Life Cycle Plan (LCP)

We Are Trojans (WAT) Network

Team01

Team members	Roles
Eirik Skogstad	Project Manager, Life Cycle Planner
Min Li	Feasibility Analyst, Operational Concept Engineer
Pittawat Pamornchaisirikij	NDI/NCS Acquirer & Evaluator, Tester
Punyawee Pakdiying	System Architect, Feasibility Analyst
Saloni Priya	Requirements Engineer, UML Modeler
Ameer Elkordy	IIV&V, Quality Focal Point
Suleyman Erten	Operational Concept Engineer, Requirements Engineer
Kamonphop Srisopha	Prototyper, UML Modeler

4/1/15

Version History

Date	Author	Version	Changes made	Rationale
09/28/14	ES	1.0	Created document from ICSM template, updated team roles and section 3.3 skills.	For VCP package submission.
10/11/14	ES	1.5	Updated sections 1-5.	For draft FCP package submission.
10/19/14	ES	2.0	Updated some sections according to feedback from FCR ARB: - Updated artifacts to be delivered before DCR ARB. - Updated team member roles during development phase.	For FCP package submission.
11/30/14	ES	2.5	Updated iteration plan, team member roles, new effort estimation.	For draft DCP submission.
12/08/14	ES	3.0	Updated section 6.1, minor changes to other sections based on ARB feedback.	For DCP submission.
2/16/15	ES	4.0	Updated deliverables, roles for construction and transition phases. Updated information on monitoring, control and feedback. Added CCD preparation plans.	For RDCP submission.
4/1/15	ES	5.0	Updated section 6.2.2 with CCD results.	For CCD Report submission.

Table of Contents

Table of Tables

Table of Figures

No table of figures entries found.

1. Introduction

1.1 Purpose of the LCP

The LCP serves as a basis for controlling and monitoring the project's progress. It identifies available people, skills and resources in order to make the best use of these in the project's life cycle. It also gives proof to key stakeholders that the major life cycle issues is known and thought trough in advance.

1.2 Status of the LCP

The LCP is currently at the DCP version number 4.0. This version reflects changes in the plan as a result of the re-baselined project phase.

1.3 Assumptions

- The duration of the project is 24 weeks, which are 12 weeks in fall 2014 and 12 weeks in spring 2015.
- All team members listed on the title page will work on the project during the fall semester and the spring semester.

2. Milestones and Products

2.1 Overall Strategy

The We Are Trojans (WAT) Network is following the architected agile process, as a big part of the project involves modification of existing software to create the online forum core capability.

Exploration phase
> **Duration:** 9/10/14-9/29/14
> **Concept:** Identify operational concept, system and software requirements and architecture, and life-cycle plan.
> **Deliverables:** Client Interaction Report, Valuation Commitment Package
> **Milestone:** Valuation Commitment Review
> **Strategy:** One Incremental Commitment Cycle, Risk assessment analysis, Win-Win Negotiation Sessions.

Valuation phase
> **Duration:** 9/30/14-10/15/14
> **Concept:** Identify Objectives, Constraints and Priorities, Develop operation concept, Explore alternatives, Provide project feasibility evidence, Prototyping, Assess and plans to mitigate risks, Plan and manage project, Perform win-win negotiation, Define quality and configuration policy.
> **Deliverables:** Draft Foundations Commitment Package
> **Milestone:** Foundations Commitment Review
> **Strategy:** One Incremental Commitment Cycle, Risk assessment analysis, Win-Win Negotiation Sessions, Planning Poker.

Foundations phase
> **Duration:** 10/16/14-12/12/14
> **Concept:** Assess Project Status, Plan and Manage Project, Manage Project Quality, Prototyping, Develop Software Architecture.
> **Deliverables:** Foundations Commitment Package, Draft Development Commitment Package, Development Commitment Package.
> **Milestone:** Development Commitment Review
> **Strategy:** One Incremental Commitment Cycle, Risk assessment analysis.

Re-baselined Foundations phase
> **Duration:** 1/12/15-2/13/15
> **Concept:** Re-baseline Project Status, Prepare for Development Phase, Plan for Testing, Plan and Manage Project.
> **Deliverables:** Re-baselined Development Commitment Package
> **Milestone:** Re-baselined Development Commitment Review
> **Strategy:** One Incremental Commitment Cycle, Risk assessment analysis.

