



Deakin University

KelpTheWorld

Project Proposal

Project Sponsor

Software Engineering 2 Unit Chair, Dr Kevin Lee

Project Team

Team #

Bronte Jurgens, 217015344

Greg McIntyre, 218356779

Sean Pain, 218137385

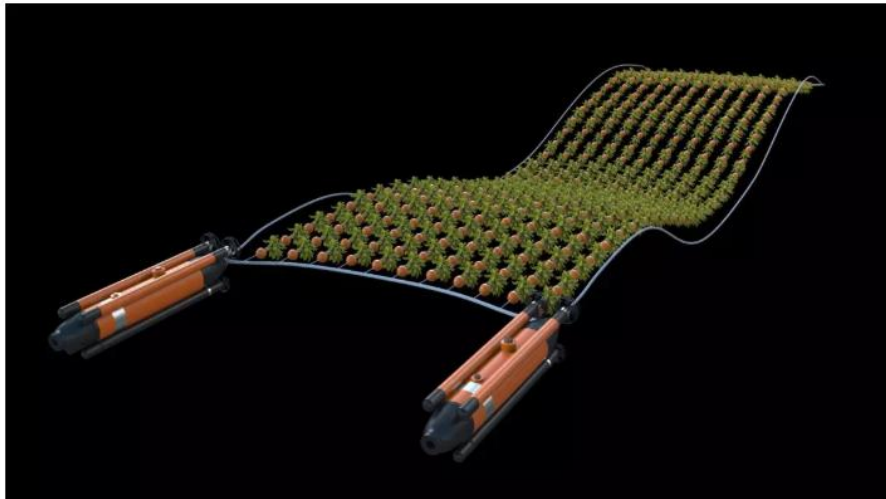
Document Version 1.0

Document Revision History

Date	Version	Editor	Reason	Supervisor Signature	Client Signature
28/08/2019	1.0	Greg M	Update	GM	
30/08/2019	1.0	Greg M	Update	GM	

Motivation / Problem Description

- A floating kelp farm is a concept that would benefit from a real-life prototype.
- There is a concept to create frames of kelp farms that float just below the surface and constantly moved downward to accommodate the growth of the kelp, pressure changes at depth and the amount of buoyancy/water displacement will required will require constant management, the kelp requires a certain temperature of water, amount of light to remain healthy. An independent pump with its own buoyancy control will also be required to move cold water from the depths, mass monitoring will be useful for knowing when forests should be harvested/managed, location tracking will be important for management during storms and other emergencies.
- The 'ultimate goal' would be to create an opensource, robust, scalable, cost efficient, easily deployable, prototype of this frame, with mass sensors, light sensors, pressure sensors, gyroscopes, GPS. And have various interfaces for different roles to monitor huge numbers of frames; farmers, maintenance people, clients, environmental groups, press etc.
- This product is a concept only, this would be the first public prototype.



Fast Company Kelp Farm Concept [3]

References

1. <https://oceana.org/blog/seaweed-could-be-scrubbing-way-more-carbon-atmosphere-we-expected>
2. <https://carboninstitute.org/kelp-and-carbon-sequestration-bringing-terrestrial-carbon-accounting-to-the-deep-sea/>

Context

- Climate change is going to be an issue by 2040. We are past the tipping point with the amount of carbon that has been released into the atmosphere and it is no longer viable to just stop carbon emissions, we also need to begin claiming carbon back. Bull kelp is the fastest growing plant on earth, it can reclaim carbon, reduce acidity levels in the water, provide food and shelter for marine life and provide a viable food source for humans.

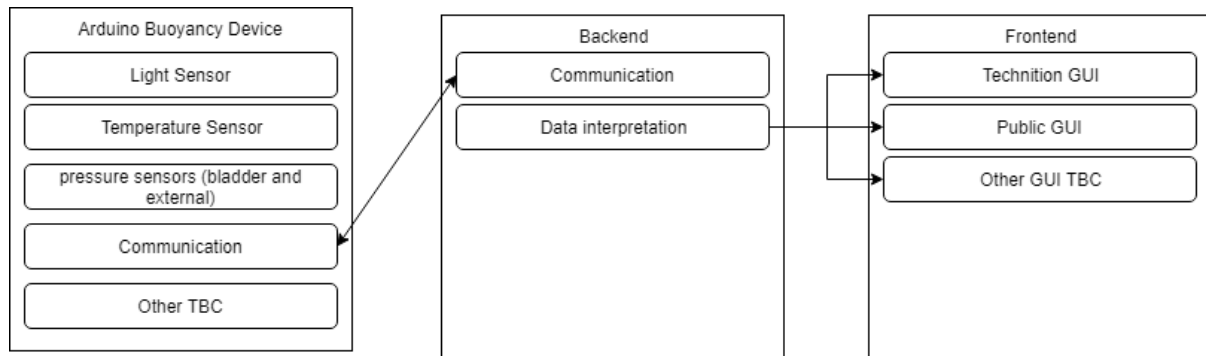
- This concept is implementable with our current levels of technology
- The lack of funding for environmental/climate change projects means there has been very little interest in the project.

