Software Project Management Plan for a Cervical Image Database and Patient History System (CIDPHS)

433-340 Team V

October 31, 2003

Maintained by: zyu

Version: 4.1	31^{st} 10 2003

Abstract

This project aims to build a *Cervical Image Database and Patient History System* for storing patients' cervical images and related medical histories for diagnostic purposes as well as monitoring for the development of pre-cancerous signs.

This document defines the project management procedures and standards the team will follow for the duration of the project for the purpose of achieving both efficient management process and producing a high quality software product.

Contents

1 Introduction

This document defines the project management procedures and standards the team will follow for the duration of the project, for the purpose of achieving both an efficient management process and delivering a high quality software product.

1.1 Project Summary

This purpose of this project is to build a *Cervical Image Database and Patient History System* for storing patients' cervical digital images and related medical histories for diagnostic purpose as well as monitoring for the development of pre-cancerous signs.

1.2 Project Objective

The objectives below are listed in order of importance. All work allocations, scheduling and conflict resolution will aim to realize these objectives.

1. Successfully deliver a final product that meets all core requirements of the client
2. Successfully deliver a final product on time
3. Successfully deliver a final product which is of high quality
4. Successfully deliver all required documents on time with a high level of quality
5. Let all team members gain experience in team-oriented software engineering.
6. Let all team members gain experience in each of the software development phases including requirements analysis, specification development, system design, implementation and testing.
7. Let all team members appreciate the importance of project management, eg planning and time-tracking.
8. Enhance all team members' problem solving and research capabilities.

1.3 Project Scope

The final product will involve:

- A Database for storing patient details and images
- A simple image viewer

- A letter generator

For detailed requirements of the project, please refer to the Software Requirements Specification.

1.4 Project Deliverables

The following artifacts must be delivered throughout this year:

- SPMP
- SRS
- SADD
- TP
- UD

The Client will receive the SRS, the UD and the final working system.

1.5 Project Personnel

- Client
  - Dr Judith Fleming, Obstetrician/Gynaecologist
  - Tel: 5331 3566
  - Mobile: 0418 518 061
  - Fax: 5332 9634
  - Address: 704 Mair Street. Ballarat, Vic, 3350
  - Email: `jhish@netconnect.com.au`
- Supervisor
  - John Fung Cheong FOO
  - Email: `jfcf@students.cs.mu.oz.au`
- 440 Review Committee

Chen-Po SUN	`chenps@students.cs.mu.oz.au`
Joel Beach	`jtbeach@students.cs.mu.oz.au`
Joyce Kwong Hiun SHIA	`joyces@students.cs.mu.oz.au`
Lars YENCKEN	`lljy@students.cs.mu.oz.au`
Venisia TANUWIDJAJA	`venisia@students.cs.mu.oz.au`

- Members of Team V

Alan HUANG	`jhua@students.cs.mu.oz.au`
Chris NOLAN	`csnr@students.cs.mu.oz.au`
Hendy SUMANTO	`hsumanto@students.cs.mu.oz.au`
John YU	`zyu@students.cs.mu.oz.au`
Peng WANG	`peng@students.cs.mu.oz.au`

1.6 Evolution of the SPMP

This document is subject to modification and improvement as the project progresses. Changes will be made, based on:

- Supervisor advice
- 440 review committee reviews
- Any constructive suggestions by team members
- Events that render any procedure no longer applicable

The Project Manager and the SQAM are the only team members to make alterations to the SPMP.

The version number gets incremented by 1.0 after each external review. All development history will be recorded in CVS logs.

1.7 Reference Materials

These reference materials are being used to help writing this SPMP:

- *IEEE standard 1058-1998* for software project management plans
- Hans van Vliet, *Software Engineering Principles and Practice* 2nd edition, John Wiley & Sons Ltd.
- Roger S. Pressman, *Software Engineering A practitioner's approach*, 5th Edition, McGraw-Hill
- Stephen R. Schach, *Software Engineering with JAVA* 4th Edition, McGraw-Hill
- Stephen R. Schach, *Object-Oriented and Classical Software Engineering*, McGraw-Hill

In particular, the IEEE standard 1058-1998 is referred to frequently as this document closely follows the standard and structure set out therein.

1.8 Definitions and Acronyms

A number of acronyms will be used. Their definitions are:

- ASSESS Assessment Management System
- CIDPHS Cervical Image Database and Patient History System
- CVS Concurrent Version System
- IP Implementation Plan
- RAP Review and Audit Plan
- RCS Revision Control System
- RMP Risk Management Plan
- SADD Software Architectural Design Document
- SDK Software Development Kit
- SPMP Software Project Management Plan
- SQA Software Quality Assurance
- SQAM Software Quality Assurance Manager
- SRS Software Requirements Specifications
- TP Test Plan
- TUTOS The Ultimate Team Organization Software (A task allocation tool)
- UD User Documentation
- XML Extensible Markup Language

2 Project Organization

This section specifies the process model for the project, describe the organizational process, identify and describe organizational boundaries and interfaces for the project and define responsibilities for each team member.