Development phase – Construction
> **Duration:** 1/26/15-4/17/15.
> **Concept:** Construction Iteration, build and test all modules.
> **Deliverables:** CCD Report, TRR package.
> **Milestone:** Transition Readiness Review.
> **Strategy:** Two development cycles.

Development phase – Transition
> **Duration:** 4/17/15-5/1/15.
> **Concept:** Transition Iteration, client training and system deployment.
> **Deliverables:** Project Archive.
> **Milestone:** End of project.
> **Strategy:** Provide client training and deploy system.

2.2 Project Deliverables
2.2.1   Exploration Phase

Table 1: Artifact deliverables in Exploration Phase

Artifact	Due date	Format	Medium
Client Interaction Report	9/19/2014	.doc, .pdf	Soft copy
Program Model & Business Process Model	9/21/2014	.doc, .pdf	Soft copy
Valuation Commitment Package • Operational Concept Description (OCD) Early Section • Life Cycle Plan (LCP) Early Section • Feasibility Evidence Description (FED) Early Section	9/29/2014	.doc, .pdf	Soft copy
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

2.2.2   Valuation Phase

Table 2: Artifact deliverables in Valuation Phase

Artifact	Due date	Format	Medium
Prototype Presentation	10/3/2014	.pdf	Soft copy
Draft Foundations Commitment Package • Operational Concept Description (OCD) All sections • Life Cycle Plan (LCP) Sections 1-5 • Feasibility Evidence Description (FED) Sections 1-5 • Prototype Description (PRO) All sections • System and Software Architecture Description (SSAD) Sections 1 & 2	10/13/14	.doc, .pdf	Soft copy
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

2.2.3   Foundations Phase

Table 3: Artifact deliverables in Foundations Phase

Artifact	Due date	Format	Medium
Foundations Commitment Package • Operational Concept Description (OCD) All sections • Life Cycle Plan (LCP) Sections 1-5 • Feasibility Evidence Description (FED) Sections 1-5 • Prototype Description (PRO) All sections	10/20/14	.doc, .pdf	Soft copy

• System and Software Architecture Description (SSAD) Sections 1 & 2			
Client Feedback Form			
Draft Development Commitment Package	12/01/14		
Development Commitment Package	12/08/14		
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

2.2.4   Rebaselined Foundations Phase

Table 4: Artifact deliverable in Rebaselined Foundations Phase

Artifact	Due date	Format	Medium
Rebaselined Development Commitment Package	9/02/15	.doc, .pdf	Soft copy
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

2.2.5   Development Phase

Table 5: Artifact deliverable in Development Phase

Artifact	Due date	Format	Medium
Core Capabilities Drive-through Report	3/30/15	.doc, .pdf	Soft copy
Transition Readiness Review Package	4/13/15	.doc, .pdf	Soft copy
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

2.2.6   Transition Phase

Table 6: Artifact deliverable in Development Phase

Artifact	Due date	Format	Medium
Project Archive	5/4/15	.zip	Soft copy
Project Effort	Every Monday	Text	Bugzilla
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

