

Quality Management Plan

Data Mining of Digital Library Usage Data

Team 07

Clients

Jewel Ward

Team Members

Project Manager: Maxim Krivokon

Developer: Bo Lee

Developer: Genesan Kim

Developer: Vu Nguyen

IV&Ver: Shing-Cheung Chan

IV&Ver: Marie Chi

IV&Ver: Kristine Guevara

Table of Contents

1	Introduction	Error! Bookmark not defined.	
	1.1	<i>Overview</i>	1
	1.2	<i>References</i>	1
	1.3	<i>Change Summary</i>	1
2	Quality Guidelines	2	
	2.1	<i>Design Guidelines</i>	2
	2.2	<i>Coding Guidelines</i>	2
	2.3	<i>Quality Assessment</i>	2
	2.4	<i>Process Assurance</i>	4
	2.5	<i>Product Assurance</i>	4
	2.6	<i>Problem Reporting and Tracking System</i>	5
	2.7	<i>Configuration Management</i>	5
	2.8	<i>Testing Guidelines</i>	9
	2.9	<i>Defect and Change Management</i>	10
3	Common Definition Language	11	
4	Appendix	12	
	4.1	<i>Appendix with Agile Artifact Review Form Set and Instructions</i>	12

Version control

Date	Author	Version	Changes made
2/4/05	Bo Lee	1.0	<ul style="list-style-type: none">• Initial Draft
2/25/05	Bo Lee	1.1	<ul style="list-style-type: none">• Updated Figure 1 Change request procedure• Updated List of figures, List of tables, List of contents
4/01/05	Genesan Kim	1.2	<ul style="list-style-type: none">• Added section 2.5.3 Automated Analysis
4/23/05	Vu Nguyen	2.0	<ul style="list-style-type: none">• Removed Avoidable and Unavoidable defect types.• Updated Section 2.9.2 Tracking• Changed configuration tool used from ClearCase to Subversion

List of Figures

Figure 1 Change request procedure	8
--	---

List of Tables

Table 1: Configuration Item and Rationale 6

1 Purpose

1.1 Overview

The primary purpose of this plan is to provide practices and standards to the development team members and the future maintainer(s) of the system developed as the result of CS577 (Fall 2004-Spring 2005) project, the Data Mining of Digital Library Usage Data, in order to achieve the high quality software. With respect to these practices, the result system is developed and delivered to the clients with all capabilities met specific requirements described in the SSRD and the customer/user's operational concept defined in the OCD.

To assure the quality of the system, the project utilized several review techniques along with the MBASE guidelines and WIN-WIN spiral Model. The review techniques are:

- Internal Agile Review
- Peer Review
- Verification and Validation by the IV&Vers.

Please refer to LCP section 4 for more information.

1.2 References

MBASE Guidelines v 2.4.2:

http://cse.usc.edu/classes/cs577a_2004/guidelines/MBASE_Guidelines_v2.4.2.pdf

Project Website:

<http://seacliff.usc.edu/~team7b/>

1.3 Change Summary

1.1	Updated Figure 1 Change Request Procedure.
1.2	Added the section Automated Analysis, 2.5.3
2.0	Updated coding guidelines. Provided more specific guidelines on coding standards. Updated Section 2.3 Quality Assessment Updated Section 2.9.2 Tracking. Changed configuration tool used from ClearCase to Subversion

2 Quality Guidelines

2.1 Design Guidelines

- The design depicted in SSAD is based on the common agreements between all the developers in the team, so that each developer has the same concept on the system and how the design is implemented to prevent future time-consuming argues on the base design concepts in the middle of the implementation.
- The design of this system is documented in SSAD using UML diagrams created with Rational Rose.
- All changes occurred in the design phase will be modified in Rose model. Afterward, this model will be used to modify the code.
- The design of all classifiers, classes, and objects associated with this project should be simply mapped to classes and objects in the code.

2.2 Coding Guidelines

- For better communication among team members and better maintainability, we take the C++ coding guideline stated in <http://www.nfra.nl/~seg/cppStdDoc.html>.
- The header of each source code file should contain: version control and history, implementation consideration, unit verification, integration, and additional information that help understanding of the source code.
- The information about a version of each module, a name of person who did or modified each code, and change details should be placed at the beginning of each file.
- All source codes should be properly and sufficiently commented, for example, there should be a comment at each function or procedure call.
- All function, procedure, and variable names should have some understandable meaning.

