MAKERERE UNIVERSITY
COLLEGE OF COMPUTING AND INFORMATION SCIENCES
BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING
BSE 4100: SOFTWARE ENGINEERING PROJECT 1

NAME	REGISTRATION NO.	SIGNATURE
Lwanga Darline	17/U/5836/EVE	
Kisa Jeremiah	17/U/5250/EVE	
Kyadondo Solomon	17/U/5538/EVE	
Aliho Walter		

PROJECT MEMBERS BSE22-27

Project Blog: http://sites.google.com/view/drowndetection

Supervisor Dr Odongo Steven

SOFTWARE REQUIREMENTS SPECIFICATION

for

The Drown Detection System:

A Swimming pool embedded system for user recognition

Version 2.0

Prepared by BSE22-27 Team

Final Year Project

15th April 2022

Table of Contents

Contents

List of Tables

Revision History

Name	Date	Reason of Change	Version
Darline Lwanga	23rd March 2022	Addition of the proper Product description	1.1.1
Kyadondo Solomon	15th April 2022	- Refactoring of the specific requirements - Addition of the detection model requirements	2.0.0

1. Introduction

1.1 Purpose

This document describes the system requirements for the drown detection system. It is intended for all stakeholders who will use the system, system administrators, testers and coders.

1.1.1 Main objective

To develop a system that detects drowning in swimming pools and makes alerts

1.2 Intended audience

All stakeholders which include; lifeguards and operators. Swimmers, emergency response unit Since we intend to have the idea that affects swimming, anyone who uses the swimming pool is a stakeholder in the project. These stakeholders provide the requirements for the project and are thus an audience for this SRS. This document also serves to guide the developers and testers in the implementation of the project.

1.3 Scope

The project aims to detect drowning cases in swimming pools both public and private and also both indoor and outdoor swimming pools.

This description entails the different features of the system, their functions and the interfaces of the system.

This document is intended for system stakeholders such as end users, system developers, project supervisors and potential owners of the system. [1] [2]

1.4 Definitions, Acronyms, and Abbreviations

Table 1.1 Description of the outputs of the Drown Detection System

Term	**Meaning**
Depth	Distance of an object from the surface of the water
Object	An acronym for a test subject undertaking the event of swimming

1.5 References

[1] M. A. Y. G. I. A. S. A. &. M. M. Hayat, Comprehensive and Comparative Study of Drowning Person Detection and Rescue Systems, (2019, November).

[2] P. &. J. A. Laxman, Robust Drown Alerting, Preventing and Autonomous Rescue System, The Scientific Bulletin of Electrical Engineering Faculty, 2021.

1.6 Overview

This document contains the following sections;

Overall Description Section

This gives a general idea of the functionality of the product. It includes a conceptual description of the requirements.

Specific Requirements

this is for the developers and it describes in technical terms the details of the functionality of the product.

