Life Cycle Plan (LCP)

SnApp - Voice Communication System

Team 05

Divij Durve - Integrated Independent Verification and Validation
Harsh Mhatre - System/Software Architect
Khyati Thakur - Prototyper
Monica Varhale - Feasibility Analyst
Nikita Gupta - Project Manager
Shlok Naik - Operational Concept Manager
Shruti Gotmare - Life Cycle Planner
Sushmaja Bondili - Requirements Engineer

10/13/14

Version History

Date	Author	Version	Changes made	Rationale
09/29/14	DD,HM,KT, MV,NG,SN, SG,SB	1.0	Section 3 filled	Initial draft to be included in the Valuation Commitment Package
10/13/14	MV,SG,NG	2.0	Filled in section 1- 6	To be included with the Foundations Commitment Package
10/19/14	MV, SG	3.0	Made changes to section 2 and 5	Fixed the defects suggested during the FCR and made changes to be included in the DCP

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the LCP

The purpose of this LCP is to provide a detailed description of the process to be followed throughout the project's life cycle. The LCP will help our team to plan resources so that team members are mapped to the roles in accordance with their skills in order to make the project successful in the given time period.

Life cycle plan will also help our team in analyzing the potential issues in the project so that we are aware of the potential risks in the project. We can also identify risk mitigation strategies for potential risks that will help our team reduce the risk exposure in the future.

1.2 Status of the LCP

The status of the LCP is currently at the development commitment package (version 3.0).   In the previous version, we had done the COCOMO estimation as well as completed the document till section 6.1.We had a Foundations commitment review and in this document, we are incorporating the changes that came up during the foundations commitment review and also the sections which need to be completed for the Development Commitment Package.

1.3 Assumptions

The life cycle of our project assumes that:
- The 12 weeks available for our project in Fall 2014 are sufficient for requirements analysis, design, development, rigorous testing and training.
- The project will be developed following the guidelines from the ICSM – Electronic Process Guide.
- Our team is functioning as an architected agile team that is focused on getting the right architecture for the project with extensive use of Twilio, a net-centric service.
- The 8 member on the team are capable of completing the required tasks in the estimated time frame.

2. Milestones and Products

2.1 Overall Strategy

Our project SnApp VCS follows the combination of Architectured Agile methodology and Net-Centric Services (NCS). Twilio API (a NCS) provides us most of the features we need for our project. As most architected agile teams, we are a small group focusing on incremental development of the project.

Exploration Phase:
Duration: 09/15/2014 - 09/19/2014
Concept: In this phase, the team worked on understanding the project requirements by having several meetings with the client and clearing the ambiguities in any of the requirements. The client also provided us with usage scenarios that gave us a clear understanding of the business workflow. The developers in our team analyzed the current system and its drawbacks.
Deliverables: Valuation Commitment Package, Client Interaction Report, Bi-weekly Project Plan and Progress Report, Weekly Effort Report.
Milestone: Valuation Commitment Review
Strategy: One Incremental Commitment Cycle

Valuation Phase
Duration: 09/22/2014 - 10/03/2014
Concept: In this phase, the team identified the objectives and prioritized the requirements, developed the operational concept, prototyped high risk factors to provide feasibility evidence for the project. The team also assessed the risks in our project and developed plans to mitigate them. We also developed the project plan to help us plan and manage project.
Deliverables: Foundation Commitment Package, Initial Prototype, Bi-weekly Project Plan and Progress Report, Weekly Effort Report, Feasibility Evidence, Life Cycle Plan, Operation Concept Description, System and Software Architecture Description.
Milestone: Foundation Commitment Review
Strategy: One Incremental Commitment Cycle

Foundations Phase
Duration: 10/06/2014 - 10/15/2014
Concept: In this phase, the team is continuing the risk assessment process and prototyping the project. We plan to build mockups to explore the functionality and better plan our implementation work. The team will be architecting and designing the project and performing all UML diagramming.
Deliverables: Foundation Commitment Package, Initial Prototype, Bi-weekly Project Plan and Progress Report, Weekly Effort Report, Feasibility Evidence, Life Cycle Plan, Operation Concept Description, System and Software Architecture Description.
Milestone: Foundation Commitment Review
Strategy: One Incremental Commitment Cycle

Development phase
Duration: 10/16/2014 – 11/14/2014
Concept: Start with the mockups of the core capabilities agreed in the winbook. In this phase, we will be segregating the development of different components exhibiting different core capabilities amongst the team of developers and we will be employing pair programming methodology for development. The risk assessment would be continued in this phase and also we will be having weekly meeting with our client and amongst the team members to keep the client informed about the status of development and also for any technical assistance in the development. Also, there will be biweekly progress report and project plan submitted on the team website.
Deliverables: System consisting of all core capabilities, Bi-weekly Project Plan and Progress Report, Weekly Effort Report.
Milestone: Core Capability Drivethrough

