

SRS  
Digital Twin Forest

Yichen Jiang, Bowen Zhang, Jiacheng Wu, Junhong Chen, Tingyu Shi  
Team 8

March 30, 2023

Table 1: Revision History

Date	Developer(s)	Change
Sept 24, 2022	All team members	Initial Document
Sept 26, 2022	All team members	Update Requirements
Sept 29, 2022	All team members	Revision 0
March 29, 2023	All team members	Final Version

This document follows **Volere Template**. The following are some modifications that we made to the original template:

- We added traceability matrices to show relationships between functional requirements and non-functional requirements.
- We added traceability matrices to show relationships between requirements and use cases.
- We added Likely changes and Unlikely changes.
- We added priorities and completion timelines for each functional requirement.

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>6</b>
1.1	The purpose of the Project . . . . .	6
1.1.1	The User Business or Background of the Project Effect . . . . .	6
1.1.2	Goals of the Project . . . . .	6
1.2	Stakeholders . . . . .	6
1.2.1	The client . . . . .	6
1.2.2	The Customer . . . . .	6
1.2.3	Other Stakeholders . . . . .	6
1.2.4	The Hands-On Users of the Product . . . . .	6
1.2.5	Personas . . . . .	7
1.2.6	Priorities Assigned to Users . . . . .	7
1.2.7	User Participation . . . . .	7
1.2.8	Maintenance Users and Service Technicians . . . . .	7
<b>2</b>	<b>Project Constraints</b>	<b>8</b>
2.1	Mandated Constraints . . . . .	8
2.1.1	Solution Constraints . . . . .	8
2.1.2	Implementation Environment of the Current System . . . . .	8
2.1.3	Partner or Collaborative Applications . . . . .	8
2.1.4	Off-the-Shelf Software . . . . .	8
2.1.5	Anticipated Workplace Environment . . . . .	8
2.1.6	Schedule Constraints . . . . .	8
2.1.7	Budget Constraints . . . . .	9
2.1.8	Enterprise Constraints . . . . .	9
2.2	Naming Conventions and Definitions . . . . .	9
2.3	Relevant Facts and Assumptions . . . . .	10
2.3.1	Relevant Facts . . . . .	10
2.3.2	Business Rules . . . . .	10
2.3.3	Assumptions . . . . .	10
<b>3</b>	<b>Functional Requirements</b>	<b>11</b>
3.1	The Scope of the Work . . . . .	11
3.1.1	The Current Situation . . . . .	11
3.1.2	The Context of the Work . . . . .	12
3.1.3	Work Partitioning . . . . .	13
3.1.4	Specifying a Business Use Case (BUC) . . . . .	14
3.2	Business Data Model and Data Dictionary . . . . .	15
3.3	The Scope of the Product . . . . .	16
3.3.1	Product Boundary . . . . .	16
3.3.2	Product Use Case Table . . . . .	16
3.4	Functional Requirements . . . . .	17
3.5	Priority and Timeline . . . . .	20

<b>4</b>	<b>Nonfunctional Requirements</b>	<b>21</b>
4.1	Look and Feel Requirements . . . . .	21
4.1.1	Appearance Requirements . . . . .	21
4.1.2	Style Requirements . . . . .	21
4.2	Usability and Humanity Requirements . . . . .	21
4.2.1	Easy of Use Requirements . . . . .	21
4.2.2	Personalization and Internationalization Requirements . . . . .	21
4.2.3	Learning Requirements . . . . .	22
4.2.4	Understandability and Politeness Requirements . . . . .	22
4.2.5	Accessibility Requirements . . . . .	22
4.3	Performance Requirements . . . . .	23
4.3.1	Speed and Latency Requirements . . . . .	23
4.3.2	Safety-Critical Requirements . . . . .	23
4.3.3	Precision or Accuracy Requirements . . . . .	23
4.3.4	Reliability and Availability Requirements . . . . .	23
4.3.5	Robustness or Fault-Tolerance Requirements . . . . .	24
4.3.6	Capacity Requirements . . . . .	24
4.3.7	Scalability or Extensibility Requirements . . . . .	24
4.3.8	Longevity Requirements . . . . .	24
4.4	Operational and Environmental Requirements . . . . .	24
4.4.1	Expected Physical Requirements . . . . .	24
4.4.2	Requirements for Interfacing with Adjacent Systems . . . . .	25
4.4.3	Productization Requirements . . . . .	25
4.4.4	Release Requirements . . . . .	25
4.5	Maintainability and Support Requirements . . . . .	26
4.5.1	Maintenance Requirements . . . . .	26
4.5.2	Supportability Requirements . . . . .	26
4.5.3	Adaptability Requirements . . . . .	27
4.6	Security Requirements . . . . .	27
4.6.1	Access Requirements . . . . .	27
4.6.2	Integrity Requirements . . . . .	27
4.6.3	Privacy Requirements . . . . .	28
4.6.4	Audit Requirements . . . . .	29
4.6.5	Immunity Requirements . . . . .	29
4.7	Cultural and Political Requirements . . . . .	29
4.7.1	Cultural Requirements . . . . .	29
4.7.2	Political Requirements . . . . .	29
4.8	Legal Requirements . . . . .	29
4.8.1	Compliance Requirements . . . . .	29
4.8.2	Standards Requirements . . . . .	29
<b>5</b>	<b>Project Issues</b>	<b>30</b>
5.1	Open Issues . . . . .	30
5.2	Off-the-Shelf Solutions . . . . .	30
5.3	New Problems . . . . .	30
5.4	Tasks . . . . .	30

5.4.1	Project Planning . . . . .	30
5.4.2	Planning of the Development Phases . . . . .	30
5.5	Migration to the new Product . . . . .	30
5.6	Risks . . . . .	31
5.7	Costs . . . . .	31
5.8	User Documentation and Training . . . . .	31
5.9	Waiting Room . . . . .	31
5.10	Ideas for Solutions . . . . .	31
<b>6</b>	<b>Traceability Matrix</b>	<b>32</b>
6.1	Traceability between FRs and NFRs . . . . .	32
6.2	Traceability Between FRs and Use Cases . . . . .	36
6.3	Traceability Between NFRs and Use Cases . . . . .	37
<b>7</b>	<b>Likely Changes</b>	<b>39</b>
<b>8</b>	<b>Unlikely Changes</b>	<b>39</b>
<b>A</b>	<b>Reflection Appendix</b>	<b>40</b>

## List of Tables

1	Revision History . . . . .	2
2	Business Event List . . . . .	13
3	Functional Requirements Priority Table . . . . .	20
4	Traceability Matrix 1 . . . . .	32
5	Traceability Matrix 2 . . . . .	33
6	Traceability Matrix 3 . . . . .	33
7	Traceability Matrix 4 . . . . .	34
8	Traceability Matrix 5 . . . . .	34
9	Traceability Matrix 6 . . . . .	35
10	Traceability Between Use Cases and Functional Requirements . . . . .	36
11	Traceability Between Use Cases and Non-Functional Requirements(Table 1) . . . . .	37
12	Traceability Between Use Cases and Non-Functional Requirements(Table 2) . . . . .	38

## List of Figures

1	Context of the Work Diagram . . . . .	12
2	An activity diagram for the Data Visualization process . . . . .	14
3	Data Hierarchy . . . . .	15
4	Use Case Diagram . . . . .	16

# 1 Project Drivers

## 1.1 The purpose of the Project

### 1.1.1 The User Business or Background of the Project Effect

A digital twin is a virtual representation of the real forest located at Turkey Point, Ontario, it includes physical objects, processes, relationships, and behaviors. Elements of a digital twin include data capture and integration, visualization, advanced analysis including AI, automation, and information sharing and collaboration. This project can be beneficial for two groups of users. The first group of users is forest owners. This project can help them to manage the forest. The second group of users is meteorologists. This project can help them to do research.

