Life Cycle Plan (LCP)

Field Progress Web App

Team 04

Team Member	Role
Uche Uba	Project Manager
Mayank Kulkarni	Requirements Engineer
Sahithi Velma	System/Software Architect
Akanksha Diwedy	Operational Concept Engineer
Madhavi Shantharam	Life Cycle Planner
Aishwarya Joisa	Feasibility Analyst
Kevin Grimes	Website Maintainer, IIV&V, Quality Focal Point

Version History

Date	Author	Version	Changes made	Rationale
10/28/2019	Madhavi Shantharam	1.0	Initial draft	Initial draft of LCP
12/09/2019	Madhavi Shantharam	1.1	Added sections 6.2, 6.3	Submission for As-Built Package

Table of Contents

Table of Tables

Table of Figures

1. Introduction

1.1 Purpose of the LCP

The purpose of the LCP is to help complete the project on time without overrunning costs. At each phase, careful planning is carried out to eliminate any technical debts and reduce risks.

LCP document ensures that the team and clients have the same collective understanding of requirements and timeline to be followed. This helps in achieving the final aim of the project.

1.2 Status of the LCP

The status of the LCP is currently Operation phase preparing to hand-over the product to the clients.

1.3 Assumptions

- The duration of the project is 12 weeks in Fall 2019 semester
- We don’t have access to real time voter data. We are developing the algorithm assuming that the test data available with us is very similar to the real time voter information

2. Milestones and Products

2.1 Overall Strategy

The Field Progress Web App follows NDI/NCS strategy because we are using these services to develop our core capabilities.

Exploration phase

Duration: 09/11/2019 – 09/20/2019
Concept: Analyzing the proposed system and identification of success critical stakeholders. Identify software requirement, supporting technologies, COTS that could be used and conduct feasibility evidence
Deliverables: Website set up, Client interaction meeting notes
Milestone: Valuation Commitment Review

Valuation phase

Duration: 09/20/2019 – 10/04/2019
Concept: Define the project scope, develop operational concept, identify win conditions, analyze risk and prepare risk mitigation plan, prototype high risk item
Deliverables: Jira Weekly Survey, Project Plan, Progress Report, Risk and Defect Report, Win Conditions, Prototype, Cost Estimation Report
Milestone: Foundation Commitment Review

Foundations phase

Duration: 10/04/2019 – 10/25/2019
Concept: Analyze use-cases, produce system and software architecture, project plan, operational concept, feasibility evidence, further prototyping
Deliverables: Winbook, Technical Debt Report, Jira Weekly Survey, Project Plan, Progress Report, Risk and Defect Report, Development Commitment Package (OCD, SSAD, FED, LCP), Improved version of the prototype
Milestone: Development Commitment Review

Development phase

Duration: 10/25/2019 – 11/20/2019
Concept: Implement all the required capabilities and test them
Deliverables: Core Capability Drive Through report, Transition Readiness report, Jira Weekly Survey, Project Plan, Progress Report, Risk and Defect Report
Milestone: Operational Commitment Review

Operation phase

Duration: 11/20/2019 – 12/12/2019
Concept: Prepare for system delivery
Deliverables: Final product, necessary documentation

2.2 Project Deliverables

2.2.1 Exploration Phase

Table 1: Artifacts Deliverables in Exploration Phase

Artifact	Due date	Format	Medium
Client Interaction Report	9/13/2019	.pdf	Soft copy
Team Website	9/23/2019	Website	Soft copy
Win Conditions	9/26/2019	Winbook	Soft copy

2.2.2 Valuation Phase

Table 2: Artifact deliverable in Valuation Phase

Artifact	Due date	Format	Medium
Prototype Presentation Slides	10/04/2019	.pptx	Soft copy
Cost Estimation LCP	10/11/2019	.pdf	Soft copy
Cost Estimation V&V	10/11/2019	.pdf	Soft copy
Project Plan	Bi-weekly 09/25/2019	.mpp	Soft copy
Progress Report	Bi-weekly 09/25/2019	.xlsx	Soft copy
Risk and Defect Report	Bi-weekly 09/25/2019	.xlsx	Softcopy
Jira Weekly Survey	Weekly Monday	Jira ticket	Jira website

