

Virtual-Eye

**Life Guard For Swimming Pools To Detect Active
Drowning**

TEAM ID - PNT2022TMID43878

IBM-Project-51477-1660979835

TEAM MEMBERS- MOHAMMED SHAMAL

BARSHAD BASHA

JISHIN M

ABHINANDH K

1. INTRODUCTION

1.1 Project Overview

1.2 Purpose

2. LITERATURE SURVEY

2.1 Existing problem

2.2 References

2.3 Problem Statement Definition

3. IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

3.2 Ideation & Brainstorming

3.3 Proposed Solution

3.4 Problem Solution fit

4. REQUIREMENT

4.1 Functional requirement

4.2 Non-Functional requirements

5. PROJECT DESIGN

5.1 Data Flow Diagrams

5.2 Solution & Technical Architecture

5.3 User Stories

6. PROJECT PLANNING & SCHEDULING

6.1 Sprint Planning & Estimation

6.2 Sprint Delivery Schedule

6.3 Reports from JIRA

7. CODING & SOLUTIONING

7.1 Feature 1

7.2 Feature 2

8. TESTING

8.1 Test Cases

8.2 User Acceptance Testing

9. RESULTS

9.1 Performance Metrics

10. ADVANTAGES & DISADVANTAGES

11. CONCLUSION

12. FUTURE SCOPE

13. APPENDIX

1.INTRODUCTION

1.1 OVERVIEW

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, 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. To overcome this conflict, a meticulous system is to be implemented along the swimming pools to save human life.

The AI-powered system sends wi-fi alerts and sounds the alarm when anyone in the water moves in a manner that indicates panic.

1.2 PURPOSE

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 analysing the video feeds to detect any anomalies.

But as 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.

When a group of people are enjoying the space, and children are going in and out of the pool, alerts will not be sent. Should someone sink to the bottom of the pool and remain motionless for more than 15 seconds, a loud alarm will sound on the home unit as well as the poolside unit, indicating an emergency. Alerts will also be set to paired devices,

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.

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

Swimming pools are beacons of fun for summer adventurers, with children and adults alike logging plenty of splash time during the warm-weather months. But where there's fun, there can also be a risk of danger—and swimming is no different.

According to the World Health Organization, drowning is the leading cause of unintentional death in the world, with 372,000 drowning deaths reported annually

According to the CDC, various factors influence a person's chance of drowning, including:

Subpar Swim Skills: Research indicates that children who receive swimming instruction are less likely to drown or suffer a swimming-related injury. Learning swimming pool safety tips also helps.

Barriers: Four-sided fencing around swimming pools reduces the risk of drowning by about 80 percent compared to swimming pools secured by three-sided fencing.

Lack of Supervision: It almost goes without saying: Children are more likely to drown or suffer a swimming-related injury when they are in a pool unsupervised. But a swimming-related death or injury can also happen when there is supervision. According to statistics, 77 percent of those involved in a home drowning accident had been missing for no more than five minutes when they were found in the swimming pool—and 70 percent weren't expected to be in or near the pool at that time.

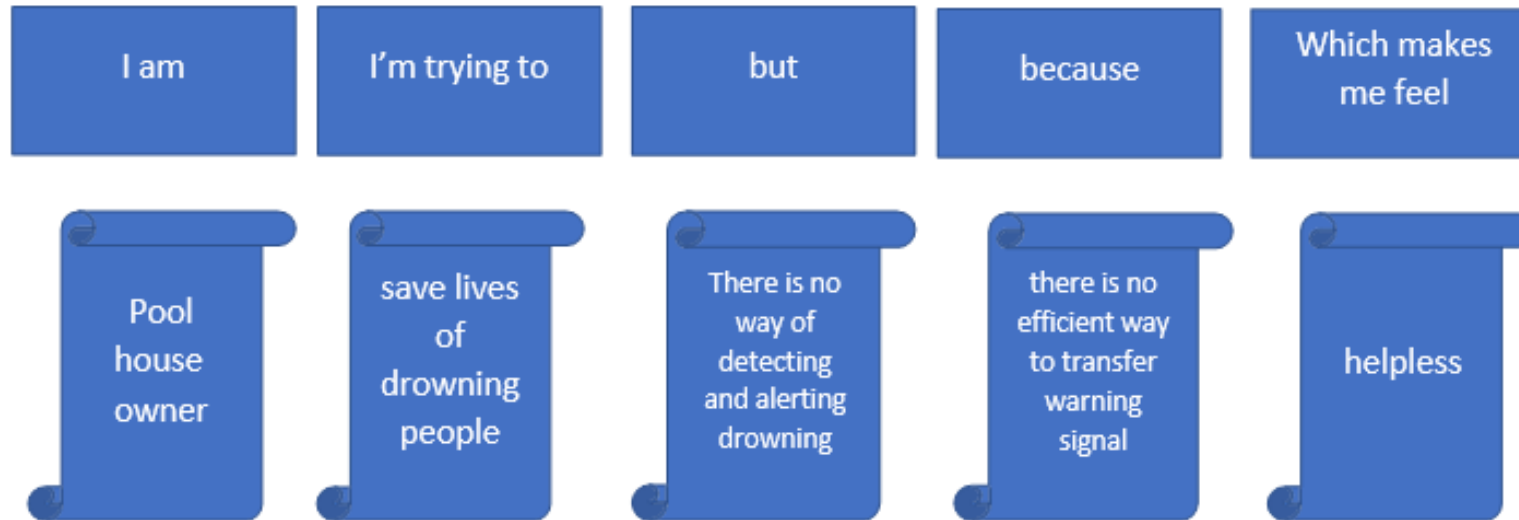
Alcohol Use: There's a reason you see those "alcohol prohibited" signs outside both residential and public swimming pools: Statistically, alcohol is attributed as a factor in as many as 70 percent of all water recreation injuries and deaths.

Seizure Disorder: For people affected by seizure disorder, drowning is the most common cause of injury or death.

2.2 REFERENCES

S no	Title	Year	Author	Method / Approach	Advantages	Disadvantages
1	Identification of Drowning Victims in Freshwater Bodies using Drift Prediction and Image Processing based on Deep Learning	2022	Anjana Unnikrishnan, Roshni A T, Anusha P R, Anju M Vinny, Anuraj C K	Using multiple sensor data in underwater human rescue detection system to spot drifting and drowning person in a natural water eco system. The water flow sensor which is attached to the portable device calculates the drift distance and tracks drowning person.	The Approach detected human drifting and drowning up to a range of 5m in water bodies. The final result achieved an average of 82.10% accuracy.	The performance of the model depends on the nature of the water body concerned as the drift distance is different for different water eco systems.
2	Falling and Drowning Detection framework using smartphone sensors	2022	Abdullah Alqahtani, Shtwai Alsubai, Sidra Abbas	Presents the novel ambient assistive framework by perceiving input from smartphone sensors such as accelerometer, gyroscope, magnetometer and GPS that provide accurate readings of movement of individual's body.	It detects falling, drowning and routine actions with good accuracy of 98%.	Limitation of this study is that WiFi and other cellular signal does not work properly underwater.
3	Video Based Drowning Detection System	2021	Pavithra P, Nandini S, Nanthana A, Noor Tabreen Aslam, Praveen Kumar P	The proposed system structure here comprises of a raspberry pi (Single Board Computer) equipped with a USB camera for taking the live feed from the pool area. The system also covers the alerting phenomena using a buzzer so that necessary actions are taken intermittently without any delay	Alerting a drowning state is done without any delay here, GPIO system for alerting and short message service used in cohesion with a raspberry pi computer makes this possible	A working implementation of this module is quite extensive to implement, and multiple hardware components working together in proximity of water can also lead to some malfunctioning
4	Drowning detection system on coastal lines using image processing techniques and neural network	2019	Kamyar Shiuee, Fardin Rezaei	The features of images are extracted through image processing techniques and background omission.	Neural network has detected drowning cases with precision of 94-96%.	Challenges include high noise of sea images and the size of the drowned from far distance.
5	An early drowning detection system for IoT	2018	M.S. Muhammad Ramdhan, Muhammad Ali, Eberechukwu Paulson, Ghazali N. Effiyana	It gives an early alarm to the guardians if the detector triggered an abnormal heartbeat and the victims are submerged under water for a long time.	The lifeguard and parents are able to monitor remotely the condition of the swimmers and alerted in real time when there is panic attack (near drowning).	System needs to be waterproof to ensure that the components are not damaged.

