Chapter 1
Introduction

1.1 Introduction

As the world’s primary means of providing food for its ever-increasing population, agriculture is one of the most crucial parts of the global economy. As the world’s population grows, so does its demand for food. This, in turn, creates a need for a faster and more efficient food production. There are several agricultural practices that have been developed to meet this need. One of those practices is hydroponics, which present quite a few benefits over traditional agriculture. Hydroponics offers faster growth rates, higher overall yields, and reduced water usage (Janes, 2019). However, the use of hydroponics introduces an increase in difficulty. Various environmental factors need to be carefully monitored to ensure optimal growth, such as temperature, humidity, CO_2 levels, pH levels, and nutrient PPM (Janes, 2019). Fortunately, internet of things (IoT) technology has advanced considerably, which has made it easier and cheaper to implement than ever. Utilizing a sensor array utilizing IoT technology will allow farmers to monitor the environmental conditions in real-time. Not only that, but advancements in cloud computing technology have also similarly made it easier and cheaper to implement than ever. Combining the reach provided by IoT and the flexibility provided by cloud can lead to the creation of a truly powerful solution. Thus, I propose the creation of an IoT-enabled hydroponic farm monitoring system using Arduino and cloud.

1.2 Background

Hydroponics technology is not new, the earliest recorded example of a hydroponic farm is the Hanging Gardens of Babylon. Unfortunately, the technology of the past rendered hydroponics a non-viable agricultural practice when compared to traditional agriculture. Due to advancements in technology and the need for more sustainable agricultural practices, hydroponics has seen a rise in popularity and viability (Janes, 2019).

Chapter 2

Theoretical Foundation

2.1 Theoretical Foundation of the Hydroponic Sensor

2.1.1 Arduino Uno

Arduino is an open-source hardware and software company based in Italy that designs and manufactures easy-to-use electronics (*About Arduino*, 2021). Arduino's open-source nature allows it to be easily modified and reproduced to suit the needs of each individual project that involves Arduino. Arduino's easy-to-use nature makes it a viable platform to serve as the basis of this project's hydroponic sensor.

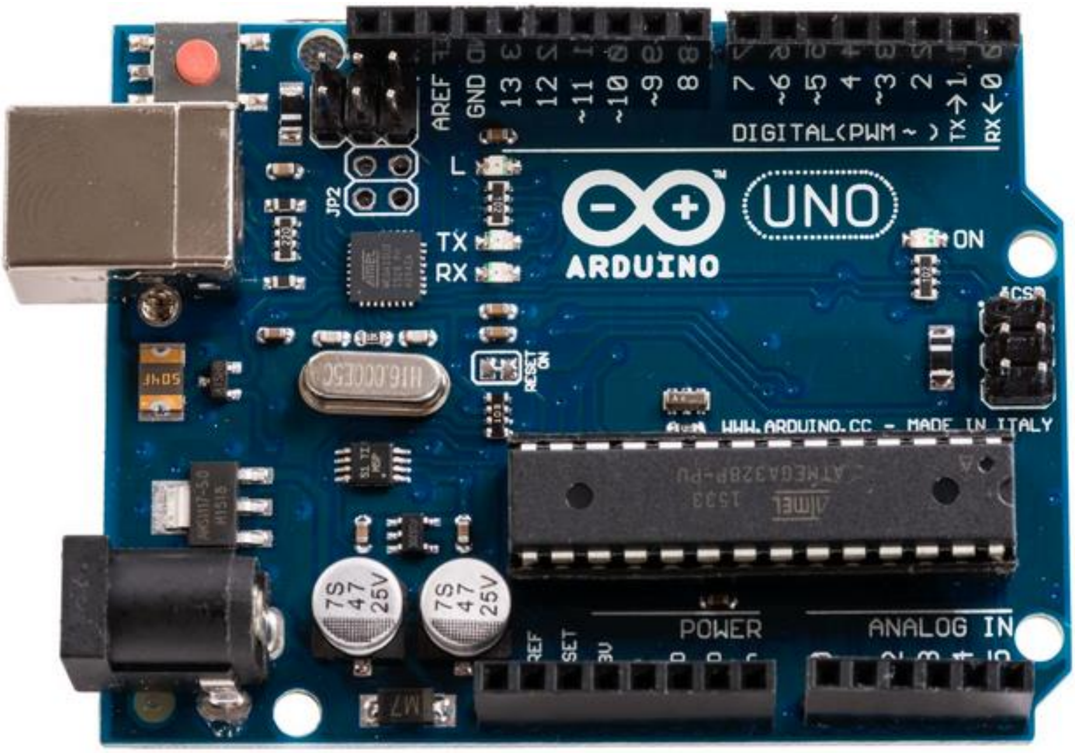


Figure 2.1 – Arduino Uno R3 16U2

2.1.1.1 Specifications

There are many types of Arduino boards available in the commercial market, each with their own microcontroller. The Arduino board chosen for this project is the Arduino Uno R3 16U2, and according to freeCodeCamp.org (2021), it has the following specifications:

