

FINAL DELIVERABLE PROJECT DOCUMENTATION

Team ID	PNT2022TMID24925
Project Name	VirtualEye-Lifeguard for Swimming Pools to Detect the Active Drowning

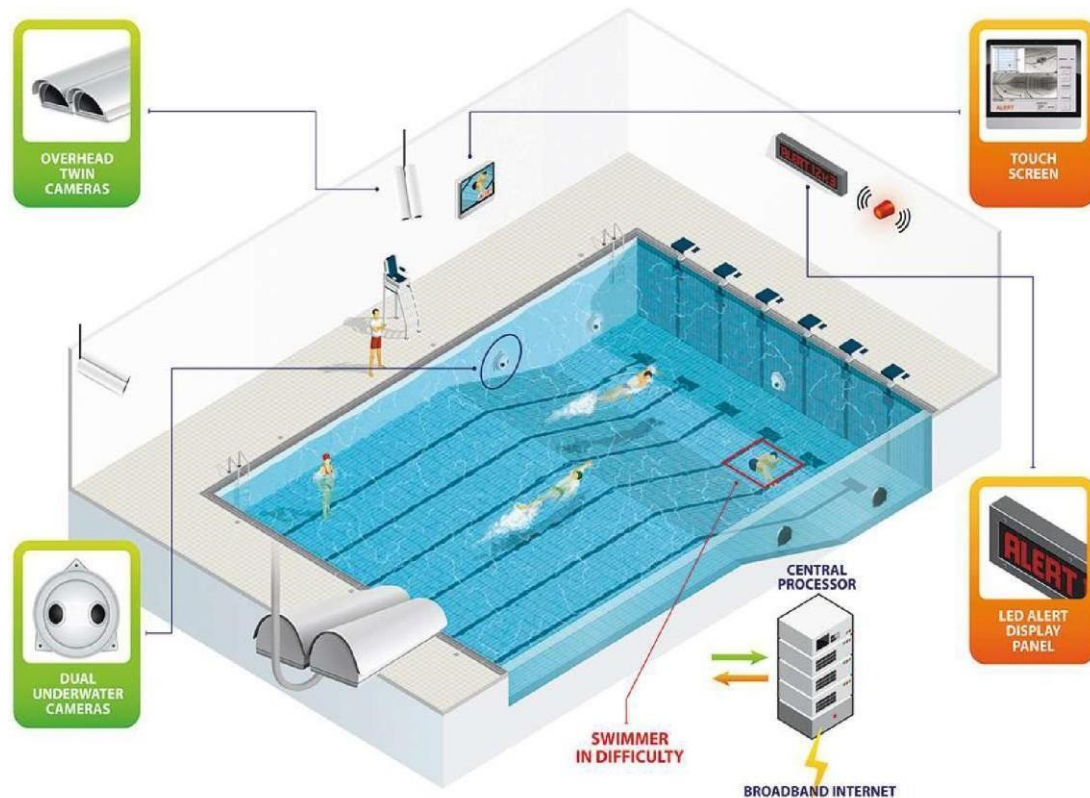
CHAPTER-1 1. INTRODUCTION

Recently, there has been growing interest around the topic of drowning detection systems (DDS) in the sport and leisure industry both across the UK and globally. Advancements in technology, coupled with the importance of pool safety, has led to its growing prominence, with mention of DDS now in documents such as HSG179 - the latest UK standards document for health and safety in swimming pools (Health and Safety Executive, 2018). However, the topic is a debated area for various reasons explored in this review.

Whilst there are plenty of academic articles dedicated to the technology and design behind these products in the fields of biometrics, computer science and electronic engineering, there is limited academic research investigating their application to real-world scenarios. Furthermore, there is uncertainty around their use alongside traditional lifeguarding; whether international testing standards (ISO standards) are robust enough; and general risks affecting the effectiveness of these products. This includes factors such as water clarity, high pool occupancy, lighting, glare and attractions such as water slides and wave machines. These concerns alongside the lack of research and high installation costs have resulted in a reluctance by some operators to incorporate DDS into their pools. This signifies the importance of independent research into DDS. intends to support the move towards the shared goal of improved pool safety.

This piece will begin with an overview of the different definitions of DDS, followed by an explanation of the aims and methodology of this review. It will then discuss what the current DDS standards are alongside legislation and guidance available around DDS, and provide a summary of the shared responsibilities towards the effective operation of DDS. Following this, the literature review will examine the co-existence between DDS and traditional lifeguarding, provide an analysis of its impact so far, and conclude with recommendations on the direction of future DDS research.

1.1 Project Overview



1.2 Purpose

- >> Establish and outline what is known on Drowning Detection Systems.
- >> Evaluate the current literature on Drowning Detection Systems, including their use in indoor pool environments along with interaction with traditional lifeguarding.
- >> Better understand where DDS are positioned in the health and safety landscape of indoor swimming pools.

The value that can be generated from these aims stem from the recognition that currently, there are no published documents drawing together all the current DDS research. The literature review aims to contribute as independent research in this field and hopes to signpost the potential future direction of DDS research.

2. LITERATURE SURVEY

Of the differing definitions of DDS, most outline three defining elements:

- 1) surveillance,
- 2) detection of a pool user in difficulty, and
- 3) raising an alarm

For example, ISO_20380 (the document published by the International Organisation for Standardization (2017) outlining the international safety requirements and test standards for DDS) defines the technology as an ‘automated system including means for digitizing series of images of people in the pool basin, means for comparing and analysing digitized images and decision means for setting off and sending an alarm to trained staff when a detection occurs’. In comparison, there are broader definitions that are inclusive of other technologies that focus on the surveillance aspect, for example, ‘DDS is used to describe various electronic systems that are designed to assist with the surveillance of swimmers within the water of a swimming pool’ (Sport England, 2011). This definition would include CCTV that helps give lifeguards an underwater view but does not have the capacity to detect a pool user in difficulty or raise an alarm. For this to be effective, staff would have to make sure the CCTV is being monitored at all times, making the staff experience with this very different to the experience of using a DDS falling under the first definition. It is important to distinguish what exactly constitutes a DDS as there are different areas of responsibility required from different actors involved in the effective operation of DDS, which will be examined in chapter 4. For this literature review, research has focused on the definition used by the ISO and other sources that incorporate all three elements of surveillance, detection and alarm raising.

2.1 Existing Problem

Whilst literature on DDS mostly agrees on areas such as the risks and issues associated with DDS performance, there are other areas where sources offer differing points of view, for example, DDS and their co- existence with lifeguards. There is debate around whether DDS can be helpful or harmful towards lifeguarding practices and how

DDS may change the landscape of traditional lifeguarding, as well as some disagreement on whether they serve as justification for reducing lifeguard numbers. The term ‘blended lifeguarding’ or ‘modern lifeguarding’ has been newly coined to describe the concept of traditional lifeguarding practices being blended with technology for drowning detection (Swimming Pool Scene, 2017).

Currently, there is little qualitative or quantitative research analysing the experiences of lifeguards themselves relating to this concept.

2.2 References

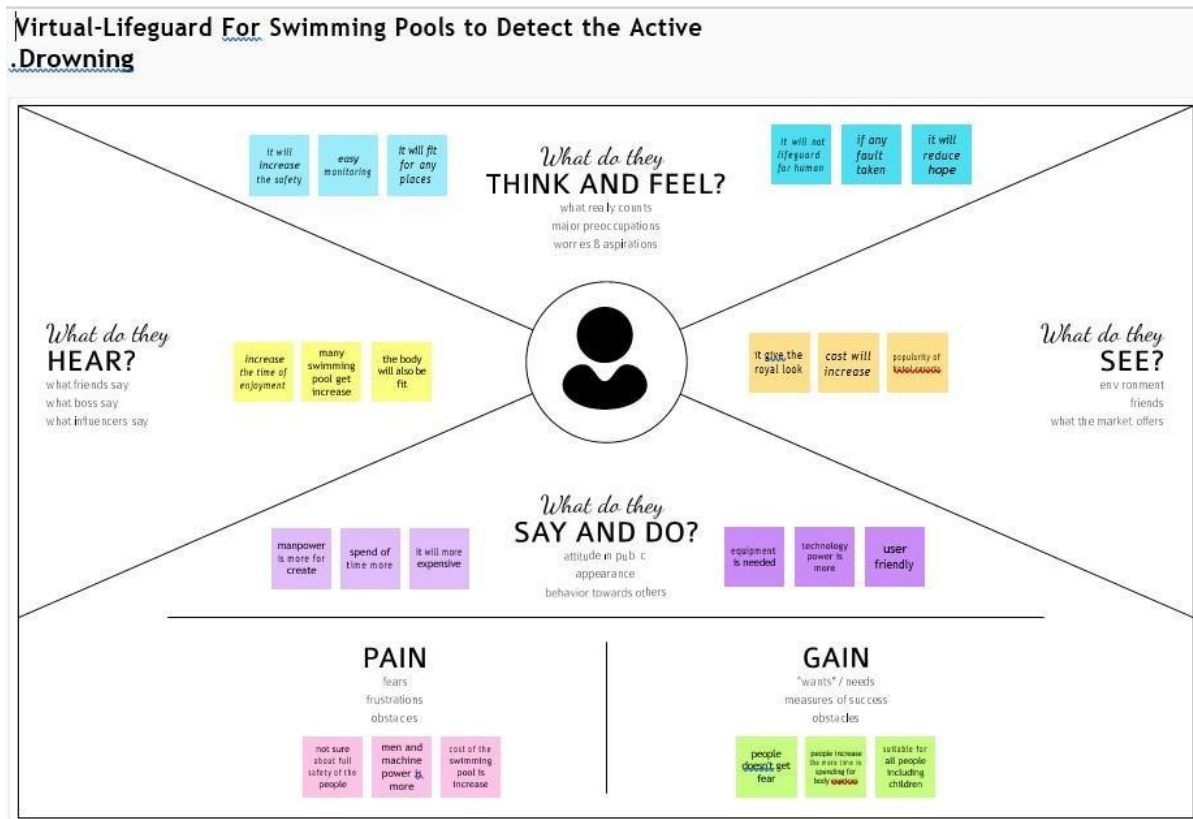
- [1] AngelEye. (2019). AngelEye – Distributors. Retrieved from: <https://www.angeleye.it/news.php?id=28&newscat=10>
- [2] Aquatics International. (2007). Traumatic Experiences – Should we make our youngest lifeguards come face to face with death? Retrieved from: https://www.aquaticsintl.com/facilities/traumaticexperiences_o
- [3] British Standards Institution. (2018). BS EN 15288-1, Swimming pools for public use. Safety requirements for design. Retrieved from: <https://shop.bsigroup.com/ProductDetail/?pid=000000000030360254>
- [4] British Standards Institution 1. (2018). BS EN 15288-2, Swimming pools for public use. Safety requirements for operation. Retrieved from: <https://shop.bsigroup.com/ProductDetail/?pid=000000000030360257>
- [5] Drowning Prevention. (2017). The Need. Retrieved from: <https://www.drowningprevention.com.au/>
- [6] German Institute for Standardization. (2019). German national guideline DGfDB R 94.15 “Test methods for camera-based drowning detection systems under operational conditions” (German Association for Public Swimming Pools).
- [7] Haizhou Li, Haizhou Li, Kar-Ann Toh and Liyuan Li. (2012). Advanced Topics in Biometrics, World Scientific Publishing Co. Pte. Ltd., ISBN-13 978-981-4287-84-5
- [8] Health and Safety Executive. (2018). HSG179, Health and safety in swimming pools (Fourth edition).
- [9] ISO (2017) ISO_20380, First edition, Public swimming pools — Computer vision systems for the detection of drowning accidents in swimming pools — Safety requirements and test methods.

