari as the Software Quality assurance leader, Hasan Alameh and Tanis Daniels-Wanamaker as lead developers and Adham Abdalla as testing lead. The team is agile and can quickly focus efforts in any area that requires it. The team’s focus is in meeting clien’s specification and catching and fixing errors early in the development.

# 2.0 SQA Tasks

# 2.1 Task Overview

## 2.1.1 Description of SQA task *m*

Tasks involve:

●Frequent communication with the client to review development progress and ask any questions that will help the team.

●Extensive and consistent design documentation

●Developers studying the previous team’s version of the project and making use of their excellent documentation to proactively tackle any issues that the previous team mentioned.

●System to ensure a working product as a whole system after every sprint

## 2.1.2 Work products and documentation

Work product and documentation produced for each task:

● Biweekly team meetings to review documentation with the aim to get feedback.

● Updating documents is done by communicating a correction request to the team, followed by the team reviewing the relevant document and making any necessary corrections to the affected document(s).

● Developers will utilize the internet and available tools that they used for previous projects that can be useful for us once development commences.

● The team will conduct code reviews after each sprint and check for any bugs or logic flaws.

## 2.2 Standards, Practices and Conventions (SPC)

● Frequent communication with the client to review progress and ask any questions that will help the team

The team has met with the client, received feedback on the provided documentation, and received answers for questions posed by the team.

● Extensive and consistent design documentation

The software quality assurance lead will ensure all documentation is complete and all assigned tasks are fulfilled. The team constantly communicates on Discord so that everyone is informed.

● Developers will utilize the internet and available tools that they used for previous projects that can be useful for us once we start coding.

The design documents will be used as coding begins, serving as a template for the code. The team will ensure that all of the client’s specifications are accounted for. Looking at past attempts at this project will help guide the team in the right direction as well.

● System to ensure a working product as a whole system after every sprint

Code reviews will be conducted and feedback will be communicated to all team members to ensure the quality of the code and lack of bugs.

## 2.3 SQA Resources

The team will use GitHub for version control of the software as well as tracking bugs. Discord and Zoom will be used for communication.

Each team member is an essential asset and their in team communication and communication with the client is vital.

# 3.0 Reviews and Audits

## 3.1 Generic Review Guidelines

### 3.1.1 Conducting a Review

We will mainly have two reviews for the project. First, we will conduct an internal review between our team members to make sure we have met our requirements and client expectations. The second review will be conducted with the client, where we will make sure we are all on the same page, and everything agreed on is present in the product.

### 3.1.2 Roles and Responsibilities

As mentioned in section 1.2 (SQA organizational role), the roles and responsibilities of team members may overlap. Still, the following roles are set by the team:

* Software Quality Assurance Lead: Ahmed Mawari
* Lead Developers: Hasan Alameh and [Tanis Daniels-Wanamaker](mailto:tanisd@umich.edu)
* Testing Lead: [Adham Abdalla](mailto:adhama@umich.edu)

### 3.1.3 Review work products

A list of bugs/issues with the software will be created and managed through GitHub’s “issue” functionality. The bugs will be ranked according to their impact on the user experience and how much they affect the data parsing. The issues will then be resolved in decreasing order of importance by different team members (moreso by the Lead Developers) in a reasonable time window.

## 3.2 Formal Technical Reviews

### 3.2.1 System Specification review

#### 3.2.1.1 Description and focus of the System Specification review

The System Specification is expanded more upon in the SRS document. We will review the general structure and delivery of the code as a whole, ensuring it meets the requirements we have set.

#### 3.2.1.2 Timing of the review

The System Specification Review will be executed after the completion of the System Specification, estimated to be less than a month after the start of development. Its completion in a timely manner will prevent future development issues.

#### 3.2.1.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the System Specification review. It will note any adjustments to the software’s internal system design, as well as including potential issues and their solutions.

#### 3.2.1.4 Review checklist

* Have the system requirements been checked and verified?
* Do the system requirements note any impacts on performance?
* Are the system requirements consistently detailed?

### 3.2.2 Software Project Plan review

#### 3.2.2.1 Description and focus of the Software Project Plan review

The product deadlines and requirements will be tracked and reviewed. In the event that deadline estimates aren’t accurate, they shall be amended at the time of review. This is expanded more upon in the Software Project Plan document.

#### 3.2.2.2 Timing of the review

The Software Project Plan review will be executed after the completion of the Software Project Plan, estimated to be less than a month after the start of development. Reviews may also be held after each sprint. The review’s completion in a timely manner will allow appropriate scheduling of development.

#### 3.2.2.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Software Project Plan review. It will note any errors in deadline estimation, amending them when necessary.

#### 3.2.2.4 Review checklist

* Is there enough time to deliver a finished product?
* Does each feature have an appropriate estimated deadline?
* Have project tasks been assigned to team members appropriately?

