492 Project Report

Salish Sea Water Weather Station

Jacob DeBoer, Karina Sandlin

4/17/2020

Table of Contents

Project Impetus	**3**
Deliverables	**3**
Use Cases	**4**
Success Criteria	**6**
Risks and Mitigation(s)	**8**
Preliminary Design	**10**
Validation Criteria	**11**
Sprint 1 Task Assignment	**13**
Sprint 1 Task Completion	**14**
Sprint 2 Task Assignment	**15**
Sprint 2 Task Completion	**16**
Sprint 3 Task Assignment	**17**
Sprint 3 Task Completion	**18**
Final Design	**19**
Testing Plan	**19**
Conclusion	**21**

Project Impetus

Each year, WWU and the SEA Discovery Center host a cyber security summer camp for about 60 middle school students. The Salish Sea Water Weather Station project was created to serve as an introduction to cyber security concepts for these students. The project consists of a waterproof buoy with a variety of sensors, a mobile app to collect data from the buoys and a website to view the collected data.

We will provide the camp with the hardware and easy to follow documentation so that the buoys can be constructed by the students with the supervision of the counselors. We will also develop an android app to collect data from the buoys. The app will be secure, engaging and illustrate basic cyber security concepts to the students. Once data has been collected it will then be stored in a database and can be viewed from a website. Data will also be available to the public so that it can be used for research purposes.

Deliverables

Deliverable ID and/or name	Description
Buoy Kits	Kits containing all the necessary hardware and documentation for constructing the buoys, deploying them and collecting the data.
Android App	Android app that will be used by the students to collect data from the buoys. The app will be engaging and help illustrate basic cyber security concepts. Once data has been collected
Website	Website that will be used to visualize the data collected from the buoys. Data can also be downloaded for use by the general public or researchers.

Use Cases

Use Case Name	Goal	Precondition	Post-Condition	Actor(s)	Description	Alternative(s)
Buoy Construction	Buoy is built correctly.	Students have received the buoy kits.	Students have constructed an operational buoy.	Students, Counselors	1. Students build circuit. 2. Students store circuit in nalgene bottle. 3. Students test buoy to ensure data collection is working and buoy is water-tight.	1a. Students encounter problems while building the circuit. 1a1. Student and counselors refer to documentation to ensure the circuit is built correctly. 1a2. Resume at Step 2.
Buoy Deployment	Buoy is deployed and begins collecting data.	Buoy has been constructed.	Buoy is deployed and collects data accurately.	Students, Counselors	1. Students and counselors go to buoy deployment destinations. 2. Students have deployed their buoys in the ocean. 3. Students ensure buoy collects data.	3a. Buoy is not collecting data. 3a1. Counselor and Student retrieve buoy and ensure the device is receiving power. 3a2. Resume at Step 3.
Data Collection	Data is collected from the buoy.	Buoy has been deployed and collected data, Student has an android device.	Data is stored on android device to be uploaded to database.	Students, Counselors	1. Students and counselors return to buoy location. 2. Students connect android device to buoy with bluetooth. 3. Data is uploaded from the buoy to the android device. 4. Data is encrypted and visualized so students can learn.	2a. Student is unable to pair android device with the buoy. 2a1. Buoy is retrieved and taken back to camp.
Data Upload	Data is uploaded from android device to database.	Data has been collected from the buoy.	Data from buoy is successfully stored in the web-app database.	Students	1. Students connect to the internet. 2. Students upload the buoy data using the android app.	2. Data is not successfully uploaded to the database. 2a1. App indicates error in upload. 2a2. Student retries upload.

View Data	Data is accessible.	Data exists within the database.	Data has been accessed.	General public, Researcher s,	1. Viewer connects to the web-app. 2. Viewer navigates the app to view desired data. 3. Viewer downloads the data in a usable format.	3a. No data exists for the desired timeframe. 3a1. Webapp indicates that no data exists for the given timeframe. 3a2. Viewer is redirected back to the data selection screen.

Buoy Construction:

After a student completes the buoy construction use-case, they should have a fully water-proof and functional buoy. The buoy will record and store time, relative salinity, turbidity, light, temperature at various depths. It will be battery powered and able to store a few months of data locally.

