IBM NALAIYA THIRAN

VIRTUALEYE-LIFE GUARD FORSWIMMING POOLS TO DETECTACTIVE DROWNING

PROJECT REPORT

Submitted by

KARTHIKEYAN S (620619106012)
MOHAMMED SUHAIL K (620619106016)
SANTHOSH S (620619106027)
SURYA V S (620619106039)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING
In
ELECTRONICS AND COMMUNICATION ENGINEERING

GANESH COLLEGE OF ENGINEERING,
METTUPATTI,
SALEM. 636111

TABLES OF CONTENTS

INTRODUCTION

1.1 PROJECT OVERVIEW:

Swimming is one of the best exercises that help people to reduce stress in this urban lifestyle. Swimming pools are found larger in number in hotels, and weekend tourist spots and barely people have them in their house backyard. Beginners, especially, often feel it difficult to breathe underwater, which causes breathing trouble which in turn causes a drowning accident. Worldwide, drowning produces a higher rate of mortality without causing injury to children. Children under six of their ages 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. To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life.

By studying body movement patterns and connecting cameras to artificial intelligence (AI) systems 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 analyzing 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 lifeguard's attention.

1.2 PURPOSE:

The main purpose of our project is to detect the active drowning in the swimming pool. Using the YOLO algorithm, drowning can be easily identified and notifies the Lifeguard and the drowning person can be easily rescued. By using our application, the drowning accident can be reduced.

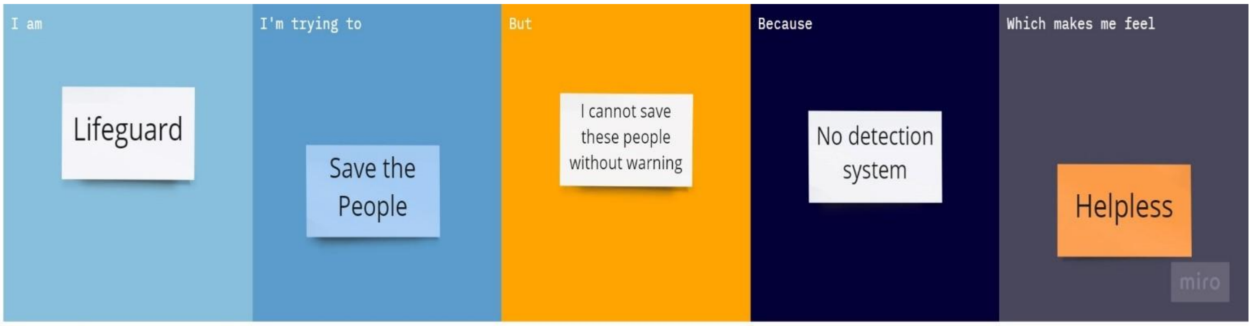
IDEATION & PROPOSED SOLUTION
LITERATURE SURVEY

SNO:	TITLE OF THE PAPER	DETAILS OF THE PAPER	OBJECTIVE	METHODOLGY USED	TAKE AWAY
1.	An ImprovedDetection Method of Human Target at Sea Based on Yolov3	2021 IEEE	To Search and rescue drowning people at the sea site.	The object detection method is based on deep learning and the Yolov3 algorithm which is a representative algorithm of the object Detection method.	From that journal,we use the YOLO algorithm. Because of its high accuracy and fast detection speed.
2.	A novel drowning section method for safety of swimmers	2018 IEEE	To ensure detection of drowning and reporting at the earlier stages	Two drowning detection sensors are placed on the side elastic of the goggles. The Alarm transmission module is used to send the alarms when the drowning is detected.It's triggered by a drowning detection unit. The Alarms are transmitted using the underwater Communication.	From this journal, we use the Novel Camera-Based Drowning Detection Algorithm, which could predict the drowning.

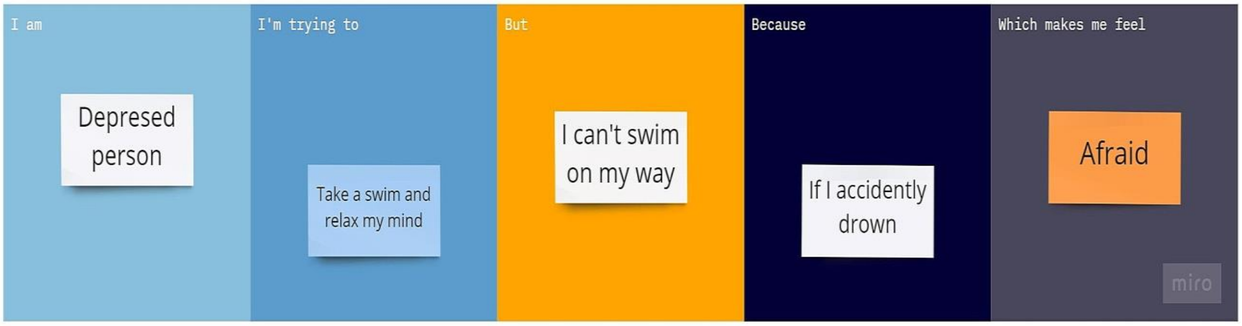
3.	Automated Vision-based Surveillance System to Detect Drowning Incidents in Swimming Pools	2020 IEEE	An automated vision-based surveillance system to prevent drowning accidents	The system consists of a Raspberry Pi with the Pixy camera,an Arduino Na no board, an alarm system. The proposed system is based on the color-based algorithm to position and rescue swimmers who are drowning. The device then sends an alarm to the Lifeguards.	From this journal, we had learned that the hybrid system will automatically Detect a drowning person and then set off an alarm to alert lifeguards.
4.	Swimmer motion analysis with application to drowning detection	2002 IEEE	To build an automated video surveillance system to detect potential drowning incidents.	Two event-inference modules have been developed: one evaluates the condition of a swimmer using a set of reasoning rules and triggers alarms after the swimmer remains in ‘Possible drowning state’ for a period longer than a preset duration.	From this journal, we had learned that the drowning incidents are detected by examining the sequence of motion and shape features extracted from swimmers in the pools.
5.	IoT Based Safety Enhanced Swimming Pool with Embedded Techniques to reduce drowning accidents.	2020 IEEE	To reduce the drowning accidents and to save human life.	The wearable device on the wrist to monitor moment of human under water is designed. The imaging techniques with robust image smoothing algorithm were proposed for the early detection of drowning.	Through this journal, we learned that the portable device can also monitor and detect drowning people in the pool.

PROBLEM STATEMENT DEFINITION

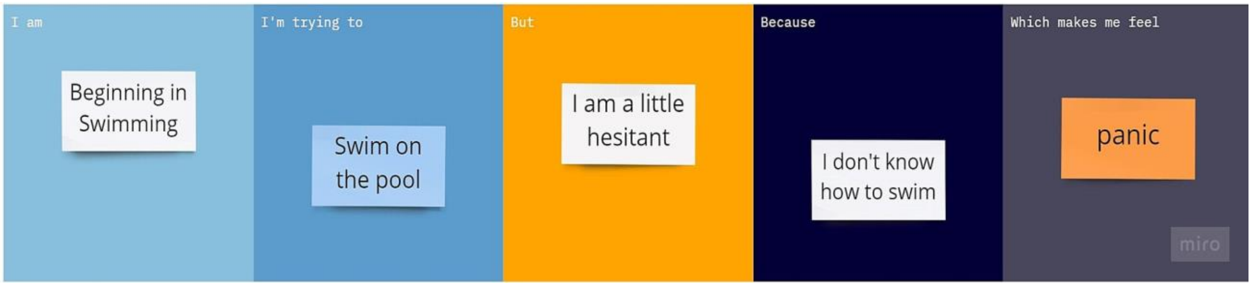
Our main aim of the project is to save the drowning people in the swimming pool with the help of drowning detection software and lifeguard in pool.



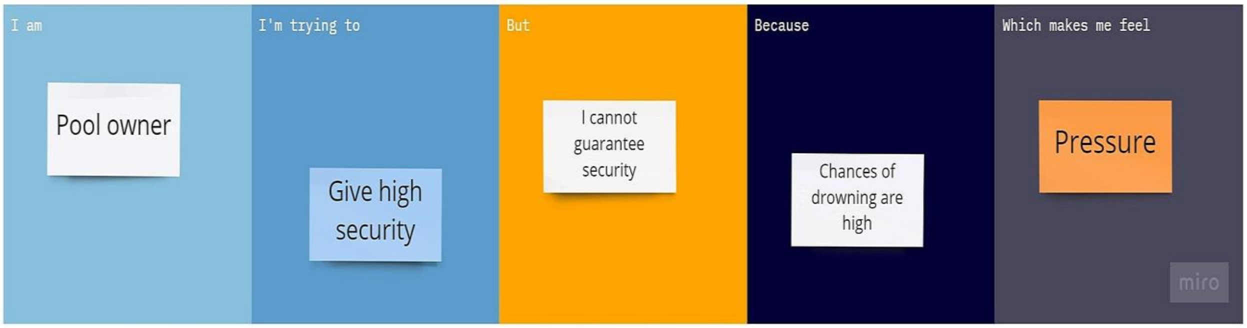
I am	I'm trying to	But	Because	Which makes me feel
Lifeguard	Save the People	I cannot save these people without warning	No detection system	Helpless



I am	I'm trying to	But	Because	Which makes me feel
Depresed person	Take a swim and relax my mind	I can't swim on my way	If I accidently drown	Afraid



I am	I'm trying to	But	Because	Which makes me feel
Beginning in Swimming	Swim on the pool	I am a little hesitant	I don't know how to swim	panic

