# Software Requirements Specification

**for**

# W A T E R R A

**Version 2.0 approved**

**Prepared by**

**Group 15**

**Cynthia(Yingxue) Liu, Jennie(Jian) Li, Zoe(Huaijin) Ning, Dr. Emil Sekerinski**

**McMaster University**

**Oct 31, 2022**

**Table of Contents**



# Revision History

| Name | Date | Reason For Changes | Version |
|---|---|---|---|
| Holly Koponen | Oct 30 | Adjust document conventions to make the SRS easier to read, edit our constraints to be based on the context, create a better name for the project. | 1.0 |

# 1. Introduction

## 1.1 Purpose

This project *Waterra*[1] contains two parts related to the water quality data. *Waterra* combines the morphemes of both the words ‘water’ and ‘terra’, which can be interpreted as the water of the earth.

The first part is a website called *Water Sensor*, later abbreviated as *the WS*[2]. The website includes a functional map where users can select and view data from any available water quality sensors. Air temperature, conductivity, dissolved oxygen, humidity, pH, turbidity, and water temperature are all displayed in real-time. To provide an intuitive and understandable view for all users, selected data will be automatically transformed into a line graph. This section is a part of the *Ohneganos*[3] program.

The second part involves incorporating the same data into a customized digital map powered by the *Terrastories*[4] project, along with place names for land areas and water features and stories about local culture. Later on, this component is referred to as *the TI*[5].

This document describes *the WS*'s and *the TI*'s functionality. This is version 2.0 of the document.

## 1.2 Document Conventions

Subheadings and highlights with the N/A label are in **bold**.
All titles and headings are in *Time New Roman* font.
The body text is all in 11-point *Arial* font.
Proper nouns, such as project and technology names, are in *italics*.
Requirements will be prioritized to indicate the order in which they will be implemented.

## 1.3 Intended Audience and Reading Suggestions

The software requirements specification include a list of stakeholders and their characteristics, detailed requirements, project-specific descriptions, and appendices.
This document is intended for *the WS* **website developers** and **project supervisors** of the *Ohneganos* program, as well as **high school teachers** who want to use *the WS* as teaching materials.
It is recommended that developers and project managers read this software requirements specification from beginning to end to gain a thorough understanding of both parts of the project.
To make it easier for high school teachers to begin using *the TI*, concentrate on sections 2, 3, and 4.

## 1.4 Product Scope

Under the *Ohneganos* program, *the WS* is a sub-project about water quality sensors. Unlike other existing products, it focuses on indigenous communities' lands and water. The data from the water quality sensor could be used for professional purposes such as data visualization and analysis through our project. Furthermore, with our website, more people who do not have a professional background can participate in documenting important cultural areas for conservation and development prevention. It also serves an educational function by educating future generations about their culture, history, and environment.

The *TI* is also a sub-project of *Terrastories*, a geo-storytelling application built to enable indigenous and other local communities to locate and map their oral storytelling traditions about places of significant meaning or value to them. By incorporating water quality data into *Terrastories*, not only aboriginal culture scholars but also indigenous peoples can gain a better understanding of indigenous lands and territories.

## 1.5 References


- George, Reece & Nesbitt, Keith & Gillard, Patricia & Donovan, Michael. (2010). Identifying Cultural Design Requirements for an Australian Indigenous Website. Conferences in Research and Practice in Information Technology Series. 106.
- H. Wright, *Incredible Indigenous Web Design From Around The World*, 09-Aug-2022.


# 2. Overall Description

## 2.1 Product Perspective

*The WS* is a follow-on development of Tianyu Zhou(graduated) and Dr. Emil Sekerinski's water sensor project. *The WS* should use the data that Tianyu achieves from the time-series database and improve the *nodes*[6] selecting a function and current data visualization. Currently, users can select the *nodes* and then the *probe*[7], then the initial data visualization graph would be generated below the map.

*The TI* is a follow-on development of the existing indigenous community. Currently, the website displays video/audio and literal stories for each marker on the map, our project aims to add the water quality data to the markers as well. That is, the water quality data would be integrated from the time-series database provided by the *Ohneganos*, and the *Terrastories* team would do the implementation to display it together with other contents.
As both parts of the project are follow-on development of an existing project, our options are limited by the technologies and tools used by the existing parts of *the WS* and *the TI*. We

also need to consider the reliability and stability of our work because what we finish might be included in other people's work in the future.

