SciFish-Smart Bio Floc Monitoring

**LARAIB NOOR**

**EQRA KHATTAK**

Logo, company name

Description automatically generated

DEPARTMENT OF COMPUTER SCIENCES

COMSATS UNIVERSITY ISLAMABAD,

ATTOCK CAMPUS-PAKISTAN

SESSION 2018-2022

SciFish-Smart Bio Floc Monitoring

*Undertaken By:*

**LARAIB NOOR**

CIIT/FA18-BSE-038/ATK

**EQRA KHATTAK**

CIIT/FA18-BSE-024/ATK

*Supervised By:*

**Dr. Saud Khan**

A DISSERTATION SUBMITTED AS A PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING

DEPARTMENT OF COMPUTER SCIENCES

COMSATS UNIVERSITY ISLAMABAD,

ATTOCK CAMPUS-PAKISTAN

SESSION 2018-2022

UNDERTAKEN

We certify that this is my/our own work. The work has not, in whole or in part, been presented elsewhere for assessment. Where material has been used from other sources it has been properly acknowledged. If this statement is untrue, we acknowledge that we will have committed an assessment offence and shall be liable to punishable action under the plagiarism rules of HEC

Laraib Noor Eqra Khattak

FA18-BSE-038 FA18-BSE-024

Dated: \_\_\_\_\_\_\_\_\_\_\_\_ Dated: \_\_\_\_\_\_\_\_\_\_\_\_\_

**FINAL APPROVAL**

Certified that we have read this project report submitted by Ms Laraib Noor and Ms. Eqra Khattak and it is, in our judgment, of sufficient standard to warrant its acceptance by Department of Computer Science, COMSATS University Islamabad, Attock Campus, for the BSSE degree.

***Committee:***

1. External Examiner  **-------------------------**A whiteboard with writing on it

   Description automatically generated with medium confidence**--------------------------------**

Dr. Khashif Sultan

Associate Professor

Department of Computer Science

Bahria University Islamabad

1. Supervisor **---------------------------------------------------------------------------------------------------------------------**

Dr. Saud khan

Associate Professor

Comsats University Islamabad, Attock Campus

1. Head of the Department **-----------------------------------------------------------------------------------------------------------------**

Dr. Khalid Mehmood Awan

Associate Professor HOD(CS)

Comsats University Islamabad,

Attock Campus

**DEDICATION**

To my Loving Parents and Teachers

**ACKNOWLEDGEMENT**

We would like to express our gratefulness to our Supervisor Dr. Saud Khan who provided us with the vision for this project and guided us despite his busy schedule and work routine. By working on this project, we learned how to integrate different sensors with Arduino. The experience we gained while working on this project is invaluable, and no doubt, it will help us in the future.

**PROJECT BRIEF**

PROJECT NAME SMART BIOFLOC MONITORING

ORGANIZATION NAME COMSATS UNIVERSITY ISLAMBAD, ATTOCK CAMPUS

OBJECTIVE TO MONITOR PH, TEMPERATURE AND WATER LEVEL OF FISH TANK

UNDERTAKEN BY LARAIB NOOR

EQRA KHATTAK

SUPERVISED BY DR SAUD KHAN

LECTURER

CS DEPARTMENT

COMSATS ATTOCK

STARTED ON FEB 2021

COMPLETED ON JUN 2022

COMPUTER USED DELL INSPIRON 500 CORE i5

SOURCE LANGUAGE DART

OPERATING SYSTEM WINDOWS 10 PRO

TOOLS USED ANDROID STUDIO

ARDUINO IDE

**ABSTRACT**

Smart Bio Floc Monitoring is a system consisting of a mobile application (SciFish), sensors to measure pH, temperature, oxygen, water level connected through wifi by esp8266 making a WSN(Wireless Sensor Network). It will contain various sensors to look at the changed properties of water. In Pakistan, there is no such system as to monitor a whole floc of fishes in a fish farm farmers must maintain conditions for better fish life manually. Also, there is no consultancy for farmers to counsel for the development and medical problems of fishes. The thought is fundamental for those farmers who are neglected to develop their business at a high proportion. The project contains a great deal of data and grandiosity in a novel technique for the fresher who needs to begin their startup as a fish farmer. Feed given to the fishes is determined by the no. of fishes and dosage per day. Fish mortality rate is being optimized through SciFish by maintaining conditions that are necessary for fish life. Whenever the conditions being monitored deviate from normal the farmer is notified through application notifications.

**Table of Contents**

CHAPTER 1 INTRODUCTION……………………………………………………………1

[1.1 Introduction 2](#_Toc105745558)

[1.1.1 Brief 2](#_Toc105745559)

[1.1.2 Objectives 3](#_Toc105745560)

[1.1.3 Hardware Design 4](#_Toc105745561)

[1.1.4 Components 4](#_Toc105745562)

[1.1.4.1 ESP8266 4](#_Toc105745563)

[1.1.4.2 pH Sensor 4](#_Toc105745564)

[1.1.4.3 Temperature Sensor 5](#_Toc105745565)

[1.1.4.4 Raindrop Water Level/Height Depth Detection Sensor Module for Arduino 6](#_Toc105745566)

[1.1.5 Tools and Technologies 7](#_Toc105745567)

[1.1.5.1 Android Studio Chipmunk 2021.2.1 7](#_Toc105745568)

[1.1.5.2 Flutter 3.0.1 7](#_Toc105745569)

[1.1.5.3 Dart 2.17 7](#_Toc105745570)

[1.1.5.4 Arduino IDE 2.0 7](#_Toc105745571)

[1.1.5.5 Firebase 7](#_Toc105745572)

[1.1.6 Relevance to Course Modules 7](#_Toc105745573)

[1.1.7 Project Background 8](#_Toc105745574)

[1.1.8 Literature Review 8](#_Toc105745575)

[1.1.9 IOT Based Fish Aquarium 8](#_Toc105745576)

[1.1.10 System Conceptual Diagram 9](#_Toc105745577)

[1.1.11 Analysis from Literature Review 10](#_Toc105745578)

[1.1.12 Methodology and Software Lifecycle for This Project 10](#_Toc105745579)

[1.1.13 The Rationale behind Selected Methodology 10](#_Toc105745580)

**CHAPTER 2 PROBLEM STATEMENT…………………………………………………12**

[2.1 Problem Statement 13](#_Toc105745582)

[2.2 Deliverables and Development Requirements 14](#_Toc105745583)

[2.3 User Interface 14](#_Toc105745584)

[2.4 Development Requirement Including Following Software Requirements 14](#_Toc105745585)

[2.4.1 Frontend 14](#_Toc105745586)

[2.4.2 Backend 14](#_Toc105745587)

[2.4.3 Arduino 14](#_Toc105745588)

**CHAPTER 3 REQUIREMENT ANALYSIS…………………………………………….15**

[3.1 Use Case Diagram(s) 16](#_Toc105745592)

[3.2 Sequence Diagram 17](#_Toc105745593)

[3.3 Detailed Use Case 17](#_Toc105745594)

[3.3.1 Authentication 17](#_Toc105745595)

[3.3.2 Area of Pond 18](#_Toc105745596)

[3.3.3 Result from Sensors 18](#_Toc105745597)

[3.4 Functional Requirements 18](#_Toc105745598)

[3.5 Non-Functional Requirements 19](#_Toc105745599)

**CHAPTER 4 DESIGN AND ARCHITECTURE…………………………………………21**

[4.1 System Architecture 22](#_Toc105745604)

[4.2 Data Representation [Diagram + Description] 23](#_Toc105745605)

[4.3 Process Flow [Representation] 23](#_Toc105745606)

[4.4 Data Flow Diagram [Level Zero] 24](#_Toc105745607)

[4.5 Data Flow Diagram [Level 1] 24](#_Toc105745608)

[4.6 Tree Structure for Firebase 25](#_Toc105745609)

**CHAPTER 5 IMPLEMENTATION………………………………………………………26**

[5.1 Tools and Technologies 27](#_Toc105745611)

[5.1.1 Tools 27](#_Toc105745612)

[5.1.2 Languages 27](#_Toc105745613)

[5.1.3 Libraries &Framework 27](#_Toc105745614)

[5.1.3.1 Hardware 27](#_Toc105745615)

[5.1.3.2 Libraries 27](#_Toc105745616)

[5.2 Development Stages 28](#_Toc105745617)

[5.2.1 Firebase Firestore Integration 28](#_Toc105745618)

[5.2.2 Hardware Set 29](#_Toc105745619)

[5.2.3 PH Calculation 30](#_Toc105745620)

[5.2.4 Pond Temperature 30](#_Toc105745621)

[5.3 User Interface 31](#_Toc105745622)

[5.3.1 Home 31](#_Toc105745623)

[5.3.2 Power Screen: 32](#_Toc105745624)

[5.3.3 Fish Life Screen: 34](#_Toc105745625)

[5.3.4 Water Level Screen 35](#_Toc105745626)

*[5.3.5](#_Toc105745627)* [Food Screen 36](#_Toc105745627)

*[5.3.6](#_Toc105745628)* [Oxygen Level Screen 37](#_Toc105745628)

[5.3.7 pH Screen 38](#_Toc105745629)

[5.3.8 Temperature Screen 39](#_Toc105745630)

[5.3.9 Air Flow Screen 40](#_Toc105745631)

**CHAPTER 6 TESTING AND EVALUATION…………………………………………..41**

[6.1 Manual Testing 42](#_Toc105745633)

[6.1.1 System testing 42](#_Toc105745634)

[6.1.2 Unit Testing 42](#_Toc105745635)

[6.1.3 Function Testing 42](#_Toc105745636)

[6.1.4 Test Cases Results 43](#_Toc105745637)

[6.1.4.1 User Sign-in 43](#_Toc105745638)

[6.1.4.2 User Sign-up 44](#_Toc105745639)

[6.1.4.3 Configure hardware 44](#_Toc105745640)

[6.1.4.4 Stock 45](#_Toc105745641)

[6.1.4.5 Logout 45](#_Toc105745642)

[6.2 Integration Testing 46](#_Toc105745643)

**CHAPTER 7 CONCLUSION AND FUTURE WORK…………………………………..47**

[7.1 Conclusion 48](#_Toc105745645)

[7.2 Future work 48](#_Toc105745646)

**CHAPTER 8 REFERENCES………………………………………………………………49**

[8.1 References 50](#_Toc105745648)

**List of Figures**

[Figure 1.1 ESP8266 4](#_Toc105745649)

[Figure 1.1 pH Sensor Kit 5](#_Toc105745650)

[Figure 1.3 Temperature Sensor 5](#_Toc105745651)

[Figure 1.4 Water level sensor analogue 6](#_Toc105745652)

[Figure 1.4 Water level sensor digital 6](#_Toc105745653)

[Figure 1.6 System Conceptual Diagram 9](#_Toc105745654)

[Figure 1.2 Agile Model 11](#_Toc105745655)