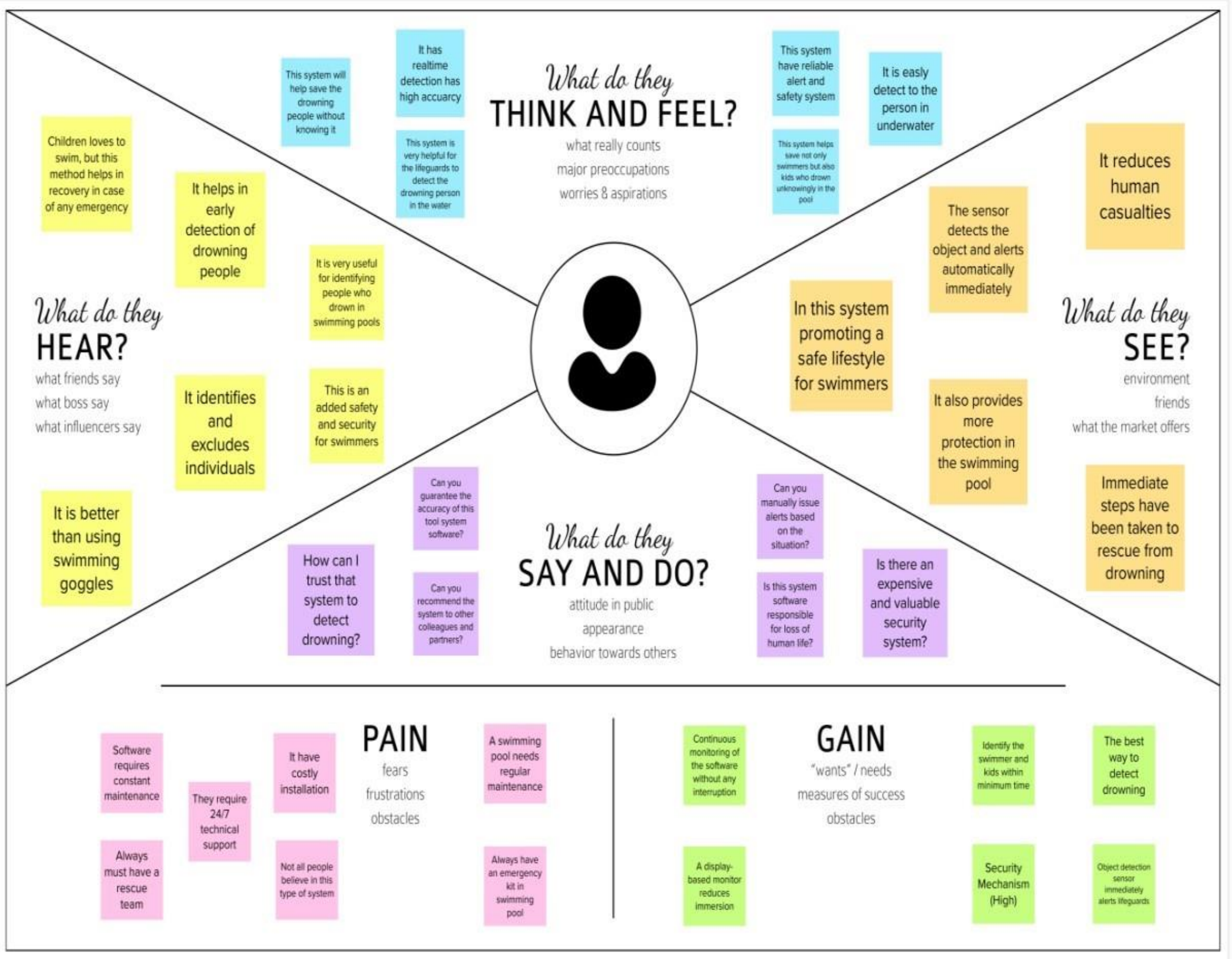


I am	I'm trying to	But	Because	Which makes me feel
Pool owner	Give high security	I cannot guarantee security	Chances of drowning are high	Pressure

EMPATHY MAP CANVAS

An empathy map is a widely-used visualization tool within the field of UX and HCI practice. The primary purpose of an empathy map is to bridge the understanding of the end user. The traditional empathy map begins with four categories: says, thinks, does, and feels. Within context of its application, this tool is used to build a shared understanding of the user's needs and provide context to a user-centered solution.

It is a simple, easy-to-digest visual that captures knowledge about a user’s behaviors and attitudes. It is a useful tool to help teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it.



What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations

This system will help save the drowning people without knowing it

It has realtime detection has high accuarcy

This system is very helpful for the lifeguards to detect the drowning person in the water

This system have reliable alert and safety system

This system helps save not only swimmers but also kids who drown unknowingly in the pool

It is easly detect to the person in underwater

What do they HEAR?
what friends say
what boss say
what influencers say

Children loves to swim, but this method helps in recovery in case of any emergency

It helps in early detection of drowning people

It is very useful for identifying people who drown in swimming pools

It identifies and excludes individuals

This is an added safety and security for swimmers

It is better than using swimming goggles

What do they SEE?
environment
friends
what the market offers

It reduces human casualties

The sensor detects the object and alerts automatically immediately

In this system promoting a safe lifestyle for swimmers

It also provides more protection in the swimming pool

Immediate steps have been taken to rescue from drowning

What do they SAY AND DO?
attitude in public
appearance
behavior towards others

Can you guarantee the accuracy of this tool system software?

How can I trust that system to detect drowning?

Can you recommend the system to other colleagues and partners?

Can you manually issue alerts based on the situation?

Is this system software responsible for loss of human life?

Is there an expensive and valuable security system?

PAIN
fears
frustrations
obstacles

Software requires constant maintenance

They require 24/7 technical support

It have costly installation

A swimming pool needs regular maintenance

Always must have a rescue team

Not all people believe in this type of system

Always have an emergency kit in swimming pool

GAIN
“wants” / needs
measures of success
obstacles

Continuous monitoring of the software without any interruption

A display-based monitor reduces immersion

Identify the swimmer and kids within minimum time

The best way to detect drowning

Security Mechanism (High)

Object detection sensor immediately alerts lifeguards

IDEATION & BRAINSTORMING

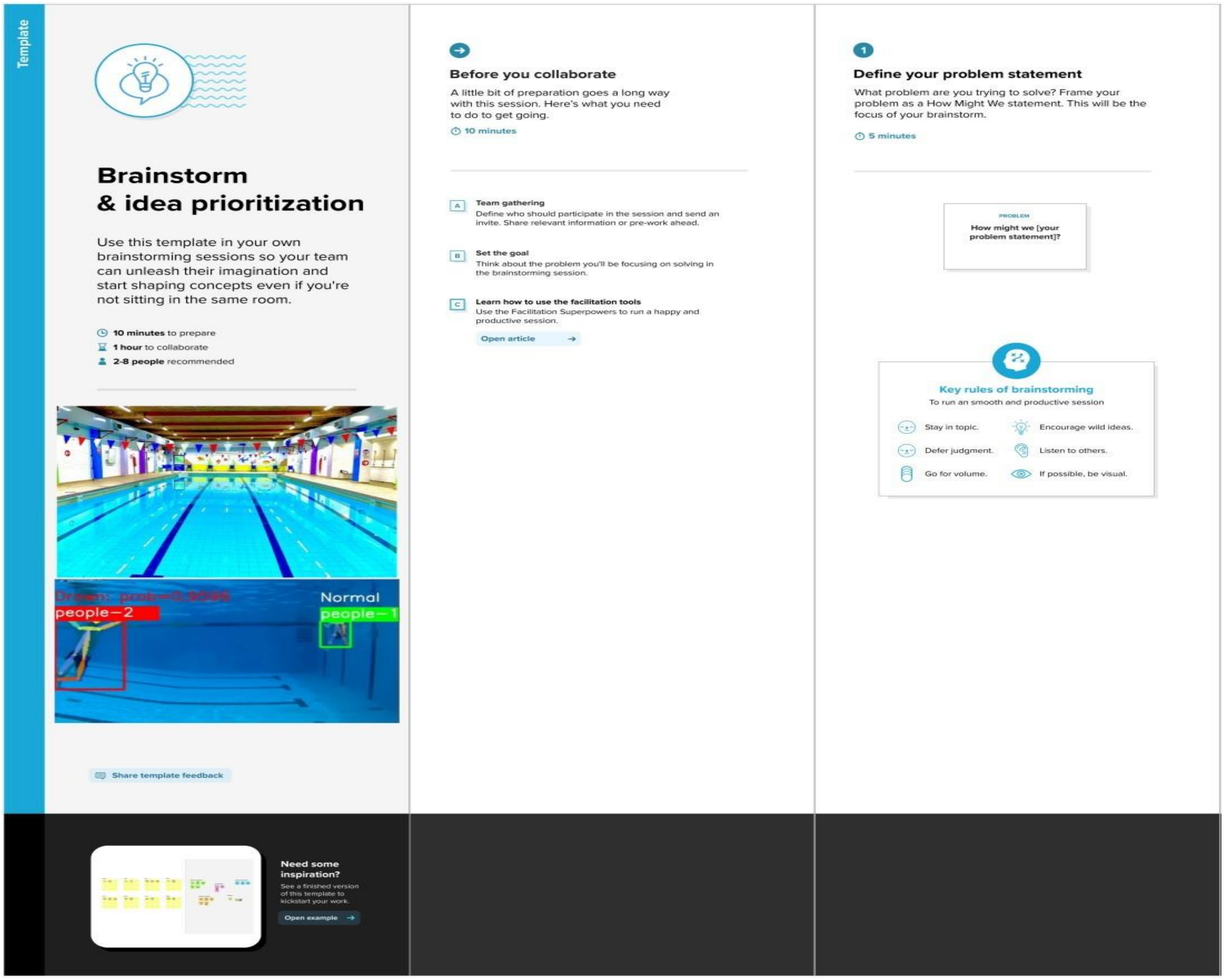
The Ideation is commonly more thought of as being an individual pursuit, while brainstorming is almost always a group activity. Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving.

Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions.

Step-1: Team Gathering, Collaboration and Select the Problem Statement.

Step-2: Brainstorm, Idea Listing and Grouping.

Step-3: Idea Prioritization



Brainstorm
& idea prioritization

Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended

Need some inspiration?
See a finished version of this template to kickstart your work.

Before you collaborate

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

1 **Define your problem statement**

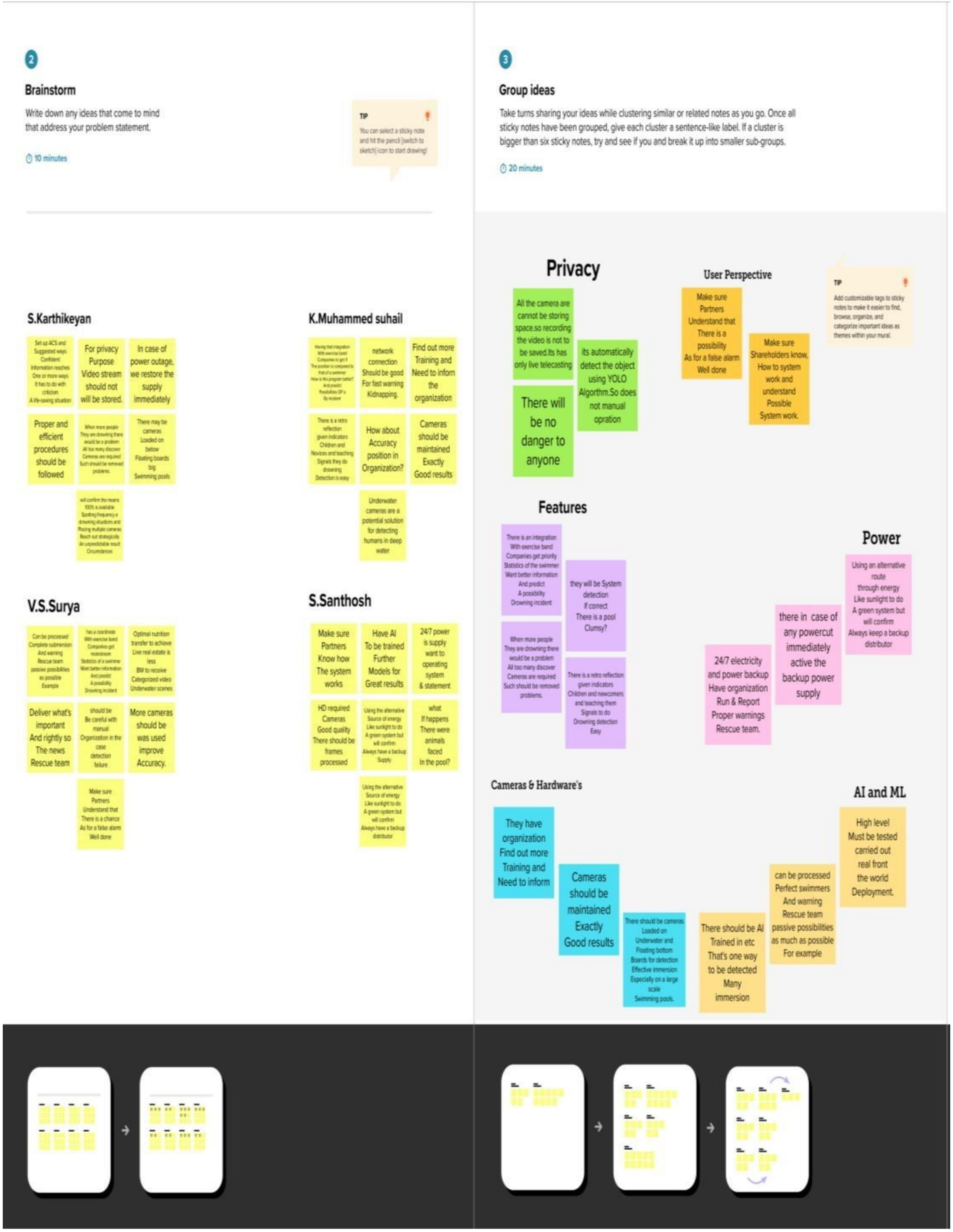
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

5 minutes

PROBLEM
How might we [your problem statement]?