*The WS*:

- Currently, the developer implementing *the WS* uses *Go* for the backend and *React* for the front end.
- A time-series database, *influxdb*, is used for retrieving data.

*The TI*:

- The *Terrastories* team is using *Mapbox* for generating the map.
- The *Terrastories* team is using *Heroku* to store data.
- The *Terrastories* team is using an API, *GeoJSON* to transmit data.
- After integrating the water quality data into the *Mapbox*, the *Terrastories* team will conduct further development and analysis with the data we provide.

## 2.2 Product Functions

***The Water Sensor***

- This project should allow users to select data by choosing *nodes* and *probes*, as well as choosing the time range.
- This project should display a well-organized, neat graph for selected data. If multiple *nodes* are selected, the data should be displayed as lines in the same graph with labels of their names.
- Users should be able to select the specified interval to zoom in/out of the graph.
- Users should be able to see the precise data on the graph when putting the mouse on it.

***The Terrastories Integration***

- The water quality data should be integrated into *Terrastories.*
- The data should be displayed on the map after selecting specific markers, along with the audio/video stories of that location.

## 2.3 User Classes and Characteristics

The user classes anticipated with both parts of our project are similar, as they both aim at providing water quality data as easy access to the public so that people could start to be aware of the importance of water.
Below users are listed from most frequently used to least frequently used.

- **Indigenous people**
    - People with indigenous backgrounds want to check out the water quality of their area. They might interact with this project by looking at the color code (i.e. Use color to distinguish the level of data) to learn the level of data
    - Limited educational background (partially)
    - Limited literacy skills in English
- **Professional users**
    - People with a profession in the field of water

    - Perform water quality data analysis
    - Care about water
    - Require the data to be accurate
- **High school teachers**
    - Want to demonstrate water quality to student
    - Rich geographic knowledge
    - Comprehensive understanding of the website
    - Data analysis and representation required
- **High school students**
    - For educational purposes, i.e. learning more about water quality and its importance of it
    - Basic understanding ability
    - Limited hardware (computer) resources and experience with hardware
    - Could be instructed by their teachers to interact with the project
- **Community (casual)**
    - People from all age groups, such as the elderly with limited computer experiences
    - Related to daily life, casual use, and curious about the water quality in neighborhood
    - Less professional but concerned about water
- **Other developers**
    - Might use the website and the water quality data for other purposes, for example, to perform deeper analysis
    - Might inherit, maintain, and further developer the website after
    - Rich software knowledge

## 2.4 Operating Environment

- Both parts of the project require a browser and an operating system.
- The browser should be one of the several recent versions of *Google Chrome*, *Firefox*, *MS Edge*, and *Safari*.
- Operating system: *Windows 7*+, *Mac OS X 10.7*+, or *Ubuntu*[8] *10*+.
- *The WS* should work well on both phone and desktop versions of the above browsers and must have an internet connection. The ideal internet connection is around 0.3+ Mbps.
- *The TI* should be able to work online and offline, so an internet connection is unnecessary.
- *The TI* must have *Docker* installed. For *Windows* systems, *WSL*[9] 2.0 or virtualization supported by *Hyper-V*[10] is required in order to work.

## 2.5 Design and Implementation Constraints

*The TI:*

- Because we are only integrating the data into the tool used by the *Terrastories* team, we cannot make and see exact changes on the map.
- This website is aimed at Indigenous and high school students, but it is also universal and educational. As a result, data presentation should not be overly specialized and difficult to understand.

*The WS:*

- We can not use other technical tools, i.e. new programming languages or frameworks, as we need to follow the existing format.

### 2.6 User Documentation

There would not be other user documentation needed along with the deliverable of this requirement.

### 2.7 Assumptions and Dependencies

It is assumed that our users know English and have common sense. There are no dependencies needed. Both of our projects should work in most situations, as we adapt them to multiple browsers and mobile devices.

## 3. External Interface Requirements

### 3.1 User Interfaces

The brief description of UI and low-fidelity mockup prototypes will be demonstrated below.