2.3 Quality Assessment

The quality assessment will be conducted among team members, the IV&V members, and the clients. Also, the quality assessment will help to improve the quality of software. The reviews will be done on each iteration artifacts to capture problems as soon as possible. Iteration Assessments are also performed at the end of each iteration in Construction Phase.

Please refer to LCP section 4.1.2 and section 4.4 for more information.

2.3.1 Peer Reviews

Please refer to the Peer Review Plan for more specific details

2.3.1.1 Types of peer reviews

Please refer to the Peer Review Plan for more specific details.

2.3.1.2 Degree of data gathering

Severity

- **Major**
 - Condition that causes operational failure, malfunctioning, or prevents attainment of an expected result.
 - ◆ This can be a case where the design of the architecture is such that it doesn't satisfy the requirements of the client.
 - Information that would lead to an incorrect response or misinterpretation of the information by the user.
 - ◆ Wrong UML Diagrams
 - Deviates from the requirements.
 - ◆ Assumptions by the development team and not clarifying it with the client
- **Minor**
 - Is a violation of standards, guidelines, or rules, but would not lead to a discrepancy report.
 - ◆ If MBASE Guidelines followed is not the one prescribed for the coursework
 - Information that is undesirable but would not cause a malfunction or unexpected results.
 - Information that if left uncorrected may decrease maintainability
 - ◆ Spelling mistakes and grammatical errors

Priority

- How important the issues to the stakeholders and clients. Values are high/medium/low.

- These can be those defects that affect on the main system capabilities (SSRD 4.2), which may drive different, undesirable results when not get fixed.

Criticality

- The degree of loss when the issues did not get fixed. Values are high/medium/low
 - These can be those defects caused by misunderstanding of the constraints, or miscommunication between team members or between client and team members.

Please refer to the Peer Review Plan for more specific details.

2.4 Process Assurance

Details for this section under development.

(Please refer to LCP section 4.4.1 for more information.)

2.5 Product Assurance

2.5.1 Requirement Verification

The client meetings will be conducted throughout the development life cycle in order to discuss problems and reach agreements. As a result, it could guarantee that the development progress is on schedule and that all clients' requirements are fulfilled.

Any evolution requirement changes from the clients will be documented in the SSRD and all other documents as evolutionary information to develop the future version of this software.

All the requirements in SSRD are analyzed and simulated simultaneously. When a problem occurs, we held a client meeting and demonstrate the requirement with the problem and make compromise and agreement on the requirement with the clients. During each iteration, we inspect the requirements by tracking their consistency among artifacts and make sure they were achieved as desired by testing.

2.5.2 Independent Verification and Validation

We have three IV&Vers: Shing-Cheung Chan, Marie Chi, and Kristine Guevara. They are responsible to review and test all the deliverables to ensure the completeness and correctness of each deliverable and verify and validate the project's compliance with its documentation, code standards, and feasibility of the solution. They will mainly check on the consistency among artifacts/documents according to the MBASE GL v.2.4.2, evaluate the feasibility of the solutions to each requirement, and check on the readability of the artifacts/documents.

The IV&V tasks on the project include:

- Review all deliverables and provide quality reports for each document.
- Evaluate quality reports.
- Perform acceptance test and report results

2.5.3 Automated Analysis

The tools used to perform this management detailed above include concern logs that the IV&V complete with respect to our documents and the MBASE guidelines. These logs are then passed to the development team and changes are made respectively and problem lists are updated (which is also included in the same excel file). This tool can be found at the following link,

[http://sunset.usc.edu/classes/cs577b_2005/tools/AgileArtifactReview_Form\(exCeBASEv1.2a\).zip](http://sunset.usc.edu/classes/cs577b_2005/tools/AgileArtifactReview_Form(exCeBASEv1.2a).zip).

The IV&V also review the documents by making milestone review documents which can be found at the following link in the IV&V assignments subsection,
http://sunset.usc.edu/classes/cs577b_2005/assignments/index.html.

2.6 Problem Reporting and Tracking System

As shown in LCP for QR, all of concerns and defects will be reported in the form of Agile Review Form provided by CSE. The reports from team members including IV&V's will be collected via electronic copies. All reports are to be sent to the author and the review leader of each artifact/document. (refer to Peer Review Plan). Collecting QRs, reporting, tracking of the concerns, defects, and problems is the result of the Internal Agile Review that the team is performing throughout the development. These reports and forms must be available on the project website for the clients and TAs. The responsibility of monitoring and ensuring the availability of these reports is mainly on the project manager.

