Software Requirement Specification Document for IFish Farm: Monitoring and analysis of fish anomaly behavior in ubiquitous environment

Ali Ahmed, Hussam ElDin, Omar Anas, Youssef Mohamed

Supervised by: Dr. Ayman Ezzat, Dr. Ayman Nabil and Eng. Noha ElMasry

February 21, 2020

1 Introduction

1.1 Purpose of this document

The purpose of this document is to provide a full description of how the iFish Farm monitoring system works. The monitoring system is an online web-based application system to monitor the fish ponds of a farm to ease the maintenance process for farmers to reduce the labor. This software design document (SDD) will describe the aim of the system and its functionalities. In addition, the document will show all constraints on the system, all interfaces designs and all diagrams that were needed to build the system.

1.2 Scope of this document

The purpose of the IFish Farm system is to ease the monitoring process and to create a convenient and easy-to-use web application for farmers to monitor their fish farms. The system is based on an unsupervised learning methodology to cluster the different fish behaviors occurring in the farm while providing necessary alerts to the farmers based on the events detected and suggesting a solution to the problem of the event. We will have a cloud server that will process the captured footage to be done there to speed up the monitoring process.

1.3 Overview

This document describes most of the system diagrams and architectures. It also previews how the system main functionalities work and how the user views and interacts with the software. The sections in this document gives a detailed

description for the diagrams that help the developer developing the system. It includes the class diagrams, sequence diagrams and the 4 architectures diagrams.

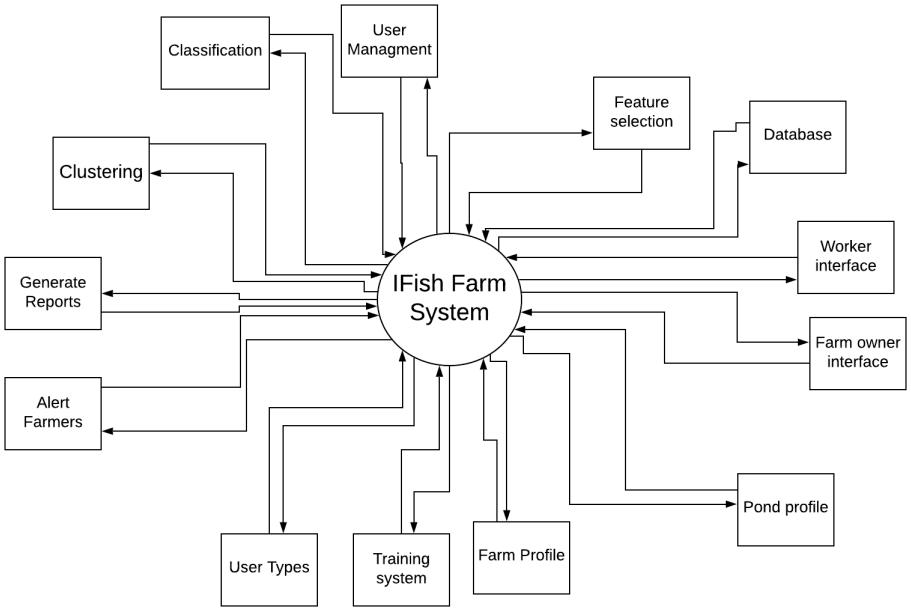

Figure 1: Context Diagram

Firstly,two cameras, a camera for detecting toxic ammonia and a camera for detecting anomalies in the pond,They will be connected to a mobile device/ arduino device. The device will send the raw data to the cloud storage. The cloud storage then sends the data to be pre-processed. In the pre-processing, the image will be enhanced to remove any water turbidity. Our code will be written and converted to an API so we can use it on the device to provide notification of pond status. After the API gets the image and video it will be divided into two sections. The first section is for processing and enhancing the colors in the video footage. Firstly, the video will be analyzed then the behavior will be detected after that the behavior will be clustered as normal or abnormal also speed, abnormal size .. etc will be detected. The second section is for processing the image of the paper to detect toxic level of ammonia. The color will be detected and classified as toxic or non toxic ammonia level in water. All of the results of the processed data will be sent to a database. Our application then retrieves this data and shows it to the farmer on the webapp or notify the farmer on his device about any abnormal events that happen in the pond.