2.3 PROBLEM STATEMENT DEFINITION

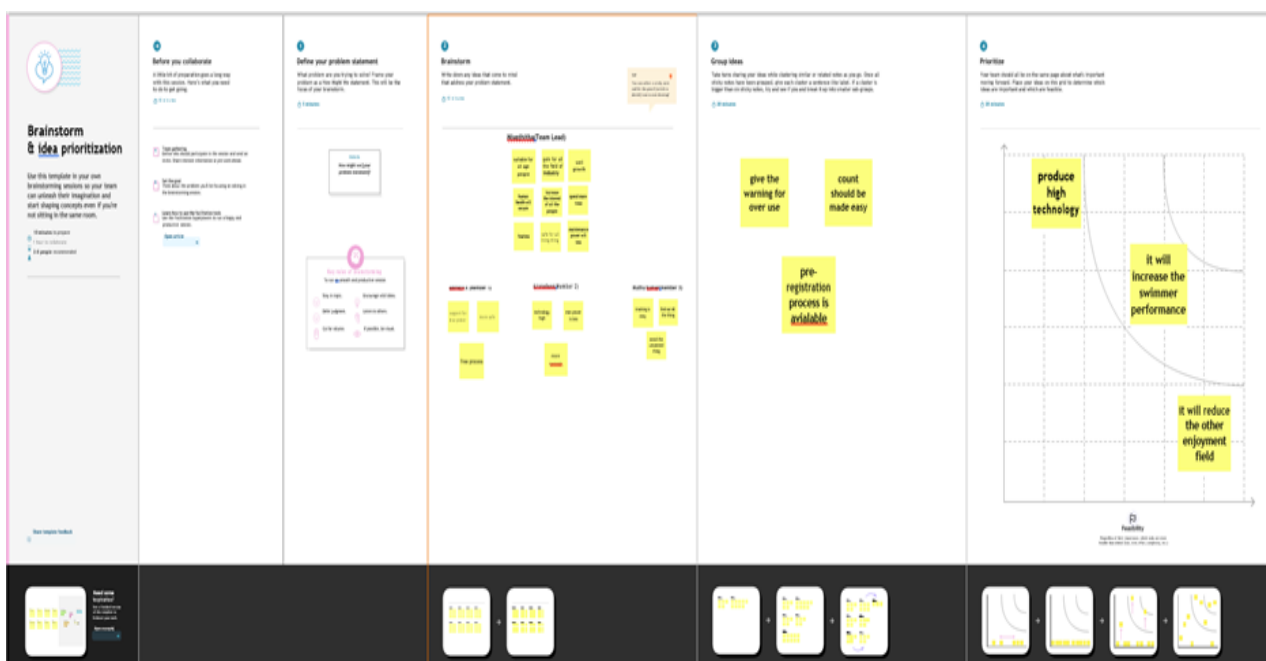
1. Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in hotels.
2. Applying the CNN algorithm to the dataset. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident.
3. To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life.

3.IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING



3.3 PROPOSED SOLUTION

Proposed Solution Template:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	People use the swimming for enjoyment, health Exercise but for all age of the people pool is really dangerous we need lifeguard, in duty swimming pools are very dangerous in the underwater.
2.	Idea / Solution description	In this project, using Artificial intelligence technology, using the camera help we can detect the people action and positions and also we check breathing level of the people inside the underwater and use of any alarms system we can detect the some of them are in the problem
3.	Novelty / Uniqueness	The uniqueness of the our system is track the people position and body condition in the drowning using YOLO Algorithm. It is fast and very speed in the detection
4.	Social Impact / Customer Satisfaction	In world most of them are unexcepted cause very serious death in the underwater not only in the city but most occurs in the rural area in the public places (well, lakes) we should avoid the accident in the underwater drowning
5.	Business Model (Revenue Model)	In the software field this well increase good income. Safety innovation in the swimming related issues this makes attractive for end users to use our software product
6.	Scalability of the Solution	IBM cloud server will collect all the data and stored in the server. This will more safe and secure

3.4 PROBLEM SOLUTION FIT

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS Children, adult, pet animal, old age people.	6. CUSTOMER CONSTRAINTS CC Spending more time for family, freedom for safety guards near the resources.	5. AVAILABLE SOLUTION AS Swimmers, gain for people how having the resources.	Explore AS, different
	2. JOBS-TO-BE-DONE/PROBLEM J&P If any unwanted action will cause in the underwater using camera detection will analysis the action and recover it.	9. PROBLEM ROOT CAUSE RC People use the cause for feel relax, need more energy for the health for body exercise increase human blood circulation	7. BEHAVIOUR BE Some place we have detection room near the water area, app in mobile phone or other devices	
Identify strong T & E	3. TRIGGERS TR People using the advance technology in some place This increase some other people how doesn't have such features feel to work on that job	10. YOUR SOLUTION SL Camera should be work on any condition With or without the electricity, detection Room and full Protection for camera for Any time even in night also.	8. CHANNELS of BEHAVIOUR CH 8.1 ONLINE Detection should be control by using mobile device	Identify strong T & E
	4. EMOTIONS: BEFORE / AFTER EM Before: people feel unsafe After: they feel comfortable and More secure		8.2 OFFLINE People should wear Any safety Guards or have full trained people while in drowning inside the underwater	

4. REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENT

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Installation	Install the camera inside the underwater, connect necessary app in the phone or other device
FR-2	Detection	Near swimming pool area use detection room for monitor or use IBM cloud for storage purpose of the details
FR-3	Audio	Give the alert signal for the people enter into the underwater and leaving into underwater
FR-4	Support	Extra support from the lifeguard if any person pulse rate will decrease inside the water
FR-5	Prior alert	Extreme level problem should be occurs give the alert signal for the entire pool

4.2 NON-FUNCTIONAL REQUIREMENT

FR No.	Non-Functional Requirement	Description
NFR-1	Usability	A Lifeguard should be present in all the time near pool
NFR-2	Security	Alert message or signal should be give by the lifeguard of swimmer
NFR-3	Reliability	Triggers if any immediate needs of the swimmer inside the pool
NFR-4	Performance	If any unwanted position changes and the pulse rate will decrease this will detect it.

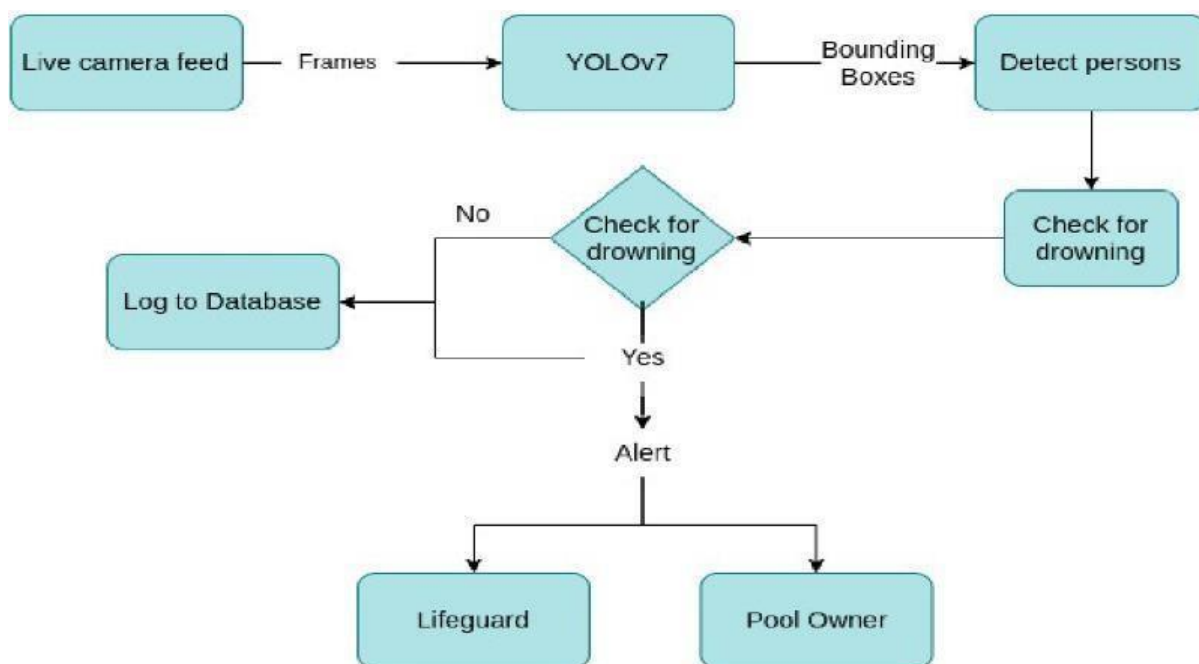
NFR-5	Availability	Equipment and other requirement should bechecked by the lifeguards
NFR-6	Scalability	Virtual eye lifeguard detects potential drownings and it should be notifies you.