2.2.3 Foundations Phase

Table 3: Artifact deliverable in Foundations Phase

Artifact	Due date	Format	Medium
Jira Weekly Summary	Weekly Monday	Jira ticket	Jira Website
Project Plan	Bi-weekly 10/09/2019, 10/23/2019	.mpp	Soft copy
Progress Report	Bi-weekly 10/09/2019, 10/23/2019	.xlsx	Soft copy
Risk and Defect Report	Bi-weekly 10/09/2019, 10/23/2019	.xlsx	Soft copy

Team Technical Debt Report	10/23/2019	.xlsx	Soft copy
Development Commitment Review (DCR) ARB Slides	10/25/2019	.ppt	Soft copy
Development Commitment Package • Operational Concept Description (OCD) • Feasibility Evidence Description (FED) • System and Software Architecture Description (SSAD) • Life Cycle Plan (LCP)	10/28/2019	.doc, .pdf	Soft copy
Winbook and Prioritization report	10/15/2019	.xlsx	Soft copy
Win Conditions	10/21/2019	.pdf	Soft copy

2.2.4 Development Phase

Table 4: Artifact deliverable in Development Phase

Artifact	Due date	Format	Medium
Jira Weekly Summary	Weekly Monday	Jira ticket	Jira Website
Project Plan	Bi-weekly 11/06/2019, 11/20/2019	.mpp	Soft copy
Progress Report	Bi-weekly 11/06/2019, 11/20/2019	.xlsx	Soft copy
Risk and Defect Report	Bi-weekly 11/06/2019, 11/20/2019	.xlsx	Soft copy

2.2.5 Operation Phase

Table 5: Artifact deliverable in Operation Phase

Artifact	Due date	Format	Medium
Jira Weekly Summary	Weekly Monday	Jira ticket	Jira Website
Project Plan	Bi-weekly 12/04/2019	.mpp	Soft copy
Progress Report	Bi-weekly 12/04/2019	.xlsx	Soft copy
Risk and Defect Report	Bi-weekly 12/04/2019	.xlsx	Soft copy
Team Technical Debt Report	12/04/2019	.xlsx	Soft copy
Core Capability Drive Through Report (CCD)	11/25/2019	.pdf	Soft copy
As-Built Package	12/09/2019	.pdf, .docx	Soft copy
Project Archive	12/11/2019	.zip	Soft copy
Individual Critique Report	12/12/2019	.pdf	Soft copy

3. Responsibilities

3.1 Project-specific stakeholder's responsibilities

Stakeholders for Field Progress Web Application are Client, User, Maintainer, Developers and IIV&V.

3.2 Responsibilities by Phase

Table 5: Stakeholder's Responsibilities in each phase

Team Member / Role	Primary / Secondary Responsibility				
	Exploration	**Valuation**	**Foundations**	**Development-** Construction Iteration	**Development-** Transition Iteration
Name: Evan Shulman, Isaac Wang, Nikolaj Baer **Role:** Client	**Primary Responsibility** • Outline the requirements of the desired system • Attend Win-Win Negotiation Session 1	**Primary Responsibility** • Attend Win-Win Negotiation Session 2 **Secondary Responsibility** • Provide voters information based on which algorithm should be implemented	**Primary Responsibility** • Hold meetings with the development to monitor progress and re-align project goals according to project progress	**Primary Responsibility** • Provide feedback on DCR prototype and suggest improvements **Secondary Responsibility** • Provide necessary platform support for application development	**Primary Responsibility** • Participate in CCD to have hands experience on the system and provide feedback • Involve other stakeholders (if any, maintainers) and train them
Name: Uche Uba **Role:** Project Manager	**Primary Responsibility** • Understand the project scope. Get to know teams' strengths and weaknesses • Create biweekly project	**Primary Responsibility** • Create biweekly project plan, progress report, and risk and defect report • Assess high risk features to prototype	**Primary Responsibility** • Create biweekly project plan, progress report, and risk and defect report • Set up React framework and	**Primary Responsibility** • Help create a development schedule with LCP and ensure that the development progress	**Primary Responsibility** • Ensure that the transition plan is in place • Prepare the system to be delivered with necessary documentation