3. Responsibilities

3.1 Responsibilities by Phase

Table 7: Stakeholder's Responsibilities in each phase

Team Member / Role	Primary / Secondary Responsibility				
	Exploration	**Valuation**	**Foundations**	**Development-Construction Iteration**	**Development - Transition Iteration**
Eirik Skogstad CS577a: Project Manager, Life-cycle planner CS577b: Project Manager, Life-cycle planner, Builder, Tester	**Primary Responsibility** Create and follow up action items. Record project progress. **Secondary Responsibility** Detail Project Plan. Identify responsibilities and skills.	**Primary Responsibility** Create and follow up action items. Record project progress. **Secondary Responsibility** Identify milestones ad products. Estimate project effort and schedule.	**Primary Responsibility** Create and follow up action items. Record project progress. **Secondary Responsibility** Provide Process Feasibility Evidence. Develop Iteration Plan.	**Primary Responsibility** Create and follow up action items. Record project progress. Build software. Test software. **Secondary Responsibility** Assess development iteration. Develop Transition Plan.	**Primary Responsibility** Create and follow up action items. Record project progress. Build software. Test software. **Secondary Responsibility** Assess development iteration. Develop Support Plan.
Min Li CS577a: Feasibility Analyst, Operational Concept Engineer CS577b: Builder, Tester	**Primary Responsibility** Gather risks. Assess and plan to mitigate risks.	**Primary Responsibility** Provide evidence of feasibility of architecture. Analyze Business Case.	**Primary Responsibility** Assess Feasibility Evidence. Provide conclusion and recommendation.	**Primary Responsibility** Build software **Secondary Responsibility** Testing	**Primary Responsibility** Build software **Secondary Responsibility** Testing
Punyawee Pakdiying CS577a: System Architect, Feasibility Analyst CS577b: System Architect,	**Primary Responsibility** Gather risks. Assess and plan to mitigate risks. **Secondary Responsibility** Explore NDI alternatives.	**Primary Responsibility** Define high-level architecture. Document architecture feasibility. **Secondary Responsibility**	**Primary Responsibility** Define platform-dependent architecture. Document architecture feasibility. **Secondary**	**Primary Responsibility** Assess system architecture. Build software. Test software. **Secondary Responsibility** Identify risks and	**Primary Responsibility** Assess system architecture. Build software. Test software. **Secondary Responsibility** Identify risks and

		Explore NDI alternatives.	**Responsibility** Explore NDI alternatives.	feasibility coming up during development period	feasibility coming up for transition
Builder, Tester					
Pittawat Pamornchaisirikij CS577a: NDI Acquirer & Evaluator, Tester CS577b: Tester, Builder	**Primary Responsibility** Gather risks. Assess and plan to mitigate risks. **Secondary Responsibility** Explore NDI alternatives.	**Primary Responsibility** Explore NDI alternatives. Assess and evaluate NDI components. **Secondary Responsibility** Explore NDI alternatives.	**Primary Responsibility** Acquire NDI components. Assess and evaluate NDI components. **Secondary Responsibility** Explore NDI alternatives.	**Primary Responsibility** Develop testing strategies. Test Software. **Secondary Responsibility** Build software.	**Primary Responsibility** Develop testing strategies. Test Software. **Secondary Responsibility** Build software.
Saloni Priya CS577a: Requirements Engineer, UML Modeler CS577b: Builder, Tester	**Primary Responsibility** Gather win-conditions from stakeholders. Capture win-conditions in win-win session. **Secondary Responsibility** Perform feasibility evidence for the requirements	**Primary Responsibility** Document most significant requirements. Prioritize the requirements. **Secondary Responsibility** Develop the respective UML diagrams	**Primary Responsibility** Document detailed requirements. **Secondary Responsibility** Familiarize with the NDI/NCS and COTS	**Primary Responsibility** Update requirements. Build software. Test software. **Secondary Responsibility** Check whether requirements or being developed as per the negotiations	**Primary Responsibility** Client training. **Secondary Responsibility** Build software.
Kamonphop Srisopha CS577a: Prototyper, UML Modeler CS577b: Builder, Tester	**Primary Responsibility** Assess Prototype and Components, Develop Prototype **Secondary Responsibility** Model a UML component of the system, Analyze the proposed system	**Primary Responsibility** Develop prototype. Get prototype feedback. **Secondary Responsibility**Model a UML component of the system, Analyze the proposed system, modified existing UML	**Primary Responsibility** Tailor components. **Secondary Responsibility** Model a UML component of the system, Analyze the proposed system,modified existing UML	**Primary Responsibility** Build software. Test software. **Secondary Responsibility** Assess, develop prototype.	**Primary Responsibility** Client training **Secondary Responsibility** Build software.
Suleyman Erten CS577a: Operational Concept Engineer,	**Primary Responsibility** Identify shared vision. **Secondary**	**Primary Responsibility** Develop new operational concept.	**Primary Responsibility** Assess operational concept.	**Primary Responsibility** Identify organizational and operational transformation.	**Primary Responsibility** Verified whether the developed project is satisfied

