Combined-SRS-SDD-Test (CRDT) Document

Project Name: CENG-I FISH

Group Name: CodeSwimmers

Group Members:

Cansu Alptekin 1941749

Doğukan Yalçın 1942465

Pınar Bil 2043552

Yeliz Topcu 1942416

Table of Contents

Introduction 1

System Requirements 2

1. Short Information about System Requirements 2

1.1. Functional Requirements 2

1.2. Performance Requirements 2

1.3. Interface Requirements 3

1.4. Environmental Conditions 3

1.5. Physical Constraints 4

2. Project plan 4

3. Use case diagram 4

System Design 5

1. Module Structure 5

2. Source Code Structure (under git) 5

3. Component diagram 6

4. Deployment diagram 6

Testing 7

1. Details for system test plan 7

1.1. Test items and their identifiers 7

1.2. Features to be tested 7

1.3. Approach 8

1.4. Pass/fail criteria 8

1.5. Test deliverables 9

2. Test management 9
2.1. Planned activities and tasks/test progression 9
2.2. Environment / Infrastructure 9
3. Test Case Details 10
3.1. Component Tests 10
3.2. Component Integration Tests 14
3.3. System Tests 17
3.4. Testing Code Structure under git 19
4. Test Results 19
4.1. Overview of test results 19
4.2. Detailed test results 19
4.3. Rationale for decisions 21
4.4. Conclusions and recommendations 22
References 22
Appendices 22

Combined-SRS-SDD-Test (CRDT) Document

Abstract

This document provides some details for system requirements (SRS), design and implementation details (SDD), and test documentation (TD) of the project CENG-I FISH.

Introduction

The purpose of this study is to design and development of a robotic fish. The goal is to make a research about unmanned underwater vehicles, which we think there are not enough R&D activities especially in Turkey, instead of unmanned aerial and ground vehicles which are used more commonly almost everywhere nowadays.

The robotic fish designed and developed in the scope of the study, is able to perform its movements through two lateral fins (up & down movement), one caudal fin (left & right movement) and one propeller (propulsion). The robotic fish can be controlled either via a remote control by an operator or via a computer again in a wireless manner. There are one camera and various distance sensors on the fish, which enables it have the ability to avoid obstacles. Through the camera on itself, by communicating with the computer, the robotic fish has the ability to give a sign, e.g. turning the light on, when a face is detected on the video stream.

The motivation for the project is to develop a new approach for underwater search & rescue activities, but the project can also be customized for underwater research and military activities.

System Requirements

1. Short Information about System Requirements

1.1. Functional Requirements

The CENG-I FISH shall provide a Graphical User Interface to user to control the movements of fish, change the speed of fish and turn the light on/off.

The CENG-I FISH shall provide an interface to user to control the movements of fish, change the speed of fish and turn the light on/off via RF Remote Controller.

The CENG-I FISH shall avoid from obstacles and operate all the tasks without collision.

The CENG-I FISH shall provide wireless data transmission between the system and interface which is provided to user.

The CENG-I FISH shall provide an interface to user to detect faces from real time video stream.

The CENG-I FISH subsystems shall accept requests from an operator to transition the CENG-i FISH subsystem's mode. (Virtual Remote Controller/RF Remote Controller)

1.2. Performance Requirements

The CENG-I FISH shall perform 1 hour with the 2 fully charged 11.1 V LiPo batteries.

The CENG-I FISH shall perform its abilities up to 2 metres away from the transmitter source (i.e. computer)

1.3. Interface Requirements

The CENG-I FISH shall provide internal interfaces to decoupled CENG-i FISH services for a distributed system.

The CENG-I FISH shall provide a Graphical User Interface to get commands from user.

The CENG-I FISH shall provide an external interface via RF Remote Controller to get commands from user.

The CENG-I FISH shall provide a physical interface via data cable between software running on the computer and transmitter module which transfers the commands to the controller module of the system.

The CENG-I FISH shall provide a physical interface via transmitter/receiver module between on-water and underwater modules.

1.4. Environmental Conditions

The facility housing the CENG-I FISH's underwater equipment shall operate in ambient temperatures 15oC to 60oC

The facility housing the CENG-I FISH's underwater equipment shall not be exposed to UV or direct sun rays.

