Life Cycle Plan (LCP)

The Los Angeles Community Garden Inventory and Locator

Team 13

Ardalan Yousefi	**Project Manager**
Cole Cecil	**Integrated Independent Verification & Validation**
Jeff Tonkovich	**Implementer**
Shi-Xuan Zeng	**Tester**

April 25, 2012

Version History

Date	Author	Version	Changes made	Rationale
09/19/11	Ardalan	0.1	Identified the roles and skills of the members of the development team.	In order to determine each team member's responsibilities, their roles and skills have to be identified first.
09/23/11	Ardalan	0.2	Started completing sections 1 & 2.2.	The duties of the life cycle planner in the exploration phase are not as many as the ones in the valuation phase. It is a good practice to gradually complete LCP while there is still time.
10/05/11	Ardalan	1.0	Changed section 2.2. corrected the skills of team members in section 3.3. added a new stake holder to section 3.1.	some deadlines had changed. 3.3 should list the required skills of each role instead of personal skills of team members.
10/06/11	Ardalan	1.1	Corrected section 1.2 and Table of Tables.	This section was changed in order to reflect the current status of the LCP, which is in the FCP instead of the VCP. The version number was also corrected.
10/14/11	Ardalan	1.2	Completed section 4 and 5.	The cost and effort required for this project was determined to see if this project can be done by a team of 6 within 24 weeks. Specified what tools and methods are going to be used in this project.
10/15/11	Ardalan	1.3	Completed section 2.1, 3.2. Corrected the defects found by the TA.	An overall picture of the project's life cycle will show the client at which milestones they should expect what type of results. Each team member's responsibility for each phase was specified so that all members know what they have to do in each phase, especially regarding their secondary roles.
10/17/11	Ardalan	1.4	Fixed the bugs found by the IIV&V.	
10/18/11	Ardalan	1.5	Edited project deliverables, roles and responisibilties.	
10/24/11	Ardalan	2.0	Edited section 3.2. Added a new table to 3.3 listing the required skills for 577b.	Suggestions made in the ARB meeting.
11/07/11	Ardalan	2.1	A minor change in Table 3.	This table was edited to reflect the minor change in the schedule on the course's website.

Date	Author	Version	Changes made	Rationale
11/19/11	Ardalan	2.2	Changes suggested by the TA. Updated section 5.	TA's comments from the FCP evaluation. Section 5 now contains the latest estimation of the project created in the 5th iteration on COTIPMO website.
11/21/11	Ardalan	2.3	Added more assumptions to section 1.3. Updated section 1.2.	There were a couple of assumptions in the project, but not mentioned in this section. This section should reflect the current status of the document.
12/01/11	Ardalan	2.4	Updated section 1.2 and 2.2. Added the responsibilities of users and clients to table 6. Updated section 1.3.	The detailed schedule of 577b was presented in class. Must determine the responsibilities of the user and the client during the development phase. Corrected the assumption about number of team members during spring semester.
12/05/12	Ardalan	2.5	In section 2.1, reduced the length of the rebaselined foundation phase to 2 weeks. Updated section 5 to reflect the latest iteration (7th).	Comments from the DCP ARB meeting.
02/06/12	Ardalan	3.0	Updated section 1.3 to show the current assumptions of the project. Updated section 2.1 and 2.2 to include more details about the development phase. Updated section 4.1.1. Added subversion to Table 9. Updated section 5 to reflect the latest resource estimates. Created section 6 for iteration plan.	Corrections mentioned by the TA.
02/15/12	Ardalan	3.1	Updated section 5 to reflect the latest changes to the cost estimate. Update section 6 to reflect the latest changes to the new requirements.	
03/26/12	Cole, Ardalan	4.0	Added the Iteration Assessment, minus the CCD results. Updated section 5. Removed TBD from table 6.	The COTIPMO results had changed since the previous version of LCP.
04/02/12	Cole	4.1	Added the CCD results.	

Date	Author	Version	Changes made	Rationale
4/14/12	Ardalan	4.2	Corrected the development phase information in section 2.1. updated the COTIPMO results.	IIV&V feedback.
4/25/12	Ardalan	4.3	Updated the COTIPMO results.	

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the LCP

Life Cycle Plan provides the answer to the most common questions about the project, such as "why?" (the purpose of the project), "when?" (project's milestones), "what?" (project deliverables), "who?" (responsibilities and roles), "how?" (monitoring, methods), and so forth.

1.2 Status of the LCP

The status of the LCP is currently at the IOC #2 Package version 4.3. This version includes the latest changes in the effort estimation.

1.3 Assumptions

- The duration of the project is 24 weeks, which are 12 weeks in Fall 2011 and 12 weeks in Spring 2012.
- Team13 has four members in the spring semester and this number will not change until project delivery.
- The client will not request a major change to the measure of the project during the 24 weeks.

2. Milestones and Products

2.1 Overall Strategy

The Los Angeles Community Garden Inventory and Locator is following Architected Agile process because more than 70% of the functionalities have to be developed by the team; there is no NDI or NCS that can fulfill more than 30% of the functionalities.

The development period is decomposed into the following phases:

Exploration phase

Duration: 09/09/11- 9/28/11
Concept: In this phase, the team specifies the initial scope of the system, identifies operational concepts, and identifies the necessary skills for this project.
Deliverables: Valuation Commitment Package, Client Interaction Report, Project Effort Report, Project Plan, Progress Report
Milestone: Valuation Commitment Review
Strategy: One Incremental Commitment Cycle

Valuation phase

Duration: 09/29/11- 10/21/11
Concept: In this phase, the success-critical stakeholders have win win negotiations in order to gather requirements, find risks, make mitigation plans, and understand and define the proposed system. In addition, the team builds some initial prototype of the main functionalities of the system and analyzes the behavior of the system with UML modeling.
Deliverables: Core Foundation Commitment Package, Evaluation of Core Foundation Commitment Package, Draft Foundation Commitment Package, Foundation Commitment Package, Project Effort Report, Project Plan, Progress Report
Milestone: Foundation Commitment Review
Strategy: One Incremental Commitment Cycle

Foundation phase

Duration: 10/22/11- 12/05/11
Concept: In this phase, the life cycle of the project is completely defined, the architecture of the system is designed, and feasibility evidence is provided.
Deliverables: Evaluation of Foundation Commitment Package, Draft Development Commitment Package, Evaluation of Draft Development Commitment Package, Development Commitment Package, Project Effort Report, Project Plan, Progress Report
Milestone: Development Commitment Review
Strategy: At least one Incremental Commitment Cycle

Rebaselined Foundation phase

Duration: 01/10/12 – 02/08/12

Concept: Three members left the team. In this phase, the main activities are finding a new member and coordinating between the new and old team members, distributing tasks among remaining members if no new member is found, and completing the iteration and test plans.
Deliverables: Draft Rebaselined Commitment Package, Rebaselined Commitment Package
Milestone: Rebaselined Development Commitment Review
Strategy: one Incremental Commitment Cycle