1. Microcontroller: ATMEGA328P

Chapter 3
Problem Analysis

3.1 Current Processes

3.1.1 Data Measurement & Collection

Figure 3.1 – The HI98301 TDS Meter

Currently, Just Hydroponics collects and measures data manually using hand-held tool. They primarily use the HI98301, but due to its prohibitively high cost, they only have a small number of these. They make up for the small number of these sensors with much cheaper (and much less accurate and reliable) models. They collect data once a day using these tools.

Collecting data only once a day creates its own problems. Should there be a sudden change in a hydroponic plantation's environment, by the time the staff realizes the undesirable conditions, it would be too late, and the damage would have been done. Just Hydroponics can certainly attempt to mitigate this issue by increasing data collection to several times per day, but scaling in this direction is not sustainable, as it would require a tremendous amount of man hours to meaningfully increase the frequency of data collection for relatively little gain. The amount of effort would also scale up linearly with the size of the farm, which could be changed to scale up logarithmically by introducing automation.

Even though the model illustrated above is the primary model that use, they still utilize other models. This lack of a standardized measuring tool can compromise data quality and consistency because different tools made by different manufacturers will have variations in their tolerances.

Chapter 4

Solution Design

The solution is divided into two major components. The first component is the physical Arduino-based sensor. The second component is the web application which allows for the information collected by the sensor to be displayed to the user.

4.1 Arduino

4.1.1 Architecture Overview

The Arduino-based sensor is responsible for collecting information via an array of five sensors, which are the DS18B20, DHT11, MH-Z19B, PH-4502C, and DFRobot TDS Meter, each of them collecting temperature, humidity, CO_2, pH value, and nutrient ppm respectively.

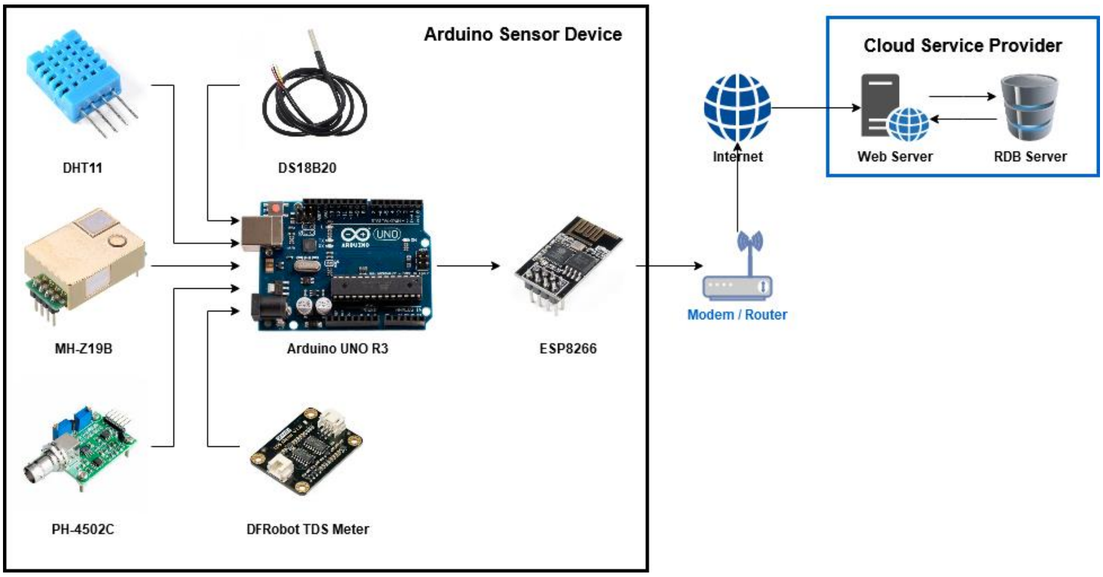


Figure 4.1 - Arduino System Architecture

The data collected by the sensors will then be compiled and formatted into a string URL which adheres to the PHP $_GET superglobal URL pattern. The data will then be sent into the internet via the ESP8266 Wi-Fi module.

This action is performed continuously with a configurable delay so long as the device is powered, and nothing goes wrong in the process cycle. This is called the “Loop Cycle”, which is explained in the point below.