2.7 Configuration Management

Every document has its own version change history. All the major changes to the document should be stated in the change history with its modifier, version, and date.

Source code files should follow the coding standard and coding conventions.

2.7.1 Configuration Item and Rationale

The Configuration Management Items (CIs) and the rationale to select them as CIs are depicted in the Table 1 in the next page.

CI name	Category	Volatility and Rationale	Impact of Change and Rationale
OCD, SSRD, SSAD, LCP, FRD, Iteration Plan, Transition Plan, PRP, QMP	Documents	Low to None There will be no change in these documents during construction phase, unless the clients request.	Severe Since those documents are the blueprint of how this system will be developed, even a small change can affect the software development a lot.
Code	Source Code	Medium to High The code is changed all the time during the construction phase. The change rate is high at first part of construction phase. After the program gets mature, the system becomes stable, the change rate will be slowed down.	High Coding is the process that consumes the most of time in CS 577b, so any change in coding may cause schedule slippage.
Test plan, Test case	Tests	Medium Test Plan will not be changed unless the client requires. Test report may change during the construction phase.	Medium The result of test cases will be used to verify functions and modifying codes.
User Manual	Object	Medium to High The user manual will be developed along with coding.	None The user manual has no impact to the delivered system.
Rose Model	Object	Low to None There will be no change in the Rose Model during construction phase, unless the clients request.	Severe Since Rose Model is the part of other documents which play as the blueprint of how this system to be developed, even a small change can affect the software development a lot.

Table 1: Configuration Item and Rationale

2.7.2 Identification System

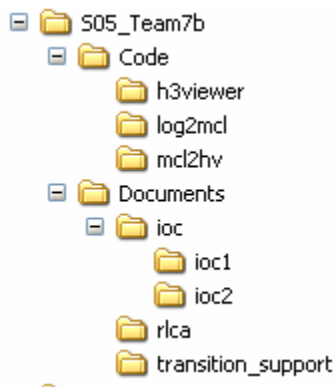
Refer to LCP Section 4.3.1 Product Element Identification

2.7.3 Storage and Configuration Items

All documents will be made available to view at
<http://seacliff.usc.edu/~team7b/>

The team is using Subversion 2.0 as a tool for configuration management.

Folder structure in Subversion is as follows:



2.7.4 Configuration Control

- The following flowchart represents the change management process that the project applies. This process is applied to changes to the baseline artifacts including source code.