5. PROJECT DESIGN

5.1 DATAFLOW DIAGRAMS

Data Flow Diagrams:

A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically. It shows how data enters and leaves the system, what changes the information, and where data is stored

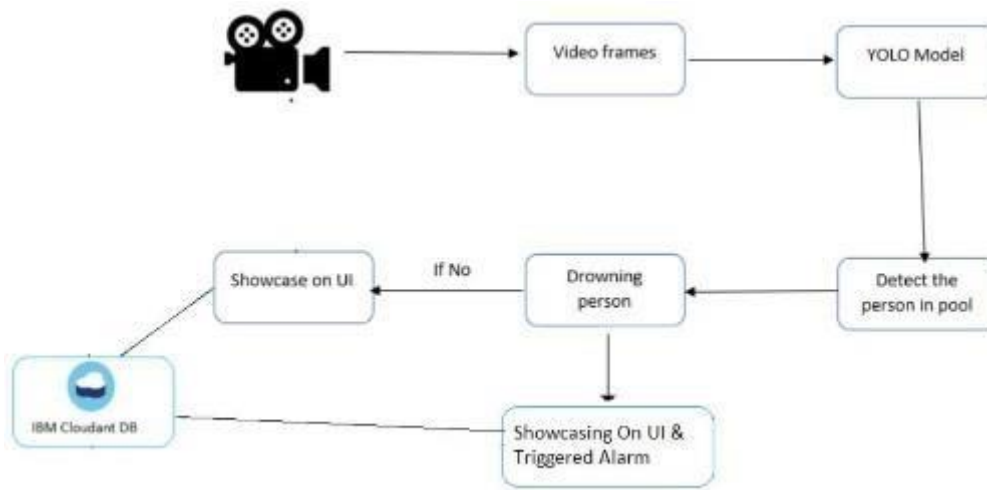


5.2 SOLUTION & TECHNICAL ARCHITECTURE

Solution Architecture:

- ❖ To find underwater movement while person is drowning they have any problem or anything else we will find the solution using the Artificial Intelligence (AI) detection technology.
- ❖ Usually, such systems can be developed by installing more than 16 cameras underwater and ceiling and analyzing the video feeds to detect any anomalies. As a POC we make use of one camera that streams the video underwater and analyses the position of

swimmers to assess the probability of drowning, if it is higher then an alert will be generated to attract lifeguards' attention.



5.3 USER STORIES

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Pool owner)	Installation	USN-1	Install the camera inside the underwater, connect necessary app in the phone or other device	I can cameras to the IBM cloud DB	High	Sprint-1
Customer (Lifeguard)	Secure the people	USN-2	As a user, I can secure the drowning persons from the pool	I can save the drowning person	High	Sprint-1
Customer (swimmers)	safety	USN-3	As a user, I can swim inside the underwater without fear of the	I can swim safely	medium	Sprint-2

			Drowning			
Customer care (Executive)	Contact	USN-4	As a user, I Can resolve if any problem occurs with any device technically	I can contact the customer care executive to resolve any issues	Medium	Sprint-3
Administrator	Dashboard	USN-5	Management of the drowning detection system and database management	I can access the system's logs and any other data instantly	High	Sprint-4

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date(Actual)
Sprint-1	8	6 Days	24 Oct 2022	29 Oct 2022	6	29 Oct 2022
Sprint-2	14	6 Days	31 Oct 2022	05 Nov 2022	12	05 Nov 2022
Sprint-3	16	6 Days	07 Nov 2022	12 Nov 2022	11	12 Nov 2022
Sprint-4	12	6 Days	14 Nov 2022	19 Nov 2022	12	19 Nov 2022

6.2 SPRINT DELIVERY SCHEDULE

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	VLGFSP-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Nivethitha
Sprint-1	Registration	VLGFSP-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Muthukumar
Sprint-1	Registration	VLGFSP-3	As a user, I can register for the	2	Low	Azarudeen

			application through Facebook			
Sprint-1	Registration	VLGFSP-4	As a user, I can register for the application through Gmail	2	Medium	Abinaya
Sprint-1	Login	VLGFSP-6	As a user, I can log into the application by entering email & password	1	High	Nivethitha
Sprint-2	Dataset Collect	VLGFSP-11	Collect number of datasets and get accuracy	2	Medium	Muthukumar
Sprint-2	Pre-processing	VLGFSP-12	The dataset is extracted	2	High	Azarudeen
Sprint-2	Train the model	VLGFSP-13	Train the model.	4	High	Abinaya
Sprint-2	Test the model	VLGFSP-14	Test the model	6	High	Nivethitha
Sprint-3	Detection	VLGFSP-15	Load the trained model.	3	High	Muthukumar
Sprint-3	Detection	VLGFSP-16	Identify the person by collecting real-time data through a webcam.	5	Medium	Azarudeen
Sprint-3	Detection	VLGFSP-16	classify it by using a trained model to predict the output	8	High	Abinaya
Sprint-4	Detection	VLGFSP-17	If person is drowning, the system will ring an alarm to give signal	7	High	Nivethitha
Sprint-4	Detection	VLGFSP-18	As a User, I can detect the drowning person.	3	Medium	Muthukumar
Sprint-4	Logout	VLGFSP-19	As a User, I can logout the application.	2	Low	Azarudeen

6.3 REPORT FROM JIRA Backlog (scrum)

The screenshot shows the Jira Backlog for the project 'VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning'. The backlog is organized into three sprints:

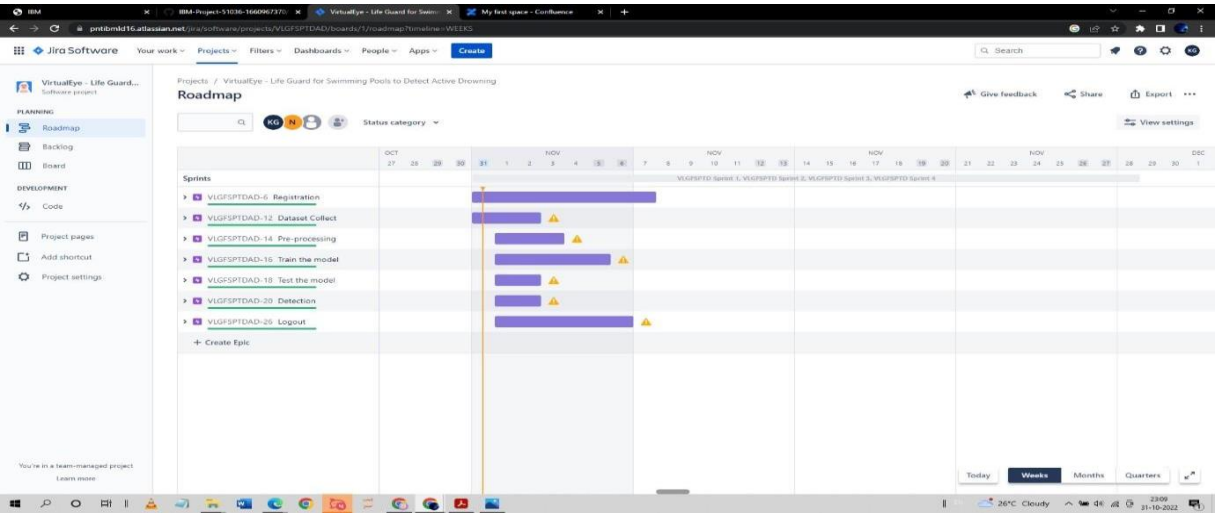
- VLGFSPTD Sprint 2** (7 Nov - 14 Nov, 4 issues):
 - VLGFSPTDAD-13: Collect number of datasets and get accuracy (DATASET COLLECT) - 2 TO DO
 - VLGFSPTDAD-15: The dataset is extracted (PRE-PROCESSING) - 2 TO DO
 - VLGFSPTDAD-17: Train the model (TRAIN THE MODEL) - 4 TO DO
 - VLGFSPTDAD-19: Test the model (TEST THE MODEL) - 6 TO DO
- VLGFSPTD Sprint 3** (14 Nov - 21 Nov, 3 issues):
 - VLGFSPTDAD-21: Load the trained model (DETECTION) - 3 TO DO
 - VLGFSPTDAD-22: Identify the person by collecting real-time data through a webcam (DETECTION) - 5 TO DO
 - VLGFSPTDAD-23: classify it by using a trained model to predict the output (DETECTION) - 8 TO DO
- VLGFSPTD Sprint 4** (21 Nov - 28 Nov, 3 issues):
 - VLGFSPTDAD-24: If person is drowning, the system will ring an alarm to give signal (DETECTION) - 7 TO DO
 - VLGFSPTDAD-25: As a User, I can detect the drowning person (DETECTION) - 3 TO DO
 - VLGFSPTDAD-30: As a User, I can logout the application. (LOGOUT) - 2 TO DO

The interface includes a sidebar with navigation options (Roadmap, Backlog, Board, Code, Project pages, Add shortcut, Project settings) and a top navigation bar with filters and search options.