### 1.1.2 Goals of the Project

- A virtual representation of the real forest, allowing monitoring and analyzing from distance.
- Visualizing important data related to scientific research and decision-making. For the environment, important data include temperature, humidity, etc. For tree parameters, important data include tree density, tree heights, tree diameters, etc.
- A forest model that can change dynamically according to the modification of the data.

## 1.2 Stakeholders

### 1.2.1 The client

- Dr. Alemu Gonsamo from School of Earth, Environment and Society McMaster University (Dr. Gonsamo is the supervisor of this project.)

### 1.2.2 The Customer

- Forest Owners(The final project can be helpful for forest owners to better manage the forest and make decisions)
- Meteorologists(The final product can be helpful for researchers to study climate change)

### 1.2.3 Other Stakeholders

- Dr. Gonsamo's lab members(Graduate students from the lab will provide suggestions and data needed to assist this project)

### 1.2.4 The Hands-On Users of the Product

The hands-on users of this product are the same as the customers mentioned in section 1.2.2. Users' responsibilities are also mentioned in section 1.2.2. For subject matter experience, these users are masters. For both forest owners and meteorologists, they are definitely familiar with the real forest and our product can help them better do their jobs. From the point of technology, we assume that they have little experience with virtual representation technologies.

### 1.2.5 Personas

- Dr. Aly is a 55-year-old assistant professor working in a university, living with his family near his lab. His research mainly focuses on remote sensing of vegetation, global change ecology, and climate change impact. His research is supported by an association of several forest owners, who are willing to let Dr. Aly locate part of the research and collect needed information in their forest. If he needs to go to the field that day, the typical schedule of him is to get up at 6 am, take 90 minutes to commute to the target forest with some students he supervises, collect and record data there, and come back to his lab with another 90 minutes.

Dr. Aly's goals are to find a convenient way to free him from frequent commuting and to look for a new method to manage and visualize all the significant data for a purpose of teaching.

- Mrs. Miller owns 40-acre land including a 2-acre forest. She is now 36 years old and lives with her husband, who works as a financial analyzer, in Toronto. She runs a coffee house as a full-time job and drives to her forest every two months to check everything's fine. Though she cares about her forest, spending over three days every two months going through every area seems impossible for her, she would love to have a new method to manage her forest and get updated information regularly, in order to make better strategies for the forest.

### 1.2.6 Priorities Assigned to Users

- Key Users: Dr. Gonsamo, Forest owners, meteorologist
- Secondary Users: Lab members

### 1.2.7 User Participation

- Dr. Gonsamo: Dr. Gonsamo will provide business knowledge(forest data), interface prototyping and usability requirements for this project. Since Dr. Gonsamo is a meteorologist, he can also provide suggestions for meteorologists.
- Lab members: Lab members will help this project by providing some business knowledge about the forest.
- Forest Owners: Forest owners can provide commercial data about the forest. Commercial data include tree-cutting data, annual profits, etc.

### 1.2.8 Maintenance Users and Service Technicians

The project members will be responsible for maintaining and changing the product.

## 2 Project Constraints

### 2.1 Mandated Constraints

#### 2.1.1 Solution Constraints

- Modeling Technology: Our team will use Unity for the modeling task. The Unity version will be 2021.3. Our team members use different platforms(Windows or MacOS). Unity is suitable for our team because it is a cross-platform software. Also, Unity has existing packages that can speed up our modeling process.
- Project Documents Technology: Our team will use  $\text{\LaTeX}$  for our documents.
- Project Cooperation Technology: Our team will use GitHub to cooperate.

#### 2.1.2 Implementation Environment of the Current System

The product will be installed on Windows and MacOS operating systems.

#### 2.1.3 Partner or Collaborative Applications

N/A(Our product is an independent product)

#### 2.1.4 Off-the-Shelf Software

- LiDAR: iPad Pro LiDAR should be used to scan the forest and collect data for modeling.
- Unity: Unity should be used for modeling and post-processing.
- ArcGIS: An online tool to visualize forest distribution and tree data. The team can calculate the the density of different tree species and the formulas according to the tree distribution based on the satellite pictures.

#### 2.1.5 Anticipated Workplace Environment

Both indoor and outdoor are possible workplace environments.

#### 2.1.6 Schedule Constraints

Please check our project schedule [here](#). The following are known deadlines:

- SRS Rev0 (October 5)
- Hazard Analysis 0 (October 19)
- V&V Plan Rev0 (November 2)
- POC Demo (November 14)
- Design Document Rev0 (January 18)
- Demonstration Rev0 (February 14)



- V&V Report Rev0 (March 8)
- Final Demo (April 1)
- Final Documentation (April 5)

### **2.1.7 Budget Constraints**

Total expenses should not exceed \$750.

### **2.1.8 Enterprise Constraints**

N/A This project is not invested by any enterprise.

## **2.2 Naming Conventions and Definitions**

- LiDAR: Light Detection and Ranging(Scanning Technology)
- Plot: A square-shaped area in the forest. There are 14 plots in total.
- LAI: Leaf Area Index
- DBH: Diameter at Breast Height
- Target Forest: Target forest refers to the natural forest located at Turkey Point in Ontario.
- Digital Twin Forest: The virtual representation of the target forest.
- Forest Data: Forest Data include environmental data and tree parameters.
- Environmental Data:
  - Humidity
  - Temperature
  - Soil Carbon Content
  - Soil Nitrogen Content
  - LAI
- Tree parameters:
  - Density
  - Height
  - Age
  - DBH
- Tree types. There are 7 different types of trees, the following are the details:
  - Red Pine
  - Oak

- Birch
- Beech
- White Pine
- Red Maple
- Red Oak

## **2.3 Relevant Facts and Assumptions**

### **2.3.1 Relevant Facts**

- Forest data keep changing all the time, but our system will only update weekly. Therefore, the system may not be able to reflect the most accurate data. However, since the forest data will not change dramatically within a short period of time, the system should be close to the real forest.
- The product can be beneficial for meteorologists to do research.
- The product can be beneficial for forest owners to better manage the forest.

### **2.3.2 Business Rules**

N/A

### **2.3.3 Assumptions**

- Unity and required toolboxes should be accessible to all developers' devices.
- Users will use our product on Windows and MacOS.
- Forest data will not change dramatically within a short period of time.
- Related data to the forest shown in the product will be manually updated by the users.

## 3 Functional Requirements

### 3.1 The Scope of the Work

#### 3.1.1 The Current Situation

Currently, digital twin technologies are being utilized in many areas, such as Aerospace and Aeronautics, manufacturing, automotive, etc. For more detailed information, please check paper *Applications of Digital Twin across Industries: A Review* in our reference folder. This paper talks about many applications of digital twin technologies in different areas.

The following is the reference of the above paper:

Singh, M., Srivastava, R., Fuenmayor, E., Kuts, V., Qiao, Y., Murray, N., & Devine, D. (2022). Applications of Digital Twin across industries: a review. *Applied Sciences*, 12(11), 5727.

After searching online, we found a similar product from Tietoery Company called **Digital Forest Twin**. According to its introduction, this product can be accessed using head-mounted devices or web browsers, which is different from our product. Also, from the company's website, we can know little details about how this product works. Therefore, our team does not know too much about existing business processes.