Chapter 5

Solution Implementation

5.1 Arduino

5.1.1 Architecture

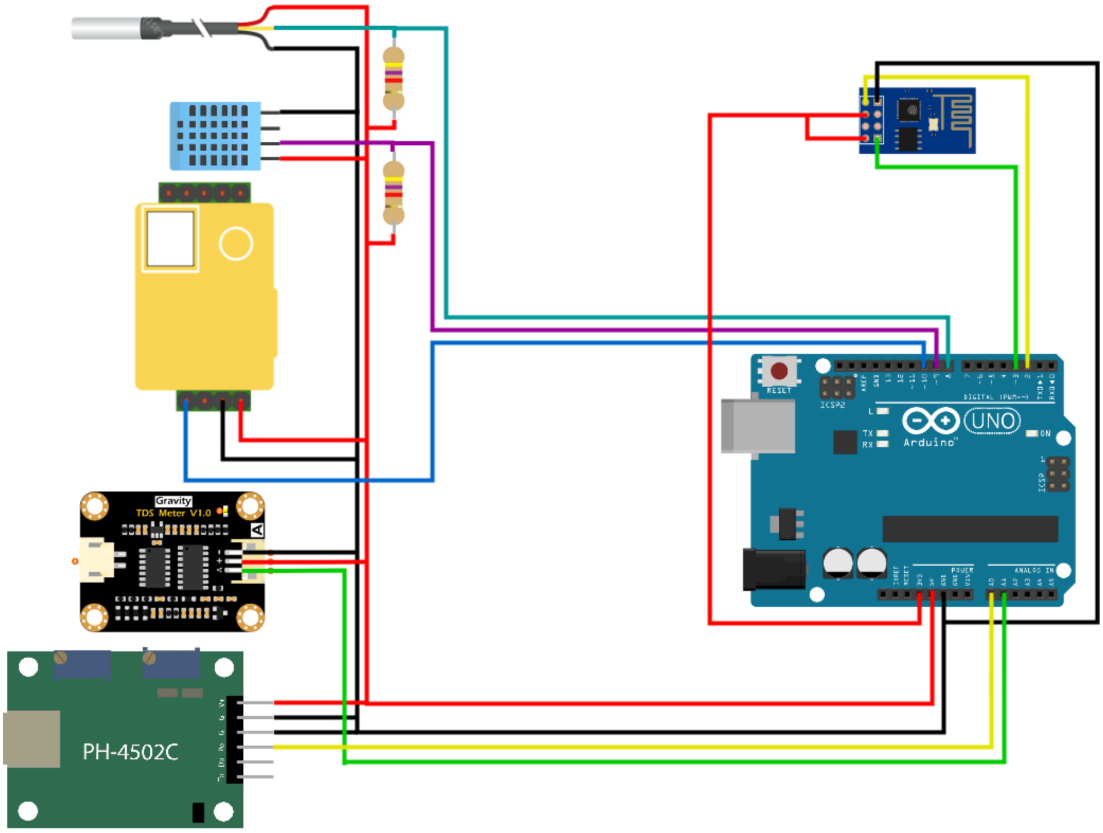


Figure 5.x – Schematic of the Arduino Prototype

The figure above represents the Arduino’s architecture. The real design would be slightly more complicated and less elegant due to the need for a breadboard. The details are as follows:

- The 3.3V power supply is used to power only the Wi-Fi module.
- The 5V power supply is used to power all five sensor modules.
- Data collected from the pH module (PH-4502C) is collected via the Analog port A0.
- Important to note that the PH-4502C has two ground pins. Both should be connected to the Arduino’s ground pin.
- Data collected from the TDS module (Gravity TDS Meter) is collected via the Analog port A1.

Chapter 6
Evaluation & Discussion

6.1 Evaluation

6.1.1 Methodology

The methodology used for evaluating the results of this case study is **Case Study Evaluation**. A Case Study Evaluation is an evaluation methodology that is focused on identifying common themes and patterns to allow better understanding of the challenges and successes experienced by the solution ("Case Study Evaluation Approach", n.d.). The Case Study Evaluation methodology is chosen because it perfectly suits this case study's needs to analyze what went right and what went wrong during the solution's implementation.

It is important to note that because this thesis is a case study, most of the way this thesis is structured adheres to the structure of a Case Study Evaluation. However, for reasons of practicality, this case study's adherence to the formal structure of a Case Study Evaluation is not rigidly enforced.

This formal "Evaluation" chapter will summarize the results of the case study's solution's implementation.

