Progress Report

1 Main Tasks Completed

* Architecture:
  + Description of subsystems and interfaces
  + Evaluation of architectural styles
  + Block Diagram
* Detailed Design:
  + Design-Level Class Diagrams for Android and Web Applications
  + Description and operations of entity, controller, boundary classes and DTOs
* Back-end Implementation of “Plant Tree”, “Cut Down Tree”, and “List All Trees” features:
  + Implementation of back-end service
  + Implementation of REST controller
  + Implementation of persistence
  + Implementation of DTOs
  + Implementation of validation checks
  + Implementation of test cases (for backend service + REST API)
* Front-end Implementation of “List All Trees” feature for Web Application
* Implementation-Level sequence diagrams for “Plant Tree” and “List All Trees”

2 Leadership roles

While every member of the team has a responsibility to ensure the overall success of this project, we have assigned roles that best suit the expertise of each member. This allows us ensure accountability and efficiency within the team.

**Project Team Leader:** *Ilana Haddad*

General Tasks

* Helps maintain schedule and organizational structure to team members and external influencers.
* Develops project plan to achieve project criteria.
* Schedules meetings, takes minutes at meetings and ensures project documentation is always in order.

Software Tasks

* Along with the CTO and Test Manager, implemented Java Spring backend implementation and developed test cases for the TreePLE test classes.

**Chief Technical Officer:** *Diana Serra*

General Tasks

* Oversees the entire project design and development teams and ensures interfacing of all software components.

Software Tasks

* Along with the project manager and the Test Manager, implemented Java Spring backend implementation and developed test cases for the TreePLE test classes.
* Designed the UMPLE diagram and contributed to the design-level class diagrams.

**Software Developer:** *Jessica Udo*

General Task:

* Along with the project manager, also manages the project documentation.

Software Tasks:

* Designed Implementation-level sequence diagram for the “List All Trees” method.
* Implemented the “List All Trees” use case (backend + web) components.

**Software Developer:** *Thomas Hannaford*

Software Tasks:

* Responsible for building the android front end and the web front end application for the TreePLE.
* Designed design level class diagrams for web & Android applications.

·

**Software Test Manager:** *Asma Alromaih*

Software Tasks:

* Tries to anticipate all the ways the application or system might be used and how it could fail. Responsible for preparing test scripts and macros and analyzing results.
* Designed the block diagram for the software architecture.
* Designed Implementation-level sequence diagram for the “Create Tree” method

3 Work Hours

3.1 Work Hours for Deliverable 1

|  |  |  |
| --- | --- | --- |
| **Members** | **Hours** | **Summary** |
| **Asma** | 10 | Overall structure of the document(1.5 hrs), brainstorming sessions(use case diagram sketch, requirements, Umple model sketch)(5hrs), Umple model & class diagram(2hr), editing requirements(1.5hr). |
| **Diana** | 11 | Two group meetings for requirements, use case diagram sketch, and domain model (5 hours), Use Case Diagram descriptions (5 hours) and helping with others' tasks (1 hour) |
| **Ilana** | 15 | Two group meetings for requirements, use case diagram sketch, and domain model (5hrs), State Chart (5hrs), Use Case Diagram completion (2hrs), contributing to others’ tasks (3hrs) |
| **Jessica** | 9 | Two group meetings (5hrs), writing out documentation for meeting logs and progress report (3hrs), editing and completing deliverable (1hr). |
| **Thomas** | 11 | Two group meetings consisting of brainstorming sessions on the use case diagrams, requirements and the domain model (5 hrs), activity diagrams (3 hrs), various help and editing (3 hrs) |

3.2 Work Hours for Deliverable 2

|  |  |  |
| --- | --- | --- |
| **Members** | **Hours** | **Summary** |
| **Asma** | 27.5 | Overview of the project and tasks – Group meeting (1hr), Design level class diagrams for web & Android– Group meeting(7hrs), Block diagram (6hrs), Android front-end basics (2hrs), Architectural style  documentation (10hrs), Sequence diagram for “createTree” (1.5hrs) |
| **Diana** | 37.5 | Group Meetings (14 hrs) Back-end implementation of controllers,services and tests, (10 hrs), implementation of REST services (10.5 hrs), touch ups on JUnit tests (3hrs) |
| **Ilana** | 41 | Group Meetings 1-4 (14hrs), Draw.io the Design-Level Class diagrams (3hrs). Back-end implementation of controllers, services, and tests (10hrs), implementation of REST services (5.5 hrs), Final touches of web app + descriptions for boundary, entity, controller classes (6hrs), Touch ups on sequence diagrams, formatting, writing work plan for this deliverable and future iterations (2.5hrs) |
| **Jessica** | 22 | Group Meetings 1-4 (12hrs). Contribution to the back-end implementation (2hrs), and sequence diagrams (4hrs). Documentation (4hrs). |
| **Thomas** | 24 | Overview of the project and tasks – Group meeting (1hr), Design level class diagrams for web & Android– Group meeting(7hrs), Web Front-End(11hrs), Web app + descriptions for boundary, entity, controller classes (4hrs), Editing and formatting(1hr) |