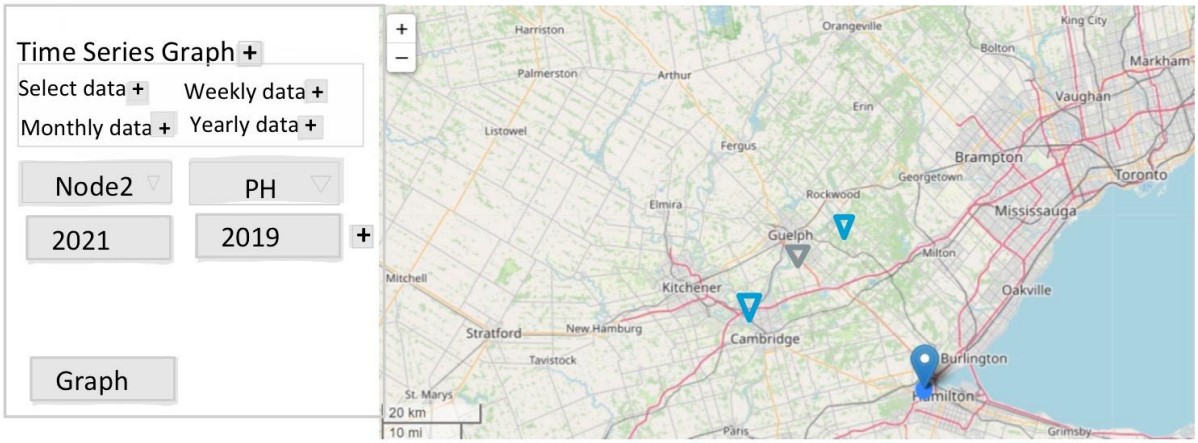
Generally, the main page of *the WS* consists of three parts: a map, a sidebar, and a data analysis area.

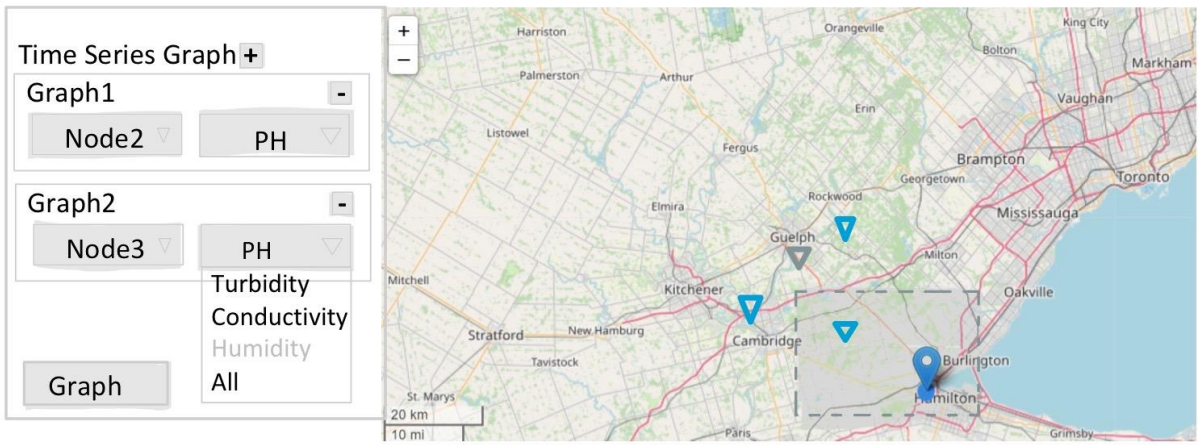
The map contains markers at all sensor locations, with the color of the markers varying depending on the most recent data update time. Recent ones are highlighted in blue, while others are highlighted in grey. Users can zoom in/out and pan the map by clicking on the buttons in the map's corner.