[Figure 3.1 Use Case Diagram 16](#_Toc105745656)

[Figure 3.2 Sequence Diagram 17](#_Toc105745657)

[Figure 4.1 System Architecture 22](#_Toc105745658)

[Figure 4.2 Process Flow 23](#_Toc105745659)

[Figure 4.3 Level 0 Data Flow Diagram 24](#_Toc105745660)

[Figure 4.4 Level 1 Data Flow Diagram 25](#_Toc105745661)

[Figure 4.5 Firebase Tree Structure 25](#_Toc105745662)

[Figure 5.1 Firestore Connection 29](#_Toc105745663)

[Figure 5.6 Firebase Node 29](#_Toc105745664)

[Figure 5.7 ISO CODE OF PH 30](#_Toc105745665)

[Figure 5.9 Pond Temperature Calculation 31](#_Toc105745666)

[Figure 5.10 Home Screen 32](#_Toc105745667)

[Figure 5.11 Config hardware 33](#_Toc105745668)

[Figure 5.12 Fish Life screen 34](#_Toc105745669)

[Figure 5.13 Water level Screen 35](#_Toc105745670)

[Figure 5.14 Food Screen 36](#_Toc105745671)

[Figure 5.15 Oxygen Level Screen 37](#_Toc105745672)

[Figure 5.16 pH Screen 38](#_Toc105745673)

[Figure 5.17 Temperature Screen 39](#_Toc105745674)

[Figure 5.18 Air flow Screen 40](#_Toc105745675)

**List of Tables**

[Table 3.1 Authentication 17](#_Toc105661467)

[Table 3.2 Area of Pond 18](#_Toc105661468)

[Table 3.3 Result from Sensors 18](#_Toc105661469)

[Table 3.4 Non-Functional RE#01 19](#_Toc105661470)

[Table 3.5 Non-Functional RE#02 19](#_Toc105661471)

[Table 3.6 Non-Functional RE#03 19](#_Toc105661472)

[Table 3.7 Non-Functional RE#04 20](#_Toc105661473)

[Table 3.8 Non-Functional RE#05 20](#_Toc105661474)

[Table 6.1 Sign-in 43](#_Toc105661475)

[Table 6.2 User Sign-up 44](#_Toc105661476)

[Table 6.3 Configure hardware 44](#_Toc105661477)

[Table 6.4 Stock 45](#_Toc105661478)

[Table 6.5 Log-out 45](#_Toc105661479)

[Table 6.6 Integration testing 46](#_Toc105661480)

**LIST OF ABBREVIATIONS**

SRS Software Require Specification

PC Personal Computer

PH Power of Hydrogen

DO Dissolve Oxygen

OOP Object-Oriented Program

OTP One Time Password

IoT Internet of Things

Chapter 1

Introduction

# Introduction

Traditional fish farming is not sensitive to aquatic life as the ponds are already made, there’s always more water, more space, more fish and more of about everything but it has its perks. Such as, it can prove too unhealthy and contaminated for the fish to survive. The waste excreted by the fish and the leftover food can cause the formation of ammonia gas which is harmful for fish. In contrast, Bio floc is a symbiotic process that includes confined aquatic animals, heterotypic bacteria, and microbial species in the water. Bio floc provides environment that is good for fish growth and survival providing certain parameters are fulfilled. Consumption of bio flocs also provides nutritional value to cultured aquatic species.

Smart Bio Floc Monitoring is the monitoring of a bio floc in an artificially generated pond. This project will be an ultimately huge helping guide for bio floc fish farmers to do successful bio floc farming, from seed deployment to selling fish in the market. It will be looking into various factors such as monitoring sensors and their integration in the mobile app to look for any slight change in the properties of water or food. For example, dissolved oxygen level detection, temperature, pH, and water level etc. Also, to undergo the survival rate of fish in a pond according to its symmetry and then provide tips for its betterment. The sensors are directly connected to the mobile app and can be used remotely from anywhere to keep an eye on things. It can be used for both natural and artificial ponds.

## Brief

Smart Bio Floc Monitoring is a project based on a controlled synthetic environment which is highly suitable for survival and growth of a floc. Automated by a mobile application that runs on both Android and iOS with a single codebase Dart. This project is a simple program for a bio-floc fish farmer to help him in doing his job properly and efficiently with minimum efforts.

Computer literacy is a major issue in our country for farmers that are either on a small scale or in a large-scale production. They lack critical information about most of their farming practices and are unaware of the upgrade in resources. Numerous fish die in their ponds due to poor consideration.

Specifically, in Pakistan there’s no consultancy for farmers to consult for the growth, fertilization, and health issues of fish. Our objective is to provide them with the solutions to their problems so they can become better and successful fisherman of their region.

Considering, the farmers must create artificial ponds in order to grow fish in ideal environments, there might be an issue with the area and the design of the pond. The problems can be a rectangular shaped pond which might stop the fish in their paths as fish tend to keep swimming most of the time, to avoid that, we decide to shape it circular.

It’s a fact that fish requires utmost care as their environment is different from land animals, so they need specific conditions like oxygen level, temperature, pH, water level, food etc. Fish feed is deployed according to the pound areas but first these parameters must be satisfied which will ultimately lead to reduction in fish death rate.

Survival area for 1000 fishes in a pond is 1 square meter. If 1 Acre farm is converted in to 1 single tank, a 1000 fish can survive. Parameters that must be met for bio floc are Temperature, Dissolved Oxygen, pH, Salinity, Solids (total suspended solids and settling solids), Alkalinity, and Orthophosphate. So, for its management, a technology called WSN (Wireless Sensor Network) is used in this project. WSN monitors and records the physical conditions of the environment and organize the collect data at a central location.

This project will decrease labor time and increase efficiency as it can be controlled from a mobile phone when installed on an Android or an iPhone. The sensors attached to the ponds will be integrated in the mobile app and the user will continuously examine the situation of the pond such as water quality and environment for better growth. In addition to that our app will advocate the best marketplace and farmers can manage all expenses and stocks record.

## Objectives

The objectives of this application are as follows:

* Wireless connection through Wi-Fi with sensors allowing us to monitor from a distance saving the data in Firebase Firestore.
* To check the level of ammonia by checking its PH.
* To observe the water temperature according to the ideal temperature conditions for certain fish.
* Observe the amount of dissolved oxygen in the water based on number of fish and its providence through air pump.
* Deployment of fish feed in tanks based on the amount set in the app.
* Users will be notified when there is a change in the ideal conditions set for certain fish in the tank.
* To fetch the data from database to the mobile application.
* Monitoring the fish through the camera.

## Hardware Design

In designing the hardware of our project, we have integrated all the components. The detailed explanation of each component is and its working has been explained in this chapter. We have also explained how different components are linked to each other.

## Components

In Hardware architecture, we have used the following components.

### ESP8266

The ESP8266 Wi-Fi Module is an independent SOC (system on chip) that can help any microcontroller to connect to the Internet with dual-mode Bluetooth and Wi-Fi. You can essentially attach this to your Arduino gadget and get about as much Wi-Fi-capacity as a Wi-Fi Shield offers. The ESP8266 includes built-in antenna switches, RF balun, power amplifier, low- noise receive amplifier, filters, and power management modules. ESP8266 is created by a Shanghai-based Chinese company and is manufactured by TSMC using their 40 nm process.

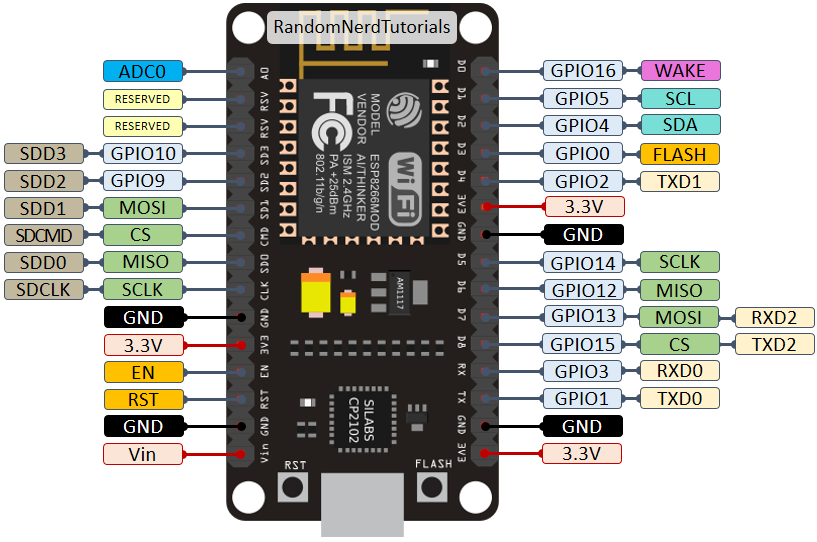


Figure 1.1 ESP8266

### pH Sensor

PH sensor is used to measure hydrogen ion concentration in a solution. Glass pH electrode is widely used in pH sensors. The electrode is the main part of measuring the pH in a solution. It works on the principle of voltmeter and uses a potential difference to check solution voltages and compare them with existing ones. The ideal value for a solution should be pH=7 and if it is more than 7 it will a basic solution and if pH is less than 7then the solution will be acidic.



Figure 1.1 pH Sensor Kit

### Temperature Sensor

Temperature sensor plays an important role in many applications like in the case of fish farming it is necessary to check the temperature. Temperature sensors are usually thermocouple or RTD. We have used a thermistor-based temperature sensor, which is capable of monitoring water temperature. It works on the inverse time characteristics phenomena. The resistance of the thermistor decreases when the temperature increases and gives the signal of rising in temperature.



Figure 1.3 Temperature Sensor

### Raindrop Water Level/Height Depth Detection Sensor Module for Arduino

The water level sensor is super easy to use and only has 3 pins to connect. S (Signal) pin is an analog output that will be connected to one of the analog inputs on your Arduino. + (VCC) pin supplies power for the sensor. It is recommended to power the sensor with between 3.3V – 5V.

The below figure shows analogue water level sensor and a digital water level sensor that we used in the project due to lesser number of pins.



Figure 1.4 Water level sensor analogue



Figure 1.4 Water level sensor digital

## Tools and Technologies

The tools and technologies which we will be using for our project are:

### Android Studio Chipmunk 2021.2.1

Android Studio is the IDE that provides Google to develop professional Android applications. It is used to develop a different variety of applications for the Android operating system. It is an IDE & platform to design a user-friendly interface by drag and drop. Firstly, we started up on Arctic Fox 2020.3.1, then Bumblebee 2021.1.1 and project ended at the android studio version Chipmunk 2021.2.1.

### Flutter 3.0.1

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, desktop, and embedded devices from a single codebase.

### Dart 2.17

Dart is an Open-Source, client-side programming language. It is easy to learn, stable, and creates high-performance applications.