2 General Description

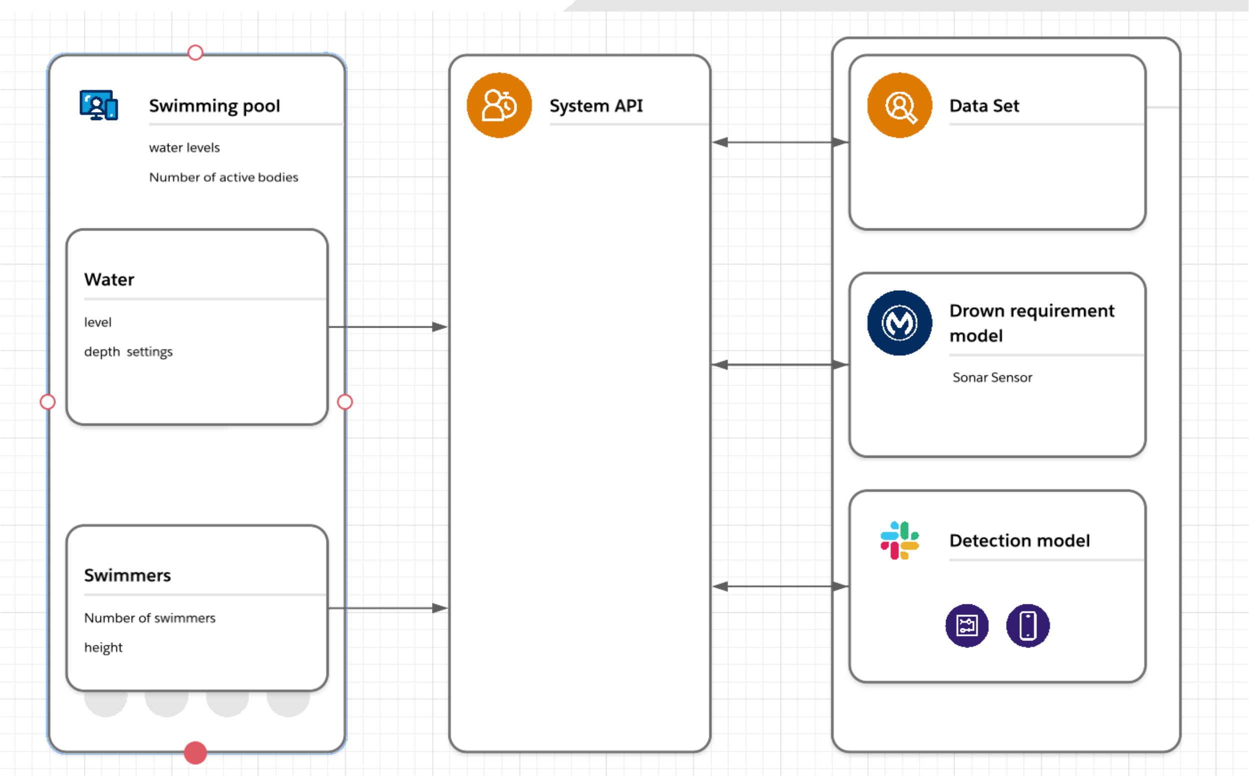
2.1. Product perspective

The system will contain sonar sensors that will be able to keep track of the objects under water. Sonar sensors work in a way that the sensor produces sound and waits for the feedback or echo from an object or obstacle. [2]

The distance pf this obstacle is measured and plotted on a graph.

Using the plotted graph, we can tell the position and motion of the obstacle or object in water.

In case it detects that the object is stuck in one positon, the system will sound a red alert to warn the guide.



2.1.1 System interfaces

The interface will be able to provide an active real time report to the administrator with no fail, this information includes the environment behavior and its changes.

2.1.1.1 Web client

This interface avails the user the ability to interact with the system, this interface will have the major role of creating a log of the system and enabling the administrator create a report of the daily or monthly events.

2.1.1.2 System API

- The system will use the dataset and sensors to evaluate whether an object/body is drowning and will be able to use these properties to send alerts in form of a report to the admins
- The system will detect changes in the levels from the closed environment testing and stop the model so as to prevent improper readings.

2.1.1.3 Swimming pool API

a) For prediction of water levels and swimming depths

   In situations of neutral or normal floating situations water levels and depths for specific bodies need to be known for example a body that is standing upright with their head above water.

b) For prediction of the number of objects active in the environment

   The system will use sensors to detect the amount of normally functioning beings in the pool and their attributes so as to provide appropriate feedback to the system.

2.1.1.4 Dataset

The database contains the various swimmer details such as height and admissible body conditions for example heartbeats per minute and stress levels and also their distress information for a group of individuals that are near drowning.

The reports of daily events are also kept here and any external data.

2.1.1.5 Drown Requirement Model

This considers the parameters from the dataset to determine whether our entity a swimmer of a specific type from the system API is experiencing any of the conditions that are substantial to drowning.

2.1.1.6 Detection model

This model takes in input from the system API to carry out an algorithm of classification from body types and external user data to drown computations.