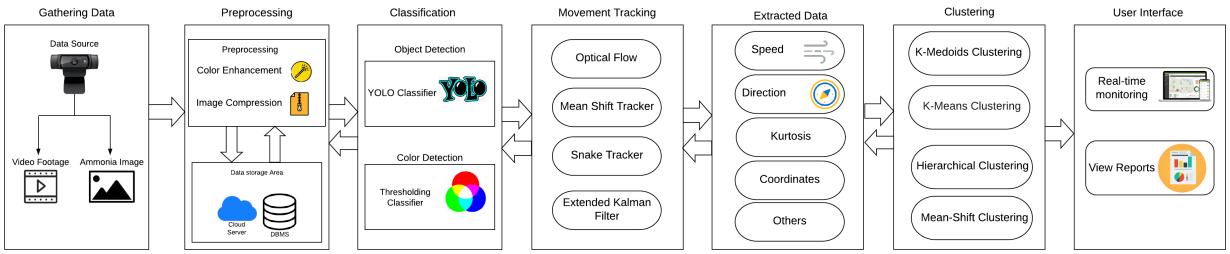

Figure 2: Block Diagram

1.4 Business Context

Since the country is putting a lot of focus on fish farms in the mean time for economical growth, Fish farms suffer from many problems as farmers have to check the water quality frequently and ammonia levels in water, they need continuous surveillance over the fish ponds to check if the fish are dying to know that the pond is having a problem and then they have to discover the cause, while keeping track of fish size so they can be moved to other ponds. so a system is needed to monitor fish behavior, health, size, count, feeding times, and check the water quality. According to Shaalan et al. [1] the aquaculture business in Egypt provides about 77% of national fish production and provides about 580,000 jobs for workers in this part. In addition, The estimate for aquaculture fish production exceeds USD 2 billion annually, as shown in figure 1.

Fish species	Production value (USD 1000 /year)
Nile tilapia	1,039,056
Carps	449,150
Mullets	303,484
Gilthead seabream	90,558
European seabass	50,731
Penaeus shrimp	48,103
Catfish	22,933
Meagre	19,539
Total	2,023,554

Figure 3: Fish production value in Egypt

2 General Description

2.1 Product Functions

"IFish Farm" is a system who's main function is to offer a monitoring system for fish farms to detect any anomaly in the fish ponds and fish behavior also

detect any change in the ammonia level in water to alert the workers and ease there jobs

2.2 User Characteristics

In this document, we proposed a system that deals with Fish farms issues, therefore, the system is user friendly enough that any user is able to use it and it can be installed in most conditions.

2.3 User Problem Statement

The main problems fish farmers suffer from are the fast and unpredictable change in ammonia level in the water which cause instant death of fish in the fish farm ponds and the need for continues surveillance on the fish ponds for detecting any abnormal behavior in the ponds.

2.4 User Objectives

The solution is designed specifically for fish farmers who suffer from difficulties in monitoring their fish farms. The main objective is to be able to detect any anomalies in the fish ponds to prevent any problems before the death of fish. In addition, the system will be able to generate reports and statistics to monitor the farms' productivity rates and profits.

2.5 General Constraints

The system needs to be as fast as possible to be able to detect any abnormal behavior in the fish pond at the moment it happens, also setting up the cameras at the appropriate position to get the whole fish ponds is difficult process in the fish farms environment. In addition, the continuous availability of large database and high speed internet would be challenging for the system.

3 Functional Requirements

3.1 FR1

Title	Show Pond
Description	When the user opens the system it starts recording the video footage in the fish pond. It takes the object of type camera
Action	Checks if camera is opened. If it's opened the camera will start recording. Else it will open the camera to collects video footage to be processed.
Input	None.
Output	video footage for fish pond.
Precondition	User Starts the system
Post-condition	None.
Dependencies	FR9
Priority	10/10

3.2 FR2

Title	Enhance Video
Description	The function will enhance the water colors in order to make fish more visible. It takes video frames to apply on it the MSR algorithm to enhance each video frame.
Action	Checks if the frame is not corrupted or null. If not null or corrupted the function will start enhancing the frame.
Input	Pond Video Footage
Output	Color Enhanced video footage
Precondition	Cameras have captured the video footage of the pond
Post-condition	None.
Dependencies	FR1
Priority	10/10

3.3 FR3

Title	Enhance Image
Description	The function will enhance the water colors in order to make ammonia paper more visible. It takes the image to apply on it the MSR algorithm to be enhanced.
Action	Checks if the image is not corrupted or null. If not null or corrupted the function will start enhancing the image.
Input	Unclear Image.
Output	Color Enhanced Image.
Precondition	Cameras have captured the Image of the ammonia paper
Post-condition	None.
Dependencies	FR1
Priority	10/10