2.1 Process Model

The process model will be based on the "Waterfall Model", in which development proceeds linearly through the phases of requirements engineering, design, implementation, testing (validation), integration and deployment. The following is a rough outline of the five phases:

- Requirement Engineering Phase
  - Entry Criteria: Project Allocation
  - Exit Criteria: the SRS approved by the SQAM and passed an external review
  - Product: SRS
- Architectural Design and Detailed Design
  - Entry Criteria: Completion of the SRS
  - Exit Criteria: SADD approved by the SQAM and passed an external review
  - Product: SADD
- Implementation Phase
  - Entry Criteria: Completion of the SADD
  - Exit Criteria: All source code can be compiled as a whole
  - Product: A compiled system
- Test Phase
  - Entry Criteria: Completion of all coding
  - Exit Criteria: All test cases have been run, test reports have been generated and recorded
  - Product: A tested system.
- Deployment
  - Entry Criteria: Completion of testing
  - Exit criteria: Official Client Sign over
  - Product: Deliverable system

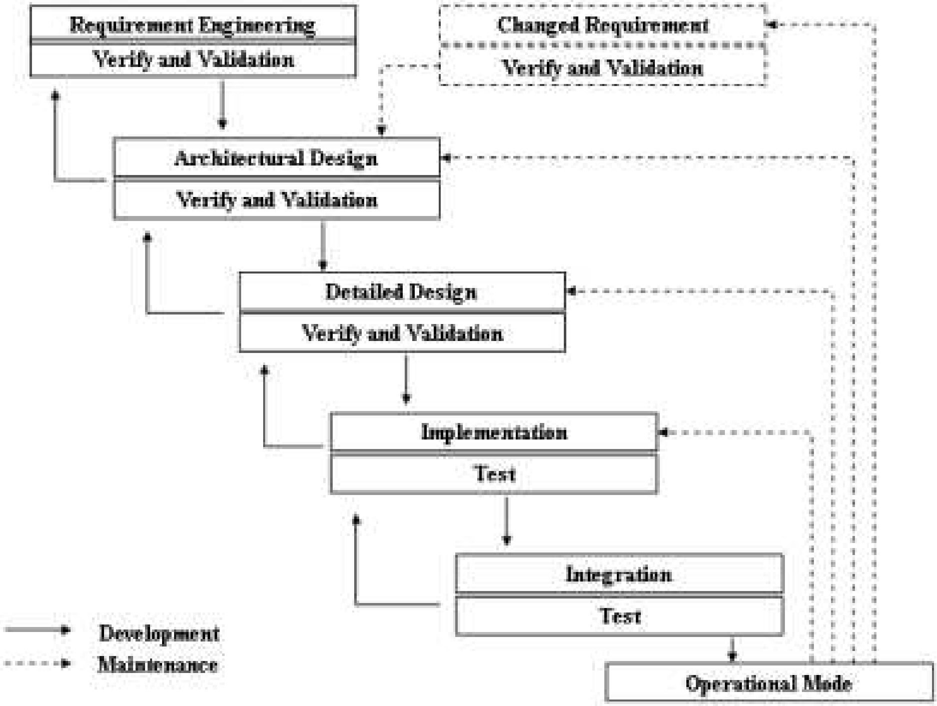


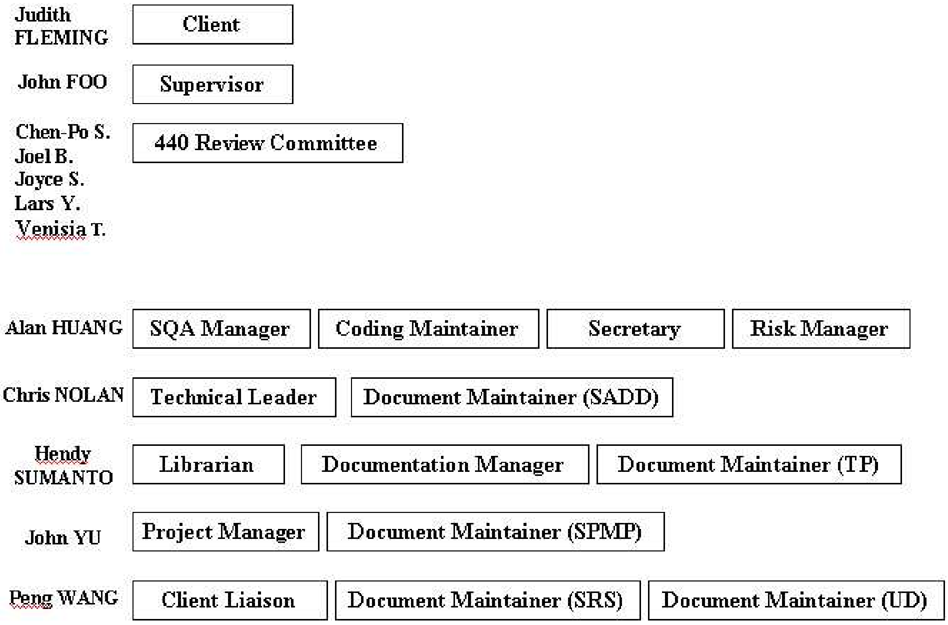
Figure 1: Waterfall Model

However, in order to accommodate our project team's strengths and weaknesses, the following modifications will be made to the traditional "Waterfall Model":