2.2 Project Deliverables

2.2.1 Exploration Phase

Table 1: Artifacts Deliverables in Exploration Phase

Artifact	Due date	Format	Medium
Client Interaction Report	9/19/2014	.pdf	Soft copy
Valuation Commitment Package • Life Cycle Plan (LCP) Early Section • Feasibility Evidence Description (FED) Early Section	09/28/2014	.doc, .pdf	Soft copy
Project Effort	Every Monday	Text	Bugzilla report
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xlsx	Soft copy

2.2.2   Valuation Phase

Table 2: Artifact deliverable in Valuation Phase

Artifact	Due date	Format	Medium
Operational Concept Description	10/13/2014	.doc, .pdf	Soft copy
Life Cycle Plan	10/13/2014	.doc, .pdf	Soft copy
System and Software Architecture Description	10/13/2014	.doc, .pdf	Soft copy
Feasibility Evidence	10/13/2014	.doc, .pdf	Soft copy
Prototype Report	10/03/2014	.pptx	Soft copy
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xlsx	Soft copy
Project Effort	Every Monday	Text	Bugzilla website

2.2.3   Foundations Phase

Table 3: Artifact deliverable in Foundations Phase

Artifact	Due date	Format	Medium
Operational Concept Description	10/20/2014	.doc, .pdf	Soft copy
Life Cycle Plan	10/20/2014	.doc, .pdf	Soft copy
System and Software Architecture Description	10/20/2014	.doc, .pdf	Soft copy
Feasibility Evidence	10/20/2014	.doc, .pdf	Soft copy
Prototype Report	10/20/2014	.pptx	Soft copy
Quality Plan	10/15/2014	ARB	Presentation
Traceability Matrix	10/15/2014	ARB	Presentation
Test Plan and Test cases	10/15/2014	.doc,.pdf	Soft copy
Project Plan	Every other Wednesday	.mpp	Soft copy
Progress Report	Every other Wednesday	.xlsx	Soft copy
Project Effort	Every Monday	Text	Bugzilla website

2.2.4   Development Phase

Table 2: Artifacts deliverable in Development Phase

Artifact	Due date	Format	Medium
Draft System with core capabilities	11/14/14	.html	Soft Copy, Demo
Project Effort	Every Monday	Text	Bugzilla website
Project Plan	Every other Wednesday	.mpp, .pdf	Soft copy
Progress Report	Every other Wednesday	.xls	Soft copy

3. Responsibilities

3.1 Project-specific stakeholder's responsibilities

The client is Charles Zivko, Co-founder of SnaApp Dev. The users of the system are sales agents, managers and company executives. The developer is Team #05. Mr. Charles Zivko is project-specific stakeholder and his role is to supply existing project assets to developers and also the detailed requirements of the system and he is also the maintainer of the project as he would be responsible foe maintaining the system after it is deployed on Heroku.

3.2 Responsibilities by Phase

Table 5: Stakeholder's Responsibilities in each phase

Team Member and Roles	Primary / Secondary Responsibility			
	Exploration	**Valuation**	**Foundations/Development-** Construction Iteration	**Development-** Transition Iteration
Charles Zivko. *Client*	**Primary Responsibility:** 1. Explain the scope of the project to the team 2. Attend and participate in the win win negotiation sessions 3. Attend the team meeting and resolve the team issues	**Primary responsibility** 1. Interact with the team and analyze their work 2. Provide suggestions and feedback to the team	**Primary Responsibility:** 1. Assist the developers with all the technical details and provide the Level of service for the system. 2. Provide feedback based on the current system. 3. Provide changed requirements.	**Primary Responsibility:** 1. Provide feedback based on the current system. 2. Provide changed requirements. 3. Receive training for the new system. 4. Maintain the system.
Nikita Gupta. *Project Manager, Operational concept manager*	**Primary Responsibility** 1. Plan the overall project 2. Identifying the risks involved in the project	**Primary responsibility:** 1. Create a detailed project plan 2. Interact with the client and the team to resolve issues 3. Record the	**Primary Responsibility:** 1. Plan the project 2. Assess risks 3. Distribute work, give specific task to each team member 4. Update bugzilla	**Primary Responsibility:** 1. Assess Metrics 2. Co-ordinate Meetings 3. Plan test cases and perform testing. **Secondary**