2.1.2 Hardware interfaces

The system will use Led displays to check if the components come together properly and give initial feedback on there readings before they can be logged.

2.1.3 Software Interfaces

The system will function with Arduino software and will also support kernel OS such as LINUX or MACOS for API calls.

It will also interact with web APIs to perform data logging.

2.1.4 Memory Constraints

The software necessary for embedded development will not allow the creation of programs bigger than 2048KB of SRAM.

2.1.5 Operations

There are no operations defined in this version of the SRS document.

2.1.6 Site adaptation requirements

The environment (Swimming pool) will have not more than three depth setting.
The swimming pool will hold a steady level of water that is not adjustable and admits little or no wave forms
The deepest end of the pool will not be deeper than 6 feet.

2.2. Product features

The system will support the following phases and processes: -

1. Taking measurements: - the system will use the sonar sensor to take recording such as the distance of the object and direction or bearing of that object. The results will be stored in a database
2. Plotting the results: - the system will plot the position of an obstacle of a graph using the measurements obtained
3. Detection model: - the system will then use a detection model to detect whether the object in moving or it is in one position.
4. Alerting: - using the detection model, if the object is stuck for more than 10 seconds, it will issue a warning. If the object is stuck for more than 20 seconds, the system will issue a red alert and sound an alarm.

2.3. User class and characteristics

the users of this system include two main types, the swimmers and the system administrators who mainly control its API functionality.
the swimmers interact with the system using external/additional devices.

2.4 Constraints

I. The developers of the system must have knowledge of Arduino architecture.
II. Reports will be accessed via web through use of web browsers.
III. Modification of the system API and detection model will only be authorized to the system developers.
IV. The reports of functionality of the system will only be accessed by the administrator.

2.5 Assumptions and dependencies

1. The swimming pool shall have an electricity supply for the system devices.
2. The swimming pool has access to a computer.
3. The detection model is dependent on the drown requirement model.
4. The system will have real time access to the internet with little noise or delay.

2.6 Apportioning of Requirements

Generation of web reports by the system will be developed in the later stages of the system.

3. Specific Requirements

The system will be composed of x major modules. For each of the modules, we detail the use-case narratives, which show steps involved in executing the processes. Furthermore, each module functional requirement is uniquely identified by 2 initials, corresponding to the name of the module.

3.1. External interfaces

3.1.1. User Interfaces

This section illustrates how the system will be presented to the users through the graphical user interface (GUI). It will allow its users to interact with the menus, outputs and other GUI elements.
This subsection provides sketches and a brief description of the key interfaces of the Malaria Parasite Detection System for interaction of the users with the web client of the system.

3.1.1.1. display interface

This interface will be loaded every time the user wants to perform parasite detection where a blood smear image has been provided. It is on this interface where the user shall be able to initiate image upload of blood smear.

3.1.1.2. sonar interface

This interface will capture and feed data to the system.

3.1.2 System API

This is the API that conveys data to the processing and machine learning
Algorithm for detection of an incident

3.2 Functions

R001: Drown detection system shall be able to detect all objects in the swimming pool using a sonar sensor
R002: Drown detection system shall be able to process and plot the data collected
R003: Drown detection system shall use a machine learning algorithm to process data to detect drowning cases
R004: Drown detection system shall sound an alarm when it detects drowning case
R005: Drown detection system shall store the data into a database

3.2.1 Object detection

Table 1. Object detection

Process Name	Object detection,
Actors	sonar sensor
Goal	To detect all objects in the swimming pool
Precondition	No object detected
Main Flow	1. The sensor measures the distance of the obstacle 2. The distance and the bearing of the object is measured and stored in the database.
Exceptions	
Post Condition	Object distance and bearing measured.