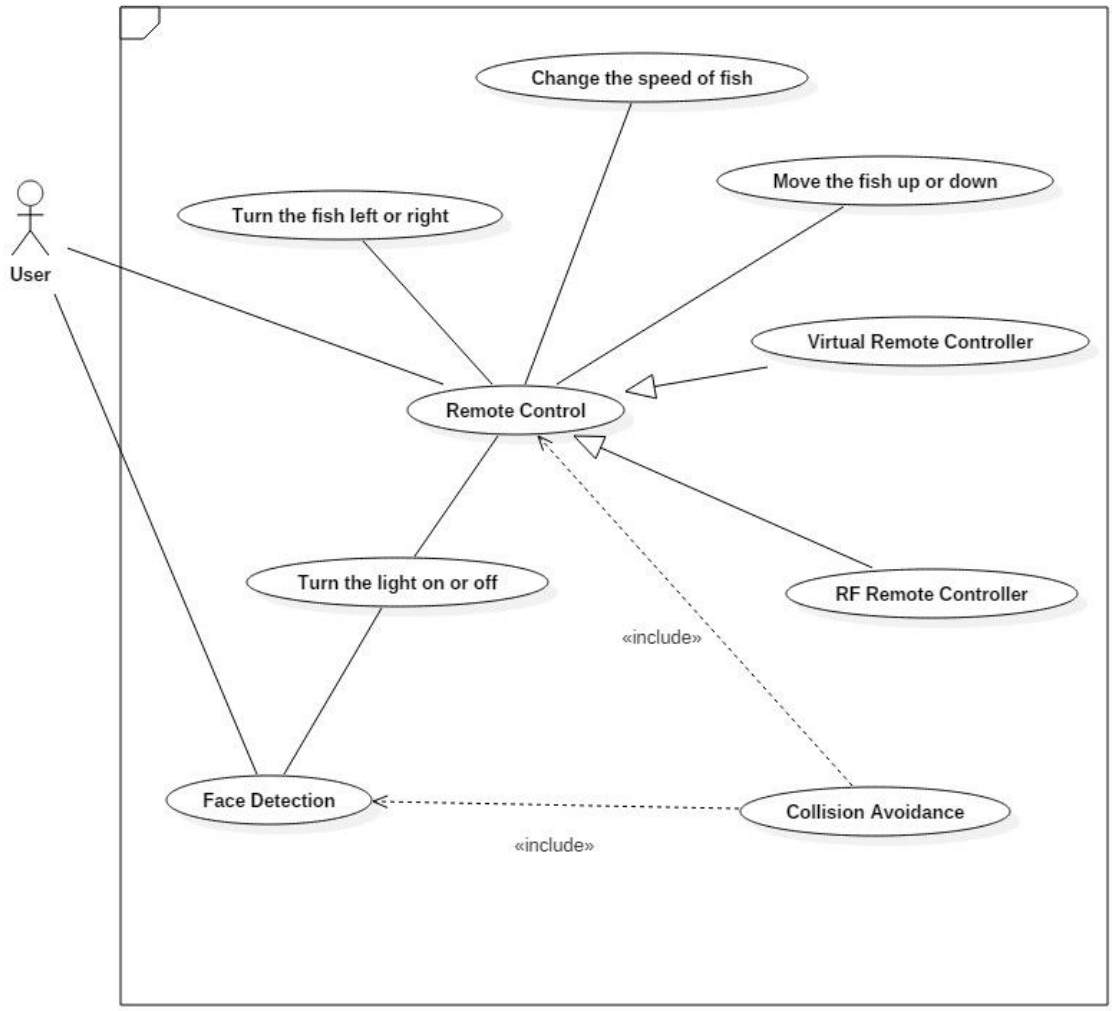
1.5. Physical Constraints

The facility housing the CENG-I FISH's underwater equipment has the dimensions 30 cm*25 cm*45 cm. Therefore, the facility shall operate in the environment which has a wider volume than the facility.