Development phase (construction)
Duration: 02/09/12 – 04/21/12
Concept: The proposed system is implemented in this phase in two iterations. The first iteration's milestone is the Core Capability Drive-through.
Deliverables: Initial Operational Capability Package, Draft Transition Package, Transition and Support Set Package
Milestone: Core Capability Drive-through, Transition Readiness Review
Strategy: Two Incremental Commitment Cycle

Development phase (transition)
Duration: 04/22/12 – 05/04/12
Concept: In this phase, the implemented system is transitioned and installed. The client is trained how to use the system.
Deliverables: Initial Operational Capability Working Set, Transition and Support Set Package
Milestone: End of the semester
Strategy: one Incremental Commitment Cycle

2.2 Project Deliverables

2.2.1 Exploration Phase

Table 1: Artifacts Deliverables in Exploration Phase

Artifact	Due date	Format	Medium
Client Interaction Report	09/21/2011	doc, pdf	Team website
Valuation Commitment Package • Operational Concept Description (OCD) Sections 1,2, and 3.1 • Life Cycle Plan (LCP) Section 3.3 • Feasibility Evidence Description (FED) Section 3	09/28/2011	doc, pdf	Team website
Project Effort Report	Every Monday	Text	ER system
Project Plan	Every Wednesday	mpp, pdf	Team website

Progress Report	Every Wednesday	xls	Team website

2.2.2 Valuation Phase

Table 2: Artifact deliverable in Valuation Phase

Artifact	Due date	Format	Medium
Core Foundation Commitment Package • Initial Prototype • OCD All sections • PRO • SSRD All sections • Software and System Architecture Description (SSAD) Sections 1, 2.1.1 – 2.1.3 • LCP Sections 1 and 3.3 • FED Sections 1, 3, 4.1, 4.2.1, and 4.2.2 • SID all sections	10/07/2011	doc, pdf, xls	Team website
Evaluation of Core Foundation Commitment Package	10/10/2011	doc, xls	Team website
Draft Foundation Commitment Package • OCD All sections • PRO All sections • SSRD All • SSAD Sections 1 and 2 • LCP All sections • FED Sections 1 to 5 • SID All Sections	10/14/2011	doc, pdf, xls	Team website
Evaluation of Draft Foundation Commitment Package	10/17/2011	doc, xls	Team website
Foundation Commitment Package • OCD All sections • PRO All sections • SSRD All Sections • SSAD Sections 1 and 2 • LCP All sections • FED Sections 1 to 5 • SID All sections • Quality Management Plan (QMP) Sections 1 and 2	10/24/2011	doc, pdf, xls	Team website
Project Effort Report	Every Monday	Text	ER system

Project Plan	Every Wednesday	mpp, pdf	Team website
Progress Report	Every Wednesday	xls	Team website

2.2.3 Foundations Phase

Table 3: Artifact deliverable in Foundations Phase

Artifact	Due date	Format	Medium
Quality Management Plan #1	10/24/2011	pdf	Team website
Evaluation of Foundation Commitment Package	10/31/2011	doc, xls	Team website
Response to Evaluation of FCP	11/07/2011		
Quality Management Plan #2	11/14/2011	pdf	Team website
Draft Development Commitment Package • OCD All Sections • PRO All Sections • SSRD All Sections • SSAD All Sections • LCP All Sections • FED Sections 1 to 5 • SID All Sections • QMP All Sections	11/21/2011	doc, pdf, xls	Team website
Evaluation of Draft Development Commitment Package	11/28/2011	doc, xls	Team website
Development Commitment Package • OCD All Sections • PRO All Sections • SSRD All Sections • SSAD All Sections • LCP All Sections • FED All Sections • SID All Sections • QMP All Sections • TP sections 1 and 3 • IP section 1 • ATPC sections 1 and 3	12/05/2011	doc, pdf, xls	Team website
Response to Draft Development Commitment Package	12/05/11		
Project Effort Report	Every Monday	Text	ER system
Project Plan	Every Wednesday	mpp, pdf	Team website
Progress Report	Every	xls	Team website

	Wednesday		
COTIPMO Survey	Every Wednesday		COTIPMO website

2.2.4 Rebaselined Foundation Phase

Table 4: Artifact deliverable in Rebaselined Foundation Phase

Artifact	Due date	Format	Medium
Draft Rebaselined Development Commitment Package • OCD All Sections • SSRD All Sections • SSAD All Sections • UML Diagrams • LCP All Sections, New Section 6 Iteration Plan • FED All Sections • SID All Sections • QMP All Sections • TP sections 1 and 3 • TPC sections 1 and 3	02/06/12	doc, pdf, xls, ras	Team website
Rebaselined Development Commitment Package • OCD All Sections • SSRD All Sections • SSAD All Sections • UML Diagrams • LCP All Sections, New Section 6 Iteration Plan • FED All Sections • SID All Sections • QMP All Sections • TP sections 1 and 3 • TPC sections 1 and 3	02/15/12	doc, pdf, xls, ras	Team website
Project Effort Report	Every Monday	Text	ER system
Project Plan	Every Wednesday	mpp, pdf	Team website
Progress Report	Every Wednesday	xls	Team website
COTIPMO Survey	Every other Wednesday		COTIPMO website

2.2.5 Development Phase

Table 5: Artifact deliverable in Development Phase

Artifact	Due date	Format	Medium
Initial Operational Capability (IOC) Package • OCD All Sections • SSRD All Sections • SSAD All Sections • UML Diagrams • LCP All Sections • FED All Sections • SID All Sections • QMP All Sections • TP All sections • TPC All sections • Test Procedure and Results • Iteration Assessment Report	03/26/12	doc, pdf, xls, ras	Team website
Core Capability Drive-through Report • CCD Report • CodeCount Report • CodeCount Output file • COCOMO II Estimation Uncertainty At CCD • COCOMO Report • Value-based Testing Procedure and Results	04/02/12	doc	Team website
Draft Transition Package • TP • User Manual • Support Plan • Training materials • Regression Test Package	04/09/12	doc,pdf	Web
Transition and Support Package • TP • UM • SP • TM • RTP	04/16/12	doc,pdf	Team website
IOC Working Set and TS Package	04/27/12		Team website
Close Out Report	05/04/12		Team website, SAL 329
Project Effort Report	Every	Text	ER system

	Monday		
Project Plan	Every Wednesday	mpp, pdf	Team website
Progress Report	Every Wednesday	xls	Team website
COTIPMO Survey	Every other Wednesday		COTIPMO website

3. Responsibilities

3.1 Project-specific stakeholder's responsibilities

Database Manager: The database manager is a member of the one of the Los Angeles Community Gardening Council (LACGC), the Los Angeles Neighborhood Land Trust (LANLT), UCCE, or LACC organizations. The database manager is responsible for updating the information of community gardens stored in the database.