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.	Encourage wild ideas.
Defer judgment.	Listen to others.
Go for volume.	If possible, be visual.



2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

S.Karthikeyan

Set up ACS and Suggested ways Confident Information reaches One or more ways It has to do with criticism A life-saving situation

For privacy Purpose Video stream should not will be stored.

In case of power outage, we restore the supply immediately

Proper and efficient procedures should be followed

When more people They are drowning there would be a problem All too many discover Cameras are required Such should be removed problems.

There may be cameras Loaded on below Floating boards big Swimming pools

will confirm the means 100% is available Spotting frequency a drowning situations and Placing multiple cameras Reach out strategically An unpredictable result Circumstances

K.Muhammed suhail

Having that integration With exercise band Companies to get it The position is compared to that of a swimmer How is this program better? And predict Possibilities OF a By incident

network connection Should be good For fast warning Kidnapping.

Find out more Training and Need to inform the organization

There is a retro reflection given indicators Children and Novices and teaching Signals they do drowning Detection is easy

How about Accuracy position in Organization?

Cameras should be maintained Exactly Good results

Underwater cameras are a potential solution for detecting humans in deep water

V.S.Surya

Can be processed Complete submersion And warning Rescue team passive possibilities as possible Example

has a coordinate With exercise band Companies get mainstream Statistics of a swimmer Want better information And predict A possibility Drowning incident

Optimal nutrition transfer to achieve Live real estate is less BW to receive Categorized video Underwater scenes

Deliver what's important And rightly so The news Rescue team

should be Be careful with manual Organization in the case detection failure

More cameras should be was used improve Accuracy.

Make sure Partners Understand that There is a chance As for a false alarm Well done

S.Santhosh

Make sure Partners Know how The system works

Have AI To be trained Further Models for Great results

24/7 power is supply want to operating system & statement

HD required Cameras Good quality There should be frames processed

Using the alternative Source of energy Like sunlight to do A green system but will confirm Always have a backup Supply

what If happens There were animals faced In the pool?

Using the alternative Source of energy Like sunlight to do A green system but will confirm Always have a backup distributor

3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

20 minutes

Privacy

All the camera are cannot be storing space.so recording the video is not to be saved.Its has only live telecasting

There will be no danger to anyone

its automatically detect the object using YOLO Algorthm.So does not manual opration

User Perspective

Make sure Partners Understand that There is a possibility As for a false alarm Well done

Make sure Shareholders know, How to system work and understand Possible System work.

TIP

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

Features

There is an integration With exercise band Companies get priority Statistics of the swimmer Want better information And predict A possibility Drowning incident

they will be System detection If correct There is a pool Clumsy?

When more people They are drowning there would be a problem All too many discover Cameras are required Such should be removed problems.

There is a retro reflection given indicators Children and newcomers and teaching them Signals to do Drowning detection Easy

Power

Using an alternative route through energy Like sunlight to do A green system but will confirm Always keep a backup distributor

there in case of any powercut immediately active the backup power supply

24/7 electricity and power backup Have organization Run & Report Proper warnings Rescue team.

Cameras & Hardware's

They have organization Find out more Training and Need to inform

Cameras should be maintained Exactly Good results

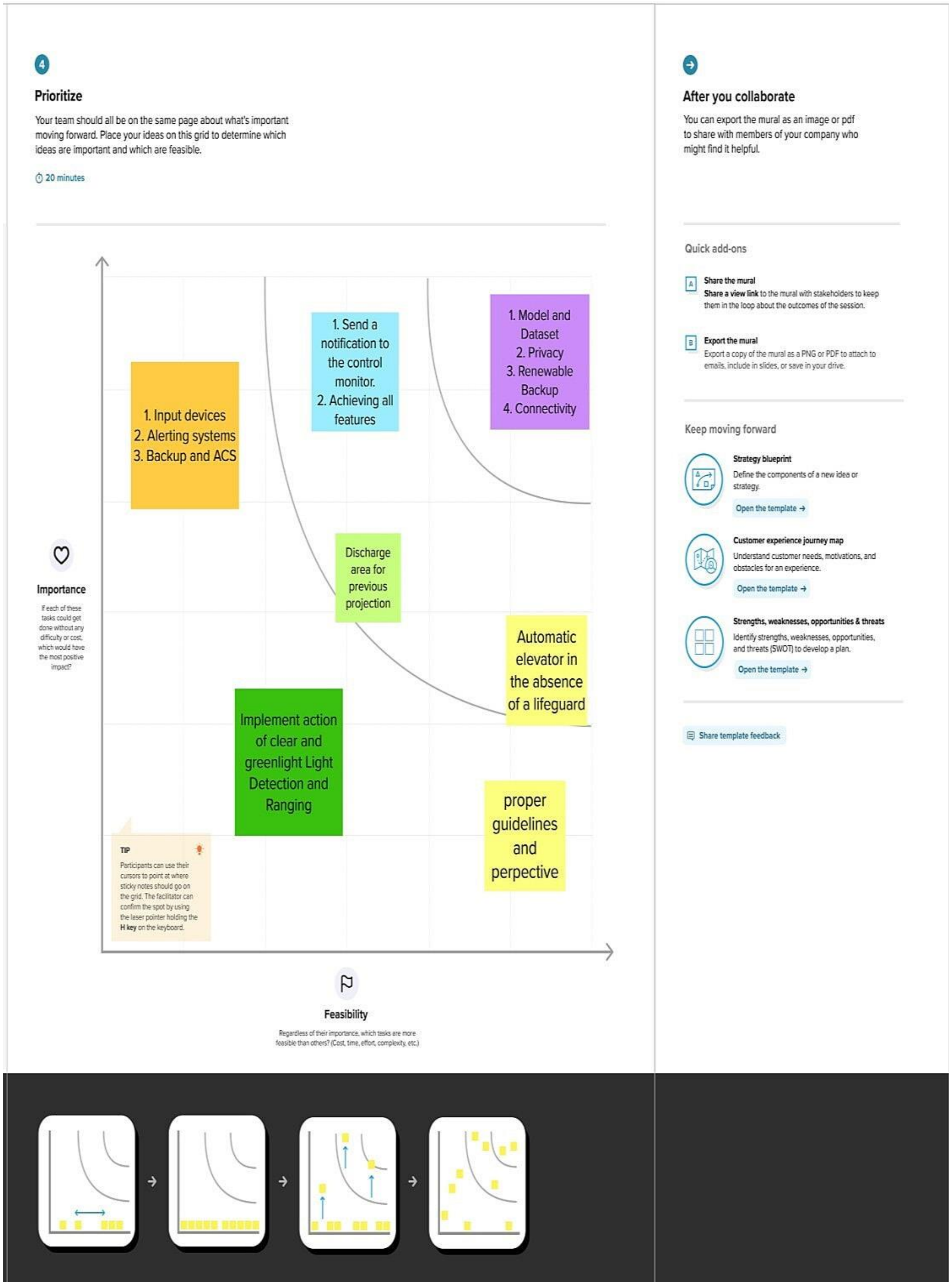
There should be cameras Loaded on Underwater and Floating bottom Boards for detection Effective immersion Especially on a large scale Swimming pools.

AI and ML

High level Must be tested carried out real front the world Deployment.

can be processed Perfect swimmers And warning Rescue team passive possibilities as much as possible For example

There should be AI Trained in etc That's one way to be detected Many immersion



4

Prioritize

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

20 minutes

1. Input devices
2. Alerting systems
3. Backup and ACS

1. Send a notification to the control monitor.
2. Achieving all features

1. Model and Dataset
2. Privacy
3. Renewable Backup
4. Connectivity

Discharge area for previous projection

Automatic elevator in the absence of a lifeguard

Implement action of clear and greenlight Light Detection and Ranging

proper guidelines and perpective

Importance

If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

TIP

Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

After you collaborate

You can export the mural as an image or pdf to share with members of your company who might find it helpful.

Quick add-ons

A **Share the mural**
Share a view link to the mural with stakeholders to keep them in the loop about the outcomes of the session.

B **Export the mural**
Export a copy of the mural as a PNG or PDF to attach to emails, include in slides, or save in your drive.

Keep moving forward

Strategy blueprint
Define the components of a new idea or strategy.
Open the template →

Customer experience journey map
Understand customer needs, motivations, and obstacles for an experience.
Open the template →

Strengths, weaknesses, opportunities & threats
Identify strengths, weaknesses, opportunities, and threats (SWOT) to develop a plan.
Open the template →