2. Project plan

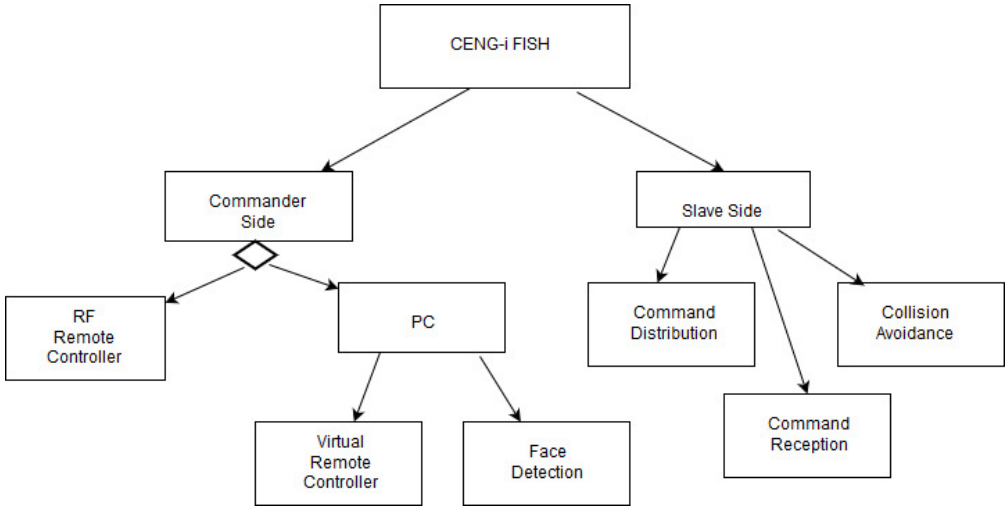
Detailed project plan can be found in Appendix A: Detailed Project Plan