3.4 FR4

Title	Detect fish
Description	This function detects the fish in the enhanced video. It takes the enhanced video as a parameter and returns an array with the detected objects' coordinates. The detection of fish is done using YOLO algorithm.
Action	It should draw only the bounding box around the detected fish if the confidence value is higher or equal the desired number.
Input	Enhanced Pond Video Footage
Output	Array of coordinates of detected objects.
Precondition	The video footage should be enhanced
Post-condition	None.
Dependencies	FR2
Priority	10/10

3.5 FR5

Title	Capture Image
Description	This function connects to the web camera and captures an image of the ammonia paper through it. The captured image is captured periodically and should be enhanced through the enhancement function and then sent to the color detection to detect toxic ammonia levels.
Action	It should check if the image size is appropriate to use for enhancing and if the image is not null.
Input	None.
Output	Captured Image.
Precondition	None.
Post-condition	The image should be enhanced.
Dependencies	None.
Priority	10/10

3.6 FR6

Title	Clustering
Description	This function takes 2D array of features that contains the selected features needed so we can apply clustering algorithms to split data to different clusters according to fish behaviors.
Action	It takes an 2d array of features to cluster behaviors. The array of features should not be null. If null, the function will break and return nothing. Else, it will apply k-medoids to cluster the data.
Input	2D array of features.
Output	String with the Nearest Cluster.
Precondition	Features should be extracted and put into 2D array .
Post-condition	Alert for abnormal behavior.
Dependencies	FR17
Priority	10/10

3.7 FR7

Title	Detect colors
Description	The function detects different ammonia level in water by checking the area where the test paper is placed and extracting the color from it where each color indicates the ammonia level.
Action	Checks on the four different colors of the ammonia alert paper. If Blue, then the water is toxic. Else, the water is clear from ammonia.
Input	Image of the ammonia paper.
Output	String WaterStatus
Precondition	The image should be enhanced.
Post-condition	None.
Dependencies	FR3
Priority	10/10

3.8 FR8

Title	Register
Description	Allows user to make a new account. It takes First Name , Last Name , Email and password of the user.
Action	Checks if the user email does not exist in the database. Also, Checks if any of the data entered is not empty or null. If any of them is empty it alerts the user. Else, the account is made.
Input	FirstName , LastName, Email, Password
Output	Boolean true or false
Precondition	None.
Post-condition	A new account is added to the database.
Dependencies	None.
Priority	10/10

3.9 FR9

Title	Login
Description	User can not use the system without logging in first. It takes the users' Email and password.
Action	Checks if the user email and password exists in the database. If exists, then the user is logged in. Else, the user is alerted by invalid email or password.
Input	Email and Password of the user
Output	Boolean true or false
Precondition	The user must have an account.
Post-condition	Redirect to homepage.
Dependencies	FR8
Priority	10/10

3.10 FR10

Title	Logout
Description	Logs out the user from the system. It stops the video footage and redirect user to login page.
Action	Checks if the user is currently logged in to make him logout.
Input	None.
Output	None.
Precondition	The user must be logged in.
Post-condition	Redirect to login page.
Dependencies	FR9
Priority	10/10

3.11 FR11

Title	User CRUD
Description	Fish Farm Owner has the ability to add,edit,delete and show all users in that are available on the system. Takes the new/edited data with all required information and insert/update it into the database. Deleted users are marked as deleted in the database.
Action	Checks if the edited/new data is valid and not empty. Checks if the deleted data exists in the database.
Input	User first name, last name, email and password
Output	Boolean true or false.
Precondition	Fish Farmer must be logged in with an account that have permissions to manipulate users data.
Post-condition	New user or edited data are added to the database. Deleted user(s) are marked as deleted in the database.
Dependencies	FR9
Priority	6/10

3.12 FR12

Title	Name behavior
Description	User can name an unknown abnormal behavior that is detected by the system. The name will added in the database and will be sent to the fish farm owner to be approved/declined.
Action	Checks for duplicate names in the database. If the name exists, then notify the user with name already exists. Also, checks if the name is null or empty.
Input	String Behavior Name
Output	None.
Precondition	The name of the behavior was unknown
Post-condition	The name is sent for approval
Dependencies	FR1, FR2, FR4,FR6
Priority	5/10

