Easy WinWin

Life Cycle Plan (LCP)

Team X

Jim Black
Joe William
David Jain

Table of Contents

Version Control

Date	Author	Changes	Version
09/17/1999	Jim Black	Initial draft	0.1
10/01/1999	Joe William	Revisions	0.2
10/02/1999	Joe William	Update life cycle plan to reflect spiral cycles	0.3
10/15/1999	Jim Black	Update assumptions, plans	0.4
11/18/1999	Jim Black	Incorporate LCO ARB reviews	1.1
11/20/1999	Jim Black	Approach Section reworked	1.2

List of Figures

1 Introduction

1.1 Purpose of the Life Cycle Plan

The Life Cycle Plan document is written to achieve the following objectives for Easy WinWin project.

1. Guide the effort of cs577 project, Easy WinWin
2. Optimize the use of personnel and material resources throughout the life cycle of this project.
3. Anticipate and mitigate risks likely to arise during the course of project development.
4. Form the basis for the controlling of the project's progress in achieving this project.

1.2 Assumptions

The Life Cycle Plan makes the following assumptions:

1. Stability of the project requirements, advance notification of requirement changes should such changes occur, and availability of the customer to clarify such changes.
2. Continuity of funding for the resources required by the project such as hardware, software, networking and system administration, and licenses:
    (a) Availability of the Pentium server beginning at the construction phase
    (b) Availability of Window NT 4.0 server as our operating system
    (c) Availability of hardware and network resources specified in SSRD 3.1.
    (d) Availability of an lab administrator to maintain the NT server
    (e) Availability of system administrator to maintain GroupSystems software
3. Stability of expectations in the Form and the content of deliverables not only from the project perspective, but also from the academic administration perspective of cs577.
4. Stability of the customers' relationship with the COTS vendor, GroupSystems software to obtain the software licenses and extend the capabilities of the GroupSystems software API.

If one or more of these assumptions is no longer satisfied, then the requirements (System Requirements or Project Requirements) may need to be re-negotiated; or this Life Cycle Plan may need to be re-evaluated.

1.3 References

1. Operational Concept Description 2.0
2. System and Software Requirements Definition 2.0
3. System and Software Architecture Description 2.0
4. Feasibility Rationale Description 2.0
5. Software Project Management – A Unified Framework, Walker Royce, 1998.
6. "Anchoring the Software Process", Barry Boehm, 1996.
7. Rapid Development: Taming Wild Software Schedules , Steve McConnell. Microsoft Press, 1996.
8. Easy WinWin Project Description.
9. Client Meeting Minutes, 09/16/1999, 09/23/1999, 09/29/1999, 10/8/1999, 10/15/1999, 11/1/1999, 11/11/1999, 11/19/1999, 11/26/1999, 12/10/1999

2 Milestones and Products

This section describes the software development process chosen to implement the Easy WinWin system and the rationale behind selecting such process. The process consists of identifying responsibilities, achieving specific milestones (specifying what product will be delivered when), and satisfying stakeholders.

2.1 Overall Life Cycle Strategy

In order to bring the project from inception to completion, a strategy is adopted to ensure that all essential requirements are met at every stage of the project's life cycle. For Easy WinWin project, we have adopted the WinWin Spiral Model and design to schedule for software development along with anchor points such as Life Cycle Objective (LCO), Life Cycle Architecture (LCA), Rebaselined Life Cycle Architecture (RLCA) and Initial Operational Capability (IOC).

The WinWin Spiral Model is described by Dr. Barry Boehm and it is discussed in detail in class CSCI 577a, therefore it is well understood by the project team. The Staged Delivery approach will also be used to prioritize the delivery of the various system components. The most critical components of the system are to be delivered first, while the least the last.

The Staged Delivery approach mitigates risks posed when the schedule is an independent variable. We have divided the life cycle into cycles that encompasses a particular aspect of the life cycle of the project. In addition, a prototype will be developed to be shown in LCO and LCA architectural board meetings and reviews to demonstrate the basic interface of the system and to address IKIWISI (I know it when I see it) issues.

The following is the breakdown of process models used in each stage of the project's life cycle.

- **Engineering Stage**
  - Inception Phase (Life Cycle Objectives): one prototype, one WinWin Spiral cycle, both completed by LCO Architectural Review Board (ARB) review.
  - Elaboration Phase (Life Cycle Architecture): one prototype, one WinWin Spiral cycle, demonstration of some riskiest areas of the system, both completed by LCA ARB, a short WinWin Spiral cycle to re-baseline LCA artifacts, completed by RLCA ARB
- **Production Stage**
  - Construction Phase: two WinWin Spiral cycles (anticipated) to implement the system, completed by a Initial Operational Capability demonstration and Transition Readiness Review where each cycle is synchronized with the iteration of the implementation
  - Transition Phase: one short WinWin Spiral cycle (anticipated) to be completed by Release Readiness Review.
- **Evolution stage**
  - A series of releases, each with its appropriate choice of stages and phases, completed by a Release Readiness Review, USC CSE responsibility.

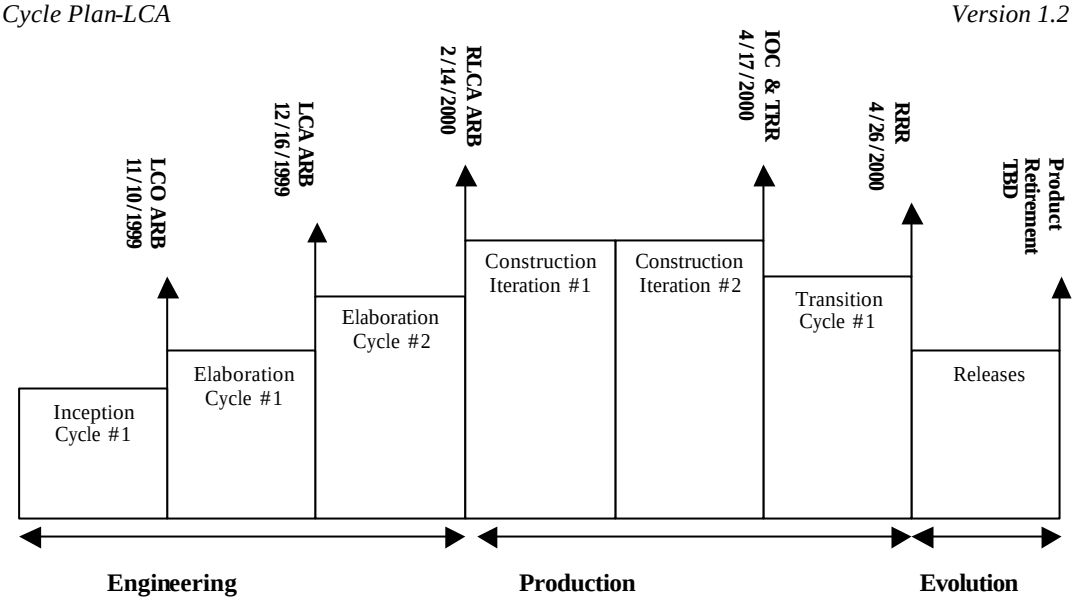

Figure 1 Milestones in Different Stages of Life Cycle Plan

2.2 Phase Milestones and Schedules

The following are the milestones to be achieved:

• LCO (Life Cycle Objectives)-ARB	Wed 11/10/1999
• LCA (Life Cycle Architecture)-ARB	Tue 12/16/1999
• RLCA (Rebaselined Life Cycle Architecture) -ARB	Mon 2/14/2000
• IOC & TRR (Transition Readiness Review)	Mon 4/17/2000 (anticipated)
• RRR (Release Readiness Review)	Wed 4/26/2000 (anticipated)
• Easy WinWin 1.0 Released	TBD

The schedule of the project is delineated below, however, our planning might deviate from the schedule we have prepared based on the meeting that is scheduled with the client and the COTS Vendor. If they refuse to agree to extend some capabilities of the API or do not allow us to do so, there might be a switch to other COTS or architectural alternatives which might eventually effect some of the less significant dates discussed below.