Buoy Deployment:

After a student deploys their buoy, it should be in a static location in the Salish Sea where it will consistently be collecting data. The student will be able to connect to their buoy using bluetooth and download the data to their phone using the android app.

Data Collection:

Students will be able to collect data from their buoy using bluetooth. Using the android app, the student can connect over bluetooth to their buoy and download the data that the buoy has stored.

Data Upload:

After collecting data from their buoys, students will be able to upload the data from the android app to the web database.

View Data:

As students upload data to the web database, a web-app will interpret the data and make it accessible to the public through data-visualization and downloadable data which can easily be incorporated into research projects.

Success Criteria

Success Criteria Name	Description	Time Constraint	Technical Constraint	Goal (for stakeholders, and development team)
Buoy Construction	There will be a completed design and parts-list for buoy system which isn't expensive and has a well-documented and accessible build process. The codebase for the microcontroller is fully-functional.	30 hours	-Budget per buoy: $50 -Complexity of set-up: less than 30 minutes focused build time -Battery and storage to last one month	The majority of students will be able to easily set up their buoys using the guide. Counselors will be able to fix set-up errors without investing more than 5 minutes in general.
Database Functionality	There will be a functional database which can be accessed using the android app and can store buoy data intuitively. The web-app will also have access and make the data accessible to the public.	10 hours	-Must have both the storage space and organized design to accommodate several classes of students with continual data collection per student	The database should be well-designed so that as data piles up from students over the years the database can easily accommodate all the data without being readjusted.
Data Collection	The android app will be able to connect to a buoy using bluetooth and download the data onto the phone.	30 hours	-Uses bluetooth to connect to specific buoys	The goal is for students to easily be able to access their buoy's data. They will be able to download an app which can collect data from their buoy using bluetooth.
Data Upload	The android app will be able to connect to the database and upload the data collected from the buoy to it.	30 hours	-Must connect and upload from the app to the database using wifi	The goal is for students to be able to easily upload the data downloaded to their phone onto the database. The android app should have an intuitive interface to make this process easy.
Data Access / Visualization	The web-app will use the buoy-data database to create visualizations of the data and also will provide an interface for downloading the data in a usable format.	50 hours	-Website must be continuously hosted so that data can be viewed and downloaded.	The web-app should provide a clean interface which allows parents, researchers, and the general public to view the data collected by the students. It should also have a feature which allows researchers to download the data easily and in a format which can be incorporated into their research.

Buoy Construction:

This step will require us to design a cost-effective ocean-water testing device. We'll design it to be less than $50 in cost, and to take measurements of water salinity, turbidity, light, and temperature at three depths. It will need to be easily and safely constructed by children with no electrical engineering background. We'll write documentation on its set-up and how it works so that the students will have an enjoyable experience building their buoys. It will also have to be designed in a way so that it is easily water-proofed and durable enough to last a while bobbing about in the sea, with enough battery life and storage to last. Lastly, it'll have to be connected to via bluetooth to collect the data.

Database Functionality:

We'll need to design a database which can store the buoy data of several camps filled with students, and will need a clear and intuitive design which will make data upload and access easy and streamlined.

Data Collection:

The data will need to be collected off the buoys easily by the students. The buoys will need bluetooth functionality which will allow an app on the students phones to connect to the buoys. The students will then be able to collect the data off the buoys using the app, which will be downloaded onto their phone. To be successful, the students must be able to connect to the buoy over bluetooth without too much difficulty and to pull the data on their phone.

Data Upload:

Once the data is collected onto the students' phones, there should be an easy system to upload the buoy data to the database. This will be successful if the students can complete this step without hassle, and if the data is incorporated into the database.

Data Access / Visualization:

We'll design a web-app which uses the database to visualize the data for the general public, and also has an option to download the data in a format accessible for researchers. To be successful, it must have expressive and aesthetically pleasing data visualization, a clean interface with good UX design, continuous pulling from the database, and an easy way for researchers to incorporate the data into their research.