3.13 FR13

Title	Approve naming
Description	Fish Farm owner should be able to approve/decline on the naming of the unknown abnormal behavior that is detected by the system and named by other users
Action	An appropriate name for that behavior will be added to the newly detected cluster in the dataset if approved. If declined, the naming will be removed from database.
Input	None.
Output	Boolean.
Precondition	The name of the behavior was on stall.
Post-condition	The Name is either approved and assigned to that behavior or declined and deleted from the database.
Dependencies	FR12
Priority	5/10

3.14 FR14

Title	Alert users
Description	This function first checks if any variables reflecting the behavior or water status are changed during the system execution and notifies all users on the system when any abnormal event happens in the farm
Action	function receives variable for the water toxicity and current fish behavior Check if the water status and current behavior variables are changed to alarming values and then send a notification to the users.
Input	water toxicy level and current fish behavior.
Output	None.
Precondition	None.
Post-condition	Send a notification to the users.
Dependencies	FR1, FR2, FR4, FR6, FR7
Priority	10/10

3.15 FR15

Title	Rate Alerted Behavior
Description	This function allows the farmer to rate the behavior classification based on its accuracy after checking it in order to get feedback on the system.
Action	The function is called after notifying the user with abnormal behavior. It also, allows him to put comments.
Input	Integer rating from 0 to 5
Output	None.
Precondition	abnormal behavior detected.
Post-condition	New rating is inserted to the database and is used for further improvements in the system.
Dependencies	FR14
Priority	5/10

3.16 FR16

Title	Solve Problem
Description	This function starts after an abnormal event is detected and outlines steps to help the farmer deal with the problem in the pond.
Action	The function retrieves steps for the solution of the detected abnormal behavior if exists in the database.
Input	None.
Output	String containing solution steps
Precondition	Detection of Abnormal Behavior
Post-condition	None.
Dependencies	FR14
Priority	5/10

3.17 FR17

Title	Get features
Description	This function retrieves the speed, position, and direction of objects detected during the yolo model execution and adds those features into a 2-dimensional array to be used for clustering the behaviours.
Action	It checks the box coordinates each given number of frames and calculates the speed and direction based on the values extracted.
Input	Two dimensional array of position X,Y, Integer number of frames to be used in each iteration
Output	Two dimensional array containing speed and direction
Precondition	enhanced video footage.
Post-condition	Cluster features.
Dependencies	FR1, FR2, FR4
Priority	10/10

3.18 FR18

Title	Upload footage
Description	this function uploads recorded footage to the cloud to be enhanced and processed to extract the needed data from them.
Action	It will be taking an array of frames of pond and upload it to the cloud and it will return true or false if its uploaded or not
Input	array of frames
Output	Boolean
Precondition	video footage.
Post-condition	none.
Dependencies	FR1
Priority	10/10

3.19 FR19

Title	Generate Reports
Description	This function generates reports for all the behaviors and status the fish pond went though to be checked by the farm owners .
Action	It will automatically generate a report for fish behaviors , count , any problem the fish pond and regarding the farms profit and productivity rates.
Input	None.
Output	Reports PDF.
Precondition	None.
Post-condition	None.
Dependencies	None.
Priority	6/10

3.20 FR20

Title	Compress Frame
Description	This function takes a frame as input from the input video or livestream and applies a compression algorithm to reduce the size of the frame for faster image processing.
Action	It compresses the given frame losslessly and outputs the same frame with a lower size.
Input	An image (frame) file
Output	An image file (lower in size).
Precondition	None.
Post-condition	None.
Dependencies	FR1.
Priority	7/10

3.21 FR21

Title	Encrypt
Description	This function takes a user password as input and produces an encrypted string to be stored in the database.
Action	It converts a normal string into an encrypted string.
Input	String (password)
Output	String (encrypted password).
Precondition	User should register.
Post-condition	None.
Dependencies	FR8.
Priority	9/10

3.22 FR22

Title	Decrypt
Description	This function takes an reads all the encrypted passwords in the database and check if any of them matches the user input password after it is encrypted using the same algorithms.
Action	Checks if the input password matches any encrypted record in the database.
Input	String (password)
Output	Boolean (if password exists in the database or not).
Precondition	User Should Login.
Post-condition	None.
Dependencies	FR9
Priority	9/10

3.23 FR23

Title	View pond profile
Description	It shows the details of the pond by taking such as fish count, current fish behavior and others. It takes the pond ID as a parameter.
Action	It checks if the pond id exists in the database or if it is not equal null.
Input	Integer pond ID
Output	Array of String containing all the details.
Precondition	None.
Post-condition	Shows all the details of the pond.
Dependencies	FR1
Priority	7/10