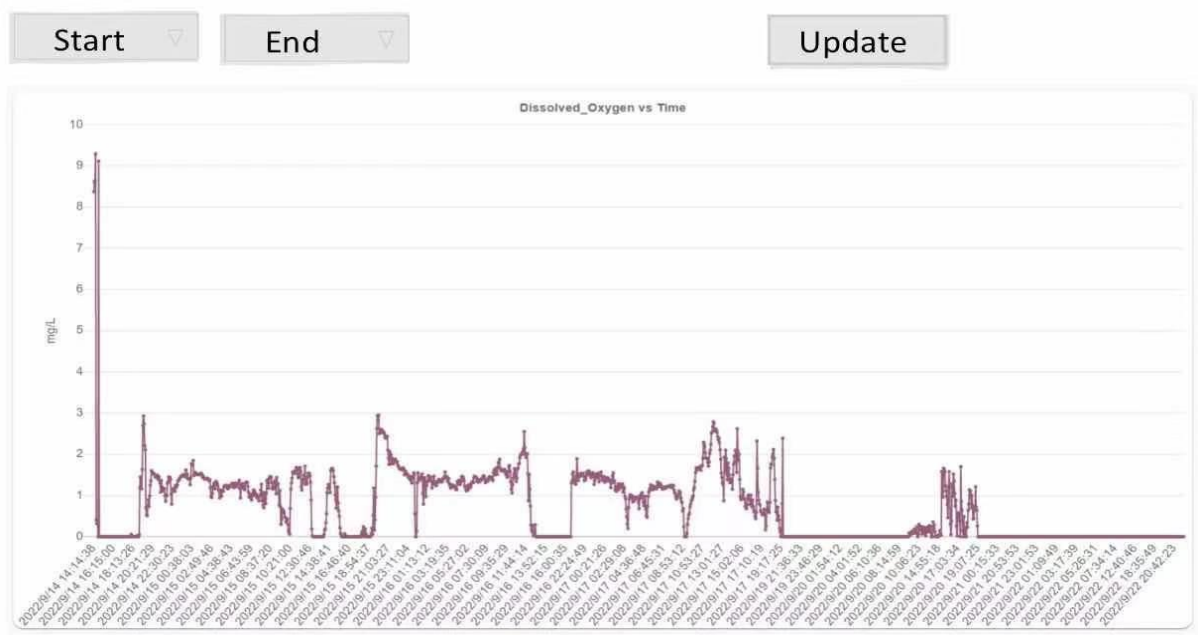
The sidebar contains a number of drop-down lists and buttons that allow users to manually select *nodes* and *probes*, as well as *probe* categories and data time periods. After selecting an option, the map will display the corresponding changes, such as highlighting the selected *nodes*.

The data analysis area displays all data visualizations, primarily in table and graph formats. The table displays specific numbers from real-time data. To clearly show the condition, the graph is horizontally divided into three areas and represented by different colors to indicate the data status (i.e. Low, Average, High, Emergency, etc.). The data graph also has zoom-in/out buttons. When the cursor is over a point on the line graph, the exact data appears.

The theme color of the website is McMaster maroon. This style is used on all pages.






For *the TI*, the interface should be very similar to the original interface of *Terrastories*.

## 3.2 Hardware Interfaces

**N/A**

*The WS* requires a large system consisting of water sensors and gateways to gather data. *The TI* demands an installation and setting up of a procedure to start using. If it is used offline, extra devices might be required.
Since our team is **not responsible** for the hardware part, N/A is labeled here.

## 3.3 Software Interfaces

Following are the software components used for *the WS*.

| Name | Description |
|---|---|
| *MacWater API* | Present and explain all the data-related requests between *the WS* website and the database. https://documenter.getpostman.com/view/5847961/2s83tDpshb |
| *Chart.js* | A flexible JavaScript charting API used to generate data line graphs. https://www.chartjs.org |
| *OpenStreetMap* | An open data map API for the functional map in *the WS*. https://www.openstreetmap.org/ |

| | |
|---|---|
| *React.js* | A free and open-source JavaScript library for front-end.<br>https://reactjs.org/ |
| *Go* | A statically-typed programming language used for the backend.<br>https://go.dev/ |
| *InfluxDB* | A time-series database for retrieving data.<br>https://www.influxdata.com/ |

Following are the software components used for *the TI*.

| Name | Description |
|---|---|
| *Terrastories* | Github repo of the *Terrastories* team.<br>https://github.com/Terrastories/terrastories |
| *Docker* | A set of platform-as-a-service products that use OS-level virtualization to deliver software in packages called containers.<br>https://www.docker.com/ |
| *Mapbox* | A provider of custom online maps for the website.<br>https://www.mapbox.com/ |
| *Heroku* | A cloud platform to store data.<br>https://www.heroku.com/ |
| *GeoJSON* | A format for encoding a variety of geographic data structures.<br>https://native-land.ca/resources/api-docs/#Maps |

### 3.4 Communications Interfaces

- *The WS* is secured by an *SSL certificate*[11] and needs to install new plugins, update already installed ones, and update some of its components (i.e. from macWater APIs, modules, etc.).
- *The WS* provides links to relative communities, such as *SixNations*, *Ohneganos*, as well as social media to contact.
- *The TI* requires an internet connection to update *Docker* and *Terrastories* regularly.