	3. Monitoring and updating the bugzilla **Secondary Responsibility:** 1. Conceptualize the system	projects progress 4. Update and monitor bugzilla **Secondary responsibility:** 1. Establish new operational concept 2. Identify the shared vision 3. Establish operational and organizational transform	**Secondary responsibility:** 1. Interact with the client	**responsibility:** 1. Perform activities required for transition
Sushmaja Bondili **Requirements Engineer, Prototyper**	**Primary Responsibility:** 1. Facilitate the win-win negotiations 2. Monitor and update the winbook **Secondary responsibility:** 1. Study the system features in order to understand the system functionality and requirements	**Primary responsibility:** 1. Prioritize and score the MMF's and win conditions 2. Capture progress of win conditions **Secondary responsibility:** 1. Analyze the system requirements 2. Develop the prototype for the system	**Primary responsibility:** 1. Plan test cases 2. Manage the quality of the project **Secondary responsibility:** 1. Develop mockups for the remaining clients requirements	**Primary responsibility:** 1. Develop the different subcomponents of the system. **Secondary Responsibility:** 1. Prototype the high risk functionalities to ensure they are feasible.
Monica Varhale **Feasibility Analysts, Requirements engineer**	**Primary responsibility:** 1. Identify the risks in the project plan 2. Mitigate the risks **Secondary responsibility:** 1. facilitate win win negotiations	**Primary responsibility:** 1. Analyze the feasibility of NDI and NCS components 2. Analyze the business case **Secondary responsibility:** 1. Capture MMF's and win conditions 2. Capture the progress of win	**Primary responsibility:** 1. Mitigate risk **Secondary responsibility:** 1. Identify system and software requirements definition	**Primary responsibility:** 1. Develop the different sub-components of the system. **Secondary Responsibility:** 1. Ensure that the development is in compliance with the requirements.

			conditions	
Khyati thakur **Prototyper, Project Manager**	**Primary responsibility:** 1. Identify various modules of the system 2. Develop the initial prototype for the system **Secondary responsibility:** 1. Plan the project's schedule 2. Interact with the client	**Primary responsibility:** 1. Analyze the different capabilities of the system to be prototyped 2. Identify the suitable prototyping tool 3. Develop the prototype **Secondary responsibility:** 1. Check the progress of the team and resolve issues 2. Record team's progress	**Primary responsibility:** 1. Analyze and prioritize capabilities to prototype 2. Develop Prototype 3. Access prototype and components **Secondary responsibility:** 1. Plan the project 2. manage client interaction	**Primary responsibility:** 1. Develop the different subcomponents of the system. **Secondary Responsibility:** 1. Interact with the client and the team to resolve issues 2. Record the projects progress
Harsh Mhatre **System/ Software Architect**	**Primary responsibility:** 1. Explore the current system 2. Identify NDI/NCS for the system	**Primary responsibility:** 1. Assess the NDI and NCS component candidates	**Primary responsibility:** 1. Develop the system architecture	**Primary responsibility:** 1. Develop the different subcomponents of the system. **Secondary Responsibility:** 1. Ensure that the development of the system is in compliance with the system architecture and all the use cases are implemented in the system.
Shlok Naik **Operational Concept Manager, Feasibility Analyst**	**Primary responsibility:** 1. Analyze the current system and conceptualize it. 2. Review	**Primary responsibility:** 1. Establish new operational concept 2. Identify the shared vision	**Primary responsibility:** 1. Create operational concept description 2. Assess operational concept	**Primary responsibility:** 1. Plan test cases and perform testing. **Secondary Responsibility:** 1. Assist with the

	deliverables and work products and provide their evaluation **Secondary Responsibility:** 1. Identify the risks involved in the project snapp	3. Identify organizational and operational transform **Secondary responsibility:** 1. Analyze business case 2. Assess and evaluate NDI and NCS components candidates	**Secondary responsibility:** 1. Identify test plan and procedure	development of the system.
Shruti Gotmare *Life Cycle Planner, Requirements engineer*	**Primary responsibility:** 1. Plan the various phases of the projects life cycle. 2. List the various deliverables for the project 3. Identify the teams skillset and responsibilities **Secondary responsibility:** 1. Identify the risks 2. Mitigate the risks	**Primary responsibility:** 1. Identify the responsibilities and prepare a life cycle plan document **Secondary responsibility:** 1. Capture and score MMF's and win conditions 2. Capture progress of win-win negotiations	**Primary responsibility:** 1. Record Project progress 2. Modify detailed project plan **Secondary responsibility:** 1. Mitigate risk	**Primary responsibility:** 1. Develop the system 2. Update the life cycle plan document **Secondary responsibility:** 1. Maintain the project website
Divij Durve *IIV&V*	**Primary responsibility:** 1. Monitor bugzilla	**Primary responsibility:** 1. Monitor bugzilla and report bugs 2. Verify and validate work products	**Primary responsibility:** 1. Review the project artifacts 2. Manage Project Quality	**Primary responsibility:** 1. Provide evaluation of work products, Testing at higher level i.e. acceptance testing, 2. Maintain Bugzilla.