- Some phases may proceed in parallel to make the full use of all team resources.
- It is believed that each team member progresses at a different pace, while some team members are in the process of finishing their assigned tasks, other idle team members can begin tasks that belong to the successive phase.
- Some tasks may proceed early, since their difficulty may consume more time than typical tasks to mitigate the risk of overtime.

2.2 Organizational Structure

This part describes the internal structure of our project team. Every member has different responsibilities within the project. Figure 2 illustrates the different roles involved in the development of the Cervical Image Database system:



Judith FLEMING	Client			
John FOO	Supervisor			
Chen-Po S. Joel B. Joyce S. Lars Y. Venisia T.	440 Review Committee			
Alan HUANG	SQA Manager	Coding Maintainer	Secretary	Risk Manager
Chris NOLAN	Technical Leader	Document Maintainer (SADD)		
Hendy SUMANTO	Librarian	Documentation Manager	Document Maintainer (TP)	
John YU	Project Manager	Document Maintainer (SPMP)		
Peng WANG	Client Liaison	Document Maintainer (SRS)	Document Maintainer (UD)	

Figure 2: Team Organization

2.3 Organizational Boundaries and Interfaces

This part describes the administrative and managerial interfaces between the project and the primary entities involved (stakeholders).

2.3.1 External Interfaces

There are three external parties of the team. They are

- The Client
- The Supervisor
- 440 Review Committee

The Client Liaison will be the main contact intermediary between the team and the client. Outside client meetings, frequent communication between team members and the client will be made via:

- Team's common email: `s340gv@students.cs.mu.oz.au`
- Phone
- Facsimile (archived in design notebook)

The team communicates with the supervisor via:

- Weekly supervisor meetings
- Email

The SQAM will be the main contact intermediary between the team and the 440 review committee. Most communication will be done via email.

2.4 Project Responsibilities

Each role has specific responsibilities within the project.

2.4.1 Supervisor

Supervisor will be responsible for:

- Reviewing weekly team progress
- Attending weekly Supervisor meetings
- Resolving problems the team cannot solve
- Giving general advice with regard to all aspects of the project

2.4.2 Project Manager

The Project Manager will be responsible for:

- Constructing and maintaining the *Project Schedule* table and its corresponding *Gantt Diagram*
- Managing the project at a macro-level using the *Project Schedule* table and the *Gantt Diagram*
- Allocating tasks to team members using `TUTOS` (refer to 3.15)
- Acting as chairperson for weekly supervisor meetings and team meetings
- Coordinating team member activity
- Boosting team morale by providing motivation and encouragement to the team

2.4.3 Technical Leader

The Technical Leader will be responsible for:

- Choosing the most appropriate technical methodologies, tools and strategies to suit the needs of the project

- Communicating with all team members regularly to give expertise via emails when necessary
- Providing technical training to the team when necessary
- Maintaining the design_decision_log

2.4.4 SQAM

The SQAM will be responsible for:

- Composing and maintaining the RAP
- Being in charge of Verification and Validation at the end of each development phase,
- Negotiating review and audit time with 440 review committee
- Performing internal reviews for SADD, TP and UD, and producing "Internal Review Reports"
- Deciding whether to approve a document by sending emails to the Document Maintainer after the "post internal review modification" of a document

2.4.5 Documentation Manager

The Documentation Manager will be responsible for:

- Setting document standards for all documents
- Creating templates for all deliverable documents
- Creating templates for the following log files:
  - status file of all deliverables
  - all meeting agendas
  - all meeting minutes
  - all decision logs
  - research log

2.4.6 Document Maintainer

For each of deliverable document, there will be a maintainer associated with it.

- SPMP: John YU

- SRS: Peng WANG
- SADD: Chris NOLAN
- TP: Hendy SUMANTO
- UD: Peng WANG

Each Document Maintainer will be responsible for:

- Breaking down the document [1] into subfiles as specified in section 3.3.3
- Merging individual subfiles to a complete document
- Recording statuses into the document's status file
- Acting as the sole modifier the document after internal reviews and external reviews
- Acting as the sole modifier of any templates used in document composition
- Any electronic submission involving `ASSESS` as required by the subject submission schedule

2.4.7 Coding Maintainer

The Coding Maintainer will be responsible for:

- Composing and maintaining the IP
- Carrying out code inspections
- Recording implementation decisions into the implementation_decision_log (refer to 3.9)

2.4.8 Testing Manager

The Testing Manager will be responsible for:

- Setting up Ant and JUnit
- Constructing all test cases as stated in the TP
- Recording testing decisions into the testing_decision_log (refer to 3.9)