	plan, progress report, and risk and defect report • Assign tasks to team members	• Develop Personas for prototype presentation • Prototype implementation – render output of clustering algorithm on MapBox	develop front-end of the application • Plan weekly meetings for developing and working on ARB presentation	is on track to be completed before the deadline	
Name: Mayank Kulkarni **Role:** Requirements Engineer	**Primary Responsibility** • Analyze the system requirements from win-conditions	**Primary Responsibility** • Prototype - Implement algorithm to form clusters of voters based on number of volunteers • Integrate NDIs and ensure working of prototype • Set up GitHub repository	**Primary Responsibility** • ARB Prototype - Implement algorithm to form clusters of voters based on number of volunteers within given precincts • Integrate NDIs and ensure working of prototype	**Primary Responsibility** • Ensure that the ongoing development is in accordance with the requirements • Development of algorithm /user interface	**Primary Responsibility** • Ensure that the system to be delivered is according to the requirements specified • Prepare comprehensive documentation of the features implemented and not implemented (if any)
Name: Sahithi Velma **Role:** Software/System Architect	**Primary Responsibility** • Research different COTS, APIs, and other technologies to be used • Determine whether	**Primary Responsibility** • Prototype implementation – render output of clustering algorithm on MapBox	**Primary Responsibility** • Develop module architecture for the proposed system • Work on SSAD documentation for	**Primary Responsibility** • Ensure that the application being developed is in sync with the architecture and keep it up to date in case of any changes	**Primary Responsibility** • Review system architecture • Testing of the system

	the different NDIs are interoperable		DC Package	• Development of algorithm/user interface	
Name: Akanksha Diwedy **Role:** Operational Concept Engineer	**Primary Responsibility** • Understand requirements and win conditions • Begin developing the operational concept description	**Primary Responsibility** • Continue to work on the operational concept description as Win - Win conditions change and evolve	**Primary Responsibility** • Work on OCD documentation for DC Package • Select optimal NDI for application implementation • ARB Prototype - Integrate the backend framework with the front end • Prepare for prototype presentation	**Primary Responsibility** • Development of algorithm/user interface • Ensure that the objectives are met as the product is being developed	**Primary Responsibility** • Check whether all the objectives have been met • Prepare operation phase documentation
Name: Madhavi Shantharam **Role:** Life Cycle Planner	**Primary Responsibility** • Determine the responsibilities of each stakeholder • Start building the life cycle plan • Determine the deliverables of each milestone	**Primary Responsibility** • Discuss with team members to determine the values of factors and calculate the estimated effort using COCOMO II • Prototype - Implement algorithm to form clusters of voters based on	**Primary Responsibility** • Re-estimate efforts using COCOMO II • Create iteration plan for further iterations in LCP • Work on LCP documentation for DC Package	**Primary Responsibility** • Create development schedule for product implementation • Implementation of algorithm/user interface	**Primary Responsibility** • Testing of the product • Prepare transition plan to deliver the product to the clients

		number of volunteers • Setup system for prototype presentation			
Name: Aishwarya Joisa **Role:** Feasibility Analyst	**Primary Responsibility** • Determine the feasibility of features proposed by the client and verify whether it is feasible within the given time frame	**Primary Responsibility** • Analyze NDI and COTS • Determine high risk items and come up with a risk mitigation strategy • Select optimal NDI for implementation of prototype	**Primary Responsibility** • Work on FED documentation for DC Package • Research API s / NDIs that could be used for product implementation • Develop Personas for prototype presentation	**Primary Responsibility** • Implementation of algorithm/ user interface • Continuously check feasibility of the system	**Primary Responsibility** • Document NDI/NCS component of the product • Testing of the product
Name: Kevin Grimes **Role:** Website Maintainer, IIV&V, Quality Focal Point	**Primary Responsibility** • Build Team Website and keep it up to date • Review win conditions	**Primary Responsibility** • Ensure that the team website is up to date • Perform Independent Cost Estimation using COCOMO II • Understand the intended behavior of desired capabilities • Verify working prototype	**Primary Responsibility** • Verify and validate Jira tickets • Verify working prototype • Start preparing test plan to run unit and integration tests on developed modules	**Primary Responsibility** • Create a test plan • Review code • Perform unit and integration tests	**Primary Responsibility** • Test the application and ensure that it is ready for delivery • Prepare a quality management document for maintainers to help maintain the application

3.3 Skills

Team members	Role	Skills
Uche Uba	Project Manager	**Current Skills:** Python, JavaScript, Angular, Node **Required Skills:** Project Management, Python, Django, React, Deck.gl, MS Project, Jira
Mayank Kulkarni	Requirements Engineer	**Current Skills:** Python, Django, Winbook **Required Skills:** Python, Django, React, Deck.gl, Winbook
Sahithi Velma	System/Software Architect	**Current Skills:** Java, Python, Backend Development **Required Skills:** Python, Django, React, Deck.gl, Analyzing NDI interoperability, UML Modelling, Technical writing
Akanksha Diwedy	Operational Concept Engineer	**Current Skills:** Python, Django, Backend Development, ML/AI, Scala **Required Skills:** Python, Django, React, Deck.gl, Technical writing
Madhavi Shantharam	Life Cycle Planner	**Current Skills:** Python, Java backend development, JavaScript **Required Skills:** Project/Life Cycle Planning, MS Project, Python, Django, React, Deck.gl, Technical writing, COCOMO II
Aishwarya Joisa	Feasibility Analyst	**Current Skills:** Python, Backend Development