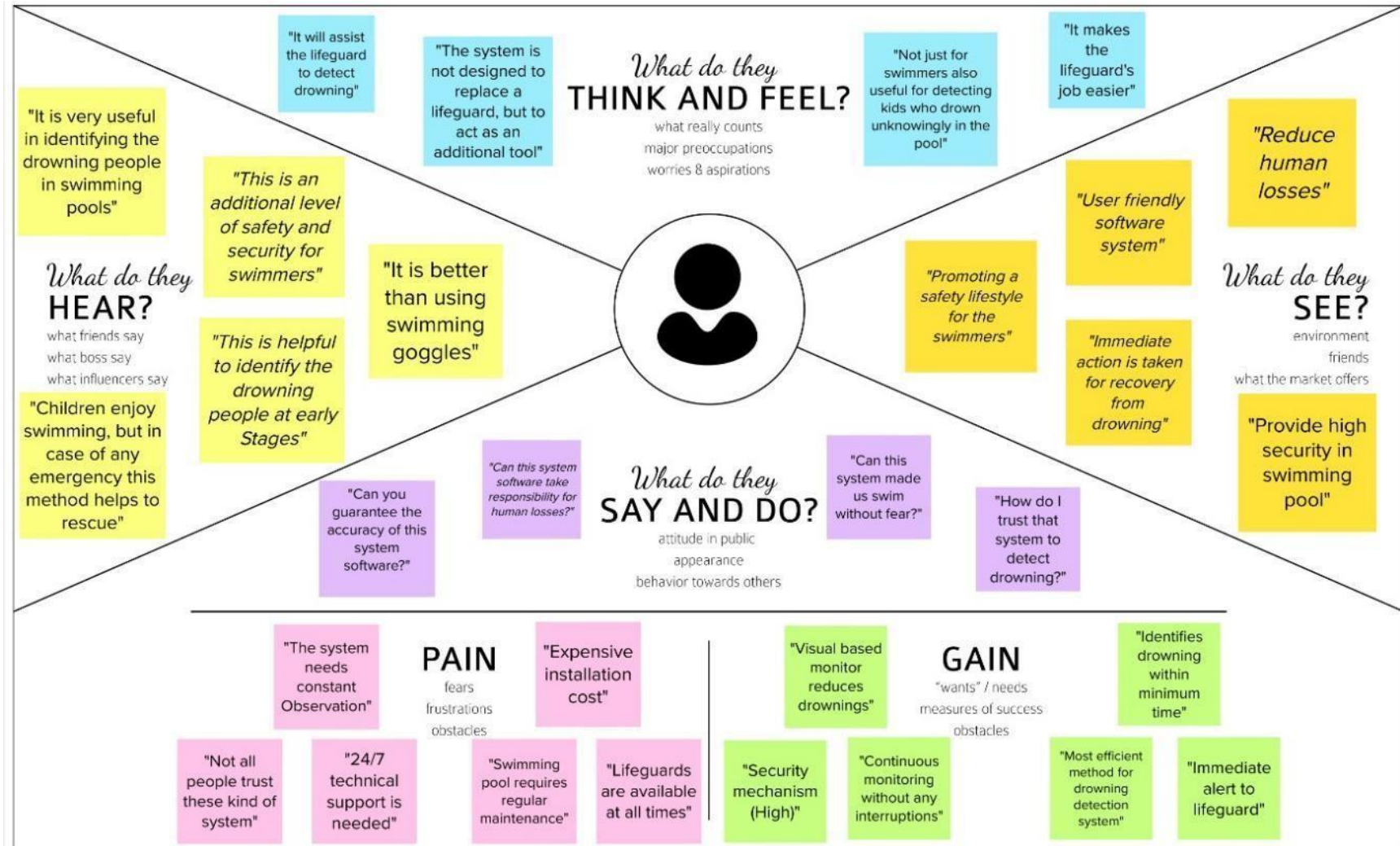
2.1 PROBLEM STATEMENT DEFENITION



Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	Pool house owner	save lives of drowning people	There is no way of detecting and alerting drowning	there is no efficient way to transfer warningsignal	helpless

3.IDEATION & PROPOSED SOUTION

3.1 EMPATHY MAP CANVAS



3.2 IDEATION & BRAINSTORMING



Brainstorm And Idea Prioritization

In this our objective is to achieve a good foundation to start our project. Further we will combine our ideas and thoughts and end up with feasible solution

Ground Rules

1. Think innovative idea
2. Make your points clear and understandable
3. Adopt modern technology
4. Think from all possible angles

1

Choose your best "How Might We" Questions

Share the top 5 brainstorm questions that you created and let the group determine where to begin by selecting one question to move forward with based on what seems to be the most promising for idea generation in the areas you are trying to impact.

10 minutes

How might we develop VirtualEye - Life Guard For Swimming Pools To Detect Active Drowning?

QUESTION

How might we integrate the modern technology and Algorithm?

QUESTION

How might we train to use YOLO Model?

QUESTION

How might we analyse the video feeds to detect anomalies?

QUESTION

How might we ensure the security using this system?



Need some inspiration?
Here is a limited version of this template to kickstart your work.

[Open example](#)



→



→

2

Brainstorm solo

Have each participant begin in the "solo brainstorm space" by silently brainstorming ideas and placing them into the template. This "silent-storming" avoids group-think and creates an inclusive environment for introverts and extroverts alike. Set a time limit. Encourage people to go for quantity.

10 minutes

Shamal

Testing tools and equipments should be implemented

Proper Electronic devices should be chosen

customized and specific model should be developed

use advanced camera to process good quality frames

Backup plan should adopted for critical situations

uninterrupted power supply to operate system

Jishin

To get optimized video under water footage

using the video feed only for detection and discarded later

Able to process the detection of drowning process and starting the rescue

using renewable energy source

Back up power supply should be available in case of emergency situation

Increase the survival ratio

Basha

Ensure proper communication system to report

Awareness about the system functioning to people

System should be regularly monitored by panel members

AI should be trained with more sample for good results

The system should be eco-friendly and human friendly

Accuracy of the system should be checked periodically

Abhinand

Accurate and good quality system camera should be used for more detection

what happens if somebody pushes another person on purpose?

Better plans should be implemented if animals were encountered in the pool

Installation of system should be adjustable to environment condition

Health condition should be checked for better health status of swimmer

Periodic checking of electronic entity to prevent hazards in swimming pools



→



→



→

3

Brainstorm as a group

Have everyone move their ideas into the "group sharing space" within the template and have the team silently read through them. As a team, sort and group them by thematic topics or similarities. Discuss and answer any questions that arise. Encourage "Yes, and..." and build on the ideas of other people along the way.

15 minutes

TIP

You can use the **Voting session** tool above to focus on the strongest ideas.



4

Decide your focus

Give each person two icons to vote which idea should your team focus on.

5 minutes



→

After you collaborate

A brainstorm like this typically results in a handful of promising ideas that you can carry forward and act upon.

Quick add-ons

- A Cluster related ideas**
Look for patterns or similarities in the standout ideas. Could any be combined together to form a stronger concept? Cluster similar ideas and label each cluster with a theme.
- B Vote on the most promising ideas**
Narrow your focus to only the strongest few ideas by holding a **Voting Session**. Give each person 2 votes

Keep moving forward

- 2x2 Prioritization matrix**
Build shared understanding and make collective decisions for moving ideas forward.
[Open the template →](#)
- Storyboarding**
Show existing and/or future consumer experiences through the act of sketching.
[Open the template →](#)
- Pre-mortem**
Harness the collective experience and wisdom of the team, before the project even starts.
[Open the template →](#)

[Share template feedback](#)



3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Swimming pools are generally places of fun and healthy exercise, but they can be deadly as well. Even with a lifeguard observer on duty, swimmers may still have trouble in underwater or in parts of the pool beyond the lifeguard's field of view.
2.	Idea / Solution description	In this project, we use Artificial Intelligence. We install the cameras in underwater to detect the drowning people. Using deep learning, image can be recognized. If the image is detected, it triggers the alarm to alert the Life Guard who rescue the 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 PROBLEM SOLUTION FIT

Define CS, fit into CC	<div>1. CUSTOMER SEGMENT(S)<div>CS</div></div> <div>Who is your customer? I.e. working parents of 0-5 y.o. kids</div> <div>Designed for whom has to guarantee every day the safety in public and intensive-use pools, VirtualEye -LifeGuard detects potential drownings and promptly notifies you.</div>	<div>6. CUSTOMER CONSTRAINTS<div>CC</div></div> <div>What constraints prevent your customers from taking action or limit their choices of solutions? I.e. spending power, budget, no cash, network connection, available devices.</div> <div>1. Teach basic swimming and water safety skills to school-age children aged 6 years and older.</div> <div>2. Training bystanders in safe rescue and resuscitation.</div> <div>3. Use life guard while swimming</div>	<div>5. AVAILABLE SOLUTIONS<div>AS</div></div> <div>Which solutions are available to the customers when they face the problem or need to get the job done? What have they tried in the past? What pros & cons do these solutions have? I.e. pen and paper is an alternative to digital notetaking</div> <div>1. There are many actions to prevent drowning. Installing barriers (e.g. covering wells, using doorway barriers and playpens, fencing swimming pools etc.) to control access to water hazards, or removing water hazards entirely greatly reduces water hazard exposure and risk.</div>	Explore AS, differentiate
	<div>2. JOBS-TO-BE-DONE / PROBLEMS<div>J&P</div></div> <div>Which jobs-to-be-done (or problems) do you address for your customers? There could be more than one; explore different sides.</div> <div>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.</div>	<div>9. PROBLEM ROOT CAUSE<div>RC</div></div> <div>What is the real reason that this problem exists? What is the back story behind the need to do this job? I.e. customers have to do it because of the change in regulations.</div> <div>1. The most common cause of drowning is not knowing how to swim. Many adults and children will attempt to get into the water without proper swim training.</div> <div>2. Drowning is the 3rd reason for the highest unintentional deaths, and that's why it is necessary to create trustable security mechanisms.</div>	<div>7. BEHAVIOUR<div>BE</div></div> <div>What does your customer do to address the problem and get the job done? I.e. directly related: find the right solar panel installer, calculate usage and benefits; indirectly associated: customers spend free time on volunteering work (I.e. Greenpeace)</div> <div>In order to quickly help lifesavers judge whether people are drowning in the swimming pool,</div>	
Focus on J&P, tap into BE, understand RC	<div>3. TRIGGERS<div>TR</div></div> <div>What triggers customers to act? I.e. seeing their neighbour installing solar panels, reading about a more efficient solution in the news.</div> <div>Drowning Awareness through the Advertisements and other social medias like Instagram , Facebook, LinkedIn & News etc</div>	<div>10. YOUR SOLUTION<div>SL</div></div> <div>If you are working on an existing business, write down your current solution first, fill in the canvas, and check how much it fits reality. If you are working on a new business proposition, then keep it blank until you fill in the canvas and come up with a solution that fits within customer limitations, solves a problem and matches customer behaviour.</div> <div>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.</div>	<div>8. CHANNELS of BEHAVIOUR<div>CH</div></div> <div>8.1 ONLINE What kind of actions do customers take online? Extract online channels from #7 Search engines are considered to be the best marketing channel for acquiring new leads for drowning detection system. Social media continues to grow every year as a powerhouse marketing channel.</div>	Focus on J&P, tap into BE, understand RC
	<div>4. EMOTIONS: BEFORE / AFTER<div>EM</div></div> <div>How do customers feel when they face a problem or a job and afterwards? I.e. lost, insecure > confident, in control - use it in your communication strategy & design. 1. (i) struggle to keep the airway clear of the water, (ii) initial submersion and breath-holding, (iii) aspiration of water, (iv) unconsciousness, (v) cardio-respiratory arrest and (vi) death – inability to revive. 2. In order to quickly help lifesavers judge whether people are drowning in the swimming pool,</div>		<div>8.2 OFFLINE What kind of actions do customers take offline? Extract offline channels from #7 and use them for customer development. Local media advertising- Advertising on both television and radio is still very effective in creating brand awareness about drowning in the swimming pool</div>	
Identify strong TR & EM				Extract online & offline CH of BE

4.REQUIREMENT ANALYSIS

4.1 FUNCTIONAL REQUIREMENTS

Following are the functional requirements of the proposed solution.