3.3 Skills

Table 6: Skills of team members

Team members	Role	Skills
Divij Durve	Integrated Independent Verification & Validation	Current skills: Android SDK, Java, MySQL, Oracle db, PostgreSQL,HTML, CSS, jQuery, Spring, Architecture design & Discussion Required skills: ability to objectively judge requirements without taking sides.
Harsh Mhatre	System/Software Architect and Developer	Current Skills: UML Designing, Web Technologies, DBMS & SQL, Java, Ruby Required Skills: PostgreSQL
Khyati Thakur	Prototyper and Developer	Current Skills: Web technologies, Axure RP Required Skills: Ruby on Rails, PostgreSQL
Monica Varhale	Feasibility Analyst and Developer	Current Skills: COCOMO, Analysis skills, JAVA, Web technologies, MySql, Python. Required Skills: Ruby on Rails, PostgreSQL, Gitlab.
Nikita Gupta	Project Manager and Tester	Current Skills: Software & project management, C++, Web technologies, Python, SQL (PostgreSQL/MySQL) Required Skills: Ruby on Rails
Shlok Naik	Operational Concept Manager and Tester	Current Skills: Web Technologies, JAVA, UML Design Required Skills: Ruby On Rails, PostgreSQL,Gitlab
Shruti Gotmare	Life Cycle Planner and QA	Current Skills: C#, VBScript, Quality management, Git, Project planning skills Required Skills: Ruby on Rails, PostgreSQL
Sushmaja Bondili	Requirements Engineer and Developer	Current Skills: C, JAVA, jQuery, HTML, CSS, JavScript, MySQL, Linux, Communication Skills Required Skills: Ruby on Rails, PostgreSQL

4. Approach

4.1 Monitoring and Control

The following are the various tools and documentation used by Team 05 to assist in the monitoring and control of the project:
1. Bi-weekly project report
2. Bi-weekly project plan
3. Weekly team meetings for planning and review
4. Weekly client meetings for feedback
5. Reporting on and monitoring Bugzilla
6. Gitlab to manage the workspace for our project

4.1.1 Closed Loop Feedback Control

All deliverables are hosted on Gitlab (for code) and Google Drive (for documents and planning) while the team members are working on them. This allows for all team members to access and review the work that is being done. This keeps everyone in the loop about everyone's progress and allows everyone to provide input on the work being done.

4.1.2 Reviews

- The project manager team member reviews all documents before submitting the document to the database of the project.
- All the documents are shared on the Google Drive so that at any time, any team member to access and provide input on any document.
- We also have weekly meetings with our client where the client reviews our team's work and suggests improvements and changes if necessary.
- Commitment review that is held at each milestone
- Bi-weekly project plan and project report review status of the project.
- Bugzilla is used to review all the bugs and tasks in the project
- Weekly team meeting

4.2 Methods, Tools and Facilities

Table 7: Methods, Tools and Facilities

Tools	Usage	Provider
MS Project	Used to plan the life cycle of the project and assess and mitigates risks in the system	USC License
Sublime text	Robust text editor useful for coding in Ruby on Rails	Open Source
Heroku cloud	Cloud service provider	Open Source
MS Office	Used for documenting deliverables, LCP, OCD, FED	Open Source
Balsamiq	Useful in developing the prototype of the project	Demo Version
GitHub	Helpful in managing versions of the project	Client
Postgresql	Used as the database for the project	Open Source

5. **Resources**

To estimate the software cost:
- Estimated CSCI577a Effort: 8 team members at 8hrs/week for 12 weeks
- Total estimated effort: 800 hours approx.
- Budget information: $100
- Project duration: 12 weeks (Fall 2014)
- Component modules in your development project: Call Monitoring (Call Recording, Listening, Conferencing), Auto-dialer, Incoming call handling, Direct dialer, Auto-provisioning
- Programming language used: Ruby on Rails, HTML, CSS, JavaScript

Table 8: Module lists and SLOC of each module

No.	Module Name	Brief Description	SLOC	REVL
2	Call Listening	Enables the manager listen in on an ongoing conversation in mute mode	800	0%
3	Call Recording	Records all calls and store them	250	0%
4	Dialer	Enables the sales agent and mangers to make direct calls and autodial calls to the customers or leads	250	0%
5	Incoming Call Handling	Handles the incoming calls of the customers	800	0%
6	Call Logger	Logs all the call detail information for business analytics	800	0%