		Required Skills: Python, Django, React, Deck.gl, Technical writing
Kevin Grimes	Website Maintainer, IIV&V, Quality Focal Point	**Current Skills:** Python, Jira, Technical Writing, Test Planning **Required Skills:** Python, Django, React, Deck.gl, Jira, Technical writing, Test Planning

4. Approach

4.1 Monitoring and Control

We use Microsoft Project to prepare the project plan, create tasks in Jira to keep track of all the individual and team activities. We use GitHub repository to review and check-in code. Bi-Weekly Progress reports, and Risk and Defect reports are being prepared to monitor the progress of the project.

4.1.1 Closed Loop Feedback Control

Team uses Slack, Gmail to share ideas within the team and to facilitate communication with the clients. All resources and documents are uploaded to a shared Google Drive and to the team website. Also, team members meet at least once in a week to discuss project milestones, progress and provide feedback for each other's work.

4.1.2 Reviews

Team conducts bi-weekly meetings to discuss and review the current project plan and progress, identify any new risks and defects and, assess the goals and issues that needs to be resolved from previous report. We review each other's work, develop test cases to find bugs and improve code quality. We setup timely meetings with the clients to receive feedback on the ongoing work.

4.2 Methods, Tools and Facilities

Tools	Usage	Provider
Project Professional 2016	Prepare Bi-Weekly Project plan	Microsoft
Jira	Create tasks to keep track of planned work and monitor progress and to track defects	USC
COCOMO II	Estimate cost and efforts	USC
GitHub	Developers to collaborate on Software Development	GitHub
Microsoft Excel	To create Progress report, and Risk and Defect Report	Microsoft
Slack	Team and Client communication	Slack
Zoom Meetings	Video conferencing with DEN Students and Clients	Zoom
Winbook	Add/Prioritize Win Conditions	USC
Visual Paradigm	Create UML Models	USC
Balsamiq	Create UI Mockups	Balsamiq

React	JavaScript library for creating UI Interfaces	React
Django	Backend server	Django
Deck.gl/MapBox	MapBox is used to render clusters of voters on a map based view.Deck.gl is used to provide visualizations on top of MapBox based maps.	Open Source

5. Resources

- Estimated CSCI577a Effort: 7 team members at 12 hours/week for 12 weeks. Increased efforts per/team member, per/week with the increase in scope of the project (need for an input module) from previous cost estimation specification.
- Total estimated effort: 1008 hours
- Budget information: We currently don't have any defined budget i.e. budget is $0
- Project duration: 12 weeks
- Component modules: Input Module, Turf Cutting Module, Output Module and Testing Framework
- Programming language used: Python, JavaScript

Table 6: Module lists and SLOC of each module

No.	Module Name	Brief Description	SLOC
1	Input module	Provide an interface for the users to input volunteer information	500
2	Turf cutting module	Clustering of voter data within the given precincts	400
3	Output module (Visualization)	Provide an interface for the users to view the cut turfs visually in a map-based view	300
4	Testing framework (internal use)	Build a testing framework for validating the system for its effectiveness both module wise and integration with each other	200

Table 7: COCOMOII Scale Driver

Scale Driver	Value	Rationale
PREC	NOMINAL	Team is familiar with some technologies like usage of Google Maps API. However, there is a need for innovative data processing architectures, and algorithms to accommodate the requirements of this project
FLEX	NOMINAL	The Development team is given a lot of leeway in how they want to implement the project, but the final product must plug into Field Progress' existing tech stack. So, there is a need for Software conformance with external interface specifications
RESL	HIGH	All critical risk items, schedule and internal milestones are identified. However, there is a potential risk in implementing certain features cutting turfs based on walkability, availability of the volunteers, the type of terrain among others. We plan to have regular sync ups

		with clients and collaborative brainstorming to resolve this risk
TEAM	HIGH	We have the Win-Win negotiations in place to minimize the conflicts. Also, each stakeholder has considerable consistency of objectives and willingness to accommodate each other's objectives
PMAT	NOMINAL	We are at Maturity Level 1