3.2 Responsibilities by Phase

The responsibilities of the members that left the team have been moved to the end of the table.

Table 6: Responsibilities by Phase

	Primary / Secondary Responsibility				
Team Member / Role	**Exploration**	**Valuation**	**Foundations**	**Development-** Construction Iteration	**Development-** Transition Iteration
Name: Ardalan Yousefi Project Manager, Implementer, Trainer, System/Software Architect, Life Cycle Planner	**Primary Responsibility** - Identify Responsibilities and Skills **Secondary Responsibility** - Record Project Individual Effort	**Primary Responsibility** - Analyze the proposed system **Secondary Responsibility** - Plan for project life cycle - Record Project Individual Effort	**Primary Responsibility** - Specify architecture styles, patterns, and frameworks - Define technology-dependent architecture - Assess system architecture **Secondary Responsibility** - Identify development iteration - Develop transition plan - Assess life cycle content - Record Project Individual Effort	**Primary Responsibility** - Detail project plan - Record project progress **Secondary Responsibility** - Implement the system - Record Project Individual Effort	**Primary Responsibility** - Detail project plan - Record project progress - Develop Support Plan **Secondary Responsibility** - Deploy and install the system - Train users - Record Project Individual Effort
Name: Cole Cecil Integrated Independent Verification &	**Primary Responsibility** - Verify and Validate Work	**Primary Responsibility** - Verify and Validate Work	**Primary Responsibility** - Verify and Validate Work	**Primary Responsibility** - Verify and Validate Work	**Primary Responsibility** - Verify and Validate Work

Validation, Quality Focal Point, Tester	Products **Secondary Responsibility** - Record Project Individual Effort	Products - Identify Quality Management Plan **Secondary Responsibility** - Record Project Individual Effort	Products - Identify configuration management strategy - Assess quality management strategy **Secondary Responsibility** - Construct traceability matrix - Record Project Individual Effort	Products - Assess Quality Management Plan - Assess Configuration Management Strategy **Secondary Responsibility** - Assess development iteration - Test the system - Record Project Individual Effort	Products - Assess Quality Management Plan - Assess Configuration Management Strategy **Secondary Responsibility** - Test the deployed system - Record Project Individual Effort
Name: Jeff Tonkovich Implementer, Operational Concept Engineer	**N/A**	**N/A**	**N/A**	**Primary Responsibility** - Implement the system **Secondary Responsibility** - Assess Operational Concept - Record Project Individual Effort	**Primary Responsibility** - Deploy and install the system - Develop transition plan **Secondary Responsibility** - Assess Operational Concept - Record Project Individual Effort
Name: Shi-Xuan Zeng Tester, Trainer, Feasibility Analyst, Requirements Engineer	**Primary Responsibility** - Assess and plan to mitigate risks **Secondary Responsibility** - Record Project Individual Effort	**Primary Responsibility** - Explore Alternatives - Provide project feasibility evidence **Secondary Responsibility** - Record Project Individual Effort	**Primary Responsibility** - Assess feasibility evidence **Secondary Responsibility** - Record Project Individual Effort	**Primary Responsibility** - Test the system **Secondary Responsibility** - Assess Feasibility Evidence - Assess Requirements Definition - Record Project Individual Effort	**Primary Responsibility** - Test the deployed system - Train users **Secondary Responsibility** - Assess Feasibility Evidence - Assess Requirements Definition - Record Project Individual Effort
User	N/A	N/A	N/A	- Review and test the system in development environment - Provide feedback about system output and performance	- Attend training
Client	- Analyze current system	- Identify objectives, constraints, and priorities - Define	- Review design, prototypes, plan, and feasibility during ARB - Provide	- Assess development iteration - Perform core capability drive-	- Develop transition plan - Develop support plan - Be prepared to

		Operational Concept - Provide win conditions and negotiate with development team - Review design, prototypes, plan, and feasibility during ARB	feedback on the progress of the project so far - Develop transition plan	though	receive training from the trainer - Hire system maintainers
Name: Cheng-Yin Wu Prototyper, System/Software Architect	**Primary Responsibility** - Analyze current system **Secondary Responsibility** - Record Project Individual Effort	**Primary Responsibility** - Prototyping **Secondary Responsibility** - Analyze the proposed system - Record Project Individual Effort	**Primary Responsibility** - Prototyping - Assess prototype and components **Secondary Responsibility** - Specify architecture styles, patterns, and frameworks - Define technology-dependent architecture - Assess system architecture - Record Project Individual Effort	N/A	N/A
Name: Chih-rung Larry Chen Project Manager, Feasibility Analyst	**Primary Responsibility** - Detail project plan - Record project progress **Secondary Responsibility** - Assess and plan to mitigate risks - Record Project Individual Effort	**Primary Responsibility** - Detail project plan - Record project progress **Secondary Responsibility** - Explore Alternatives - Provide project feasibility evidence - Construct Traceability Matrix - Gather Definitions - Record Project Individual Effort	**Primary Responsibility** - Detail project plan - Record project progress **Secondary Responsibility** - Identify development iteration - Develop transition plan - Plan for testing - Assess feasibility evidence - Record Project Individual Effort	N/A	N/A
Name: Pei-Chen Liao Operational Concept Engineer, Requirements	**Primary Responsibility** - Analyze current system **Secondary**	**Primary Responsibility** - Identify objectives, constraints, and	**Primary Responsibility** - Assess operational concept	N/A	N/A

Engineer	**Responsibility** - Record Project Individual Effort	priorities - Define operational concept **Secondary Responsibility** - Negotiate Win Conditions - Define Requirements	**Secondary Responsibility** - Prototyping - Assess prototype and components - Assess requirements definitions - Record Project Individual Effort		

3.3 Skills

Table 7: Skills of team members based on their roles

Team members	Role	Skills
Ardalan Yousefi	Project Manager, Implementer, System Architect, Life Cycle Planner, Trainer	Project Management, MS Project, ASP .Net, Visual Studio, SQL Server, Google Maps API, Analysis and Design, UML, COCOMO II, COTIPMO, Good presentation skills, SVN
Cole Cecil	IIV&V, QFP, Tester	Bugzilla, WinBook, Subversion, Unit Testing, Integration Testing, Value-Based Test Prioritization, Requirements-Test Traceability, Regression Testing, SVN
Jeff Tonkovich	Implementer, Operational Concept Engineer	ASP .Net, Visual Studio, SQL Server, Google Maps API, System Analysis, UML, SVN
Shi-Xuan Zeng	Tester, Trainer, Feasibility Analyst	Unit Testing, Integration Testing, Value-Based Test Prioritization, Requirements-Test Traceability, Regression Testing, Cost Analysis, Benefit Analysis, ROI Analysis, Good presentation skills, Winbook, Value Based Requirements Prioritization Techniques