2.2.1 Engineering Stage

- **9/13/1999 Team Formation**
  - Three students (Jim Black, Joe William, David Jain) with a cohesive set of skills and interests are teamed together
  - The process is achieved in a self-organizing and voluntary basis
- **9/15/1999 Project Assigned**
  - Our team has submitted their preferences to choose a project from one of the pre-conceived list of the instructor
  - The collective strength and weakness of the team is matched to the most appropriate project by the instructor
  - We are assigned Easy WinWin project for cs577 class
- **9/16/1999 Client Meeting**
  - For further inFormation a meeting is organized with the client
  - Set up shared vision
- **9/17/1999 Beginning of Inception Spiral Cycle #1**

  - The objectives of the product, especially overall functionality, are identified
  - Schedule and cost constraints are identified by the client
  - Start working on LCO package elements
- **9/23/1999 Client Meeting**
  - For further information a meeting is organized with the client
  - Update shared vision
- **9/29/1999 Client Meeting**
  - A meeting is organized to prepare for the proposed system description
  - Informal training by Paul Grunbacher about GroupSystems software
- **10/1/1999 OCD Draft is Due**
  - The first version of the Operation Conception Description is done
  - It is to be published in the web for other users and team member to review
- **10/3/1999 Alternatives Evaluated**
  - The other alternatives such as other COTS that supports collaboration or designing from scratch are evaluated with respect to the schedule constraints and product objectives.
- **10/5/1999 Developing risk resolution strategy**
  - The top most significant risks is identified, COTS mismatch
  - Plan for initial prototype to mitigate these risks
- **10/8/1999 Client Meeting**
  - Prototype due, a collection of screen shots that provide the look and feel of Easy WinWin is provided. No actual functional content is necessary at this stage
  - The basic interfaces and ergonomics of the system is evaluated
  - A meeting is organized for the prototype feedback from client to resolve IKIWISI issues
  - Update shared vision
  - Plan for next level of prototype
- **10/15/1999 Client Meeting and WinWin Negotiation**
  - Project Stakeholders are identified
  - Win conditions for all the stakeholders are identified
  - Negotiations among stakeholders are done so that all outstanding Win conditions are met by their respective resolutions.
- **10/22/1999 LCO Draft on the web**
  - The final version of the OCD is due to be published on the web
  - Inputs from the first draft are to be incorporated
  - Formalized comments are collected for the OCD and SSAD
  - Inspection Reports are due
  - First version of System and Software Architecture Description is due
  - Cocomo is run to produce the first estimation of the overall cost of the project
- **11/1/1999 Client Meeting**
  - A meeting is organized to clarify some issues
- **11/5/1999 LCO Package is due**
  - The final LCO version of OCD, SSAD, SSRD, FRD, LCP are due.
  - Screenshots from the prototype are generated
  - First complete characterization of the entire project

- **<u>11/10/1999 INCEPTION PHASE CYCLE #1 CONCLUSION: LCO ARB</u>**
  - Developers are to meet with customers to perform an architecture review of the entire LCO package
  - The members of the team will meet with CSE stakeholders from USC
  - The review material is sent to all review members one week in advance
  - The generated suggestions will be incorporated for the LCA drafts
- **11/11/1999 Client Meeting**
  - A meeting is organized to get more detailed information about requirements
  - Review shared vision

- **11/17/1999 Beginning of Elaboration Spiral Cycle #1**
  - The objectives for the architecture of the product are identified
  - Start working on LCA package elements
- **11/19/1999 Client Meeting**
  - A meeting is organized for more detailed information about the proposed system
- **11/22/1999 Developing risk resolution strategy**
  - The top most significant risks are identified
  - Develop a strategy to mitigate these risks, such as overloading the rest of the team members for some time
- **11/26/1999 Client Meeting**
  - A meeting is organized for reviewing LCA draft
  - Update shared vision
- **11/29/1999 Architectural Analysis**
  - The alternative architectural elements are discussed among team members
  - Update risk resolution strategy based on new findings of architectural study
- **12/06/1999 LCA Draft is due**
  - The first LCA version of OCD, SSAD, SSRD, FRD, LCP are due
  - These documents are to be based on that of LCO
- **12/10/1999 Client Meeting**
  - Prototype due, A new prototype is presented to the client to resolve IKIWISI issues
  - Plan for next level prototype
- **12/13/1999 LCA Package due**
  - The final LCA version of OCD, SSAD, SSRD, FRD, LCP are due.
  - Printed documents along with prototype software are submitted
  - More detailed characterization of the entire project

- **12/16/1999 ELABORATION PHASE CYCLE#1 CONCLUSION: LCA ARB**
  - Developers are to meet with customers to perform an architecture review of the entire LCO package
  - The members of the team will meet with the CSE stakeholders from USC
  - The review material is sent to all review members one week in advance
  - Review and update any material based on LCA ARB comments before proceeding any further
- **12/23/1999 Client Meeting**
  - Any remaining issues are discussed
- **01/24/2000 New Team Formation**
  - Three students from cs577b with a cohesive set of skills and interests are teamed together, whether they will be the same students or not are TBD at this point
- **01/30/2000 Client Meeting**
  - Customers and team members review the LCA package in order to approve the specifications and features as a baseline the development of the Easy WinWin project
  - This provides stability for the requirement and features of the project going forward
  - Customers meet with the new team
  - If the team members are changed, then the new team will have to review both LCO and LCA package and organize a new meeting with the client
  - Update shared vision
- **02/01/2000 Beginning of Elaboration Spiral Cycle #2**
  - The objectives for the architecture of the product are identified
  - Start working on RLCA package elements
  - Start also to think about some specific construction set elements, like test data, test plan
- **02/02/2000 Meeting with COTS Vendor (GroupSystems Corporation)**
  - Discuss all critical issues, like licensing, extending capabilities by GroupSystems API

- **02/03/2000 Update risk resolution strategy**
  - Update risk resolution strategy based on the information obtained from COTS vendor and client
- **02/05/2000 Client Meeting**
  - Inform customer about more architectural specific issues
- **02/06/2000 RLCA Draft due**
  - The draft RLCA version of OCD, SSAD, SSRD, FRD, LCP are due.
- **02/08/2000 RLCA package due**
  - Inform customer about more architectural specific issues
  - The final RLCA version of OCD, SSAD, SSRD, FRD, LCP are due.
  - Printed documents along with prototype software are submitted
- **02/14/2000 ELABORATION PHASE CYCLE#2 CONCLUSION: RLCA ARB**
  - Each team member will be assigned their responsibility toward IOC

2.2.2 Production Stage

All the dates in this section are anticipated according to last year's schedule. The exact dates of weekly client meetings will also be scheduled later.

- **02/15/2000 Setup construction environment**
  - Procure the availability of development machine
  - Procure the necessary COTS software packages including all the necessary developmental licenses to GroupSystems software
  - All necessary software including the operating system are installed and operational, Secure the development and administrative access privileges to the NT Server (our development machine)
  - The necessary cost of software acquisition is to be secured by CSE
- **02/17/2000 Client Meeting**
  - Discuss production stage issues
- **02/18/2000 Beginning of Construction Iteration #1**
  - Review and update any material based on RLCA ARB comments before proceeding any further
  - Design to schedule, therefore start tailoring the GroupSystems for *very high priority* requirements: negotiation (RQ-1), storage of artifacts (RQ-2)
  - Plan for operational prototype
- **02/21/2000 Construction Set, IOC #1 documents package**
  - Work on IOC#1 documents
  - Start to think about Transition Set package
- **02/24/2000 Client Meeting**
  - Discuss production stage issues
- **03/19/2000 Construction Iteration #1 Documents (IOC#1) Due**
- **03/20/2000 CONSTRUCTION PHASE Iteration #1 CONCLUSION**
- **03/21/2000 Client Meeting**
  - Discuss iteration 1 prototype
- **03/22/2000 Beginning of Construction Iteration #2**
  - Incorporate feedback from the prototype
  - Decide on the rest of high, medium and low level priorities
  - Start tailoring the GroupSystems for *rest of high, medium and low priority* requirements: sharing (RQ-3) and organization of artifacts (RQ-4)
  - Plan for next level prototype
- **03/23/2000 Construction Set, IOC #2 documents package**
  - Work on IOC#2 documents, Transition Set Package
  - Start to think about Support set package
- **03/28/2000 Client Meeting**
  - Provide an outline for the rest of the life cycle plan