Table 8: COCOMOII Cost Driver for Input Module

Cost Driver	Value	Rationale
RELY	HIGH	Turf cutting and Visualization modules depend on this module. The effect of software failure is high, as without input other modules cannot function
DATA	NOMINAL	Input to the system from user interface includes volunteer information like name, availability, walkability, and others Approx. 500 SLOC to provide an interface to take in this input data. Keeping an average window of 5 to 50 volunteers per transaction, test data will not exceed approx. 1200 Bytes. D/P = 2.4
DOCU	NOMINAL	Development process follows ICSM, the document for life-cycle needs is normal.
CPLX	NOMINAL	Simple UI component which takes in user input
RUSE	NOMINAL	Interface to take the input is specific to the application being designed.
TIME	NOMINAL	System should be able to take in user data in a considerable time. <50% of the available execution time
STOR	NOMINAL	Should have the capability to store the provided input as it in turn serves as an input to the algorithm. <50% of available storage
PVOL	LOW	Creates a dynamic array for the Volunteer info in React
ACAP	NOMINAL	Development team was quite good in gathering the requirements and producing a high-level design of the product to be developed
PCAP	NOMINAL	Development team was able to choose the COTS to be used in the project implementation in a collaborative way
PCON	NOMINAL	Not quite risky as it is a 12-week project and team members are committed to course guidelines
APEX	LOW	Development team has very little experience in building this kind of application
LTEX	NOMINAL	Few team members have prior experience in building Python applications. But we are learning React.js to implement front-end of the application and integrate it with the back-end
PLEX	NOMINAL	Few team members have experience working on Django. However, we are not familiar with React and integrating Django and React

TOOL	NOMINAL	Team is familiar in using some software tools like GitHub, Slack. However, tools like Microsoft Project, Jira is new for most of the team members
SITE	HIGH	Team consists of 6 on-campus and 1 off-campus student. Team collaboration has never been a problem as the team is proficient in using tools for video conferencing and exchanging emails for communication
SCED	NOMINAL	The schedule is set at 12 weeks and we are not planning on stretching out

Table 9: COCOMOII Cost Driver for Turf Cutting Module

Cost Driver	Value	Rationale
RELY	NOMINAL	If the turf cutting fails, campaign managers will have to resort to cutting the turfs manually. This could cost Field Progress quite a bit of money and reputation, as this functionality is a major piece of their offering. So, providing a reliable algorithm is quite important
DATA	VERY HIGH	Input to the system includes thousands of entries of voter data consisting of latitudes, longitudes and precincts and volunteer information like no. of volunteers, availability, capability etc. Approx. 400 SLOC to process the input and cut turfs. D/P > 1000
DOCU	NOMINAL	Because the development process follows ICSM, the document for life-cycle needs is normal.
CPLX	HIGH	This involves implementation of clustering algorithms with input parameters such as volunteer availability, terrain, walkability, and calls to APIs to obtain the distance and other information.
RUSE	NOMINAL	Algorithm should be designed in such a way that it could be plugged into Field Progress' existing stack
TIME	VERY HIGH	We will be running complex algorithms on considerably huge amount of data. Application is expected to use almost 85% of the available execution time
STOR	NOMINAL	Less than 50% of the available storage will be used
PVOL	LOW	As we are using Python libraries for implementation of algorithm, it may not be too dependent on the underlying platform
ACAP	NOMINAL	Development team was quite good in gathering the requirements and producing a high-level design of the product to be developed
PCAP	NOMINAL	Development team was able to choose the COTS to be used in the project implementation in a collaborative way
PCON	NOMINAL	Not quite risky as it is a 12-week project and team members are committed to course guidelines

APEX	LOW	Development team has very little experience in building this kind of application
LTEX	NOMINAL	Few team members have prior experience in building Python applications. But we are learning React.js to implement front-end of the application and integrate it with the back-end
PLEX	NOMINAL	Few team members have experience working on Django. However, we are not familiar with React and integrating Django and React
TOOL	NOMINAL	Team is familiar in using some software tools like GitHub, Slack. However, tools like Microsoft Project, Jira is new for most of the team members
SITE	HIGH	Team consists of 6 on-campus and 1 off-campus student. Team collaboration has never been a problem as the team is proficient in using tools for video conferencing and exchanging emails for communication
SCED	NOMINAL	The schedule is set at 12 weeks and we are not planning on stretching out