### 3.1.2 The Context of the Work

The following is the context of the work diagram:

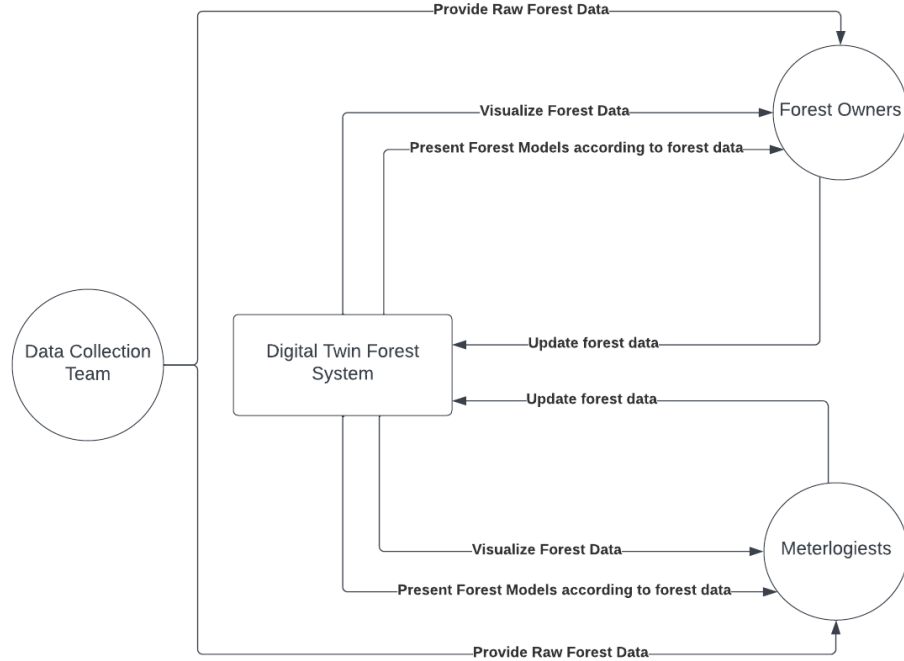


Figure 1: Context of the Work Diagram

Forest data include the following:

- Environmental Data:
  - Humidity
  - Temperature
  - Soil Carbon Contents
  - Soil Nitrogen Contents
  - LAI
- Tree Parameters:
  - Density
  - Age
  - Height
  - DBH

### 3.1.3 Work Partitioning

Table 2: Business Event List

Event Name	Input and Output	Summary of BUC
Update Forest Data	Forest Data(in)	After the system receives new data, it should store the data in the Data Store.
Data visualization	Forest Data(output)	The system demonstrates the organized forest data to the users. The data presented should be consistent with the data in the Data Store.
Model Demonstrations	Forest Models(output)	The system demonstrates the forest models to the users. The models presented should be generated according to the data stored in the Data Store

### 3.1.4 Specifying a Business Use Case (BUC)

The following is an activity diagram for the Data Visualization process.

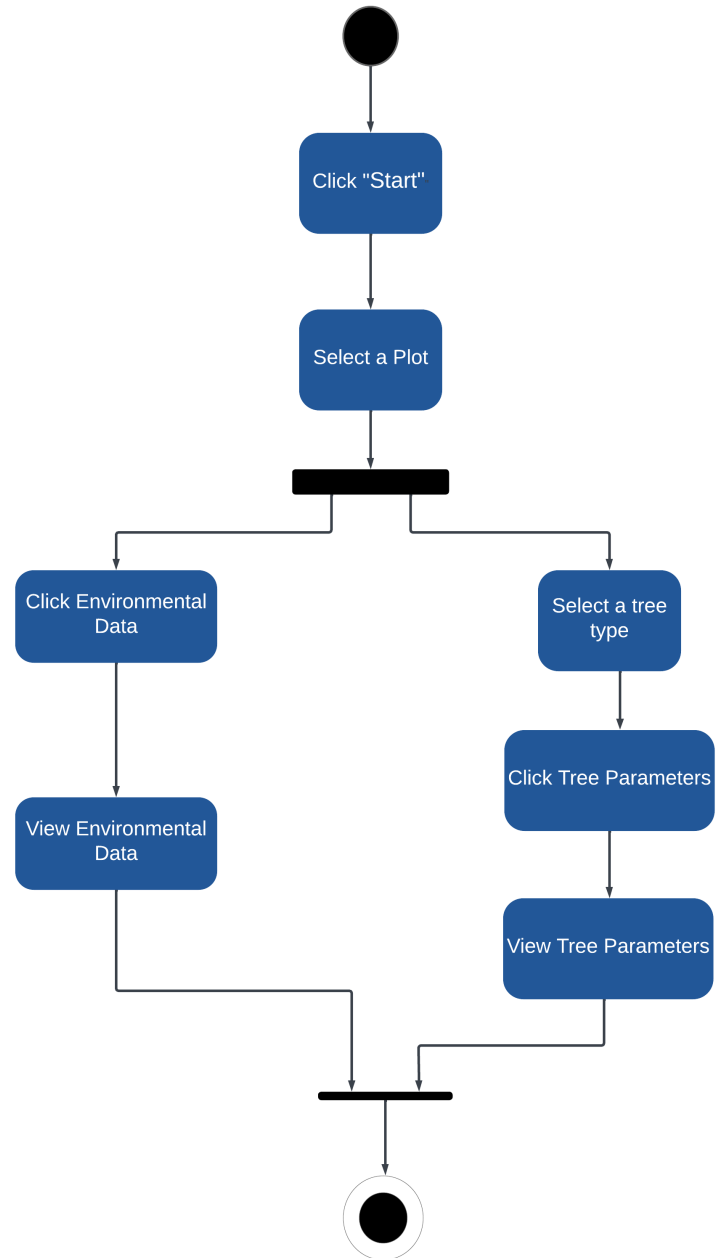


Figure 2: An activity diagram for the Data Visualization process

### 3.2 Business Data Model and Data Dictionary

The following picture shows the forest data hierarchy:

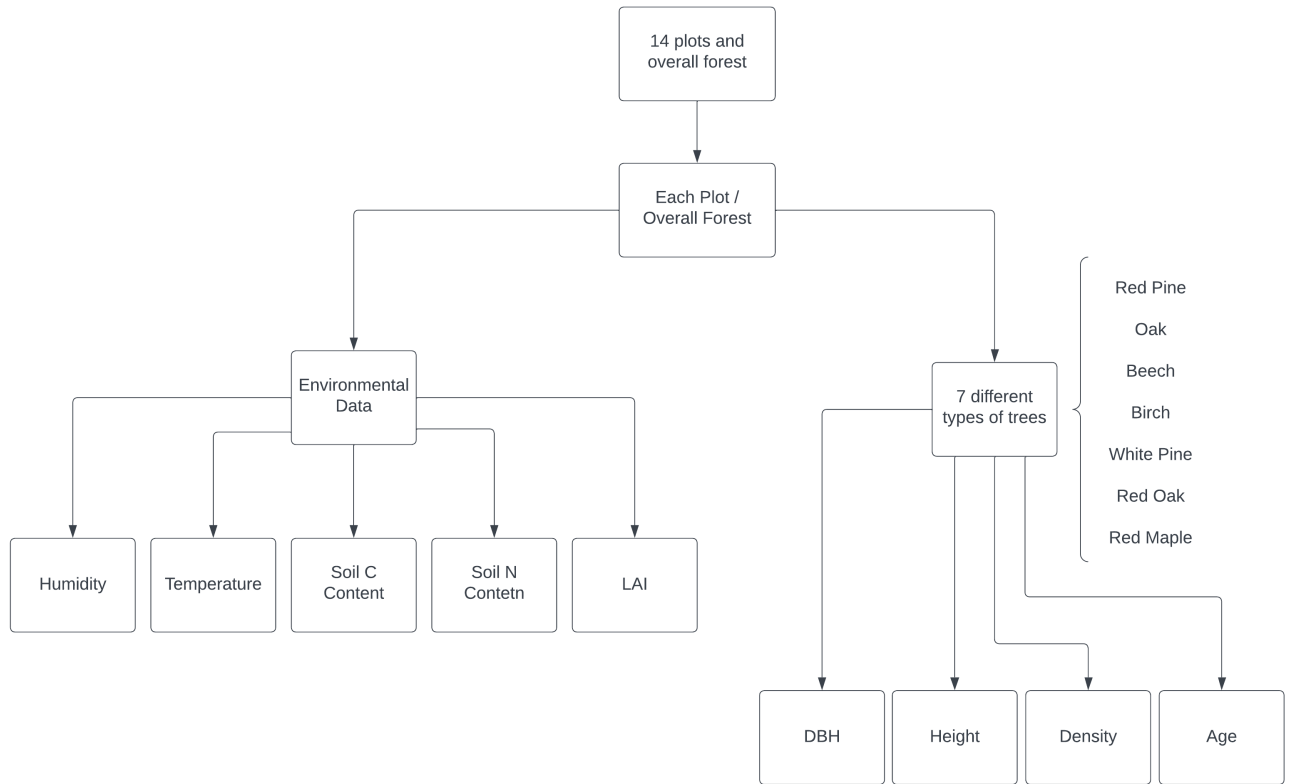


Figure 3: Data Hierarchy

### 3.3 The Scope of the Product

#### 3.3.1 Product Boundary

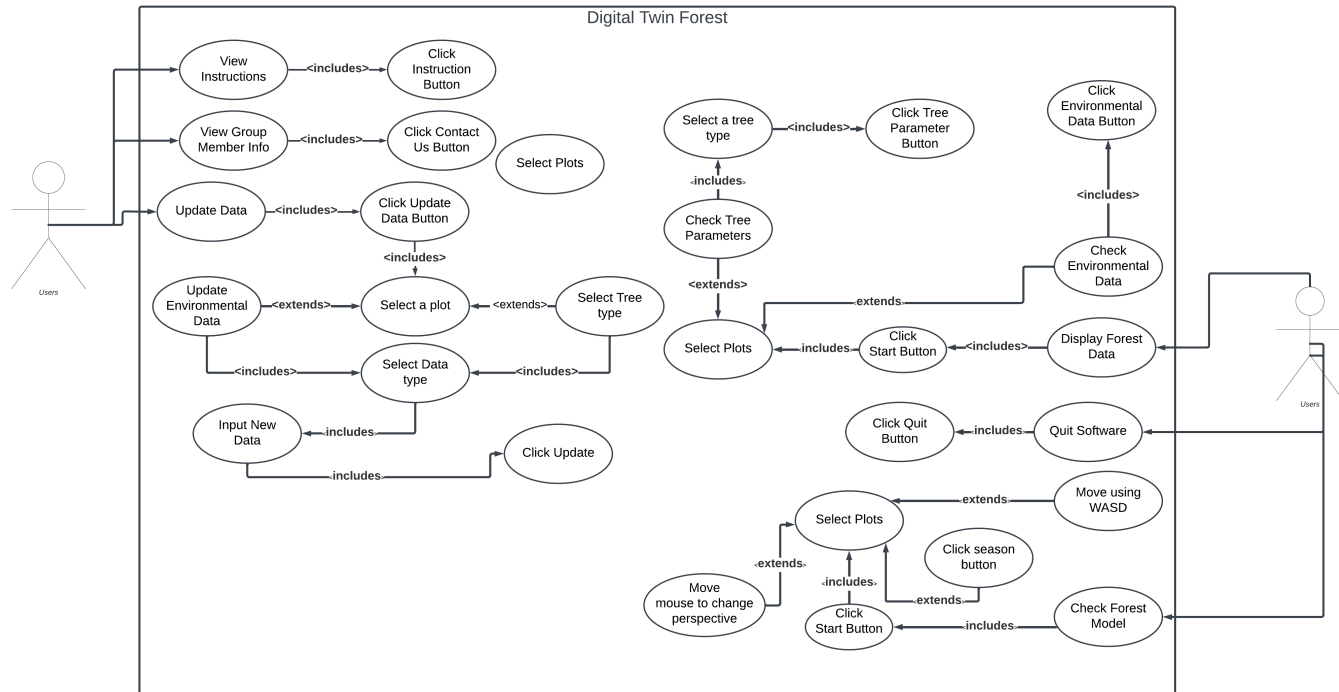


Figure 4: Use Case Diagram

#### 3.3.2 Product Use Case Table

1. PUC Name: View instructions  
Actor: Users  
Input: Users click **Instruction** button at the beginning of the application.  
Output: Application instructions should appear on the screen.
2. PUC Name: View group member information  
Actor: Users  
Input: Users click **Contact Us** at the beginning of the application.  
Output: Group members' contact information should appear on the screen.
3. PUC Name: Update Data  
Actor: Users  
Input: This use case involves many actions. Please check Figure 4.  
Output: Newly entered data should be stored in Data Store successfully.



4. PUC Name: View forest data  
Actor: Users  
Input: This use case involves many actions. Please check Figure 4.  
Output: Environmental data or tree parameters should appear on the screen.
5. PUC Name: View forest models  
Actor: Users  
Input: This use case involves many actions. Please check Figure 4.  
Output: Forest models appear on the screen.
6. PUC Name: Quit software  
Actor: Users  
Input: Click **Quit** button.  
Output: Software should quit successfully.

### 3.4 Functional Requirements

- FR1 The product shall provide an option to display basic instructions.  
**Rationale:** Users might not be familiar with our product, especially when they use it for the first time.  
**Fit Criterion:** A basic instruction option shows up on the user interface after the software is launched.  
**Supporting Materials:** All requirements using the instruction page.  
**Dependencies:** N/A
- FR2 The product shall provide an option to display the contact information of the development team.  
**Rationale:** Users may want to contact us when they have questions about the product.  
**Fit Criterion:** A contact-us page shows up on the user interface after the software is launched.  
**Supporting Materials:** All requirements using the contact us page.  
**Dependencies:** N/A
- FR3 The product shall load the forest model when users access the model page.  
**Rationale:** The clicking of 'start' indicates the user has been through the instruction and is ready to use our product. The product can now initialize the functions by loading the model.  
**Fit Criterion:** The software loads the data and models of the forest once the users select the 'start' button.  
**Supporting Materials:** All requirements using the forest model.  
**Dependencies:** N/A
- FR4 The product shall allow users to minimize the user interface. (User interface includes windows that present forest data).  
**Rationale:** The user might want to pay more attention to the model instead of information related to it. Minimizing the user interface allows the user to do so.  
**Fit Criterion:** The windows of the user interface will be minimized to the side of the screen

once the user clicks the *minimized* button on the user interface.

**Supporting Materials:** All requirements interacting with the panel of forest data.

**Dependencies:** N/A

FR5 The product shall be able to display environmental data when users select a specific forest plot.

**Rationale:** Environmental data are very essential for the real forest. Therefore, we need a way to visualize environmental data in our product.

**Fit Criterion:** Users can check environmental data successfully from the User Interface.

**Supporting Materials:** All requirements using the environmental data.

**Dependencies:** N/A

FR6 The product shall be able to display tree parameters when users select a specific tree type.

**Rationale:** Tree parameters are very essential for the real forest. Therefore, we need a way to visualize tree parameters in our product.

**Fit Criterion:** Users can check tree parameters successfully from the User Interface.

**Supporting Materials:** All requirements using the tree parameters.

**Dependencies:** N/A

FR7 The product shall display the forest data without interrupting users viewing forest models.

**Rationale:** The product should display the important data that our users might care about on the user interface while not letting the information interrupt the display of the model. Therefore, it should distribute the information on both sides.

**Fit Criterion:** When the full view of the forest is displayed, windows containing the overall information about the forest shall be shown on both sides of the screen.

**Supporting Materials:** All requirements asking for usability.

**Dependencies:** N/A

FR8 The product shall allow users to move around within the forest.