3.24 FR24

Title	Add Pond
Description	This function adds new pond in the database. It takes pond name, fish count, date created.
Action	Checks if the pond name doesn't exist in the database. Also, if the entered data is not null or empty.
Input	Pond name, fish count, date created.
Output	Boolean true or false.
Precondition	Fish Farmer must be logged in with an account that have permissions to add pond data.
Post-condition	New pond(s) are added to the database.
Dependencies	None.
Priority	6/10

3.25 FR25

Title	List Ponds
Description	This function lists all the ponds that a farmer have in his pond even if its not monitored by the system
Action	Checks if the edited/new data is valid and not empty. Checks if the deleted data exists in the database.
Input	Pond Data.
Output	Boolean true or false.
Precondition	Fish Farmer must be logged in with an account that have permissions to manipulate pond data.
Post-condition	New pond or edited data are added to the database. Deleted pond(s) are marked as deleted in the database.
Dependencies	None.
Priority	6/10

4 Interface Requirements

4.1 User Interfaces

4.1.1 GUI

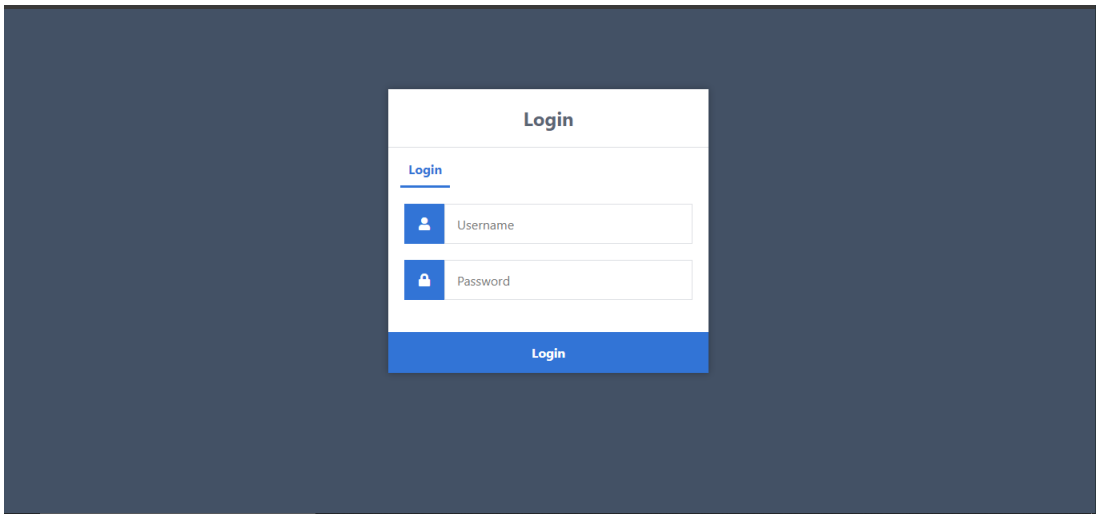

Figure 4: Login Screen

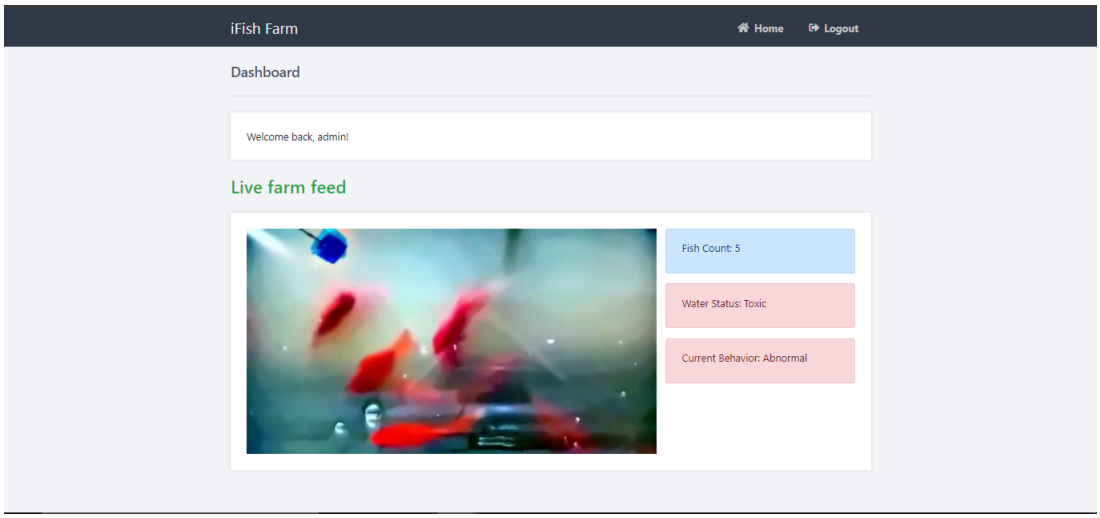

Figure 5: Homepage Screen

4.1.2 API

- openCV
- Flusk

5 Performance Requirements

For the IFish Farm, the system that shall be able to process and enhance frames. Also, The system must be able to handle training datasets for model creation in order to ensure model precision.