[1] the one the maintainer is responsible for

2.4.9 Librarian

The Librarian will be responsible for:

- Backing up the team space 3.3.1 on a weekly basis
- Updating the *backup_log* as specified in section 3.11 upon completing a backup
- Restoring files from backup discs in the event of data being lost or accidentally removed
- Creating top level directories and (in some cases) 2nd level directories as starting points for other team members to use
- Setting directory permissions as specified in section 3.3.4 upon creating of a directory
- Overseeing all configuration management processes and issues
- Archiving all team email messages relating to the project

2.4.10 Client Liaison

The Client Liaison will be responsible for:

- Keeping the client informed regarding current project progresses
- Conveying communications from the client to the team
- Negotiating, with the client, the time and venue of a client meeting
- Preparing meeting agendas for client meetings
- Chairing client meetings
- Maintaining the *Progress Report* page of the project website

2.4.11 Risk Manager

The Risk Manager will be responsible for:

- Composing and maintaining the RMP
- Identifying risks associated with the project
- Creating/maintaining risk logs for the team
- Developing mitigation strategies, with a view to preventing risks
- Monitoring the progression of risks
- Maintaining the *Risk Table* on the team website

2.4.12 Secretary

The Secretary will be responsible for:

- Recording meeting minutes during the all meetings
- Committing meeting minutes into `CVS` repository within 48 hours using the standard template for meeting minutes

3 Managerial Process

This section defines the approach the team will use to keep the project development process well managed.

3.1 Management Objectives and Priorities

The main objectives for this project management process, specified in order of importance, are to:

1. Develop a final product with good quality as a core
2. Develop a final product on time
3. Deliver documents and other deliverables on time
4. Promote high overall team productivity
5. Minimize management overhead
6. Maintain harmony within the team

3.2 Assumptions, Dependencies and Constraints

This section states the assumptions on which the project will be based, any external events the project will be dependent upon, and the constraints within which the project is to be conducted.

3.2.1 Assumptions and Dependencies

- Access to the department's machines will be available at all times during the year.
- Any development tools the team chooses to use at the start of the project will still be available during the whole course of the development process.

3.2.2 Constraints

- The length of the project will be 10 months, which is a limited amount of time to build the system.
- The client, who is a medical specialist and runs her own clinic, only has limited availability for requirements elicitation
- The team has limited free time, budget and means of transport to travel to and from Ballarat for client meetings

3.3 Quality Control Plan

This section defines the standards and procedures that will be followed by the team in order to achieve a minimum level of quality in every artifact they produce and in the final product they deliver.

3.3.1 Team Space (/home/se340/s340gv)

Team Space serves as the main storage for all documents, design notebooks and source files the team produces for the duration of the project. The team space will be conform to the follow directory structure:

```
|-- s340gv/
    |-- 440_audits/               - All audit related documents
    |   |-- Managerial/           - Managerial audit related documents
    |   |-- Product/              - Product audit related documents
    |
    |-- 440_reviews/              - All review related documents
    |   |-- SPMP/                 - SPMP review related documents
    |   |-- SRS/                  - SRS review related documents
    |   |-- SADD/                 - SADD review related documents
    |   |-- TP/                   - TP review related documents
    |
    |-- ant/                      - Automated build tool
    |   |-- lib                   - libraries directory.
    |
    |-- cvs/                      - Team repository
    |   |-- Client_Communication/ - Client fax, feedbacks
    |   |-- design_notebook       - Decision logs, miscellaneous files
    |   |
    |   |-- ip/                   - Implementation Plan
    |   |-- IEEEStandard/         - IEEE standards for various documents
    |   |-- meeting/              - All meeting related documents
    |   |   |-- Client_Meeting
    |   |   |-- Supervisor_Meeting
    |   |   |-- Group_Meeting
    |   |
    |   |-- rap                   - Review and Audit Plan
    |   |-- risk/                 - All risk related documents
    |   |-- rmp/                  - Risk Management Plan
    |   |-- Research/             - Individual team member's research
    |   |-- Resource/             - Any other resources used
    |   |-- sdd/                  - Software Design Document
    |   |-- source/               - Source Codes
    |   |-- spmp/                 - Software Project Management Plan
    |   |-- srs/                   - Software requirements Specification
    |   |-- td/                    - Traceability Matrix Document
    |   |-- tp/                    - Testing Plan
    |   |-- testing/               - Testing related documents
    |   |   |-- acceptanceTest
    |   |   |-- bugReport
    |   |   |-- integrationTest
    |   |   |-- releaseEngTest
    |   |   |-- systemTest
    |   |   |-- unitTest
    |   |   |-- usabilityTest
    |   |
    |   |-- ud                     - User Documentation
    |   |-- Template               - Documents templates
    |
    |-- db                         - Back end database of the product
    |
    |-- email                      - Team email archive
    |   |-- mail
    |
    |-- internal_reviews           - Internal review related documents
    |   |-- SPMP/                  - SPMP review related documents
    |   |-- SRS/                   - SRS review related documents
    |   |-- SADD/                  - SADD review related documents
    |   |-- TP/                    - TP review related documents
    |
    |-- workspace                  - Team member's workspace
    |   |-- csnr
    |   |-- hsumanto
    |   |-- jhua
    |   |-- peng
    |   |-- zyu
    |
    |-- www_public                 - Files of the team website
```