References

3. <https://www.fastcompany.com/40458564/could-these-robotic-kelp-farms-give-us-an-abundant-source-of-carbon-neutral-fuel>
4. <https://theconversation.com/how-farming-giant-seaweed-can-feed-fish-and-fix-the-climate-81761>
5. <https://blogs.scientificamerican.com/observations/soil-and-seaweed-farming-our-way-to-a-climate-solution/>

Core Idea/User Stories/Requirements

- Create a prototype of the floating kelp forest, all its web back end and GUI.



Target Deliverables

The following goals have been identified as dependencies that need to be addressed early in the life cycle of the project.

1. Create a service API for KelpTheWorld
2. Create an Arduino frame concept to receive sensor data and transmit to web server
 - a. Mass
 - b. Light
 - c. Pressure
 - d. Buoyancy control
3. Create an Arduino control for water pump control
 - a. Water volume
 - b. Temperature
 - c. Buoyancy control
4. Create a web server to receive data from various farms
 - a. MQTT (farm/1, farm/2, pump/1, pump/2)
 - b. Translation of raw data into usable information.
5. Create a web interface to display data
 - a. Maintenance
 - b. Farmers
 - c. Public

Roadmap

The roadmap to the execution and delivery of this project is detailed subsequently.

Execution Strategy

- Explore the input data provided and confirm if acceptable for the focus of Proof of Concept
- Incrementally,
 - Build and deliver a floating, self-managing kelp farm prototype with web integrated communication for management.
- Prepare research report

Sprint 1

Goals

The goal of Sprint 1 is to deliver scope document and work with the client to agree on the acceptance criteria and priority for the deliverables. These can be decomposed to:

- Project success criteria
- Problem domain clarification
- Visual depiction of the workflow in a flow chart of the processes that this project will automate
- Draw the system design for the project
- Determine what physical system will be created and how it will interact with the application

Target deliverables

- Workflow flowchart that has been agreed upon by all parties
- Scope document (this document) that has been agreed upon by all parties
- Communication and delivery expectations that has been agreed upon by all parties
- Set up the back-end of the app and its interaction with any necessary API's

Sprint 2

Goals

The goal of Sprint 1 is to deliver the end to end infrastructure so we can start collaboratively planning the interfaces to enable integration efforts to commence on client's side. These can be decomposed to:

- Get an end to end solution working
- Collaboratively create an output data format / schema (in collaboration with client)
- Prepare a suitable environment within a docker container to encapsulate and execute the transformation process

Target deliverables

- A docker container encapsulating the transformation engine
- An invocation script that accepts the input folder, output folder and invocation parameters
- A deployment document that describes how to install and use the solution

Sprint 3 (TBC)

Goals

The goal of Sprint 3 is to build upon the Increment in Sprint 2, namely by adding:

- Error logging
- Input/Output Validation
- Transformation logic

Target deliverables (can be amended based on how Sprint 2 goes)

- A docker container encapsulating the transformation engine
- An invocation script that accepts the input folder, output folder and invocation parameters
- A deployment document that describes how to install and use the solution
- List of errors and associated meaning
- Sample dataset to validate the transformation engine reported results

Sprint 4 (TBC)

Goals (can be amended based on how Sprint 3 goes)

The goal of Sprint 3 is to...

- bang
- pow
- smash

Final deliverables (can be amended based on how Sprint 3 goes)

- thing 1
- thing 2
- thing 3
- thing n

Limitations, Constraints and Considerations

The limitations, constraints and considerations of the project are as follows:

The following constraints apply for the PoC and need to be considered when integrating the outputs produced in a larger workflow/pipeline,

- The blah needs to work on AWS as that is technology stack used by (client).
- The transformation engine needs to be in Python as that is technology stack used by (client).
- This project will not focus on UI/UX refinements, instead focusing on implementation of the functionality
- The front end will conform to Web Content Accessibility Guidelines of at least AA.