Requirements Engineer CS577b: Builder, Tester	**Responsibility** Negotiate win-win conditions with the client	**Secondary Responsibility** Fill in Bugzilla tickets associated to OCD	**Secondary Responsibility** Perform Feasibility Analysis	Build software. Test software. **Secondary Responsibility** Analyze the possible changing in requirements	operational concepts. Build software. Test software. **Secondary Responsibility** Analyze the possible changing in requirements
Ameer Elkordy CS577a: IIV&V, Quality Focal Point CS577b: IIV&V, Quality Focal Point, Builder, Tester	**Primary Responsibility** Verify and validate work products.	**Primary Responsibility** Verify and validate work products. Construct traceability matrix. **Secondary Responsibility** Nag team on using Bugzilla more consistently.	**Primary Responsibility** Verify and validate work products. Assess quality management strategy. Identify configuration management strategy. **Secondary Responsibility** Nag team on using Bugzilla more consistently.	**Primary Responsibility** Verify and validate work products. Build software. Test software. **Secondary Responsibility** Nag team on using Bugzilla more consistently. Test software bugs.	**Primary Responsibility** Verify and validate work products. **Secondary Responsibility** Build software. Test software.
Matthew Wong Client		Review prototype and give feedback. Clarify doubts.	Review prototype and give feedback. Clarify doubts.	Provide feedback. Test system.	Participate in training.
Linda Suen Client		Review prototype and give feedback. Clarify doubts.	Review prototype and give feedback. Clarify doubts.	Provide feedback. Test system.	Participate in training.

3.2 Skills

Team members	Role	Skills
Eirik Skogstad	Project Manager, Life Cycle Planner	**Current skills:** Software Project Management, C++, Java, HTML, PHP, MySQL, JavaScript, XML, Winbook, MS Project, Life-Cycle tools, Git

		Required skills:
Min Li	Feasibility Analyst, Operational Concept Engineer	**Current skills:** HTML, CSS, PHP, JavaScript, JSP, JQuery, Strut 2.0, Spring 2.0, Ibatis, Java, C++, API application, JSON, XML **Required skills:** Analytical skills, web development, UML modeling
Pittawat Pamornchaisirikij	NDI/NCS Acquirer & Evaluator, Tester	**Current Skills:** C, C++, HTML, PHP, JavaScript, CSS, socket programming, device driver programming in Linux, Joomla CMS, Drupal CMS, Thai language, English language **Required Skills:** Experience in web development domain
Punyawee Pakdiying	System Architect, Feasibility Analyst	**Current Skills:** Linux/Unix Command line (beginner level), Java, PHP, MySQL, SQL, HTML, JavaScript, CSS, Joomla, Drupal **Required Skills:** High level Linux/Unix Command line, DBMS, Middleware, Analytical skills, and NDI/NCS used in the project
Saloni Priya	Requirements Engineer, UML Modeler	**Current Skills:** General purpose modelling language, ArgoUML, Creately for UML, MySQL,PHP,CSS, HTML, rustic JavaScript, C# **Required Skills:** Winbook, Visual Paradigm,

		Communication skills for effective negotiation, Knowledge Contributor, Analytical skills, NDI/NCS/COTS experience and knowledge about web server hosting.
Ameer Elkordy	IIV&V, Quality Focal Point	**Current skills:** Database, Analytics, General Scripting **Rusted skills:** HTML, CSS, PHP, Java **Required skills:** Organization, Tracking, Winbook, Loud Voice, and Process & Deliverable Knowledge
Suleyman Erten	Operational Concept Engineer, Requirements Engineer	**Current Skills:** HTML, JavaScript, Java **Required Skills:** Goal Decomposition, Use Case Scenarios, UML Modelling, Winbook, Knowledge Contributor
Kamonphop Srisopha	Prototyper, UML Modeler	**Current Skills:** HTML, Java, PHP, MySQL, Rust, C++, Winbook **Required Skills:** Advanced UNIX/ Linux Command line, R (statistical programming), NDI/ NCS/COTS integration knowledge

4. Approach

4.1 Monitoring and Control

The project is monitored with a bi-weekly progress report and project plan. The progress report includes a progress description with total time spent on the project. It also includes the top project risks, number of SLOC, COTS software and defects/concerns. In addition, development tasks and bugs are monitored with Bugzilla. Development progress is asserted in weekly team meetings.