Risks and Mitigation(s)

Risk ID and/or name	Description	Analysis	Mitigation(s) / Resolution(s)	Priority
Buoy flooded	Buoy's seal is broken and becomes flooded.	Buoy is flooded likely destroying most components contained inside. Would require a replacement kit which is ~ $50.	Seal is tested in a controlled environment without components before deployment. Whenever data is collected buoy is inspected for any small leaks. Marine-grade epoxy and resin is applied.	Medium - High
Bad actor submits fake data.	Fake data is submitted to the database.	Fake data invalidates the database.	User authentication Buoy authentication	Low-Medium
Buoy is lost	Buoy is lost, stolen or mistakenly discarded.	Buoy is lost requiring a replacement kit which is ~$50.	Printed label on inside of the nalgene bottle explaining what the buoy is. "GPS tracking" label on buoy to discourage anybody from stealing it.	Medium - High
Buoy runs out of storage space	Buoy's data is not collected in time and the buoy runs out of space to store new data.	New data cannot be stored or older data is overwritten. Data is lost.	Data is retrieved monthly to ensure no data is lost. Data is deleted from the buoy once uploaded to an android device.	Low-Medium
Buoy runs out of battery	The buoy's batteries die and no longer can collect data.	Buoy loses ability to collect data and data is lost.	Ensure buoy goes into low-power mode correctly when not being accessed via bluetooth. Battery pack is large enough to last several months. Students are encouraged to replace batteries every visit.	Low-Medium

Buoy flooded:

The buoy could become flooded with sea water which would likely destroy all components contained within the buoy. The risk here is relatively high since the buoys will be placed in salt water for months at a time. In the event that this happens the buoy would need to be replaced entirely, costing the camp an additional $50 of their already limited budget. This risk could be mitigated by testing the buoy's seal in a controlled environment without any components contained inside to ensure there are no leaks before deployment. A marine-grade epoxy and resin will also be applied to the nalgene bottles to ensure the risks of a leak is minimal.

Buoys could also quickly be inspected each time data is retrieved to ensure there is no moisture inside the nalgene bottle.

Bad actor submits fake data:

The database could accept fake data from a bad actor which would invalidate the entire database or all measurements from a compromised buoy. The risk here is relatively low since it’s unlikely anybody will attempt to submit fake data and the data itself is not the main purpose of the buoys. This risk can be mitigated by requiring user authentication to connect to the buoys, upload data and have that data verified as coming from a registered buoy. We expect that this will be unlikely and that buoy and user authentication will be sufficient.

Buoy lost or stolen:

It’s entirely possible that the nalgene bottle could be mistaken for trash in the ocean and discarded. It could also be observed that it contains a lot of electronics and is stolen. The risk for this situation is relatively high since losing a buoy would cost the camp an additional $50 and the budget is very limited. We can mitigate these risks by printing a label to the inside of the nalgene bottles explaining that they’re water weather stations belonging to the SEA Discovery Center along with a URL for the website to view the data. We could also print some fake “GPS tracking” warning labels to discourage anyone from attempting to steal the buoys. While true GPS tracking would be ideal, it would consume too much power.

Buoy runs out of storage space:

The sparkfun board contains 4MB of flash memory which should be enough memory to store at least a months worth of data. The risk of this happening is relatively low as we expect the buoys will be checked on frequently and this would only result in a gap between data measurements. We can mitigate the risk of data being lost by having the data collected from the buoys at least once a month. Once data is uploaded to an android device it can be cleared from the sparkfun board to ensure there is enough memory for a month's worth of data after each upload.

Buoy runs out of battery:

To collect data, the sparkfun board requires a small amount of power. The buoy comes supplied with a battery pack to power the board. If the battery dies, the board will no longer be able to collect data, which will result in some data loss. The battery needs to have enough power to last the board at least a month to prevent this from happening. In addition, the students will have to replace the batteries in the battery pack frequently to ensure the board does not lose power. The board consumes a lot more power when active and being connected to via bluetooth, but the vast majority of the time it should go into a sleep mode in which it keeps passively collecting data and stays available for bluetooth connection but does little else, and in this mode the board will be a lot more energy-efficient.

Preliminary Design

There are three major components to the Weather Station system. These are the hardware buoy, the android app, and the website. At the core of the system is the buoy. This consists of a simple hardware set-up and data-collection software. The buoy is made of a SparkFun esp32 board with several sensors connected to it. On the board there is flashed a small program in C which collects the data from the sensors, converts it into a usable format, stores it, and transmits stored data securely over bluetooth to an authorized android app when it connects. The android app will download the data over bluetooth on to the phone. It will then send this data to a database common for all android apps/buoys. Lastly, there will be a web-app which will make the data accessible to the public.