3.3.2 Directory Naming Conventions

The acceptable set of characters for naming a directory is:
[A - Z, a - z, 0 - 9, _]

3.3.3 File Naming Conventions

The following file types must adhere to a particular naming convention:

- Meeting agendas and minutes
  They will be named with respect to the date of the meeting, with the format MmmDD.tex, where Mmm is the three-letter abbreviation for the particular month (e.g. Apr) and DD is a two-digit number for day of the month (e.g.12).
  - The acceptable set of values for Mmm is:
    [Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec]
  - The acceptable set of characters for DD is:
    [01, 02. . . , 30, 31]
  - The acceptable set of characters for all other files within the team workspace (not including private workspace files) is:
    [A - Z, a - z, 0 - 9, _]
- Documents:
  If a deliverable document written in LaTeXconsists of multiple sections, then each will be a separate file, e.g. `SRS_section4.tex` will contain section 4 of the SRS.

  The naming convention of all the subfiles:
  - `DOCNAME.tex` (The central file which includes all other subfiles)
  - `DOCNAME_section1.tex`
  - `DOCNAME_section2.tex`
  - `DOCNAME_section3.tex`
  - . . .
  - . . .
  - . . .
  - `DOCNAME_sectionN.tex` (if the document has N sections)
  - `DOCNAME_appendix.tex` (if the document needs an appendix)

3.3.4 Directory Permissions

Permission for the directories in team space must conform to the following standards:

- `www_public` directory will have the permission (`drwxrwxr-x`)
- publicly inaccessible directories have the last three permission bits turned off
- Each team member's private workspace will have the permission (`drwx------`)

3.3.5 File Permissions

Permissions for the files in team space must conform to the following standards:

- publicly accessible files such as index.html inside `www_public` will have the public read permission bit turned on
- publicly inaccessible files will have the last three permission bits turned off

3.3.6 Document Standards

Documents in the team space must conform to the following formats:

- All deliverable documents will be formatted using LaTeXand will be submitted in `PDF` format.
- Documents to be externally reviewed by 440 review committee will also be in `PDF` format. directory)
- Meeting agendas and minutes will be compiled in LaTeXformat.

3.3.7 Deliverable Standards

This section defines the standards of various documents. It will also serve as criteria against which review on deliverables will be conducted.

- SPMP

  The SPMP should meet the following standards:

  - The contents are accurate.
  - The document covers everything it should.
  - The document does not contradict itself, in terms of both style and meaning.
  - The document is concise and easy to understand.
  - All procedures defined in the document are verifiable.

- SRS

  The main objective of the SRS is to ensure that the listed requirements provide a clear and detailed description of the proposed system. The SRS should meet the following requirements:

  - All requirements stated are correct.
  - All requirements stated are clear.
  - All required functionalities are completely included.

  - The document does not contradict itself, in terms of both style and meaning.
  - The document is concise.
  - The requirements can be verified.
  - The requirements are ranked according to their importance.

- SADD

  The main objective of the SADD is to ensure the clarity and modularity of the proposed architecture, as well as its ability to fulfill the requirements. The SADD should meet the following criteria:

  - The architecture is implementable.
  - The architecture covers all use cases stated in the SRS.
  - The roles of each module/component in the architecture are clear.
  - Each module can be traced to one or more requirements in the SRS.

- TP

  The main objective of the TP document is to ensure the adequacy and completeness of testing procedures and ensuring all sections of the implementation are covered by the TP. The TP should conform to the following standards:

  - Testing methodologies and procedures are clearly defined.
  - Code inspection methodologies and procedures are clearly defined.
  - Bug reporting procedures are clearly defined.

- UD

  The main objective of the UD is to ensure the documentation specifies how the user needs to interact with the system to provide fulfillment of functional requirements set out in the SRS. Specifically:

  - The document should conform to the requirements stated in the SRS.
  - The document is comprehensive.

3.4 Risk Management Plan

Detailed risk management procedures will be defined in the RMP.

3.5 Monitoring and Controlling Mechanisms

3.5.1 Review Procedures

Detailed review procedures will be defined in the RAP.

3.5.2 Audit Procedures

Detailed audit procedures will be defined in the RAP.

3.6 Traceability Matrix

The *Traceability Matrix* will be a document serves to verify the correctness of:

- The modules in the SADD against the functional requirements in the SRS
- The system test cases against the SRS
- The integration test cases against the SADD

Their relations are illustrated below.