4.1.1 Closed Loop Feedback Control

Draft documents/sketches/diagrams are shared on Google Drive where team members can quickly review and update them. Feedback for a specific artifact is given to the artifact leader either person-to-person or in team meetings. Developed features are reviewed each week in a team meeting. Overall project status is discussed in ad-hoc meetings after each major ARB/review based on feedback from the professor and the TAs.

4.1.2 Reviews

- Team meetings; we have a team meeting every Tuesday to debrief and review what we have accomplished, what we should do and what are some difficulties we are facing.
- Win-win negotiation; the negotiation helps us and the client to be on the same page and share the same understanding.
- IIV&V evaluations; Our DEN team member make sure that our artifacts look good and check if time is logged on Bugzilla and provide feedback about the bug/artifact on the ticket for the owner to correct.
- TA feedback; the comments from all TAs and graders are also taken into account.

4.2 Methods, Tools and Facilities

Tools	Usage	Provider
Github	Provide a version control system to store our team website files.	Github
Google drive	Contain all documents created by our team so that we can collaborate	Google
Visual paradigm	Provide a tool to create various UML diagrams used in the project	Visual paradigm
Balsamiq	The tool was used to develop the user interface prototype	Balsamiq
Microsoft office	The office was used to create docs, presentation and XML files.	Microsoft
Facebook	Provide a group feature to collaborate and make an appointment	Facebook

(Group/Chat)	among team members and client	
Join.me	The tool was used to share screen with DEN student	LogMeIn, Inc.
Skype	Video and/or Voice Chat	Microsoft
Bootstrap	to decorate our team website	Open source
Bugzilla	Provide a tool to create tasks, track task's statuses, and identify effort used in each task	USC
Winbook	Provide a tool to facilitate the team to identify the project's goals, win conditions of all stakeholders and prioritize them	USC

5. **Resources**

Estimated CSCI577a Effort: 8 team members working 10 hrs./week for 12 weeks.
Estimated CSCI577b Effort: 8 team members working 10 hrs./week for 12 weeks.

Estimated total effort: 10 hours per week * 6 months = 60 PM.

Programming languages: PHP, HTML/CSS, SQL, JavaScript.

Budget: assume that this project has no budget for development efforts. There might be a cost for hosting a server and acquire a domain name.

Table 8: Modules

No	Module	Description	SLOC (estimation)	REVL (estimation)
1	Point System	The point system that will provide users incentives to participate in the We Are Trojans network	1,000	15%
2	Forum Module	This module provides the students a platform to interact with each other	3,000	7%
3	Profile Module	The system that allow users to add and modify their information	800	10%
4	Store Module	The store used to redeem WAT points. First iteration will be limited to gift cards.	500	5%
5	Notification Module	The system that notify users with changes, such as liking, disliking, and a special campaign	800	5%

Table 9: COCOMOII Scale Driver

Scale Driver	Value	Rationale
PREC	High	Many examples exist with features similar to this project (i.e. Stackoverflow.com, Facebook)
FLEX	High	Project has a very high flexible set of requirements but the limited schedule pull down FLEX from very high to high.
RESL	Nominal	The project has thoroughness with architectural definition, but the system has a high tendency of risks.

| TEAM | High | Client's limited schedule availability could pose complexities |
| PMAT | Nominal | Relatively new team with limited experience working together under stressful deadlines |

