Contents

[1 Introduction 2](#_Toc317836546)

[1.1 Purpose 2](#_Toc317836547)

[1.2 Scope 2](#_Toc317836548)

[2 Applicability 2](#_Toc317836549)

[3 Applicable Documents 2](#_Toc317836550)

[4 Document Structure 3](#_Toc317836551)

[5 Documentation 3](#_Toc317836552)

[6 Development 3](#_Toc317836553)

[7 Implementation 3](#_Toc317836554)

[7.1 Operation 3](#_Toc317836555)

[7.2 Transition 3](#_Toc317836556)

[7.3 Revision 3](#_Toc317836557)

Software Quality Plan

# Introduction

## Purpose

The purpose of this software quality plan is to define the standards to which the language and roles project will be developed which will enable SegFault Software to make Language & Roles a high quality project.

## Scope

This document will assure that: (1) Language & Roles is a fully functioning and robust application; (2) the program suits the client’s needs; (3) documentation is of a high standard; (4) the development process cycle itself is high quality with all components acting in a professional manner.

# Applicability

This document will be used throughout the: analysis, design, implementation, testing and evaluation of Language & Roles. This SQP is in effect until all deliverables including the L&R program, documentation, installation, presentation and 2 years of phone support have been delivered to the client in their entirety.

This plan applies only to software developed by SegFault Software. Products and services provided by third parties may not abide by the same quality standards.

# Applicable Documents

Microsoft UI Guidelines: <http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=2695>

# Document Structure



Quality can be broken down into several categories; this plan is structured according to this breakdown.

* Documentation – These apply just to the documentation used to support the program
* Development – These standards apply to the development process itself
* Implementation – These standards apply to the L&R program
  + Operation – Refers to the way the program functions
  + Transition – Refers to the ability to move parts of the program and interface it with other programs
  + Revision – Refers to the ease of supporting and modifying the program

# Documentation

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard Number** | **Category** | **Definition** | **Time of Implementation** |
| DO1 | Consistency | All documents must show consistent facts and figures | Throughout |
| DO2 | Legibility | All documents must be written in British English | Throughout |
| DO3 | Legibility | All documentation must use proper grammar and spelling | Throughout |
| DO4 | Consistency | All documents must use Segoe UI font | Throughout |
| DO5 | Consistency | The default font size for all documents is 11 | Throughout |
| DO6 | Comprehensiveness | All documents must be include all sections used by reputable established companies | Start of each document |
| DO7 | Explicit | All documents should contain a list of definitions for any terms not in common use | Throughout |
| DO8 | Explicit | Unnecessary technical terminology should be avoided | Throughout |

# Development

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard Number** | **Category** | **Definition** | **Time of Implementation** |
| DE1 | Punctuality | All team members should attend all meetings | Throughout |
| DE2 | Punctuality / Communicativeness | Emails should be read and responded to within 24 hours | Throughout |
| DE3 | Planning | A plan that covers the entire project must be made to schedule tasks | Beginning of project |
| DE4 | Planning | The plan must be adjusted throughout the development cycle so that it consistently reflects reality | Throughout |
| DE5 | Delegating | Skills matrices must be completed by all team members to allow tasks to be delegated properly | Beginning of project |
| DE6 | Responsibility | Each team member is responsible for making sure their work is compliant with all quality standards | Throughout |
| DE7 | Communication | Each member must communicate either by email or in person that they have started a piece of work to prevent duplication | Throughout |

# Implementation

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard Number** | **Category** | **Definition** | **Time of Implementation** |
| I1 | Correctness | The program must match the: specification that the client provided, the software requirements specification and the conceptual and technical designs | Implementation |
| I2 | Correctness | Each test in the test plan must correspond with a point in the technical design | Testing |
| I3 | Reliability | Results, tests and configuration data must be transmitted over the network accurately | Implementation |
| I4 | Reliability | Results must be accurate to at least the nearest whole number | Implementation |
| I5 | Reliability | Graph data must be accurate to at least 3 decimal places | Implementation |
| I6 | Efficiency | Distribution of new results over the network will take at most 5 seconds per client over a 100mb/s TCP/IP network | Implementation |
| I7 | Efficiency | Distribution of new questions and answers will take a maximum 1 second per client per question/answer pair over a 100mb/s TCP/IP network | Implementation |
| I8 | Efficiency | Distribution of new configuration options will take at most 5 seconds per client over a 100mb/s TCP/IP network | Implementation |
| I9 | Integrity | It must be impossible to use the software without logging in | Design |
| I10 | Integrity | Only teachers and system administrators will be able to access results and configuration data | Design |

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard Number** | **Category** | **Definition** | **Time of Implementation** |
| I11 | Usability | The UI of the pupil section of the software must not use language that does not comply with a British reading age of 4 | Design |
| I12 | Efficiency | The program, not including the questions and answers, must not use more than 100mb of backing store space | Implementation |
| I13 | Correctness | The program must run on the client’s systems | Implementation |
| I14 | Usability | The GUI must meet the Windows User Interface Guidelines (29/09/2010) | Design |
| I15 | Integrity | The program must not introduce security flaws into the school’s IT system | Design / Implementation |
| I16 | Portability | The software must not require any configuration to install on the client’s computer | Implementation |
| I17 | Portability / Integrity | It must be possible to fully remove the program including all files | Implementation |
| I18 | Maintainability | GUI objects will use a suitable standardised prefix for the object type followed by an underscore and then an appropriate variable name | Design |
| I19 | Maintainability | All code blocks will be indented by exactly one tab | Implementation |
| I20 | Maintainability | Only one statement will occur per line | Implementation |
| I21 | Maintainability | Left hand comparisons shall be used i.e. 42==a not a==42 | Implementation |
| I22 | Testability / Flexibility / Maintainability /  Interoperability | The configuration data will be stored in a separate readable file from the executable | Design |

|  |  |  |  |
| --- | --- | --- | --- |
| **Standard Number** | **Category** | **Definition** | **Time of Implementation** |
| I23 | Testability / Flexibility / Maintainability /  Interoperability | Questions and answers will be stored in a separate file from the executable | Design |
| I24 | Flexibility / Reusability | Class names will be named with semantic names in upper camel case ie FileInputOutput | Implementation |
| I25 | Flexibility / Reusability | Class names will not use acronyms unless it is more commonly used than its unabbreviated counterpart | Implementation |
| I26 | Flexibility / Reusability | Variable names will be semantic and in lower camel case with the first word written in lower case and subsequent words beginning with a capital letter i.e. numberOfCars | Implementation |
| I27 | Maintainability / Flexibility | Variables with single character names should not be used unless scope is confined to a single code block i.e. I, j, k | Implementation |
| I28 | Flexibility / Reusability | Method names will have semantic lower camel case names similar to variable names with the first word written in lower case and subsequent words beginning with a capital letter i.e. getNumberOfCars | Implementation |
| I29 | Flexibility / Reusability | All names used within the code will fully follow standard British English language including all grammar and punctuation | Implementation |
| I30 | Maintainability / Testability | Version control will use the .NET convention of: major version, minor version, revision number and build number i.e. 1.0.5.20042 | Implementation |
| I31 | Reliability | The program must be fully tested in compliance with IEEE 829-2008 |  |