6.1.2 Arduino

6.1.2.1 Unit Testing

The **test condition** for the unit testing of each of these modules are identical to the ones in the **theoretical foundation chapter**. The passing criteria for all the sensor modules are simply to successfully display environmental data 10 times in a row. For the Wi-Fi module, the passing criteria is to successfully insert data into the database 10 times in a row.

- **Wi-Fi Module**
  Result: **Pass**

Chapter 7
Conclusion & Recommendation

7.1 Conclusion

Just Hydroponics currently relies on manual data collection using a set of hand-held measuring tools. This infrequent data collection poses challenges in responding to sudden environmental changes, making it difficult to maintain the optimal conditions for their hydroponic plantations. The absence of standardized measuring tools further jeopardizes data quality, leading to an incomplete picture of the condition of their plantations.

Moreover, the use of whiteboards for data storage is precarious due to data vulnerability. The whiteboards can be easily wiped, leading to potential data loss, while the manual transcription of data is labor-intensive. The current processes are not only time-consuming but also prone to errors, causing inaccuracies and inconsistencies in data.

This manual approach to data collection and management results in a labor-intensive and inefficient allocation of resources, with an inadequate response to environmental changes and delayed troubleshooting. These challenges hinder the scalability and profitability of Just Hydroponics.

To address these problems, an automated system, based on Arduino sensors and a cloud-based web application, is proposed. The transition to an automated approach offers solutions to data vulnerability, data management inefficiencies, labor intensity, reduced accuracy, delayed responses, and inefficient allocation of financial resources.

This case study has managed to successfully produce a working prototype of such a system, albeit only as a proof of concept that is currently commercially unviable. However, it proved that an Arduino-based hydroponic sensor is not only possible to develop, but quite viable. Unfortunately, the limited resources and skillsets (mine alone) allocated to this case study means that commercialization is far

References

About Arduino. (2021, September 15). Arduino. https://www.arduino.cc/en/about

Ada, L. (2015). *Adafruit HUZZAH ESP8266 Breakout*. Adafruit. https://learn.adafruit.com/adafruit-huzzah-esp8266-breakout/overview

Amazon. (2023). *What is AWS?* Amazon AWS. https://aws.amazon.com/what-is-aws/

Amazon. (2023). *What is Amazon Relational Database Service (Amazon RDS)?* Amazon AWS. https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/Welcome.html

Amazon. (2023). *What is Amazon EC2?* Amazon AWS. https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/concepts.html

Arduino. (2023, June 26). *SoftwareSerial Library*. Arduino Documentation. https://docs.arduino.cc/learn/built-in-libraries/software-serial

Barela, A. (2023). *ESP8266 Temperature / Humidity Webserver*. Adafruit. https://learn.adafruit.com/esp8266-temperature-slash-humidity-webserver/overview

Bhimsen. (2022, September 9). *ESP8266 AT Commands List and Working Explained***.** Electronics Fun. https://electronics-fun.com/esp8266-at-commands/

Binance. (2023). *Turing Complete*. Binance Academy. https://academy.binance.com/en/glossary/turing-complete

Bootstrap – The most popular HTML, CSS, and JS library in the world. (n.d.). Bootstrap. Retrieved April 17, 2023, from https://getbootstrap.com/

Campbell, S. (2015, October 1). *How to Set Up the DHT11 Humidity Sensor on an Arduino*. Circuit Basics. https://www.circuitbasics.com/how-to-set-up-the-dht11-humidity-sensor-on-an-arduino/

Case Study Evaluation Approach. (n.d.). EvalCommunity. Retrieved September 26, 2023, from https://www.evalcommunity.com/career-center/case-study-evaluation-approach/

Curriculum Vitae

Name	:	Jason Alexander Tan
Place, Day of Birth	:	Jakarta, 5 November 2001
Sex	:	Male
Address	:	Jl. Tajuk Rencana H-117 Cipinang Muara, Jatinegara, Jakarta Timur 13420
Telephone	:	(+62) 858 8888 9900
Education and Training	:	