- **04/05/2000 Core Capability Demo**
  - Demonstrate features of the team's system to the client
  - Turn in core capability demo report
- **04/11/2000 Client Meeting**
  - Discuss Evolution stage related issues
- **04/16/2000 Construction Iteration #2 Documents (IOC#2) and Transition Set Due**
- **04/17/2000 CONSTRUCTION PHASE Iteration #2 CONCLUSION: TRR**
- **04/18/2000 Beginning of Transition Cycle #1**
  - Transition the system to the CSE
  - Organize training sessions
  - Update the support plan
- **04/28/2000 Client Meeting**
  - Discuss any evolutionary requirements
- **05/02/2000 TRANSITION PHASE Cycle #1 CONCLUSION: RRR**
  - Support Set package and Data Collection Set due as well

2.2.3 Evolution Stage

- **TBD Easy WinWin Release 1.0**
  - More Training, Demos
  - Update plan for evolutionary requirements

- **TBD EVOLUTION PHASE CONCLUSION: Product Retirement**
  - This stage is anticipated to be around five years

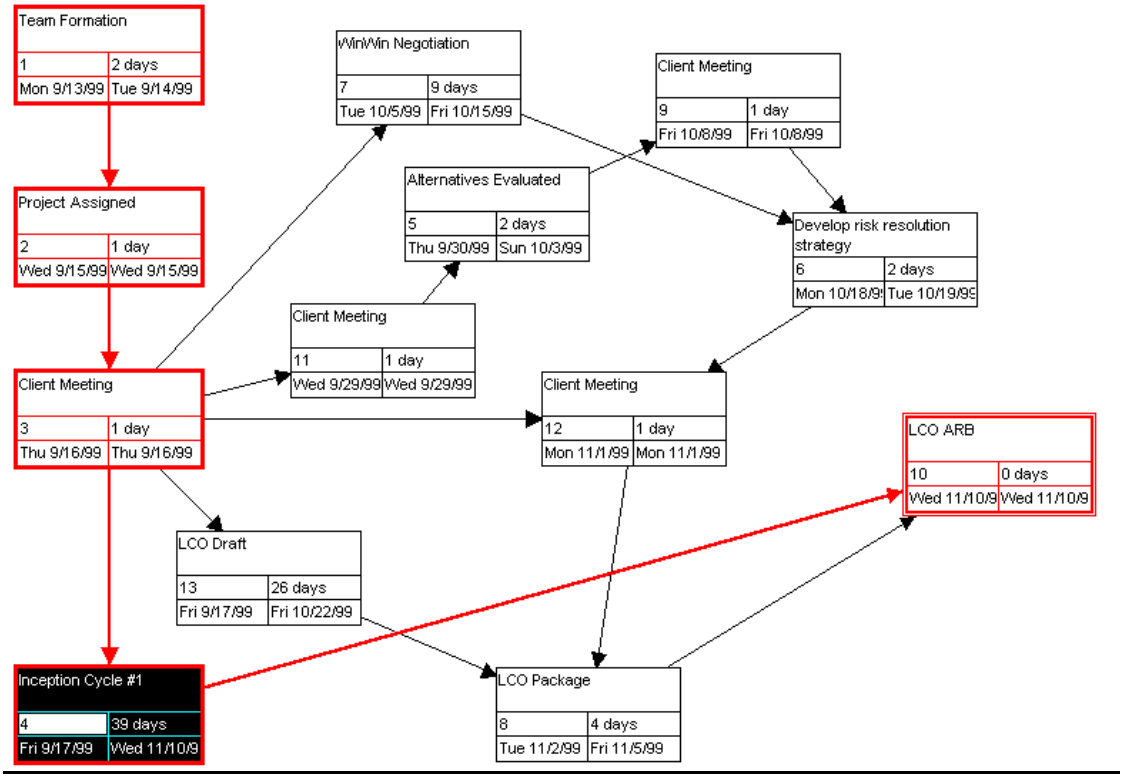

Figure 2 LCO PERT Chart

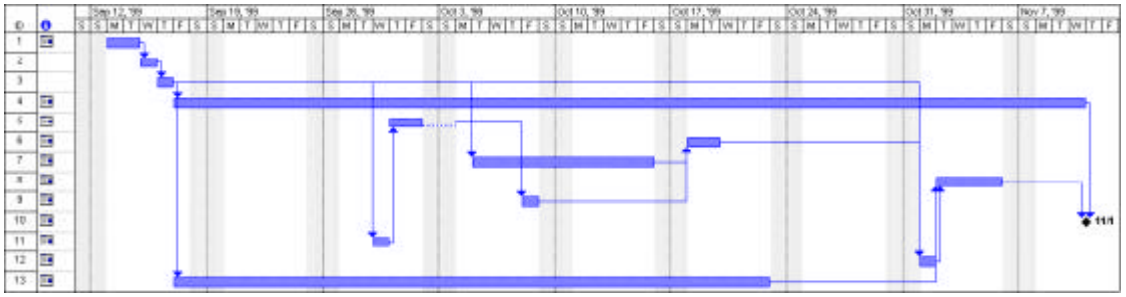

Figure 3 LCO Gantt Chart

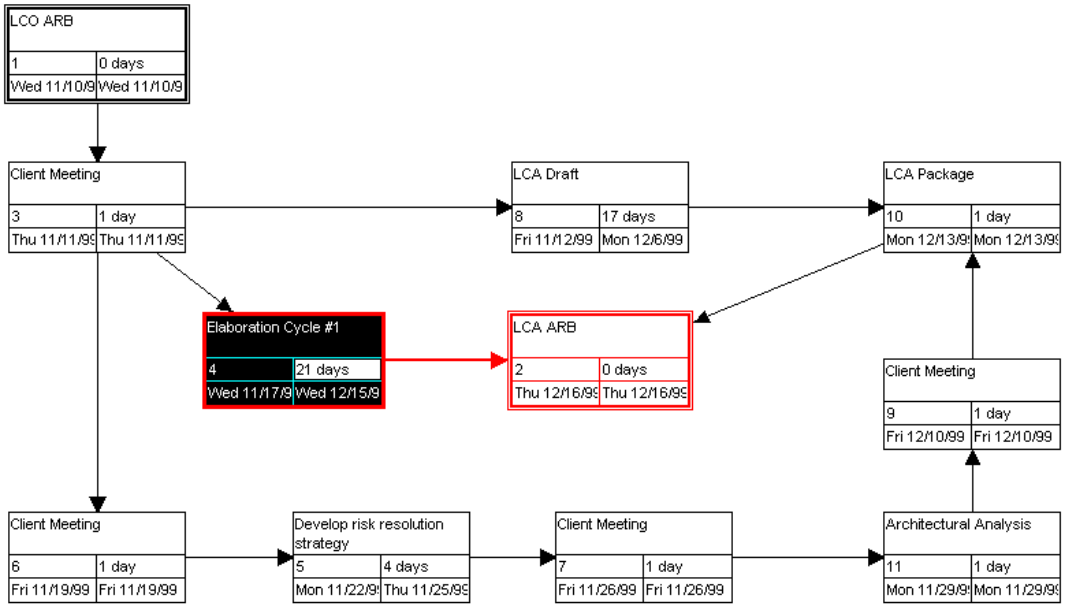

Figure 4 LCO-LCA PERT Chart

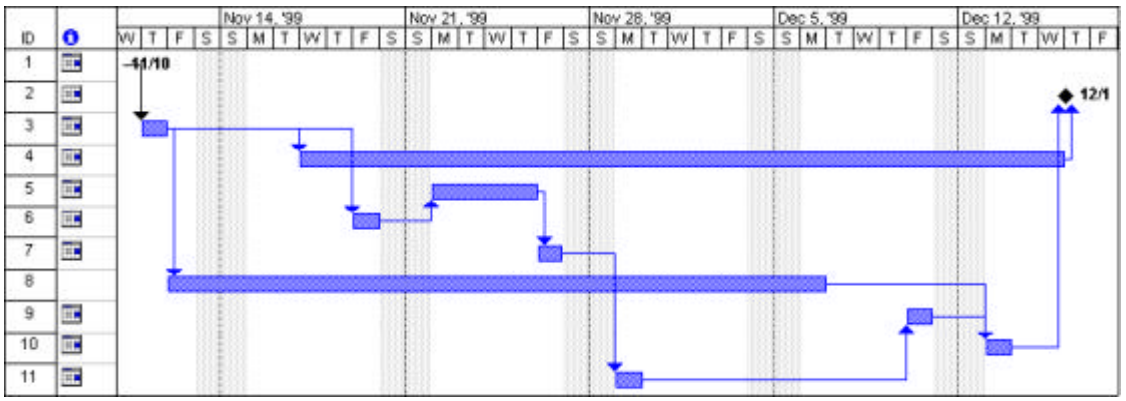

Figure 5 LCO-LCA Gantt Chart

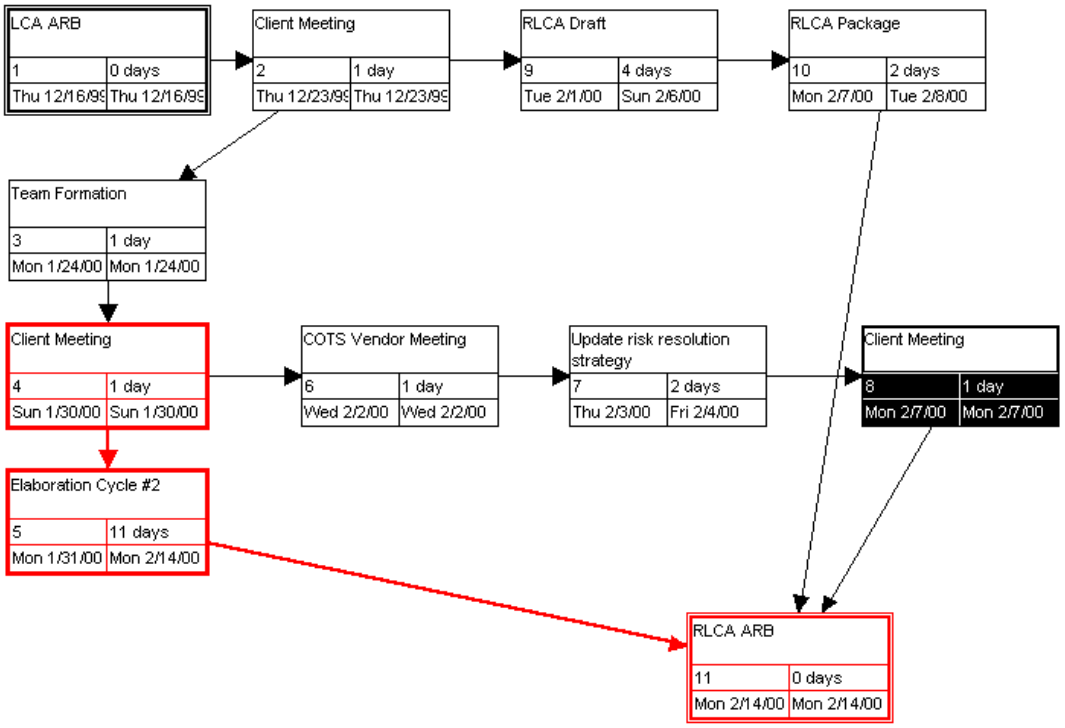

Figure 6 LCA-RLCA PERT Chart

Figure 7 LCA-RLCA Gantt Chart

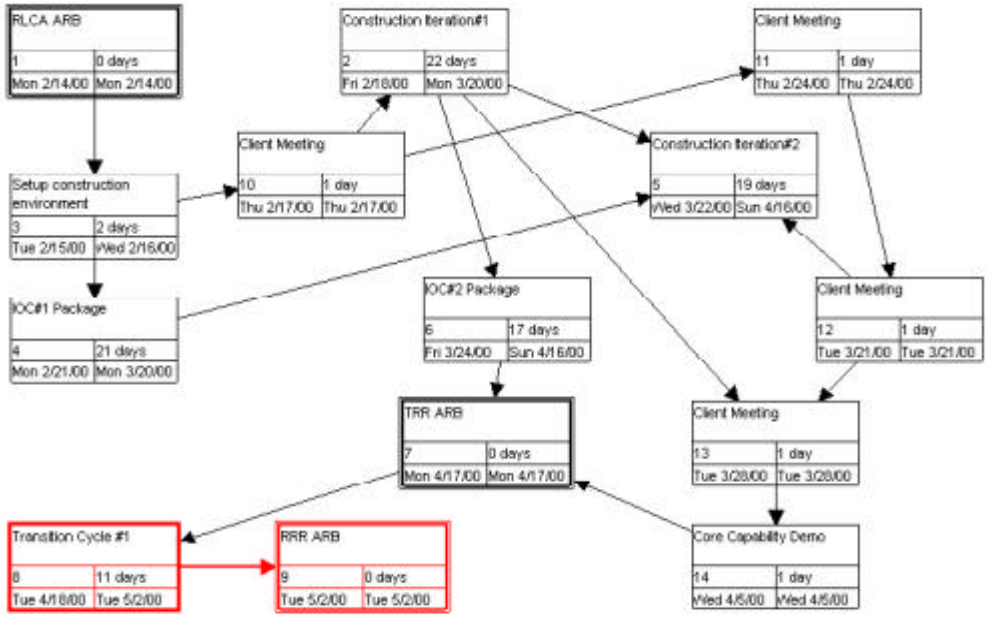

Figure 8 RLCA- RRR PERT Chart

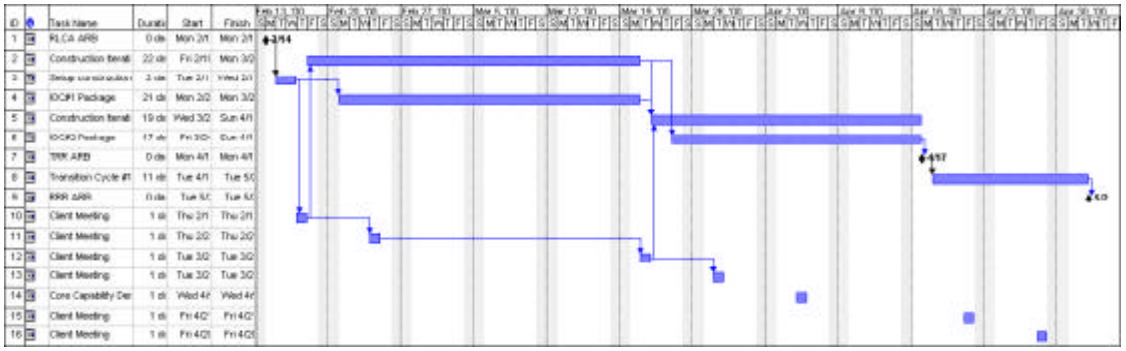

Figure 9 RLCA- RRR Gantt Chart

2.3 Project Deliverables

This section will identify the project deliverables for engineering and production stage specifying the name, version number, date, content, media, recipient, acceptance criteria for a specific deliverable. Refer to LCP 3.2 about the authors of a specific artifact.

2.3.1 Engineering Stage

LCO package elements

Date	Content	Version	Form	Media	Recipient	Acceptance

						Criteria
11/5/1999	OCD	LCO_1.0	Document	Printed	LCO ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
11/5/1999	SSRD	LCO_1.0	Document	Printed	LCO ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