4. Approach

4.1 Monitoring and Control

The following methods are used for monitoring and controlling various aspects of the project:

- **Progress Report:** This document is a weekly report of the progress of the project. It includes top priority works to be completed that week, accomplishments of the previous week, planned and actual man-hours, risks identified so far and their mitigation plans, list of COTS being considered by the team, and any defects in the project found during the previous week.
- **Project Plan:** This document is a Microsoft Project file which has the detailed plan of the project. It includes all the milestones of each phase, all the tasks and the team members responsible for each task, and a Gantt chart of the project plan.
- **Effort Report:** Each team member fills out this report weekly as a way to report how much time they spent on doing a task assigned to them.
- **Client Meeting Notes:** After every meeting with the client, the minutes are saved to this document and uploaded to the team website. This document ensures that all the main points of the meeting are recorded for future reference.

4.1.1 Closed Loop Feedback Control

Email is the primary method of providing feedback within the team, especially between on-campus students and the DEN students. In addition, the on-campus students have video-conference meetings once a week with the off-campus students. In some rare urgent cases, IM and texting is preferred.

4.1.2 Reviews

Team 13 uses various techniques for reviewing the project:

- **Peer Review:** Each team member occasionally reviews artifacts developed by other members, particularly the artifacts related to their secondary roles.
- **IIV&V Review:** The off-campus member of the team reviews all the artifacts submitted for each commitment package. The on-campus members and the off-campus member work collaboratively on the bugs using the BugZilla bug tracking system.
- **TA Review:** The teaching assistants of the course review all the artifacts after each submission and suggest corrections and improvements.
- **Architecture Review Board:** The ARB is a meeting during which the development team gives a presentation about the progress of the project so far. Among participants of this meeting are the client, the instructor and the teaching assistants of the course, and sometimes people from the industry. All of these people review the progress of the project and give feedback during this meeting.

4.2 Methods, Tools and Facilities

Table 8: List of tools used in this project

Tools	Usage	Provider
Winbook	Lists the win conditions of the project. Provides a virtual environment for the client and the development team to discuss the win conditions, assess risks, and agree to win conditions.	USC
BugZilla	Helps keep track of the bugs found in the artifacts.	USC
Microsoft Project	Facilitates project planning, setting milestones, defining tasks for each team member, etc.	USC
Microsoft Word	All the ICSM documents are created with this tool.	Team members
COTIPMO	Calculates the effort and time estimates of the project and helps the life cycle planner keep track of the size and progress of the project for each iteration.	USC
Rational Software Modeler	Helps the team create UML diagrams.	USC
iCard	Each team member reports their weekly individual effort with this tool.	USC
Microsoft Excel	Progress report, SSRD, and evaluation review report are created with this tool.	Team members
Subversion	Used for configuration management of the project.	USC

5. Resources

The client prefers the budget to be zero; but if inevitable, it should not exceed $1000.

The project duration is 24 weeks.

Team 13 has four members.

The Los Angeles Community Garden Inventory and Locater system consists of four modules:

- Garden Locator (Map)
- Garden Management
- User Management
- Report Generator

Below you can see the latest values of the scale drivers and cost drivers for cost and effort estimation using COTIPMO. These values change over time based on prototyping and suggestions made by the COTIPMO tool.

Table 9: Scale Drivers

Scale Driver	Value	Rationale
PREC	High	A similar system called "P-Patch" is available online, which maps the community gardens of Seattle.
FLEX	Nominal + 50%	The client is not too strict on specifications.
RESL	High	The software is developed using the risk-based ICSM process.
TEAM	Nominal + 50%	The developers have some experience in team work, but not much. All stakeholders have been consistent in their objectives so far.
PMAT	Nominal + 50%	While not completely at the third level of CMMI, the team does some activities of this level such as peer reviews and risk management.

Table 10: Garden Locator Cost Drivers

Cost Driver	Value	Rationale
RELY	Low	The Garden Locator is not safety-critical, and its sole purpose is information sharing with the public.
DATA	Nominal	Probably only testing the whole database (80 records) will be enough, and since this module uses a map web service, it does not have that many LOC. Therefore, the ratio will not exceed 100.
CPLX	Nominal	Integrating this module with a map web service is probably the main complexity in this module.

RUSE	Low	No mention of reusability in the requirements.
DOCU	High	ICSM requires a lot of documentation.
TIME	Nominal	This system is a web application. Therefore, the system will not use more than 50% of the available execution time.
STOR	Nominal	Storage is not an issue for this system.
PVOL	Low	Platform is stable.
ACAP	Nominal + 50%	The team has good analysis and design skills; and a new member with high analytical skills joined the team.
PCAP	High	The team has good programming capability, and the team members communicate well with each other via email and face-to-face meetings.
PCON	Nominal	None of the team members will leave the team until project delivery.
APEX	Low	The team's experience with map web services is limited.
PLEX	Nominal	All team members have at least one year of experience in working with database, UI, and web applications, either in the industry or at college.
LTEX	High	Most team members have very good experience of .Net programming, and are familiar with HTML, Javascript, and CSS.
TOOL	Nominal	Bugzilla for bug tracking
SITE	Nominal	The IIV&V is done by a DEN student. On-campus team members communicate with the DEN student mostly via email.

Table 11: Garden Management Cost Drivers

Cost Driver	Value	Rationale
RELY	Nominal + 50%	It is important for the client that the website provide some basic security for the garden database.
DATA	Nominal	Even though all the 80 records will not be tested in this case, but each record has a lot of fields, and the LOC of this module is not small, either. Therefore, we chose Nominal instead of Low.
CPLX	High	Adding, modifying, and deleting columns makes this module more difficult than just updating a simple database table.
RUSE	Low	No mention of reusability in the requirements.
DOCU	High	ICSM requires a lot of documentation.
TIME	Nominal	Web application + small database. Therefore, the system will not use more than 50% of the available execution time.
STOR	Nominal	Storage is not an issue for this system.
PVOL	Low	Platform is stable.
ACAP	Nominal	The team has good analysis and design skills; and a new

	+ 50%	member with high analytical skills joined the team.
PCAP	High	The team has good programming capability, and the team members communicate well with each other via email and face-to-face meetings.
PCON	Nominal	None of the team members will leave the team until project delivery.
APEX	Nominal + 50%	All team members have at least one year of experience of developing database management software, either in the industry or at college. Also, the new member has considerable experience in developing ASP .NET web applications.
PLEX	Nominal	All team members have at least one year of experience in working with database, UI, and web applications, either in the industry or at college.
LTEX	High	Most team members have very good experience of .Net programming, and are familiar with HTML, Javascript, and CSS.
TOOL	Nominal	Bugzilla for bug tracking
SITE	Nominal	The IIV&V is done by a DEN student. On-campus team members communicate with the DEN student mostly via email.