**Rationale:** The user might want to see different perspectives of the forest.

**Fit Criterion:** When users press W, S, A, and D keys on the keyboard, the camera moves forward, backward, leftward, and rightward accordingly.

**Supporting Materials:** All requirements related to performance.

**Dependencies:** N/A

FR9 The product shall allow users to change their perspectives.

**Rationale:** Users might need to observe the forest from different perspectives.

**Fit Criterion:** The user's perspective changes when he or she moves the mouse.

**Supporting Materials:** All requirements related to performance.

**Dependencies:** N/A

FR10 The product shall provide a way to present all the data when the amount of data does not fit on one page.

**Rationale:** The amount of information might not fit on a single page. Instead of compromising the user interface and minimizing the font size, displaying the information on several pages would be a better choice.

**Fit Criterion:** Several pages will show up when showing information with very long content.

**Supporting Materials:** All requirements interacting with panels of forest data.

**Dependencies:** N/A

FR11 The product shall provide ways to allow users to quit the product.

**Rationale:** The user would need a clear signal to exit.

**Fit Criterion:** Once the user clicks on the 'exit' button, the application shall quit, and the process of the software shall be terminated.

**Supporting Materials:** All requirements about security.

**Dependencies:** N/A

FR12 The product shall allow users to update forest data.

**Rationale:** In order to let the application show the latest information, it's necessary for users to update forest data by themselves.

**Fit Criterion:** After the user updates the latest information, the virtual forest shall display this latest information.

**Supporting Materials:** All requirements about updating and maintenance.

**Dependencies:** N/A

FR13 The tree models shall synchronize with the tree parameters.

**Rationale:** Tree parameters vary in different plots. The synchronization of tree models and tree parameters can show different models according to different plots, making each plot unique and realistic.

**Fit Criterion:** After users change a tree parameter, like tree height, the height of the tree model will change accordingly.

**Supporting Materials:** All requirements asking for accuracy.

**Dependencies:** N/A

FR14 The tree models shall allow the users to switch between summer view and winter view.

**Rationale:** A forest has two different appearances in summer and winter.

**Fit Criterion:** The model's appearance will change when users click the button to switch seasons.

**Supporting Materials:** All requirements asking for realism.

**Dependencies:** N/A

FR15 The product shall provide a pie chart to users to indicate the proportions of different types of trees in each plot.

**Rationale:** Users might want to proportions of different types of trees in a plot.

**Fit Criterion:** The information window shows a pie chart indicating the proportions of different types of trees..

**Supporting Materials:** All requirements about usability.

**Dependencies:** N/A

FR16 The distribution of trees in the model shall correspond to the target forest.

**Rationale:** The distributions of trees in different plots are not the same in the actual forest.

**Fit Criterion:** Once users enter different plots, they will see different distribution patterns of trees.

**Supporting Materials:** All requirements asking for accuracy.

**Dependencies:**N/A

### 3.5 Priority and Timeline

Priority and Timeline		
Functional Requirements	Priority	Timeline
FR1	low priority	October 20th, 2022 to October 27th, 2022
FR2	medium priority	October 27th, 2022 to November 3rd, 2022
FR3	high priority	January 26th, 2022 to February 9th, 2023
FR4	low priority	November 10th, 2022 to November 17th, 2022
FR5	medium priority	December 15th, 2022 to January 12th, 2023
FR6	low priority	November 24th, 2022 to December 1st, 2022
FR7	high priority	February 9th 2023 to February 16th, 2023
FR8	high priority	February 16th, 2023 to February 23rd, 2023
FR9	high priority	November 17th, 2022 November 24th, 2022
FR10	high priority	January 12th, 2023 to January 15th, 2023
FR11	high priority	January 15th, 2023 to January 19th, 2023
FR12	low priority	December 1st, 2022 to December 8th, 2022
FR13	low priority	December 8th, 2022 to December 15th, 2022
FR14	high priority	January 19th, 2023 to January 26th, 2022
FR15	medium priority	November 3rd, 2022 to November 10, 2022
FR16	high priority	February 23rd, 2023 to March 9th, 2023

Table 3: Functional Requirements Priority Table

## 4 Nonfunctional Requirements

### 4.1 Look and Feel Requirements

#### 4.1.1 Appearance Requirements

LF1.1 The model of the digital twin forest should look like the real forest.

**Rationale:** The graphic quality can optimize users' experiences.

**Fit criterion:** Among 10 samples of users who use it for the first time, 80% of them agree the forest model is close to the real forest.

**Supporting Materials:** N/A

**Dependencies:** All requirements related to seasonal change and model quality.

#### 4.1.2 Style Requirements

LF2.1 The product shall appear authoritative based on the real forest.

**Rationale:** The product is reliable only if it's based on real world data.

**Fit criterion:** The data in our project will not exceed a relative error of 5%.

**Supporting Materials:** N/A

**Dependencies:** All requirements about data accuracy.

LF2.2 The UI design of our product shall appear professional.

**Rationale:** The UI design should appear professional to fit the scenario where the product is used, as the product is mainly used in the scientific research.

**Fit criterion:** Among 10 samples of users all sample users during the testing phase, at least 80% of them think the system is professional.

**Supporting Materials:** N/A

**Dependencies:** All requirements about user interfaces.

### 4.2 Usability and Humanity Requirements

#### 4.2.1 Easy of Use Requirements

UH1.1 The product instructions shall be easy to understand.

**Rationale:** Clear instructions can take users less time to learn how to use the product.

**Fit criterion:** Among all the sample users during the testing phase, at least 80% of them can learn how to use the system within half an hour.

**Supporting Materials:** N/A

**Dependencies:** All requirements about instructions.

#### 4.2.2 Personalization and Internationalization Requirements

UH2.1 The product shall support English.

**Rationale:** English is a general language that most users will know.

**Fit criterion:** 100% of the text in the product is written in English.

**Supporting Materials:** N/A

**Dependencies:** All requirements using user interfaces.

#### 4.2.3 Learning Requirements

UH3.1 The instructions for the product shall be displayed on the instruction page.

**Rationale:** Users can easily access the instructions on the instruction page.

**Fit criterion:** Instructions option is displayed on the main page. After clicking it, users can see the instructions shown on the instruction page.

**Supporting Materials:** N/A

**Dependencies:** All requirements about the instruction page.

#### 4.2.4 Understandability and Politeness Requirements

UH4.1 The product shall use icons to highlight core functions.

**Rationale:** Icons can make it easier for users to understand the core function of the system.

**Fit criterion:** Among all the sample users during the testing phase, over 80% of them can notice and understand the icons.

**Supporting Materials:** N/A

**Dependencies:** All requirements about user interfaces.

UH4.2 The product shall use icons that are appealing to all adults.

**Rationale:** Icons shall be user friendly to users of all ages.

**Fit criterion:** Among all sample adult users during the testing phase, over 80% of them think the icons are appealing.

**Supporting Materials:** N/A

**Dependencies:** All requirements about user interfaces.

#### 4.2.5 Accessibility Requirements

UH5.1 The product shall be usable by people who are able to use a keyboard and a mouse.

**Rationale:** The interface of the system shall be consistent with other software (e.g use a mouse and keyboard).

**Fit criterion:** 100% of the actions can be done with a mouse and a keyboard.

**Supporting Materials:** N/A

**Dependencies:** All requirements about user movement.

UH5.2 The product's user interface should be easy to learn.

**Rationale:** Lower learning costs can help attract more customers.

**Fit criterion:** Among all the sample users during the testing phase, over 80% of them can learn how to use the system within half an hour.

**Supporting Materials:** N/A

**Dependencies:** All requirements about instruction.

### 4.3 Performance Requirements

#### 4.3.1 Speed and Latency Requirements

PR1.1 The product shall respond to user requests quickly.