Figure1. Scale factors

Table 9: COCOMOII Scale factors

Scale Driver	Value	Rationale
PREC	NOMINAL	The project is not much similar but fairly similar with respect to the functionalities it provides and we can refer to those modules and incorporate some of these into our current system.
FLEX	NOMINAL	The need for software to conform to the pre-established requirements is not much as out client Charles from SnAppDev is willing to accept different approaches and changes in the requirements that the team members suggest during the client meetings
RESL	HIGH	Our team has identified all the critical risk items and also have risk

		mitigation plans for these risk items. Also, we have a detailed understanding of the system architecture and have dedicated two team members who are skilled in design for the architecture design.
TEAM	HIGH	All the team members and the client accommodate each other's objectives and we as a team work together to achieve the shared vision and we have a high team cohesion.
PMAT	NOM CMM Level 2	ICSM Principles and Guidelines are followed strictly

Table 10: COCOMOII Cost Driver for Call listening

Cost Driver	Value	Rationale
RELY	NOM	If the call listening module fails it will be inconvenience to the managers of SnAppDev, the system has a maintainer to reduce the possibility of failure
DATA	LOW	The testing database for this module is small compared to the SLOC as we have only a few managers.
DOCU	NOM	Nominal Documentation is required for future maintenance.
CPLX	NOM	No complex numerical analysis or calculations needed
RUSE	NOM	This module can be reused across project
TIME	NOM	The system consumes nominal time and resources
STOR	NOM	This component require nominal storage in database
PVOL	LOW	The platforms are stable and do not require frequent upgrades

ACAP	NOM	Analysts in our team have basic experience with architecture design
PCAP	HI	All the team members have experience with software development and our team programmers showcase good communication and co-operation within the team
PCON	HI	All team members will be available throughout the project
APEX	LOW	The team members don't have experience with such applications
LTEX	VLOW	Very low programming language and software tool experience
PLEX	NOM	Few team members have high platform experience
TOOL	NOM	We have basic life cycle tools and also need integration with the Twilio API.
SITE	HIGH	The entire team is located in the same city except for the IIV & V member who is working remotely with us.

Figure 2: COCOMOII Cost Drivers for Call Listening

Table 11: COCOMOII Cost Driver for Call logger

Cost Driver	Value	Rationale
RELY	VHI	If the call logger module fails it, the SnApp cannot perform analytics which is one of the primany purpose of their project.
DATA	VHI	The testing database for this module is high as all the call information is going to be logged for the purpose of analytics
DOCU	NOM	Nominal Documentation is required for future maintenance.
CPLX	NOM	No complex numerical analysis or calculations needed