6 Design Constraints

The availability of internet all the time for continuous monitoring.

7 Other non-functional attributes

7.1 Reliability

- Speed is an important feature in the system as it should provide fish farmers with real-time notifications to notify them on any anomalies in the ponds.

- Accuracy should be nearly 90% in classifying and detecting the type of each anomaly in the fish pond to provide fish farmers with trustworthy feedback and notifications.

7.2 Maintainability

- This feature is applied by implementing the Model-View-Controller MVC design pattern and other design patterns which make the system more flexible to be improved or fixed.
- Implementing the Entity-Attribute-Value EAV that allows the developer to add any new requirements dynamically.
- The system can be improved by engaging the user in some actions which can improve the accuracy of the system.

7.3 Portability

- This feature is applied by implementing a responsive website that allows any user to use the system on any web browser from any device.

7.4 Usability

- Nielsens heuristics will be applied to ensure a simple and easy interface. A usability study is to be conducted with the client to get feedback and improve the usability.
- The system will be easy to use and learned as fish farmers usually are not familiar with modern technologies.
- The system will be memorized easily as there won't be much tasks for the fish farmer to do.

8 Preliminary Object-Oriented Domain Analysis

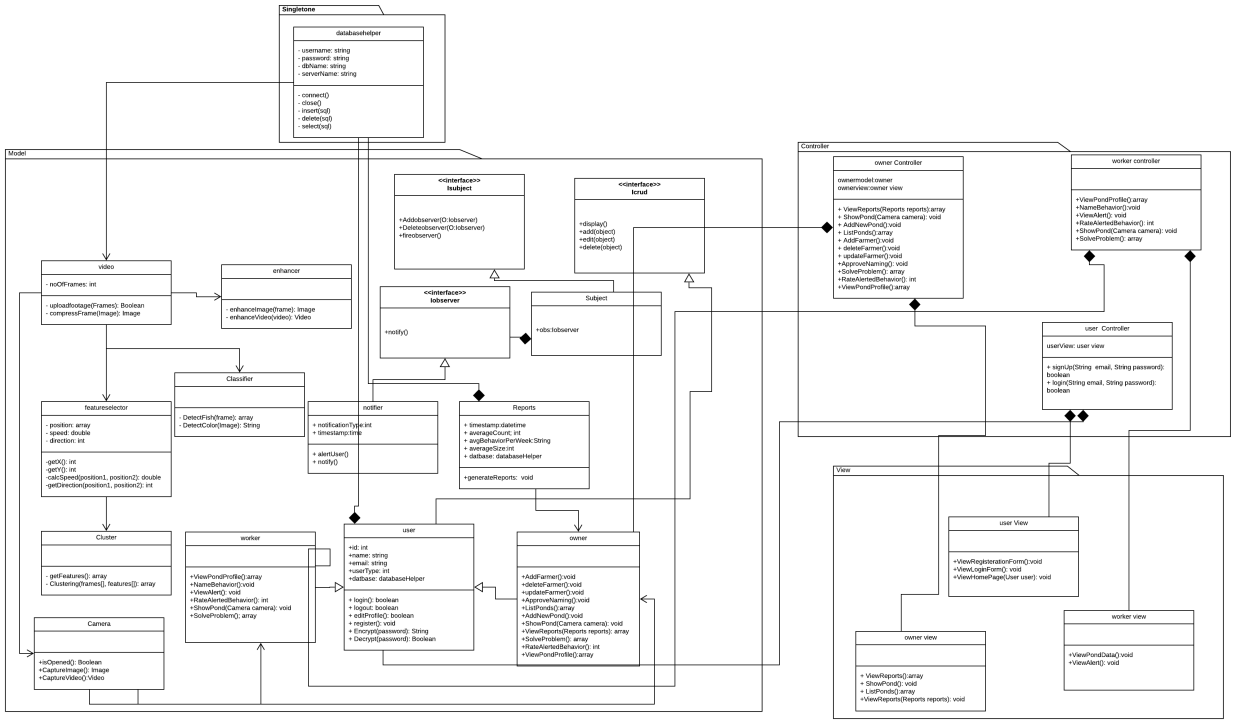

Figure 6: Class Diagram