**Rationale:** Quick response can improve users' experience.

**Fit Criterion:** The system can respond within 2 seconds for all the user requests.

**Supporting Materials:** N/A

**Dependencies:** FR3

PR1.2 The product shall run at high FPS.

**Rationale:** Smooth experience can improve users' experience.

**Fit Criterion:** The system should run at least 30 frames per second over 80% of the time.

**Supporting Materials:** N/A

**Dependencies:** FR8, FR9

PR1.3 The product shall be able to load the models quickly.

**Rationale:** Less waiting time for users can improve users' experience.

**Fit Criterion:** The waiting time will not exceed 10 seconds 100% of the time.

**Supporting Materials:** N/A

**Dependencies:** FR3

#### 4.3.2 Safety-Critical Requirements

N/A

#### 4.3.3 Precision or Accuracy Requirements

PR3.1 The forest data that displayed on the screen shall be accurate.

**Rationale:** The data is supposed to be reliable and match the real world data.

**Fit Criterion:** All the data in the system should have at most 5% relative errors compared with the target forest data.

**Supporting Materials:** N/A

**Dependencies:** FR12

#### 4.3.4 Reliability and Availability Requirements

PR4.1 The product shall be available whenever users access to it.

**Rationale:** The product is reliable if it's available most of the time.

**Fit criterion:** The system is required to work any time of a day.

**Supporting Materials:** N/A

**Dependencies:**FR3,FR8,FR9

PR4.2 The product shall avoid crashing while running.

**Rationale:** Stable running environment can make the product more reliable.

**Fit Criterion:** The system shall not crash over once out of 100 hours of running.

**Supporting Materials:** N/A

**Dependencies:** FR3,FR8,FR9,FR12

#### 4.3.5 Robustness or Fault-Tolerance Requirements

PR5.1 The product shall be able to run locally.

**Rationale:** The product can be more reliable if it runs locally.

**Fit Criterion:** The product can perform all the functions without internet connection.

**Supporting Materials:** N/A

**Dependencies:**FR3

#### 4.3.6 Capacity Requirements

PR6.1 The software size shall not be too large.

**Rationale:** Less size can make the software more portable.

**Fit Criterion:** The size of the software shall not exceed 10 GB.

**Supporting Materials:** N/A

**Dependencies:**FR3

#### 4.3.7 Scalability or Extensibility Requirements

N/A

#### 4.3.8 Longevity Requirements

N/A

### 4.4 Operational and Environmental Requirements

#### 4.4.1 Expected Physical Requirements

OE1.1 The product shall be used on personal computers and laptops.

**Rationale:** This product is designed for personal computer and laptop users, for these devices provide enough computing capabilities. The product should be able to run on target devices.

**Fit criterion:** The product shall be launched successfully and run without errors on both personal computers and laptops.

**Supporting Materials:** N/A

**Dependencies:**FR8, FR9



OE1.2 The product shall be used with a mouse and a keyboard.

**Rationale:** Some of the functions of this product, like switching plot, are designed to be used with a mouse. And the movement within the model are designed to be used with a keyboard. The product must be used with one and take the input from both mouse and keyboard.

**Fit criterion:** The devices with a mouse and a keyboard should be able to successfully realize the designed functions.

**Supporting Materials:** N/A

**Dependencies:**FR8,FR9,FR13

#### 4.4.2 Requirements for Interfacing with Adjacent Systems

OE2.1 The product shall be able to run on the devices with Windows 10, macOS 12 or any later version.

**Rationale:** Windows 10 and macOS 12 are mainstream operating systems which are running on most target devices. The developer team will ensure the product work on these operating system with the highest priority.

**Fit criterion:** The product shall be able to launch and complete all the functions as this document describes on the devices with Windows 10, macOS 12 or any later version.

**Supporting Materials:** N/A

**Dependencies:**FR8, FR9

#### 4.4.3 Productization Requirements

OE3.1 The product shall be distributed as an application to be installed on computers.

**Rationale:** The user might be a forest owner with no experience with coding. A reasonable method to distribute the product should be using an installer that can be downloaded.

**Fit criterion:** The product should be able to be downloaded and installed by users with clicks instead of installing using the terminal or commands.

**Supporting Materials:** N/A

**Dependencies:** FR8, FR9

#### 4.4.4 Release Requirements

OE4.1 The maintenance releases will be offered to end users weekly for at least one year after releasing.

**Rationale:** The effectiveness of this product significantly depends on the timely information related to the target forest. The latest information should be updated weekly for a higher quality of this product.

**Fit criterion:** The update should be released weekly.

**Supporting Materials:** N/A

**Dependencies:** FR13

OE4.2 Each release shall not cause existing functions or features to fail.

**Rationale:** The product shall be updated frequently, while each release should keep all the core functions and should not previous features to fail.

**Fit criterion:** Each release should pass every test case before releasing to the users.

**Supporting Materials:** N/A

**Dependencies:** FR13, FR14, FR16

## 4.5 Maintainability and Support Requirements

### 4.5.1 Maintenance Requirements

MS1.1 Documentation of this product shall be kept up to date.

**Rationale:** Documentation related to this product may change along with any changes in the product.

**Fit criterion:** Any modifications of the features or functions of the product should be documented before the change.

**Supporting Materials:** N/A

**Dependencies:**N/A

MS1.2 All functions shall be clearly documented.

**Rationale:** The development of this project should follow the documentation for a well-organized development process.

**Fit criterion:** The project should not be modified without documentation, which means it should realize all and only documented functions.

**Supporting Materials:**N/A

**Dependencies:**N/A

MS1.3 Any detected bug in the product shall be fixed within three days.

**Rationale:** Limited time for fixing bugs ensures the product to work normally.

**Fit criterion:** Any time when the developer team detects any bug, the bug should be fixed within three days.

**Supporting Materials:** N/A

**Dependencies:**FR1

### 4.5.2 Supportability Requirements

MS2.1 Users shall be able to provide feedback to developers.

**Rationale:** The users should have the method to send feedback to or contact the developer team if they want.

**Fit criterion:** Users should be able to find the contact method of the developer team in the product.

**Supporting Materials:** N/A

**Dependencies:**FR1

### 4.5.3 Adaptability Requirements

MS3.1 The product is expected to run on different operating systems.

**Rationale:** Users use different operating systems. The above non-functional requirement can allow more users to experience our product.

**Fit criterion:** The product shall be launched successfully and run without errors on at least two different operating systems.

**Supporting Materials:** N/A

**Dependencies:**FR8,FR9

MS3.2 The product is expected to be used indoors and outdoors.

**Rationale:** The product should be able to be used wherever as long as the user has a required device.

**Fit criterion:** The product should be able to launch and work as expected for the devices located either indoors or outdoors.

**Supporting Materials:** N/A

**Dependencies:**FR8,FR9

## 4.6 Security Requirements

### 4.6.1 Access Requirements

SR1.1 The product shall only be accessed by users who download the product from our GitHub website.

**Rationale:** The product is supposed to be used through a proper approach.

**Fit criterion:** Testers cannot download the product in any way other than the GitHub.

**Supporting Materials:** N/A

**Dependencies:**N/A

SR1.2 Forest data shall only be modified through the interface provided by developers. **Rationale:** Modifying the data through the interface provided by developers is the only safe method to update data. Some errors which would leads to crash like unintentionally deleting the stored data might occur if failing to do so.

**Fit criterion:** The users shall not be able to modify the data anywhere else except the update data interface provided by the developers.

**Supporting Materials:** N/A

**Dependencies:** FR12

### 4.6.2 Integrity Requirements

SR2.1 The system shall not propagate errors throughout the users' devices in case of failure.