### 3.2.3 RMMM review

#### 3.2.3.1 Description and focus of the RMMM review

Risk management will be conducted in case any of our work does not go as planned. Different measures will be taken depending on the type of issue we face. This is expanded more upon in the Risk Mitigation and Monitoring Plan document.

#### 3.2.3.2 Timing of the review

The RMMM review will be executed after the completion of the Software Project Plan, estimated to be less than a month after the start of development. The review’s completion in a timely manner will allow for appropriate handling of potential risks in the event that they occur.

#### 3.2.3.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the RMMM review. It will note any missing risks or errors in existing risks. A team discussion will be held in order to propose new risks or amending those with errors.

#### 3.2.3.4 Review checklist

* Have all project risks been listed?
* Have the consequences to all risks been noted and explained?
* Have the contingency plans (solutions) for all risks been thoroughly explained?

### 3.2.4 Requirements reviews (models, specification)

#### 3.2.4.1 Description and focus of the Requirements Specification review

This is to review the expectations and requirements we have agreed on with the client. Use cases, data dictionaries, and other requirements diagrams will be tracked and reviewed.

#### 3.2.4.2 Timing of the review

The Requirements Specification review will be executed after the completion of the Requirements Specification, estimated to be less than a month after the start of development. The review’s completion in a timely manner will allow for appropriate internal design that will prevent future issues during development.

#### 3.2.4.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Requirements Specification review. It will note any design defects, with a team discussion being held in order to propose solutions.

#### 3.2.4.4 Review checklist

* Have all notable design flaws been listed and explained?
* Are any additional improvements or quality of life features required?
* Is the software design the best implementation of program functionality?

### 3.2.5 Data Design review

#### 3.2.5.1 Description and focus of the Data Design review

Data input, output, and flow will be reviewed and monitored. We will review whether the data output is as expected, and will be testing different inputs. The SRS document also expands more on this subject.

#### 3.2.5.2 Timing of the review

The Data Design review will be executed after the completion of the System Requirements Specification and System Specification, estimated to be less than a month after the start of development. The review’s completion in a timely manner will allow for appropriate internal data structures that will prevent future issues during development. Team members will also conduct Data Design reviews after each sprint to ensure compliance with the original documentation. In the event that a member believes a data structure could be improved, a team meeting will be held to discuss such changes.

#### 3.2.5.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Data Design review. It will note any design defects in the data structures, with a team discussion being held in order to propose solutions.

#### 3.2.5.4 Review checklist

* Are the data structures arranged in the most efficient manner?
* Does the usage of data objects simplify the program’s code?
* Can data objects efficiently communicate when necessary?

### 3.2.6 Architectural Design review

#### 3.2.6.1 Description and focus of the Architectural Design review

Code architecture and software design patterns will be reviewed. We will compare the design with implementation and make sure they follow the same logic and blueprints.

#### 3.2.6.2 Timing of the review

The Architectural Design review will be executed after the completion of the System Requirements Specification and System Specification, estimated to be less than a month after the start of development. The review’s completion in a timely manner will prevent future issues regarding internal architecture during development. Team members will also conduct Architectural Design reviews after each sprint to ensure compliance with the original documentation. In the event that a member believes an architectural element could be improved, a team meeting will be held to discuss such changes.

#### 3.2.6.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Architectural Design review. It will note any design defects in the internal architecture, with a team discussion being held in order to propose solutions.

#### 3.2.6.4 Review checklist

* Are all architectural elements arranged in the most efficient manner?
* Could the architecture be improved to increase program performance?
* Can the architectural design be replicated in Python?

### 3.2.7 Interface (GUI) Design review

#### 3.2.7.1 Description and focus of the Interface Design review

The expectations set for the interface (usability, user-friendliness, and reliability) will be tested and reviewed. Elements of the interface design will be changed if requested by the client.

#### 3.2.7.2 Timing of the review

The Interface Design review will be executed after the completion of the System Requirements Specification and System Specification, estimated to be less than a month after the start of development. The review’s completion in a timely manner will prevent future issues regarding internal architecture during development. Team members will also conduct Interface Design reviews after each sprint to ensure compliance with the original documentation. In the event that a member believes an interface element could be improved, a team meeting will be held to discuss such changes.

#### 3.2.7.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Interface Design review. It will note potential improvements and any errors in the interface, with a team discussion being held in order to propose solutions.

#### 3.2.7.4 Review checklist

* Is the interface intuitive and easy to use?
* Do interface functions follow user expectations?
* Is the “readme” file helpful to the user?

### 3.2.8 Component Design review(s)

#### 3.2.8.1 Description and focus of the Component Design review

The basic components (input PDF, output CSV) will be reviewed to make sure they are handled by the code as expected.

#### 3.2.8.2 Timing of the review

The Component Design review will be executed after each sprint to ensure compliance with the original documentation. In the event that a component deviates from its expected value(s), a team meeting will be held to discuss any necessary changes.

#### 3.2.8.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Component Design review if deemed necessary. It will note potential improvements and any errors in the components, with a team discussion being held in order to propose solutions.