3. Use case diagram



System Design

1. Module Structure

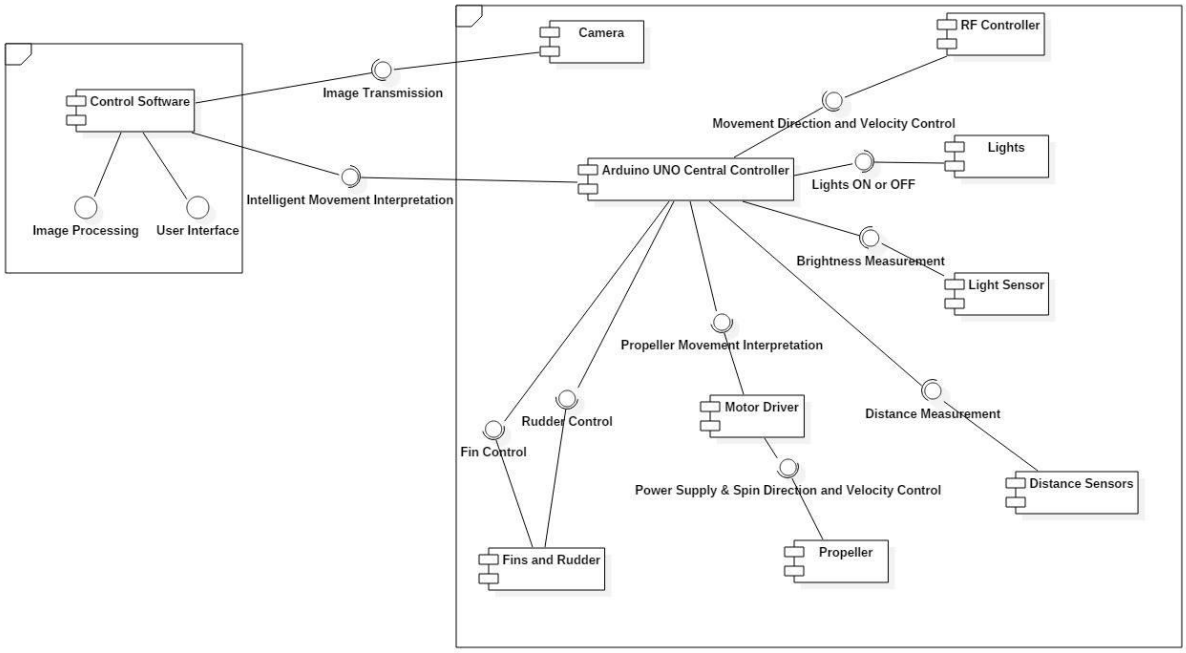


2. Source Code Structure (under git)

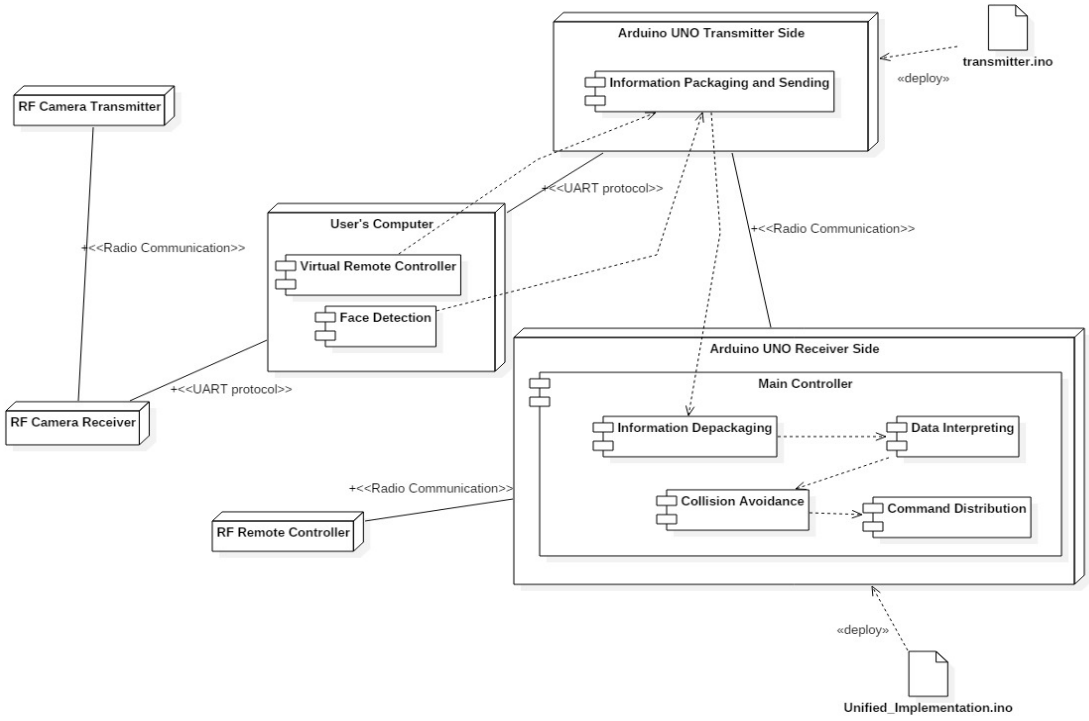
All the source codes are under the directory called “src”

- Unified_Implementation.ino : The .ino file which operates on the Arduino UNO Receiver Side
- transmitter.ino: The .ino file which operates on the Arduino UNO Transmitter Side
- cascadeclassifier.cpp : The .cpp file which captures a real time video stream, detects faces and eyes, and sends a command to turn the light on.
- haarcascade_eye_tree_eyeglasses.xml : The xml file which face detection code requires to detect eyes.
- haarcascade_frontalface_alt.xml : The xml file which face detection code requires to detect faces.
- Qt_GUI: The directory which consists of required files for Virtual Remote Controller GUI.

3. Component diagram



4. Deployment diagram



Testing

1. Details for system test plan

1.1. Test items and their identifiers

QTCreator and OpenCV are used in software part. QTCreator is used for Virtual remote controller. Face detection includes OpenCV library. The system's hardware consists of servo motors, DC motors, distance sensors, transmitter/ receiver, RC controller, light, and camera. Visual inspection is applied for all those hardware components. Some of the components are working together. RC controller-servo motors, RC controller-DC motor, RC controller-servo motors-distance sensors, transmitter/receiver-servo motors, transmitter/receiver-light etc. can be given as an example of components working concurrently. In addition to hardware part, horizontal and vertical sliders, push buttons, radio button are tested on QT creator. Face detection is also tested with respect to distance from the camera.

Moreover, the skeleton of the fish is tested for whether there is water leak or not.