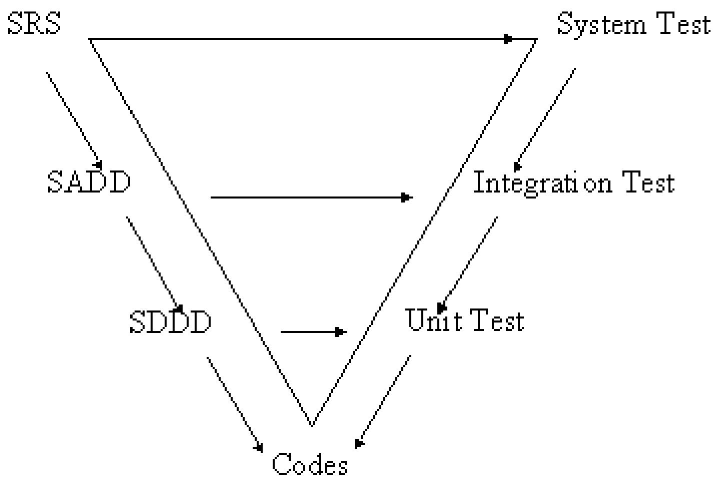


Figure 3: The V Model

3.7 Meeting Types

3.7.1 Supervisor Meeting

- Present: Supervisor, All team members
- Time: Every Friday 6:30pm - 7:30pm
- Venue: ICT Building Room 209
- Meeting Scope:
  - Report progresses of the project to the Supervisor
  - Receive feedback from the Supervisor
  - Discussion of any problem that can't be solved by the team
- Chair Person: Project Manager

3.7.2 Team Meeting

- Present: All team members
- Time: Every Thursday 1:00pm - 2:00pm
- Venue: ICT Building Room UG 13
- Meeting Scope:
  - Discuss progress of the project
  - Discuss common problems
  - Discuss critical issues and tasks
  - Discuss major changes related to the project or team organization
  - 5-10 mins for team members to report risks to the risk manager (if any new risks exist, updating on the status of old risks where appropriate)
- Chair Person: Project Manager

3.7.3 Client Meeting

- Present: The client, the Client Liaison, other team members
- Time: Not Fixed, depends on the availability of client
- Venue: Not Fixed, depends on the availability of client
- Meeting Scope:
  - Elicit requirements from the client

  - Report Progresses of the project to the client
  - Demonstrate prototypes to the client
  - Discuss changes related to client requirements

- Chair Person: Client Liaison

3.8 Meeting Procedures

3.8.1 Before Meetings

Group and Supervisor Meeting

- The chairperson of the meeting will be responsible for writing up the agenda and checking it into the repository within 24 hours before the meeting.
- After the chairperson checks in the meeting agenda, he must record the committing date in the `tracking_meeting.txt` file inside `cvs/meeting`
- The meeting quorum will be 4 members
- Should the quorum not be met, the meeting will be cancelled and rescheduled for a different time and, if needed, different venue
- 5-10 minutes of the meeting agenda will always be allocated to the reporting of new risks and the status of existing risks to the Risk Manager
- If the chairperson decides to cancel the meeting, he must inform the team prior to the meeting

Client Meeting

- The chairperson will be responsible for writing up the agenda and checking the file into the repository within 24 hours before the meeting
- After checking in the meeting agenda, he must record the committing date in the `tracking_meeting.txt` file inside `cvs/meeting`
- The chairperson must send the latest version of the agenda to the Client via email, fax or via Ms. Kathleen Keogh.

3.8.2 During Meetings

Group and Supervisor Meeting

- The chair person of the meeting will conduct the meeting according to the agenda

- The Secretary records all the important issues during the discussion in the meeting minutes

Client Meeting

- The chairperson will preside over the meeting according to the agenda
- The Secretary records all the important issues during the meeting, if necessary, tools like tape recorders will be used

3.8.3 After Meetings

- The Secretary will commit all meeting records within 48 hours.
- The Secretary will record the date of check in the `tracking_meeting.txt` file inside `cvs/meeting`.

3.8.4 Meeting Related Files and Directory Structure

`cvs/meeting` will be used to store all the meeting agendas and meeting records so that members can check them out for reference. Templates for agendas and meeting minutes will be kept in `cvs/Template`. (refer to section 3.3.1)

3.9 Development Decision

All relevant technical methodologies, tools and strategies will be recorded into three separate log files

- `design_decision_log`, recording design related tradeoffs and decisions. [2]
- `implementation_decision_log`, recording tradeoffs and decisions of choose certain coding strategies, tools and methodologies.
- `testing_decision_log`, recording tradeoffs and decisions of choose certain testing strategies, tools and methodologies.

These log file must be kept in the `cvs/design_notebook` directory.
Each entry of a decision log must specify the date when a decision was made and why that decision was made. These log files will be created using the same template `design_decision_log_template` can be found inside in `cvs/Template` directory.

[2] The SADD maintainer will be responsible for recording decisions into this log

3.10 Research

All team members must upload their research files (downloaded from various sources) to `cvs/Research` directory for future reference.