### Arduino IDE 2.0

The open-source Arduino Software (IDE) makes it easy to compose code and transfer it to the board.

### Firebase

For database operation firebase will be used to store the records.

## Relevance to Course Modules

As we studied different courses in the previous semester like Report writing skills. This course helps me a lot in improving my English and I am writing this report with help of it also programming, Database, software engineering, OOP in java, android, Applied physics for engineers. Android would be used for Designing interface and other functionality will be done by using different concepts of programming languages. In this project. Applied physics and Microprocessor and Assembly language will be used in Arduino Uno and hardware related work like Arduino Uno and for different sensors calibrations like PH, temperature, etc. In addition to that, the firebase database will also use for storing data. Software engineering will be favorable for us in making the use case diagrams activity diagrams.

## Project Background

The idea behind this project particularly occurred in our mind when the business encountered such Problems in fish farming. So, we prefer this idea because this is a profitable business in our country there are a lot of apps available which provide a guideline about the farming of fish some provide only tutorials which are not enough for a farmer, some app provided techniques, but they use some other languages, the only need is a farmer to have proper knowledge regarding how to feed fish, how to check the PH of the water. Also, lower-level farmers can use this application as our main target is to provide a maximum guideline about fishes that will help the farmers to grow their business on a high level. The feed is the main thing for fishes also users can check weather prediction this will be done by using different sensors like PH sensors as fish is dependent upon water to grow and breath, understanding the physical quality and chemical quality of water because success and failure depend on it. Less than 7 PH and greater than 8 PH are not suitable for fishes. Young fish may die to the PH level below. High PH levels above between 8 to 14 pH can be harmful for fish. One more sensor will be used i.e., Temperature sensor and weather predictions API will be used. For Selling sell their fishes Fish markets information will also be given. Users will get to know that how much seed they can deploy in the pound by providing the basic information of the pound.

## Literature Review

We look at the current system that is in use and state its strengths and weakness. Pakistan is an agricultural country, acknowledging the fact that fisheries play an important role in the life of humans and country development. In the past few years, fish farmers are facing numerous challenges such as type checking method for detecting fish’s health. These factors greatly affect the Economy of the country and food security. If we look at recent past years, there were several detection systems exist but when we talk about fish farming in Pakistan there is no consultancy for farmers to grow and health issues of fishes. By the time previous ways are replacing by the most efficient and latest ways, some current systems were made for the identification of fish.

## IOT Based Fish Aquarium

IoT Based fish aquarium is a fully automated and remotely monitored aquarium, fully capable of operating without human intervention or interaction. Major features of this project are the mechanical feed design, fish feeding over the internet, and remote monitoring of all the parameters. The parameters include feed level, water temperature, pH, and water level. They can be accessed through a website on a computer or through the app on a phone.

## System Conceptual Diagram

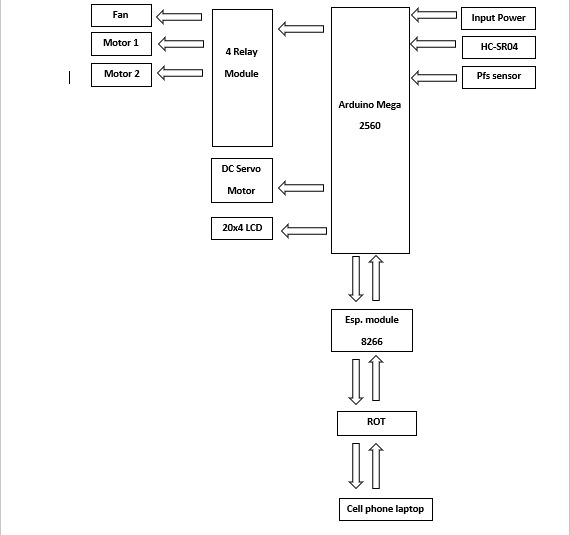


Figure 1.6 System Conceptual Diagram

In this project, they have used different types of sensors, which are connected to the microcontroller. Arduino gets the data from sensors, controls, and all the processes according to the commands given to it. The water level can also observe and refill the tank. This project has the following objectives

* To monitor the water level of the aquarium.
* To automate the refilling and drainage systems of the aquarium.
* In case if the aquarium is leaking or the water level drops to some extent it will refill the tank.
* To automatically feed the fishes at a specific interval of times.
* To check the purity of water by checking its ph.
* To transfer all the data to the internet.
* To make a system that can display and monitoring the sensor’s data online in real- time.
* To display the useful data of the aquarium on the screen

## Analysis from Literature Review

A thought comes to our mind from IoT based systems because humans do not do routine tasks very well and must discipline themselves over time, but machines or computers do not have these limitations. Every individual nowadays has cell phones, so our point is to create an android application, which handy & on the Go. We will provide the best solutions to their problems, attempting to incorporate everything in one application and propose the best market to sell their stocks. Our application will be beneficial for uniquely little scope farmers.it is effectively replaceable by a cell phone which is portable, time-Saving, and accessible at any time.

## Methodology and Software Lifecycle for This Project

The fundamental and building block of a project is to describe the methodology. The methodology is essential for managing the project. We make an all-around considered structure record, there are a few highlights that despite everything do not give a similar impact as it gave on the paper. During the execution stage, numerous highlights are included or altered. One needs to make a few changes in system design and requirement analysis. For creating of "Ultimate fishing guide", we have chosen a flexible model type that can be molded easily after designing. Used as an agile development methodology for the development of our model.

## The Rationale behind Selected Methodology

Agile development is an interactive software development strategy that involves cooperation and self-organizational cross-functional teams. This involves the delivery of the project incrementally rather build and deliver the whole project at once. We must deliver a progress report to the supervisor on an incremental basis and on that we gradually develop this application. Henceforth, agile is a combination of methodology and technique.



Figure 1.2 Agile Model

Chapter 2

Problem Statement

# Problem Statement

Considering the farmers must create artificial ponds in order to grow fish in ideal environments, there might be an issue with the area and the design of the pond. The problems can be a rectangular shaped pond which might stop the fish in their paths as fish tend to keep swimming most of the time, to avoid that, we decide to shape it circular.

It’s a fact that fish requires utmost care as their environment is different from land animals, so they need specific conditions like oxygen level, temperature, pH, water level, food etc. Fish feed is deployed according to the pound areas but first these parameters must be satisfied which will ultimately lead to reduction in fish death rate.

Survival area for 1000 fishes in a pond is 1 square meter. If 1 Acre farm is converted in to 1 single tank, a 1000 fish can survive. Parameters that must be met for bio floc are Temperature, Dissolved Oxygen, pH, Salinity, Solids (total suspended solids and settling solids), Alkalinity, and Orthophosphate. So, for its management, a technology called WSN (Wireless Sensor Network) is used in this project. WSN monitors and records the physical conditions of the environment and organize the collect data at a central location.

This project will decrease labor time and increase efficiency as it can be controlled from a mobile phone when installed on an Android or an iPhone. The sensors attached to the ponds will be integrated in the mobile app and the user will continuously examine the situation of the pond such as water quality and environment for better growth. In addition to that our app will advocate the best marketplace and farmers can manage all expenses and stocks record.

The major reason behind developing the ultimate fishing guide is to facilitate farmers either old or fresher because the government is assuming no job in preparing fish farming and giving them specialized help. They show their obliviousness on issues of how to think about the strength of the developing seed, their weight, and contrasts between the kinds of seeds Research on the possible species, taking care of and seeding advancement, and medical issues are the need of great importance. Most peoples don’t have access to higher levels for the seed, feed, and recovery from different diseases that a high-level farmer can easily manage thus their business doesn’t have a growth ratio as compared to other successful farmers. Several Fish identification Recommendations, and automated Fish farms controlling systems deployed for farmers using features of fishes to identify its health and maintaining other necessary problems. However, in a case wherein our country these techniques are failed due to ignorance of government we aim to provide an android application. Keeping in view the above performance problems; [Smart Bio floc Monitoring] is introduced for recommending feeds, seeds, and the best market for their stock. Users will get to know that how much seed they can deploy in the pound by providing the basic information of the pound.

# Deliverables and Development Requirements

When we launch our application. Users visit our application and know about the features of the smart Bio floc monitoring app the user must know about some basic knowledge that they can understand easily. While developing the Applications. We use android studio for developing our screens. User will see all the following features in our application.

# User Interface

Android application is developed through which:

* Results from sensors
* PH of water
* Temperature of water
* Temperature of environment
* Water level

# Development Requirement Including Following Software Requirements

## Frontend

Android studio (Flutter)

## Backend

Database: Firebase

Programming Languages: Dart

## Arduino

Attached with the sensors in order to get readings and to save in database.

Chapter 3

Requirements Analysis

# Use Case Diagram(s)

It is the representation of the user with the system. It is used to describe the use cases (set of different actions) that can be performed by the actors (external users) by using the system. With the help of the use case, you can visualize the behavior of the system when the user uses it.

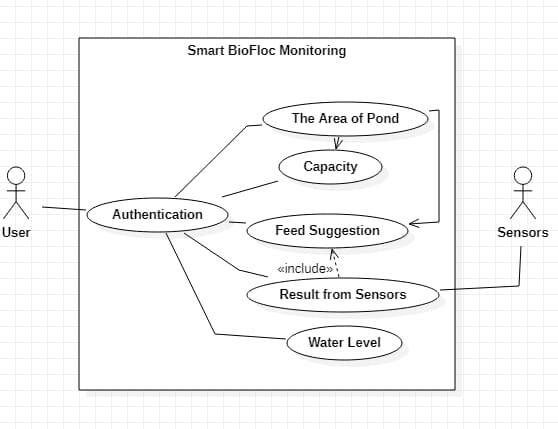


Figure 3.1 Use Case Diagram

# Sequence Diagram

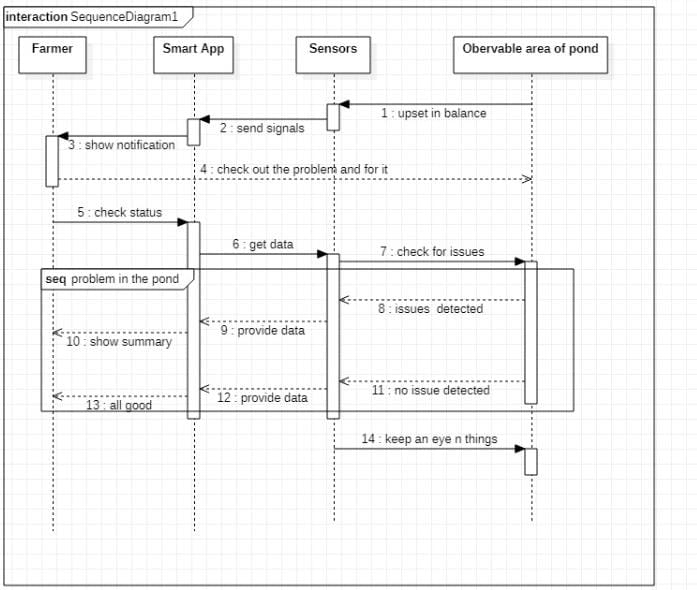


Figure 3.2 Sequence Diagram

# Detailed Use Case

## Authentication

Table 3.1 Authentication

|  |  |
| --- | --- |
| **Use-case name** | Authentication |
| **Actor** | User |
| **Description** | User will authenticate his/her phone No |
| **Pre-condition** | Users Login/Signup. |
| **Post-condition** | The user should get the desired outcome. |

## Area of Pond

Table 3.2 Area of Pond

|  |  |
| --- | --- |
| **Use-case name** | Area of pound |
| **Actor** | User |
| **Description** | The user must provide information about the area of the pound. |
| **Pre-condition** | Users must be using the application. |
| **Post-condition** | The user should get the desired outcome. |

## Result from Sensors

Table 3.3 Result from Sensors

|  |  |
| --- | --- |
| **Use-case name** | Result from Sensors |
| **Actor** | Sensors |
| **Description** | Using this Arduino sensor all the  Important test on the water is done which are important for fishes’ growth. |
| **Pre-condition** | All sensors must be attached to Arduino  and Arduino with our android app. |
| **Post-condition** | Based on test result feed suggestions are  Done. |

# Functional Requirements

Functional requirements are those functionalities that software must offer. It describes the basic behavior of the software. Functional requirements should include the details of functionality or operations provided by each screen. The complete workflow of the software should be made clear.