Table 12: User Management Cost Drivers

Cost Driver	Value	Rationale
RELY	High	This module deals with the database managers' user name and password and requires a higher security than other modules.
DATA	Nominal	Moderate test data size, not too many LOC.
CPLX	Nominal	The usual complexities of connecting to a database and updating it.
RUSE	Low	No mention of reusability in the requirements.
DOCU	High	ICSM requires a lot of documentation.
TIME	Nominal	Web application + small database. Therefore, the system will not use more than 50% of the available execution time.
STOR	Nominal	Storage is not an issue for this system.
PVOL	Low	Platform is stable.
ACAP	Nominal + 50%	The team has good analysis and design skills; and a new member with high analytical skills joined the team.
PCAP	High	The team has good programming capability, and the team members communicate well with each other via email and face-to-face meetings.
PCON	Nominal	None of the team members will leave the team until

		project delivery.
APEX	Nominal	The team has good experience of developing user authentication systems.
PLEX	Nominal	All team members have at least one year of experience in working with database, UI, and web applications, either in the industry or at college.
LTEX	High	Most team members have very good experience of .Net programming, and are familiar with HTML, Javascript, and CSS.
TOOL	Nominal	Bugzilla for bug tracking
SITE	Nominal	The IIV&V is done by a DEN student. On-campus team members communicate with the DEN student mostly via email.

Table 13: Report Generator Cost Drivers

Cost Driver	Value	Rationale
RELY	Low	The report generator is not safety-critical.
DATA	Nominal	Moderate test data size, not too many LOC.
CPLX	Nominal + 50%	Prototyping two NDIs for exporting the gardens to PDF and XLS files showed that implementing this module is not as complex as we thought it would be.
RUSE	Nominal	The external section of the website will also use this module to generate reports for end users.
DOCU	High	ICSM requires a lot of documentation.
TIME	Nominal	The report is generated on the web server, the database is small. Therefore, the system will not use more than 50% of the available execution time.
STOR	Nominal	Storage is not an issue for this system.
PVOL	Low	Platform is stable.
ACAP	Nominal + 50%	The team has good analysis and design skills; and a new member with high analytical skills joined the team.
PCAP	High	The team has good programming capability, and the team members communicate well with each other via email and face-to-face meetings.
PCON	Nominal	None of the team members will leave the team until project delivery.
APEX	Low	There is no one on the team who has any experience with generating reports from a database to PDF/XLS files.
PLEX	Nominal	All team members have at least one year of experience in working with database, UI, and networking, either in the industry or at college.
LTEX	High	Most team members have very good experience of .Net programming, and are familiar with HTML, Javascript,

		and CSS.
TOOL	Nominal	Bugzilla for bug tracking. Visual Studio for ASP .NET development.
SITE	Nominal	The IIV&V is done by a DEN student. On-campus team members communicate with the DEN student mostly via email.

Iteration List Add

#	Start Date	End Date	Description	Scale Factor	Modules	Spent PM	Estimated PM	Actions
1	10/19/11	10/26/11		16.32	4	-	8.69 (1321 hrs)	
2	10/26/11	11/2/11	Updated the scale factors according to the survey results and suggestions.	16.40	4	-	8.69 (1321 hrs)	
3	11/2/11	11/9/11	Updated the FLEX scale factor from NOM to NOM+50% After a team meeting on Nov 2, decided to change the complexity of the Garden management module from Nominal to High.	14.49	4	-	9.25 (1406 hrs)	
4	11/9/11	11/16/11	No change from last iteration.	14.49	4	-	9.25 (1406 hrs)	
5	11/16/11	11/23/11	Reduced the complexity of report generator by 50%. Prototypes showed that generating reports is not as difficult as we thought it would be. Also, this module needs to be reused by the external side of the web site; as a result, RUSE cost driver was change from LOW to NOM.	14.49	4	-	9.21 (1400 hrs)	
6	11/23/11	11/30/11	Reduced the SLOC of User Management module because we decided to use ASP .Net Membership and Roles service.	14.49	4	0.97 (147 hrs)	7.01 (1066 hrs)	
7	11/30/11	12/7/11	Increased PCON to NOM+50% because only one person will not continue next semester.	14.49	4	1.40 (213 hrs)	4.88 (741 hrs)	
8	12/7/11	12/14/11		14.49	4	1.40 (213 hrs)	4.88 (741 hrs)	
9	2/1/12	2/14/12	Changed ACAP to NOM+50% because the new team member has impressive analysis skills. Change PCON from NOM+50% to NOM because hopefully there will be no more turnover until the end of the project! Increased APEX of Garden Management to NOM+50% because the new member seems to be highly familiar with ASP .NET web applications that deal with databases.	14.49	4	1.35 (205 hrs)	4.65 (707 hrs)	
10	2/15/12	2/29/12	Increased the RELY cost driver of Garden Management to NOM + 50%. The database should be prepared for SQL injection attacks and cross-site scripting. Increased the CPLX cost driver of Garden Management to HI because implementing dynamic columns turned out to be more difficult than expected.	14.49	4	1.45 (220 hrs)	5.32 (808 hrs)	
11	2/29/12	3/20/12	Considerable progress in implementing the modules. No changes to the cost drivers.	14.49	4	5.20 (790 hrs)	7.53 (1144 hrs)	
12	3/21/12	4/4/12	Finishing the implementation for the CCD session. No changes to the cost drivers. Changed the REVL of the modules now that most of them are done.	14.49	4	5.44 (827 hrs)	7.79 (1184 hrs)	
13	4/4/12	4/24/12	Fixed a couple of the bugs pointed out in the CCD. No changes to the cost drivers.	14.49	4	6.00 (912 hrs)	6.68 (1015 hrs)	
14	4/25/12	5/1/12	The implementation of all modules is finished. The system is ready for deployment.	14.49	4	6.92 (1052 hrs)	6.92 (1052 hrs)	