Table 10: COCOMOII Cost Driver for Output Module

Cost Driver	Value	Rationale
RELY	NOMINAL	Visualization is important as it provides the user to view the output of the algorithm i.e. turf cutting in a map view. Plan is to plug in the data into visualization tools like mapbox/deck.gl
DATA	HIGH	Test data would be GeoJSON list of voters returned by the turf cutting algorithm. Approx. 400 SLOC to process the response and render it on the map. Approx. D/P between 100 and 1000
DOCU	NOMINAL	Because the development process follows ICSM, the document for life-cycle needs is normal.
CPLX	HIGH	Involves integration of output from Django server with UI designed using React
RUSE	LOW	Required for verification during development cycle. May or may not be required after integrating with actual output module on Field Progress' tech stack
TIME	NOMINAL	Less than 50% of the available execution time
STOR	NOMINAL	Less than 50% of the available storage will be used
PVOL	NOMINAL	Browser and COTS upgrades might pose a potential risk. However, they are not prone to change very frequently
ACAP	NOMINAL	Development team was quite good in gathering the requirements and producing a high-level design of the product to be developed
PCAP	NOMINAL	Development team was able to choose the COTS to be used in the project implementation in a collaborative way

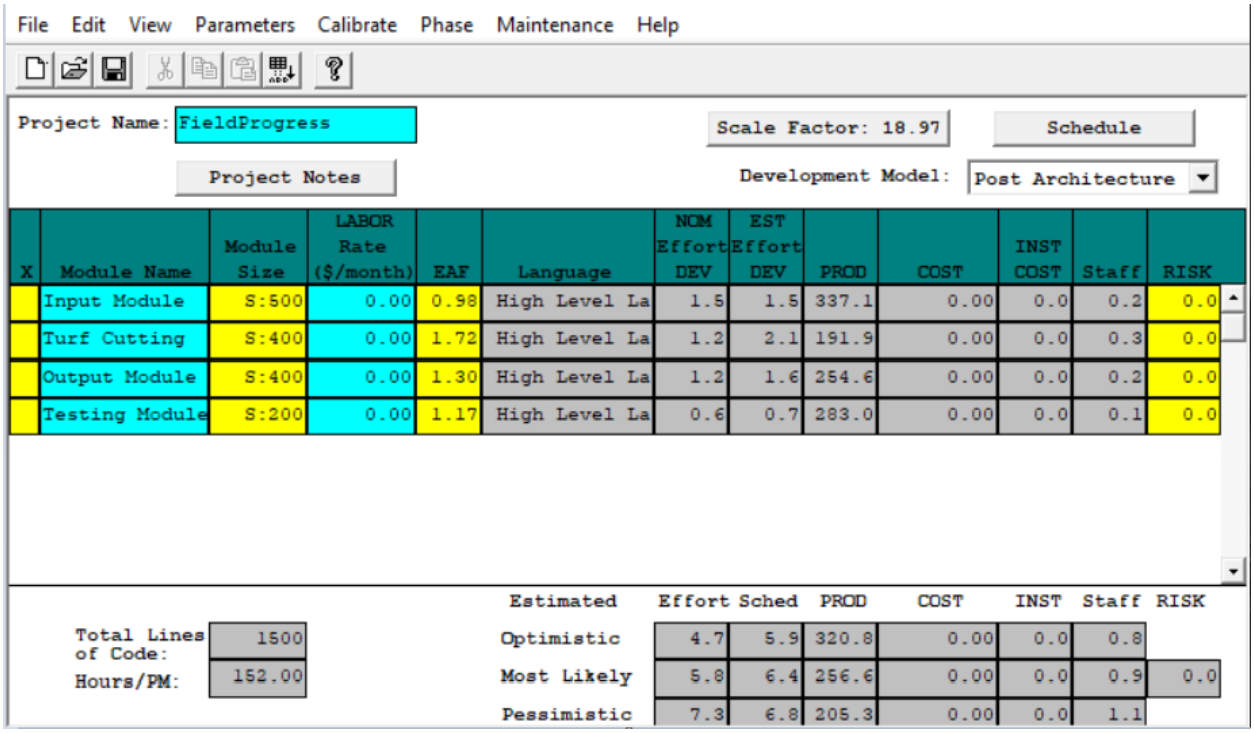
PCON	NOMINAL	Not quite risky as it is a 12-week project and team members are committed to course guidelines
APEX	LOW	Development team has very little experience in building this kind of application
LTEX	NOMINAL	Few team members have prior experience in building Python applications. But we are learning React.js to implement front-end of the application and integrate it with the back-end
PLEX	NOMINAL	Few team members have experience working on Django. However, we are not familiar with React and integrating Django and React
TOOL	NOMINAL	Team is familiar in using some software tools like GitHub, Slack. However, tools like Microsoft Project, Jira is new for most of the team members
SITE	HIGH	Team consists of 6 on-campus and 1 off-campus student. Team collaboration has never been a problem as the team is proficient in using tools for video conferencing and exchanging emails for communication
SCED	NOMINAL	The schedule is set at 12 weeks and we are not planning on stretching out