Table 10: COCOMOII Cost Drivers for WAT point system

Cost Driver	Value	Rationale
RELY	Nominal	All other modules rely on this module. However, it is not risk to human life.
DATA	Low	We do not have much test data.
CPLX	Nominal	The analysis relating to the "WAT" Point system requires some numerical analysis that is prone to precision problems needed to be handled.
RUSE	Nominal	This WAT point module is developed specifically for this project. So it is only reusable across project (nominal)
DOCU	Nominal	The documentation was developed for each phase of development.
TIME	Nominal	The system is expected to use less than 50% of the available execution time.
STOR	Nominal	It will take up at most 20% use of available storage.
PVOL	Nominal	In this project, we do not update the change frequently.
ACAP	High	The team cohesively works on gathering requirements and high level design of the system.
PCAP	High	Most of our team members are capable. We have mix skills from 90th percentile to 55 percentile so we are going to estimate in between = 75 percentile.
PCON	Very High	There is no personnel turnover.
APEX	Nominal	The team is not fully familiar with the COTS/NDI that will be used in developing the system.
PLEX	Nominal	In the team, there are mixes of people familiar and unfamiliar with the platforms
LTEX	High	Every member of the team has used the language that will be used in development of this system.
TOOL	Nominal	Basic life-cycle tools, moderately integrated.
SITE	High	The clients and the DEN students are not present at the same site, but connection is achieved using Skype.
SCED	Nominal	The schedule is fixed for 12 weeks.

Table 11: COCOMOII Cost Drivers for Forum Module

Cost Driver	Value	Rationale
RELY	Nominal	This module is crucial for the system, but a failure would only result in moderate losses.
DATA	Nominal	In forum functionality, we may have some amount of data like images and texts for testing.
CPLX	Nominal	Development of this module will mostly involve modification of an existing COTS forum.
RUSE	Low	Forum will be tailored for this specific project, no reuse.
DOCU	Nominal	The documentation was developed for each phase of development.
TIME	Nominal	The system is expected to use less than 50% of the available execution time.
STOR	Very High	It will take up at < 80% use of avaliable storage
PVOL	Nominal	In this project, we do not update the change frequently.
ACAP	High	The team cohesively works on gathering requirements and high level design of the system.
PCAP	High	Most of our team members are capable. We have mix skills from 90th percentile to 55 percentile so we are going to estimate in between = 75 percentile.
PCON	Very High	There is no personnel turnover.
APEX	Nominal	The team is not fully familiar with the COTS/NDI that will be used in developing the system.
PLEX	Nominal	In the team, there are mixes of people familiar and unfamilar with the platforms
LTEX	High	Every member of the team has used the language that will be used in development of this system.
TOOL	Nominal	Basic life-cycle tools, modertaly integrated.
SITE	High	The clients and the DEN students are not present at the same site, but connection is achieved using Skype.
SCED	Nominal	The schedule is fixed for 12 weeks.

Table 12: COCOMOII Cost Drivers for Profile Module

Cost Driver	Value	Rationale
RELY	Nominal	moderate, easily recoverable losses
DATA	Low	The amount of test data will be relatively low compared to SLOC.

CPLX	Low	Straightforward nesting of structured. Simple GUI builder, straightforward database structure.
RUSE	Nominal	Reusable across project.
DOCU	Nominal	The documentation was developed for each phase of development.
TIME	Nominal	The system is expected to use less than 50% of the available execution time.
STOR	Nominal	It will take up at most 20% use of avaliable storage
PVOL	Nominal	In this project, we do not update the change frequently.
ACAP	High	The team cohesively works on gathering requirements and high level design of the system.
PCAP	High	Most of our team members are capable. We have mix skills from 90th percentile to 55 percentile so we are going to estimate in between = 75 percentile.
PCON	Very High	There is no personnel turnover.
APEX	Nominal	The team is not fully familiar with the COTS/NDI that will be used in developing the system.
PLEX	Nominal	In the team, there are mixes of people familiar and unfamilar with the platforms
LTEX	High	Every member of the team has used the language that will be used in development of this system.
TOOL	Nominal	Basic life-cycle tools, modertaly integrated.
SITE	High	The clients and the DEN students are not present at the same site, but connection is achieved using Skype.
SCED	Nominal	The schedule is fixed for 12 weeks.

Table 13: COCOMOII Cost Drivers for WAT Store Module

Cost Driver	Value	Rationale
RELY	Low	It does not affect the system too much
DATA	Low	We do not have much test data.
CPLX	Low	Straightforward nesting of structured. Simple GUI builder, straightforward database structure.
RUSE	Nominal	Reusable across project.
DOCU	Nominal	The documentation was developed for each phase of development.
TIME	Nominal	The system is expected to use less than 50% of the available execution time.
STOR	Nominal	It will take up at most 40% use of avaliable storage