1.2. Features to be tested

The hardware below is tested:

- Servo motors angle movement
- DC motor speed
- Distance sensor's range of distance
- Data transmission among Transmitter/Receiver module
- The range of control stick of RC controller
- Capability of turn the Light on/off
- Receiving data from camera
- Water leak on skeleton of the fish

The software below is tested:

- The horizontal and vertical sliders on QT creator
- Push buttons on QT creator
- Switching radio buttons on QT creator
- Detecting faces with respect to distance to the camera

1.3. Approach

Visual inspection method is applied for testing generally. Component test consists of testing servo motors, testing DC motor, testing IR Distance Sensor, testing transmitter/receiver module, testing RF Camera, Testing LED light. Component Integration tests consist of testing RC controller, testing Virtual Remote Controller (GUI), and testing of Face Detection GUI. Moreover, system tests compose of testing of the whole system, testing of the whole system underwater.

For component testing, all hardware components are expected to work properly. Similar to the tests for each individual component, whole system is also expected to work properly during the system tests.

1.4. Pass/fail criteria

Each component of the system has its own criteria whether it fails or not. However, since all components are hardware, basically, after giving corresponding command, if the component works properly, the test is passed, otherwise, is failed.

1.5. Test deliverables

Test cases, test reports and testing code structure under git are the deliverables of the test documents.

2. Test management

2.1. Planned activities and tasks/test progression

The whole system consists of different components. Before ensuring that all the components of the system are working together simultaneously, some tests needed to be applied. The best way to test the system is starting from its sub-systems. The system consists of different components and some of the components are working together. The each sub-system is tested individually and some of them are tested together. After integration of all sub-systems is completed, the whole system is tested. In some situations, especially for hardware based functionalities, visual inspection is applied.

2.2. Environment / Infrastructure

Test environment includes both hardware and software. In order to test the hardware, Arduino IDE is used regardless of the OS. Moreover, in GUI part, QT creator is used for testing for virtual remote control regardless of the OS. Finally, any OS can be used for face detection. Colour and shape detection and QR detection are also available on any OS. However, those are not used in the system actively.

3. Test Case Details

3.1. Component Tests

Test Case Name	**Testing of Servo Motors**
Test Case ID	TC_C1
Objective	Testing the operability of servo motors
Scenario	Variable angles which are between 0 and 180 degrees will be sent to the servo motors.
Input	Integers representing the angles mapped to the corresponding microsecond values [0,180]->[750,2250]
Outcome	Servo motors' movement according to the given angle
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE (Servo library included)
Special Procedural Requirements	Servo motors' data pins should be attached to the pins of Arduino Uno which is specified in the Arduino code.
Intercase Dependencies	-

Test Case Name	**Testing of DC Motor**
Test Case ID	TC_C2

Objective	Testing the operability of DC motor
Scenario	Variable speed values between 0-255 will be sent to the DC motor. The direction of spin will also be tested.
Input	Integers representing the speed values Setting the +/- as HIGH/LOW or LOW/HIGH
Outcome	DC motor's spin according to the given speed and spin direction
Environmental Needs	Hardware: Arduino Uno, Motor Driver Software: Arduino IDE
Special Procedural Requirements	Motor Driver's enA, inA, inB pins should be attached to the pins of Arduino Uno which is specified in the Arduino code. Motor Driver and DC motor should also be connected together.
Intercase Dependencies	-

Test Case Name	**Testing of IR Distance Sensor**
Test Case ID	TC_C3
Objective	Testing the operability of Distance Sensors
Scenario	Reading values from distance sensor according to the distance of the obstacle
Input	Variable distances far from distance sensor

Outcome	Values between 10 and 500 (representing the distance of the obstacle)
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE
Special Procedural Requirements	Distance sensors' data pins should be attached to the pins of Arduino Uno which is specified in the Arduino code.
Intercase Dependencies	-

Test Case Name	**Testing of Transmitter/Receiver Module**
Test Case ID	TC_C4
Objective	Testing the operability of servo motors
Scenario	A string will be sent from transmitter and received by receiver
Input	Test strings
Outcome	Test strings are received correctly
Environmental Needs	Hardware: 2 Arduino Unos (one for transmitter & one for receiver) Software: Arduino IDE (VirtualWire library included)
Special Procedural Requirements	Transmitter's and receiver's data pins should be attached to the pins of Arduino Unos which each of them belongs to, which is specified in the Arduino code.
Intercase Dependencies	-

