Chemistry

**Software Quality Assurance**

**SQA Design**

**Roland Heintze, John Gibbons, Tim Elam and Chris Lansing**

Contents

[Support Material: 2](#_Toc354171617)

[Inspection Section: 2](#_Toc354171618)

[Variables: 3](#_Toc354171619)

[Structure: 3](#_Toc354171620)

[Finalization: 3](#_Toc354171621)

|  |  |  |  |
| --- | --- | --- | --- |
| **Revision Number** | **Revision Date** | **Author** | **Summary of Changes** |
| 1 | 4-17-2013 | John Gibbons | Initial creation of document and draft. |
| 2 | 4-19-2013 | John Gibbons | Second revision of draft and added additional items. |
|  |  |  |  |

# Support Material:

The following material should be available to person(s) completing this check.

* A full and completed UML diagram should be provided.
* All state diagrams should be properly designed as well as logically flow and should be provided.
* All main uses cases involving this program's requirements should be provided.
* All use cases must contain at least one use case scenario that should be provided.
* All documents must fully reflect all the customer's requirements.
* The clients signature of approved must be provided showing approval of design.
* All documents should adhere to Object Oriented Design principles.
* Any change or update requests for documentation should be provided.
* All classes, interfaces, and objects should be provided in the UML documentation.

# Inspection Section:

* Have all classes been named appropriately for the code segments and processes they contain?
* Are all class relations properly connected?
* Have the functions been named appropriately for the processes they will be committing?
* Have the methods been named appropriately for the processes they will be committing?
* Have the variables been named appropriately for the information they are storing?
* Have professional procedures been followed?
* Are all the relationships and number of possible instances between entities correct?
* Has any core functionality been left out that was outlined in the requirements documentation?

# Variables:

* Are there any redundant functions, methods, or variables?
* Are any variables hard-coded?
* If hard-coded variables exist, if they are changed will they deter the program from functioning properly?
* Has any and all variables and arrays been properly set up and instantiated?
* Are all method return types properly type casted?
* Are all Data types properly typed and/or casted?

# Structure:

* Will changes to the class diagrams alter the functionality of the program?
* Will changes to data types within the diagrams cause problems elsewhere?
* Will changes in method parameters cause problems elsewhere?
* Will any changes possibly result in pointer exceptions?

# Finalization:

* Does the design contain appropriate error catching methods?
* Does the error catching methods provide adequate information to the user on how to resolve the error?
* Is the design modular enough such that further requirements and design changes are possible?
* Is the design realistic enough such that the development team can complete the tasks in the allotted timeframe?
* Does the design take into account all constraints in which may hinder the functionality of the program?
* Does the design ensure optimized usage and flow for the user?
* Does the design logically flow from one process/step to the next?
* Does the design and layout of the animations match the requirements?
* Are all options for the user to perform from the SRS document accounted for?