A file called `research_log` will be placed into the "cvs/Research" directory. Each time a team member uploads a file, he must update the `research_log` file. The template of this log can be found inside `cvs/Template`.

3.11 Backup Procedure

The following are the procedures the team space backup.

- The whole team space `/home/se340/s340gv` must be backed up weekly
- All backup files will be stored on CDs and each disk must contain only one week's backup
- Each backup CD will contain 2 files: s340gv.zip and a README file specifying the date of the backup
- The `backup_log` will be updated after the successful completion of each backup
- In the case of the team's backups failing, the team will request access to the backups made by the department

3.12 Email Policy

- The archiving of emails will be done by using `Procmail`. `Procmail` will be used to sort the incoming mail into separate files
- To view the email archive, every member of the team needs to obtain a copy of .pinerc, which will be located at `cvs/Resource/Procmail/.pinerc`.
- All email archives can be viewed using Pine.

3.12.1 Team Email Archiving

In order for team emails to be archived, all emails from team members to the supervisor, 440 review committees, or to other team members must be contain the address s340gv@students.cs.mu.oz.au in at least one of the To, Cc fields.

All team emails will be archived monthly. The archives will be located at: email/mail/MONTH.mail (MONTH can be March, April, etc).

3.12.2 Supervisor Email Archiving

Incoming emails from the supervisor and outgoing emails to the supervisor will be archived in: email/mail/Supervisor.mail

3.13 Website Procedure

The team web site is `http://www.cs.mu.oz.au/SE-projects/s340gv`

3.13.1 Creation of the web site

- The Librarian creates the `www_public` directory inside the team space
- The Librarian creates the `RCS` directory within `www_public` 3.3.4 and section 3.3.5)

3.13.2 Maintenance of the web site

Only the Project Manager, Client Liaison and the Risk Manager will maintain the website.

- Client Liaison The *Project Progress* page of the team's web site will be updated twice a month to show the progress of the project to the client.
- Project Manager
  - The *Project Schedule* table will be updated by the Project Manager.
  - SPMP, SRS, SADD, TP, RAP, RMP, IP, UD, Traceability Matrix Document and the *detailed design* be put on the web and be updated by the Project Manager. The update will be done with the help of two shell scrips exist in `cvs/Resource`.
- Risk Manager The *Risk Table* will be updated by the Risk Manager.

3.14 Status files for deliverables

The status of a document must include the following items:

- "Pre-Compose Research"
- "Initial Development"
- "Internal Review"
- "Post Internal Review Modification"

- “First External Review”
- “Post First External Review Modification”
- “Second External Review” (optional)
- “Post Second External Review Modification” (optional)
- “Others” (specifying others)
- The actual start and finish date of each status
- Whether it has been submitted via ASSESS (if submitted, the submission date)
- Whether it has been approved by the SQAM (if approved, the approved date)
- Whether it passed the 440 external review (if passed, the passed date)
- Whether it has been delivered to the marker (if delivered, the delivery date)

For the SRS, there will be another status which is “Client Signed Off”, the client sign off date must be clearly stated.

For the UD, instead of having

- “First External Review”
- “Post First External Review Modification”
- “Second External Review” (optional)
- “Post Second External Review Modification” (optional)

the following extra status will be included

- “Product Audit”
- “Post Product Audit Modification”

The status file will be named as `DOC_NAME.status` where `DOC_NAME` is the name of the document. They will be kept in the same directory as the document.

3.15 Task Allocation

The team uses `TUTOS` as their task allocation and task tracking system. The project manager has the responsibility to assign tasks to individual team members, and each team member will periodically record the progresses on every tasks assigned.

3.15.1 Procedures for the Project Manager

The procedures that the Project Manager will follow (when using `TUTOS`) are:

- The Project Manager will be the only person who has the permission to create a task for the members of the team.

  In order to create a task, the Project Manager must specify the following:

  - Name of the task
  - State (must set to *pre* state when first create the task)
  - To-do items (can be one or more of them)
  - The person to whom the task will be assigned
  - Expected start date of the task
  - Expected end date of the task
  - Work Volume (in hours)

- The Project Manager can add/delete a task into/from *Watch List* if necessary.

3.15.2 Procedures for each team member

- Change state from *Pre* to *Ongoing* [3]

  Notice when the Project Manager allocates a task for member, he always set the state as “Pre”. When the member starts doing the allocated task, he must change the state from “Pre” to “Ongoing” indicating they have actually started doing the allocated task.

- Record progresses for each *To-Do* item

  For each assigned task, if there are usually multiple *To-Do* items, each member will record the progress (in one or two lines) for each To-Do item, if he has made any progress for that *To-Do* item.

- Record *Volumes Done*

  For each assigned task, record the *volume done* [4], if he has made any progress. e.g. if the task is allocated on 10th of August expecting the team member to finish the task in 10 hours, initially the *volume done* is 0 hours. If the team member has done 3 hours on 11th of August, the team member should change *volume done* from 0 to 3 on 11th of

[3] In TUTOS, Pre, Ongoing, Finish are three different states of a task.