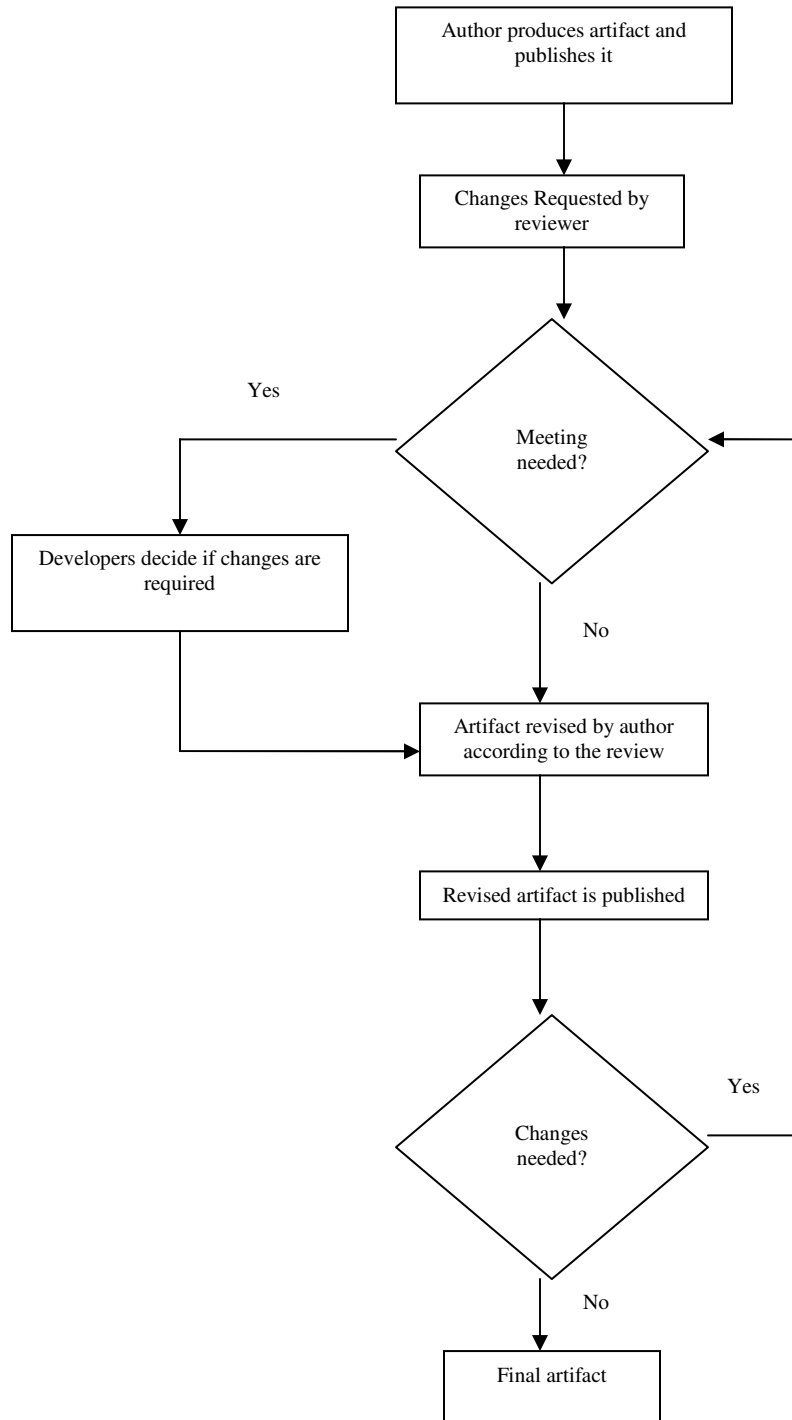


Figure 1 Change request procedure

2.7.5 Status and Accounting

All of the changes made must be reported in the change history in that respective document/artifact/code by the modifier. In addition, the status of the document must be reported in weekly progress report (PR) in order to provide other members information about that document. For the major change during the Peer Review, it will be reported in Agile Review Report as QR.

2.7.6 Baseline Events

As shown in LCP, the baseline event of this project in CS577b will be

1. RLCA
2. Construction Iteration I
3. Construction Iteration II
4. Transition Phase 1

2.7.7 Resources and Personnel

- We are using Subversion as the tool for configuration management.
- Genesan Kim is in charge of configuration management and performs CM tasks including administrating Subversion.
- Bo Lee is in charge of Peer Review Plan and assignment of reviewers.

2.7.8 Tools

- Subversion is used for version controlling.
- The Agile Review Form is used to keep review results.

2.8 Testing Guidelines

Please refer to the Test Plan for specific details

2.8.1 Testing Requirements

Please refer to the Test Plan for specific details

2.8.2 Deliverables

Test report will be generated for each test session to indicate whether this test is success or not. If not, the test report will give reason and found errors.

2.8.3 Tools

None

2.9 Defect and Change Management

The reviewers will use the latest Agile Review Form template to report any defects and concerns with the standard described in the form. The review process will be performed during a certain period of time with certain personnel assigned to that particular review. Refer Peer Review Plan for more specific details.

2.9.1 Reporting procedures

In the Peer Review, participants will use the latest Agile Form template.

Refer to LCP Section 4.1.5 for details on communication process in the project.

2.9.2 Tracking

By using version control, each document will change its version number, when the author conducts major changes. All those changes are to be logged in change summary in that document.

Because the lack of software system for tracking defect (such as PVCS Tracker), we use Excel spreadsheet to track defects. CM manager is responsible for tracking all defects reported by team members including IV&Vers.

2.9.3 Resolution

- Each team member is responsible for his/her primary part.
- The team manager must have and keep track of all the artifacts.
- The assigned reviewers and IV&Vers are responsible for the completeness and correctness of the reviewing artifact.

3 Common Definition Language

Term	Definition
ARB	Architecture Review Board
FRD	Feasibility Rationale Description
LCA	Life Cycle Architecture
LCO	Life Cycle Objective
LCP	Life Cycle Plan
RLCA	Rebasedlined Life Cycle Architecture
OCD	Operational Concept Description
SSRD	Software and System Requirement Definition
SSAD	Software and System Architecture Description
PR	Progress Report
QMP	Quality Management Plan
PRP	Peer Review Plan

4 Appendix

4.1 Appendix with Agile Artifact Review Form Set and Instructions

Agile Artifact Review Form Set and Instructions can be found at:

http://sunset.usc.edu/classes/cs577a_2004/assignments/Team/AgileInternalReview.zip