Deliverable	Task Owner	Task List	Status
Buoy Kits	Karina Sandlin	Convert data to standard units	Everything still raw data, salinity in progress
		Ensure data successfully transmits to android app	Unknown (potentially complete)
		Dive into existing code and determine all missing components / required non-implemented features	In Progress
		Add missing features	Not Complete
		Document building the prototype	Unknown, Most likely not completed
	Sara Morimoto		
Android App	Jacob DeBoer	Help get Elana up and running w/ Android Studio environment and the prototype running on the Android Emulator.	In Progress
		Examine existing code to determine missing functionality and what needs to be completed.	In Progress
		Figure out how to register a weather station to a given user.	Not Completed
		Get data encryption working	Not Completed
		Improve look and feel of UI	Not Completed
		Get GPS location when deploying a new buoy.	Not Completed
	Elana Cueto	Get started w/ Android Studio	In Progress
		Get familiarized w/ the existing prototype and code.	Not Completed
Website / Web App	Grant Barton	Dockerize existing webapp database	In Progress
		Begin developing the website where data will be hosted and visualized.	Not Completed
		Integrate website with the existing backend prototype.	Not Completed
	Morgan Stimpson	Get familiarized w/ the existing backend prototype.	In Progress

Validation Criteria

Validation ID and/or name	Description / Testing and Validation Methodology	Deliverable(s)	Comments
Convert raw data, standardize data measurements	Collect raw data, convert to standard units, compare to real measurements to ensure accuracy	Buoy	Data needs to be in standard units and usable across the project, rather than arbitrary voltage readings
Data transmission	Collect data on buoy, connect using registered android app, collect data	Buoy	Data collected on buoy successfully is transmitted to a connected android device
Missing features	Determine missing features, add them, test them	Buoy	Will result in a completed buoy code-base
Document Prototype	Document in detail the process to build a buoy prototype	Buoy	Needs to be a user-friendly guide to building a buoy, accessible to students and counselors
Weather station registration	Weather station is successfully registered to the correct user within the database.	Android App	Weather stations are registered to a specific user whenever it is first deployed so that only that user (Or admin) can access the data collected.
Data encryption	Android app successfully encrypts the data before being sent to the database.	Android App	Data is encrypted before being sent to the database. Encryption should also be illustrated through the app to teach students about cybersecurity.
Weather station GPS	Android app collects the current gps location whenever paired to a weather station. Location is then updated within the database.	Android App	GPS location is always updated within the database upon pairing.
Dockerized Web App	Web app is contained in a docker container	Web App	The dockerization will make a deployment pipeline simpler to implement.
Web App frontend	Web app has a simple, user-friendly front end	Web App	This will be used by the general public, researchers, and students so should be designed with these use-cases in mind
Web App backend	Web app connects to the database containing the buoy data and displays it on the front end	Web App	After front-end is complete, it will need to communicate with the database through the backend so data is accessible through the web-app

Sprint 1 Task Assignment

Task Owner	Task ID	Task Description	Comments
Karina	"Convert raw data" *Buoy* task 1	Convert raw data into a usable format	Salinity may be quite time-consuming, other sensors should be somewhat simple
Karina	"Data transmission" *Buoy* task 2	Ensure data is being successfully transmitted to connected android app	Data transmission may already be mostly functional, but registration process unclear
Karina	"Feature round-up" *Buoy* task 3	Examine existing code-base, evaluate what work needs to be completed	The board code seems relatively functional, but with a small task-list of uncompleted features
Jacob	"Data Encryption" *Android App* task 1	Examine existing encryption functionality and fix errors.	Seems to be largely completed but still has a couple of errors preventing it from working 100%.
Jacob	"Weather Station GPS" *Android App* task 2	Make app record location whenever paired to a weather station. Location is then updated when data is uploaded to the database.	Weather station latitude and longitude is currently hard-coded.
Jacob	"Elana environment setup" *Android App* task 3	Help Elana get her android studio environment up and running with the project.	Elana is our newest team member and will be working on the Android App w/ me.