. The system will provide the following functional requirements in regards to the object detection process:-

FROD01: measure the distance of the object from the sensor

FROD02: measure the bearing of the object relative to the sensor position

3.2.2 Plotting the data

The position of the object in the swimming pool is determined

Table 2: Plotting data

Process Name	Plotting the data
Actors	System
Goal	The goal is to process the data collected and use it to determine the position of the object in the swimming pool
Precondition	Distance and bearing of the object obtained in the database.
Main Flow	1. Retrieve the data from the database 2. Process the data to locate the position of the object 3. Plot the object's position
Exceptions	
Post Condition	The position of an object in the swimming pool is known

FRPD01: process the data to determine the position of the object in the swimming pool

FRPD02: plot the position of the object on a graph

3.2.3 Machine learning (Drown detection)

In this process, the system will analyze the data process to keep track of the object and to know whether the object is moving or whether the object is constant. If the object is not moving, it is flagged as a drowning case.

Table 3. machine learning (drown detection)

Process Name	Machine learning
Actors	System
Goal	To keep track of the objects movement or motion.
Precondition	Position of the object is know
Main Flow	1. The previous graph plot is compared to the new plot 2. The positions of the object are analyzed on the graphs to know whether the object is moving or it is in one position 3. The ML model will use the patterns of the graphs keep track of the objects 4. If an object is in one positons for 10 seconds, a warning will be issued 5. If the object is in one place for more than 20 seconds, a red alert will be issued

Exceptions	
Post Condition	Motion of an object is known.

The machine learning process has the following functional requirements;

FRML01: detect objects that are in one position and not moving.

3.2.4 Alerting

This process is triggered in case an emergency or any drowning is detected within the pool.

Table 4. Alerting

Process Name	Alerting
Actors	Speakers, system
Goal	To ensure that an alarm is made to notify the guide or nearby person.
Precondition	A drowning is detected

Main Flow	1. The detection model detects a motion less object for more than 20 seconds 2. The alarm is triggered
Exceptions	
Post Condition	Responsible personals respond to the call and the drowning person is attended to

Alerting process has the following functional requirements;

FRALT01: turn on the alarm to make a sounding alert

FRALT02: issue a red alert

3.3. Performance Requirements

The drown Detection System should employ data quality techniques on

Input through the sensor to limit errors.

The drown Detection System should provide at least 95% accuracy of object detection

3.4. Logical Database Requirements

Data will be stored with respect to time.

Data will be stored in numeric format

3.5. Design Constraints

3.6. Software System Attributes

3.6.1. Reliability

The system should be able to restart automatically in case of any crash. The system should be able to keep logs.

The system should have a reliable power source. The system should be able to perform well in both indoor and outdoor swimming pools.

The system should be able to take accurate readings

3.6.2. Availability

The drown detection system should be up and running every time the swimming pool is in use.

3.6.3. Security

The results should not be altered or modified by unauthorized personnel.

The system should be restricted to staff personals only.

The system administrators will have special authentication credentials that will give them access to higher privileged functions.