**Rationale:** Propagating errors in users' machines will affect users when they use other applications.

**Fit criterion:** Injecting 100 errors on purpose, at most 2 errors will be propagated.

**Supporting Materials:** N/A  
**Dependencies:**FR3, FR11

SR2.2 The product shall avoid crashing when being used.

**Rationale:** The product should provide a continuous and comfortable user experience.

**Fit criterion:** The product shall not crash or unexpectedly quit while using.

**Supporting Materials:** N/A

**Dependencies:**FR3, FR8, FR9

SR2.3 The product shall check if user updates(user inputs) are legal before updating them to the system.

**Rationale:** The invalid input should be refused when updated or input by the users to keep consistency and normal performance of the product.

**Fit criterion:** The product should fail to update the data when the input is invalid.

**Supporting Materials:** N/A

**Dependencies:** FR12

SR2.4 Data displayed in the application shall be consistent with the data stored.

**Rationale:** The product should be consistent in the data storage and data display to avoid any potential confusion.

**Fit criterion:** The data should be consistent with the data just updated.

**Supporting Materials:** N/A

**Dependencies:** FR5, FR6

SR2.5 The system shall provide one-to-one mapping relationships between each data and GUI.

**Rationale:** The product should not provide redundant service.

**Fit criterion:** Any data should have a unique position to display in the GUI, and each position for data display should correspond to unique data.

**Supporting Materials:** N/A

**Dependencies:**FR5,FR6

#### 4.6.3 Privacy Requirements

SR3.1 The product shall not ask the users to provide personal information. (personal information refers to users' name, contact information and location, etc).

**Rationale:** Users will feel uncomfortable when they expose their information.

**Fit criterion:** 100% of the actions will not require personal information from users.

**Supporting Materials:** N/A

**Dependencies:**N/A

SR3.2 The product shall not send notifications to the users without permissions.

**Rationale:** Sending notifications without permissions will interrupt users' other activities.

**Fit criterion:** None user shall receive notification without permission.  
**Supporting Materials:** N/A  
**Dependencies:**N/A

#### 4.6.4 Audit Requirements

N/A

#### 4.6.5 Immunity Requirements

N/A

### 4.7 Cultural and Political Requirements

#### 4.7.1 Cultural Requirements

CP1.1 The product shall not have elements that offend the users of the environment in which the system is deployed.

**Rationale:** Offending users will decrease users' experience.

**Fit criterion:** None user should feel offended.

**Supporting Materials:** N/A

**Dependencies:**FR5, FR6

#### 4.7.2 Political Requirements

N/A

### 4.8 Legal Requirements

#### 4.8.1 Compliance Requirements

N/A

#### 4.8.2 Standards Requirements

LR2.1 The product shall abide by all Canadian laws and regulations.

**Rationale:** The product must obey all legal rules so that it could be published or used.

**Fit criterion:** None of the users shall notice any part lawfully unreasonable.

**Supporting Materials:** N/A

**Dependencies:**FR5, FR6

## 5 Project Issues

### 5.1 Open Issues

The process of collecting data, building models, and eventually generating a product takes a long time. During this process, real-world data may have changed due to environmental factors and human factors. Environmental factors such as thunderstorms, conflagrations, earthquakes, floods, and the natural growth of plants and human factors such as cutting and planting might change the data of trees and the structure and density of the forest. When we are modelling the actual forests, The data that we use is non-real-time, which will lead to the data shown in the virtual forest that we model being different from the actual data in the real world. This might cause inaccuracy in the actual use of the product.

### 5.2 Off-the-Shelf Solutions

Our product does not use any other softwares as off-the-shelf solutions.

### 5.3 New Problems

1. Our project may provide a new method for meteorologists to obtain information, which influences the traditional workflow.
2. The new management system will affect the work of forestry practitioners.

### 5.4 Tasks

#### 5.4.1 Project Planning

Please check our project schedule [here](#).

#### 5.4.2 Planning of the Development Phases

There are five development phases of this project:

- Design: We are going to determine the functional and non-functional requirements after specifying our stakeholders.
- Measure: We are going to scan the trees and measure the parameters of the trees.
- Implementation: Implement the project in unity. The first part will be modelling and post-processing. The second part will design the user interface and display the data.
- Test each module of the project in Visual Studio 2019 by unit testing and check the code coverage.
- Export the project and apply it in Dr.Gonsamo's lab.

### 5.5 Migration to the new Product

N/A

## **5.6 Risks**

1. Bad weather when field modelling may result in inadequate modelling.
2. Budget might not cover the cost.
3. Excessive schedule pressure.
4. The trees might be too high to scan all aspects of the trees or result in poor precision.
5. The project might occupy too much memory of the devices.
6. The natural forest continuously changes, which brings high maintenance costs to update related information or misleads the forest owner to make decisions.

## **5.7 Costs**

The cost of this project will not exceed 750 Canadian dollars.

## **5.8 User Documentation and Training**

User manuals with a few lines of instructions.

## **5.9 Waiting Room**

1. The product can connect with sensors installed in the virtual forest and then update data automatically.

## **5.10 Ideas for Solutions**

The representation of the overall data of the forest might be realized in a separate module. And the same for the recorded significant data. Our current project is designed to run locally on a certain device, while a possible solution could be, to design an online mode for the users to access to the latest information. The users might click a certain button to browse the overall data of the forest and its historical versions.

## 6 Traceability Matrix

### 6.1 Traceability between FRs and NFRs

The following are traceability matrices to show relationships between functional and non-functional requirements.

If there is a relationship between a functional requirement and a non-functional requirement, this means that the completion of this non-functional requirement will depend on the implementation of the functional requirement.

FR/NFR	LF 1.1	LF 2.1	LF 2.2
FR1			
FR2			
FR3			
FR4			✓
FR5			✓
FR6			✓
FR7			
FR8			
FR9			
FR10			
FR11			
FR12		✓	
FR13	✓	✓	
FR14	✓		
FR15			
FR16	✓	✓	

Table 4: Traceability Matrix 1



FR/NFR	UH 1.1	UH 2.1	UH 3.1	UH 4.1	UH 4.2	UH 5.1	UH 5.2
FR1	✓	✓	✓				✓
FR2	✓						
FR3							
FR4							
FR5		✓		✓	✓		
FR6		✓		✓	✓		
FR7							
FR8						✓	
FR9						✓	
FR10				✓			
FR11						✓	
FR12						✓	
FR13							
FR14				✓			
FR15							
FR16							

Table 5: Traceability Matrix 2

FR/NFR	PR 1.1	PR 1.2	PR 1.3	PR 3.1	PR 4.1	PR 4.2	PR 5.1	PR 6.1
FR1								
FR2								
FR3	✓		✓		✓	✓	✓	✓
FR4								
FR5								
FR6								
FR7								
FR8		✓			✓	✓		
FR9		✓			✓	✓		
FR10								
FR11								
FR12				✓		✓		
FR13								
FR14								
FR15								
FR16								

Table 6: Traceability Matrix 3

FR/NFR	OE 1.1	OE 1.2	OE 2.1	OE 3.1	OE 4.1	OE 4.2
FR1						
FR2						
FR3						
FR4						
FR5						
FR6						
FR7						
FR8	✓	✓	✓	✓		
FR9	✓	✓	✓	✓		
FR10						
FR11						
FR12						
FR13		✓			✓	✓
FR14						✓
FR15						
FR16						✓

Table 7: Traceability Matrix 4

FR/NFR	MS 1.1	MS 1.2	MS 1.3	MS 2.1	MS 3.1	MS 3.2
FR1			✓	✓		
FR2						
FR3						
FR4						
FR5						
FR6						
FR7						
FR8					✓	✓
FR9					✓	✓
FR10						
FR11						
FR12						
FR13						
FR14						
FR15						
FR16						

Table 8: Traceability Matrix 5

FR/NFR	SR 1.1	SR1.2	SR 2.1	SR 2.2	SR 2.3	SR 2.4	SR 2.5	SR 3.1	SR3.2	CP 1.1	LR 2.1
FR1											
FR2											
FR3			✓								
FR4											
FR5						✓	✓			✓	✓
FR6						✓	✓			✓	✓
FR7											
FR8				✓							
FR9				✓							
FR10											
FR11			✓								
FR12		✓		✓							
FR13											
FR14											
FR15											
FR16											

Table 9: Traceability Matrix 6

Some justifications: Some non-functional requirements are out of the scope of the software requirements(for example: hardware, environment, IO, etc). Therefore, they are not related to functional requirements.