9 Operational Scenarios

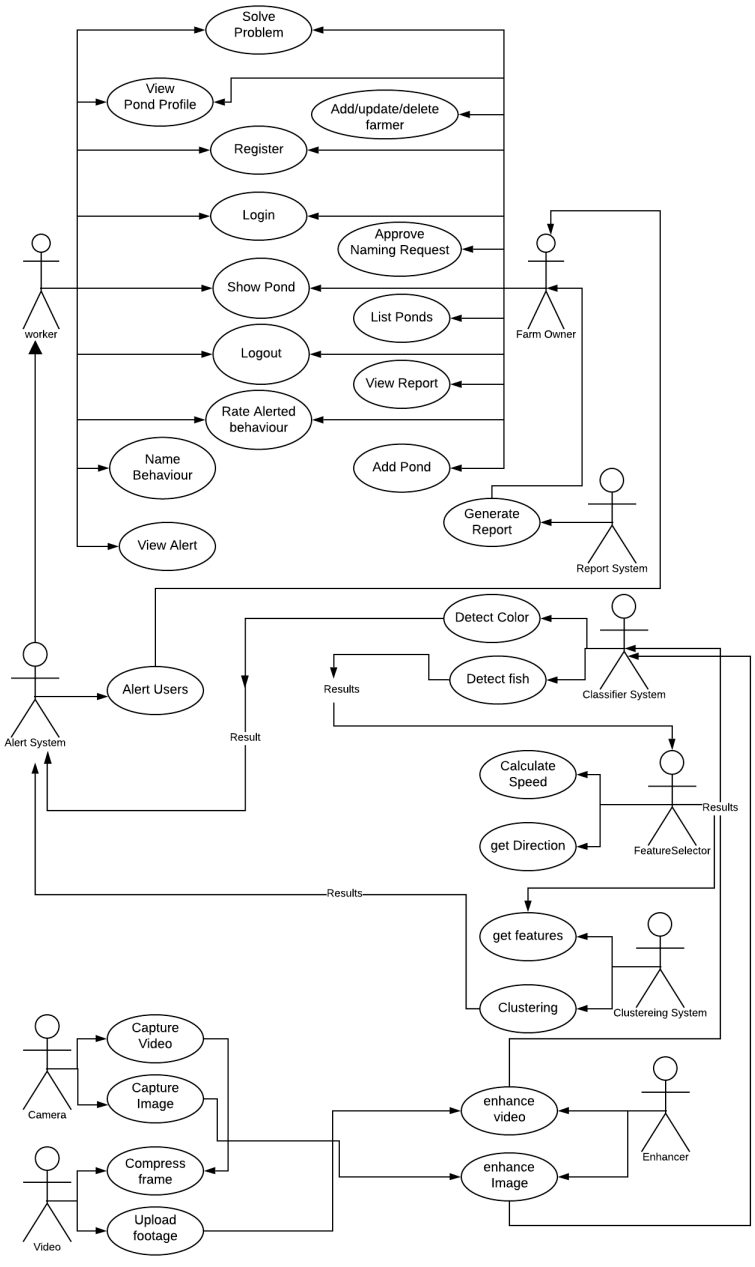

Figure 7: Use Case Diagram

10 Preliminary Schedule Adjusted

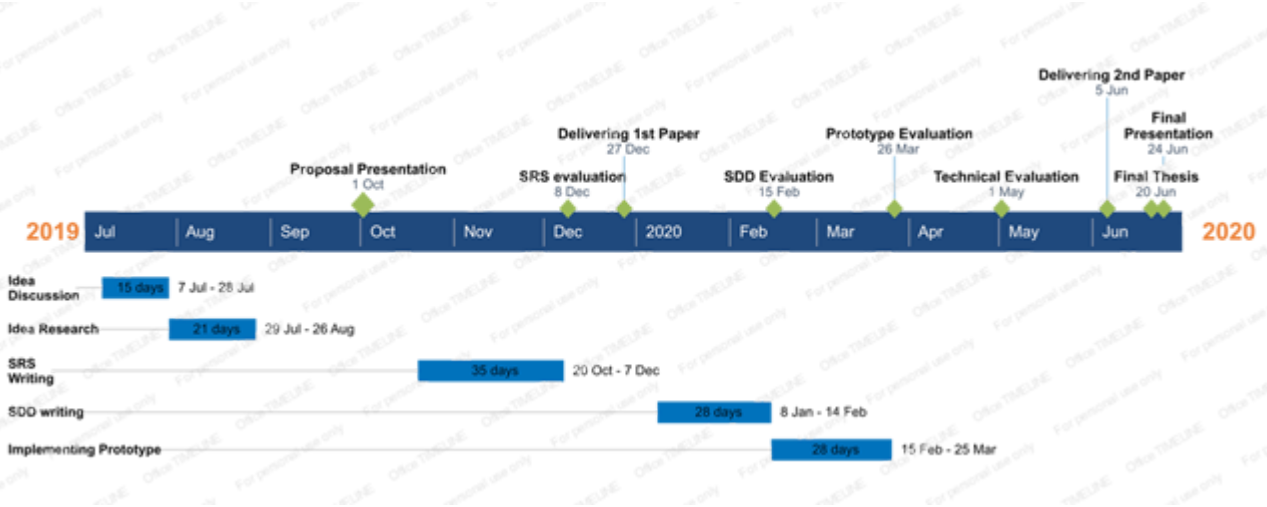

Figure 8: Time Plan

11 Preliminary Budget Adjusted

Endoscope camera: 25 USD
Wireless webcam: 40 USD
Fish pond: 25 USD
Fish pond equipment: 25 USD

References

[1] M. Shaalan, M. El-Mahdy, M. Saleh, and M. El-Matbouli, "Aquaculture in egypt: Insights on the current trends and future perspectives for sustainable development," *Reviews in Fisheries Science & Aquaculture*, vol. 26, no. 1, pp. 99–110, 2018.

[2] C. Beyan and R. B. Fisher, "Detecting abnormal fish trajectories using clustered and labeled data," in *2013 IEEE International Conference on Image Processing*, pp. 1476–1480, IEEE, 2013.

[3] A. Nady, A. Atia, and A. Abutabl, "Real-time abnormal event detection in crowded scenes," *Journal of Theoretical and Applied Information Technology*, vol. 96, pp. 6064–6075, 09 2018.

[4] S. Duggal, S. Manik, and M. Ghai, "Amalgamation of video description and multiple object localization using single deep learning model," in *Proceedings of the 9th International Conference on Signal Processing Systems*, ICSPS 2017, (New York, NY, USA), pp. 109–115, ACM, 2017.

[5] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," 2018.

[6] Y. Toh, T. Ng, and B. Liew, "Automated fish counting using image processing," in *2009 International Conference on Computational Intelligence and Software Engineering*, pp. 1–5, IEEE, 2009.