## 4. System Features

### 4.1 Data Visualization - Initial Map Browsing

### 4.1.1 Description and Priority

Users should be able to browse water quality situations in different geographic areas by marking symbols on an initial map. This is of high priority.

### 4.1.2 Stimulus/Response Sequences

Users access the website with an initial map. The real-time data shall pop up on top of the map when a mark representing a single *node*(location) is aimed.

### 4.1.3 Functional Requirements

1. The website should include an initial map with marks on all sensor locations(*nodes*). Sensors whose data have been updated within a certain period of time (i.e. 3 days) should be distinguished by a different color of marks. For example, updated sensors are marked in blue, others in grey.
2. This initial map should be able to be zoomed in and panned.
3. There should be a pop-up on top of the map when one of the location marks is selected, which should list the real-time data of each *probe*. Different colors should be used to distinguish the status of each data(i.e. Low, Average, High, Emergency, etc.).
4. The additional notes of colored data and the expected range of data should be shown along with the pop-up.

## 4.2 Data Visualization - Overlapping Graph Generalization

### 4.2.1 Description and Priority

Users should be able to select specific data and compare different sets of data by an overlapping line graph. This is of high priority.

### 4.2.2 Stimulus/Response Sequences

Users can start by selecting the data type or squaring an area on the initial map. The front end will return the layout of detailed options and any constraint options. When users confirm selected data, the front end will send a requested task to the backend in order to get the data. The final graph will be updated by the front end every time a new time period is selected.

### 4.2.3 Functional Requirements

1. The website should have a side block with selection bars for selecting data by manually selecting the *node*, the *probe*, weekly data, monthly data, and yearly data.
2. The website should allow adding up to 5 sets of data to generate the final overlapping data graph.
3. Multiple *nodes* can be selected by squaring an area from the map. When more than 5 *nodes* are selected, randomly 5 of them will be added to the side block for generating the final graph. The user needs to manually change undesired *nodes*.
4. When selecting a *probe* category(i.e. PH), only *probes* with data can be selected, otherwise, will be greyed out. The same goes for selecting a *node*.

5. When selecting yearly data with a certain *node* and *probe*, up to 5 years can be added to generate the final overlapping graph. The same goes for selecting weekly or monthly data.
6. The overlapping graph should update every time after selecting a start time and an end time.
7. The overlapping graph should be able to zoom in and zoom out by a dragging operation.
8. The overlapping graph should be able to show the numeric data by putting a mouse on a certain point.

# 5. Other Non-functional Requirements

## 5.1 Performance Requirements

When *the WS* website is working with an internet connection, it should take less than five seconds to fully load. If one or more *nodes* are selected, information related to all selected *nodes* (for example, real-time data table, overlapped graph) should be displayed completely below the map within one second. Actions like selecting a new *node*, deselecting a *node*, zooming the graph in/out, or displaying the graph only in a specific period of time should cause the website to regenerate the graph and update all corresponding information immediately, then the changes show up after several hundred milliseconds. If the real-time data is updated before action, the latest data should also be displayed on the new graph to demonstrate the timeliness. Exporting and downloading data in a specific format can take a little more time to achieve. The time required varies slightly depending on the data volume, and should not exceed one minute at most. All performance should be able to work as mentioned above 80% of the time, for some extreme circumstances, the time taken is still to be determined, but all of them would be finished.

Loading time for *the TI* should be no longer than eight seconds due to its map complexity. Basic operations such as adding a marker and editing story content are supposed to be finished within half a second. With an internet connection, the efficiency of uploading video/audio materials highly relies on internet speed. In the case of offline, the performance of *the TI* mainly depends on the capabilities of the devices used. Generally, local servers should be very efficient.

## 5.2 Safety Requirements

To ensure that the data is accurate and no data is lost when users are using this project (due to a bug or a crash), unauthorized users only have permission to view the data. Although users can visualize the data in different forms, the raw data the project generates from the time-series database is highly protected.

## 5.3 Security Requirements

- Users' personal information is not stored, and the website would not track users' activities.
- Users' private information is not needed anywhere for using the project.