* Firebase authentication.
* Sensors are connected wirelessly through WSN.
* Temperature sensor gets temp data of water stores it in Firebase Firestore.
* pH sensor data fetch and stored in firebase Firestore.
* Water level observed through sensors and data stored in Firebase Firestore.
* Power the system ON and OFF.
* Camera Live feed in the app.
* Dissolved oxygen level observation and data collection.
* Fish mortality rate.
* Change feed palette in the app and show graphs for all the changing variables in the app.
* App will notify the farmer about the change in set conditions.

# Non-Functional Requirements

Non-functional requirements show that how the system should perform certain functionality. Non-functional requirements generally deal with the quality of the software. Following are the non-functional requirements of this system:

Table 3.4 Non-Functional RE#01

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Performance** | | | | | |
| **Summary** | Performance  application | defines | how | efficient | is | the |
| **Requirement** | The ultimate Fishing guide will be good in a performance like it will provide the user all the necessary information to carry out fishing from seeds to sell them in the market  involving most of the process with details | | | | | |

Table 3.5 Non-Functional RE#02

|  |  |
| --- | --- |
| **Name** | **Portability** |
| **Summary** | Portability means when using an application user is not restricted to use it by sitting on a fixed place or it does not need a different  system to operate |
| **Requirement** | It will be in your android mobile phone with  attached sensors which you always carry with you on your farm side. |

Table 3.6 Non-Functional RE#03

|  |  |
| --- | --- |
| **Name** | **Usability** |
| **Summary** | By usability of software, you mean that it  should be easy to learn. The user should not feel any difficulty in using this system |
| **Requirement** | In Ultimate Fishing Guide, we have included this in our priorities that this application should be user friendly, moreover it should  not be complex in its working mechanism |

Table 3.7 Non-Functional RE#04

|  |  |
| --- | --- |
| **Name** | **Compatibility** |
| **Summary** | It means to ensure that customer is satisfied or  not by testing. |
| **Requirement** | This application is compatible with the android operating system, it is being designed for specifically android OS. |

Table 3.8 Non-Functional RE#05

|  |  |
| --- | --- |
| **Name** | **Maintainability** |
| **Summary** | In which component can be modified to  improve performance. |
| **Requirements** | The requirement of maintainability is easily achieved in this application because first it is providing the functionality of user profile maintainability and all the basic record of the application is saved in the database. |

Chapter 4

Design and Architecture

# System Architecture

Our system architecture consists of 3 main modules: Frontend, Backend and Hardware. The frontend is an interactive interface that allows our users to visit our application and ease of use. Users can monitor the basic requirements for ideal pond conditions such as Oxygen level, pH of water, temperature, and other nutrients crucial for the fishes’ survival. Feed suggestions, latest updates on farming technologies and practices and the nearest Fish Market through Google Map. The results are displayed on the screen. Our project architecture diagram is shown below:

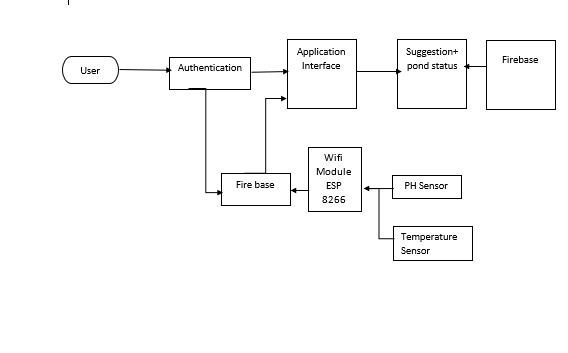


Figure 4.1 System Architecture

This diagram explains how the overall interaction between user and application takes place. A user interacts using the android interface which is used to control the whole process and background processing. The application gets input through different sensors which are PH sensors and temperature sensors also user can check the current weather of his city. Users can add multiples pond information and can easily manage every pond data which includes the area of a pond. This application will suggest the best marketplace to sell fishes users will follow the direction of the market by Google map API. Also, the user will maintain its record in the database.

# Data Representation [Diagram + Description]

It contains different symbols & connectors. It is known as graphical representations between people and objects. An ER model is made of an entity and some specific relationships which exist between entities.

# Process Flow [Representation]

It is a way of representing the flow of data through the system.it provides information about the outputs and inputs of entities and processes.it shows a complete flow of data between system modules. Ultimate fishing guide main modules are a value of PH of the water, suggest Nearby Fish Market, and gives a feed suggestion According to Area of a pound. The application will take input i.e., water pH, temperature from the user, and give results. Following is a data flow diagram of the ultimate fishing guide app.

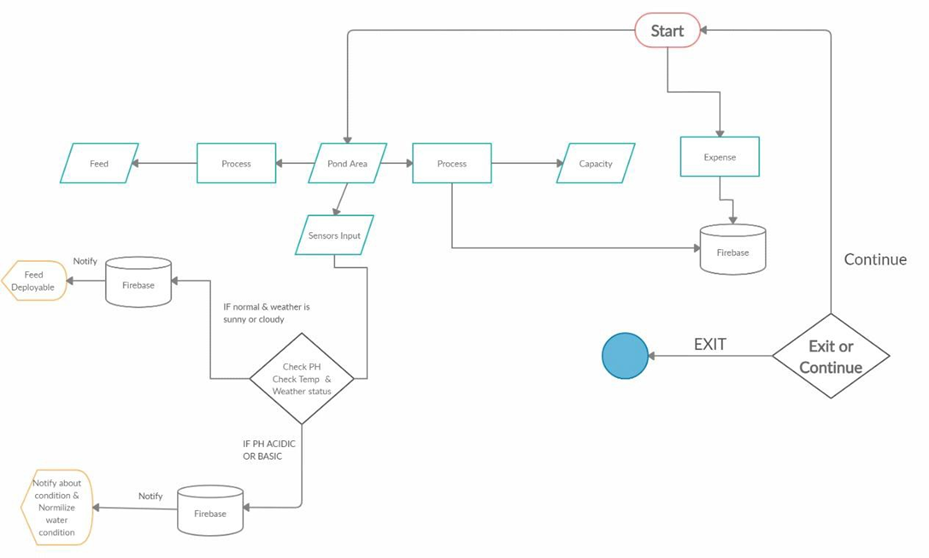


Figure 4.2 Process Flow

The figure demonstrates the data process of the overall system that how a user will operate our application. To do this he will first look for seed information by giving an area of a pound. The further user gives input to the system using a sensor then these sensors upload this data to our firebase and the firebase sent a notification to the farmer to generate respective output. Also, our firebase contains data about pond status including area, so it suggests feed to the farmer too. When the farmer is done using the application, he can exit the process and leave!

# Data Flow Diagram [Level Zero]

User

Figure 4.3 Level 0 Data Flow Diagram

In 0 level we have 2 entities “user” and “application”. Users can see multiple options like pond status, manage expenses & stokes, and weather prediction through our app user will also get to know about feed suggestions. In Figure 41, the system architecture diagram of the "Ultimate fishing guide" application is shown, represents that the user can check the value of ph. and can check the temperature. Based on these values and weather prediction user can deploy a suitable feed. Users will find the best marketplace. Availability of internet is necessary.

# Data Flow Diagram [Level 1]

The complete flow and functionality of each module have been shown in the figure below.

Diagram

Description automatically generated

Figure 4.4 Level 1 Data Flow Diagram

# Tree Structure for Firebase

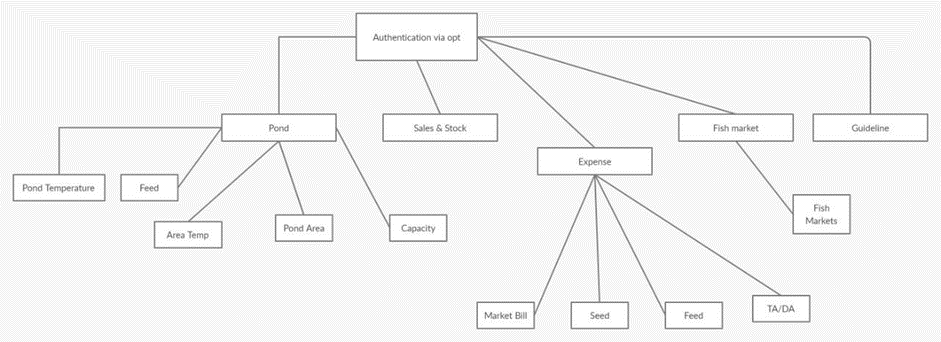
****

Figure 4.5 Firebase Tree Structure

This Diagram shows the full structure of our firebase nodes and their hierarchy. The root node is the authentication node under that all the application hierarchy is explained in this tree structure. As our pond’s node contains multiple child nodes all the graphical information about these child nodes is being explained in a diagrammatical manner. The expense node contains all the data about sales and expenditures in its child node.