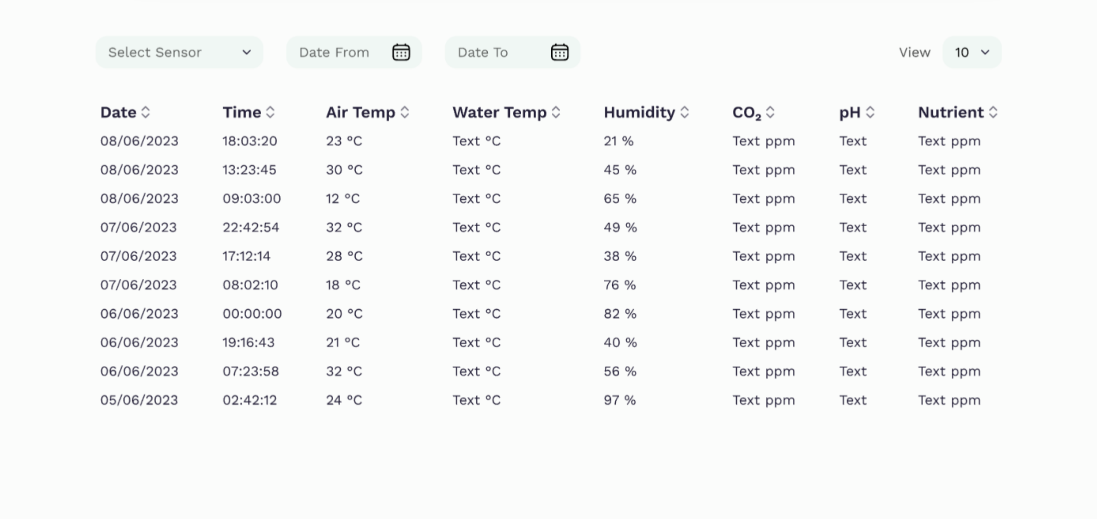
Binus University International	**July 2020 - Present**
Undergraduate, Business Information Systems	Jakarta, Indonesia
SMAK 1 BPK Penabur Jakarta	**July 2017 – Jun 2020**
High School Diploma, School of Science	Jakarta, Indonesia

Work History :

PT. Bank DBS Indonesia	**February 2023 - Present**
Business Platform Management (Internship)	Jakarta, Indonesia
NHS England	**February 2022 – September 2022**
Project Management (Internship, Remote)	Jakarta, Indonesia

Appendices

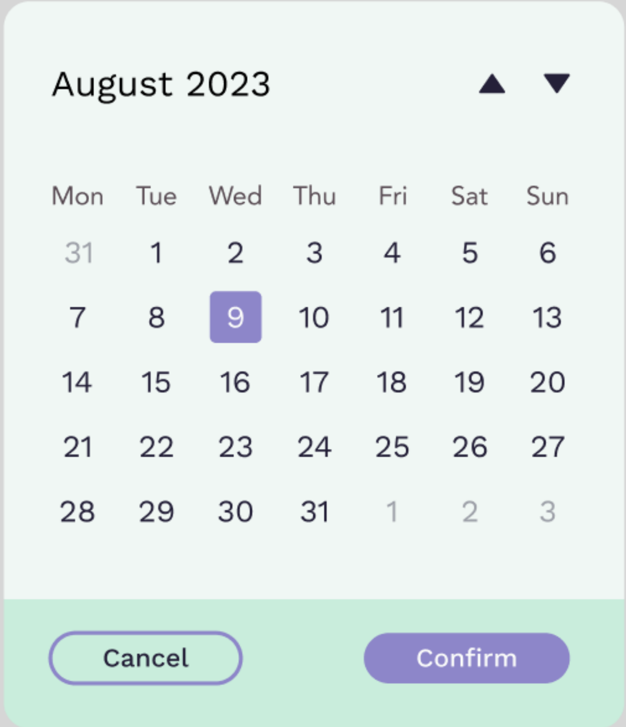
Appendix 1: UI Design Suggestions



View 10

Date	Time	Air Temp	Water Temp	Humidity	CO_2	pH	Nutrient
08/06/2023	18:03:20	23 °C	Text °C	21 %	Text ppm	Text	Text ppm
08/06/2023	13:23:45	30 °C	Text °C	45 %	Text ppm	Text	Text ppm
08/06/2023	09:03:00	12 °C	Text °C	65 %	Text ppm	Text	Text ppm
07/06/2023	22:42:54	32 °C	Text °C	49 %	Text ppm	Text	Text ppm
07/06/2023	17:12:14	28 °C	Text °C	38 %	Text ppm	Text	Text ppm
07/06/2023	08:02:10	18 °C	Text °C	76 %	Text ppm	Text	Text ppm
06/06/2023	00:00:00	20 °C	Text °C	82 %	Text ppm	Text	Text ppm
06/06/2023	19:16:43	21 °C	Text °C	40 %	Text ppm	Text	Text ppm
06/06/2023	07:23:58	32 °C	Text °C	56 %	Text ppm	Text	Text ppm
05/06/2023	02:42:12	24 °C	Text °C	97 %	Text ppm	Text	Text ppm

Figure A1.1 – UI Suggestion 1



August 2023

Mon	Tue	Wed	Thu	Fri	Sat	Sun
31	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23	24	25	26	27
28	29	30	31	1	2	3

Figure A1.2 – UI Suggestion 2