## 6.2 Traceability Between FRs and Use Cases

FR/Use Case	PUC1	PUC2	PUC3	PUC4	PUC5	PUC6
FR1	✓					
FR2		✓				
FR3					✓	
FR4				✓	✓	
FR5				✓		
FR6				✓		
FR7				✓	✓	
FR8					✓	
FR9					✓	
FR10				✓		
FR11						✓
FR12			✓			
FR13					✓	
FR14					✓	
FR15				✓		
FR16					✓	

Table 10: Traceability Between Use Cases and Functional Requirements

### 6.3 Traceability Between NFRs and Use Cases

NFR/Use Case	PUC1	PUC2	PUC3	PUC4	PUC5	PUC6
LF1.1					✓	
LF2.1			✓		✓	
LF2.2				✓	✓	
UH1.1	✓	✓				
UH2.1	✓			✓		
UH3.1	✓					
UH4.1				✓	✓	
UH4.2				✓		
UH5.1			✓		✓	✓
UH5.2	✓					
PR1.1					✓	
PR1.2					✓	
PR1.3					✓	
PR3.1			✓			
PR4.1					✓	
PR4.2			✓		✓	
PR5.1					✓	
PR6.1					✓	
OE1.1					✓	
OE1.2					✓	
OE2.1					✓	
OE3.1					✓	
OE4.1					✓	
OE4.2					✓	

Table 11: Traceability Between Use Cases and Non-Functional Requirements(Table 1)

NFR/Use Case	PUC1	PUC2	PUC3	PUC4	PUC5	PUC6
MS1.1						
MS1.2						
MS1.3	✓					
MS2.1	✓					
MS3.1					✓	
MS3.2					✓	
SR1.1						
SR1.2			✓			
SR2.1					✓	✓
SR2.2			✓		✓	
SR2.3						
SR2.4				✓		
SR2.5				✓		
SR3.1						
SR3.2						
CP1.1				✓		
LR1.1				✓		

Table 12: Traceability Between Use Cases and Non-Functional Requirements(Table 2)

## 7 Likely Changes

1. The data structures used to store the overall forest data could be changed.
2. Data could be changed. First, data magnitudes may change since users and the developing team may update the data. Secondly, data types may change since the developing team may add or delete some data types.
3. The way to update data could be changed. Users can update the data in the product manually, but it is likely that the data can be changed automatically through sensors in the future.
4. User interface may change since the development team may add, adjust or delete some elements on the user interface.
5. Other contents apart from trees in the virtual representation may change. For instance, the development team may introduce new models such as grass, bushes, and so on to the virtual representation.

## 8 Unlikely Changes

1. The product must display basic instructions when users access the product.  
Justification: According to our user characterises, most of our target users may not have experience with digital twin technologies. Therefore, instructions are essential.
2. Including tree models as a part of the virtual forest is unlikely to change.  
Justification: Trees are the most important part of a real forest.

## A Reflection Appendix

The preparation of our project includes collecting data, scanning on field, and recording and organizing the data for later use. Before we determined our project, our group reached Dr.Gonsamo for possible collaboration. With help of members in his lab, we gained basic experience of scanning models on field and determined a target forest. In the data-collecting phase, we would need to keep fluent communication with Dr.Gonsamo and the members in his lab. We also need to handle the huge amount of measurements to get all the data we need later and try to manage the data in a reasonable and convenient way, which should be easy to both manage and use, to make the next phase easier. The measurement work might take days and could be heavy labour, which means we need to do a detailed plan before we go to the target forest, maintain a concise team, and make a reasonable division of labour during it. After the measurements, we need to process and reorganize the data for modelling. We will need data-processing skills in this phase. Besides, as the project relates to the domain of the ecosystem, related knowledge is also required.

In the modeling part, the work will be finished in Unity. We will need to get familiar with Unity to construct 3D models and develop a user interface. As we will construct the model based on the data we collect in the former phase, we will use C# to import the data. We will achieve this purpose with a relatively small amount of codes, while the expertise C# knowledge is still required. To share the model and get everyone synchronized, we will use Git in our project.

We are going to keep working on the capstone for over six months, and this is the first time that we manage a relatively large project involving five individuals. Team management skills are essential for us. During the whole process, we will continuously work on documentations, and will present our project three times (proof of concept; revision 0; and EXPO). We will treat our project as a serious business case, and try to make it both formal and attractive. This purpose requires outstanding skills in writing and delivering presentations.

In conclusion, we will need to acquire:

1. the skills of communication and group management,
2. the knowledge related to ecosystem,
3. the skills of scanning 3D models and collecting data,
4. the skills of managing and processing data,
5. the skills of C#,
6. the skills of working with Unity,
7. the skills of working with git,
8. the skills of writing and delivering presentations.

The division of labour is listed below:

- Bowen Zhang: 1, 3, 6, 8
- Yichen Jiang: 1, 2, 5, 8
- Jiacheng Wu: 1, 3, 4, 8



- Tingyu Shi: 1, 6, 7, 8
- Junhong Chen: 1, 4, 5, 8

As shown above, the skills of communication, group management, writing, and delivering presentations are mandatory for every member of our group. These skills are essential not only for this project, but also in the future when we handle even larger projects. The approaches for these skills include continuous exercising, gaining information online, and communicating among our team members. We are willing to improve ourselves towards qualified engineers during the capstone, so the related training is significant for every one of us.

The knowledge related to ecosystems can be obtained by reading related papers or books, and/or communicating with Dr.Gonsamo and the members in his lab. Yichen will be responsible for this task, for a better ability of reading professional papers and a more flexible schedule. The skills of scanning 3D models and collecting data will be pursued by Bowen and Jiacheng. These skills can be obtained by watching tutorials online, and frequent practice. Bowen will be the lead developer on modelling part. He will manage the data on the former phase to control the quality of models. Jiacheng expertises on collecting data and has transportation, which provides a convenient situation for working on field.

The skills of managing and processing data will be pursued by Junhong and Jiacheng. The content has been covered by an introduction course related to data management, and extra materials can be obtained on some public online courses. Junhong and Jiacheng have related backgrounds, finished database courses, and have strong interests in strong-logic-related work.

The skills of C# will be pursued by Junhong and Yichen. These skills can be obtained by tutorial videos and by communicating to the experts in C#. Junhong has related working experience and Yichen is working as a teaching assistant in a coding course.

The skills of Unity will be pursued by Bowen and Tingyu. Bowen has one-year experience working with Unity in a full-time CO-OP, thus he will be the lead developer related to Unity. And as a complement of working environment, Tingyu will test the model on macOS supporting Bowen's work. These skills will mainly be obtained from Bowen's working experience, and some supporting content will be obtained by searching online. The skills of working with git will be pursued by Tingyu. These skills can be obtained on the tutorial of 4G06 and the shared material by TA. As we have been familiar with git, Tingyu will provide support only when necessary.

Though each skill has been assigned to one or two specific members, the skills will be mastered by every member at the end of the project. The member who is responsible for a certain skill should study the related materials and help every other member to catch up. Our final purpose is to let every member acquires every skill needed in our project.