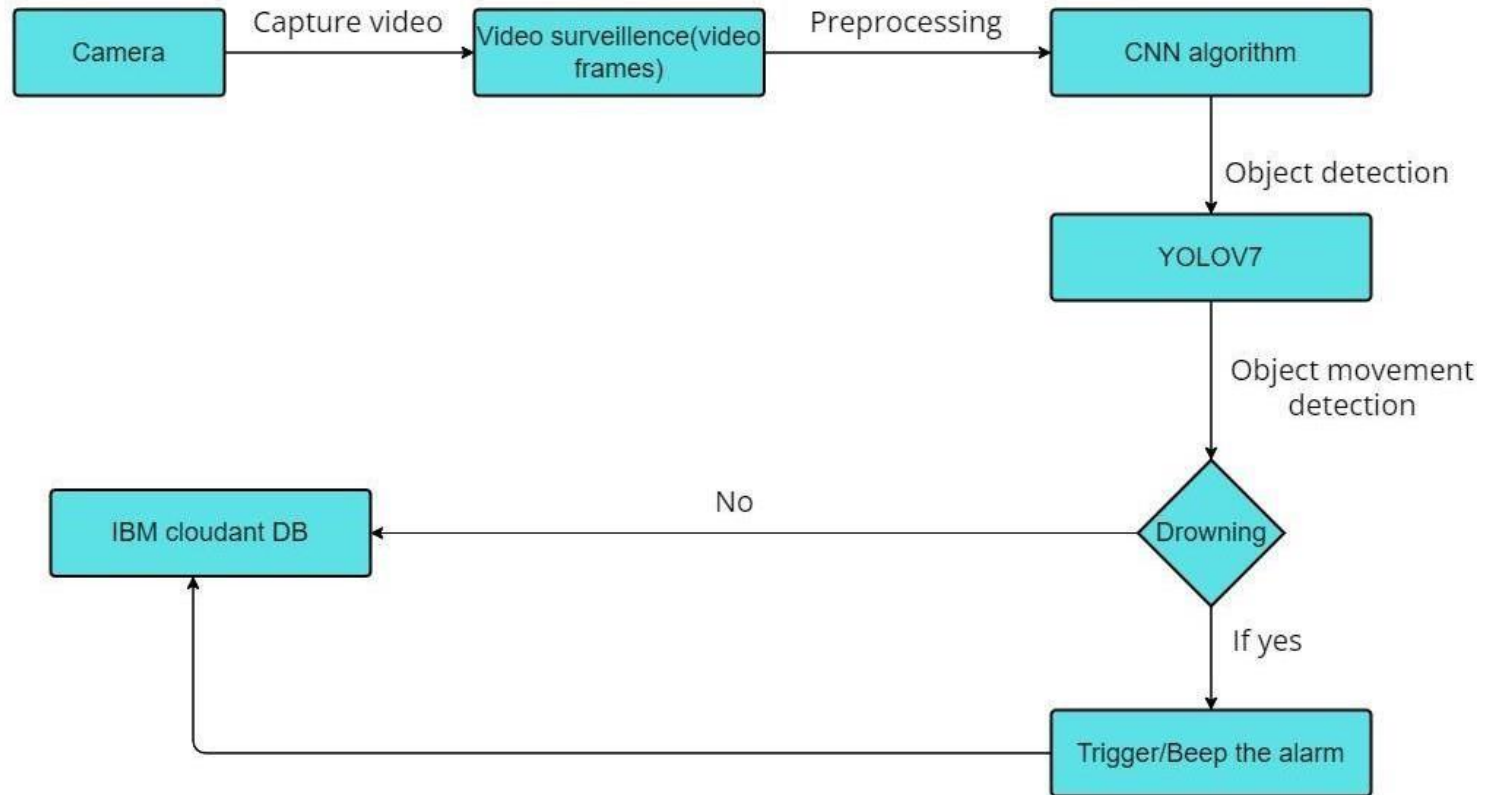
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Installation	Needed to be fixed under the water without Creatingany disturbance to the people in the swimming pool.
FR-2	User registration	Register via Email/Phone number and get verified forfurther use
FR-3	Deduction	Either not moving or in unconscious state
FR-4	Support	Take swim tubes or take the help of rescuer.
FR-5	Alert	Set alarm and send message through the application tolife guard.
FR-6	Output	Vision based monitor Image, position and movementdetection Drowning is detected Resue drowning people by Life Guard

4.2 NON-FUNCTIONAL REQUIREMENTS

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	Lifeguards should be aware of the alert message to save the life of the swimmer.
NFR-3	Reliability	Virtual eye lifeguard 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 is detected as drowning
NFR-5	Availability	Equipment and accessories include lifesaver rings, inflatable vests, a Shepherd's Crook, life hooks, spine boards, rescue tubes, and a first aid kit. Remember to keep them accessible to quickly pull someone from the water safely.
NFR-6	Scalability	Virtual eye lifeguard detects potential drownings and promptly notifies you. It features the latest artificial intelligence technology and adapts to the needs of the user.

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAMS

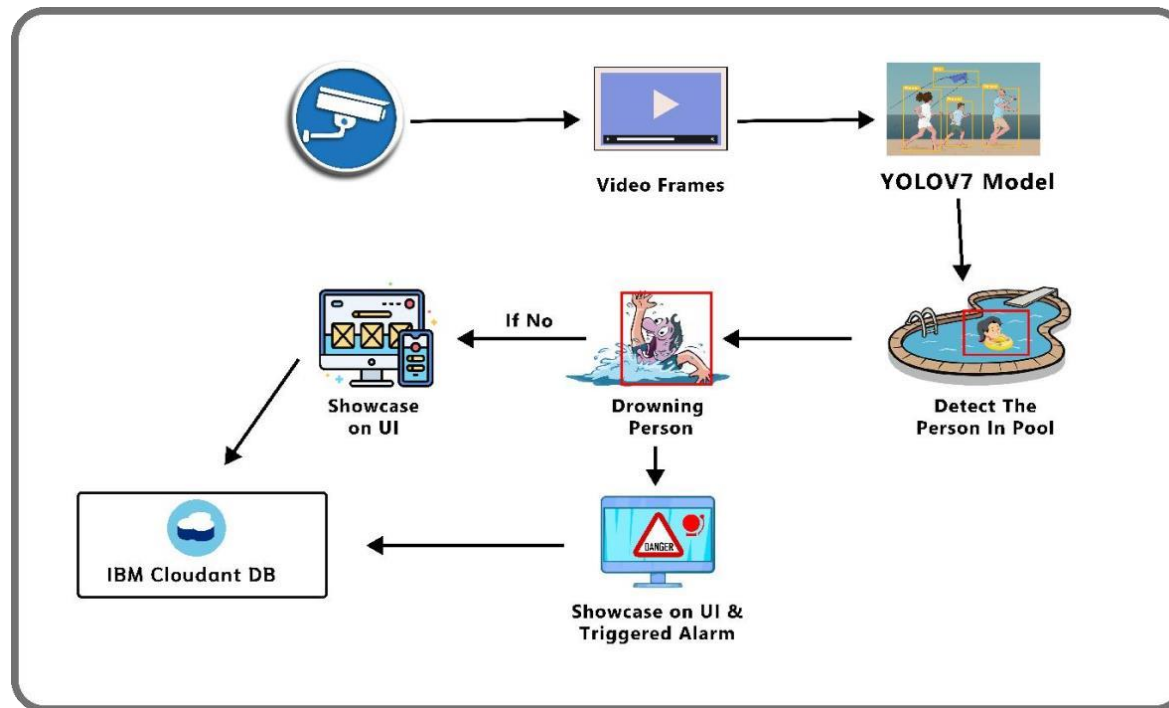


5.2 SOLUTION & TECHNICAL ARCHITECTURE

SOLUTION ARCHITECTURE:

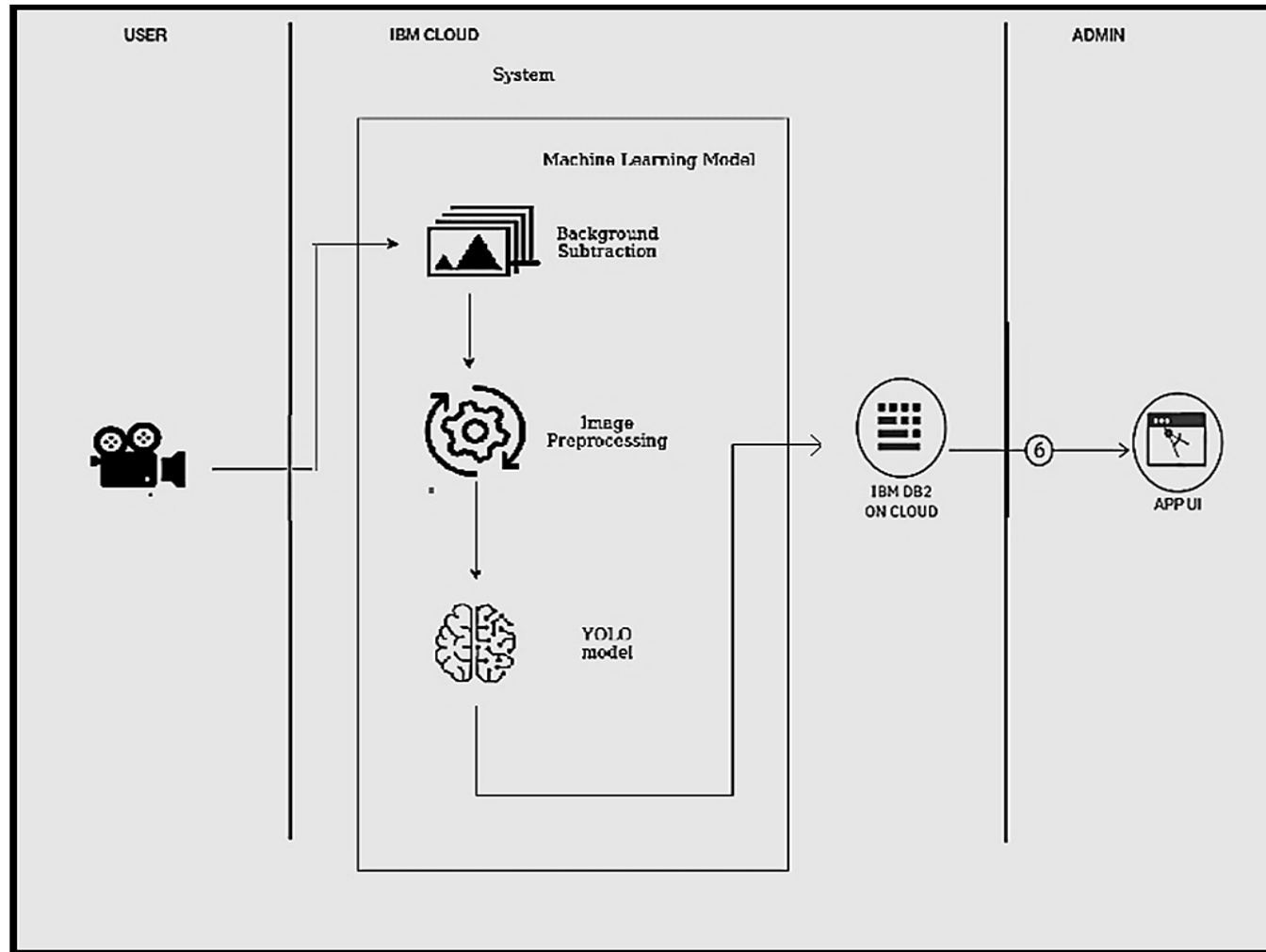
Solution architecture is a complex process – with many sub-processes – that bridges the gap between business problems and technology solutions. Its goals are to:

- Find the best tech solution to solve existing business problems.
- Describe the structure, characteristics, behavior, and other aspects of the software to project stakeholders.
- Define features, development phases, and solution requirements.
- Provide specifications according to which the solution is defined, managed, and delivered.



TECHNICAL ARCHITECTURE:

The Deliverable shall include the architectural diagram as below and the information as per the table1 & table 2



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 (Supervisor)	Installation	USN-1	They set the camera and install and configure the system in swimmingpools	The software is installed and cameras are setup	High	Sprint-1
	Pre-processing	USN-2	Train and test the model	Train the model byusing datasets	High	Sprint-1
	Detection of drowning	USN-3	The swimmers can be monitored by cameras	Camera surveillance	High	Sprint -2
		USN-4	Swimmers can be detected through their actions	Detection of drowning	High	Sprint-2
	Alarm rings	USN -5	Alarm rings When the system detects drowning person	Alert the lifeguard	High	Sprint-3
Lifeguard	Saves the person	USN-6	The Lifeguard saves the swimmer who is drowning once the alarm rings	Saves the life of people	High	Sprint-3

6. PROJECT PLANNING & SCHEDULING

6.1 SPRINT PLANNING & ESTIMATION

S.NO	MILESTONE	DESCRIPTION	DURATION
1	Prerequisites	Prerequisites are all the needs at the requirement level needed for the execution of the different phases of a project.	1 WEEK
2	Create & Configure IBM cloud services	IBM Cloud provides solutions that enable higher levels of compliance, security, and management, with proven architecture patterns and methods for rapid delivery for running mission-critical workloads.	2 WEEK

3	Develop the python script	A Python script is a set of commands included in a file that is intended to be run similarly to a program. The concept is that the file will be run or performed from the command line or from within a Python interactive shell to perform a particular activity. Of course, the file includes methods and imports different modules.	1 WEEK
4	Develop web application	A web application (or web app) is application software that runs in a web browser, unlike software programs that run locally and natively on the operating system (OS) of the device.	3 WEEK
5	Ideation phase	Ideation is the process where you generate ideas and solutions through sessions such as Sketching, Prototyping, Brainstorming, Brain writing, Worst Possible Idea, and a wealth of other ideation techniques.	1 WEEK
6	Project design phases	Project design is an early phase of a project where the project's key features, structure, criteria for success, and major deliverables are planned out. The aim is to develop one or more designs that can be used to achieve the desired project goals.	2 WEEK

6.2 SPRINT DELIVERY SCHEDULE

PRODUCT BACKLOG, SPRINT SCHEDULE, AND ESTIMATION:

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can sign up for the application by providing my email address, a password, and a password confirmation.	2	High	Mohammed Shamal
		USN-2	As a user, I can also sign up for the application using Gmail, Phone Number or Facebook	2	Medium	
		USN-3	When I register for the application as a user, I will get a confirmation email	1	Low	
	Login	USN-4	I can access the application as a user by providing my sign-in address and password.	2	High	
	Dataset Collection	USN-5	The uploaded data in the prediction page will assist the user in spotting drowning movements.	2	High	
		USN-6	The dataset gathered will provide highly accurate information on the person's drowning details.	1	Low	
Sprint-2	Data Pre-Processing	USN-7	The extracted dataset is utilized to develop the model.	5	High	Barshad Basha
	Train the Model	USN-8	We can train the model then.	8	High	
		USN-9	And we can also test the model.	7	High	

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-3	Detection	USN-10	Load the tested model.	4	High	Jishin M
		USN-11	We can identify the person by collecting real-time data.	6	High	
		USN-12	The real-time data obtained will be checked with the pre-fed data.	8	High	
Sprint-4	Alert	USN-13	If any abnormal movement is detected, the sensor will ring an alarm and notify the lifeguard to save the person.	6	High	Abhinandh K
		USN-14	The drowning person is detected	4	High	
	Logout	USN-15	Now, the user can logout of the application	2	Medium	

6.3 REPORTS FROM JIRA

BOARD

The screenshot displays the Jira Board interface for the project "VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning". The board is titled "VIR board" and shows a Kanban workflow with three columns: "TO DO", "IN PROGRESS", and "DONE 15 ISSUES".

Left Sidebar:

- VirtualEye - Life Guard... Software project
- PLANNING
 - Roadmap
 - Board (selected)
- DEVELOPMENT
 - Code
- Project pages
- Add shortcut
- Project settings

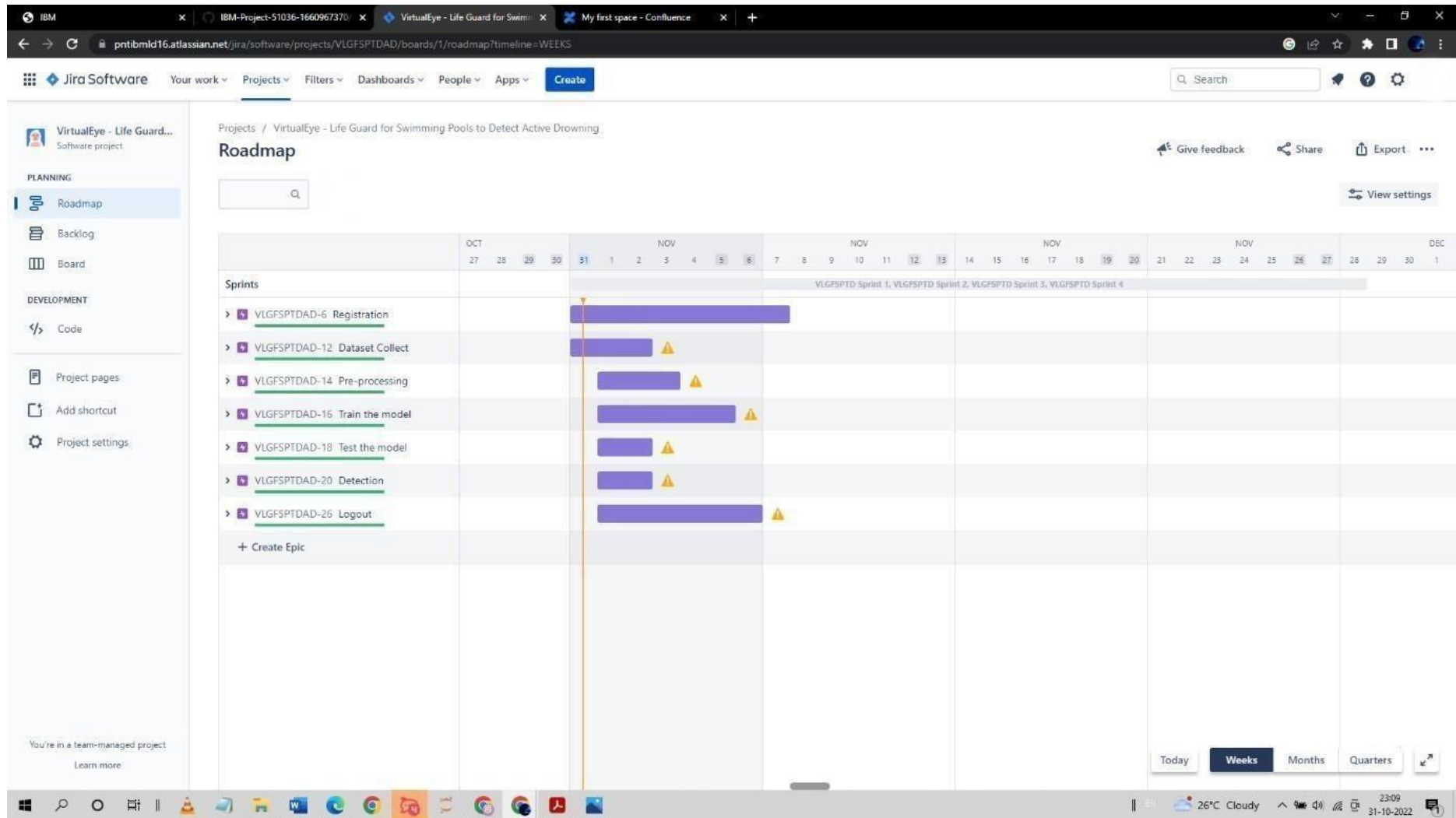
Top Navigation:

- Jira Software
- Your work
- Projects
- Filters
- Dashboards
- People
- Apps
- Create
- Search
- Help
- Settings

Board Columns:

- TO DO:** + Create issue
- IN PROGRESS:**
- DONE 15 ISSUES:**
 - As a user, I can register for the application by entering my email, password, and confirming my password...
 - VIR-1
 - Register
 - VIR-2
 - As a user, I can register for the application through Facebook
 - VIR-3
 - As a user, I can register for the application through Gmail
 - VIR-4
 - As a user, I can log into the application by entering email & password

CHART



BACKLOG

The screenshot displays the Jira Software interface for a project named 'VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning'. The left sidebar shows navigation options: 'Roadmap', 'Backlog' (selected), 'Board', 'Code', 'Project pages', 'Add shortcut', and 'Project settings'. The main area is titled 'Backlog' and shows a search bar and an 'Insights' link. 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

At the bottom, a status bar indicates the user is in a team-managed project and provides a link to 'Learn more'. The system tray at the very bottom shows the date as 31-10-2022 and the time as 22:48.

ROADMAP

The screenshot shows the Jira Roadmap interface for the project 'VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning'. The interface includes a sidebar with navigation options like 'Roadmap', 'Backlog', 'Board', 'Code', 'Project pages', 'Add shortcut', and 'Project settings'. The main area displays a timeline from September to February 2023. A search bar is located at the top left of the roadmap view. The timeline shows a sprint 'VLGFSPTDAD-6 Registration' starting in November. A vertical orange line indicates the current date, which is 31-10-2022. The bottom of the screen shows a Windows taskbar with various application icons and a system tray with weather and time information.

VirtualEye - Life Guard for Swimming Pools to Detect Active Drowning

Roadmap

Search

Give feedback Share Export View settings

Sprints

> VLGFSPTDAD-6 Registration

+ Create Epic

Today Weeks Months Quarters Quickstart

26°C Mostly cloudy 22:09 31-10-2022

7. CODING & SOLUTIONING

7.1 FEATURE 1

Login.html

```
<!DOCTYPEhtml>

<html>

<head>

<metacharset="UTF-8">

<metaname="viewport"content="width=device-width,initial-scale=1">

<title>VirtualEye</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'>

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

<!linkrel="stylesheet"href="{{url_for('static',filename='css/style.css')}}">

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

<link href='https://fonts.googleapis.com/css?family=JosefinSans'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:0px8px4pxgrey;overflow:hidden;
padding-left:20px;
font-family:'JosefinSans';
font-size:2vw;
width:100%;
height:8%;
text-align:center;
}
.topnav{overflow: hidden;
background-color:#333;}
.topnav-righta{
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-righta.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:0px0px;}

.login
{
margin-top:100px;
}

form{border:3pxsolid#f1f1f1;margin-left:400px;margin-right:400px;}

input[type=text],input[type=email],input[type=number],input[type=password]{width:100%;
```

```
padding: 12px20px;display: inline-block;margin-bottom:18px;border: 1px solid #ccc;
box-sizing:border-box;
}
button{
background-color:#28272c;color:white;
padding:14px20px;margin-bottom:8px;
border: none;cursor: pointer;width:100%;
font-weight:bold;}
button:hover{opacity:0.8;
}
.cancelbtn {width:auto;
padding: 10px 18px;background-color:#f44336;
}
.imgcontainer
{text-align:center;margin:24px012px0;}
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%">VirtualEye</div><div class="topnav-right" style="padding-top:0.5%">
<a href="{{url_for('index')}}">Home</a><a class="active" href="{{url_for('login')}}">Login</a><a href="{{url_for('register')}}">Register</a>
</div></div>

<div id="login" class="login">

<form action="{{url_for('afterlogin')}}" method="post"><div class="imgcontainer">
```

```
</div>
<divclass="container">
<inputtype="email"placeholder="Enter  registeredemailID"name="_id"required><br>

<inputtype="password"placeholder="Enter  Password"name="psw"required>
<buttontype="submit">Login</button><br>{{pred}}
</div></form>
</div>
</body>
</html>
```

Register.html

```
<!DOCTYPEhtml>

<html>

<head>

<metacharset="UTF-8">

<metaname="viewport"content="width=device-width,initial-scale=1">

<title>VirtualEye</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'>

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

<linkrel="stylesheet"href="{{url_for('static',filename='css/style.css')}}">

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

<link href='https://fonts.googleapis.com/css?family=JosefinSans'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:0px8px4pxgrey;overflow:hidden;

padding-left:20px;

font-family:'JosefinSans';

font-size:2vw;width:100%;

height:8%;

text-align:center;

}.topnav{

overflow: hidden;background-color:#333;

}

.topnav-righta{

float: left;

color:#f2f2f2;text-align: center;

padding: 14px 16px;text-decoration:none;font-size:18px;}

.topnav-righta: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:0px0px;
}.login{
margin-top:100px;}

form{border:3pxsolid#f1f1f1;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:14px20px;margin-bottom:8px;border: none;cursor: pointer;width:100%;
}

button:hover{opacity:0.8;
```

```
}

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

.imgcontainer {text-align:center;
margin:24px012px0;}

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>  
  
<bodystyle="font-family:Montserrat;">  
  
<divclass="header"><div style="width:50%;float:left;font-size:2vw;text-align:left;color:white;padding-top:1%">VirtualEye</div><divclass="topnav-right">  
  
<a href="{{url_for('home')}}">Home</a><a href="{{url_for('login')}}">Login</a>  
  
<a class="active" href="  
  
{  
  
{  
  
url_for('register')}}">Register</a>  
  
</div></div>  
  
<div id="login" class="login">  
  
<form action="{{url_for('afterreg')}}" method="post"><div class="imgcontainer">  
  
</div>  
  
<div class="container">
```

```

<inputtype="text"placeholder="EnterName"name="name" required><br>
<inputtype="email"placeholder="EnterEmail ID"name="_id"required><br>
<inputtype="password"placeholder="Enter Password"name="psw"required>
<buttontype="submit">Register</button><br>{{pred}}
</div>
<divclass="container"style="background-color:#f1f1f1">
<divclass="psw">Alreadyhaveanaccount?&nbsp;&nbsp; <a href="{{url_for('login')}}">Login</a></div>
</div>
</form>
</div>
</body>
</html>

```

Base.html

```

<html>
<htmllang="en">
<head>
<metacharset="UTF-8">
<meta name="viewport" content="width=device-width,initial-scale=1.0">
<metahttp-equiv="X-UA-Compatible"content="ie=edge">

```



```
<title>HighQualityFacialRecognition</title>

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

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

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

>

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

<linkhref="{{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";><headerid="head"class="header"><sectionid="navbar">

<h1class="nav-heading"></i>VirtualEye</h1><divclass="nav--items">

<ul>

<li><a href="{{url_for('index')}}">Home</a></li><li><a
```

```
href="{{url_for('logout')}}">Logout</a></li>

<!--<li><ahref="#about">About</a></li><li><ahref="#services">Services</a></li>-->

</ul></div>

</section></header>

<divclass="container">

<divid="content"style="margin-top:2em"><divclass="container">

<divclass="row"><divclass="col-sm-6bd" >

<h2><em style="color:white;">HighQuality FacialRecognition</em></h2>

<br>

<p><h5><i style="color:white;">Emotion DetectionThroughFacialFeatureRecognition</i></h5></p>

</div>

<divclass="col-sm-6"><div>

<h4style="color:white;">Upload ImageHere</h4>

<form action = "http://localhost:5000/"id="upload-file"method="post" enctype="multipart/form-data">

<labelfor="imageUpload"class="upload-label">

ChooseImage</label>

<inputtype="file"name="image"id="imageUpload"accept=".png,.jpg,.jpeg,.pdf">
```

```
</form>

<divclass="image-section"style="display:none;">
<divclass="img-preview"><divid="imagePreview">
</div>
</div><div>
<buttontype="button"class="btnbtn-info btn-lg" id="btn-predict">Analyse</button>
</div></div>

<divclass="loader"style="display:none;"></div>

<h3>
<spanid="result"></span></h3>
</div></div>
</div></div>
</div>
</div></body>

<footer>

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

</html>
```