Chapter 5

Implementation

In this chapter, we will discuss the techniques and tools we use to develop the project. This phase is the most prominent phase of development; as from this step, we transform our idea into a meaningful picture. This is a significant and challenging step toward learning or developing skills. An application is the result of the successful implementation of tools and technologies. Following are the tools and techniques we use during the development of our application

# Tools and Technologies

## Tools

* Android Studio 4.1 chipmunk 2021.2.1
* Arduino IDE
* MS Word
* MS PowerPoint
* MS Project

## Languages

* Dart

## Libraries &Framework

### Hardware

* ESP 8266
* Water level sensor
* Temperature sensor
* PH sensor
* Dissolved oxygen level sensor
* Jumper wires
* Bread Board
* LED

### Libraries

* + cupertino\_icons: ^1.0.2
  + font\_awesome\_flutter: ^10.1.0
  + firebase\_core: ^1.17.0
  + firebase\_auth: ^3.3.18
  + cloud\_firestore: ^3.1.16
  + fluttertoast: ^8.0.9
  + connectivity\_plus: ^2.3.0
  + internet\_connection\_checker: ^0.0.1+4
  + lottie: ^1.3.0
  + local\_auth: ^2.1.0
  + flutter\_local\_notifications: ^9.5.3+1
  + ESP8266WiFi.h
  + Firebase\_ESP\_Client.h
  + OneWire.h
  + DallasTemperature.h

# Development Stages

After the designing phase, we move towards the developmental phase to have a clear demonstration of the project. In the implementation phase, we will discuss the phases, which we face incrementally during the development of the application. We use the “Arduino” IDE for our project.

## Firebase Firestore Integration

Using Nodejs we created a function named firebase cloud function that has defined method for calibration of hardware and integration of weather API and other function with firebase. Calibration includes Change of PH, change in temperature, change in weather, change in time before and after changing the value of PH and Temperature. There is a Function name sendfollowernotification() that is responsible for alarming and sending the notification to the user. The notification includes Change in PH. If the Ph value is high or low, then it will suggest a solution. If there is a rise in temperature above a certain threshold it will notify the user to stop the increase. For feed and fertilizer function will check the weather condition if the weather is rainy, sunny, or cloudy according to rule it will notify the user what to do.

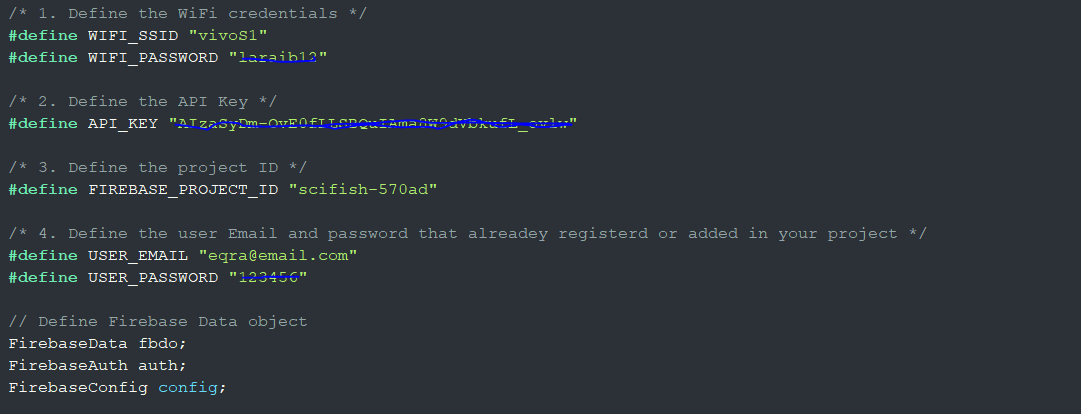


Figure 5.1 Firestore Connection

## Hardware Set

For the initialization of our ESP32 module for the pond status i.e. phone number that that farmer will register via OTP and location where the pond is situated. Initial values will be given to the function defined in firebase.



Figure 5.6 Firebase Node

After these values are initialized by the firebase the Esp32 will start working.it will get the node position and send all the data to our cloud function for the computational work. The user will be able to change the Node position from the android application. As it's not possible for the farmer to have different hardware for the different pond. For the reusability of our hardware anytime anywhere we have created a function in android that will get the correct value of pond name and location it will automatically overwrite the function of set hardware() in firebase and hardware will start working on the corresponding pond.

## PH Calculation

Using Arduino IDE, we created a C++ file that contains code for our hardware that is integrated inside our ESP32. We will be explaining how the Calculation of Ph is carried out first we will look out at our mathematical expression and then we will go through the code.

To calculate the pH value, we took an average of 10 samples and then store these values in an array. After adding these values we sort the array using the swapping technique in ascending order. Using loop we take out mid 6 values .adding them up and by taking an average of these values we applied conversion on the value and then the value is a return from the function.

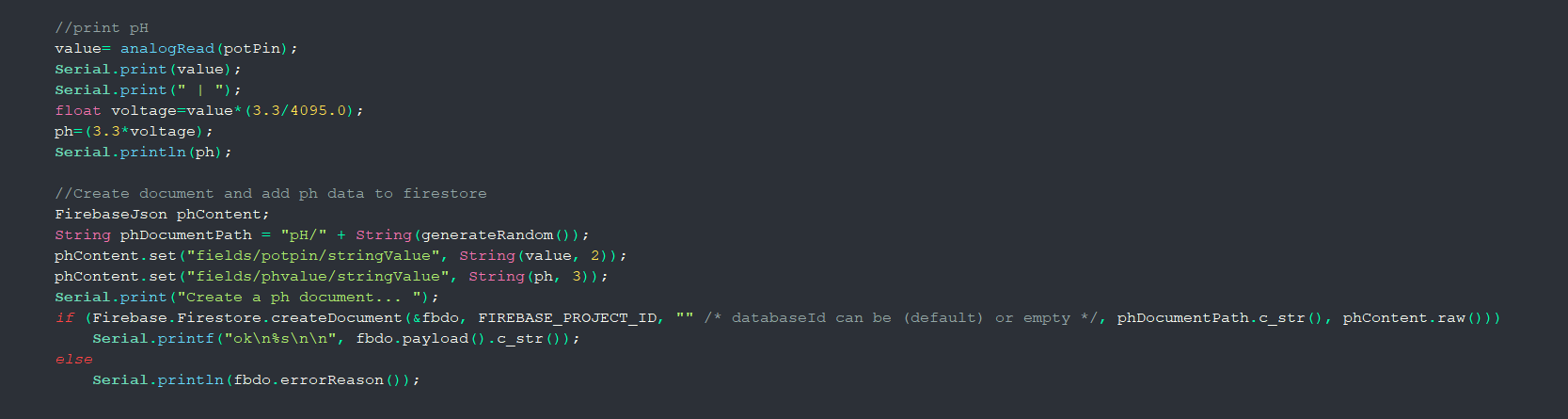


Figure 5.7 ISO CODE OF PH

## Pond Temperature

Our attached sensor has an embedded code for temperature calibration. Using the sensor library method name request temperatures() is simply called. This method initializes the sensor and sends a request for the temperature. After all this, a method from the sensor library is called the name getTempCByIndex(0) reads the temperature in Degree centigrade and returns the values to the main method.

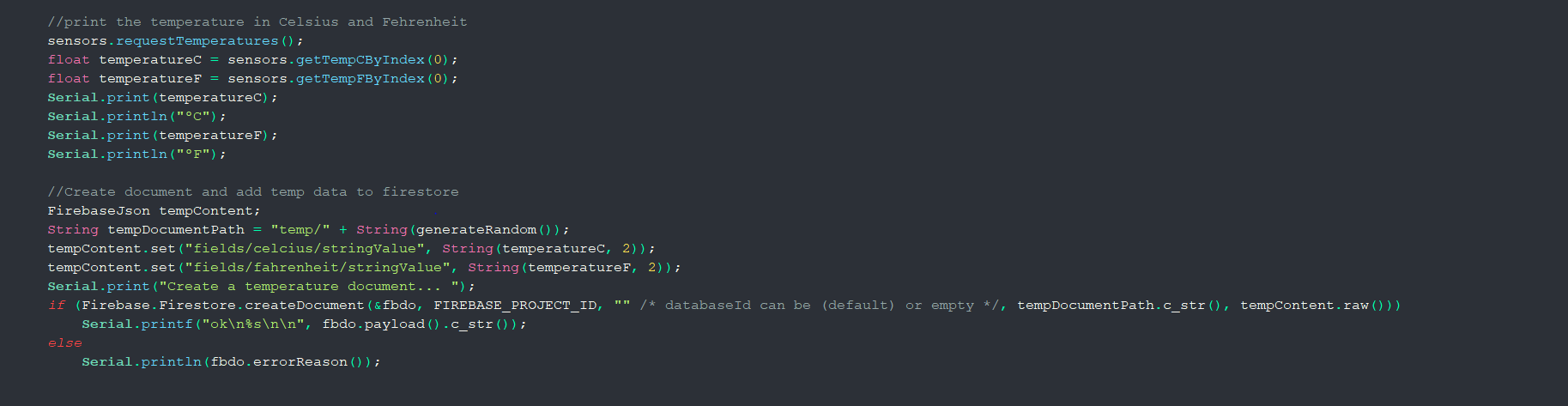


Figure 5.9 Pond Temperature Calculation

# User Interface

## Home

On the Home, screen the user can go through all of the major activities of the application.