#### 3.2.8.4 Review checklist

* Is the input file in the correct format?
* Does the output file display values in the correct format?
* Does the output file have any unanticipated results?

### 3.2.9 Code reviews

#### 3.2.9.1 Description and focus of the Component Design review

Code inspections will be regularly held to conduct internal code reviews of the different team members. We need to make sure that the code is following the product’s documentation and design, and the code logic is appropriate for the best user experience.

#### 3.2.9.2 Timing of the review

The Code reviews will be executed after each sprint to ensure compliance with the original documentation. As each sprint will only include a handful of logic nodes, the reviews could prove to be relatively brief. In the event that an issue arises, it will be logged into the GitHub issue tracker and a team meeting will be held to discuss finding a solution.

#### 3.2.9.3 Work products produced

Any unexpected, undesirable results, logic issues, or other bugs found in the code will be logged into the GitHub issue tracker, along with a ranking of importance based on the issue’s adverse effects on the program.

#### 3.2.9.4 Review checklist

* Are plenty of comments included to explain the source code?
* Is the source code easy to read and understand?
* Is the source code modular and easy to maintain?

### 3.2.10 Test Specification review

#### 3.2.10.1 Description and focus of the Test Specification review

The Test Specification review will focus on the types of tests and testing procedures performed on the software.

#### 3.2.10.2 Timing of the review

The Test Specification review will be executed after the completion of the Test Specification to ensure the appropriate tests are performed on the program. Another review will be executed just prior to program completion to ensure all tests were attempted and passed. In the event that a team member believes a test is missing or could be improved, a team meeting will be held to discuss such changes.

#### 3.2.10.3 Work products produced

The SQA leader will produce an evaluation document pertaining to the Test Specification review if deemed necessary. It will note any changes and errors in the testing procedures, with a team discussion being held in order to discuss further changes.

#### 3.2.10.4 Review checklist

* Does the testing thoroughly cover all lines of code?
* Are all tests meaningful and useful?
* Are all tests passed? If not, are proposed solutions included?

### 3.2.11 Change control reviews and audits

The use of Github will make it easy for the team members to control code push/pull changes and revert to previously working commits if any errors occur.

## 3.3 SQA Audits

The most important part is making sure the clients are always informed about our recent progress towards the final products. Following that, internal team audits, comments, and suggestions are always made to make sure our quality testing is acceptable. The use of Github and instant chat between team members makes it easy to communicate and make changes in a timely manner.

# 4.0 Problem Reporting and Corrective Action/Follow-up

## 4.1 Reporting mechanisms

Problems will be reported in a document stored on the teams google drive and will be communicated to each team member in our Discord.

## 4.2 Responsibilities

Software quality assurance lead is responsible for all corrective actions and follow ups with team members, the rest of the team will assist but it is mainly the responsibility of the Software quality assurance lead.

## 4.3 Data collection and evaluation

A running list of errors/defects will be used to evaluate the current state of the software. Communication with the client and getting feedback will help the team with design and functionality uncertainties.

## 4.4 Statistical SQA

The team will rank defects from 1 (minor) to 4(critical) in a running error/defect list and this will help the focus efforts in fixing bugs in order of importance to the quality of the software.

# 5.0 Software Process Improvement Activities

## 5.1 Goal and objectives of SPI

The goal and objectives of SPI are to make sure all defects are dealt with accordingly and record how much time it takes in order to fix any defect that shows up. First the team must identify how critical the defect is and if it affects the software heavily. The goal is to use previously recorded defects and how long it took to fix them in order to plan for less mistakes in the upcoming future.

A scale to determine how critical the defects are:

| Critical score | Description |
| --- | --- |
| 4 | Critical and cannot continue without fixing it |
| 3 | Big issue that doesn’t stop the program but will affect the performance of the program greatly. |
| 2 | Moderate issue that does something that allows the program to continue going and is noticeable but doesn't impact the end result too badly. |
| 1 | Very small issues that are usually just cosmetic or something along those lines. |

## 5.2 SPI tasks and responsibilities

The whole group is responsible for any tasks or issues to come up since we are a small team and will be working together in order to take responsibility and make the product to the best of our ability.

# 6.0 Software Configuration Management Overview

The team is always in reach of each other in case of any urgue communication needed but the team constantly meets in order to resolve any issues that come up. The developers will always update the document with any new defects or bugs that come up in order to review them and allow time to fix them. The client will receive every new milestone achieved in order to stay on board with the team and see how far along the development of the project has come. The team will use google drive, github, and discord to keep track of the software and documents.

# 7.0 SQA Tools, Techniques, Methods

The tools that are going to be used are programming platforms such as PyCharm or Visual Studio in order to make the source code for the project. Github and Google Drive will be used in order to keep everything in the drive shared with the whole team for editing and viewing. Techniques and methods used will be designing documents, researching about the project, splitting tasks between team members and communicating with each other and the clients. Another important technique to be implemented is time management in order to keep track of how far the project is coming along.

# 8.0 Appendix