Index.html

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

<head>

<metacharset="UTF-8">

<metahttp-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>

<scriptsrc="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPsk vXusvfa0b4Q" crossorigin="anonymous"></script>

<scriptsrc="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootst rap.min.js" integrity="sha384-
JZR6Spejh4U02d8j0t6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5
+76PVCmYl"crossorigin="anonymous"></script>

<scriptsrc="https://kit.fontawesome.com/8b9cdc2059.js"crossorigin="anonymous"></script>

<linkhref="https://fonts.googleapis.com/css2?family=Akronim&family=Roboto&display=swap"rel="stylesheet">

<linkrel="stylesheet"href=" ../static/style.css">

<!--<scriptdefersrc=" ../static/js/main.js"></script>-->
```

```
<title>VirtualEye</title>

</head><body>

<headerid="head"class="header"><sectionid="navbar">

<h1class="nav-heading"></i>VirtualEye</h1><divclass="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><ahref="{{url_for('login')}}">Demo</a></li></ul>

</div></section>

<sectionid="slider">

<divid="carouselExampleIndicators"class="carousel"data-ride="carousel">

<olclass="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>

<divclass="carousel-inner">
```

```
<divclass="carousel-itemactive">
<imgclass="d-blockw-100"src="../../static/img/1.png"alt="Firstslide">
</div><divclass="carousel-item">

</div><divclass="carousel-item">
<imgclass="d-blockw-100"src="../../static/img/third.jpg"alt="Thirdslide">
</div></div>
<a class="carousel-control-prev"href="#carouselExampleIndicators"role="button" data-slide="prev">
<span class="carousel-control-prev-icon"aria-hidden="true"></span>
<spanclass="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>
<spanclass="sr-only">Next</span>
</a></div>
</section></header>
<sectionid="about"><divclass="top">
<h3class="titletext-muted">ABOUTPROJECT
</h3>
<divclass="line"></div></div>
```

```
<divclass="body">
<divclass="left"><h2>Problem:</h2>
<p>
</p></div>
<divclass="left"><h2>Solution:</h2>
<p>
</p></div></div>
<divclass="bottom"><p><b>
</b></p>
</div></section>
<sectionid="footer">
<p>CopyrightÂ©2022.AllRightsReserved</p><divclass="social">
<a href="#" target="_blank"><i class="fabfa-2xfa-twitter-square"></i></a>
<a href="#" target="_blank"><i class="fabfa-2xfa-linkedin"></i></a>
<a href="#">
<i class="#"></i></a>
</div></section></body>
</html>
```

Logout.html

```
<!DOCTYPEhtml><html>

<head>

<metacharset="UTF-8">

<meta name="viewport" content="width=device-width,initial-scale=1">

<title>VirtualEye</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=JosefinSans' 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>
</form>
</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/bootstrap.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-
JZR6Spejh4U02d8j0t6vLEHfe/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 for Swimming 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>
                </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="Click Me! For a Demo">

```

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

    </br></br>

    <section id="footer">
        <p>Copyright Â© 2021. All Rights Reserved</p>
    </section>
</body>

</html>
```


7.2 FEATURE 2

App.py

```
import time
import cv2
import numpy as np
from cloudant.client import Cloudant
from flask import Flask, request, render_template, redirect, url_for from playsound import playsound

import cvlib as cv
from cvlib.object_detection import draw_bbox # Loading the model
# Authenticate using an IAM API key
client = Cloudant.iam('5a1ffd26-d995-410e-af77-546fb6498fd8- bluemix','5rUgrIfVeYtIyTqJ0hAvTOvIYvJDNiKlr-
sDUHQRClnN', 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.route('/')
def index():
    return render_template('index.html')
```

```

@app.route('/index.html') def home():
    return render_template("index.html")
#registration page
@app.route('/register') def register():
    return render_template('register.html')
@app.route('/afterreg', methods=['POST']) def afterreg():
    x = [x for x in request.form.values()] print(x)
    data = {
        '_id': x[1], # Setting _id is optional 'name': x[0],
        'psw':x[2]
    }
    print(data)
    query = {'_id': {'$eq': data['_id']}}
    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 afterlogin():

```

```
user = request.form['_id'] passw = request.form['psw'] print(user,passw)
query = {'_id': {'$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]['_id'] and passw==docs[0][0]['psw'])): return redirect(url_for('prediction'))
else:
print('Invalid User')

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

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

@app.route('/result',methods=["GET","POST"]) def res():
webcam = cv2.VideoCapture('drowning.mp4')

if not webcam.isOpened(): print("Could not open webcam") exit()
```

```

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 webcam.isOpened():
# 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)
#print('Seconds since last epoch: ', time.time()-t0)

# display output
cv2.imshow("Real-time object detection", out) if(isDrowning == True):
    playsound('alarm.mp3') webcam.release() cv2.destroyAllWindows()
    return render_template('prediction.html',prediction="Emergency
    !!! The Person is drowining")
#return render_template('base.html')

# press "Q" to stop
if cv2.waitKey(1) & 0xFF == ord('q'): break

# release resources webcam.release() cv2.destroyAllWindows()
#return render_template('prediction.html',)

""" Running our application """ if __name__ == "__main__":
    app.run(debug=True)
```

Init.py

```
from .object_detection import detect_common_objects
```

Object_detect.py

```
import necessary packages import 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 script
```

```
class_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 blue if 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

```

Utils.py

```

import requests
import progressbar as pb import os
def download_file(url, file_name, dest_dir): if not os.path.exists(dest_dir):
os.makedirs(dest_dir)
full_path_to_file = dest_dir + os.path.sep + file_name if os.path.exists(dest_dir + os.path.sep + file_name):
return full_path_to_file
print("Downloading " + file_name + " from " + url) try:
r = requests.get(url, allow_redirects=True, stream=True)
except:
print("Could not establish connection. Download failed") return None

file_size = int(r.headers['Content-Length']) chunk_size = 1024
numBars = round(file_size / chunk_size)
bar = pb.ProgressBar(maxval=numBars).start() if r.status_code != requests.codes.ok:
print("Error occurred while downloading file") return None
count = 0

```

```
with open(full_path_to_file, 'wb') as file:
for chunk in r.iter_content(chunk_size=chunk_size): file.write(chunk)
bar.update(count) count +=1
return full_path_to_file
```

Detect.py

```
import cvlib as cv
from cvlib.object_detection import draw_bbox import cv2
import time
import numpy as np
from playsound import playsound #for PiCamera
#from picamera Import PiCamera #camera = PiCamera #camera.start_preview()
# open webcam
webcam = cv2.VideoCapture(0)

if not webcam.isOpened(): print("Could not open webcam") exit()

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 webcam.isOpened():
```

```
# 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)
#print('Seconds since last epoch: ', time.time()-t0)

```

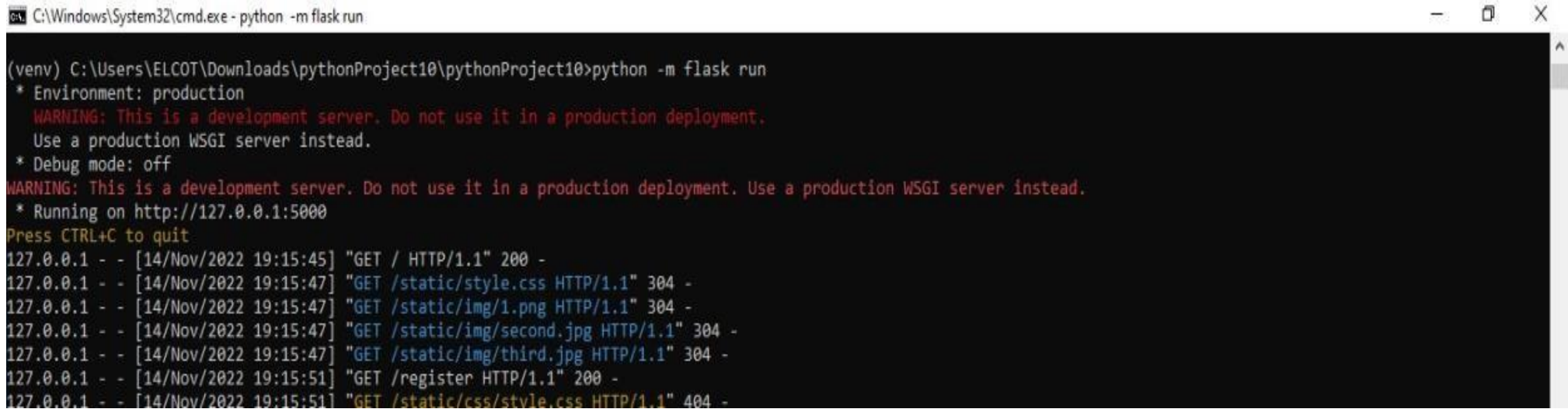
```
# display output
cv2.imshow("Real-time object detection", out) if(isDrowning == True):
    playsound('alarm.mp3')
# press "Q" to stop
if cv2.waitKey(1) & 0xFF == ord('q'): break
# release resources
webcam.release()
cv2.destroyAllWindows()
```

8. TESTING

8.1 TEST CASES

1: Run the application

In the command prompt, navigate to the folder in which the flask app is present. When the python file is executed the localhost is activated on 5000 port and can be accessed through it.



```
C:\Windows\System32\cmd.exe - python -m flask run

(venv) C:\Users\ELCOT\Downloads\pythonProject10\pythonProject10>python -m flask run
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [14/Nov/2022 19:15:45] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 19:15:47] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:15:47] "GET /static/img/1.png HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:15:47] "GET /static/img/second.jpg HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:15:47] "GET /static/img/third.jpg HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:15:51] "GET /register HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 19:15:51] "GET /static/css/style.css HTTP/1.1" 404 -
```

2: Open the browser and navigate to <http://127.0.0.1:5000> to check your application
The home page looks like this. You can click on login or register