Figure 1 - COTIPMO Tool result

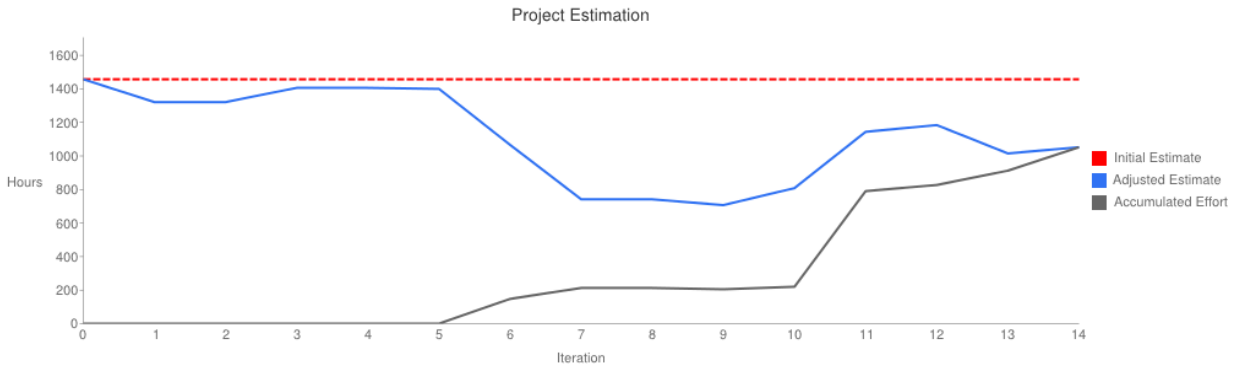

Figure 2 – Project Progress

6. Iteration Plan
6.1 Plan

The first iteration for the development phase will be based on the core capability drive through milestone target. The core capability drive through will have all the core capabilities implemented for the functioning system and must-have requirements addressed.
The second iteration will be determined based on the first iteration results, core capability drive through review, and testing results, and subject to changes depending on the bugs still remaining by the time of the core capability drive through review milestone and discovered after the review.

6.1.1First Iteration

This is the first system deliverable that will subject to various tests in order to provide a clear picture of how the system will function and if the requirements are satisfied. This iteration will end with the core capability drive through – on March 26 or 28 - and document problems or improvements that will be addressed in the next iteration.

6.1.1.1 Capabilities to be Implemented

The first iteration will implement all the core requirements and capabilities in order to satisfy a functional system. All the "must-have" requirements will be developed at this iteration. Other requirements will be considered in the next iteration.

Table 14: Capabilities to be Implemented in First Iteration

Capability Requirement	Description	Priority
CR-1: Provide Web Interface to Database Accessible by Database Managers	The database must be accessible through a web interface to allow for users to view and manipulate garden information	(M) Must Have
CR-2: Provide Web Interface to Database Accessible by End-Users	The database must be accessible, through another web interface, to allow end-users to view and search garden information	(M) Must Have
CR-3: Login to the System	User must log in to the system to access the functionalities relating to garden information	(M) Must Have
CR-4: Logout of the System	User must be able log out of the system to prevent the system from being used by other users	(M) Must Have
CR-5: View Garden Information	Users should be able to view all the garden information in the database	(M) Must Have
CR-6: Sort Garden Information	Users should be able to sort the garden information by the columns, any column in the garden database table can be selected to sort	(M) Must Have
CR-7: Search Garden Information	Users need to be able to search the database for	(M) Must Have

	garden records matching the search keyword(s)	
CR-8: Export Garden Information	The system should be able to export the garden information in the XLS/PDF formats	(M) Must Have
CR-9: Add Garden Record	The system should be able to add new garden information	(M) Must Have
CR-10: Modify Garden Record	The system should be able to update existing garden information	(M) Must Have
CR-11: Delete Garden Record	The system should be able to delete existing garden information	(M) Must Have
CR-12: Add Garden Table Column	The system should be able to expand the garden table to include more columns to store more information	(M) Must Have
CR-13: Delete Garden Table Column	The system should be able to delete existing columns from the garden table	(M) Must Have
CR-14: Add User	The system should provide user management to add new user accounts	(M) Must Have
CR-15: Modify User	The system should provide user management to modify existing user accounts	(M) Must Have
CR-16: Delete User	The system should provide user management to delete existing user accounts	(M) Must Have
CR-17: View Public Garden Information Available to End-Users	The system should allow end-users to view all gardens in limited details (only garden name and address)	(M) Must Have
CR-18: View Public Garden Information Map Available to End-Users	The system should have a mapping component to display gardens on Google Map	(M) Must Have
CR-19: Search Public Garden Information Available to End-Users	The system should allow end-users to search for gardens	(M) Must Have
CR-20: View Public Garden Information Detail Available to End-Users	The system should allow end-users to view specific garden information in more, but still limited details (only garden name, address, and other columns selected by the database managers)	(M) Must Have
CR-21: Download Garden Report for End-Users	The system should allow the end-users to download the garden report in either XLS or PDF format	(M) Must Have

The capabilities that are not implemented the first iteration will be implemented in the next iteration, after evaluation of the core capability drive through and subject to development team capability and time constraints.

Table 15: Capabilities not being Implemented in First Iteration

Capability Requirement	Description	Priority
CR-22: View Public Garden Information with Pictures to End-Users	The system should allow for the end-users to view garden information with pictures attached to the garden	S (Should have)
CR-23: Log Changes to Database Records	The system should record every change to the database records	W (Want to have)

CR-24: View Database Log	The system should allow the database managers to view the log file.	W (Want to have)
CR-25: Add Picture to Garden Record	The system should allow the database managers to attach pictures to garden record	C (Could have)
CR-26: Delete Picture from Garden Record	This system should allow the database managers to remove pictures from garden record	C (Could have)
CR-27: View Pictures of Garden Record	This system should allow user to view pictures of gardens	C (Could have)
CR-28: View Garden Driving Direction	This system should provide a driving direction to selected garden for end-users.	C (Could have)

6.1.1.2 Capabilities to be Tested

The capabilities and requirements that will be tested in the first iteration are listed here. In addition, the non-functional requirements specified in the SSRD document will also be tested.

Table 16: Capabilities to be Tested in First Iteration

Capability Requirement	Priority	Rationale
CR-1: Provide Web Interface to Database Accessible by Database Managers	(M) Must Have	The first overall requirement for the web interface to the system is required for the system to function.
CR-2: Provide Web Interface to Database Accessible by End-Users	(M) Must Have	The second overall requirement for the web interface to the system is required for the system to function.
CR-3: Login to the System	(M) Must Have	User login is the first requirement in order to access most of the functionalities of the system and must be tested.
CR-4: Logout of the System	(M) Must Have	User logout is the complement of the CR-3: Login to the System and must be tested.
CR-5: View Garden Information	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.
CR-6: Sort Garden Information	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.
CR-7: Search Garden Information	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.
CR-8: Export Garden Information	(M) Must Have	The ability to export out the garden records is another essential part of the system and must be tested.
CR-9: Add Garden Record	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.
CR-10: Modify Garden Record	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.