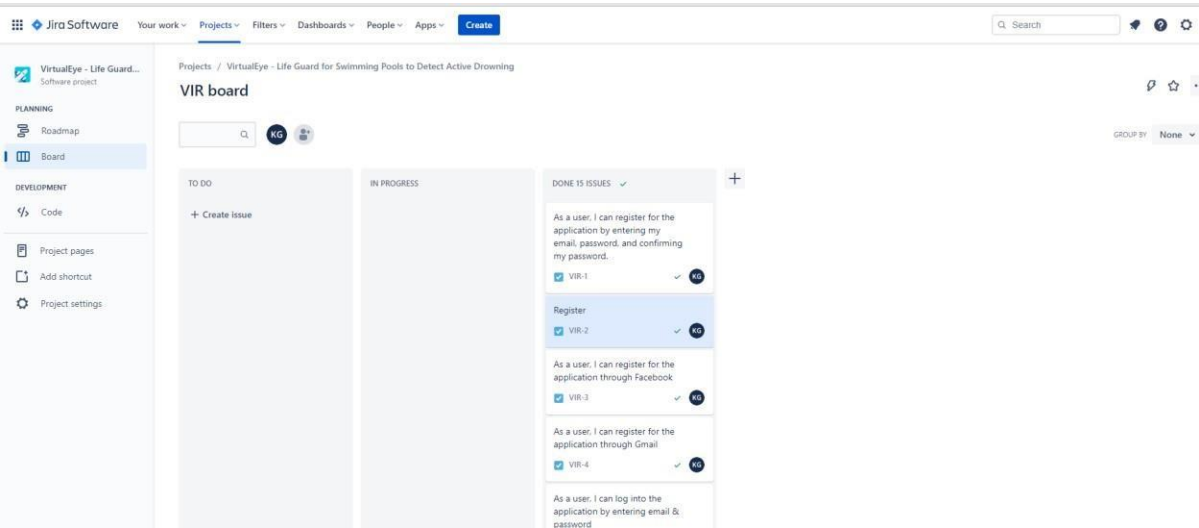
Roadmap

The screenshot shows the Jira Roadmap for the project 'VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning'. The roadmap is displayed as a timeline from October to February 2023. A single sprint, 'VLGFSPTDAD-6 Registration', is visible, spanning from late October to early November. The interface includes a sidebar with navigation options (Roadmap, Backlog, Board, Code, Project pages, Add shortcut, Project settings) and a top navigation bar with filters and search options. The bottom of the screen shows a status bar with weather information and a 'Quickstart' button.

Chart



Board (Kanban)



7. CODING & SOLUTION

7.1 FEATURE 1

```
[net]
# Testing# batch=1 #
subdivisions=1#
Training  batch=64
subdivisions=16
width=608
height=608
channels=3
momentum=0.9
decay=0.0005
angle=0 saturation =
1.5
exposure      =
1.5hue=.1

learning_rate=0.01
burn_in=1000 max_batches
=      500200policy=steps
steps=400000,450000
scales=.1,.1

[convolutional]
batch_normalize
=1      filters=32
size=3
stride=1
pad=1
activation=le
aky

# Downsample
```

```
[convolutional]
batch_normalize
=1 filters=64
size=3
stride=2
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize
=1 filters=32
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize
=1 filters=64
size=3 stride=1
pad=1
activation=leaky
[shortcut]from=
- 3
activation=linear#
```

Downsample

```
[convolutional]
batch_normalize
=1 filters=128
size=3
stride=2
pad=1
```

activation=leaky

[convolutional]

batch_normaliz

e=1 filters=64

size=1 stride=1

pad=1

activation=leaky

[convolutional]

batch_normaliz

e=1 filters=128

size=3 stride=1

pad=1

activation=leaky

[shortcut]from

=- 3

activation=line

ar

[convolutional]

batch_normalize

=1 filters=64

size=1

stride=1

pad=1

activation=leaky

[convolutional]

batch_normalize

=1 filters=128

size=3

stride=1

pad=1

activation=leaky

```
[shortcut]from  
=-          3  
activation=line  
ar
```

```
# Downsample
```

```
[convolutional]  
batch_normalize  
=1  
filters=256size=3  
stride=2  pad=1  
activation=leaky
```

```
[convolutional]  
batch_normaliz  
e=1  filters=128  
size=1  stride=1  
pad=1  
activation=leaky
```

```
[convolutional]  
batch_normaliz  
e=1  filters=256  
size=3  stride=1  
pad=1  
activation=leaky
```

```
[shortcut]from  
=-          3  
activation=line  
ar
```

```
[convolutional]  
batch_normaliz  
e=1  filters=128
```

size=1 stride=1
pad=1
activation=leaky

[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=256
size=3 stride=1
pad=1
activation=leaky

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normalize
=1 filters=128
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=256
size=3
stride=1
pad=1
activation=leaky

[shortcut]from
=- 3
activation=linear

[convolutional]
batch_normalize
=1 filters=128
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=256
size=3 stride=1
pad=1

activation=leaky

[shortcut]from

=- 3

activation=line

ar

[convolutional]

batch_normalize

=1 filters=128

size=1

stride=1

pad=1

activation=le

aky

[convolutional]

batch_normalize

=1 filters=256

size=3 stride=1

pad=1

activation=leaky

[shortcut]from

=- 3

activation=line

ar

[convolutional]

batch_normaliz

e=1 filters=128

size=1 stride=1

pad=1

activation=leaky


```
[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from
=- 3
activation=line
ar
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leaky
```

```
[shortcut]from
=- 3
activation=line
ar#
```

Downsample

```
[convolutional]
batch_normalize
```

=1 filters=512
size=3 stride=2
pad=1 activation=leaky

[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leaky

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normalize
=1 filters=256
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz
e=1 filters=512

size=3 stride=1
pad=1
activation=leak
y

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=512
size=3 stride=1
pad=1 activation=leaky

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normalize
=1 filters=256
size=1
stride=1
pad=1
activation=le
aky

```
[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leaky
```

```
[shortcut]from
=- 3
activation=line
ar
```

```
[convolutional]
batch_normalize
=1 filters=256
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leaky
```

```
[shortcut]from
=- 3
activation=line
ar
```

```
[convolutional]
batch_normalize
```

=1 filters=256
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=512
size=3 stride=1
pad=1
activation=leaky

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normalize
=1 filters=256
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=512
size=3 stride=1
pad=1
activation=leaky

[shortcut]from
=- 3

activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize
=1 filters=512
size=3
stride=1
pad=1
activation=leaky

[shortcut]from
=- 3
activation=line
ar#

Downsample

[convolutional]
batch_normalize
=1 filters=1024
size=3 stride=2
pad=1
activation=leaky

[convolutional]
batch_normaliz
e=1 filters=512
size=1 stride=1
pad=1

activation=leaky

[convolutional]

batch_normalize

=1 filters=1024

size=3

stride=1

pad=1

activation=le

aky

[shortcut]from

=- 3

activation=line

ar

[convolutional]

batch_normalize

=1 filters=512

size=1

stride=1

pad=1

activation=le

aky

[convolutional]

batch_normaliz

e=1

filters=1024

size=3 stride=1

pad=1

activation=leaky

[shortcut]from

=- 3

activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=512
size=1 stride=1
pad=1
activation=leaky

[convolutional]
batch_normaliz
e=1
filters=1024
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from
=- 3
activation=line
ar

[convolutional]
batch_normalize
=1 filters=512
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normaliz
e=1
filters=1024


```
size=3 stride=1
pad=1
activation=leaky
```

```
[shortcut]from
=- 3
activation=line
ar
```

```
#####
```

```
[convolutional]
batch_normaliz
e=1 filters=512
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normalize=1size
=3 stride=1 pad=1
filters=1024
activation=leaky
```

```
[convolutional]
batch_normalize
=1 filters=512
size=1
stride=1
pad=1
activation=leaky
```

```
[convolutional]
batch_normalize=1size
=3 stride=1 pad=1
```

filters=1024
activation=leaky

[convolutional]
batch_normalize
=1 filters=512
size=1
stride=1
pad=1
activation=leaky

[convolutional]
batch_normalize=1 size
=3 stride=1 pad=1
filters=1024
activation=leaky

[convolutional]si
ze=1 stride=1
pad=1
filters=255
activation=linear

[yolo]
mask =
6,7,8
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,
156,198,
373,326
classes=80
num=9
jitter=.3
ignore_thresh
= .7
truth_thresh = 1 random=1

[route] layers = -4

[convolutional]

batch_normaliz

e=1 filters=256

size=1 stride=1

pad=1

activation=leaky

[upsample]

stride=2

[route]

layers = -1,

61

[convolutional]

batch_normaliz

e=1 filters=256

size=1 stride=1

pad=1

activation=leaky

[convolutional]

batch_normalize=1 siz

e=3 stride=1 pad=1

filters=512

activation=leaky

[convolutional]

batch_normaliz

e=1 filters=256

size=1 stride=1

pad=1

activation=leak

y

[convolutional]

batch_normalize=1 size

kernel=3 stride=1 pad=1

filters=512

activation=leaky

[convolutional]

batch_normalize

kernel=1 filters=256

size=1 stride=1

pad=1

activation=leaky

[convolutional]

batch_normalize=1 size

kernel=3 stride=1 pad=1

filters=512

activation=leaky

[convolutional]si

kernel=1 stride=1

pad=1

filters=255

activation=linear

[yolo]

mask =

3,4,5

anchors =

10,13,

16,30,

33,23,

30,61,

```
62,45,  
59,119,  
116,90,  
156,198,  
373,326  
classes=80  
num=9  
jitter=.3  
ignore_thresh  
= .7  
truth_thresh = 1 random=1
```

```
[route] layers = -4
```

```
[convolutional]  
batch_normaliz  
e=1 filters=128  
size=1 stride=1  
pad=1  
activation=leaky
```

```
[upsample]  
stride=2
```

```
[route]  
layers = -1,  
36
```

```
[convolutional]  
batch_normalize  
=1 filters=128  
size=1  
stride=1  
pad=1
```

activation=leaky

[convolutional]

batch_normalize=1 size

in=3 stride=1 pad=1

filters=256

activation=leaky

[convolutional]

batch_normalize

in=1 filters=128

size=1

stride=1

pad=1

activation=leaky

[convolutional]

batch_normalize=1 size=3

stride=1

pad=1

filters=256

activation=leak

y

[convolutional]

batch_normaliz

in=1 filters=128

size=1

stride=1

pad=1

activation=leak

y

[convolutional]

batch_normalize=1 size

in=3 stride=1 pad=1

```
filters=256
```

```
activation=leaky
```

```
[convolutional]siz
```

```
e=1      stride=1
```

```
pad=1 filters=255
```

```
activation=linea
```

```
r
```

```
[yolo]
```

```
mask    =
```

```
0,1,2
```

```
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,
```

```
156,198,
```

```
373,326
```

```
classes=80
```

```
num=9
```

```
jitter=.3
```

```
ignore_thresh
```

```
= .7
```

```
truth_thresh =
```

```
1 random=1
```