11/5/1999	SSAD	LCO_1.0	Document	Printed	LCO ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
11/5/1999	LCP	LCO_1.0	Document	Printed	LCO ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
11/5/1999	FRD	LCO_1.0	Document	Printed	LCO ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
11/5/1999	Prototype	LCO_1.0	Screen shots	Electronic	LCO ARB	Available from our project web page by due date
11/5/1999	Cocomo II	LCO_1.0	Cocomo data	Electronic	LCO ARB	Available from our project web page by due date

LCA package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
12/13/1999	OCD	LCA_2.0	Document	Printed	LCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
12/13/1999	SSRD	LCA_2.0	Document	Printed	LCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
12/13/1999	SSAD	LCA_2.0	Document	Printed	LCA ARB	Date stamped, delivered in printed format

						to CS577TA by Jim Black
12/13/1999	LCP	LCA_2.0	Document	Printed	LCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
12/13/1999	FRD	LCA_2.0	Document	Printed	LCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
12/13/1999	Prototype	LCA_2.0	Screen shots	Electronic	LCA ARB	Available from our project web page by due date

RLCA package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
2/14/2000	OCD	RLCA_3.0	Document	Printed	RLCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
2/14/2000	SSRD	RLCA_3.0	Document	Printed	RLCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
2/14/2000	SSAD	RLCA_3.0	Document	Printed	RLCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
2/14/2000	LCP	RLCA_3.0	Document	Printed	RLCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
2/14/2000	FRD	RLCA_3.0	Document	Printed	RLCA ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
2/14/2000	Prototype	RLCA_3.0	Screen shots	Electronic	RLCA ARB	Available from our project web page by due date
2/14/2000	Cocomo II	RLCA_3.0	Cocomo data	Electronic	RLCA ARB	Available from our

						project web page by due date

2.3.2 Production Stage

All dates in the production stage are anticipated.