4 Key Design Decisions

4.1 Deliverable 1

When formulating our requirements and designing our diagrams we had to make several key design decisions in order to clarify any ambiguities and fully understand the needs of the client.  Our design decisions were therefore taken based on the two following principles:

·       The Android app would be primarily used to enter information and data relating to each new or existing tree

·       The Web application would be primarily used to browse, filter and compile reports on all the information and data entered by users in the Android app

**Interface and functionalities**

Using these principles, some of the key design decisions taken that were not explicitly outlined in the requirements include:

·       The web application will include a map rather than a list for a more aesthetic user interface.

·       In order to offer as much flexibility as possible when compiling reports by area/location, the user will be able to select the area they want to analyze by delimiting a perimeter on the map. This will use the trees’ information relating to latitudes and longitudes.

·       Trees that are cut down will remain in the system as historical data. They are therefore not ‘deleted’ or suppressed from the system. Our system currently does offer a ‘delete’ functionality for mistakes should a user have entered the wrong information. We will assess whether this is needed in subsequent meetings.

**Use Cases**

·       We have designed two separate use case diagram for the *Web* and *Android* applications for simplicity/

·       There are two actors included in the android use diagram: The *Resident* and the *Professionals. As per,* the project description document, these are the users with access to the TreePLE from the android application.

·       Since, the forester, scientist and municipal arborist have similar access rights, we decided to categorize them together for our Android use case diagram.

·       Only the Professionals, however, have access to the TreePLE system from the Web front end, as indicated in the Web use case diagram.

·       Each user acts upon multiple use cases respective to their functionalities mentioned in the Requirements section of this document.

**Activity Diagrams**

·       Each user, whether a local resident or professional is required to login before accessing the system.

·       This log in allows us authenticate the user. Moreover, since it is required that only local residents can mark tree as planted or cutdown, it also allows us validate the location of the user and the ownership of a tree by the user.

**Sequence Diagram**

·       For convenience and concision, we decided to design sequence diagrams for each main user action.

·       The sequence diagram may contain actions from multiple user classes to emphasize the sequence.

4. 2 Deliverable 2:

Changes to domain model:

* Add Municipality class in order to list all trees in that municipality, change attribute in tree accordingly.
* Change Local class to LocalResident
* Change Report class to SustainabilityReport
* Change location attributes to associations with multiplicity 4: associate to LocalResident and SustainabilityReport
* Add Forecast class:
  + Associate it to Location with multiplicity 4 in order to get a rectangle of area to forecast
  + Associate to tree: existingTrees, treesToCutDown, treesToPlant
  + Associating new Municipality class to Forecast class
* Add Version class that is associated to Tree, Forecast, and SustainabilityReport
* Added a TreeManager class that is associated to Tree, Survey, User, Municipality, Location

5 Meeting Logs

*5.1 Deliverable 1 Meetings*

*5.1.1 Meeting 1*

Date: February 6, 2018

Time: 7:30-9:30pm - 2hrs

Location: Trottier 5103

Purpose: Deliverable 1 preparation

Members: Everyone

Meeting minutes:

·       Introductions

·       Overview

·       Listing of requirements

·       Created use models for Android and Web

·       Action Items for next meeting

·       We should all come up with our own domain models

·       Brainstorm the other models to be completed in Deliverable 1

·       Next Meeting

·       Scheduled for Saturday, February 10th @ 16:00

*5.1.2 Meeting 2*

Date: February 10, 2018

Time: 4:00-7:00pm (3hrs)

Location: Trottier 5103

Purpose: Deliverable 1 preparation (continued)

Members: Everyone

Agenda:

* Compare all our individual domain models and amalgamate them into one model
* Draw our use case diagrams on draw.io & describe in natural language
* Activity diagram
* Traceability between diagrams and requirements

Meeting minutes:

* Finished domain model and began defining it with *Umple*
* Collaboratively created class diagram, activity diagrams, use case diagrams and state charts.
* Assigned remaining tasks for the "D*eliverable 1"* document
* Tasks Assigned:

o   Convert hand-written use case diagrams to digital format: *Ilana*

o   Convert domain model into Umple: *Asma*

o   Change format of requirements to table + trace them to use cases: *Asma*

o   Write use cases description for each use case: *Diana*

o   Create domain-level state-chart: *Ilana*

o   Create requirement-level activity diagram: *Thomas*

o   Write progress report: *Jess*

o   Complete meeting logs in project documentation: *Jess*

***5.2 Deliverable 2 Meetings***

*5.2.1 Meeting 1*

Date: Friday February 16, 2018

Time: 17:30-18:30 – 1 hour

Location: Trottier Building, Room 5106

Purpose: Deliverable 2 overview and allocation of tasks

Members: Everyone

Meeting minutes:

* Reading over requirements for deliverable 2
* Allocating tasks for Architecture and Detailed Design
* Work plan for next week:
  + Finish Architecture and Detailed Design (40% of deliverable) by end of Tuesday (Feb 20th) night, using Tuesday’s office hours for help if needed
* Next Meeting
  + Scheduled for Monday, February 19th @ 15:30

*5.2.2 Meeting 2*

Date: Monday February 19, 2018

Time: 15:30-19:30 – 4 hours

Location: Trottier Building, Room 5106

Purpose: Completing 40% of deliverable 2: Architecture and Detailed Design

Members: Ilana, Thomas, Diana, Asma

Agenda: Everyone works on their task individually but also in coordination with others for help, guidance, and conformity

Meeting minutes:

* Creating docs in github
* Updating class diagram and key design decisions
* Updating use case diagrams
* Generated java code from .ump file with updated domain model
* Created project in eclipse and linked it to github
* Work plan for next meeting:
  + Asking all our questions during Tuesday’s office hours

*5.2.4 Meeting 3*

Date: Tuesday February 20, 2018

Time: 16:30-22:30 – 6 hours

Location: McConnell 627 & Schulich 5th floor

Purpose:  Office hours + design-level class diagrams

Members: Everyone

Agenda: Ask questions relating to deliverable 2

Meeting minutes:

* During office hours
  + Clarification relating to forecasting. Suggestion that there needs to be a version control for each model whether it is the current situation or future models. Sustainability reports can then be evaluated based on the model that is being pointed to. This version control would also allow to create forecasting of hypothetical models and not only forecasting of the current model (Russian-doll forecasting)
  + Difference between creating a tree and planting a tree
  + Should try and reduce redundancies because they are essentially the same function
  + ca.mcgill.ecse321.TreePLE.model instead of TreePLE.model is best practice
  + Clarification relating to architecture block diagrams
* Schulich group meeting
  + Completed design-level class diagram for Android and Web

*5.2.4 Meeting 4*

Date: Wednesday February 21, 2018

Time: 20:30-23:30 – 3 hours

Location: McLennan Library, Room M3-17B

Purpose: Begin Development of Features

Members: Everyone

Agenda: Start Implementation of features

Meeting minutes:

Splitting up tasks for implementation:

* Asma: Android front-end for plant tree and cut down tree, block diagrams
* Thomas: Web front-end for list all trees, 2 sequence diagrams
* Ilana: all of backend implementation for plant tree
* Diana: all of backend implementation for cut down tree
* Jess: all of backend implementation for list all trees

*5.2.5 Meeting 5*

Date: Thursday February 22, 2018

Time: 14:30-15:30 – 1 hour

Location: Trottier Building, Room

Purpose:  Office Hours with Mentor

Members: Diana, Ilana, Thomas

Agenda: Ask questions about class diagrams (design-level and domain model)

Meeting minutes:

* Asked our mentor for guidance, tips, and help confirming our class diagram was headed in the right direction

*5.2.6 Meeting 6*

Date: Thursday February 22, 2018

Time: 15:30-00:30 – 9 hours

Location: Trottier 5th floor

Purpose: Implementation of back-end

Members: Diana, Ilana, Asma

Agenda: Work together to implement back-end needed for features

Meeting minutes:

* Asma: finished block diagram
* Diana and Ilana: setting up controllers, services, DTOs

*5.2.7 Meeting 7*

Date: Friday February 23, 2018

Time: 13:00-18:30 – 5.5 hours

Location: Schulich 5th floor

Purpose: Implement REST services for back-end for createTree

Members: Diana and Ilana

Agenda: finish implementing REST services

Meeting minutes:

* Wrote POST requests for creating a new tree, a new municipality
* Wrote GET requests for all municipalities

*5.2.8 Meeting 8*

Date: Saturday February 24, 2018

Time: 18:00-00:00 – 6 hours

Location: Redpath Library Main Floor

Purpose:  Wrap-up deliverable

Members: Everyone

Agenda:

* Showcase our individual progress so far
* Re-allocate tasks to finish deliverable requirements

Meeting minutes:

* Implemented ListAllTrees GET request + service
* Connected back-end with front-end
* Tested Web app, made minor design changes, completed implementation of list all trees feature for web app
* Wrote descriptions + operations for both design-level class diagrams
* Finished TestSurveyService and TestTreeManagerService