7.2 FEATURE 2

```
#import          necessary
```

```
packagesimport cv2
```

```
import os
```

```
import numpy as np
```

```
from .utils import download_file
```

```
initialize      =
```

```
Truenet = None
```

```
dest_dir = os.path.expanduser('~') + os.path.sep + '.cvlib' + os.path.sep +  
'object_detection' + os.path.sep + 'yolo' + os.path.sep + 'yolov3'
```

```
classes = None
```

```
#colors are BGR instead of RGB in python
```

```
COLORS = [0,0,255], [255,0,0]
```

```
def populate_class_labels():
```

```
    #we are using a pre existent classifier which is more reliable and more efficient than  
    one#we could make using only a laptop
```

```
    #The classifier should be downloaded automatically when you run this  
    scriptclass_file_name =
```

```
    'yolov3_classes.txt'    class_file_abs_path    =  
    dest_dir + os.path.sep + class_file_name
```

```
    url = 'https://github.com/Nico31415/Drowning-Detector/raw/master/yolov3.txt'if  
    not os.path.exists(class_file_abs_path):
```

```
        download_file(url=url, file_name=class_file_name, dest_dir=dest_dir)f =  
        open(class_file_abs_path, 'r')
```

```
        classes = [line.strip() for line in f.readlines()]
```

```
    return classes
```

```
def get_output_layers(net)
```

```
#the number of output layers in a neural network is the number of  
possible#things the network can detect, such as a person, a dog, a tie, a  
phone... layer_names = net.getLayerNames()
```

```
output_layers = [layer_names[i[0] - 1] for i in net.getUnconnectedOutLayers()]
```



```
return output_layers
```

```
def draw_bbox(img, bbox, labels, confidence, Drowning, write_conf=False):
```

```
    global COLORS
```

```
    global classes
```

```
    if classes is None:
```

```
        classes =  
        populate_class_labels()
```

```
    for i, label in enumerate(labels):
```

```
        #if the person is drowning, the box will be drawn red instead  
        of blueif label == 'person' and Drowning: color =  
        COLORS[0] label  
        = 'DROWNING'
```

```
    else:
```

```
        color = COLORS[1]
```

```
    if write_conf:
```

```
        label += ' ' + str(format(confidence[i] * 100, '.2f')) + '%'
```

```
    #you only need to points (the opposite corners) to draw a rectangle. These  
    points#are stored in the variable bbox
```

```
    cv2.rectangle(img, (bbox[i][0],bbox[i][1]), (bbox[i][2],bbox[i][3]), color, 2)
```

```
cv2.putText(img, label, (bbox[i][0],bbox[i][1]-10),  
cv2.FONT_HERSHEY_SIMPLEX, 0.5, color, 2)
```

```
return img
```

```
def detect_common_objects(image, confidence=0.5, nms_thresh=0.3):
```

```
Height, Width = image.shape[:2]scale =  
0.00392
```

```
global classes  
global dest_dir
```

```
#all the weights and the neural network algorithm are already preconfigured#as we  
are using
```

```
YOLO
```

```
#this part of the script just downloads the YOLO files
```

```
config_file_name = 'yolov3.cfg'
```

```
config_file_abs_path = dest_dir + os.path.sep + config_file_name
```

```
weights_file_name = 'yolov3.weights'
```

```
weights_file_abs_path = dest_dir + os.path.sep + weights_file_name
```

```
url = 'https://github.com/Nico31415/Drowning-Detector/raw/master/yolov3.cfg'
```

```
if not os.path.exists(config_file_abs_path):
```

```
download_file(url=url, file_name=config_file_name, dest_dir=dest_dir)
url = 'https://pjreddie.com/media/files/yolov3.weights'
```

```
if not os.path.exists(weights_file_abs_path):
```

```
download_file(url=url, file_name=weights_file_name,
dest_dir=dest_dir)
```

```
global initialize
global net
```

```
if initialize:
```

```
classes =
    populate_class_labels()
```

```
net = cv2.dnn.readNet(weights_file_abs_path,
config_file_abs_path)initialize = False
```

```
blob = cv2.dnn.blobFromImage(image, scale, (416,416), (0,0,0), True, crop=False)
```

```
net.setInput(blob)
```

```
outs = net.forward(get_output_layers(net))
```

```
class_ids =
[]
confidences
```

```
= [] boxes =
```

```
[]
```

```
for out in outs:
```

```
    for detection in out: scores =
```

```
        detection[5:] class_id =
```

```
        np.argmax(scores)
```

```
        max_conf            =
```

```
        scores[class_id]    if
```

```
        max_conf            >
```

```
        confidence:
```

```

center_x = int(detection[0] * Width)
center_y = int(detection[1] * Height)w =
int(detection[2] * Width) h =
int(detection[3] * Height)x = center_x - w
/ 2 y = center_y - h / 2
class_ids.append(class_id)
confidences.append(float(max_conf))boxes.append([x, y, w, h])

```

```

indices = cv2.dnn.NMSBoxes(boxes, confidences, confidence, nms_thresh)

```

```

bbox = []label = []conf =

```

```

[]

```

```

for i in indices:

```

```

    i = i[0]

```

```

    box = boxes[i]x = box[0] y = box[1] w = box[2]h =
    box[3] bbox.append([round(x), round(y),
    round(x+w), round(y+h)])
    label.append(str(classes[class_ids[i]]))
    conf.append(confidences[i])

```

```

return bbox, label, conf

```

8. TESTING

8.1 TEST CASES

Test case ID	Feature Type		Test Scenario	Steps TO Execute	Test	Expected Result	Actual Result
LoginPage_TC_001	Functional	Home Page	Verify user is able to see the Login/Signup popup when user clicked on My account button	1.Enter URL and click go 2.Click on My Account dropdown button 3.Verify login/Signup popup displayed or not	Login.html	Login/Signup popup should display	Working as
LoginPage_TC_002		Home Page	Verify the UI elements in Login/Signup popup	1.Enter URL and click go 2.Click on My Account dropdown 3.Verify login/Signup popup with below UI elements: a.email text box b.password text box c.Login button d.New customer? Create account link e. Last password? Recovery password link	Login.html	Application should show below elements: a.email text box b.password text box c.Login button with orange colour d. New customer? Create account link e.Last password? Recovery password link	Working as expected
LoginPage_TC_003	Functional	Home page	Verify user is able to log into application with Valid credentials	1.Enter URL and click go 2.Click on My Account dropdown 3.Enter Valid username/email in Email text box 4.Enter valid password in password text box 5. Click On in button	Username:lax@gmail password:lax26	User should navigate to prediction homepage	working as
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with Invalid credentials	1. Enter URL and click go 2.Click on My Account dropdown button 3.Enter Invalid username/email in Email text box 4.Enter valid password in password text box 5.Click on in button	Username:lax password:lax26	Application should show 'incorrect email or password' validation message.	working as
LoginPage_TC_004	Functional	Login page	Verify user is able to log into application with Invalid credentials	1-Enter URL and click go 2.Click On My Account dropdown 3.Enter Valid username/email in Email text box 4.Enter Invalid password in password text box 5.Click on in button	username:lax26@mail password:lax26	Application should show 'incorrect email or password' validation message.	working as
LoginPage_TC_005	Functional	Login page	Verify user is able to into application with Invalid credentials	1.Enter URL and click go 2.Click on My Account dropdown 3.Enter Invalid username/email in Email text box 4. Enter Invalid password in password text box 5. Click on in button	username:lax26@mail password:1803	Application should show 'incorrect email or password' validation message.	working as
Predictionpage_TC_006	Functional	Prediction Page	Page should display whether the person is drowning or not	1. Camera should take pictures of people swimming in pools. 2. It should predict the probability of drowning. 3. It should show a bounding box displaying the probability Of drowning	image Of people drowning	generate a alert to lifeguard if people are drowning	Working as

8.2 USER ACCEPTANCE TESTING

Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [ProductName] project at the time of the release to User Acceptance Testing (UAT).

Defect Analysis

This report shows the number of resolved or closed bugs at each severity level, and howthey were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	7	3	1	2	13
Duplicate	1	0	2	0	3
External	2	3	0	1	6
Fixed	10	2	4	10	26
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	24	14	13	26	77

Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	1	0	0	41
Security	42	0	0	42
Outsource Shipping	3	0	0	3

Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULT

9.1 PERFORMANCE METRICS

```
<html lang="en">
```

```
<head>
```

```
  <meta charset="UTF-8">
```

```
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
```

```
  <title>High Quality Facial Recognition</title>
```

```
  <link href="https://cdn.bootcss.com/bootstrap/4.0.0/css/bootstrap.min.css" rel="stylesheet">
```

```
  <script src="https://cdn.bootcss.com/popper.js/1.12.9/umd/popper.min.js">
```

```
</script>
```

```
  <script src="https://cdn.bootcss.com/jquery/3.3.1/jquery.min.js">
```

```
</script>
```

```
  <script src="https://cdn.bootcss.com/bootstrap/4.0.0/js/bootstrap.min.js">
```

```

</script>
<link href="{ { url_for('static', filename='css/main.css') } }"rel="stylesheet">
    <style>
        .bg-dark { background-color: #42678c!important;
        }
        #result { color: #0a1c4ed1;
        }
    </style>
</head>

<body    style="background-color:black";>    <header
id="head" class="header">
    <section id="navbar">
        <h1 class="nav-heading"><i>Virtual Eye</i></h1>
        <div class="nav--items">
            <ul>
                <li><a href="{ { url_for('index') } }">Home</a></li>
                <li><a
href="{ { url_for('logout') } }">Logout</a></li>
                <!-- <li><a href="#about">About</a></li>
                <li><a href="#services">Services</a></li> -->

            </ul>
        </div>
    </section>