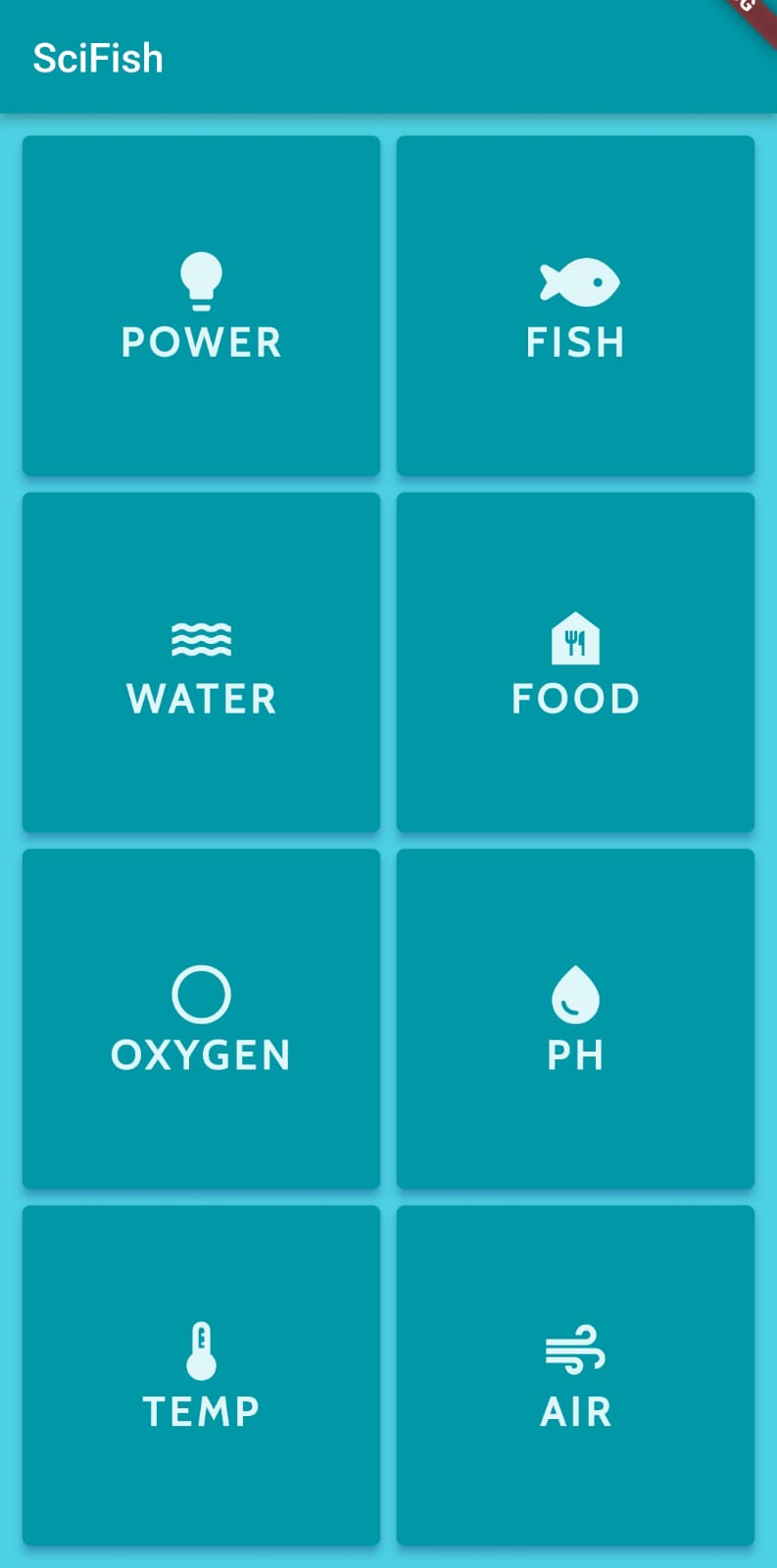


Figure 5.10 Home Screen

## Power Screen:



Figure 5.11 Config hardware

## Fish Life Screen:

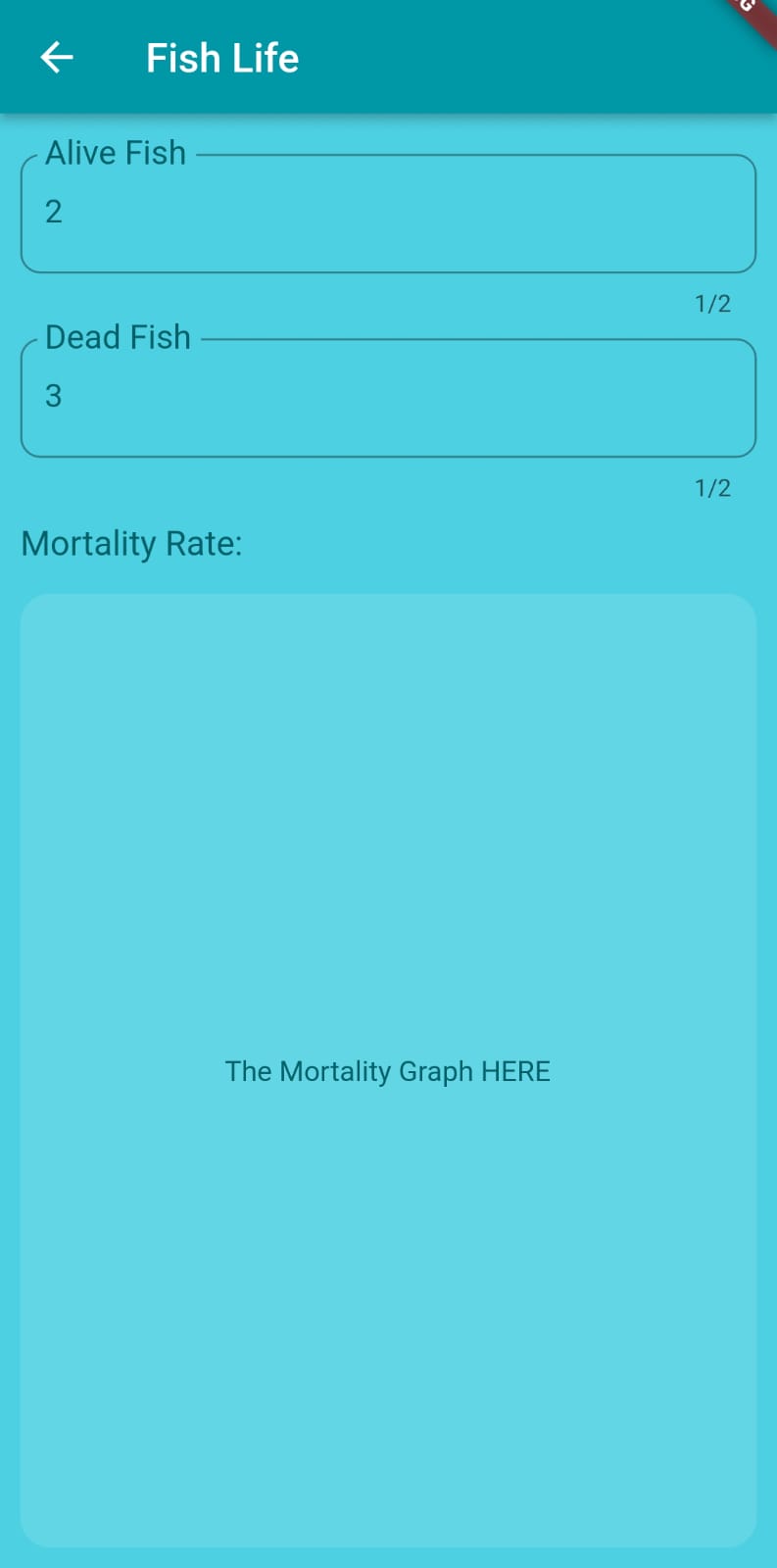


Figure 5.12 Fish Life screen

## Water Level Screen

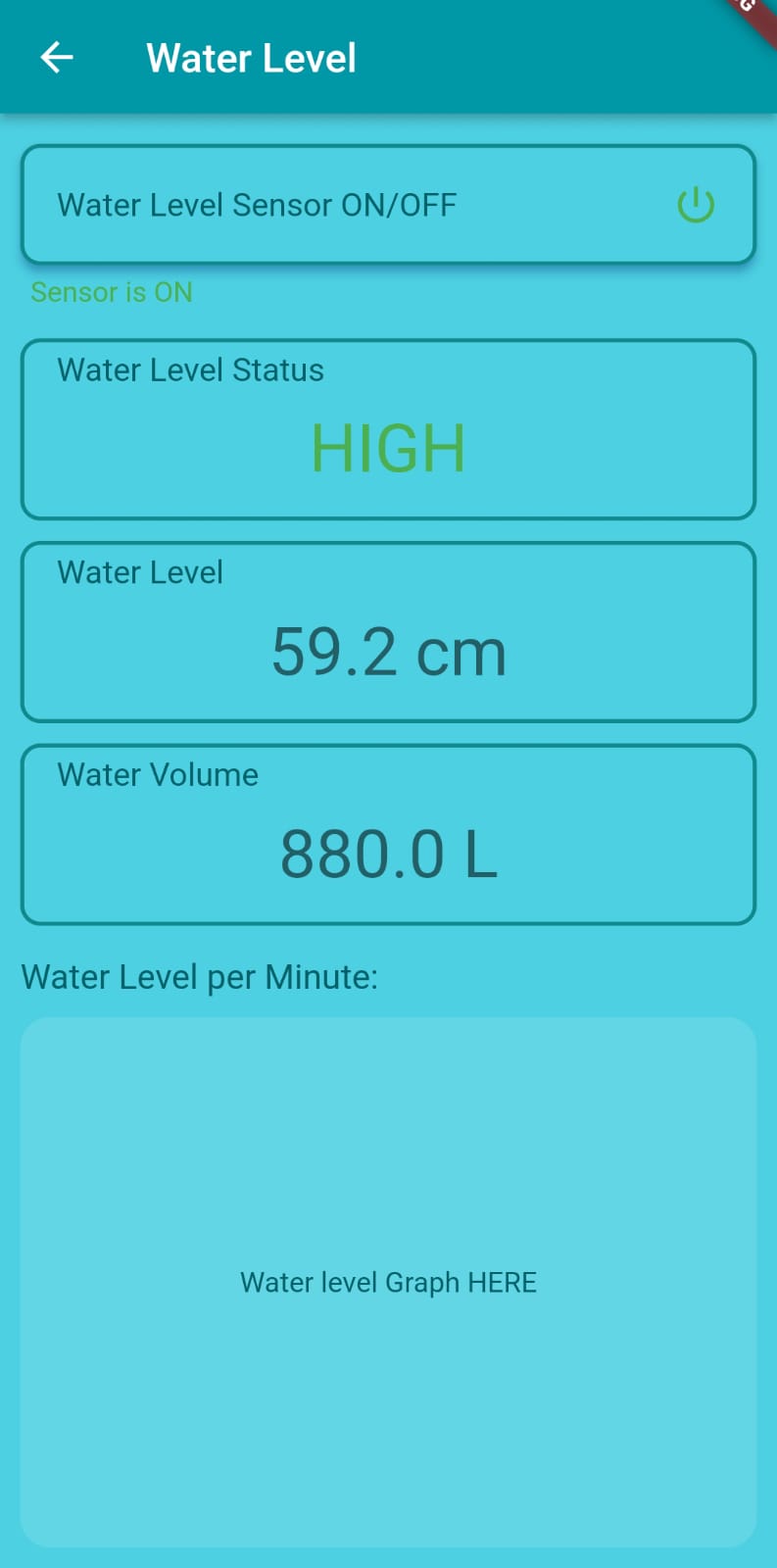


Figure 5.13 Water level Screen

## Food Screen



Figure 5.14 Food Screen

## Oxygen Level Screen

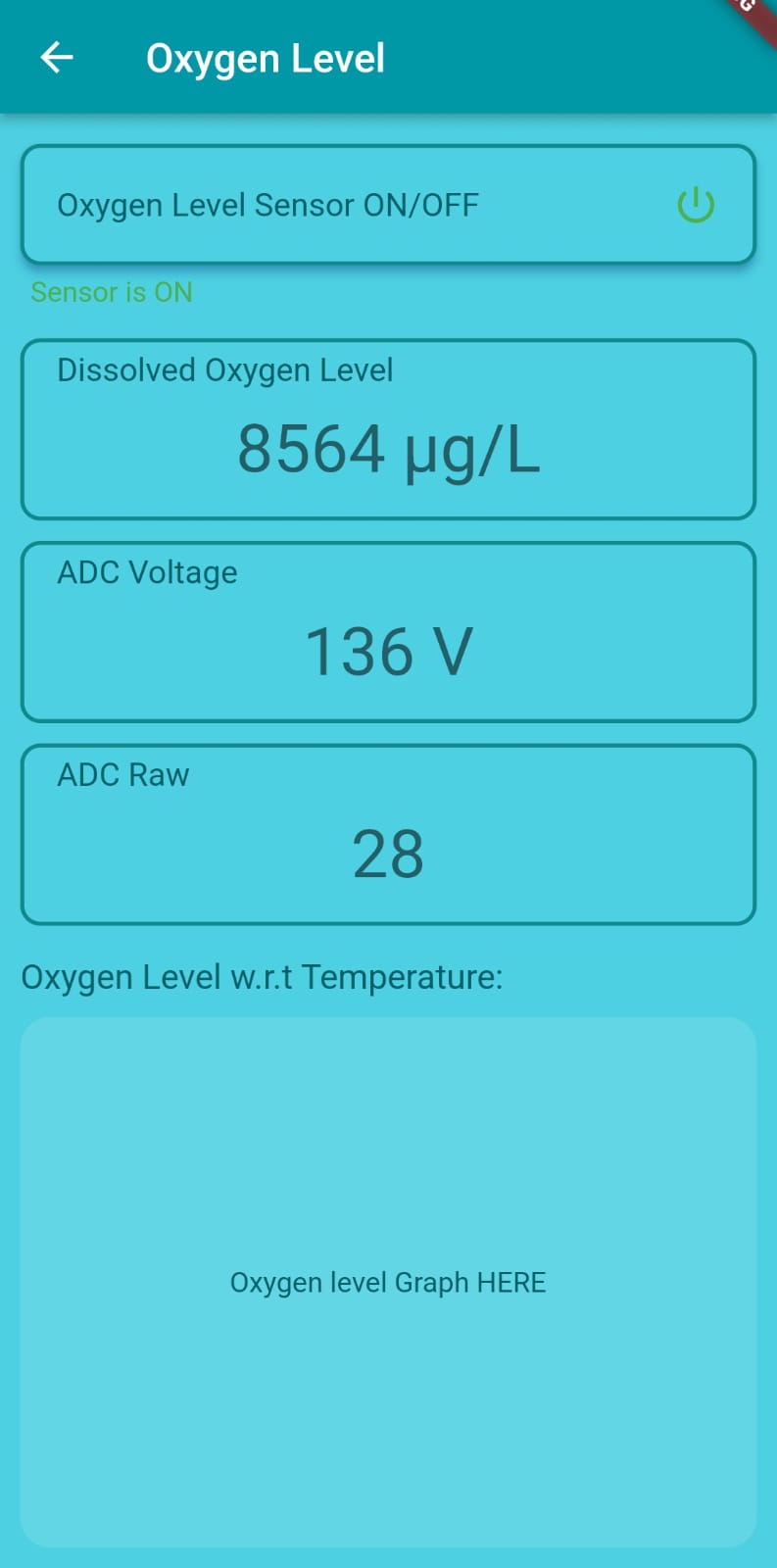


Figure 5.15 Oxygen Level Screen

## pH Screen

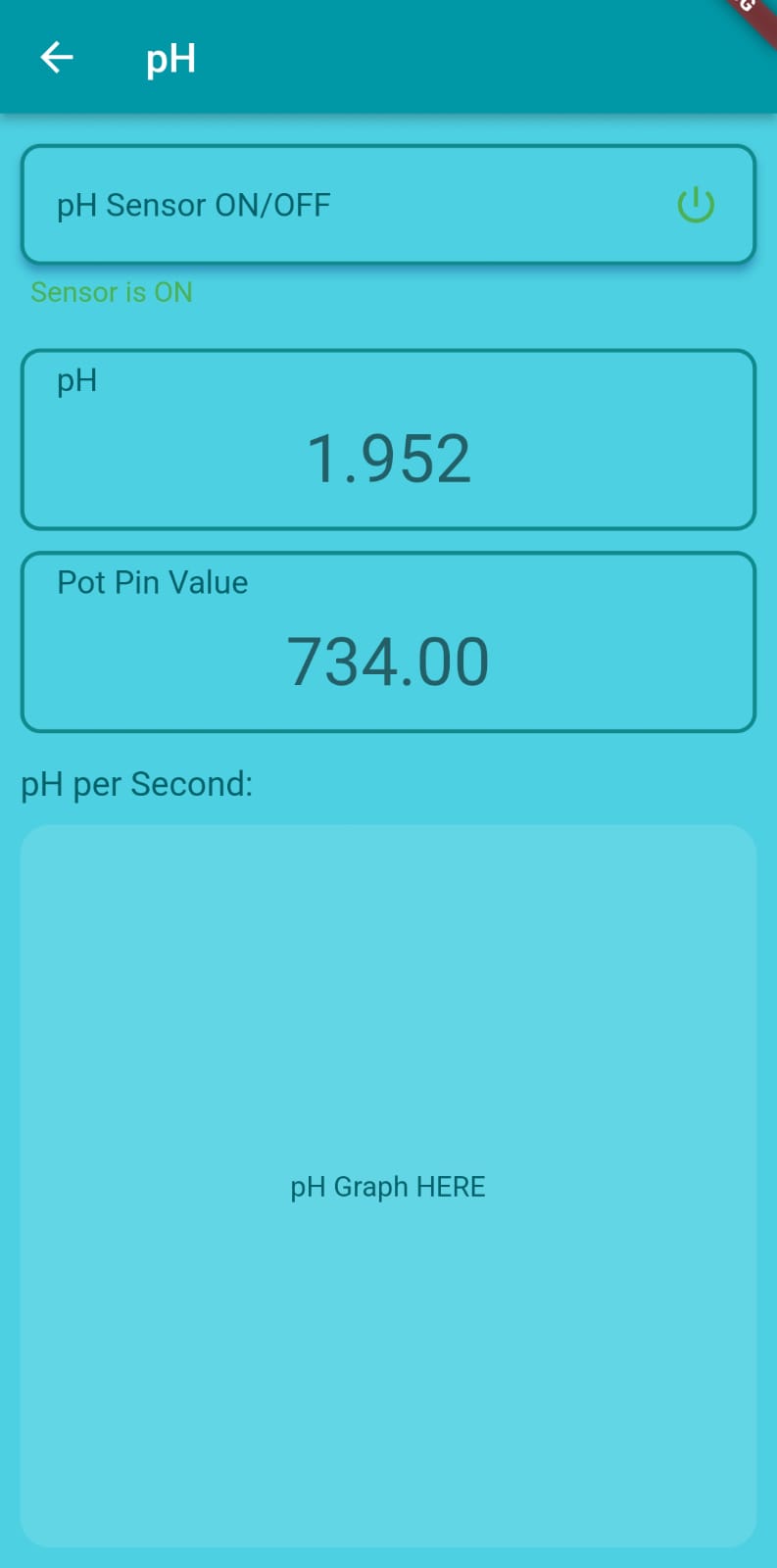


Figure 5.16 pH Screen

## Temperature Screen

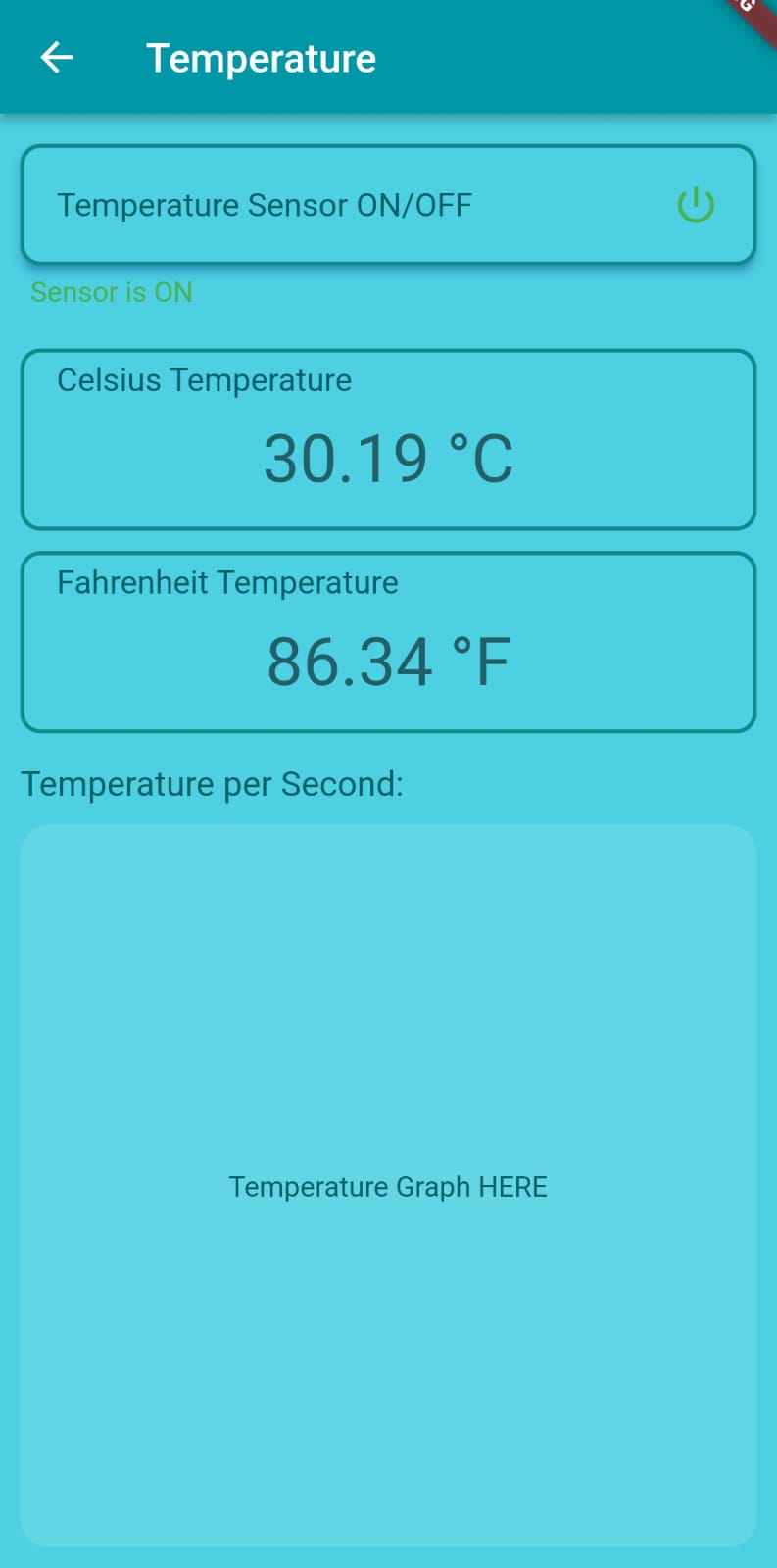


Figure 5.17 Temperature Screen

## Air Flow Screen

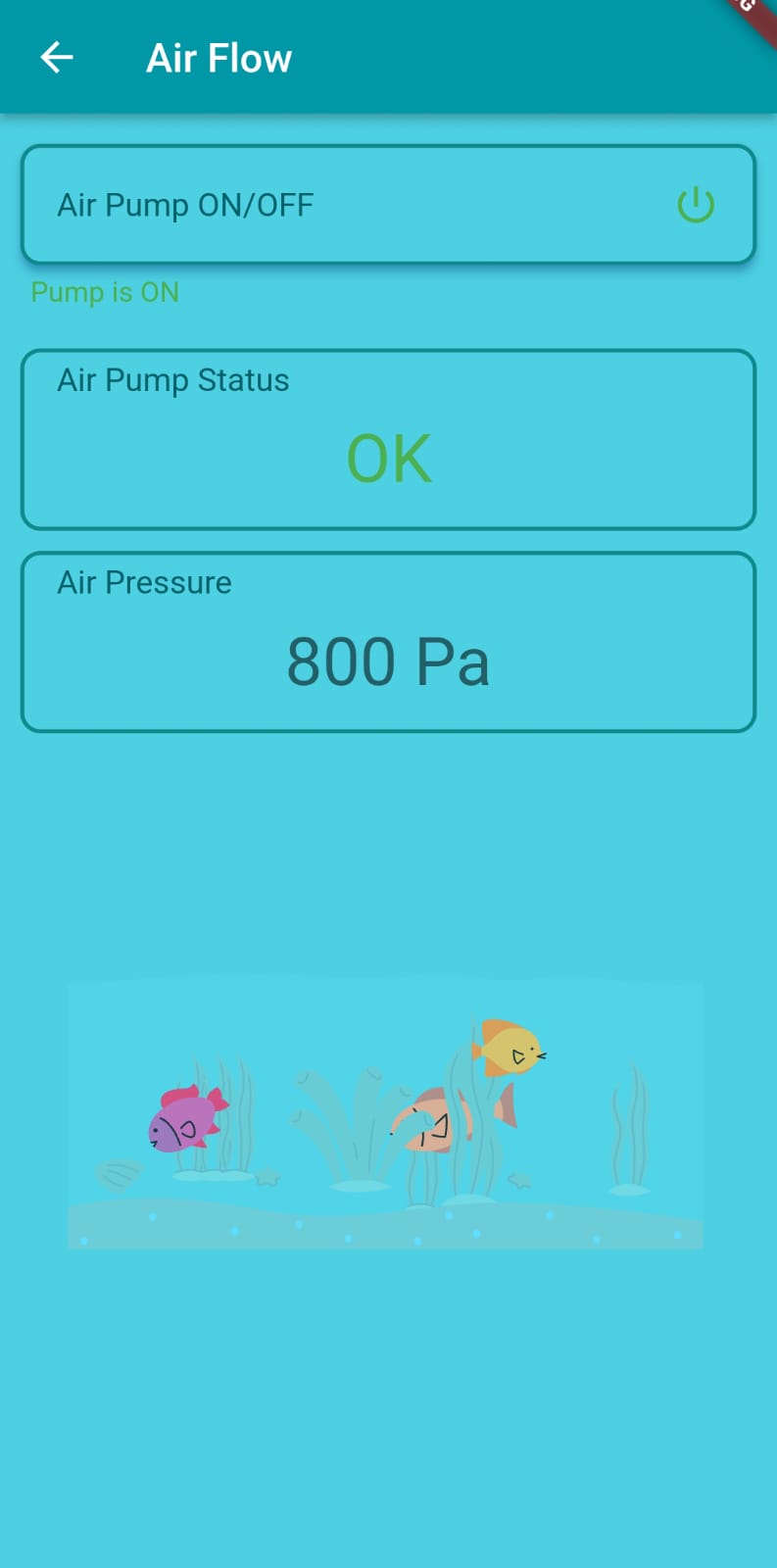


Figure 5.18 Air flow Screen

Chapter 6

Testing and Evaluation

# Manual Testing

Once we create any application we must check or analyze its features so that when it gets delivered, it is free of error, bugs, and other issues. The process of testing any software manually. To ensure completeness of testing. By following some written test plan that leads it to test cases. The test cases are done one by one by a person who is involved in the testing phase or testers. It is done in an automated way without using any tool. The main purpose of manual testing is to resolve or fix bugs and issues in software. It is the most primitive technique that helps to find bugs in the application. Every new application must be tested manually first. The two main reasons for ensuring the quality of software are given below:

* Finding out errors that arise during running the application.
* Check that the application fulfills the system requirement or not

## System testing

System testing validates the integrated software product. It takes an input of all integrated components. It is performed on the entire system.it tells the expectation of the customer. It tells us the actual result or behavior of the product when the system is tested. In our system, we will test that all hardware is configured correctly or not we check the value that comes from sensors and then finally we integrate it with our android application.

## Unit Testing

In unit testing, the smallest part of an application is tested independently. we check each unit of our system in accordance with user requirements. we check each piece of code to check the continuity and flow of an application. Every module has a relation with the previous and next steps. We can find errors by inspect code by click on analyze after the inspection result, we can see detailed errors and warnings messages. USB debugging mode helps to the copied the app via USB to the device for testing. We also faced several exception handlings during done this project. Unit testing for the whole application has been mentioned.

## Function Testing

In functional testing, we generate test cases to check the functionality of each module and its accuracy compared with the actual outcome. It is the stage where we analyze the accuracy of the module as correct or not. We not only tried to make every functionality performing but also had a focus on making it easy and attractive. We went through the complete process of testing. Starting with user authentication, log in through the app, and see all the results that happen in the app. These things are done by the admin. Admin is responsible for managing data firebase. They are performed to check the usefulness of the application from the user point of view.