PVOL	Nominal	In this project, we do not update the change frequently.
ACAP	High	The team cohesively works on gathering requirements and high level design of the system.
PCAP	High	Most of our team members are capable. We have mix skills from 90th percentile to 55 percentile so we are going to estimate in between = 75 percentile.
PCON	Very High	There is no personnel turnover.
APEX	Nominal	The team is not fully familiar with the COTS/NDI that will be used in developing the system.
PLEX	Nominal	In the team, there are mixes of people familiar and unfamilar with the platforms
LTEX	High	Every member of the team has used the language that will be used in development of this system.
TOOL	Nominal	Basic life-cycle tools, modertaly integrated.
SITE	High	The clients and the DEN students are not present at the same site, but connection is achieved using Skype.
SCED	Nominal	The schedule is fixed for 12 weeks.

Table 14: COCOMOII Cost Drivers for Notification Module

Cost Driver	Value	Rationale
RELY	Low	No other module relies on this. Failure will only cause slight inconvenience.
DATA	Low	We do not have much test data.
CPLX	Nominal	Mostly simple nesting. Some intermodule control.
RUSE	Nominal	Reuse across project.
DOCU	Nominal	The documentation was developed for each phase of development.
TIME	Nominal	The system is expected to use less than 50% of the available execution time.
STOR	Nominal	It will take up at most 20% use of avaliable storage
PVOL	Nominal	In this project, we do not update the change frequently.
ACAP	High	The team cohesively works on gathering requirements and high level design of the system.
PCAP	High	Most of our team members are capable. We have mix skills from 90th percentile to 55 percentile so we are going to estimate in between = 75 percentile.
PCON	Very High	There is no personnel turnover.

APEX	Nominal	The team is not fully familiar with the COTS/NDI that will be used in developing the system.
PLEX	Nominal	In the team, there are mixes of people familiar and unfamilar with the platforms
LTEX	High	Every member of the team has used the language that will be used in development of this system.
TOOL	Nominal	Basic life-cycle tools, modertaly integrated.
SITE	High	The clients and the DEN students are not present at the same site, but connection is achieved using Skype.
SCED	Nominal	The schedule is fixed for 12 weeks.

System's Overview — COPSEMO

Scale Factors 15.62   Schedule (SCED)

⊕ Subcomponent   ⊖ Subcomponent(s)   More Actions ... ▼

X	Name	Size	Labor Rate ($/Month)	EAF	Language	NOM Effort DEV	EST Effort DEV	PROD	COST	INST COST	Staff	Risk
☐	Forum Module	3,210	0.0	0.57	Non-s...	10.69	6.09	52...	0.00	0.00	0.8	0.0
☐	Point System	1,150	0.0	0.46	Non-s...	3.83	1.77	65...	0.00	0.00	0.2	0.0
☐	Notificatio...	840	0.0	0.42	Non-s...	2.80	1.19	70...	0.00	0.00	0.2	0.0
☐	Profile Module	880	0.0	0.40	Non-s...	2.93	1.18	74...	0.00	0.00	0.2	0.0
☐	Store Module	525	0.0	0.37	Non-s...	1.75	0.65	81...	0.00	0.00	0.1	0.0

Estimation

Total Lines Of Code: 6,605   Hours/PM: 152.0

Estimated	Effort	Schedule	PROD	COST	INST	Staff	Risk
Optimistic	8.70	7.20	759.29	0.00	0.00	1.2	
Most Likely	10.87	7.71	607.43	0.00	0.00	1.4	0.0
Pessimistic	13.59	8.27	485.94	0.00	0.00	1.6	

Total number of week = 24 weeks (2 semesters).
One person can work 10 hours per week.
According to COINCOMO II;

    The COCOMO II estimation effort calculated from the 5 modules gives an effort of **11.07 PM**

    13.59 PM * 152 hrs/PM = 2065.68 hr to do the work
    10 hrs/week/person * 24 weeks = 240 hr/person
    2065.68 hrs / 240 hrs / person = 8.6 person

We have 8 team members, so according to the pessimistic estimate we will be able to finish the project within the time constraint if we work a little extra.