[4] volume done is a text field into which a user can input an number

August. If the team member has done another 4 hours on 12th of August, the team member should change *volume done* from 3 to 7 on 12th of August.

- Team members
  - cannot assign tasks to other team members
  - cannot remove any task from any watchlist [5]

[5] watchlist is a feature of TUTOS

4 Technical Process

This section gives brief explanation about what methods, tools and techniques will be used during the project.

4.1 Methodologies

4.1.1 Requirement Methodology

The team adopts the following methodologies for the requirement engineering process:

- Requirement elicitation
  During requirement elicitation stage, the team conducts interviews and meetings with the client to gather requirements as completely as possible. Instead of letting the client talk about her requirements for the entire duration of the meeting, the team will prepare questions, as well as possible options, beforehand as a means of gathering information about the aspects of the system that the team expect might be of interest and/or use to the client.

- Requirement analysis
  After adequate requirements have been gathered, team members will analyze and classify the requirements gathered for their feasibility. As another important procedure, various programming language candidates will be considered for implementing the system. Final choice(s) will be made and the reason for choosing a particular candidate will be recorded in the team's design notebook in the repository.

- Requirement specification
  During the requirement specification phase, system prototypes will be produced (preferably using Visual Basic and paper prototyping) and put into the design notebook in the repository. The main product of this requirement engineering phase, the SRS will be produced, in which use case diagrams will also be drawn as a brief but important explanation of the final functionalities the system will have.

4.1.2 Design Methodology

Design Methodologies will be documented in the SADD.

4.2 Tools

4.2.1 Tools for Documentation

- Converting LaTeXto html: `LaTeX2html`

- Document preparation: LaTeX, MS Word
- Diagram Drawing tool: MS Visio, SmartDraw
- Text editors: vi, vim, gvim
- Viewing LaTeXdvi files: xdvi
- Viewing PS files: ghostview
- Viewing PDF files: Acrobat Reader Snagit

4.2.2 Tools for Coding

- Application programming interface: JDBC
- Database: MySQL
- Software Development Kit: JBuilder, NetBeans
- Virtual Machine: J2SE 1.4.2 SDK

4.2.3 Tools for Testing

- Java unit testing tool: JUnit
- XML based building tool: Ant

4.2.4 Tools for Configuration Management

- Version control of documents and source codes: CVS
- Version control of web pages: RCS

4.2.5 Tools for Planning and Management

- MS Project
- TUTOS

4.2.6 Tools for Communication

- Email client: Pine
- Email archiving tool: Procmail

4.2.7 Tools for other activities

- Presentation preparation: MS PowerPoint, Snagit
- Web browsers: MS Internet Explorer, Netscape

5 Project Schedules, Dependencies

5.1 Work Packages and Project Schedules

All main activities throughout this project along with their duration and expected finishing dates will be published on the team's web site as the *Project Schedule* table.

The *Project Schedule* table will record the following items:

- ID
- Task Name
- Duration
- Start
- Finish

5.2 Dependencies

5.2.1 Requirements Elicitation

Dependent on the client's availability and characterization of project specific terminology and process. Additionally, requirements may arise from team discussion of the project's state or investigation of current process and practice at the client's place of business.

5.2.2 Project Specific Terminology and Process

Resources volunteered by the client will be the main starting point for this work package. During the process of requirements elicitation, new terms and procedures may come to the attention of the team, warranting client consultation and/or independent research.

5.2.3 Translation of Requirements in SRS

This process will begin once sufficient requirements have been elicited as to establish the core functional requirements of the product. The document will be considered to be a working document until sign off.

5.2.4 Sign off of SRS

The events dictating sign off (or attempts to sign off) on the SRS are one or both of:

- All functionality the client seeks has been completely and clearly defined in the SRS

- Addition of further requirements may jeopardize the viability or scheduling of the project.

5.2.5 Translation of SRS into SADD

This event requires that the SRS be sufficiently developed as to ensure that the basic design of the software will be well enough defined to allow further feature addition within reasonably defined constraints. It is likely that both documents will be changed fairly frequently until the SRS is finalized and signed off. The SADD itself can be finalized after this point.

5.2.6 Formulation of Testing Procedures

Completion of this task will be truly dependent on both the completion of the SRS and the SADD. As various levels of testing will be required (eg: low-level unit testing and high-level usability and system testing), formulation may begin during the development of the SRS.

5.2.7 Implementation

Implementation requires complete finalization of the SADD and SRS. Allocation of resources such as development hardware, specialized software, access to existing solutions to be accessed by the product and appropriate same data will also be required.

5.2.8 Testing

Testing in this case will need to extend over all deliverables of the project. For instance, the SADD will need to be tested against the SRS to ensure no shortfalls in design will hinder the delivery of any specified requirements. Testing will be segmented and applied to each stage of the project. Testing will start as implementation begins in the form of unit testing, followed by integration testing when a baseline is approached, followed by usability testing when a baseline is reached and finalized.