## Test Cases Results

Test Cases and Results Test cases are created for each of the functional requirements. These test cases provide the action to test case; expected results and actual results along with test Fail/Pass remarks.

### User Sign-in

Objective: To ensure Sign in working correctly

Table 6.1 Sign-in

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Action** | **Expected results** | **Actual results** | **P/F** |
| **1** | Click on the Send code button without filled the phone  number | Highlight as required fields | Highlight as required fields | **P** |
| **2** | Enter an unregistered | Highlight as entering a phone number | Highlight as entering a phone number | **P** |
| **3** | Enter the wrong phone number | Highlight as a show error message | Highlight as a show error message | **P** |
| **4** | Enter the registered phone  number | Successfully login | Successfully login | **P** |

### User Sign-up

Objective: To ensure Sign-up working correctly

Table 6.2 User Sign-up

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Action** | **Expected results** | **Actual results** | **P/F** |
| **1** | Click send code with empty fields. | Highlight as required fields | Highlight as required fields | **P** |
| **2** | Enter phone no less than 11 digits | Highlight as required fields. | Highlight as a required field. | **P** |
| **3** | Enter the wrong phone number | Highlight as entering a correct phone number | Highlight as the wrong input. | **P** |
| **4** | Enter wrong verification code | Highlight as error | Showed error message | **P** |
| **5** | Enter all the entries as required | Registered phone number | Registered | **P** |

### Configure hardware

Objective: To ensure integrated hardware with specific pond working correctly

Table 6.3 Configure hardware

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Action** | **Expected results** | **Actual results** | **P/F** |
| **1** | Click on right 3 dots click ‘‘config hardware’’ | Highlight as next activity | Highlighted as the next activity. | **P** |
| **2** | Phone no | Highlight as add successfully | Highlighted as add successfully | **P** |
| **3** | Enter pond name | Highlight as add successfully | Highlighted as add successfully | **P** |
| **4** | Enter city name | Highlight as add successfully | Highlighted as add successfully | **P** |
| **5** | Configured successfully | Highlight as configured successfully | Highlighted as configured successfully | **P** |

### Stock

Objective: To ensure data is added or removed successfully

Table 6.4 Stock

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Action** | **Expected results** | **Actual results** | **P/F** |
| **1** | Click on ‘+’ fertilizers | Add amount | Add amount | **P** |
| **2** | Click on ‘+’ powder | Add amount | Add amount | **P** |
| **3** | Click on ‘+’ synthetic feed | Add amount | Add amount | **P** |