IOC Iteration #1 package elements

Date	Content	Version	Form	Media	Recipient	Acceptance Criteria
3/19/2000	OCD	IOC1_4.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	SSRD	IOC1_4.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	SSAD	IOC1_4.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	LCP	IOC1_4.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	FRD	IOC1_4.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Prototype	IOC1_4.0	Screen shots	Electronic	IOC ARB	Available from our project web page by due date
3/19/2000	Cocomo II	IOC1_4.0	Cocomo data	Electronic	IOC ARB	Available from our project web page by due date
3/19/2000	Iteration Plan	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Inspection Plan	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in

						printed format to CS577TA by Jim Black
3/19/2000	Inspection Report	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Test Plan	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Test Report	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Quality Management Plan	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Release Description	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in paper in printed format to CS577TA by Jim Black
3/19/2000	Iteration Assessment Report	IOC1_1.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
3/19/2000	Source Code	IOC1_1.0	Code	Electronic	IOC ARB	Available from our project web page by due date
3/19/2000	Executables	IOC1_1.0	Executable	Electronic	IOC ARB	Available from our project web page by due date

IOC Iteration #2 package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
4/16/2000	OCD	IOC2_5.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	SSRD	IOC2_5.0	Document	Printed	IOC ARB	Date stamped,

						delivered in printed format to CS577TA by Jim Black
4/16/2000	SSAD	IOC2_5.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	LCP	IOC2_5.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	FRD	IOC2_5.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Prototype	IOC2_5.0	Screen shots	Electronic	IOC ARB	Available from our project web page by due date
4/16/2000	Cocomo II	IOC2_5.0	Cocomo data	Electronic	IOC ARB	Available from our project web page by due date
4/16/2000	Iteration Plan	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Inspection Plan	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Inspection Report	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Test Plan	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Test Report	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Quality	IOC2_2.0	Document	Printed	IOC ARB	Date stamped,

	Management Plan					delivered in printed format to CS577TA by Jim Black
4/16/2000	Release Description	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Iteration Assessment Report	IOC2_2.0	Document	Printed	IOC ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Source Code	IOC2_2.0	Code	Electronic	IOC ARB	Available from our project web page by due date
4/16/2000	Executables	IOC2_2.0	Executable	Electronic	IOC ARB	Available from our project web page by due date

Transition Set package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
4/16/2000	Transition Plan	TS_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	User Manual	TS_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
4/16/2000	Process Guide	TS_1.0	Electronic	Electronic	ARB	Available from our project web page by due date

2.3.3 Evolution Stage

All the dates in the Evolution stage are TBD.

Support Set package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
TBD	Support Plan	SS_1.0	Document	Printed	ARB	Date stamped,

						delivered in printed format to CS577TA by Jim Black
TBD	Training Materials	SS_1.0	Training material	Printed & Electronic	ARB	Available from our project web page by due date
TBD	Regression Test Package	SS_1.0	Test data	Electronic	ARB	Available from our project web page by due date
TBD	Packaged Tools and Procedures	SS_1.0	Document	Electronic	ARB	Available from our project web page by due date

Data Collection Set package elements

Date	**Content**	**Version**	**Form**	**Media**	**Recipient**	**Acceptance Criteria**
TBD	Size Report	DC_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
TBD	COCOMO II Data Collection Form	DC_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
TBD	COQUALMO Data Collection Form	DC_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
TBD	COCOTS Data Collection Form	DC_1.0	Document	Printed	ARB	Date stamped, delivered in printed format to CS577TA by Jim Black
TBD	CORADMO Data Collection Form	DC_1.0	Document	Printed	ARB	Date stamp ed, delivered in printed format to CS577TA by Jim Black

3 Responsibilities

This section identifies stakeholders of the project, their responsibilities for performing various life cycle functions, and where organizationally they will be performed.

3.1 Stakeholder Responsibilities

Project stakeholders have been identified. Their roles, representation within the organization and their responsibilities are described in Table 1.

Table 1 Stakeholder Responsibilities

Role	Name &Title	Organization	Responsibility
Developer	Joe William (cs577 student) Jim Black (cs577 student) David Jain (cs577 student)	USC Project Team USC Project Team USC Project Team	Provide Win Conditions for developers. Design and implement the system. Provide documentation and training.
Customer	Barry Boehm (director) Winsor Brown (assistant director) Paul Grunbacher (researcher)	CSE CSE CSE	Provide Win Conditions for CSE. Procure all necessary material and personnel resources. Provide feedback on project specifications, Win conditions, and developmental progress.
User	Hwan Dong (PhD student)	USC CS	Provide Win Conditions for users. Review and test product (or its increment) in operational environment. Provide usage feedback to maintainer
Maintainer	Mohammad AlSaid (PhD student)	USC CS	Provide Win Conditions for maintainers . Operation and Evolution support for the system and its underlying operating.

3.1.1 Stakeholder Representatives

The following table gives information about stakeholder representatives for the related organizations.

Table 2 Stakeholder Representatives

Organization	Representatives
CSE (Center for Software Engineering)	Barry Boehm
USC CS (Department of Computer Science)	Hwan Dong
USC Project Team	Jim Black

3.1.2 Engineering Stage Responsibilities

Since, this project is about tailoring a COTS product, only three people are assigned to it during the engineering stage. Each member has a primary and secondary role with respect to each artifact due in LCO and LCA packages.

Table 3 Stakeholder Engineering Stage Responsibilities

	Inception	Elaboration
Developer	Perform WinWin negotiation Get familiarized with TheoryW and negotiation model Prepare LCO package elements, architectural sketches for a non operational prototype. Evaluate alternative COTS for negotiation tool Develop a risk resolution strategy to resolve the top most risks for our project such as schedule Organize meetings with client InFormal Training sessions for GroupSystems Attend LCO ARB	Develop a risk resolution strategy to resolve the top most risks for our project such as schedule and availability of GroupSystems software to do the exploring to find out about more about the product, its capabilities Refine architecture and design based on the findings about GroupSystems Refine or rebuild further prototype based on the feedback from the client Organize meetings with client Prepare LCA package elements Attend LCA ARB
Customer	PerForm WinWin negotiation Support LCO package elements Accept or reject options offered by the developers Provide feedback on the prototype and other project deliverables Monitor progress at milestones together with the project manager Attend LCO ARB Organize InFormal Internal training for teamx developers	Monitor progress of teamx through the project deliverables, reviews and meetings Review designs, prototypes, plans and feasibility Acquisition of COTS software packages Attend LCO ARB
User	PerForm WinWin negotiation Review and exercise prototype Provide feedback about the user interface screen shots the Easy WinWin	Review and exercise prototype Provide feedback about the usability of the Easy WinWin
Maintainer	PerForm WinWin negotiation	Attend technical meetings with the developers to get familiarized with the design, alternatives and secondary features that might be of interest

3.1.3 Production Stage Responsibilities

The final assignments to developers are not specific at this point, since team assignments are likely to go under a change in spring semester. Below is a table from the guidelines that give general suggestions. This table will be updated and specialized for our team's needs later in the life cycle.