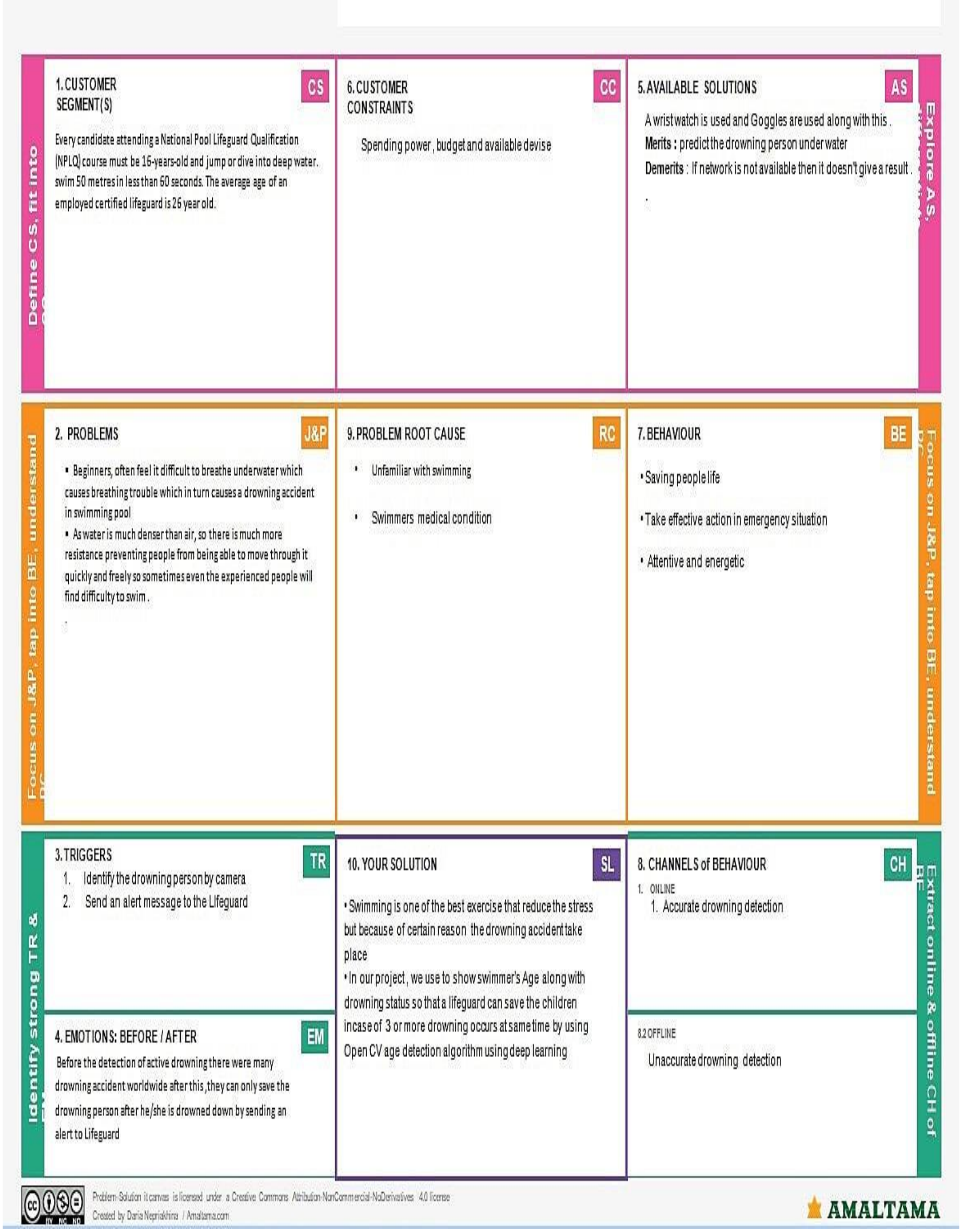
Share template feedback

PROPOSED SOLUTION

S.No	Parameter	Description
1.	Problem Statement (Problem to be solved)	Swimming pools are generally places of fun and recreation Healthy exercise, but they can also be dangerous. Even with a lifeguard on duty, Swimmers still struggle underwater or in areas beyond pool lifeguards field of view
2.	Idea / Solution description	In this project, we use artificial intelligence. We install underwater cameras for detection Drowning people. Using deep learning, Image can be recognized. If the image Detected, it is triggers an alarm to alert the lifeguard to rescue drowning people
3.	Novelty / Uniqueness	The uniqueness of our system software to track the position and the location of a drowning person. We use YOLO Algorithm. Because of its high accuracy and fast detection speed. So it helps lifeguard to save people within seconds.

4.	Social Impact / Customer Satisfaction	Drowning globally has a higher death rate and is also the third leading cause of unexpected deaths worldwide, especially among children under the age of six. To overcome this conflict our drowning detection system will have an impact on society.
5.	Business Model (Revenue Model)	We can introduce the software based approach for making a good income. It is extremely useful to lifeguards, swimmers and business operators. The number of features makes it attractive for end users to use our software system.
6.	Scalability of the Solution	Our software system can be used by the company driver who manages the pools. We use the IBM cloud server to collect and maintain the data. We will ensure the safety of the swimmers.

3.4 PROMBLEM SOLUTION FIT



Define CS, fit into

1. CUSTOMER SEGMENT(S)

CS

Every candidate attending a National Pool Lifeguard Qualification (NPLQ) course must be 16-years-old and jump or dive into deep water. swim 50 metres in less than 60 seconds. The average age of an employed certified lifeguard is 26 year old.

6. CUSTOMER CONSTRAINTS

CC

Spending power , budget and available devise

5. AVAILABLE SOLUTIONS

AS

A wristwatch is used and Goggles are used along with this .

Merits : predict the drowning person under water

Demerits : If network is not available then it doesn't give a result .

.

Explore AS,

Focus on J&P, tap into BE, understand

2. PROBLEMS

J&P

- Beginners, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident in swimming pool
- As water is much denser than air, so there is much more resistance preventing people from being able to move through it quickly and freely so sometimes even the experienced people will find difficulty to swim .

.

9. PROBLEM ROOT CAUSE

RC

- Unfamiliar with swimming
- Swimmers medical condition

7. BEHAVIOUR

BE

- Saving people life
- Take effective action in emergency situation
- Attentive and energetic

Focus on J&P, tap into BE, understand

Identify strong TR &

3. TRIGGERS

TR

1. Identify the drowning person by camera
2. Send an alert message to the LIfeguard

4. EMOTIONS: BEFORE / AFTER

EM

Before the detection of active drowning there were many drowning accident worldwide after this ,they can only save the drowning person after he/she is drowned down by sending an alert to Lifeguard

10. YOUR SOLUTION

SL

- Swimming is one of the best exercise that reduce the stress but because of certain reason the drowning accident take place
- In our project , we use to show swimmer's Age along with drowning status so that a lifeguard can save the children incase of 3 or more drowning occurs at same time by using Open CV age detection algorithm using deep learning

8. CHANNELS of BEHAVIOUR

CH

1. ONLINE
   1. Accurate drowning detection

8.2 OFFLINE

Unaccurate drowning detection

Extract online & offline CH of

AMALTAMA

REQUIREMENT ANALYSIS

Functional requirement :

The Functional requirement required in our project are User registration – The registration can be done via email or phone number, User confirmation – The confirmation can be received in either Email or through the OTP, Installation of cameras – There are about more than 16 cameras are positioned in the underwater, Alarm system – The alarm system is used to alert the lifeguard by triggering the alarm.

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions. Functional requirements define what a product must do, what its features and functions are.

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Installation	Needed to be install under the water without annoying to the swimmer in the swimming pool.
FR-2	Deduction	Either horrified or unconscious
FR-3	Audio	shout for help or keep calm if the person is unconscious.
FR-4	Support	Take swim tube or take the help of rescue team.
FR-5	Prior Alert	Send alert message to the rescue team.

Non-Functional requirements :

The Non-Functional requirements of the proposed solution includes usability, security, reliability, performance, availability, scalability. It is not related to the system functionality, rather define how the system should perform. Nonfunctional requirements describe the general properties of a system. They are also known as quality attributes.

These are the queries arises during the requirement analysis:

How often does the system experience critical failures? How much time does it take to fix the issue when it arises? And how is user availability time compared to downtime?

Security - How well are the system and its data protected?

Reliability - Is the system performing consistently well?

Usability - How easy is it for a customer to use the system?

Following are the non-functional requirements of the proposed solution.

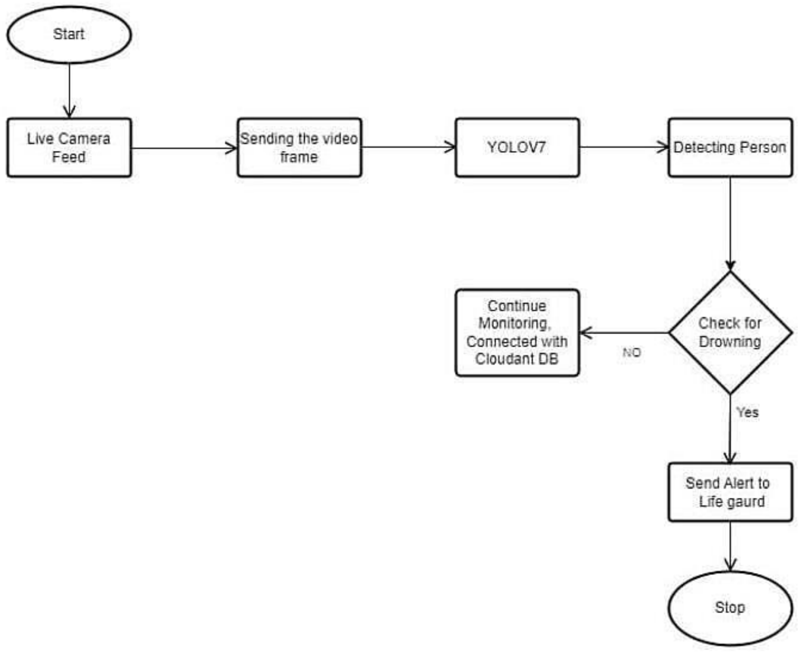
FR No.	**Non-Functional Requirement**	**Description**
NFR-1	Usability	To ensure the safety of each and every person present in the pool. A lifeguard should be present all the time in the pool.
NFR-2	Security	Rescue team should be aware of the alert message to save the life of the swimmer.
NFR-3	Reliability	Virtual eye lifeguards triggers an immediate prior alarm if a swimmer is in peril, helping to avoid panic even in critical situations.
NFR-4	Performance	The alarm is triggered when the swimmer’s pulse rate is decreasing.
NFR-5	Availability	Equipment and accessories include lifesaver rings, a shepherd’s crook, life hooks, spine boards, rescue tubes, and a first aid kit. Important to keep them accessible to quickly pull someone from the water safely.

NFR-6	Scalability	Virtual eye lifeguard finds potential drownings and promptly notifies you. It features the latest artificial intelligence technology and adapts to the needs of the user.

PROJECT DESIGN

DATA FLOW DIAGRAMS

- A Data Flow Diagram (DFD) is a traditional visual representation of the information flows with in 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.

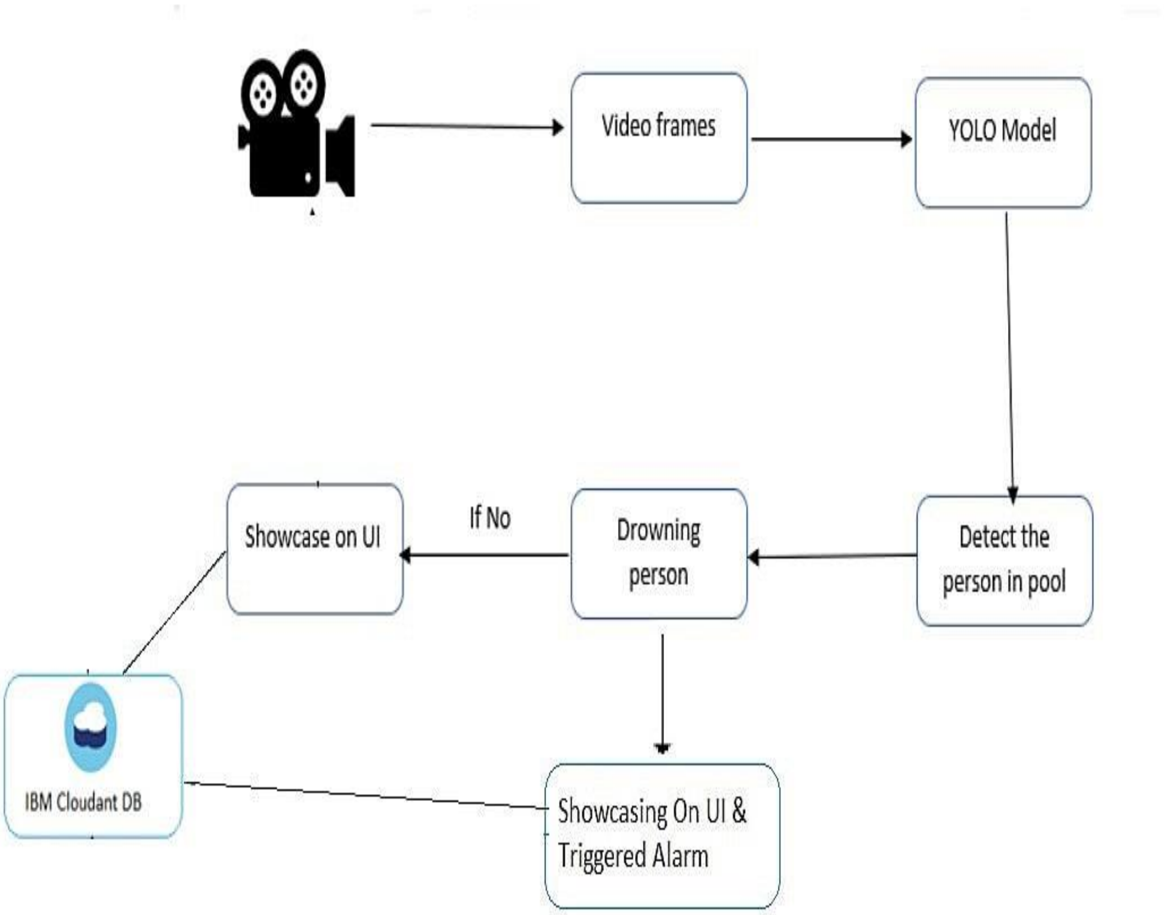


SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE (SA)

1. By studying body movement patterns and connecting cameras to artificial intelligence (AI) systems we can devise an underwater pool safety system that reduces the risk of drowning.
2. Usually, such systems can be developed by installing more than 16 cameras underwater and ceiling and analyzing the video feeds to detect any anomalies.
3. 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 then an alert will be generated to attract lifeguard's attention.