Sprint 1 Task Completion

Task Owner	**Task ID**	**Result of Sprint** Explain to what extent to goals of the sprint have been achieved.	**Status**
Jacob	"Data Encryption" *Android App task 1*	Familiarized myself with the existing encryption method. Work still needs to be done to implement the intended encryption method.	10 %
Jacob	"Weather Station GPS" *Android App task 2*	App successfully requests location permissions and gets the correct latitude and longitude if permission is given. Otherwise the app uses the latitude and longitude of WWU. This only happens during buoy registration as intended.	90 %
Jacob	"Elana environment setup" *Android App task 3*	Elana has been a bit difficult to get in touch with but, i've finally been able to get in touch with her and am in the process of setting up a meeting so we can get her environment up and running. This should be fairly quick.	50 %
Karina	"Convert raw data" *Buoy* task 1	Progress on salinity (waiting on external input) temperature sensor complete, turbidity sensor has roadblock (i don't have correct sensor), light sensor still needed	40%
Karina	"Data transmission" *Buoy* task 2	No progress, still cannot connect to app (steps required for registration unclear and potentially requires changes to android app)	0%
Karina	"Feature round-up" *Buoy* task 3	Found documentation on what is still needed, mostly minor features / cleanups, besides that it is likely not much else is needed as the program is relatively simple	90%

Sprint 2 Task Assignment

Task Owner	Task ID	Task Description	Comments
Karina	"Convert raw data" *Buoy* task 1	Convert raw data into a usable format	Salinity may be quite time-consuming, other sensors should be somewhat simple
Karina	"Data transmission" *Buoy* task 2	Ensure data is being successfully transmitted to connected android app	Data transmission may already be mostly functional, but registration process unclear
Jacob	"Data Encryption" *Android App Task 1*	Examine existing encryption functionality and fix errors.	Seems to be largely completed but still has a couple of errors preventing it from working 100%.
Jacob	"Board Setup" *Android App Task 4*	Our advisor has had a SparkFun board shipped to me so I can test the android app properly. Board needs to be flashed and have a sensor hooked up for testing data.	Haven't yet received the board but it should be arriving very shortly.
Jacob	"Registration Issues" *Android App Task 5*	The app is currently still using our old back end solution. Find out how to get the board registered properly with the old system or circumvent that part so boards can be connected to through the app.	The app is still using the old back end solution which uses Django. Search old documentation on how to unregister a device so Karina can connect to her board through the app or circumvent that security measure since it will eventually be replaced by the new solution anyways.

Sprint 2 Task Completion

Task Owner	**Task ID**	**Result of Sprint** Explain to what extent to goals of the sprint have been achieved.	**Status**
Karina	“Convert raw data” *Buoy* task 1	I tried to contact Karina but didn’t get a response. - Jacob	??
Karina	“Data transmission” *Buoy* task 2	I tried to contact Karina but didn’t get a response. - Jacob	??
Jacob	“Data Encryption” *Android App Task 1*	Familiarized myself with the existing encryption method. Work still needs to be done to implement the intended encryption method.	10%
Jacob	“Board Setup” *Android App Task 4*	Sparkfun board is flashed with the program and can be connected to. Ordered GPIO pins so that I can solder them to the board and hookup sensors to test data collection and encryption at a later time.	80%
Jacob	“Registration Issues” *Android App Task 5*	Board can now be successfully registered and connected to through the app. Implemented a system to work around connection security while we wait for the team to finish the new back-end solution. Minor testing is still needed.	90%

Sprint 3 Task Assignment

Task Owner	Task ID	Task Description	Comments
Jacob	"Data Encryption" *Android App Task 1*	Examine existing encryption functionality and fix errors. Change from RSA to something more secure.	Seems to be largely completed but still has a couple of errors preventing it from working 100%.
Jacob	"Board Setup" *Android App Task 4*	Board is mostly set up. Get GPIO pins soldered to the board and follow board documentation for setting up a temperature sensor.	Board is flashed. Once GPIO pins are soldered to the board a temperature sensor can be hooked up so that dummy data can be collected to test encryption.
Jacob	"Friendly UI" *Android App Task 6*	Modify the look and feel of the UI to make it look nicer. Potentially add animations.	Current UI is functional but is really bland.
Karina	"Convert raw data" *Buoy* task 1	Must connect with Sara to get this finalized - meeting scheduled	70% completed
Karina	"Data transmission" *Buoy* task 2	Will be trying this out with the new apk file	0% completed