Table 11: COCOMOII Cost Driver for Testing Framework

Cost Driver	Value	Rationale
RELY	NOMINAL	Testing is a fairly important component as it defines the quality of the product. Testing framework must be good enough to catch any bugs and ensure that the product is working as expected
DATA	HIGH	Requires good amount of test data to test all modules of the product individually and integration between them. Approx. 200 SLOC of unit and integration test functions. Approx. D/P between 100 and 1000
DOCU	NOMINAL	Because the development process follows ICSM, the document for life-cycle needs is normal.
CPLX	NOMINAL	Unit tests to verify working of each module separately and integration with one another to test the end to end functionality
RUSE	NOMINAL	Could be reused/extended by maintainer if there comes a need to verify any issues or test any additional functionality
TIME	NOMINAL	Less than 50% of the available execution time
STOR	NOMINAL	Less than 50% of the available storage will be used
PVOL	NOMINAL	Test scripts are not too dependent on the underlying platform. However, it might need changes if the core functionality changes which is not highly prone to changes
ACAP	NOMINAL	Development team was quite good in gathering the requirements and producing a high-level design of the product to be developed

PCAP	NOMINAL	Development team was able to choose the COTS to be used in the project implementation in a collaborative way
PCON	NOMINAL	Not quite risky as it is a 12-week project and team members are committed to course guidelines
APEX	LOW	Development team has very little experience in building this kind of application
LTEX	NOMINAL	Few team members have prior experience in building Python applications. But we are learning React.js to implement front-end of the application and integrate it with the back-end
PLEX	NOMINAL	Few team members have experience working on Django. However, we are not familiar with React and integrating Django and React
TOOL	NOMINAL	Team is familiar in using some software tools like GitHub, Slack. However, tools like Microsoft Project, Jira is new for most of the team members
SITE	HIGH	Team consists of 6 on-campus and 1 off-campus student. Team collaboration has never been a problem as the team is proficient in using tools for video conferencing and exchanging emails for communication
SCED	NOMINAL	The schedule is set at 12 weeks and we are not planning on stretching out

Figure 1: COCOMO Estimation Result



Estimated Size: 1500 SLOC

Estimated Effort:
Optimistic: 4.7 person-month x 152 = 714.4 person-hr
Most Likely: 5.8 person-month x 152 = 881.6 person-hr
Pessimistic: 7.3 person-month x 152 = 1109.6 person-hr

Estimated Schedule:
Total work time of the whole team = 12 hr/week x 4 weeks x 7 team members = 336 hr/month
Optimistic: 714.4 / 336 = 2.12 month
Most Likely: 881.6 / 336 = 2.6 month
Pessimistic: 1109.6/ 336 = 3.3 month
Thus, most likely we need 2.6 month to complete this project which is no longer than the length of one semester. However, the actual work time may be longer than that. If each team member can stretch a little more than 12 hr/week, we can avoid falling into Pessimistic estimates (which is little longer than one semester) thereby not scoping down or stretching out on the project schedule.

Note: - This is a revised estimate after considering increase in scope and complexity from the previous version of cost estimation

6. Iteration Plan

6.1 Plan

As we move ahead to the Development Phase, we are planning on enhancing the current Prototype for Turf Cutting.

Development Phase will consist of two iterations. First being, implementation of all the remaining core capabilities. This includes design and implementation of an efficient algorithm to take volunteer info, voter info, and other parameters like likelihood of conversation, terrain among others as input and produce a sub list of volunteers mapped to voters. This output must be rendered on a map view which shows turfs cut within each precinct as area-covered by polygons (Visualization purposes only, not delivered as part of the project). This iteration also includes testing of the modules individually i.e. unit tests and all the modules integrated i.e. integration tests.