Table 4 Stakeholder Production Stage Responsibilities

	Construction	Transition

Developer	If there are new team members, get training about the proposed system, architecture from the cs577a developers Refine design Implement very high priority requirements for a design to schedule process model for the first iteration. For the second iteration, implement the rest of the high, medium and low requirements. Integrate product. Perform code review, unit tests Participate in COTS vendor meetings Prepare IOC#1, IOC#2, Transition set package elements Update risk resolution strategy	Provide development support in transitioning the product. Adapt product to the CSE environment Contribute to training material.s Help maintainer about the system Prepare Support Set, Data Collection set elements
Customer	Provide a development machine and any other necessary resources Organize a COTS vendor meeting to resolve the licensing or any other relevant issues Monitor progress at milestones. Review and test operational prototype. Provide administrative support.	Provide administrative support in transitioning the product. Review the material cs577 developers deliver Help maintainer about the system description
User	Review and test product (or its increment) in development environment. Provide test support.	Review and test product (or its increment) in operational environment. Provide usage feedback to maintainer
Maintainer	N/A	Review support plan Get training from the cs577 developers about the system and get help from customer about the system description

3.1.4 Evolution stage Responsibilities

Paul Grunbacher as well as the cs577 team developers will not be able to provide any help during this stage. . Below is a table from the guidelines that give general suggestions. This table will be updated and specialized for our team's needs later in the life cycle.

Table 5 Stakeholder Evolution stage Responsibilities

	Evolution
Developer	N/A
Customer	Provide maintainer all the system information, decide on which features are to be implemented or enhanced for a year period Analyze the system performance, prepare a comparative study as a technical report Plan to make the system operational next year to be used in the classroom for next year's negotiation
User	Provide usage feedback to maintainer Prepare a report about system bugs
Maintainer	Evaluate system feedback from user, customer Review user manual Develop secondary features, enhancements or any left out low priorities Update test plan, test througly

3.2 Development Responsibilities

The following table describes development team members and their area of responsibility for each phase of the project. For more information about the responsibilities for specific stage in the life cycle, refer to LCP 3.1

Table 6 Development Responsibilities

	System Engineer (Joe William)	**Project Manager** (Jim Black)	**Designer/Implementer** (David Jain)
Inception	OCD (primary) SSRD (primary) LCP (secondary) Prototype (secondary) SSAD (secondary)	LCP (primary) FRD (primary) SSRD (secondary)	Prototype (primary) SSAD (primary) OCD (secondary) FRD (secondary)
Elaboration	OCD (primary) SSRD (primary) LCP (secondary) Prototype (secondary) SSAD (secondary)	LCP (primary) FRD (primary) SSRD (secondary)	Prototype (primary) SSAD (primary) OCD (secondary) FRD (secondary)
Construction	Tailor GroupSystems software (secondary), Software Test Plan, Code Review, User Manual, Training Manual, Quality Management Plan (primary), Inspection Plan	Planning and Control Code Review, Iteration Plan(primary), Support Plan, Transition Plan, Code Review, Iteration Reports, Release Descriptions (secondary)	Tailor GroupSystems software (primary), Unit Test Plan, Iteration Plan (secondary), Release Descriptions (primary)
Transition	Demo for the operational prototype (secondary), Training Data Collection Set(primary)	Customer Deliverables(primary) Support Set(primary) Data Collection Set(secondary)	Demo for the operational prototype (primary),, Transition to client machine Support Set(secondary)
Evolution	N/A	N/A	N/A

3.2.1 Development Organization Charts

The following chart describes the development organization and developer roles for CSCI 577.

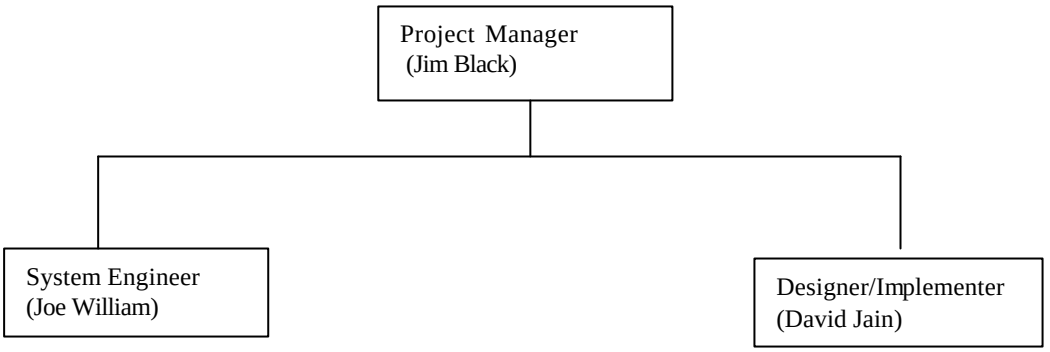

Figure 10 Development Organization Chart (CSCI 577)

Throughout the lifecycle of the project, each role will primarily correspond to the following functions:

1. Project Manager: Organize internal and external team meetings, facilitate teamwork and delegate and balance workload, plan and control, Documentation.
2. System Engineer: COTS Evaluation, Guide designer, Documentation.
3. Designer/Implementer: Design, develop software and conduct unit testing

3.2.2 Staffing

The following table describes the effort to be put in by the development team during each phase.

Table 6 Staffing Plan by Phases

Role	**Effort (Number of hours per week) for each phase**				
	Inception	**Elaboration**	**Construction**	**Transition**	**Evolution**
System Engineer	10	10	15	10	N/A
Project Manager	10	10	15	10	N/A
Designer/Implementer	10	10	15	10	N/A

Table 7 Personnel Skill Set & Planned Absences

Team Member	**Skill Set**	**Planned Absences**				
		Inception	**Elaboration**	**Construction**	**Transition**	**Evolution**
Joe William	C++, Database	2 weeks (09/1/99-09/15/99)	1 week (02/01/99-02/07/99)			N/A
Jim Black	Database	0	0			N/A
David Jain	C++, Java	0	0			N/A

3.2.3 Training

CSE is responsible for informal internal training for all of cs577 developers about the GroupSystems software. This will be performed by Paul Grunbacher in SAL322 at 9/29/1999 from 10:00-12:00. The training will explain GroupSystems software capabilities as a collaborative tool well as the usage of different tool it provides.

A need for another training at the beginning of the production stage might occur if none of the team members stay for the second semester. Otherwise, there will be an internal training given by the team member who had stayed for cs577b to the new team members about the system description, architecture and EasyWinWin project in general.

4 Approach

This section describes the activities to be performed in the life cycle of Easy WinWin and identifies how they will be managed.

4.1 Monitoring and Control

We monitor and control the progress of Easy WinWin life cycle plan by the following means:

4.1.1 Closed loop feedback control

4.1.2 Reviews

Each phase in our life cycle stage is concluded with a formal review by ARB members: cs577 teaching staff and our customers. The reviews are scheduled primarily based on cs577 schedule and secondarily on the preferences of team members among three time choices provided by cs577 TAs. All the reviews are to be conducted in conference room, SAL 322, provided by CSE for 2 hours. The purpose of the review is to ascertain whether the project is ready to move onto the next phase in the development cycle. The exit criteria is determined by the ARB members. At the end of the review, ARB members will provide an evaluation sheet discussing things done right and things need action to provide us feedback for the rest of the life cycle. Verbal, written, as well as electronic feedback will be gathered to facilitate the production of the final products. The post-review documents such as the LCO, LCA packages will be disseminated via the World Wide Web. Document and products generated by review in the later stages of the life cycle will be baselined. The baselined elements are put under configuration management to provide a stable foundation for the next phase of the project.

The following reviews are established in our Easy WinWin's project life cycle in chronological order:

- Life cycle objective architectural review board review (LCO-ARB)
- Life cycle architecture architectural review board review (LCA-ARB)
- Rebaselined life cycle architecture architectural review board review (RLCA -ARB)
- Transition readiness review (TRR)
- Release readiness review (RRR)

The project manager also reviews our weekly reports to help us better stick with the process plan or do any changes if necessary.

4.1.3 Status Reporting

- Milestones established in our life cycle plan are tracked using the Microsoft Project 98 software
- The dependencies and scheduling is established using the *Gantt chart*
- *PERT chart* is used to identify the critical path of the development cycle
- *Resource graph* is used to monitor the load of each individual resources
- WBS code is used to track the expenditure of each development task
- Weekly effort report submitted by each team members is used to track team members' individual effort on a weekly basis, refer to http://nunki.usc.edu:8082/teamx/index.html
- The project manager uses the effort reports to track the overall personnel load as well as the project development progress and to prepare for the periodic progress reports, refer to

http://nunki.usc.edu:8082/teamx/Control/Weeka/Weeka.doc where Weeka refers to the specific week such as Week 00(9/13/1999 to 9/19/1999), Week02, etc..