```

```

</header>
<div class="container">
  <div id="content" style="margin-top:2em">
    <div class="container"> <div
      class="row">
<div class="col-sm-6 bd" >
    <h2><em style="color:white;">High Quality Facial
Recognition</em></h2>
    <br>
    <p><h5><i style="color:white;">Emotion Detection Through Facial Feature
Recognition</i></h5></p>
    
    </div>
    <div class="col-sm-6">
      <div>
        <h4 style="color:white;">Upload
Image Here</h4>
<form action = "http://localhost:5000/" id="upload-file"
method="post" enctype="multipart/form-data">
        <label for="imageUpload" class="upload-
label">
          Choose Image
        </label>

```

```
<input type="file" name="image"
id="imageUpload" accept=".png, .jpg, .jpeg, .pdf">
</form>
```

```
<div class="image-section" style="display:none;">
<div class="img-preview">
  <div id="imagePreview">

</div>

</div>
<div>
<button type="button" class="btn btn-info btn-lg "
id="btn-predict">Analyse</button>
</div>

</div>
```

```
<div class="loader" style="display:none;"></div>
<h3>
  <span id="result"> </span>
</h3>

</div>
</div>
```

```
        </div>
    </div>
</div>
</body>
```

```
<footer>
    <script src="{ { url_for('static', filename='js/main.js') }}"
type="text/javascript"></script>
</footer>

</html>
```

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!--Bootstrap -->
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/boo
tstrap.min.css"
integrity="sha384-
```

Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGg

FAW/dAiS6JXm" crossorigin="anonymous">

<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpG

FF93hXpG5KkN" crossorigin="anonymous"></script>

<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-

ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPsk

vXusvfa0b4Q" crossorigin="anonymous"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js" integrity="sha384-

JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5

+76PVCmYI" crossorigin="anonymous"></script>

<script src="https://kit.fontawesome.com/8b9cdc2059.js"

crossorigin="anonymous"></script>

<link href="https://fonts.googleapis.com/css2?family=Akronim&family=

Roboto&display=swap" rel="stylesheet">

<link rel="stylesheet" href="../static/style.css">

<!-- <script defer src="../static/js/main.js"></script> --> <title>Virtual Eye</title>

</head>

<body>

<header id="head" class="header">

<section id="navbar">

<h1 class="nav-heading"><i>Virtual Eye</i></h1>

```

<div class="nav--items">
    <ul>
        <li><a
href="{{ url_for('index')}}">Home</a></li>
        <li><a
href="{{ url_for('login')}}">Login</a></li>
        <li><a
href="{{ url_for('register')}}">Register</a></li>
        <li><a href="{{ url_for('login')}}">Demo</a></li> </ul>
    </div>
</section>
<section id="slider">
<div id="carouselExampleIndicators" class="carousel" data-ride="carousel">
    <ol class="carousel-indicators ">
        <li data-target="#carouselExampleIndicators" data-slide-    to="0"    class="active
"></li>
        <li data-target="#carouselExampleIndicators" data-slide-to="1"></li> <li data-
target="#carouselExampleIndicators" data-slide-to="2"></li>
    </ol>
    <div class="carousel-inner">
        <div class="carousel-item active">

        </div>
        <div class="carousel-item">

    </div>
    <div class="carousel-item">
         </div>
    </div>
    <a class="carousel-control-prev" href="#carouselExampleIndicators" role="button" data-
slide="prev">
        <span class="carousel-control-prev-icon" aria-hidden="true"></span>
        <span class="sr-only">Previous</span> </a>
    <a class="carousel-control-next" href="#carouselExampleIndicators" role="button" data-
slide="next">
        <span class="carousel-control-next-icon" aria-hidden="true"></span>
        <span class="sr-only">Next</span>
</a>
</div>

</section>
</header>
<section id="about">
    <div class="top">
        <h3 class="title text-muted"> ABOUT
        PROJECT
        </h3>
        <div class="line"></div>
    </div>
<div class="body">
```


<div class="left">

<h2>Problem:</h2>

<p>

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in the hotels, weekend tourist spots and barely people have in their house backyard. Beginners, especially often feel it difficult to breathe under water and causes breathing trouble which in turn cause a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide..Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly.

</p>

</div>

<div class="left">

<h2>Solution:</h2>

<p>

To overcome the conflict, a meticulous system is to be implemented along the swimming pools to save the human life. By studying body movement patterns and connecting cameras to an artificial intelligence (AI) system we can devise an underwater pool safety system that reduces the risk of drowning. Usually such systems can be developed by installing more than 16 cameras underwater and ceiling and analysing the video feeds to detect any anomalies . but AS a POC we make use of one camera that streams the video underwater and analyses the position of swimmers to assess the probability of drowning ,if it is higher than an alert will be generated to attract lifeguards attention.

</p>

</div>

</div>

<div class="bottom">

<p>

Note : The system is not designed to replace a lifeguard or other human monitor, but to act as an additional tool. It helps the lifeguard to detect the underwater situation where they can't easily observe.

</p>

</div>

</section>

<section id="footer">

<p>Copyright © 2022. All Rights Reserved</p>

<div class="social">

<i class="fab fa-2x fa-twitter-square"></i>

<i class="fab fa-2x fa-linkedin"></i>

<i class="#"></i>

</div>

</section>

</body>

</html>

Logout.html

```
<!DOCTYPE html>
```

```
<html >
```

```
<head>
```

```
    <meta charset="UTF-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1">    <title>Virtual  
    Eye</title>
```

```
    <link href='https://fonts.googleapis.com/css?family=Pacifico' rel='stylesheet'  
type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Arimo' rel='stylesheet' type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Hind:300' rel='stylesheet' type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300' rel='stylesheet'  
type='text/css'>
```

```
<link href='https://fonts.googleapis.com/css?family=Merriweather' rel='stylesheet'>
```

```
<link href='https://fonts.googleapis.com/css?family=Josefin Sans' rel='stylesheet'>
```

```
<link href='https://fonts.googleapis.com/css?family=Montserrat' rel='stylesheet'>
```

```
<style> .header {
```

```
    top:0; margin:0px;
```

```
    left: 0px; right: 0px;
```

```
    position: fixed;
```

```
    background-color:    #28272c;
```

```
        color: white; box-shadow: 0px 8px
        4px grey;
    overflow: hidden; padding-left:20px; font-
    family: 'Josefin Sans';
        font-size: 2vw;
        width: 100%;
        height:8%; text-
        align: center;
    }
    .topnav { overflow: hidden;
background-color: #333;
}
```

```
.topnav-right a { float: left;
color:
#f2f2f2;
text-align: center; padding: 14px
16px; text-decoration: none; font-
size: 18px;
}
```

```
.topnav-right a:hover { background-color: #ddd;
color: black;
}
```

```
.topnav-right a.active { background-color: #565961;
```

```
        color: white;
    }
}
```

```
.topnav-right {    float:
    right;         padding-
    right:100px;
}
```

```
.login{
margin-top:-70px;
} body
{
```

```
    background-color:#ffffff;    background-repeat:
        no-repeat;    background-size:cover;
    background-position: 0px 0px;
}
```

```
.main{
    margin-top:100px; text-align:center;
} form { margin-left:400px;margin-right:400px;}
```

```
input[type=text],input[type=email],input[type=number],input[type=password] { width: 100%;
padding: 12px 20px; display: inline-block; margin-bottom:18px; border: 1px solid #ccc;
    box-sizing: border-box;
}
```

```
button { background-color: #28272c;
  color: white; padding:
  14px 20px; margin-
  bottom:8px; border: none;
  cursor:
  pointer; width: 20%;
}
```

```
button:hover { opacity: 0.8;
}
```

```
.cancelbtn { width: auto;
  padding: 10px 18px;
  background-color: #f44336;
}
```

```
.imgcontainer { text-align:
  center; margin: 24px 0
  12px 0;
}
```

```
img.avatar {
  width: 30%; border-radius:
  50%;
```

```

}

.container      { padding: 16px;
}

span.psw       { float:
right; padding-top:
16px;

}

/* Change styles for span and cancel button on extra small screens */
@media screen and (max-width: 300px) { span.psw {
display: block; float:
none;
}
.cancelbtn     { width:
100%;
}
}

</style>
</head>

<body style="font-family:Montserrat;">

```

```

<div class="header">
  <div style="width:50%;float:left;font-size:2vw;text-align:left;color:white; padding-top:1%">Virtual eye</div>
    <div class="topnav-right" style="padding-top:0.5%;">

      <a href="{{ url_for('home')}}">Home</a>
      <a href="{{ url_for('login')}}">Login</a>
      <a href="{{ url_for('register')}}">Register</a> </div>
    </div>
  <div class="main">
    <h1>Successfully Logged Out!</h1>
    <h3 style="color:#4CAF50">Login for more information</h3>

    <a href="{{ url_for('login')}}"><button
type="submit">Login</button></a>
  </div>

</body>
</html>

```

Prediction.html

```

<!DOCTYPE html>
<html lang="en">
<head>

```



```
<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0"> <!--Bootstrap -->
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/boo
tstrap.min.css" integrity="sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGg
FAW/dAiS6JXm" crossorigin="anonymous">
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpG
FF93hXpG5KkN" crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/ popper.min.js"
integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPsk
vXusvfa0b4Q" crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootst rap.min.js"
integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5
+76PVCmYl" crossorigin="anonymous"></script>

<script src="https://kit.fontawesome.com/8b9cdc2059.js"
crossorigin="anonymous"></script>
<link href="https://fonts.googleapis.com/css2?family=Akronim&family=
Roboto&display=swap" rel="stylesheet">
<link rel="stylesheet" href="../static/style.css">
```

```

<script defer src="../static/js/JScript.js"></script>
<title>Prediction</title>
</head> <body>
  <header id="head" class="header"> <section
    id="navbar">
    <h1 class="nav-heading"></i>Virtual Eye</h1>
    <div class="nav--items">
      <ul>
        <li><a href="{ { url_for('index') } }">Home</a></li>
        <li><a
href="{ { url_for('logout') } }">Logout</a></li>
        <!-- <li><a href="#about">About</a></li>
        <li><a href="#services">Services</a></li> -->