Sprint 3 Task Completion

Task Owner	Task ID	Result of Sprint Explain to what extent to goals of the sprint have been achieved.	Status
Jacob	"Data Encryption" *Android App Task 1*	Further research on encryption methods. Goal is to use AES but I've been hitting a few roadblocks.	30% Completed
Jacob	"Board Setup" *Android App Task 4*	GPIO Pins are soldered to the board but i've discovered that the temperature sensor I have wont work.	90% Completed
Jacob	"Friendly UI" *Android App Task 6*	Added more color to multiple screens. Made text more visible. Added labels to some components for extra clarity. Began work on screen transition animations.	50% Completed
Karina	"Convert raw data" *Buoy* task 1	Determined salinity sensor not precise enough to output useable data, other data conversion mostly done	90% Completed
Karina	"Data transmission" *Buoy* task 2	Can now connect to buoy with phone, haven't tested data retrieval yet	50% Completed

Final Design

There have been no changes to the preliminary design.

Testing Plan

- We will be doing mostly functional and integration testing.
- Integration testing is a logical choice for us because we have such a modular project style, consisting of several almost completely isolated parts. We'll use integration testing to combine the subprojects into a whole functioning data flow, testing in particular the connections between:
    - Buoy and Android App
    - Android App and Database
    - Database and Web App
- Functional testing will be used within each subproject to ensure that each component functions well. The following subprojects will be employing functional testing for various components, in particular focusing on the external facing elements to make integration testing go more smoothly. The functional tests will fit within the following categories:
    - Buoy
    - Android App
    - Web App
- Functional and integration testing fit together in our project nicely because functional testing will ensure each subproject will work as expected when being interacted with externally, while integration testing will piece everything together.

Test ID	Validation ID (refer to previous table)	Type of Testing That will be performed	Description of test case(s).	Success/Failure Criteria
10	Data transmission, Weather station registration	Integration (Buoy and Android App)	Connect several phones using the android app to several different buoys	Connection and data retrieval between phone and buoy is reliable
11	Data transmission	Integration (Android App and DB)	Upload buoy data to database	Android app seamlessly connects to the database and uploads the fresh data
12	Web App backend	Integration (Web App and DB)	Web app retrieves new data from database as it surfaces	Web app stays updated with most recent data
13	Convert raw data, standardize data measurements	Functional (Buoy)	Collect data at various times and intervals	Timestamping and data is correct and day/night cycle is working
			Connect/reconnect with bluetooth	Bluetooth connection/reconnection doesn't cause glitches with data collection
			Data storage / clearing testing	Storage clears successfully when full or data is retrieved
14	Data encryption	Functional (Android App)	Buoy registration	Various buoys register successfully
	Weather station GPS		Data upload	App connects to database and uploads data to it
15	Dockerized Web App	Functional (Web App)	Data update	When data is added to database, web app will update accordingly
	Web App frontend		Data download	Data can be downloaded from the website in a useable format
	Web App backend		Data visualization	Graphics on the web app seamlessly incorporates new data

Conclusion

The methodology used in the sprints were effective and we satisfied the success criteria to some extent. What we didn’t accomplish was more due to external factors rather than the methodology itself. This project requires a lot of collaboration and includes a physical component, so the disruption in face-to-face collaboration this quarter was a major roadblock.

However, significant progress was made in delving into the previous group’s already existing code and filling in missing elements. Onboarding with the previous team’s work and setting up environments has taken up a lot of time for both of us especially with limited access to resources or help, but we’ve managed to get our environments working and now both have physical boards to work with, and gotten familiar with what gaps were needed to be filled in the existing code-base. We have made improvements to the android app gui and made progress on converting the buoys’ data to something usable.

Overall, the sprints were fairly successful despite the lack of access to necessary tools. The sprints and task assignments helped give structure to the work being completed and ensured we stayed on track. The initial planning was not particularly useful simply because a large portion of the project was already completed or planned out by the previous team, but overall the project report checkpoints were useful in maintaining the momentum of progress.