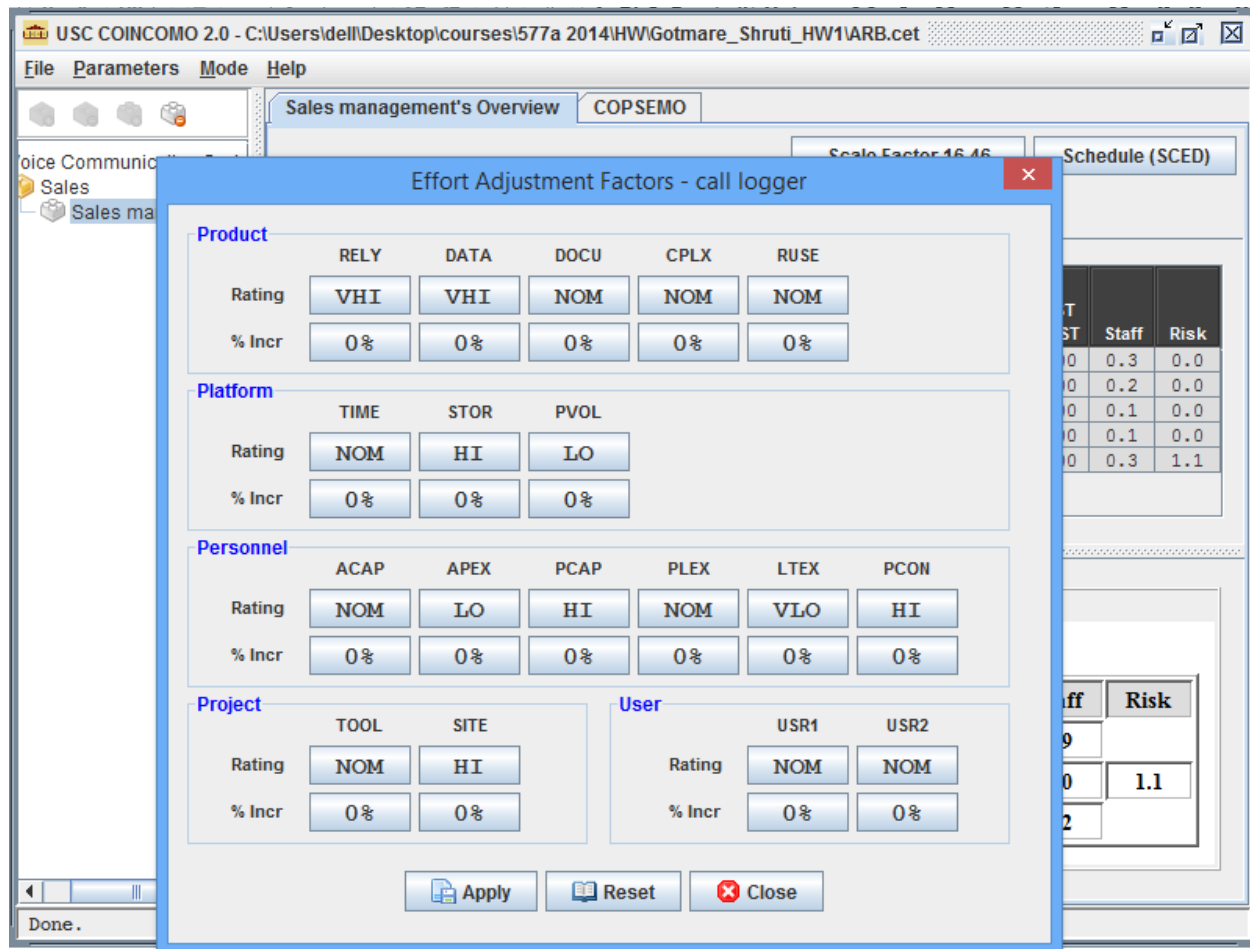| RUSE	NOM	This module can be reused across project
TIME	NOM	This module consumes nominal time and resources
STOR	VHI	This component require high storage in database as all the call information is going to be logged
PVOL	LOW	The platforms are stable and do not require frequent upgrades
ACAP	NOM	Analysts in our team have basic experience with architecture design
PCAP	HI	All the team members have experience with software development and our team programmers showcase good communication and co-operation within the team
PCON	HI	All team members will be available throughout the project
APEX	LOW	The team members don't have experience with such applications
LTEX	VLOW	Very low programming language and software tool experience
PLEX	NOM	Few team members have high platform experience
TOOL	NOM	We have basic life cycle tools and also need integration with the Twilio API.
SITE	HIGH	The entire team is located in the same city except for the IIV & V member who is working remotely with us.

Figure 3: COCOMOII Cost Drivers for Call Logger

Table 12: COCOMOII Cost Driver for Call Recording module

Cost Driver	Value	Rationale
RELY	HIGH	If this module fails, then the managers will not be able to listen to calls and also the sales agents couldn't be trained
DATA	NOM	The testing database for this module is nominal compared to the SLOC as we have 10 sales agents and only a few managers but the number of leads and the customers are more.
DOCU	NOM	Nominal Documentation is required for future maintenance.

CPLX	NOM	No complex numerical analysis or calculations needed.
RUSE	NOM	This module can be reused accross project
TIME	NOM	The module consumes nominal time and resources
STOR	NOM	This component requires nominal capacity as call recording link is stored in database.
PVOL	LOW	The platforms are stable and do not require frequent upgrades
ACAP	NOM	Analysts in our team have basic experience with the architecture design
PCAP	HIGH	All the team members have experience with Software development and our programmers showcase good communication and cooperation within the team
PCON	HGH	All team members will be available throughout the project
APEX	LOW	The team does not have much experience in developing such module in this domain
LTEX	VLOW	New technology to work on i.e Ruby on Rails and except one team member, everyone is working on this for the first time
PLEX	NOM	Most of the team members have nominal experience on working on UNIX platform and other platforms in this project.
TOOL	NOM	We have basic life cycle tools and also need integration with the Twilio API.
SITE	HIGH	The entire team is located in the same city except for the IIV & V member who is working remotely with us.

Figure 4: COCOMOII Cost Drivers for Call Recording

Table 13: COCOMOII Cost Driver for Incoming Call Handling module

Cost Driver	Value	Rationale
RELY	NOM	If this module fails, the sales agent can make call and it would not affect much.
DATA	NOM	The testing database for this module is nominal compared to the SLOC as we have 10 sales agents and only a few managers but the number of leads and the customers are more.
DOCU	NOM	Nominal Documentation is required for future maintenance.
CPLX	HIGH	We have some User interface management operations which includes simple voice I/O.