## 5.4 Software Quality Attributes

- Adaptability: the project could adapt to multiple browsers and mobile devices.
- Effectiveness: the project should do what it is intended to do, that is, display accurate water quality data.
- Learnability: the project should be easy to use for all kinds of users, including but not limited to indigenous, high school students, as well as casual users from the local community.
- Memorability: when users come across the project for the second time, they should remember how to use it. That is, the project is easy to remember.

## 5.5 Business Rules

For *the WS*, as mentioned in 5.2, only administrators and authorized users are allowed to edit water data in the database. Normal users can only use the existing data of the website. In order to protect the rights and interests of water area owners, the specific data of some regions will not be disclosed to all users, but the changing trend of water quality will still be displayed normally through data analysis.

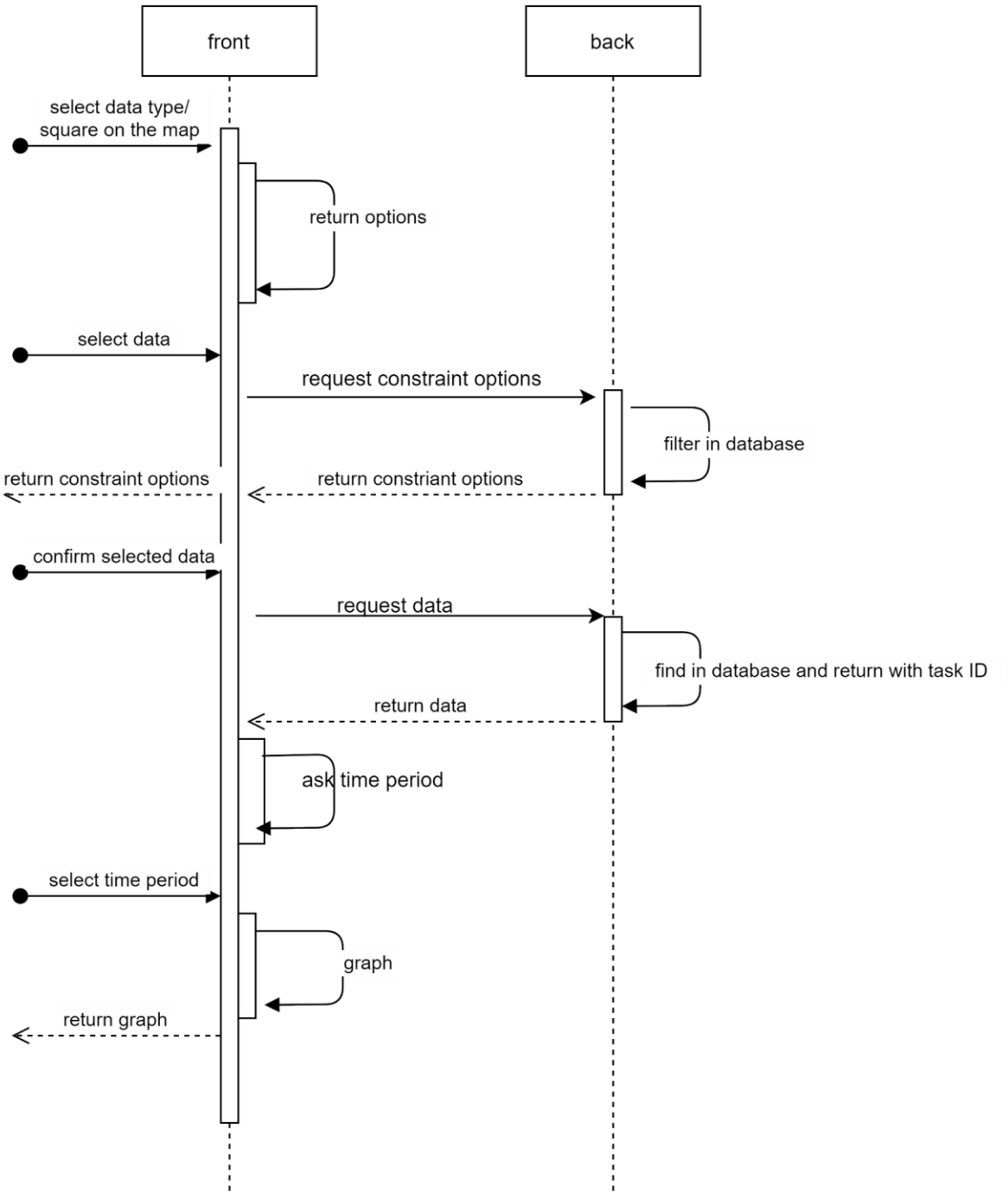
For *the TI*, teacher accounts and student accounts must be assigned different authorities. Teacher accounts should have control of the entire teaching community they create, while the authority of the student accounts is restricted and gradually relaxed by the teacher account as the teaching progresses.