Solution Architecture Diagram:



Technical Architecture (TA)

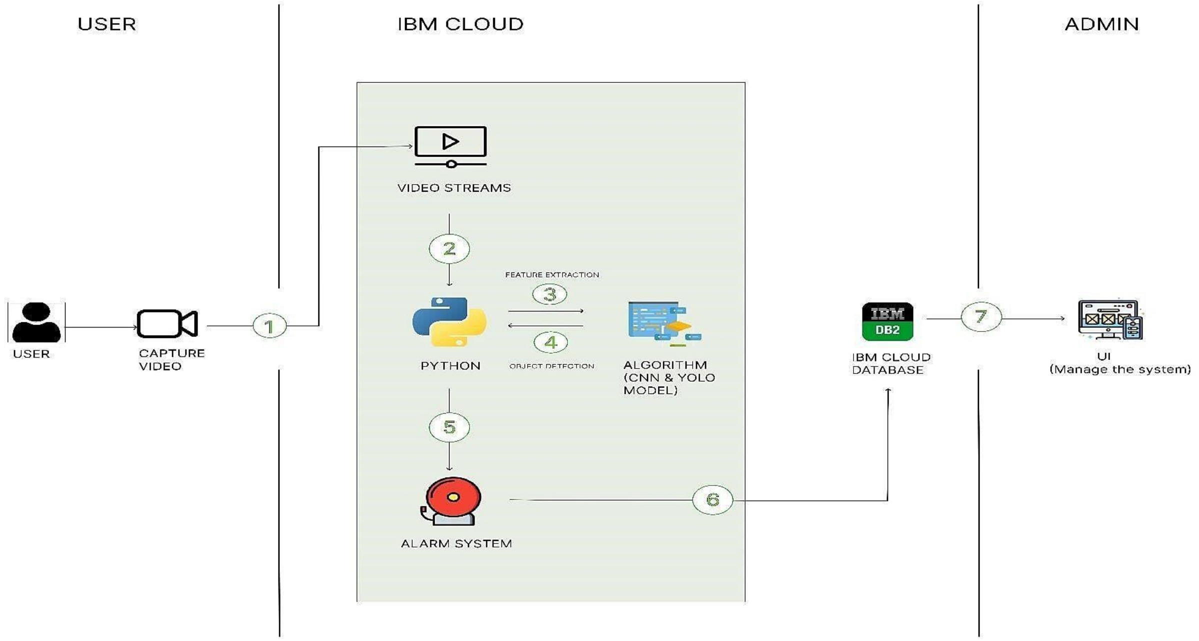


Table-1: Components & Technologies:

S. No	Component	Description	Technology
1.	User Interface	How user interacts with application	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Pre-processing the model using datasets	Python
3.	Application Logic-2	Image extraction	Python
4.	Application Logic-3	Object detection	python
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local File-system
8.	Deep Learning Model	Purpose of Deep Learning Model	Object Recognition Model, CNN etc. YOLOv7 model

9.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration:	Local, Cloud Foundry etc.,

Table-2: Application Characteristics:

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Python (Anaconda) open-source frameworks used	python
2.	Security Implementations	Camera under pools	AI
3.	Scalable Architecture	3 – tier Architecture	Python
4.	Availability	All the time persons are under surveillance	AI
5.	Performance	Many persons in the swimming pool will be detected whether the person is drowning or not	Python

USER STORIES

A user story is an informal, general explanation of a software feature written from the perspective of the end user or customer.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Pool owner)	Installation	USN-1	As a pool owner, I can install the cameras and set up the drowning detection system	I can connect the cameras to the cloud-hosted software	High	Sprint-1
	Detecting the drowning persons	USN-2	As a user, I can find the drowning persons by using the drowning detection system	I would receive an alert if a person is drowning	High	Sprint-1
	Notify the lifeguard	USN-3	As a user, I can notify the lifeguard when the system detects a drowning person	I can set up an alarm that would notify the lifeguard	High	Sprint-2
Customer (Lifeguard)	Rescue people	USN-4	As a user, I can rescue the drowning persons from the pool	I can save the drowning person	High	Sprint-2
Customer (Swimmers)	Safety	USN-5	As a user, I can swim without the fear of drowning	I can swim safely with the help of the system and the lifeguard	Medium	Sprint-2
Customer Care Executive	Contact	USN-6	resolve technical issues	I can contact the customer care executive to resolve any issues	Medium	Sprint-3
Administrator	Dashboard	USN-7	Management of the drowning detection system and database management.	I can access the system's logs and any other data instantly	High	Sprint-4

PROJECT PLANNING & SCHEDULING

SPRINT PLANNING & ESTIMATION

Use the below template to create product backlog and sprint 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	Karthikeyan
Sprint-1	Registration	VLGFSP-2	As a user, I will receive confirmation email once I have registered for the application	1	High	Mohammed Suhail
Sprint-1	Registration	VLGFSP - 3	As a user, I can register for the application through Facebook	2	Low	Santhosh
Sprint-1	Registration	VLGFSP - 4	As a user, I can register for the application through Gmail	2	Medium	Surya
Sprint-1	Login	VLGFSP - 6	As a user, I can log into the application by entering email & password	1	High	Mohammed Suhail
Sprint-2	Dataset Collect	VLGFSP - 11	Collect number of datasets and get accuracy	2	Medium	Santhosh
Sprint-2	Pre-processing	VLGFSP - 12	The dataset is extracted	2	High	Karthikeyan

Sprint-2	Train the model	VLGFSP - 13	Train the model.	4	High	Surya
Sprint-2	Test the model	VLGFSP - 14	Test the model	6	High	Karthikeyan
Sprint-3	Detection	VLGFSP - 15	Load the trained model.	3	High	Surya
Sprint-3	Detection	VLGFSP - 16	Identify the person by collecting real-time data through a webcam.	5	Medium	Mohammed Suhail
Sprint-3	Detection	VLGFSP - 16	classify it by using a trained model to predict the output	8	High	Santhosh
Sprint-4	Detection	VLGFSP - 17	If person is drowning, the system will ring an alarm to give signal	7	High	Santhosh
Sprint-4	Detection	VLGFSP - 18	As a User, I can detect the drowning person.	3	Medium	Surya
Sprint-4	Logout	VLGFSP - 19	As a User, I can logout the application.	2	Low	Mohammed Suhail

Velocity:

Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let’s calculate the team’s average velocity (AV) per iteration unit (story points per day)

For Sprint-1 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 8 / 2 = 4 V

For Sprint-2 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 14 / 2 = 7 V

For Sprint-3 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 16 / 2 = 8 V

For Sprint-4 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 12/ 2 = 6 V

TOTAL TEAM AVERAGE VELOCITY = 6.25

SPRINT DELIVERY SCHEDULE

A sprint schedule is a document that outlines sprint planning from end to end. It's one of the first steps in the agile sprint planning process and something that requires adequate research, planning, and communication.

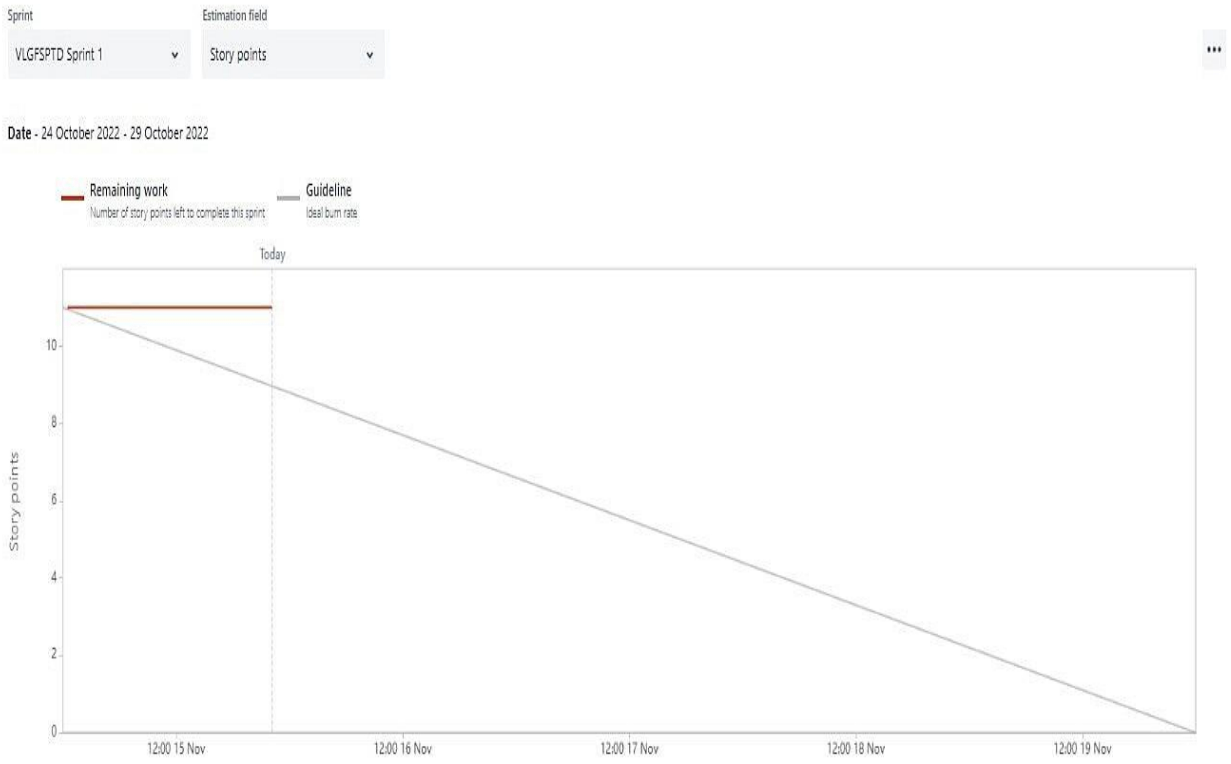
It is a graphical representation of work left to do versus time. The outstanding work is often on the vertical axis, with time along the horizontal. Burn down charts are a run chart of outstanding work. It is useful for predicting when all of the work will be completed.

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	2 Days	31 Oct 2022	02 Nov 2022	2	14 Nov 2022
Sprint-2	14	2 Days	02 Nov 2022	04 Nov 2022	1	14 Nov 2022
Sprint-3	16	2 Days	07 Nov 2022	09 Nov 2022	3	15 Nov 2022
Sprint-4	12	2 Days	13 Nov 2022	15 Nov 2022	2	16 Nov 2022

REPORTS FROM JIRA

BURNDOWN CHART

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.