6. Iteration Plan

6.1 Plan

The construction iteration of the Development phase will be divided into two cycles; one where the core capabilities of the system are developed and one for completing the full functionality of the system. This is done to make sure the most important features are ready before the Core Capability Drivethrough, and also because the core features must be in place before extra functionality can be developed. Features developed in the first cycle include the Forum, Point System and Notification System modules.

Since no legacy system exists, neither in terms of software nor hardware, the Transition phase will mainly focus on training the clients on how to use and deploy the system.

6.1.1 Capabilities to be implemented

Table 15: Construction iteration capabilities to be implemented

ID	Capability	Description	Priority	Iteration
1	OC-1	Q&A Forum	High	1
2	OC-2	WAT Point System	High	1
3	OC-3	Notification System	High	1
4	OC-4	Profile	Medium	1
5	OC-5	Leaderboard	Medium	2
6	OC-6	Redemption	Medium	2
7	OC-7	Event System	Low	2

6.1.2 Capabilities to be tested

Table 16: Construction iteration capabilities to be tested

ID	Capability	Description	Priority	Iteration
1	OC-1	Q&A Forum	High	1
2	OC-2	WAT Point System	High	1
3	OC-3	Notification System	High	1
4	OC-4	Profile	Medium	1
5	OC-5	Leaderboard	Medium	2
6	OC-6	Redemption	Medium	2
7	OC-7	Event System	Low	2

6.1.3 Capabilities not to be tested

All capabilities will be tested.

6.1.4 CCD Preparation Plans

Our clients, Matthew and Linda, will be involved in the CCD. The development team will prepare a system to be tested by the clients. The clients will be asked to perform certain tasks in the system, such as create a thread or search for a specific topic, based on what is defined in our test plan. The database will be pre-populated so that the search functionality can be properly tested and the forum can be navigated. The results, client experience and suggested improvements will be documented and followed-up by the team.

6.2 Iteration Assessment

6.2.1 Capabilities Implemented, Tested, and Results

<< Describes, in brief, the capabilities that were implemented and the test results. The capabilities implemented and tested do not necessarily need to match the ones listed in section 6.1 because some capabilities may have been pushed to the next iteration. >>

Table 17: Capabilities implemented, tested, and results

ID	Capability	Test Case	Test Results	If fail, why?
< ID >	< Capability >	< TC-XX >	Pass/Fail	< comments >
…				

6.2.2 Core Capabilities Drive-Through Results

Create account and log in
The client was overall pleased with the interface, but suggested some changes in the layout of the forms to make it look more like other webpages. The system worked as expected, but the error messages left the client confused and these should be changed.

Update profile and select favorite categories
The client generally felt that the design and layout of this page could be improved. There was a lack of a clear design style, and the elements on the page did not go well together. The profile pages also had some technical problems. One notable problem was when selecting favorite categories: only one category could be selected at a time which made the process cumbersome. She also suggested that the profile information should be made more compact, and that we list the user's recent activity (threads and posts) below it. The uploading of profile and cover picture worked as expected, but the cropping of the image was difficult because the image was in full zoom, and the client had to scroll over the picture to get it right. Changing the password was also

a bit confusing as the client had to click the "save" button twice. The client suggested we make this more user friendly by having a popup message confirming that the password was changed.

List threads and see specific threads
The client was very happy with the new design of the forum pages, and she said it now looked more professional than before. Also, the team has made a consistent color theme across all pages that the client was happy with. The client was unsure how threads were sorted. The leaderboard took a lot of space and attention, and the client requested that we make it more simplistic. For some pages there was no "back"-button, so the client suggested that we add one which operates like the browser's "back"-button as this would improve the navigation.

Write and edit threads and posts
The client thought that going to a different page for writing threads was annoying and suggested that this should be a popup on the current page the user is on. This was also true for editing posts. The client also asked if we could include a spellchecker.

Notifications
The client liked the Facebook-like popup notifications and had no particular comments about it.

Search
The client liked the drop-down search results that appeared as she was typing. Otherwise it worked as expected, no additional comments. There was a risk that the search would be difficult to implement, but we were able to integrate an existing search engine to our system that seems to work well.

6.3 Adherence to Plan

<< Describe how well the iteration ran according to plan. Was it on budget and on time? Is there any uncertainty in the Software Development Status? Provide some insight to avoid mistakes for future iterations. >>