Second iteration aims at preparing the application to be handed over to the client. This involves exhaustive testing of the algorithm to ensure that all the desired functionalities are working as expected when the algorithm is plugged into Field Progress' tech stack.

6.1.1 Capabilities to be implemented

Table 12: Construction iteration capabilities to be implemented

ID	Capability	Description	Priority	Iteration
1	Implementation of algorithm	User should be able to generate turfs	Very High	1
2	Input interface	User should be able to provide volunteer information into the system	High	1
3	Output interface	User should be able to view the cut turfs on a map-based view	High	1

6.1.2 Capabilities to be tested

Table 13: Construction iteration capabilities to be tested

ID	Capability	Description	Priority	Iteration
1	Input Interface	Verify whether user can provide inputs using the input interface	Very High	1
2	Turf Cutting	Verify whether turfs are cut efficiently	Very High	1
3	Output Interface	Verify whether user can view the cut turfs on a map-based view with zoom in and zoom out options	High	1
4	End-End flow	Verify whether end-end flow of the app is working as desired	Very High	1
5	Performance	Verify efficiency of the algorithm in terms of execution time	High	2
6	Compatibility	Verify whether algorithm output when plugged into Field Progress's stack produces the desired result	Very High	2
7	API Testing	Since the main deliverable is the algorithm in the form of a library, verify whether the user can access this library to access/fetch the desired information	Very High	1

6.1.3 Capabilities not to be tested

All the capabilities mentioned above will be tested to ensure that performance of the algorithm is efficient both in terms of accuracy and execution time.

6.1.4 CCD Preparation Plans

- Implement and test all the features as per client's requirement specification. Ensure that the system is well tested and there are no bugs
- Prepare a demonstrational workflow for the client to use the system. This involves ensuring that integration of all the modules are in place
- During CCD, clients get a chance to use the application. They can input volunteer information into the said system via an input interface designed using React, generate turfs and view the cut turfs on a map generated using Deck.gl.
- Receive feedback from the clients and course faculty. Incorporate any necessary changes in accordance with client's feedback
- Prepare for the final delivery of the system

6.2 Iteration Assessment

6.2.1 Capabilities Implemented, Tested, and Results

Table 12: Capabilities implemented, tested and results

ID	Capability	Test Case	Test Results	If fail, why?
TC-01	Loading voter files into backend	TC-01-01 to TC-01-05	Pass	
TC-02	Changing number of volunteers	TC-02-01 to TC-02-03	Pass	
TC-03	Map Rendering	TC-03-01 to TC-03-03	Pass	
TC-04	Adding volunteer cards	TC-04-01, TC-04-02	Pass	
TC-05	User volunteer input	TC-05-01 to TC-05-04	Pass	
TC-06	Toggling input panel	TC-06-01, TC-06-02	Pass	

6.2.2 Core Capabilities Drive-Through Results

Overall, the CCD was successful. The client gave some positive feedback in pointing them in the right direction to solve the problem and on having put together different modules given the complexity of the problem and learning curve of relevant technologies. Although, we couldn't deliver all the core-capabilities, at this point in the development cycle, the clients were quite satisfied with the work we had done so far . They also suggested on how we could modularize the code and prioritize the remaining core-capabilities.

Continuing with the development cycle, we did extensive research and came up with an approximate solution as the given problem is NP-Hard. With this, we implemented all the core-

capabilities, integrated all the modules and performed a complete round of testing around all the modules to ensure that all the win-conditions were met, and product is bug free. We also, refactored the code to make it more modular as per the client's suggestion. In the process, we cleaned up the code base and fixed some bugs.

Table 13: Core capabilities drive-through results

Core Capability	Positive feedbacks	Improvements needed/suggested
Algorithm to cut turfs	(as mentioned by the clients) • The challenge was quite significant and difficult. They have done a good job of getting something together given the difficulty and learning curve of the relevant technologies • Their discovery is helpful in helping us frame the problem	(as mentioned by the clients) • The underlying setup of their codebase (the deliverable) was pretty rough. We expected a bit more structure

6.3 Adherence to Plan

Overall, the project is successful. We delivered the code and all the required documents to the client as per the plan. We implemented all the core-capabilities as discussed in win-win negotiations. However, the product was tested with limited data as

- real-time voter data was not available (it is confidential)
- monetary constraints in making huge number of API calls

The project was completed on time and the cost was kept within budget as most of the APIs and tools used were open-source or free versions.