- Email communications are used to communicate unscheduled personnel unavailability

Measurement Analysis: The process of measurement will be performed during prototyping in the reviews. This way, the information will be communicated to all project stakeholders. As unit of measurement, "M" criteria for the requirement will be taken as basis.

4.1.4 Risk Monitoring and Control

The risk analysis, namely, risk's description, risk exposure, risk reduction leverage, action to mitigate the risk, contingency plan, ranking and risk assessment are explained in detail in FRD 4. Monitoring procedures are performed by tracking Top N Risk items using the Top N Risk Item List, Table 2, maintained by the project manager each week to assess current status of the project risks and identify those that need immediate attention from those whose mitigation can wait, refer to to http://nunki.usc.edu:8082/teamx/Control/Weeka/Weeka.xls where Weeka refers to the specific week such as Week 00(9/13/1999 to 9/19/1999), Week02, etc..

Table 2 Top N Risk Item List

Risk Item & Description	Weekly Ranking			Action
	Current	Previous	# Weeks	
Schedule – an independent variable, delivery in 12 weeks	1	3	4	Negotiate with the customer to move the web viewing requirement to be evolutionary
Poor communication – customers have very busy schedule to attend the meetings regularly	2	2	3	Organize a teleconference meeting when customers are out of USC

4.1.5 Project Communication

Project team members and clients use the following methods to communicate with one another:

- Email for main group communications (such as sending the meeting minutes, reminder for meetings, asking project specific questions)
- Cellular phone for emergency situations (such as when a member is missing from the meeting)
- Video & voice conference for meetings (such as when either team members or customers can not be in the same geographical region)
- Team web site for documents sharing and prototype development (such as when viewing for consistency checking among different artifacts)
- Face-to-face meetings when our schedule accommodates (such as ARB reviews, client meetings, internal project meetings)

Email is the most extensively used method of communication for group communications. Ideas as well as documents (MS Word, MS Project, MS PowerPoint, Adobe Portable Document Format (PDF)) are disseminated and reviewed via emails. As needs arise, conference meetings are held to discuss topics with the customers. Occasional weekend meetings with the entire team are held to brainstorm and establish consensus for various aspects of the project. The project team has also established a team home page on the class web site and uses it for group communication and distribution of artifacts.

4.2 Methods, Tools and Facilities

Currently, we are concerned that about availability of GroupSystems software to be used for the next level prototyping.

The following tools are used for each phase of the life cycle.

Life Cycle:

1. Microsoft Word 2000– Word processing for LCO, LCA package elements except Prototypes and Cocomo II results
2. Microsoft Project 98– Project tracking and planning for Gantt and Pert charts supplied with LCP
3. Netscape 4.5 and Internet Explorer 5.0–Accessing information on class web page World Wide Web for LCO and LCA package elements
4. POP3 compliant email client – Email communications
5. Adobe Acrobat 4.0—documentation dissemination

Engineering Stage:

1. Unix WinWin Tool – Requirements investigation and negotiation
2. Rational Rose 98i – Modeling and Design for SSAD specific diagrams
3. USC COCOMO II– Cost estimation
4. Microsoft PowerPoint 2000—ARB Presentations
5. Microsoft Front Page 2000 – Prototyping for ARBs

Production Stage:

1. Window NT 4.0—Operating system
2. USC COCOMO II– Cost estimation
3. Microsoft PowerPoint 2000—ARB Presentations
4. Rational Rose 98i – Object oriented analysis and design for SSAD specific diagrams

Evolution stage:

TBD

4.3 Configuration Management

Configuration management is required to ensure the smooth progress of artifact construction. It provides a stable foundation for development by baselining the key documents and maintaining strong control over their changes. At each of the formal reviews during the life cycle, the specific package elements (such as LCO_OCD) are baselined.

4.3.1 Product Element Identification

All product elements of a specific package are identified by the following scheme:

Prj_Elt_Milestone_Version
Prj = Project name
Elt = Element name
Milestone = Milestone the element belongs to
Version = Version Number

For example, the OCD version 1.1 from the LCA milestone are identified as:

- EasyWinWin_OCD_LCA_1.1

The prototype from the LCO milestone are identified as:

- EasyWinWin_Prototype_LCO_0.2

Ownership Information is discussed in Section 3.2. The elements are placed under the configuration management as the guidelines suggest.

4.3.2 Configuration Change Management

We have classified changes into two groups: first group consists of simple changes that do not affect budget and schedule, whereas the second group consists of changes that have an effect on the budget and schedule or any significant effect that might involve other stakeholders.

The following process outlines how the simple changes are managed.
A simple change request (CR-1) is an informal inspection form that consists of the date of the request, the change requested and which configuration item needs to be changed. Configuration Change Management Process consists of the following states:

1. Submit: create a CR-1, a team member is assigned to be the fixer to investigate the CR-1 by the project manager
2. Analyze: assigned team member reviews the requested change
3. Fix: developer fixes the requested change
4. Verify: a team member other than the assigned fixer verifies the fix
5. Release: the fix is integrated into the element where it belongs to

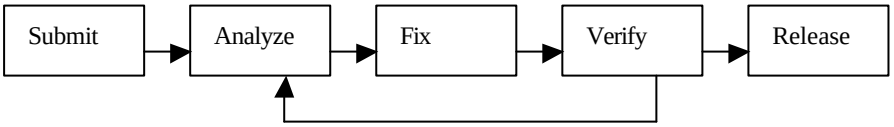

Figure 11 Configuration Simple Change Process Flow

The following process outlines how more significant changes are managed. A significant change request (CR-2) is a formal inspection form that consists of the date of the request, the change requested, which configuration item needs to be changed, analysis of the problem/change, effects that change would cause

1. Prepare: create a CR-2, hand it to the project manager
2. Review: project manager organizes an internal project meeting to discuss the changes in depth and puts all the gathered information as a report
3. Inform: team calls for an exceptional meeting to inform all of the stakeholders about the change itself and the impact
4. Decide: stakeholders decide on how the change will be handled and incorporate it to the project

Refer to ~csci577/www/teamx/Etc/CR-1 for our collection of simple change request forms and . ~csci577/www/teamx/Etc/CR-2 for significant change request forms.

4.3.3 Product Element Management

Our Easy WinWin project elements are stored at the USC team directory located at ~csci577/www/teamx. All of our written documents can be accessed directly via a web browser, as well as by accessing via file transfer or directly logging into the USC account. We rely on the USC ISD to provide back up and security. Each element is managed by its owner as listed in the Section 4.2.1.1.

4.3.4 Configuration Status Management

The development status of each element is tracked by the MS Project. Addition to that, the product elements are placed in our project directory, http://nunki.usc.edu:8082/teamx based on our product element identification strategy. At any given time, this convention will help clarify any issues regarding the configuration status. Each developer is responsible for reporting to the project manager his/her development progress.

We are also planning to establish a small scale database to keep the following version history information, the version number, the last modification date, owner of the change, description of the change. With the database capability, we will be able to run queries or gather information about our configuration status.

In order for the source code changes to be compatible, regression test cases will be run. Inspections will be used for the changes in printed format.

4.4 Quality Management

4.4.1 Quality Assurance

The Easy WinWin is mainly going to be tailoring various tools in GroupSystems, for a successful integration, the tailoring is implicitly forced to adhere to this framework. If any further coding needs to be done, we will adhere to the relevant portions of Java Coding Standard [refer to http://www.infospheres.caltech.edu/resources/code_standard/java_standard.html]
Verification of the project's compliance with its documentation and code standards will be done automatically by the tools provided as part of the specified website.

The development team will later deliver the following quality documents:

```
1. Software Test Description (STD)
2. Software Test Plan (STP)
3. Software Test Report(s) (STR)
```

These documents follow the ANSI/IEEE Format when applicable.

Each tool tailoring will be reviewed by at least one other team member. Auditing of the projects compliance of its plans, policies and procedures will be done primarily by the project manager and the cs577 class grader.