Sprint: VLGFSPTD Sprint 1

Estimation field: Story points

Date - 24 October 2022 - 29 October 2022

CODING & SOLUTIONING (EXPLAIN THE FEATURES ADDED IN THE PROJECT ALONG WITH CODE)

FEATURE 1

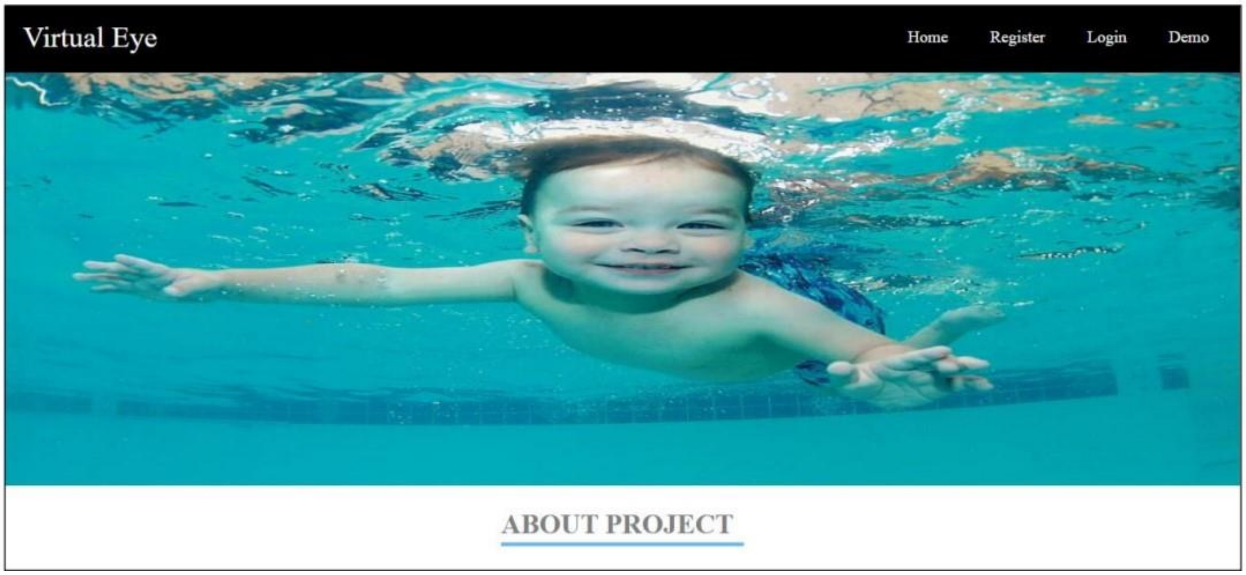
In order to manage a connection from a local system we must first initialize the connection by constructing a Cloudant client. We need to import the cloudant library.

IBM Cloud Identity & Access Management enables us to securely authenticate users and control access to all cloud resources consistently in the IBM Blue mix Cloud Platform.

1. Once a connection is established we can create a database, open an existing database.
2. Create a database as my_database.

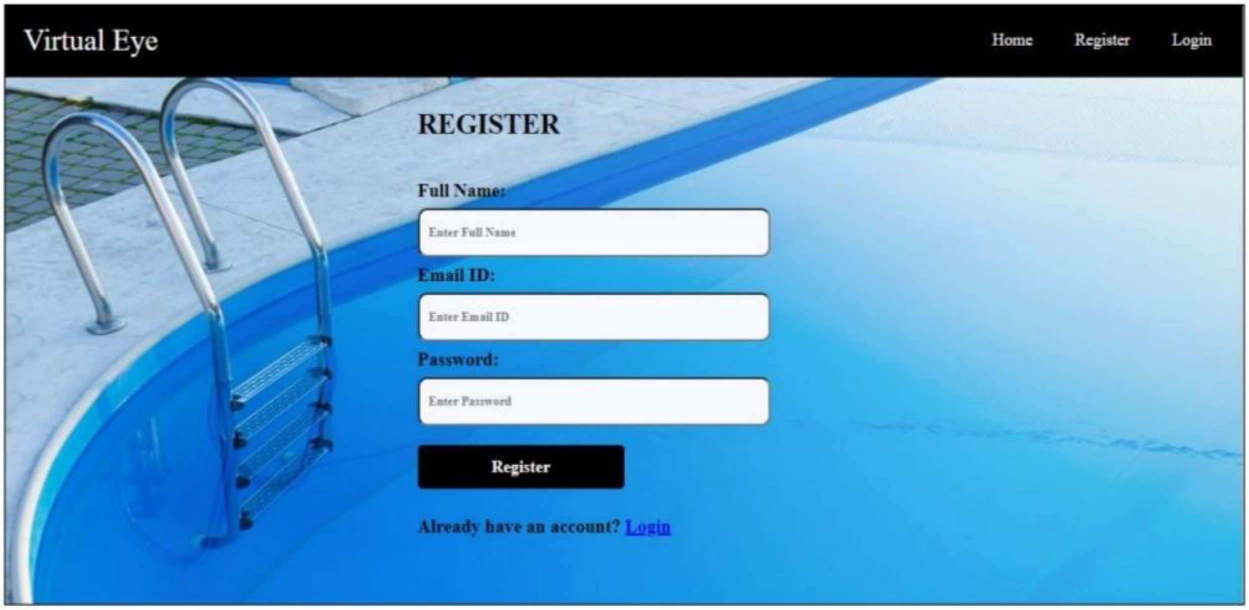
CODE

```
#default home page or route

@app.rout

e('/')     def

index():

   return render_template('index.html')

@app.route('/index.ht

ml') def home():

return render_template('index.html')
```



```
#registration page
@app.route('/regist
er') def register():
    return render_template('register.html')
@app.route('/afterreg', methods=['POST'])
def after reg():
    x = [x for x in request.form.values()]
    print(x)
    data = {
    'email': x[1], #Setting _id is optional
    'fullname': x[0],
    'password': x[2]
    }
    print(data)
    query = {'email': {'$eq': data['email']}}
    docs = my_database.get_query_result(query)
```

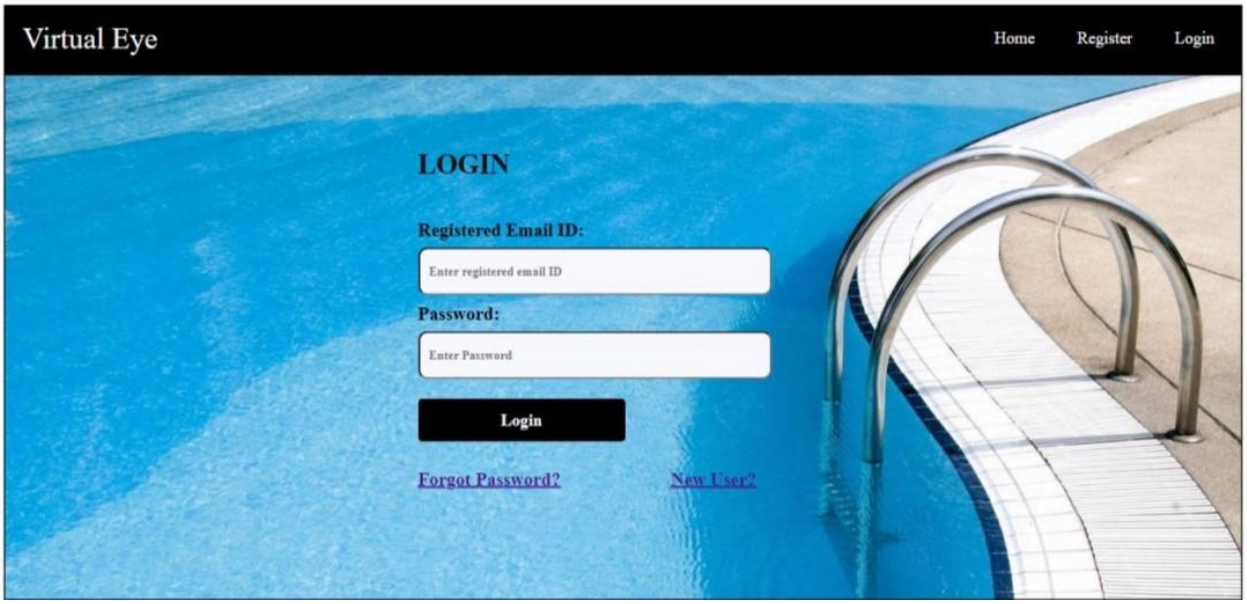
```
   print(docs)

   print(len(docs.all())

   if(len(docs.all())==0):

       url = my_database.create_document(data)

       #response = requests.get(url)

       return render_template('register.html', pred="Registration Successful, please login
using your details")

   else:

return render_template('register.html', pred="You are already a member,please login
using your details")
```



#login page

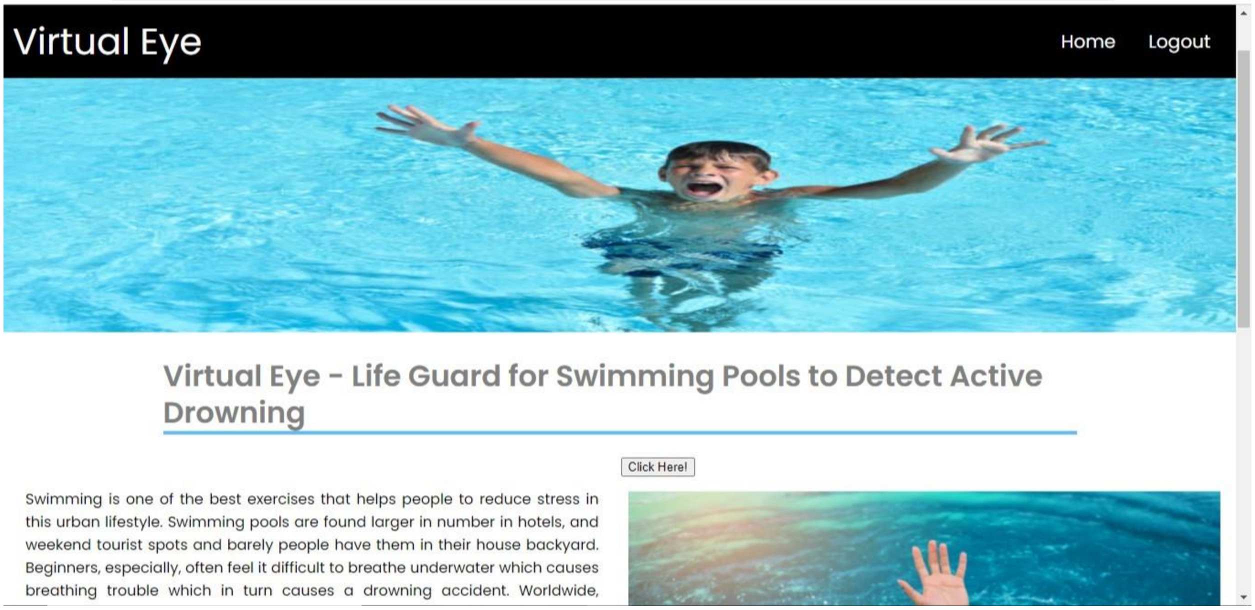
```
@app.route('/lo

gin') def login():

   return render_template('login.html')

@app.route('/afterlogin',methods=['POST'])

def after login():
```

```
user = request.form['email']

pass w = request.form['password']

print(user,pass w)

query = {'email': {'$eq': user}}

docs = my_database.get_query_result(query)

print(docs)

print(len(docs.all()))

if(len(docs.all())==0):

    return render_template('login.html', pred="The username is not found.")

else:

    if((user==docs[0][0]['email'] and passw==docs[0][0]['password'])):

        return redirect('/prediction')

       return render_template('login.html',pred="Invalid user")
```



#prediction page

```
@app.route('/prediction') def prediction(): return

render_template('prediction.html')
```