Virtual Eye

[Home](#) [Login](#) [Register](#) [Demo](#)

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.

python-3.10.0a1-a...exe ^

python-3.10.0-emb...zip ^

[Show all](#) X

Type here to search



29°C Cloudy



ENG



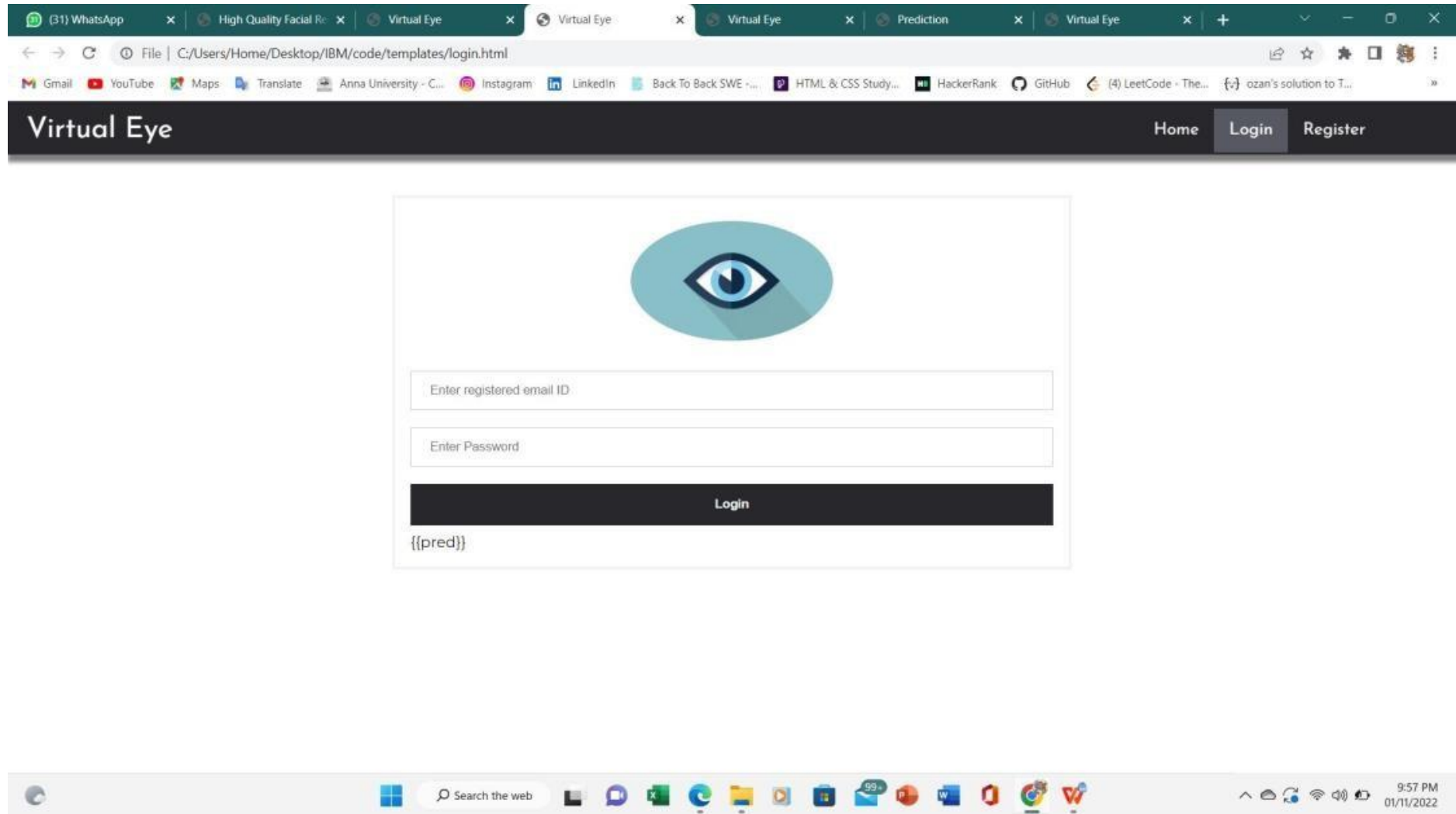
14:37

13-11-2022



While logging in you need to provide your registered credentials

After successfully login you will redirect to the prediction page where we have to click on the demo button to launch the open cv window for video analysis.



The screenshot shows a web browser window with multiple tabs open, including 'Virtual Eye' and 'Prediction'. The address bar shows the file path 'C:/Users/Home/Desktop/IBM/code/templates/login.html'. The browser's toolbar includes links to Gmail, YouTube, Maps, Translate, and various social media and development sites like Anna University, Instagram, LinkedIn, Back To Back SWE, HTML & CSS Study, HackerRank, GitHub, LeetCode, and ozan's solution to T... The page header features the 'Virtual Eye' logo and navigation links for 'Home', 'Login', and 'Register'. The main content area displays a large blue eye icon, followed by two input fields labeled 'Enter registered email ID' and 'Enter Password'. Below these fields is a dark blue 'Login' button. At the bottom of the form, there is a placeholder text '{{pred}}'. The Windows taskbar at the bottom shows the Start button, a search bar, and several application icons, including the Windows Store, Edge, File Explorer, and various productivity tools. The system clock in the bottom right corner indicates the time is 9:57 PM on 01/11/2022.

After successfully login you will redirect to the prediction page where we have to click on the demo button to launch the open cv window for video analysis.

← → ↻ ⓘ 127.0.0.1:5000/prediction 🔍 📁 ☆ ⚙️ 🗖️ 🟡 ⋮


Gmail YouTube Maps paid for article Introduction to Hac... Directory: Report V... My Camu WhatsApp Web IBM IBM Cloud

Virtual Eye

[Home](#) [Logout](#)

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

Output :-

← → ↻ ⓘ 127.0.0.1:5000/result 🔍 📄 ☆ ⚙️ 🗖️ V ⋮


 Gmail  YouTube  Maps paid for article  Introduction to Hac...  Directory: Report V...  My Camu  WhatsApp Web  IBM  IBM Cloud

Virtual Eye

Home Logout

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.

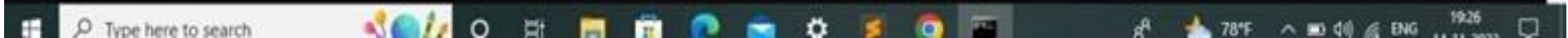


Click Me! For a Demo

Emergency !!! The Person is drowning

Copyright © 2021. All Rights Reserved

```
1
127.0.0.1 - - [14/Nov/2022 19:16:41] "POST /afterlogin HTTP/1.1" 302 -
127.0.0.1 - - [14/Nov/2022 19:16:41] "GET /prediction HTTP/1.1" 200 -
127.0.0.1 - - [14/Nov/2022 19:16:42] "GET /static/style.css HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:16:42] "GET /static/js/JScript.js HTTP/1.1" 304 -
127.0.0.1 - - [14/Nov/2022 19:16:42] "GET /static/img/second.jpg HTTP/1.1" 304 -
5.816675424575806 s
bbox: [[114, 112, 804, 372]] centre: [459.0, 242.0] centre0: [0. 0.]
Is he drowning: False
4.5444793701171875 s
bbox: [[114, 112, 804, 372]] centre: [459.0, 242.0] centre0: [459.0, 242.0]
Is he drowning: False
8.752950429916382 s
bbox: [[114, 112, 804, 372]] centre: [459.0, 242.0] centre0: [459.0, 242.0]
Is he drowning: False
12.785400867462158 s
bbox: [[120, 112, 800, 372]] centre: [460.0, 242.0] centre0: [459.0, 242.0]
Is he drowning: True
```



Successfully Logged Out!