Each review will have a moderator who is responsible for collecting the metrics associated with review. The results of the review will be stored in class project directory. The development team will meet at the end of each major milestone delivery and win-win iteration to review the results of the reviews and the tests to determine how to improve the development process. This meeting will also serve as a forum to follow up on previous corrective actions implemented to eliminate reported QA deficiencies.

In the first phase the plans will be developed for both iterations. Once development is complete, the developers will help finish up the testing. Each person will test his own module and will be responsible for completing the test report for the areas that they tested.

4.4.2 Verification and Validation

Verification of the system will be done through unit testing, integration testing and system level testing. Both the unit testing and the integration testing will be done in a non-Formal manner. It is the responsibility of each developer to come up with his or her own unit test plan, unit test environment, integration plan and associated documentation. The system test plan will verify the system level requirements.

Validation will be performed by iterative prototyping throughout the life cycle.

5 Resources

The following resources will be required

5.1 Work Breakdown Structure

A Management
- AA Inception phase management
  - AAA Top-level Life Cycle Plan (LCO version of LCP)
  - AAB Inception phase project control and status assessments
  - AAC Inception phase stakeholder coordination
  - AAD Elaboration phase commitment package and review (LCO package preparation and ARB review)
- AB Elaboration phase management
  - ABA Updated LCP with detailed construction plan (LCA version of LCP)
  - ABB Elaboration phase project control and status assessments
  - ABC Elaboration phase stakeholder coordination
  - ABD Construction phase commitment package and review (LCA package preparation and ARB review)
- AC Construction phase management
  - ACA Updated LCP with detailed transition and evolution plans
  - ACB Construction phase project control and status assessments
  - ACC Construction phase stakeholder coordination
  - ACD Transition phase commitment package and review (IOC package preparation and PRB review)
- AD Transition phase management
  - ADA Updated LCP with detailed next-generation planning
  - ADB Transition phase project control and status assessments
  - ADC Transition phase stakeholder coordination
  - ADD Evolution stage commitment package and review (PR package preparation and PRB review)

B Environment and Configuration Management (CM)
- BA Inception phase environment/CM scoping and initialization
- BB Elaboration phase environment/CM
  - BBA Development environment installation and administration
  - BBB Elaboration phase CM
  - BBC Development environment integration and custom toolsmithing
- BC Construction phase environment/CM evolution
  - BCA Construction phase environment evolution
  - BCB Transition phase CM
- BD Transition phase environment/CM evolution\
  - BDA Construction phase environment evolution
  - BDB Transition phase CM
  - BDC Evolution stage environment packaging and transition

C Requirements
- CA Inception phase requirements development
  - CAA Operational Concept Description and business modeling (LCO version of OCD)
  - CAB Top-level System and Software Requirements Definition (LCO version of SSRD)
  - CAC Initial stakeholder requirements negotiation

- CB Elaboration phase requirements baselining
  - CBA OCD elaboration and baselining (LCA version of OCD)
  - CBB SSRD elaboration and baselining (LCA version of SSRD)
- CC Construction phase requirements evolution
- CD Transition phase requirements evolution

D Design
- DA Inception phase architecting
  - DAA Top-level system and software architecture description (LCD version of SSAD)
  - DAB Evaluation of candidate COTS components
- DB Elaboration phase architecture baselining
  - DBA SSAD elaboration and baselining
  - DBB COTS integration assurance and baselining
- DC Construction phase design
  - DCA SSAD evolution
  - DCB COTS integration evolution
  - DCC Component design
- DD Transition phase design evolution

E Implementation
- EA Inception phase prototyping
- EB Elaboration phase component implementation
  - EBA Critical component implementation
- EC Construction phase component implementation
  - ECA Alpha release component coding and stand-alone testing
  - ECB Beta release (IOC) component coding and stand-alone testing
  - ECC Component evolution
- ED Transition phase component evolution

F Assessment
- FA Inception phase assessment
  - FAA Initial assessment plan (LCO version; part of SDP)
  - FAB Initial Feasibility Rationale Description (LCO version of FRD)
  - FAC Business case analysis (part of FRD)
- FB Elaboration phase assessment
  - FBA Elaboration of assessment plan (LCA version; part of SDP)
  - FBB Elaboration feasibility rationale (LCA version of FRD)
- FC Construction phase assessment
  - FCA Detailed test plans and procedures
  - FCB Evolution of feasibility rationale
  - FCC Peer reviews
  - FCD Alpha release assessment
  - FCE Beta release (IOC) assessment
- FD Transition phase assessment

G Deployment
- GA Inception phase deployment planning (LCO version; part of SDP)
- GB Elaboration phase deployment planning (LCA version; part of SDP)
- GC Construction phase deployment planning and preparation
  - GCA Transition plan development
  - GCB Evolution plan development
  - GCC Transition preparation
- GD Transition phase deployment

WBS Element	TYPE	EXPENDITURE CATEGORY	LIFE CYCLE PHASE	PROJECT ACTIVITY	BUDGET	RESPONSIBILITY
AAA, AAB, AAC,	Personnel		Inception			Project Manager
ABA, ABB, ABC,	Personnel		Elaboration			Project Manager
ACA, ACB, ACC,	Personnel		Construction			Project Manager
ADA, ADB, ADC	Personnel		Transition			Project Manager
AAD	Personnel		Inception			Teamx
ABD	Personnel		Elaboration			Teamx
ACD	Personnel		Construction			Teamx
ADD	Personnel		Transition			Teamx
BA	Personnel		Inception			Teamx
BB	Personnel		Elaboration			Teamx
BC	Personnel		Construction			Teamx
BD	Personnel		Transition			Teamx

5.2 Budgets

The project is to be designed and implemented in two 12-week periods in Fall 1999 and Spring 2000 by a project team of three people. Labor cost for the project is calculated as follows. The project requires 12 weeks for its Engineering Stage and 12 weeks for its Production Stage and Evolution stage.

Project team for Engineering Stage consists of three members. Let's assume for the time being there will be three people in Production and Evolution stage. Average compensation for each person in the project team with similar background and experience, adjusted to the southern California region, is $200/hr. Average work hours for Engineering Stage are estimated to be 10 hours for each person, which turns out to be 30 man-hours per week by the team. Thus, labor cost for Engineering and Evolution stage is estimated to be $144,000. The project is to be implemented by CSCI 577a and b students, therefore there is no monetary cost to the customer.

Cost for the computer system server specified in [SSRD 3.1] is expected to be $10,000. However, since CSE is receiving a machine with such features from USC ISD, the practical cost is free. The project team will use COTS packages that are freeware or commercial packages that are available free of charge to academic institutions. Thus, COTS package cost is negligible. However, there is a need for a half-time system administrator for Easy WinWin administration and for implementing the evolutionary requirements in the Evolution stage as a maintainer. Labor cost for a half-time system administrator, using Southern California compensation data, is expected to be $25,000 per year for first year of the system operation. Such cost will vary in subsequent years, depending on compensation data for such personnel. CSE can, however, realize significant savings by designating a graduate assistant to be the system administrator. There is a cost of training associated with the Transition phase of Production stage. The actual cost associated with facility usage and training personnel are minimal since the location of training can be on-site at the USC campus and trainers will be members of the project developer team. There might be a cost

, however, associated with the production of training material and Easy WinWin manuals. The actual cost for printed material for a production of 100 copies of training material and 100 copies of system manual would be approximately $2000.

Evolution stage consists of evolution costs consisting of implementing evolutionary features of the system as well as bug fixes and minor enhancements. These tasks can be performed by designated maintainer. For a small scale system, the maintainer is expected to spend 10 man hours per week. The labor cost is estimated to be $6,000 based on USC research assistants' salary.

STAGE/COST	Engineering	Production	Support
CSCI 577a Developers CSCI 577b Developers	$0 $0	$0 $0	$0 $0
System administrator/Research assistant	$0	$25,000/year or $6,000/year	$6,000/year
COTS software	Nominal	Nominal	Nominal
Pentium Machine	$0	$0	$0
Site Evolution	$0	Nominal	Nominal
Printed Material	$0	$2000	$0

Table 3 Easy WinWin Development Monetary Cost

Appendix

A. COCOMO Results

Provided as a separate document

B. Detailed Gantt Charts for milestones and schedules

See Figures

C. Pert Charts for milestones and schedules

See Figures

D. Cocomo II Report (sloc)

Provided as a separate document