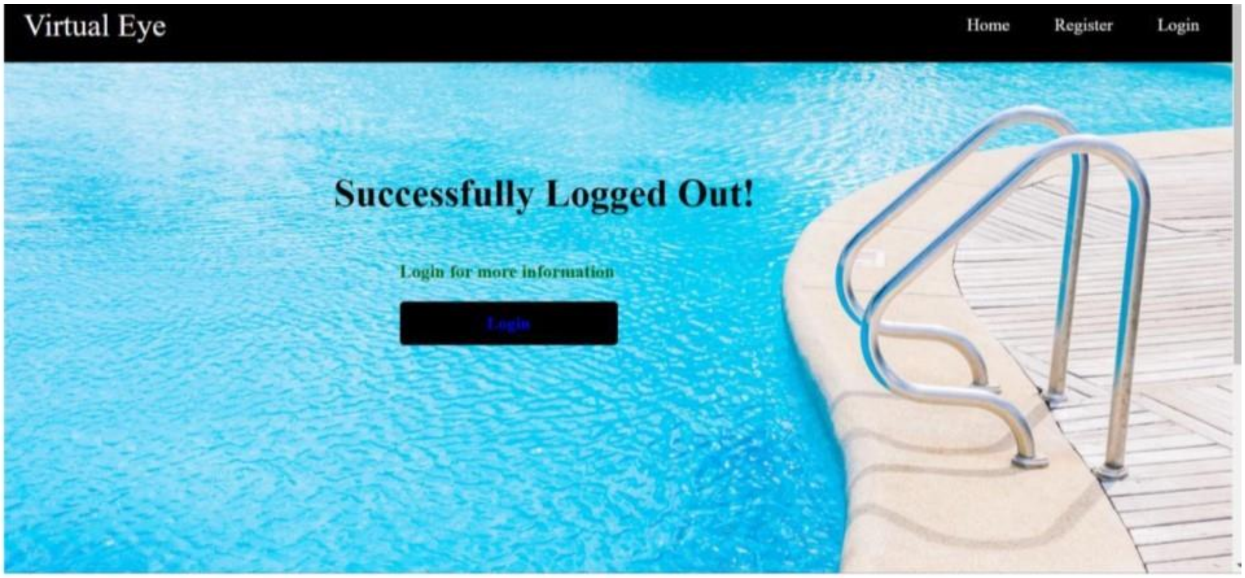


Virtual Eye - Life Guard for Swimming Pools to Detect Active Drowning

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 hotels, and weekend tourist spots and barely people have them in their house backyard. Beginners, especially, often feel it difficult to breathe underwater which causes breathing trouble which in turn causes a drowning accident. Worldwide,

#logout page

@app.route('/logout') def logout(): return



Successfully Logged Out!

Login for more information

render_template('logout.html')

CODE

from cloudant.client import Cloudant

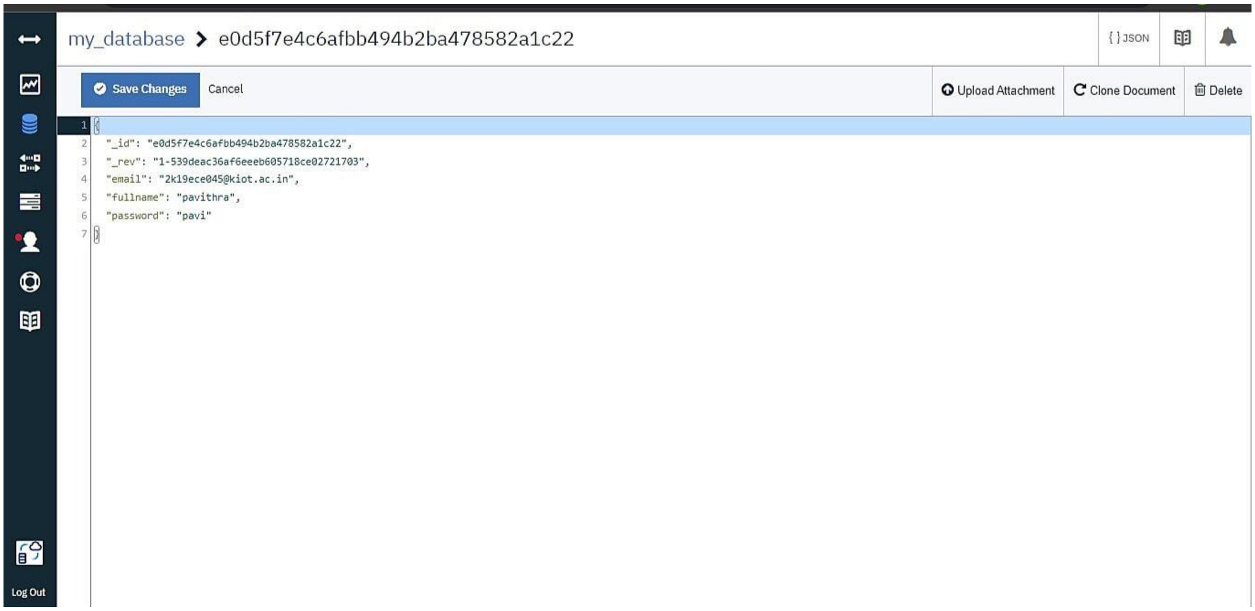
#Authenticate using an IAM API key

client = Cloudant.iam('a1885fb9-67af-469a-83e5-f1f78af7b19abluemix','dBqnX2HXtY8Dxm6Gd8nWvmiQ5R-f4HaM_seOK96b30zj', connect=True)

#Create a database using an initialized client

my_database = client.create_database('my_database')

OUTPUT



```
{
  "_id": "e0d5f7e4c6afbb494b2ba478582a1c22",
  "_rev": "1-539deac36af6eeeb605718ce02721703",
  "email": "2k19ece045@kiot.ac.in",
  "fullname": "pavithra",
  "password": "pavi"
}
```

FEATURE 2

CODE

import cvlib as cv from

cvlib.object_detection import draw_bbox

import cv2

import time import numpy as np from

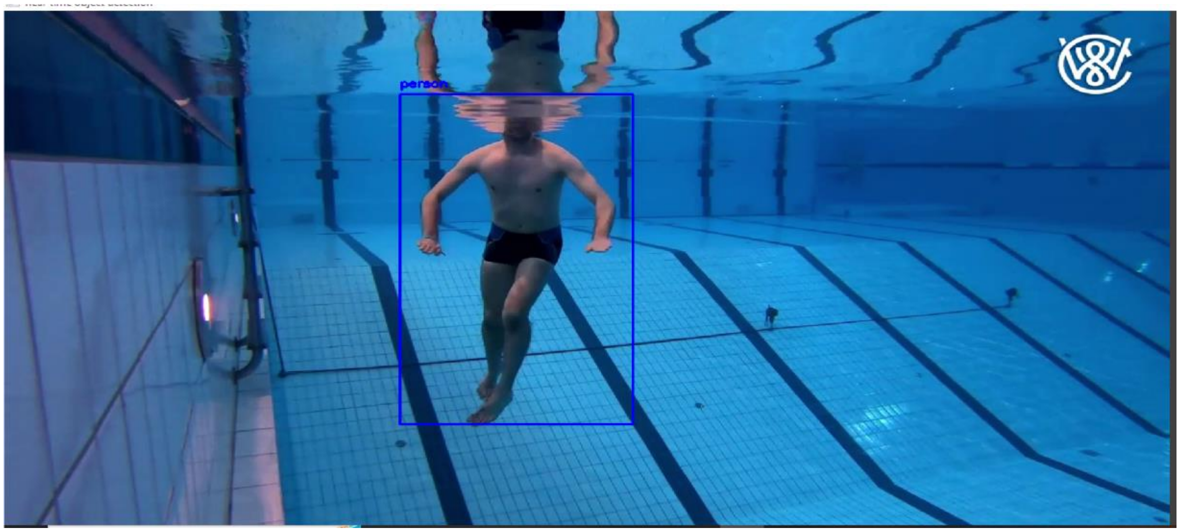
playsound import playsound webcam =

cv2.VideoCapture('sample2.mp4') t0 =

time.time() #gives time in seconds after 1970

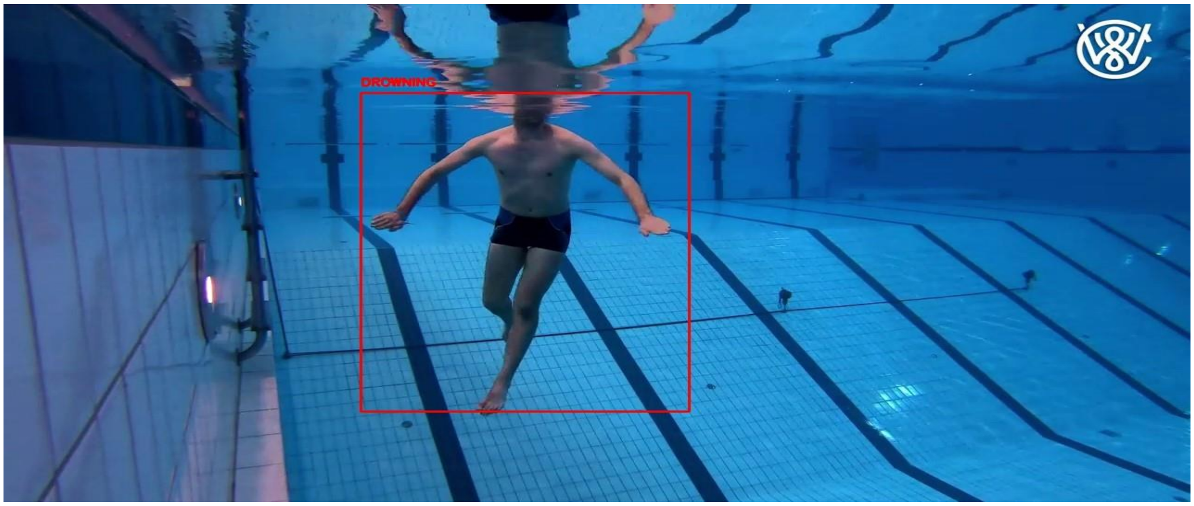
```
#variable dcount stands for how many seconds the person has been standing still
for centre0 = np.zeros(2) isDrowning = False
#this loop happens approximately every 1 second, so if a person doesn't move,
#or moves very little for 10seconds, we can say they are drowning
#loop through
frames while True:
    # read frame from webcam
status, frame =
webcam.read()     if not
status:
        print("Could not read frame")
        exit()
    # apply object detection
    bbox, label, conf = cv.detect_common_objects(frame)
    #simplifying for only 1 person
    #s       =       (len(bbox),       2)
if(len(bbox)>0):
       bbox0 = bbox[0]
       #centre = np.zeros(s)
        centre = [0,0]
       #for i in range(0, len(bbox)):
       #centre[i] =[(bbox[i][0]+bbox[i][2])/2,(bbox[i][1]+bbox[i][3])/2 ]
        centre =[(bbox0[0]+bbox0[2])/2,(bbox0[1]+bbox0[3])/2 ]
        #make vertical and horizontal movement variables
         hmov = abs(centre[0]-centre0[0])
         vmov = abs(centre[1]-centre0[1])
         #there is still need to tweek the threshold
         #this threshold is for checking how much the centre has moved
          x=time.time()
           threshold = 10
           if(hmov>threshold or vmov>threshold):
            print(x-t0, 's')
            t0 = time.time()
            isDrowning = False
            else:
                print(x-t0, 's')
                if((time.time() - t0) > 10):
                    isDrowning = True
              #print('bounding box: ', bbox, 'label: ' label ,'confidence: ' conf[0], 'centre: ', centre)
            #print(bbox,label ,conf, centre)
            print('bbox: ', bbox, 'centre:', centre, 'centre0:', centre0)
            print('Is he drowning: ', isDrowning)
            centre0 = centre
            # draw bounding box over detected objects
    out = draw_bbox(frame, bbox, label, conf,isDrowning)
    cv2.imwrite('image.jpg',out)
    if isDrowning:
        playsound(r'H:\PROJECT FILES\Drowning-Detector\alarm.mp3')
    #print('Seconds since last epoch: ', time.time()-t0)
    # display output
    cv2.imshow("Real-time object detection", out)
    # press "Q" to stop
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
```

