Samuel Nguyen

CPTR 496/497 Winter 2024

OAP Tank Monitor

*Project Proposal*

I propose the development of a web-based software application for the Open Acidification Project within the Biology Department at Walla Walla University. This initiative aims to create a comprehensive platform that efficiently connects and displays present and logged information from the tank controller devices utilized in the acidification experiments. The web product will provide monitoring, data visualization, and analysis tools. By leveraging web technologies, this site aims to enhance accessibility and collaboration among all project members, through research communication and data sharing. The proposed software will contribute to the overall success of the Open Acidification Project by providing a user-friendly interface for researchers to interact with and interpret their data effectively. This computer science capstone project aligns with the CS department's dedication to addressing a specific need, in this instance, within the biology department, and contributes to the advancement of scientific research within the academic community.

*Background*

The customer of this project is Dr. Kirt Onthank, an Associate Professor of Biology at Walla Walla University. This collaborative project aims to provide tools to conduct ocean acidification research and is an open-source initiative involving the Computer Science and Biology Departments. This collaboration has been going on for several years and I will be working with Professor James Foster in the Computer Science Department.

This project aims to enhance the monitoring of tank controllers through a functional and eye appealing web application. Lab studies and ocean acidification research in the Biology Department will be met with a computer science tool for enhanced studies. The proposed features will empower users with real-time/last logged insights and historical data analysis capabilities, contributing to more informed decision-making processes.

My role for this project is to create a user-friendly web application that provides valuable information on one or more connected tank controllers for the public and open source project. Currently existing is a web client is a manager/controller to look and interact with a tank controller. This present application has some limits with accessibility and function. These limitations include access to one tank, presenting only current data. As well as being only able to use on the same network with a ‘not secure’ website. The new contributions will be used alongside the tank controller with monitoring enhancements at large with all connected tanks.

An OAP tank monitor web application will be reading connected tanks with a display of the historical logs. Focusing on improvement features, this new client aims to be visible from anywhere with a secure public site and connection to the server. It is crucial as this enables remote monitoring on tank controllers, eliminating the need for physical checks and providing access from any research location via the web. The Biology Department currently operates between half a dozen to two dozen tank controllers. The monitoring system is set to serve the neighboring department and deploy this current school year. My project fills a needed and practical tool and provides a user experience solution, while aligning with my capstone experience.

*Requirement Goals*

List of project tasks related to the development of the web app:

* Data Transmission
  + Ensure that the connected devices are consistently sending data to the server.
* Server Data Storage
  + Verify that the server can accept incoming data from the connected devices.
  + Confirm a data storage system to save and organize logged information.
* File Retrieval
  + Enable the server to provide access to files containing the uploaded data.
  + Establish a secure sockets method to retrieve relevant files.

List of project key features for web app:

* Thumbnail Display
  + Develop a feature that presents tank controller logs in a clear and comprehensible format.
  + Arrange a grid or list of thumbnails, each representing a tank controller, accompanied by concise information.
  + Include key metrics or indicators within each thumbnail to offer a snapshot of the device's status.
  + Implement interactivity by incorporating a click-open function, allowing users to access detailed information about a specific tank controller.
  + Ensure the display is responsive and adept at different screen sizes and devices, providing a seamless user experience.
* ‘Real-time’ Data Display:
  + Prioritize the latest logged information for immediate visibility, emphasizing certain data and visualization of most important info for the Biology department.
  + Highlight changes and trends using color-coded indicators and icons.
  + Employ charts, graphs, and other visualization tools to improve data comprehension.
* Historical Data Retrieval
  + Create visualization to present all historical data stored in the root@oap:/var/opt/oap/logs directory.
  + Enable users to explore historical data by browsing different timeframes, for comprehensive understanding of historical changes.

*Design Documents*

User Interface Design: Tank Controller Dashboard:

* Clean and branded header displaying the application name and unit information.
* Dropdown or grid field allowing users to choose a specific tank controller.
* Prominent display of current data/status thumbnail per unit.
* Full display of all data with cards/tiles/graphs/charts wherever they best suit the design for optimal clarity and user experience.

Potential Research User Story:

“The application eliminates the need for physical checks and provides web access from any research location significantly enhances the efficiency and convenience of my tasks.”

*Review of Tools*

* Flutter

-​​ package:http/http.dart

-fl\_chart / syncfusion\_flutter\_charts

* Postman
* Github

*Software Development Schedule*

Week 5

* Proposal submission and approval
* Setup skeleton with “hello world”, folders, and build
* Initial test(s)

Week 6

* oap POST log files / oap GET log files
* HTTP requests from Flutter
* Present display grid/list
* Build selected tank log file view

Week 7

* oap GET flutter app
* Pipeline to deploy on oap server
* Show list linked to Open Acidification Project (deploy)
* Allow selection of tank

Week 8 - 10

* Save selected tanks in browser local storage
* Continue development on thumbnails and selected tank

CPTR 498

* Weeks 11-13: Intuitive display for important data with graphing
* Weeks 14-15: Analysis with indicators for important data
* Weeks 16-17: Alert development for data outside of range
* Weeks 18-20: Poster, Paper, Presentation

Stretch

* Additional display features

*Challenges*

Challenges expected include continuity and connection between tanks, server, and web application. Tanks should log every minute and upload successfully. Server should be able to always provide logs to web applications. In the event of a loss of connection, the application should be able to display a disconnection and duration of status between the tanks. Also, the application can show the last log retrievable and display.

Additional ongoing challenges involve familiarizing myself with the current Ocean Acidification Project. My contributions should function cohesively with the pre-existing methods. Alongside the tank controller webpage, my tasks need to be accessible and displayed in a similar way. These additions will need new libraries, which need research and review, that can accomplish my web application.

Last of the challenges will be integrating my project addition and deployment with all members' satisfaction. I look forward to committing code progress in a proper pipeline development method.