CR-11: Delete Garden Record	(M) Must Have	This is one of the capabilities to deal with garden records manipulation and is essential to the system and must be tested.
CR-12: Add Garden Table Column	(M) Must Have	Modification of the garden table is key to the flexibility of the database to store more information in the future and must be tested.
CR-13: Delete Garden Table Column	(M) Must Have	Modification of the garden table is key to the flexibility of the database to store more information in the future and must be tested.
CR-14: Add User	(M) Must Have	User account management is core to the system ability to allow multiple database managers to manage the same data and must be tested.
CR-15: Modify User	(M) Must Have	User account management is core to the system ability to allow multiple database managers to manage the same data and must be tested.
CR-16: Delete User	(M) Must Have	User account management is core to the system ability to allow multiple database managers to manage the same data and must be tested.
CR-17: View Public Garden Information Available to End-Users	(M) Must Have	This is one of the capabilities to deal with general public access to the database information and must be tested.
CR-18: View Public Garden Information Map Available to End-Users	(M) Must Have	This is one of the capabilities to deal with general public access to the database information and must be tested.
CR-19: Search Public Garden Information Available to End-Users	(M) Must Have	This is one of the capabilities to deal with general public access to the database information and must be tested.
CR-20: View Public Garden Information Detail Available to End-Users	(M) Must Have	This is one of the capabilities to deal with general public access to the database information and must be tested.
CR-21: Download Garden Report for End-Users	(M) Must Have	The ability to download latest reports by the general public is another essential part of the system and must be tested.

Table 17: Non-Functional Requirements to be Tested in First Iteration

Non-Functional Requirement	Priority	Rational
LOS-1: System Response Time	S (Should Have)	The system performance is essential to the project, and must be tested in order to improve in the next iteration.
SR-2: Defined Styles for PDF/Spreadsheet exports	M (Must have)	The styles for the PDF/spreadsheet exports should be set in stone by now to avoid creating more issues for the next iteration.
PR-3: Web browser support for Windows 7/OS X Lion 10.7	S (Should have)	The support of the web browsers should also be tested so problems can be found and fixed and give time for next iteration to focus on any remaining capabilities not be implemented.
LOS-2: System Security	S (Should have)	The system uses a database that should be only accessible to certain users; security measures should be taken in order to fulfill this

		requirement. Also, attacks such as SQL injection and cross-site scripting should be prevented.

6.1.1.3 Capabilities not to be Tested

All the capabilities implemented in the first iteration will all need to be tested, so there is no capability that is not tested.

6.1.2 Second Iteration

This iteration will address problems discovered during core capability drive through review and any leftover bugs that did not get fixed in time for the review. Any remaining capabilities that can be implemented within the time frame before the transition phase will be implemented and tested. This will be the final deliverable of the system to the client.

6.1.2.1 Capabilities to be Implemented

The second iteration will implement the rest of the capabilities that are not implemented in the first iteration only if this iteration is not being used up by bug fixes or any unforeseen problems.

Table 18: Capabilities to be Implemented in Second Iteration

Capability Requirement	Description	Priority
CR-22: View Public Garden Information with Pictures to End-Users	The system should allow for the end-users to view garden information with pictures attached to the garden	S (Should have)
CR-23: Log Changes to Database Records	The system should record every change to the database records	W (Want to have)
CR-24: View Database Log	The system should allow the database managers to view the log file.	W (Want to have)
CR-25: Add Picture to Garden Record	The system should allow the database managers to attach pictures to garden record	C (Could have)
CR-26: Delete Picture from Garden Record	This system should allow the database managers to remove pictures from garden record	C (Could have)
CR-27: View Pictures of Garden Record	This system should allow user to view pictures of gardens	C (Could have)
CR-28: View Garden Driving Direction	This system should provide a driving direction to selected garden for end-users.	C (Could have)

6.1.2.2 Capabilities to be Tested

All capabilities and requirements implemented in the second iteration, if any, should be all tested. In addition, any new non-functional requirement that was implemented in the second

iteration should be tested as well. Last, all the capabilities and requirements implemented by the first iteration should be re-tested as well.

Table 19: Capabilities to be Tested in Second Iteration

Capability Requirement	Priority	Rationale
CR-1: Provide Web Interface to Database Accessible by Database Managers	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-2: Provide Web Interface to Database Accessible by End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-3: Login to the System	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-4: Logout of the System	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-5: View Garden Information	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-6: Sort Garden Information	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-7: Search Garden Information	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-8: Export Garden Information	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-9: Add Garden Record	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-10: Modify Garden Record	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-11: Delete Garden Record	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-12: Add Garden Table Column	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-13: Delete Garden Table Column	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-14: Add User	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-15: Modify User	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing

		purpose.
CR-16: Delete User	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-17: View Public Garden Information Available to End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-18: View Public Garden Information Map Available to End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-19: Search Public Garden Information Available to End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-20: View Public Garden Information Detail Available to End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-21: Download Garden Report for End-Users	(M) Must Have	Capability implemented by first iteration needs to be tested again for regression testing purpose.
CR-22: View Public Garden Information with Pictures to End-Users	S (Should have)	All remaining capabilities implemented shall be tested.
CR-23: Log Changes to Database Records	W (Want to have)	All remaining capabilities implemented shall be tested.
CR-24: View Database Log	W (Want to have)	All remaining capabilities implemented shall be tested.
CR-25: Add Picture to Garden Record	C (Could have)	All remaining capabilities implemented shall be tested.
CR-26: Delete Picture from Garden Record	C (Could have)	All remaining capabilities implemented shall be tested.
CR-27: View Pictures of Garden Record	C (Could have)	All remaining capabilities implemented shall be tested.
CR-28: View Garden Driving Direction	C (Could have)	All remaining capabilities implemented shall be tested.

Table 20: Non-Functional Requirements to be Tested in Second Iteration

Non-Functional Requirement	Priority	Rational
LOS-1: System Response Time	S (Should Have)	Capability implemented by first iteration needs to be tested again for regression testing purpose.
SR-1: Google Spreadsheet Like UI for Garden Record Editing	W (Want to have)	All remaining capabilities implemented shall be tested.
SR-2: Defined Styles for PDF/Spreadsheet exports	M (Must have)	Capability implemented by first iteration needs to be tested again for regression testing purpose.
SR-3: View garden info in bubble in the Google maps	C (Could have)	The client thinks that having to go back and forth between the map page and the garden details page is not user-friendly, so this requirement should be tested.

PR-3: Web browser support for Windows 7/OS X Lion 10.7	S (Should have)	Capability implemented by first iteration needs to be tested again for regression testing purpose.
LOS-2: System Security	S (Should have)	Capability implemented by first iteration needs to be tested again for regression testing purpose.

6.1.2.3 Capabilities not to be Tested

All capabilities implemented should be tested at this point and no capability should be left without being tested.

6.2 Iteration Assessment

This section discusses the results of the first iteration. It lists the capabilities which were implemented, summarizes the test results, discusses how well the plan was followed, and discusses the Core Capability Drivethrough results.