#



```
release resources webcam.release() cv2.destroyAllWindows()
```

OUTPUT



TESTING

TEST CASES

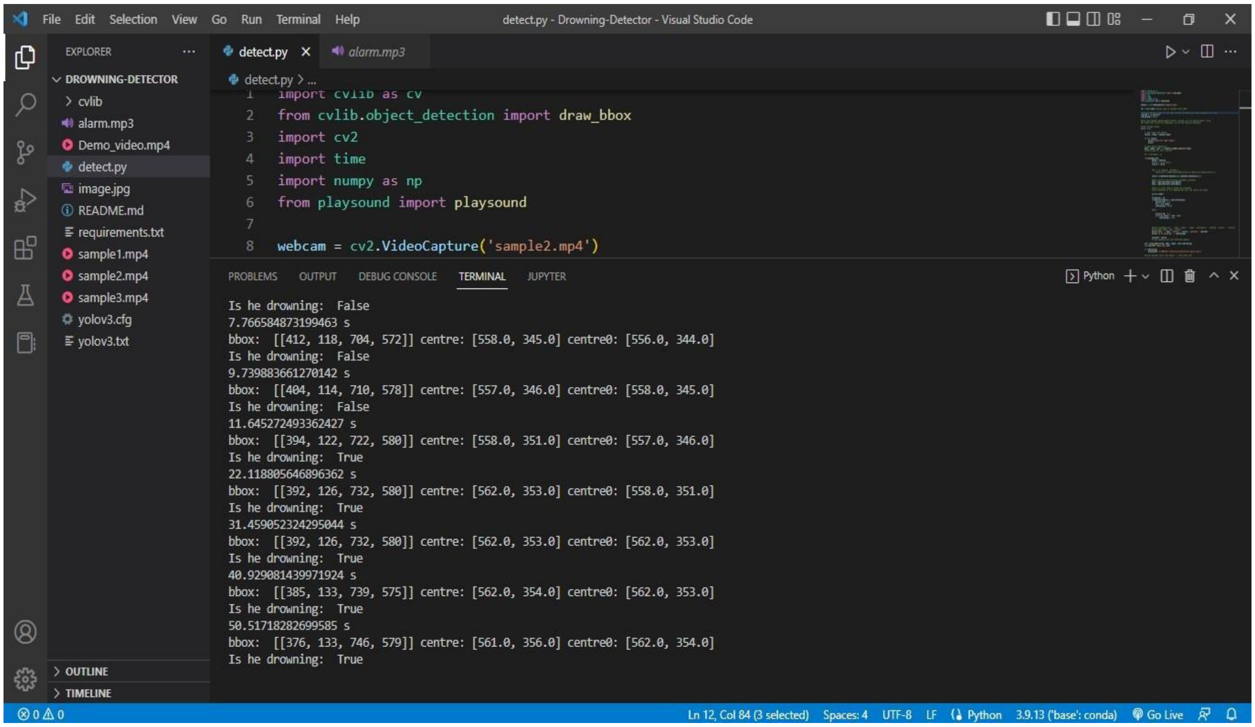
Test Case ID	Feature Type	Component	Test Scenario
Frontend & Cloudant TC	Website Functionality	Home Page	Verify it front-end are well connected and the results are expected.
Detection TC	Website detection	Python	Website built using HTML and designed using CSS.
Alarm Notification TC	mp3	Alarm File	Verify if alarm is triggered when drowning is detected.

USER ACCEPTANCE TESTING

Steps To Execute	Test Data	Expected Result	Status	Excecuted By
Register and login to our website.	app.py	User should receive the message as register and login sucessfully.	Pass	Karthikeyan S Mohammed Suhail K
After login it will move to the register page.	detect.py	Website should be live in the given URL.	Pass	Karthikeyan S Santhosh S

To see the demo video click the button.	detect.py	User should get the expected detection result and notify by alarm.	Pass	Karthikeyan S Surya V S

RESULTS


File Edit Selection View Go Run Terminal Help
detect.py - Drowning-Detector - Visual Studio Code
EXPLORER
DROWNING-DETECTOR
cvlib
alarm.mp3
Demo_video.mp4
detect.py
image.jpg
README.md
requirements.txt
sample1.mp4
sample2.mp4
sample3.mp4
yolov3.cfg
yolov3.txt
OUTLINE
TIMELINE
detect.py
alarm.mp3
detect.py > ...
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Python
Ln 12, Col 84 (3 selected) Spaces: 4 UTF-8 LF Python 3.9.13 ('base': conda) Go Live

```
import cvlib as cv
from cvlib.object_detection import draw_bbox
import cv2
import time
import numpy as np
from playsound import playsound

webcam = cv2.VideoCapture('sample2.mp4')
```

```
Is he drowning:  False
7.766584873199463 s
bbox:  [[412, 118, 704, 572]] centre: [558.0, 345.0] centre0: [556.0, 344.0]
Is he drowning:  False
9.739883661270142 s
bbox:  [[404, 114, 710, 578]] centre: [557.0, 346.0] centre0: [558.0, 345.0]
Is he drowning:  False
11.645272493362427 s
bbox:  [[394, 122, 722, 580]] centre: [558.0, 351.0] centre0: [557.0, 346.0]
Is he drowning:  True
22.118805646896362 s
bbox:  [[392, 126, 732, 580]] centre: [562.0, 353.0] centre0: [558.0, 351.0]
Is he drowning:  True
31.459052324295044 s
bbox:  [[392, 126, 732, 580]] centre: [562.0, 353.0] centre0: [562.0, 353.0]
Is he drowning:  True
40.929081439971924 s
bbox:  [[385, 133, 739, 575]] centre: [562.0, 354.0] centre0: [562.0, 353.0]
Is he drowning:  True
50.51718282699585 s
bbox:  [[376, 133, 746, 579]] centre: [561.0, 356.0] centre0: [562.0, 354.0]
Is he drowning:  True
```

Once we have the working drowning detection model we can feed live video footage of the swimming pool to it so that it can keep detecting continuously for any drowning activities. If drowning is detected it will be highlighted on the system screen as well as alarms will be raised to alert security guards so that they can initiate rescue. This application will detect the drowning victim within in minute. So we can reduce the drowning accident by using our application.

ADVANTAGES & DISADVANTAGES

Advantages:

1. It represents an additional level of safety and protection for swimmers.
2. It ensures effective and reliable drowning detection by limiting the number of alarms generated by disturbance factors.
3. Prevents drowning accidents by improving the rescue time of the lifeguards.
4. The VirtualEye system operates in compliance with privacy laws.
5. It increases visibility in areas of the pool that are difficult for lifeguards to see, such as the bottom and corners.
6. It can be installed in any type of pool and construction variant.
7. Most efficient method for drowning detection system.
8. Continuous monitoring without any interruptions.

CONCLUTION

Life safety in water has been a concern for many centuries. Latest technology advancements has enabled us to come up with effective drowning detection systems. However many of those solutions are costly and limited to few. Survey reports show us that highest numbers of deaths are reported in low and middle income countries. The survey report also mentions the children have the largest death ratio compared to adults. Also the deaths reported in these incidents are more from open water bodies than closed water bodies like swimming pools. The solution described above will be able to address these issues.

FUTURE SCOPE:

Availability of better dataset, modern methodologies, and technologies with high computational power accompanied by high-quality surveillance cameras, will help to improve the accuracy of drowning detection & even can be used in adverse conditions. After the implementation of all these essentials, this system can be used on the swimming pool.

FUTURE ENHANCEMENT:

1. 24/7 Technical support is needed.
2. Not all people trust these kind of system.
3. Swimming pool need regular maintenance.

Require, extensive calibration and expertise.

APPENDIX:

13.1 Source Code:

```
from flask import Flask, request, render_template, redirect, url_for from

detect import start_test

#loading the model


from cloudant.client import Cloudant


#Authenticate using an IAM API key

client = Cloudant.iam('a1885fb9-67af-469a-83e5-
f1f78af7b19abluemix','dBqnX2HXtY8Dxm6Gd8nWvmiQ5R-f4HaM_seOK96b30zj',
connect=True)


#Create a database using an initialized client

my_database = client.create_database('my_database')
```

```
app=Flask(__name__)


#default home page or route
@app.rou
te('/') def
index():
    return render_template('index.html')


@app.route('/index.ht
ml') def home():
    return render_template('index.html')


#registration
page@app.route('/r
egister') def
register():
    return render_template('register.html')


@app.route('/afterreg', methods=['POST'])
def after reg():
    x = [x for x in request.form.values()]
    print(x)
    data = {
    'email': x[1], #Setting _id is optional
    'fullname': x[0],
    'password': x[2]
    }
    print(data)


   query = {'email': {'$eq': data['email']}}




    docs = my_database.get_query_result(query)     print(docs)


    print(len(docs.all()))


    if(len(docs.all())==0):
        url = my_database.create_document(data)
        #response = requests.get(url)
        return render_template('register.html', pred="Registration Successful, please
login using your details")
    else:
        return render_template('register.html', pred="You are already a member,please
login using your details")


#login
page@app.route
('/login') def
login():
    return render_template('login.html')
```

```
@app.route('/afterlogin',methods=['POST'])
def after login():
    user = request.form['email']
    pass = request.form['password']
    print(user,pass)

    query = {'email': {'$eq': user}}


    docs = my_database.get_query_result(query)
    print(docs)


    print(len(docs.all()))


    if(len(docs.all())==0):
        return render_template('login.html', pred="The username is not found.")
    else:
        if((user==docs[0][0]['email'] and passw==docs[0][0]['password'])):
return redirect('/prediction')
        else:
           return render_template('login.html',pred="Invalid user")
#prediction
page@app.route('/pre
diction') def
prediction():
    return render_template('prediction.html')
```

```
#logout
page@app.route('
/logout') def
logout():
    return render_template('logout.html')

@app.route('/start',
methods=['POST']) def start():
    start_test()
    return redirect('/prediction')


if __name__=='__main__':
    app.run(debug=True)
```

GitHub link:

https://github.com/IBM-EPBL/IBM-Project-38141-1660372972

Project Demo Link:

REFERENCES

PAPER 1 - An Improved Detection Method of Human Target at Sea Based on Yolov3.

Details of the paper - 2021 IEEE https://ieeexplore.ieee.org/document/9118248

PAPER 2 - A novel drowning detection method for safety Of swimmers.

Details of the paper - 2018 IEEE

https://ieeexplore.ieee.org/document/8771844

PAPER 3 - Automated Vision-based Surveillance System to Detect Drowning Incidents in Swimming Pools.

Details of the paper - 2020 IEEE https://ieeexplore.ieee.org/document/9342056

PAPER 4 - Swimmer motion analysis with application to drowning detection.

Details of the paper - 2002 IEEE https://ieeexplore.ieee.org/document/1011439

PAPER 5 - IOT Based Safety Enhanced Swimming Pool with Embedded Techniques to reduce drowning accidents. Details of the paper - 2020 IEEE

https://ieeexplore.ieee.org/document/9825955