Test Case Name	**Testing of RF Camera**
Test Case ID	TC_C5
Objective	Testing the operability of RF Camera
Scenario	Real time video stream will be observed.
Input	Radiowaves which transmits the video stream data
Outcome	Real time video stream can be observed.
Environmental Needs	Hardware: Radio AV receiver & RF camera with transmitter antenna Software: mplayer
Special Procedural Requirements	Radio AV receiver should be connected to the PC via AV USB Grabber. Mplayer should be run with initial configuration which indicates the conversion from raw data to PAL format.
Intercase Dependencies	-

Test Case Name	**Testing of LED Light**
Test Case ID	TC_C6
Objective	Testing the operability of LED Light
Scenario	Turn on/off light via Arduino

Input	Analog values 0 and 255
Outcome	Light is turned on/off according to the commands
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE
Special Procedural Requirements	LED Light has two legs. One leg should be attached to the pins of Arduino Uno which is specified in the Arduino code. Other leg should be connected to Ground (0 V).
Intercase Dependencies	-

3.2. Component Integration Tests

Test Case Name	**Testing of RC Controller**
Test Case ID	TC_CI1
Objective	Testing the operability of RC Controller
Scenario	Sticks of the controller will be moved, data changes will be observed and corresponding command will be sent to the Servo motors, DC motor or light.
Input	Integers [750,2250]
Outcome	All the values mapped into a command set. (E.g. the value coming from the throttle pin of controller is mapped into the

	speed of propeller.) Hardware components are operating corresponding to the commands given from RC controller.
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE (PinChangeInt and ServoTimer2 library included)
Special Procedural Requirements	RC controller's data pins should be attached to the pins of Arduino Uno which is specified in the Arduino code.
Intercase Dependencies	TC_C1,TC_C2, TC_C3, TC_C4, TC_C5, TC_C6

Test Case Name	**Testing of Virtual Remote Controller (GUI)**
Test Case ID	TC_CI2
Objective	Testing the widgets of the Virtual Remote Controller
Scenario	Controlling the rudder and the fins through the sliders, controlling RF remote control/Virtual Remote Control mode change through radio buttons, turning the light on/off through push button
Input	Strings which indicates which component will do what Example Format: 30F (fins turn to 30 degrees)
Outcome	All the values mapped into a command set. (E.g. the value coming from the throttle pin of controller is mapped into the speed of propeller.) Hardware components are operating corresponding to the commands given from Virtual Remote Controller.

Environmental Needs	Hardware: Arduino Uno Software: Qt Creator
Special Procedural Requirements	Serial port which is connected to the Arduino of Transmitter module, should be opened.
Intercase Dependencies	TC_C1,TC_C2, TC_C3, TC_C4, TC_C5, TC_C6

Test Case Name	**Testing of Face Detection GUI**
Test Case ID	TC_CI3
Objective	Testing the Face Detection GUI
Scenario	Faces and eyes will be detected from real time video stream.
Input	Strings which indicates the “light on” or “light off” Format: 0L (light off) or 1L (light on)
Outcome	If detected face is near the camera under a threshold value, the command is sent to turn the light on. If there is no faces or there is a face far away, the light will be turned off.
Environmental Needs	Hardware: Arduino Uno Software: OpenCV library is included
Special Procedural Requirements	Serial port which is connected to the Arduino of Transmitter module, should be opened.
Intercase Dependencies	TC_C1,TC_C2, TC_C3, TC_C4, TC_C5, TC_C6

3.3. System Tests

Test Case Name	**Testing of the whole system**
Test Case ID	TC_S1
Objective	Testing all the components simultaneously
Scenario	Giving commands from RF remote controller, Virtual Remote Controller GUI and Face Detection GUI. Observing the fish operating according to the commands and avoiding from obstacles at the same time.
Input	String from transmitter (GUIs) Microsecond values (RF remote controller)
Outcome	If there is an obstacle in front of the front distance sensor, the propeller will be stopped. If there is an obstacle in front of left or right distance sensor, the rudder will not turn to that way. Otherwise, all the capabilities are observed.
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE, QtCreator, OpenCV
Special Procedural Requirements	All the components should be connected to the corresponding pins of Arduino.
Intercase Dependencies	TC_CI1, TC_CI2, TC_CI3