[7] B. J. Boom, P. X. Huang, C. Beyan, C. Spampinato, S. Palazzo, J. He, E. Beauxis-Aussalet, S.-I. Lin, H.-M. Chou, G. Nadarajan, *et al.*, "Long-term underwater camera surveillance for monitoring and analysis of fish populations," *VAIB12*, 2012.

[8] A. Rodriguez, A. J. Rico-Diaz, J. R. Rabuñal, J. Puertas, and L. Pena, "Fish monitoring and sizing using computer vision," in *International Work-Conference on the Interplay Between Natural and Artificial Computation*, pp. 419–428, Springer, 2015.

[9] H. Lu, Y. Li, and S. Serikawa, "Underwater image enhancement using guided trigonometric bilateral filter and fast automatic color correction," in *2013 IEEE International Conference on Image Processing*, pp. 3412–3416, IEEE, 2013.

[10] R. Lumauag and M. Nava, "Fish tracking and counting using image processing," in *2018 IEEE 10th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment and Management (HNICEM)*, pp. 1–4, IEEE.

[11] Z. Chen, J. Cao, Y. Tang, and L. Tang, "Tracking of moving object based on optical flow detection," in *Proceedings of 2011 International Conference on Computer Science and Network Technology*, vol. 2, pp. 1096–1099, IEEE, 2011.

[12] N. D. Nguyen, K. N. Huynh, N. N. Vo, and T. Van Pham, "Fish detection and movement tracking," in *2015 International Conference on Advanced Technologies for Communications (ATC)*, pp. 484–489, IEEE, 2015.

[13] C. Tang, U. F. von Lukas, M. Vahl, S. Wang, Y. Wang, and M. Tan, "Efficient underwater image and video enhancement based on retinex," *Signal, Image and Video Processing*, pp. 1–8, 2019.