6.2.1 Capabilities Implemented

Below is the list of capabilities planned to be implemented in the first iteration, along with their status. Two of the planned capabilities did not get implemented.

Capability Requirement	Implemented
CR-1: Provide Web Interface to Database Accessible by Database Managers	Yes
CR-2: Provide Web Interface to Database Accessible by End-Users	Yes
CR-3: Login to the System	Yes
CR-4: Logout of the System	Yes
CR-5: View Garden Information	Yes
CR-6: Sort Garden Information	Yes
CR-7: Search Garden Information	Yes
CR-8: Export Garden Information	Yes
CR-9: Add Garden Record	Yes
CR-10: Modify Garden Record	Yes
CR-11: Delete Garden Record	Yes
CR-12: Add Garden Table Column	No
CR-13: Delete Garden Table Column	No
CR-14: Add User	Yes
CR-15: Modify User	Yes
CR-16: Delete User	Yes
CR-17: View Public Garden Information Available to End-Users	Yes
CR-18: View Public Garden Information Map Available to End-Users	Yes
CR-19: Search Public Garden Information Available to End-Users	Yes
CR-20: View Public Garden Information Detail Available to End-Users	Yes
CR-21: Download Garden Report for End-Users	Yes

6.2.2 Summary of Test Results

In the testing of the first iteration, a total of 56 defects were found. Out of these 56, 49 of them have been closed. Of the 7 open defects, 2 are normal, 2 are minor, 1 is trivial, and 2 are enhancements. Hence, the system is currently quite stable.

Listed below is the testing status of all the implemented capability requirements:

Capability Requirement	Test Results
CR-1: Provide Web Interface to Database Accessible by Database Managers	Pass
CR-2: Provide Web Interface to Database Accessible by End-Users	Pass
CR-3: Login to the System	Pass
CR-4: Logout of the System	Pass
CR-5: View Garden Information	Pass
CR-6: Sort Garden Information	Pass
CR-7: Search Garden Information	Pass
CR-8: Export Garden Information	Pass
CR-9: Add Garden Record	Pass
CR-10: Modify Garden Record	Pass
CR-11: Delete Garden Record	Pass
CR-14: Add User	Pass
CR-15: Modify User	Pass
CR-16: Delete User	Pass
CR-17: View Public Garden Information Available to End-Users	Pass
CR-18: View Public Garden Information Map Available to End-Users	Pass
CR-19: Search Public Garden Information Available to End-Users	Pass
CR-20: View Public Garden Information Detail Available to End-Users	Pass
CR-21: Download Garden Report for End-Users	Pass

Listed below are the open defects and limitations in the system:

Defect ID	Description	Plan for Resolution
6918	Website URLs that don't start with "http://" don't show up correctly in the PDF.	This will be fixed in the next iteration.
6878	There are no arrows showing the sort direction in the Garden Management table.	This is an enhancement which is not trivial to implement, so it is not likely to be implemented.
6920	Duplicate garden names are prevented, but no error message appears to let the user know what went wrong.	This will be fixed in the next iteration.

6851	"Required" errors don't always appear when tabbing through, but they still appear when submitting.	This is a trivial issue that could be difficult to fix, so it might not be fixed.
6852	There aren't error messages for all the validations, so if information can't be submitted, the user might not know what's wrong.	This will be fixed in the next iteration.
6877	The header and footer image only show up in the first page of the printed Excel report.	This will be fixed in the next iteration.
6874	The Excel report is always sorted by the Garden Name column by default, even if this column is not present in the report.	The client is fine with this, so it will be left as is.
N/A	Multiple email address can't be entered for one garden, but there is an existing garden that currently has two email addresses.	The client is fine with this, so it will be left as is.
N/A	The search functionality doesn't search the columns of numerical and date types.	The client is fine with this, so it will be left as is.

6.2.3 Adherence to Plan

In this iteration, we got behind schedule toward the beginning, but we were able to catch up most of the way. The only part of the plan we did not follow was that we ran out of time for implementing CR-12 and CR-13 (which are related capabilities). We plan on implementing these in the next iteration, and we will have the client prioritize them as part of the CCD.

There were a few different setbacks that caused the delay in the first iteration. The first issue was that we hadn't designed the database in detail last semester, so we had to spend time doing that before we could start coding. The second issue was that there was a delay in obtaining a testing server. This caused delays in testing and also in integration of the modules. The third issue is that we ran into some problems while coding the garden management portion of the system. The problems were resolved eventually, but they caused some delays in making progress. Since all of these problems have been resolved, we will not run into them in the next iteration.

6.2.4 Summary of CCD Results

Overall, the client was very pleased with system at the CCD. We received several requests for improvements from him, but these were mainly just minor user interface changes, along with a few things that he realized he should have told us differently. This indicates that we did a very good job of developing the system how he wanted it.

At this point, the main risk is being able to complete everything on time, since we have a small team. We are pretty sure that if we fully implement the dynamic column core capability (CR-12 and CR-13), we will not have time to implement the feedback from the CCD. Therefore, we have suggested adding about 10 extra columns instead, which should meet the client's needs in this area for the time being. That way, he can have this need met and still get all his feedback incorporated.

The tables below list the functionalities planned for the next iteration and all of the feedback we received from the client during the CCD, along with priorities. Currently, these are the priorities which the development team thinks reflect the client's viewpoint. However, we are meeting with him soon in order to make sure the priorities are correct, and also to confirm what he wants us to do about the dynamic column functionality.

Garden Locator:

Requirement	Priority
Address and Location Info should get more space (2-3 rows per field instead of 5+)	High
Get rid of Last Update field and Thomas Guide field on List of Gardens and PDF.	High
Remove scrolling div from List of Gardens page.	High
Remove Website column and instead turn the garden name into a link.	High
Change icons in Google Map: narrower icons	High
Left-Justify all fields on Garden List page	High
Add filtering to List of Gardens page.	Medium
The heading row should float on top while scrolling down	Medium
Pictures in pop-up bubbles	Medium
Add Street View link on in Google Map bubbles.	Low

Change icons in Google Map: small icons when zoomed out, large icons when zoomed in	Low
Display our garden map in other website's page.	Low

Garden Management:

Requirement	Priority
Show column headers above a row being edited. Do the same for the row used for adding.	High
Every page of the spreadsheet report should have just a header (Dore will provide header).	High
Add page numbers to spreadsheet report.	High
Have another role with read only rights - Database Viewer.	High
Add a button to clear the search and show all the gardens.	High
Password requirements should always be visible - not just an error message.	High
Change column order on Garden Management page (Dore will provide order). Order on Generate Report page should probably be changed to match as well.	High
Show all page numbers in pagination (no "..."). "First" and "Last" are not required. Have pagination show up at both top and bottom of the table.	Low
Leave maybe ten columns for users defined (if we don't have time to implement the dynamic columns functionality).	?