      </ul>
    </div>
  </section>
  </header>
  <!-- dataset/Training/metal/metal326.jpg -->
  </br>
  <section id="prediction">
<h2 class="title text-muted">Virtual Eye- Life Guard forSwimming Pools to
Detect Active Drowning</h1>
    <div class="line" style="width: 900px;"></div>
    </section>

```

</br>

<section id="about">

<div class="body">

<div class="left">

<p>

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in the hotels, weekend tourist spots and barely people have in their house backyard. Beginners, especially often feel it difficult to breathe under water and causes breathing trouble which in turn cause a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide..Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly.

</p>

</div>

<div class="left">

<div class="prediction-input">

 </br>

<form id="form" action="/result" method="post"

enctype="multipart/form-data">

<input type="submit" class="submitbtn" value="ClickMe! For a
Demo">

</form>

</div>

<h5 style="text-color:Red">

<b style="text-color:Red">{{ prediction }} </h5>

</div>

</div>

</section>

</br></br>

<section id="footer">

<p>Copyright Â© 2021. All Rights Reserved</p>

</section>

</body>

</html>



Enter registered email ID

Enter Password

Login



ABOUT PROJECT

Problem:

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in the hotels, weekend tourist spots and barely people have in their house backyard. Beginners, especially often feel it difficult to breathe under water and causes breathing trouble which in turn cause a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children.

Solution:

To overcome the conflict, a meticulous system is to be implemented along the swimming pools to save the human life. By studying body movement patterns and connecting cameras to an artificial intelligence (AI) system we can devise an underwater pool safety system that reduces the risk of drowning. Usually such systems can be developed by installing more than 16 cameras underwater and ceiling and analysing the video feeds to detect any anomalies.

Virtual Eye- Life Guard for Swimming Pools to Detect Active Drowning

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in the hotels, weekend tourist spots and barely people have in their house backyard. Beginners, especially often feel it difficult to breathe under water and causes breathing trouble which in turn cause a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly.

[Click Me! For a Demo](#)

Virtual Eye- Life Guard for Swimming Pools to Detect Active Drowning

Swimming is one of the best exercises that helps people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in the hotels, weekend tourist spots and barely people have in their house backyard. Beginners, especially often feel it difficult to breathe under water and causes breathing trouble which in turn cause a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their age are found to be suffering the highest drowning mortality rates worldwide. Such kinds of deaths account for the third cause of unplanned death globally, with about 1.2 million cases yearly.



[Click Me! For a Demo](#)

Emergency !!! The Person is drowning

10. ADVANTAGES & DISADVANTAGES

***ADVANTAGES**

- a. Improve your communication skills.
- b. Increase your self-confidence.
- c. Develop sought-after skills.
- d. Make new friends.
- e. Fancy a challenge?
- f. Swimming Improves Social Well Being.
- g. Kids Who Swim Become Active Adults.
- h. Swimming Makes You Smarter.
- i. Swimming Teaches Team-Building Skills.
- j. Swimming Burns More Calories than Jogging.

Swimming Slows Down Aging.

***DISADVANTAGE**

- a. Increased exposure to chlorine.
- b. You'll get sick of your pool. Most likely, the pool where you practice is going to be the pool where you work.
- c. Intensive training is required.
- d. Indoor pools are stuffy.
- e. There's a higher risk of dehydration.
- f. Practice will seem longer.
- g. Same place.

11. CONCLUSION

This section will draw from three core documents: ISO_20380, HSG179, and the recently published German guideline, DGfDB R 94.15. A summary of each is given, outlining the key messages they disseminate and what this means for those involved with DDS.

ISO_20380 This document focuses on the requirements for the installation, operation, maintenance and performance of DDS, the testing methods, and the information required from the supplier in the operating manual. These international standards do not apply to systems used in domestic pools or pools smaller than 150m².

Prior to the installation of any DDS, 'a technical study shall be carried out by the supplier in consultation with or based on information provided by the swimming pool's owner/operator'. This is to establish the quantity and positioning of the equipment making up the system such as cameras, central processing unit, alarm tools, and other related equipment. The technical study must also provide a technical drawing of the pool basin, showing areas of 'coverage' and 'non-coverage', as well as the minimum lighting levels required above and below the water surface for the DDS to operate within performance requirements. To carry out the study, a list of factors to consider are given, outlining the variables that make each pool unique such as the architecture, and alarm reception coverage area of mobile devices to be used with the system. With this information all in one document, the technical study can be used to help optimise performance of the system, and forms part of the contract between the supplier and the pool operator. The next area of the standard is the performance requirements. This outlines the requirements needed to pass the regular maintenance testing and performance requirements for normal operation. This section covers the alarm set off time for operational performance, which is to be 15 seconds or less and displayed on the system interface. It also states that the alarm set off time must be built-in and shall not be changeable by staff. The section also discusses the areas covered by the DDS and highlights that each trained staff member must be aware of these areas. Another

coverage-related requirement is that the DDS must be able to temporarily create areas where detection is disabled, to manage specific activities such as rescue drills.

12.FUTURE SCOPE

THREE ELEMENTS OF FOCUS FOR FUTURE LIFEGUARDS.

"As the Aquatics Industry focuses on the future, there are three elements that are critical to success of Lifeguards"

The Aquatics industry has changed quite a bit since its early inception. Lifeguarding has changed. The technology has changed, the culture has changed, and the standards have changed. What has not changed is an industry full of people who want to make their mark in the field. The future of lifeguards in the aquatics industry lies in development of these three things:

1.ADVANCE A LIFEGUARD'S SCOPE OF PRACTICE:

As we all may or may not know, the EMS system in general is in crisis. We are amidst a global pandemic, EMS agencies face closures, We are short-staffed, overworked, and underappreciated. The industry lacks funding, unity, and advocacy. Imagine if Lifeguards could be the key to solving this global issue. Train Lifeguards beyond their current scope of practice. Provide them education and hands on learning to include advanced first aid skills learned in Emergency Medical Response Training and Emergency Medical Technician training. Why? So that Lifeguards can do more when EMS agencies cannot respond quickly enough to an emergency. We are at a point where 911 may not show up when called. Why the delay? No personnel, no trucks, and more important emergencies that take up priority resources. Lifeguards are not just, "first responders" they are the, "FIRST, first responders". They are on scene many minutes prior to ems arrival. Why not have lifeguards trained in advanced first aid so that they can provide advanced pre-hospital training to those in need? Create more upward mobility in a lifeguard's career track by offering a tangible future. If we trained lifeguards as EMR's or EMT's, the industry would have significantly more staff available to aid in emergency situations. Seems like a win-win in my opinion.

2.IMPLEMENTING INNOVATIVE TECHNOLOGY

Lifeguards are humans who need eyes on the backs of their heads. Technology can only help and support a lifeguards role. Technology is not meant to replace lifeguards but to aid them

in what they already do. Rescue dogs, drones, anti-drowning wearable technology and underwater cameras are rather new innovations in the last 10 years but they are not widely used. Why not? Industry professionals must not be afraid of change, they must learn to embrace it. Our society relies more and more on the use of electronics, computers, and technology. Why is the aquatics industry so far behind? Innovative technology also comes in the form of digitization of daily aquatics operations with custom software solutions that reduce facility risk and promote the utmost in safety and efficiency. Why would pool operators and aquatics professionals not want convenience and peace of mind at all times? Technology should never substitute or replace lifeguards. However, integration is never a bad thing.

3.UTILIZING AND STANDARDIZING AUGMENTED REALITY TRAINING

Lifeguards need higher quality, more realistic, and more frequent training. Imagine an immersive experience where lifeguards could learn to respond to emergencies in virtual reality? No costly equipment to purchase. No need for multiple staff to be present to run training scenarios. No issues with limited pool space, zero-increase budgets, and staffing instructors to run your training sessions. An element of virtual reality training would build individual lifeguard confidence and offer real-life response efforts with immediate feedback on their actions. Marketing virtual reality challenges could be a way to engage interest of non-lifeguards looking to become lifeguards. "Can you spot the drowning victim"? or, "What would you do in an aquatic emergency"?. Again, just like with technology, this type of training should never be designed to replace physical, in-person training. It should exist merely as another, "tool in the toolbox" for training. So who will be at the forefront of these concepts? Who will serve as the future model agency to forever change the industry and how lifeguards train? I guess we will have to wait and see. This article was written based on my own opinions and experiences. Please feel free to share and comment.

13. APPENDIX

(i) SOURCE CODE

```
[net]
# Testing# batch=1
# subdivisions=1#
Training batch=64
subdivisions=16
width=608
height=608
channels=3
momentum=0.9
decay=0.0005
angle=0 saturation
= 1.5
exposure = 1.5hue=.1

learning_rate=0.01
burn_in=1000 max_batches
= 500200policy=steps
steps=400000,450000
scales=.1,.1

[convolutional]
batch_normaliz
e=1 filters=32
size=3 stride=1
pad=1
activation=leak
y

# Downsample
```

```
[convolutional]
batch_normaliz
e=1 filters=64
size=3
stride=2
pad=1
activation=le
aky
```

```
[convolutional]
batch_normaliz
e=1 filters=32
size=1
stride=1
pad=1
activation=le
aky
```

```
[convolutional]
batch_normaliz
e=1 filters=64
size=3 stride=1
pad=1
activation=leak
y
[shortcut]from=
- 3
activation=line
ar#
```

Downsample

```
[convolutional]
batch_normaliz
e=1 filters=128
size=3
stride=2
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=64
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
activation=line
ar
```

```
[convolutional]
batch_normaliz
e=1 filters=64
size=1
stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=3
stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
activation=line
ar
```

Downsample

```
[convolutional]
batch_normaliz
e=1
filters=256size
=3 stride=2
pad=1
```


activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y

```
[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
activation=line
ar
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
```

activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz
e=1 filters=256
size=3
stride=1
pad=1
activation=le
aky

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1

