Chemistry

**Software Quality Assurance**

**SQA Testing**

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

Contents

[Requirements Testing: 2](#_Toc354172191)

[Functional Design: 2](#_Toc354172192)

[Testing Techniques: 3](#_Toc354172193)

[Environmental Testing: 3](#_Toc354172194)

[Acceptance Testing: 3](#_Toc354172195)

|  |  |  |  |
| --- | --- | --- | --- |
| **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. |
|  |  |  |  |

# Requirements Testing:

* Have all requirements from the SRS requirements document been satisfied?
* Does all the requirements in the SRS document fully meet the customers' requirements?
* Does the code properly and fully implement those requirements?
* Can the requirements' code be tested?
* Are the requirements clear and concise?
* Has the customer signed off on all requirements for verification?
* Does the program execute each step of the naming process and animation process according to the customers' models?
* Does the software meet security requirements?
* Does the software meet privacy requirements?
* Does the documentation have a description of higher architecture?

# Functional Design:

* Does the design all the user to perform any and all tasks promised?
* Does the design incorporate future software updates and expansions?
* Does the software allow for the user to create their own pentadecane molecule?
* Does the software allow for the user to enter in what they believe to be the name for the pentadecane molecule?
* Does the software allow for some actions to be performed multiple times?
* Does the design handle well under stress?
* Does the software inform the user when a mistake has been made and how to resolve it/
* Does the design give the user a simple interface to work with?
* Does the design allow the user to backtrack to previous forms?
* Does the design allow testers to add comments?
* Does the design allow the users to add comments?
* Does the design address techniques and tools that shall be used to assure software quality assurance?
* Does the design catch all exceptions correctly?

# Testing Techniques:

* Does the software pass Black Box testing?
* Does the software pass White Box testing?
* Does the software pass Unit testing?
* Does the software pass Integration testing?
* Does the software pass System testing?
* Does the software pass Alpha testing?
* Does the software pass Beta testing?
* Does the software pass Acceptance testing?

# Environmental Testing:

* Does the program run correctly on all operating systems specified in the SRS document?
* Does the program work correctly with all drivers?
* Does the program work smoothly in these environments? (not choppy)
* Does any aspect of the program crash or freeze during any step in these different environments?
* Is any functionality hindered by running in any of the operating systems specified in the SRS document?

# Acceptance Testing:

* Are all previous issues and bugs fixed?
* Is the client satisfied with the finished project?
* Is the finished project ready for release?