RUSE	NOM	This module can be integrated into other modules of the project.
TIME	NOM	The module consumes nominal time and resources
STOR	NOM	This component requires storage of the voicemail.
PVOL	LOW	The platforms are stable and do not require frequent upgrades
ACAP	NOM	Analysts in our team have basic experience with the architecture design
PCAP	HIGH	All the team members have experience with Software development and our programmers showcase good communication and cooperation within the team
PCON	HGH	All team members will be available throughout the project
APEX	LOW	The team does not have much experience in developing such module in this domain
LTEX	VLOW	New technology to work on i.e Ruby on Rails and except one team member, everyone is working on this for the first time
PLEX	NOM	Most of the team members have nominal experience on working on UNIX platform and other platforms in this project.
TOOL	NOM	We have basic life cycle tools and also need integration with the Twilio API.
SITE	HIGH	The entire team is located in the same city except for the IIV & V member who is working remotely with us.

Figure 5: COCOMOII Cost Drivers for Incoming Call Handling

Table 14: COCOMOII Cost Driver for Dialer

Cost Driver	Value	Rationale
RELY	HIGH	It would be very inconvenient for sales agents and managers as they wont be able to contact individual customers and would adversely affect their customer relationship.
DATA	LOW	The testing database for this module is nominal compared to the SLOC as we have 10 sales agents and only a few managers.
DOCU	NOM	Nominal Documentation is required for future maintenance.

CPLX	NOM	No complex numerical analysis or calculations needed.
RUSE	NOM	This module can be integrated into other modules of the project.
TIME	NOM	The module consumes nominal time and resources
STOR	NOM	Nominal storage requirements are required for this module
PVOL	LOW	The platforms are stable and do not require frequent upgrades
ACAP	NOM	Analysts in our team have basic experience with the architecture design
PCAP	HIGH	All the team members have experience with Software development and our programmers showcase good communication and cooperation within the team
PCON	HGH	All team members will be available throughout the project
APEX	LOW	The team does not have much experience in developing such module in this domain
LTEX	VLOW	New technology to work on i.e Ruby on Rails and except one team member, everyone is working on this for the first time
PLEX	NOM	Most of the team members have nominal experience on working on UNIX platform and other platforms in this project.
TOOL	NOM	We have basic life cycle tools and also need integration with the Twilio API.
SITE	HIGH	The entire team is located in the same city except for the IIV & V member who is working remotely with us.

Figure 6: COCOMOII Cost Drivers for Dialer
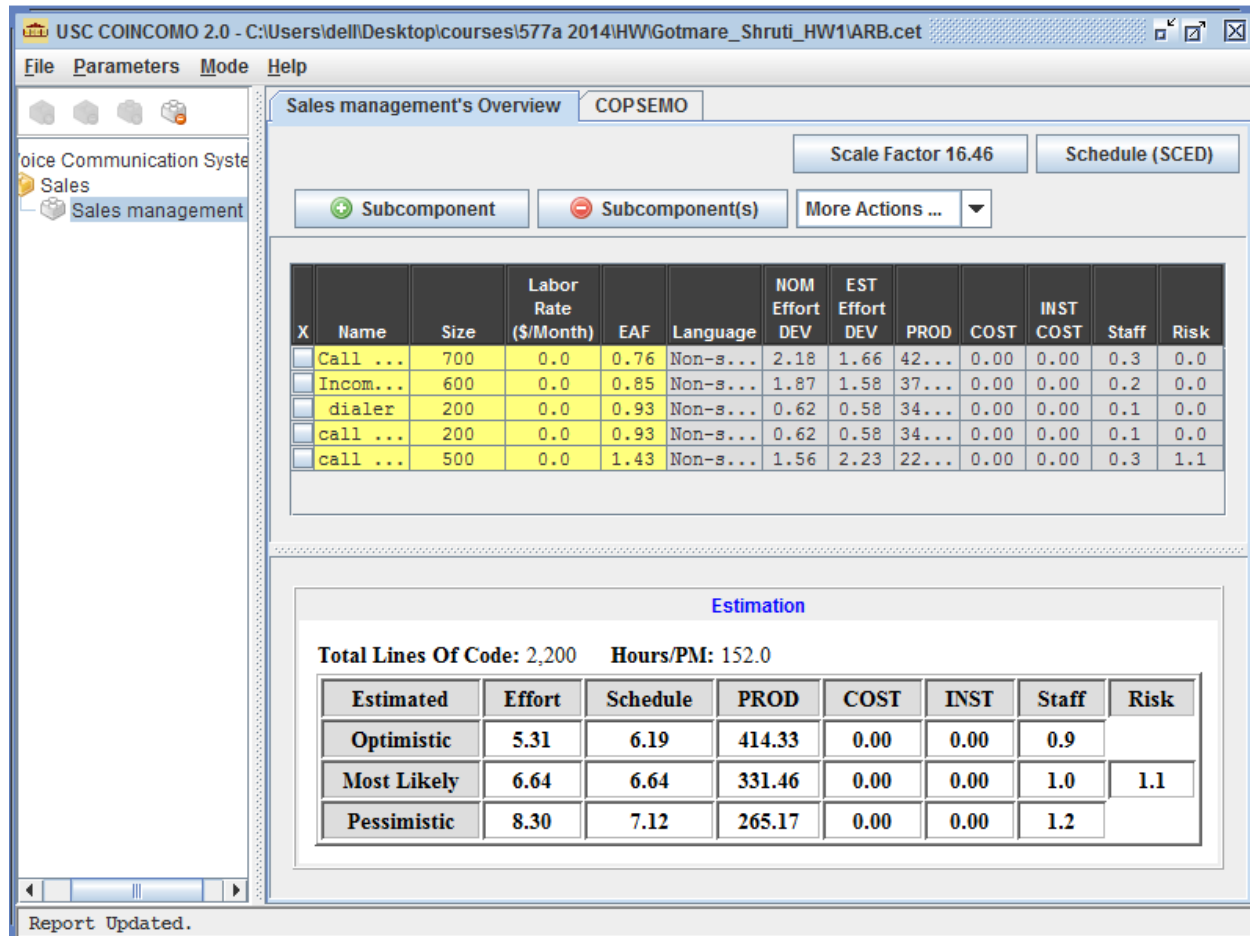
Figure 7: Final Estimation