```
stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
activation=line
ar
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=1
stride=1
pad=1
activation=le
aky
```

```
[convolutional]
batch_normaliz
e=1 filters=256
```

size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=256
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar#

Downsample

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=2
pad=1 activation=leaky

[convolutional]
batch_normaliz

e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz
e=1 filters=512

size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1 activation=leaky

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz

e=1 filters=256
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1
stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz

e=1 filters=512
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1
stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3

activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz
e=1 filters=512
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1

activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512
size=3
stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar#

Downsample

[convolutional]
batch_normaliz
e=1
filters=1024
size=3 stride=2
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1 filters=512

size=1 stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1
filters=1024
size=3
stride=1
pad=1
activation=le
aky

[shortcut]from=
- 3
activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=512
size=1
stride=1
pad=1
activation=le
aky

[convolutional]
batch_normaliz

```
e=1
filters=1024
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
activation=line
ar
```

```
[convolutional]
batch_normaliz
e=1 filters=512
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1
filters=1024
size=3 stride=1
pad=1
activation=leak
y
```

```
[shortcut]from=
- 3
```

activation=line
ar

[convolutional]
batch_normaliz
e=1 filters=512
size=1
stride=1
pad=1
activation=leak
y

[convolutional]
batch_normaliz
e=1
filters=1024
size=3 stride=1
pad=1
activation=leak
y

[shortcut]from=
- 3
activation=line
ar

#####

[convolutional]
batch_normaliz
e=1 filters=512
size=1 stride=1

pad=1
activation=leak
y

[convolutional]
batch_normalize=1 size
=3 stride=1 pad=1
filters=1024
activation=leaky

[convolutional]
batch_normaliz
e=1 filters=512
size=1
stride=1
pad=1
activation=leak
y

[convolutional]
batch_normalize=1 size
=3 stride=1 pad=1
filters=1024
activation=leaky

[convolutional]
batch_normaliz
e=1 filters=512
size=1
stride=1
pad=1


```
activation=leaky
    
```

```
[convolutional]
    batch_normalize=1 size=
    =3 stride=1 pad=1
    filters=1024
    activation=leaky
    
```

```
[convolutional]size=
    e=1 stride=1
    pad=1 filters=255
    activation=linear
    
```

```
[yolo]
    mask =
    6,7,8
    anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,
    156,198,
    373,326
    classes=80
    num=9
    jitter=.3
    ignore_thresh
    = .7
    truth_thresh = 1 random=1
    
```

```
[route] layers =
    -4
    
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y
```

```
[upsample]
stride=2
```

```
[route]
layers = -1,
61
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normalize=1siz
e=3 stride=1 pad=1
filters=512
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normalize=1siz
e=3 stride=1 pad=1
filters=512
activation=leak
y
```

```
[convolutional]
batch_normaliz
e=1 filters=256
size=1 stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normalize=1siz
e=3 stride=1 pad=1
filters=512
activation=leak
y
```

```
[convolutional]size  
e=1      stride=1  
pad=1 filters=255  
activation=linear
```

```
[yolo]  
mask      =  
3,4,5  
anchors  =  
10,13,  
16,30,  
33,23,  
30,61,  
62,45,  
59,119,  
116,90,  
156,198,  
373,326  
classes=80  
num=9  
jitter=.3  
ignore_thresh  
= .7  
truth_thresh = 1 random=1
```

```
[route] layers =  
-4
```

```
[convolutional]  
batch_normaliz
```

```
e=1 filters=128
size=1 stride=1
pad=1
activation=leak
y
```

```
[upsample]
stride=2
```

```
[route]
layers = -1,
36
```

```
[convolutional]
batch_normaliz
e=1 filters=128
size=1
stride=1
pad=1
activation=leak
y
```

```
[convolutional]
batch_normalize=1 siz
e=3 stride=1 pad=1
filters=256
activation=leaky
```

```
[convolutional]
batch_normaliz
```

e=1 filters=128
size=1
stride=1
pad=1
activation=leak
y

[convolutional]
batch_normalize=1 size=3
stride=1
pad=1
filters=256
activation=lea
ky

[convolutional]
batch_normalize=1
filters=128 size=1
stride=1
pad=1
activation=lea
ky

[convolutional]
batch_normalize=1 siz
e=3 stride=1 pad=1
filters=256
activation=leaky

```
[convolutional]siz
e=1      stride=1
pad=1 filters=255
activation=line
ar
```

```
[yolo]
mask    =
0,1,2
anchors = 10,13, 16,30, 33,23, 30,61, 62,45, 59,119, 116,90,
156,198,
373,326
classes=80
num=9
jitter=.3
ignore_thresh
=      .7
truth_thresh =
1 random=1
```

Source code(ii)

```
#import      necessary
packagesimport cv2
import os

import numpy as np

from .utils import download_file
```

```
initialize = True
net = None
dest_dir = os.path.expanduser('~') + os.path.sep + '.cvlib'
+ os.path.sep + 'object_detection' + os.path.sep + 'yolo' + os.path.sep + 'yolov3'
classes = None
```

```
#colors are BGR instead of RGB in python
```

```
COLORS = [0,0,255], [255,0,0]
```

```
def populate_class_labels():
```

```
#we are using a pre existent classifier which is more reliable and more efficient
than one#we could make using only a laptop
```

```
#The classifier should be downloaded automatically when you run this
scriptclass_file_name =
```

```
'yolov3_classes.txt' class_file_abs_path =
```

```
dest_dir + os.path.sep + class_file_name
```

```
url = 'https://github.com/Nico31415/Drowning-Detector/raw/master/yolov3.txt'if
```

```
not os.path.exists(class_file_abs_path):
```

```
    download_file(url=url, file_name=class_file_name, dest_dir=dest_dir)f =
```

```
open(class_file_abs_path, 'r')
```

```
classes = [line.strip() for line in f.readlines()]
```

```
return classes
```

```
def get_output_layers(net)
```



```
#the number of output layers in a neural network is the number of
possible#things the network can detect, such as a person, a dog, a tie, a
phone... layer_names = net.getLayerNames()
```

```
output_layers = [layer_names[i[0] - 1] for i in net.getUnconnectedOutLayers()]
```

```
return output_layers
```

```
def draw_bbox(img, bbox, labels, confidence, Drowning, write_conf=False):
```

```
    global COLORS
```

```
    global classes
```

```
    if classes is None:
```

```
        classes =
        populate_class_labels()
```

```
    for i, label in enumerate(labels):
```

```
        #if the person is drowning, the box will be drawn red instead
        of blueif label == 'person' and Drowning: color =
        COLORS[0] label
        = 'DROWNING'
```

```
    else:
```

```
color = COLORS[1]
```

```
if write_conf:
```

```
    label += ' ' + str(format(confidence[i] * 100, '.2f')) + '%'
```

```
#you only need to points (the opposite corners) to draw a rectangle. These  
points#are stored in the variable bbox
```

```
cv2.rectangle(img, (bbox[i][0],bbox[i][1]), (bbox[i][2],bbox[i][3]), color, 2)
```

```
cv2.putText(img, label, (bbox[i][0],bbox[i][1]-10),  
cv2.FONT_HERSHEY_SIMPLEX, 0.5, color, 2)
```

```
return img
```

```
def detect_common_objects(image, confidence=0.5, nms_thresh=0.3):
```

```
Height, Width = image.shape[:2]scale =
```

```
0.00392
```

```
global classes
```

```
global dest_dir
```

```
#all the weights and the neural network algorithm are already preconfigured#as we  
are using
```

```
YOLO
```

#this part of the script just downloads the YOLO files

```
config_file_name = 'yolov3.cfg'
```

```
config_file_abs_path = dest_dir + os.path.sep + config_file_name
```

```
weights_file_name = 'yolov3.weights'
```

```
weights_file_abs_path = dest_dir + os.path.sep + weights_file_name
```

```
url = 'https://github.com/Nico31415/Drowning-Detector/raw/master/yolov3.cfg'
```

```
if not os.path.exists(config_file_abs_path):
```

```
    download_file(url=url, file_name=config_file_name, dest_dir=dest_dir)
```

```
url = 'https://pjreddie.com/media/files/yolov3.weights'
```

```
if not os.path.exists(weights_file_abs_path):
```

```
download_file(url=url, file_name=weights_file_name,  
dest_dir=dest_dir)
```

```
global initialize
```

```
global net
```

```
if initialize:
```

```
classes = populate_class_labels()
```

```
net = cv2.dnn.readNet(weights_file_abs_path,  
config_file_abs_path) initialize = False
```

```
blob = cv2.dnn.blobFromImage(image, scale, (416,416), (0,0,0), True,  
crop=False)
```

```
net.setInput(blob)
```

```
outs = net.forward(get_output_layers(net))
```

```
class_ids =
```

```
[]
```

```
confidences
```

```
= [] boxes =
```

```
[]
```

```
for out in outs:
```

```
    for detection in out: scores =
```

```
        detection[5:] class_id
```

```
        = np.argmax(scores)
```

```
        max_conf =
```

```
        scores[class_id] if
```

max_conf >
confidence:

```

center_x = int(detection[0] * Width)
center_y = int(detection[1] * Height)
w = int(detection[2] * Width)
h = int(detection[3] * Height)
x = center_x - w / 2
y = center_y - h / 2
class_ids.append(class_id)
confidences.append(float(max_conf))
boxes.append([x, y, w, h])

```

```

indices = cv2.dnn.NMSBoxes(boxes, confidences,
confidence, nms_thresh)

```

```

bbox = []
label = []
conf = []

```

```

for i in indices:

```

```
i = i[0]
```

```
box      =
```

```
boxes[i]x  =
```

```
box[0]  y  =
```

```
box[1]  w  =
```

```
box[2]  h
```

```
=
```

```
box[3]
```

```
bbox.append([round(x),    round(y),    round(x+w),
```

```
round(y+h)])    label.append(str(classes[class_ids[i]]))
```

```
conf.append(confidences[i])
```

```
return bbox, label, conf
```

GITHUB:

<https://github.com/IBM-EPBL/IBM-Project-48311-1660806463>