3.6.4. Maintainability

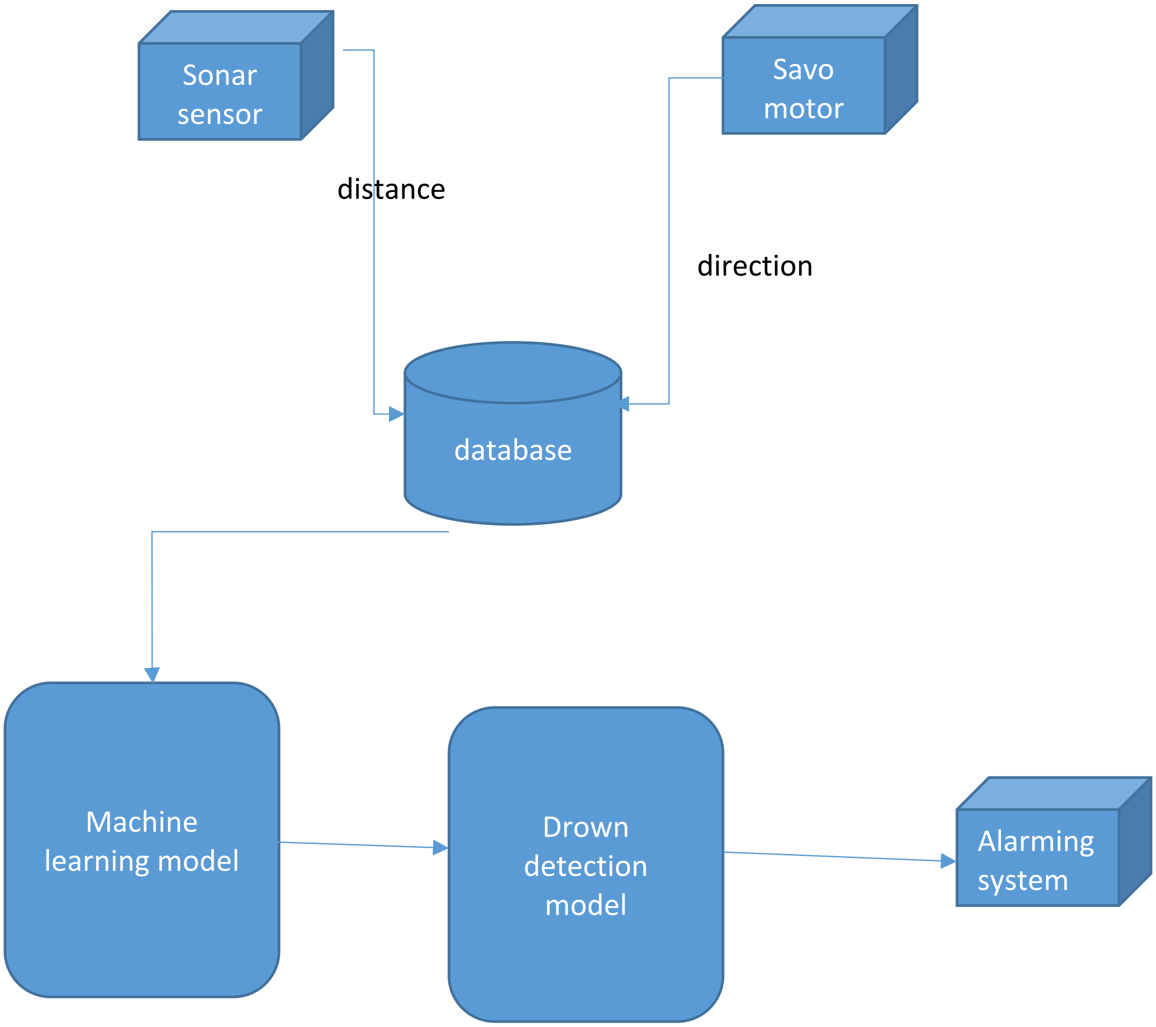
The system will be checked weekly to verify whether it is working properly.

3.6.5. Portability

The system will be developed in python and will be easily connected to a network through APIs.

3.7 Functional Hierarchy

This section describes the system using a Context Diagram.


Sonar sensor
Savo motor
distance
direction
database
Machine learning model
Drown detection model
Alarming system

4. Change Management Process

There are various stakeholders who can propose changes to this SRS - the project supervisor, coordinator, potential customers. Changes to this SRS will be made by any of the members on working on the project.

The changes will be categorized into two - Major and Minor changes. Major changes are concerned with addition/deletion/modification of functional/non-functional requirements. Minor changes are concerned with any general edition of the SRS content - tables, diagrams, definitions.

The SRS includes a Revision History page that describes the Name of member making change, Date of change, Reason for change and Version (of the document).

For any change to be made, a communication will be sent out to the BSE21-10 Group by email.

The members of the Group will collectively review the change proposal and look for the best way to apply the change.

The SRS will then be modified to include the change. The Review History table will be updated accordingly.

The project supervisor will review the changes made to make an approval or propose modification.