The COINCOMO results show that the project can be completed in the estimated time frame of 6.64 months by 1.0 members in most likely case. However, our project schedule is 12 weeks and the number of team members are 8. Since all the team members will be working on the project throughout the semester, the project seems feasible in 12 weeks by a team of 8 members.

6. Iteration Plan

6.1 Plan

The following iterations are planned for the Core Capability Drivethrough. During the development phase, the team of developers will be developing the project from scratch from setting up the database for the users of the system using PostgreSQL and developing the different components of the system that we have agreed upon like the call conferencing, call listening, call logger, call recording, incoming call handling, direct calling and autodialing. We will also be devoting some time to integration testing and system testing at the end of the development phase and would be performing unit testing and some integration testing during the development phase based on the test cases described in the TPC document.

6.1.1 Capabilities to be implemented

Table 15. Construction iteration capabilities to be implemented

ID	Capability	Description	Priority Level	Iteration
OC-1	Log call detail information	Logs all the calling information for business analytics	Must Have	1
OC-2	Direct calling	Enables the sales agents, managers to make direct calls to the customers or leads	Must Have	1
OC-3	Call conferencing	Enables the manager to conference on a current call	High	2
OC-4	Provide voice-mailbox	Providing voice mailbox facility to the customers	Low	3
OC-5	Auto dial from a pool of leads	Enables the sales agents to automatically dial to pool of leads	Very High	1
OC-6	Listen to call recordings	Records all the ongoing calls	Must Have	1
OC-7	Handle incoming calls	Handles the incoming calls of the customers	Nominal	3

6.1.2 Capabilities to be tested

Table 16. Capabilities to be tested

ID	Capability	Description	Priority Level	Iteration
OC-1	Log call detail information	Logs all the calling information for business analytics	Must Have	1
OC-2	Direct calling	Enables the sales agents, managers to make direct calls to the customers or leads	Must Have	1
OC-3	Call conferencing	Enables the manager to conference on a current call	High	2
OC-4	Provide voice-mailbox	Providing voice mailbox facility to the customers	Low	3
OC-5	Auto dial from a pool of leads	Enables the sales agents to automatically dial to pool of leads	Very High	1
OC-6	Listen to call recordings	Records all the ongoing calls	Must Have	1
OC-7	Handle incoming calls	Handles the incoming calls of the customers	Nominal	3

6.1.3 Capabilities not to be tested

In the first, second and third iteration, all the capabilities will be tested at the end of each iteration.

6.1.4 CCD Preparation Plans

<< Identify the clients and other users who will be involved in the Core Capability Drive-through, the usage scenarios that it will support, and the specific CCD preparation plans and milestones. These may include

- user context-setting
- site preparation dry runs,

- feedback forms, and
- CCD risk management plans. >>

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 19: 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

<< Briefly summarize the feedback you received from your client(s). You need to be specific enough to cover the critical capabilities or scenarios that were discussed, demoed, or shown. Your descriptions MUST, but not limited to, cover the following areas:
- Positive feedbacks
- Improvements needed/suggested
- Changes to-be considered (Reprioritized capabilities, requirements, GUI, etc.)
- Risks (New risks introduced, risks mitigated, etc.)

Note: Make sure to be specific to the capabilities shown/demonstrated/driven-through. Simply stating that the clients liked the capabilities is not sufficient. >>

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. >>