| **4** | Click on ‘-’ fertilizers | Amount removes | Amount removes | **P** |
| **5** | Click on ‘-’ powder | Amount removes | Amount removes | **P** |
| **6** | Click on ‘-’ synthetic feed | Amount removes | Amount removes | **P** |

### Logout

Objective: To ensure user log-out working correctly

Table 6.5 Log-out

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | | **Action** | | **Expected results** | | **Actual results** | | **P/F** | |
| **1** | | Click logout option | | Log out | | Logged out | | **P** | |
| **No** | | **Action** | | **Expected results** | | **Actual results** | | **P/F** | |
| **1** | | Click logout option | | Log out | | Logged out | | **P** | |

# Integration Testing

Table 6.6 Integration testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Test case** | **Expected result** | **Actual result** | **P/F** |
| **1** | Integrate Ph sensor | Send ph. value to firebase | Value send successfully | **P** |
| **2** | Temperature sensor | Send temp value to firebase | Value send successfully | **P** |
| **3** | Firebase integration with android | Send sensor values and weather forecast | Send sensors value and weather forecast | **P** |
| **4** | Jason file integration with firebase | Capture JSON and get desired data | Capture JSON and get desired data | **P** |
| **5** | Send a notification from hardware to android using firebase median | Pop-up notification on the screen | Successfully pop- up notification on screen when ph. value increases or decreases | **P** |

Chapter 7

Conclusion and Future Work

# Conclusion

Due to the fast changes in the era of computerized technology we need to convert manual systems to automated systems. Fish farming gives Pakistan the economy speed by the exportation of fishes. Our system going to boost fish farming by giving them handy techniques and procedures to fasten the growth of fishes by keeping track of each activity needed for fish farming.

There is no such app that guides the farmer using such a technique we used IOT based system to automate farmers work of keeping track of water quality and weather conditions.

Several fishes die each year because the farmer is not aware of the situation inside the ponds. This app will surely awake and keep the farmer updated about the feasibility of the pond and give the most probable solution for the cure.

Another issue related to farming was kept in view and we give solutions to them too i.e. managing sales, stock, expenses, fish market suggestions, and keeping data of all ponds.

# Future work

The following can be implemented as future Work.

1. Disease Control
2. Sensors Such as (Opacity sensor, DO sensor)
3. Fish growth Progress
4. Fish Type identification
5. Healthy and unhealthy Fish classification

Chapter 8

References

# References

1: International Journal of Fisheries and Aquatic studies

An economic analysis of the fisheries sector of Pakistan (1950-2017): Syed Babar Hussain Shah, Yongtong Mu



2: Department of Fisheries Punjab



3: Vaishnavi V. Daigavane and Dr. M.A Gaikwad: Water Quality Monitoring System Based on IoT









6

githubb for HTTPClient library

7 Arduino integeration with Esp32