Test Case Name	**Testing of the whole system underwater**
Test Case ID	TC_S2
Objective	Testing all the components simultaneously underwater
Scenario	Giving commands from RF remote controller, Virtual Remote Controller GUI and Face Detection GUI. Observing the fish underwater operating according to the commands and avoiding from obstacles at the same time.
Input	String from transmitter (GUIs) Microsecond values (RF remote controller)
Outcome	If there is an obstacle in front of the front distance sensor, the propeller will be stopped. If there is an obstacle in front of left or right distance sensor, the rudder will not turn to that way. Otherwise, all the capabilities are observed.
Environmental Needs	Hardware: Arduino Uno Software: Arduino IDE, QtCreator, OpenCV
Special Procedural Requirements	All the components should be connected to the corresponding pins of Arduino.
Intercase Dependencies	TC_CI1, TC_CI2, TC_CI3

3.4. Testing Code Structure under git

All the test codes are under the directory called Test.

- Hardware Implementation: All the test codes for component tests are under this directory.
- Image Recognition: All the test codes for testing QR-Code, Color, Shape and Face Detection are under this directory.
- Machine Learning: Data sets under this directory are created for future work.
- Serial Communication: All the test codes for testing serial communication between computer and Arduino are under this directory.
- Graphical Interface QTCreator: All the test codes for testing the components of GUI are under this directory.

4. Test Results

4.1. Overview of test results

All the test items stated in Section 3 are tested completely outside the water environment with the 3D-printed fish body (v1). Response time of hardware is fast when receiving commands from any controller. Face detection is very sensitive for all faces and the quality of video stream from RF camera is sufficient for face detection.

4.2. Detailed test results

Test Case ID	**Results**
TC_C1	PASS Possible Anomaly & Resolution: If the sufficient voltage and current is not supplied, servo motors do not work properly. Required

	voltage, regulator and heat sinks should be supplied.
TC_C2	PASS Possible Anomaly & Resolution: If the sufficient voltage is not supplied, DC motor do not work properly. Required voltage should be supplied.
TC_C3	PASS Possible Anomaly & Resolution: -
TC_C4	PASS Possible Anomaly & Resolution: -
TC_C5	PASS Possible Anomaly & Resolution: Drivers of USB AV Grabber should be manually installed to the user's computer. Otherwise, system would not work.
TC_C6	PASS Possible Anomaly & Resolution: -
TC_CI1	PASS Possible Anomaly & Resolution: PinChangeInt library and Servo library has overlapping timers (compile time error). Therefore, ServoTimer2 library is used to control the Servo motors.

TC_CI2	PASS Possible Anomaly & Resolution: If the user do not connect the USB data cable of Arduino UNO Transmitter Side to the USB port which is specified in the code, either the code should be manually changed or the user should find the correct port.
TC_CI3	PASS Possible Anomaly & Resolution: If the user do not connect the USB data cable of Arduino UNO Transmitter Side to the USB port which is specified in the code, either the code should be manually changed or the user should find the correct port.
TC_S1	PASS Possible Anomaly & Resolution : -
TC_S2	NOT TESTED YET Possible Anomaly: The system may not operate properly because of the lift force of the water, balance and water leak.

4.3. Rationale for decisions

Rationale for decisions is to provide wireless communication between computer or RF remote controller and the components in the underwater environment.

4.4. Conclusions and recommendations

All tasks and use cases are tested in a proper way. Testing the hardware components took a long time since they do not work deterministically because of the noise of the environment. We had some trouble during the testing the integration of software and hardware, since the Arduino listens the commands sometimes from RF remote controller and sometimes from Virtual Remote Controller. In addition, GUI side cannot find or open the Serial port; we should change the port manually in the code.

Because of the insufficient opportunities, we are planning to complete underwater tests until the Demo Day.

After all the tests are done, our project will be available for production use. For now, the project is not user-friendly and not easy to install and configure; it can be improved for future work.

References

IEEE Standard for Software and System Test Documentation-IEEE Std 829™-2008

Appendices

Appendix A: Detailed Project Plan