# Appendix A: Glossary

1. *Waterra:* the project name, explained as the water of the earth.
2. *The WS*: refers to the water sensor data visualization part of our project.
3. *Ohneganos*: an Indigenous water research program led by McMaster University Professor, Dr. Dawn Martin Hill. The word "*Ohneganos*" means water in Iroquoian Language.
4. *Terrastories*: an open-source application built to enable local communities to locate and map their own oral storytelling traditions about places of significant meaning or value to them.
5. *The TI*: refers to the *Terrastories* integration part of our project.
6. *Node*: locations that have installed the water sensor gateway.
7. *Probe*: the water sensor of each *node* includes multiple *probes* which may include an air temperature *probe*, conductivity *probe*, dissolved oxygen *probe*, humidity *probe*, PH *probe*, turbidity *probe*, water temperature *probe*, etc.
8. *Ubuntu*: a Linux distribution based on Debian and composed mostly of free and open-source software.
9. *WSL 2.0*: a new version of the Windows Subsystem for Linux architecture that powers the Windows Subsystem for Linux to run ELF64 Linux binaries on Windows.
10. *Hyper-V*: a Microsoft hardware virtualization product that lets you create and run a software version of a computer, called a virtual machine.
11. *SSL certificate*: a bit of code on your web server that provides security for online communications.
12. Some other technical software components are mentioned and explained in section 3.3

# Appendix B: Analysis Models

Data Visualization Flow diagram for water sensor website:



# Appendix C: Related Websites

## C1: Website Related to Real-Time Water Data


“Data - AQUARIUS WebPortal,” *Disclaimer - aquarius webportal*. [Online]. Available: https://aqrt.nrs.gov.bc.ca/Data/Map/Parameter/TW/Location/Type/Interval/Latest.

“Great Lake Data Stream,” *Datastream*. [Online]. Available: https://greatlakesdatastream.ca/explore/.

N. L. Department of Environment and Climate Change, “Water Resources Management,” *Water Resources - Dept. of Environment and Climate Change - ADRSv6.0 - List of Real Time Stations*. [Online]. Available: https://www.mae.gov.nl.ca/wrmd/adrs/v6/graphs_list.asp.

“Real Time Water Quality Monitoring System - 202.74.40.143,” *實時水質監測系統*. [Online]. Available: http://202.74.40.143/~afcdxyle/main_en.php

“Real-time hydrometric data,” *Api.weather.gc.ca*. [Online]. Available: https://api.weather.gc.ca/collections/hydrometric-realtime/items.

U. S. G. S.- U. S. G. Survey, “National real-time water quality,” *US Geological Survey Real-Time Water Quality Data For the Nation*. [Online]. Available: https://nrtwq.usgs.gov/.

“Water Quality Data Portal by government of Alberta,” *WQP*. [Online]. Available: https://environment.extranet.gov.ab.ca/apps/WaterQuality/dataportal/.


## C2: Website Related to Indigenous Water Situation


C. X. Luo, “The water crisis in Canada's First Nations communities,” *ArcGIS StoryMaps*, 14-Nov-2021. [Online]. Available: https://storymaps.arcgis.com/stories/52a5610cca604175b8fb35bccf165f96.

Government of Canada; Indigenous Services Canada, “Map of long-term drinking water advisories on public systems on reserves,” *Government of Canada; Indigenous Services Canada*, 12-Oct-2022. [Online]. Available: https://www.sac-isc.gc.ca/eng/1620925418298/1620925434679#

“Water First,” *Water First Education & Training Inc.*, 06-Oct-2022. [Online]. Available: https://waterfirst.ngo/


## C3: Other websites


“Vocabulary words in the Iroquoian language family,” *Iroquoian Words*. [Online]. Available: http://www.native-languages.org/famiro_words.htm