Login for more information

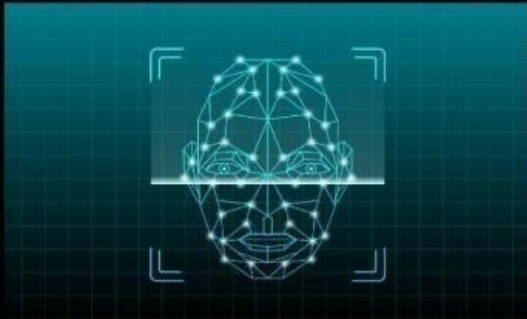
Login

High Quality Facial Recognition

Upload Image Here

Choose Image Choose File No file chosen

Emotion Detection Through Facial Feature Recognition

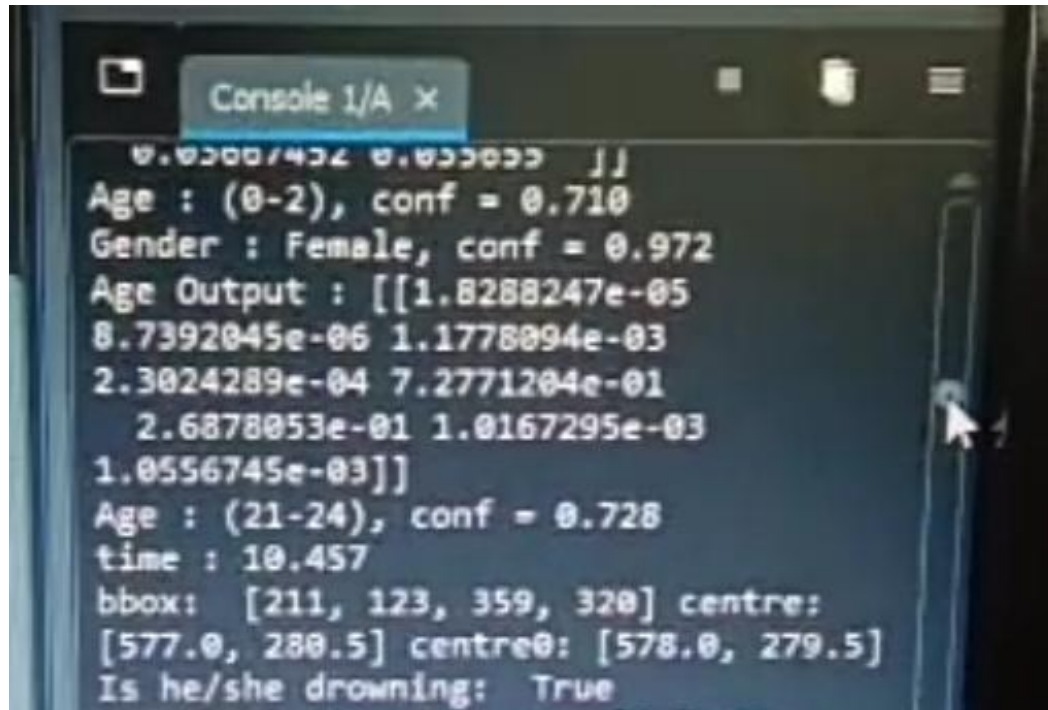


BEFORE DROWNING

```
Is he/she drowning: False  
[[525, 158, 632, 399]] ['person']  
[0.6627918481826782]  
9.57535433769226 selse  
Is he/she drowning: False  
[[526, 169, 630, 399]] ['person']  
[0.5941009521484375]  
9.769945621490479 selse  
Is he/she drowning: False  
[[522, 158, 634, 401]] ['person']  
[0.3514072000980377]  
9.977507829666138 selse  
Is he/she drowning: False  
[[520, 164, 634, 397]] ['person']  
[0.5000854134559631]  
10.17202639579773 selse
```

AFTER DROWNING

With age, gender and drowning state



```
0.03007432 0.033033 JJ
Age : (0-2), conf = 0.710
Gender : Female, conf = 0.972
Age Output : [[1.8288247e-05
8.7392045e-06 1.1778094e-03
2.3024289e-04 7.2771204e-01
2.6878053e-01 1.0167295e-03
1.0556745e-03]]
Age : (21-24), conf = 0.728
time : 10.457
bbox: [211, 123, 359, 320] centre:
[577.0, 280.5] centre0: [578.0, 279.5]
Is he/she drowning: True
```

8.2 USER ACCEPTANCE TESTING

1. 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).

2. Defect Analysis

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

3.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. RESULTS

9.1 PERFORMANCE METRICES

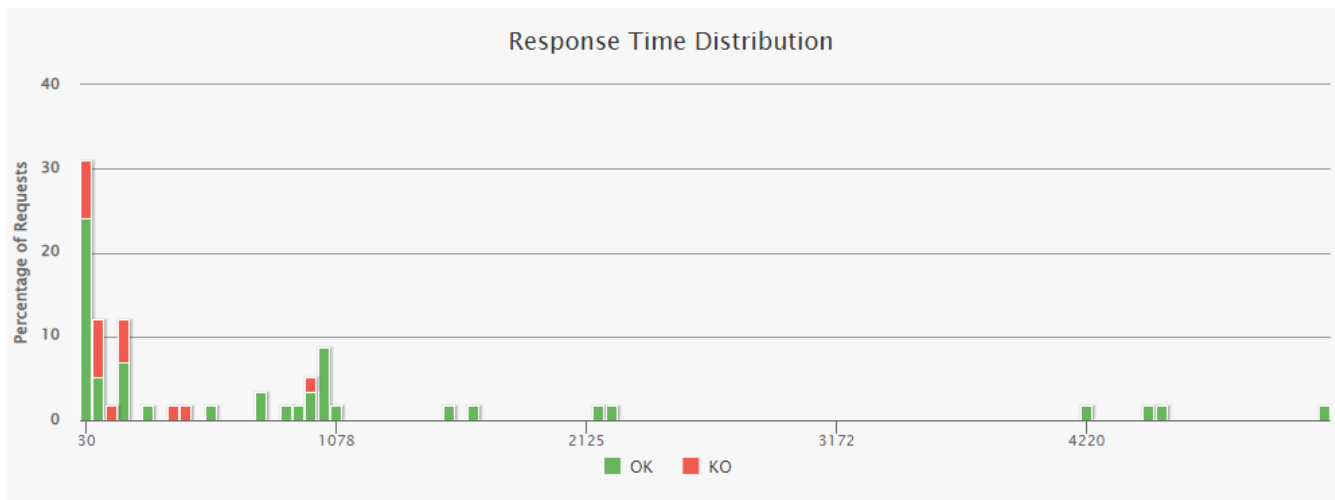
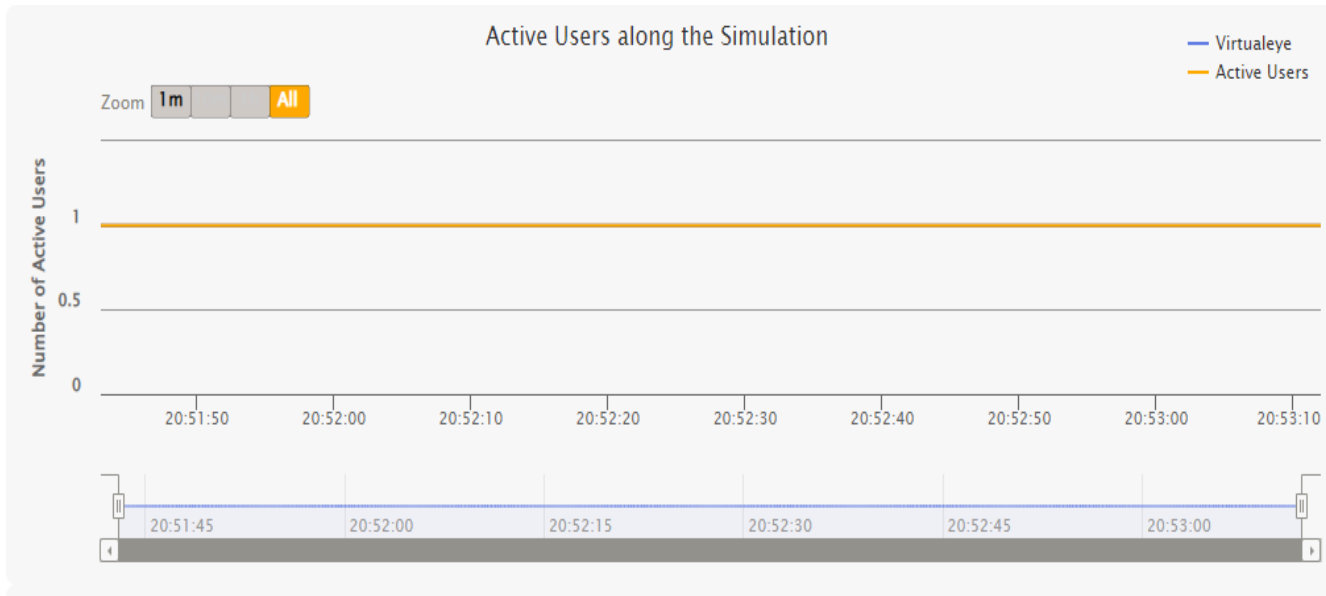


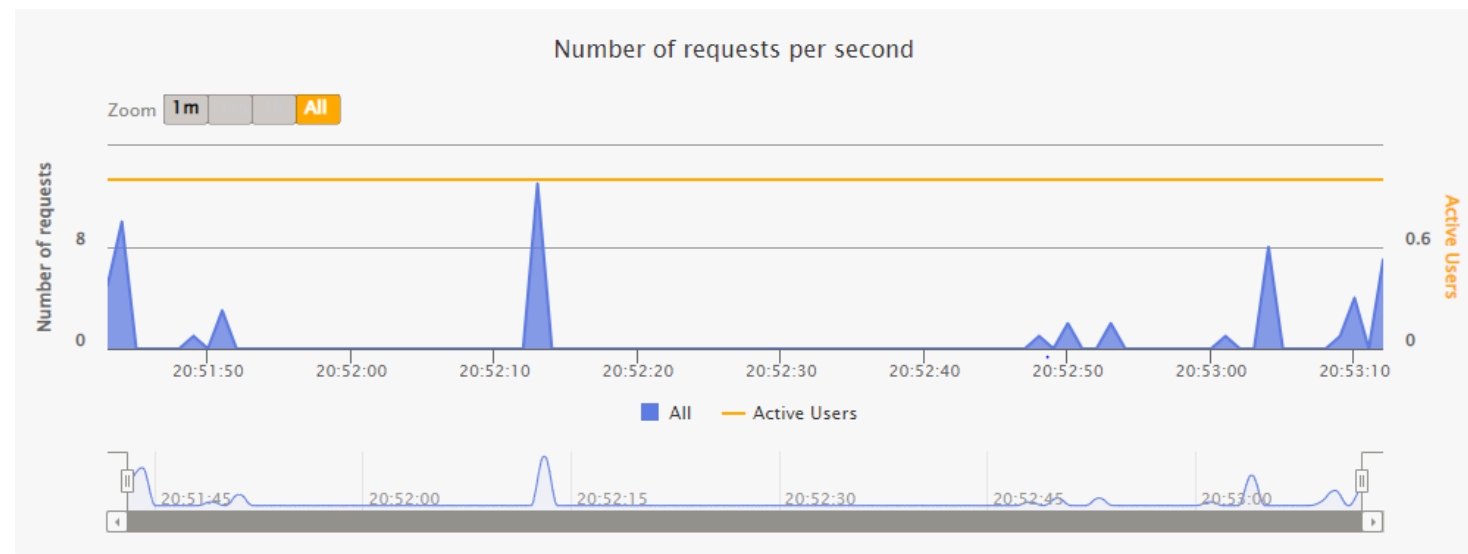
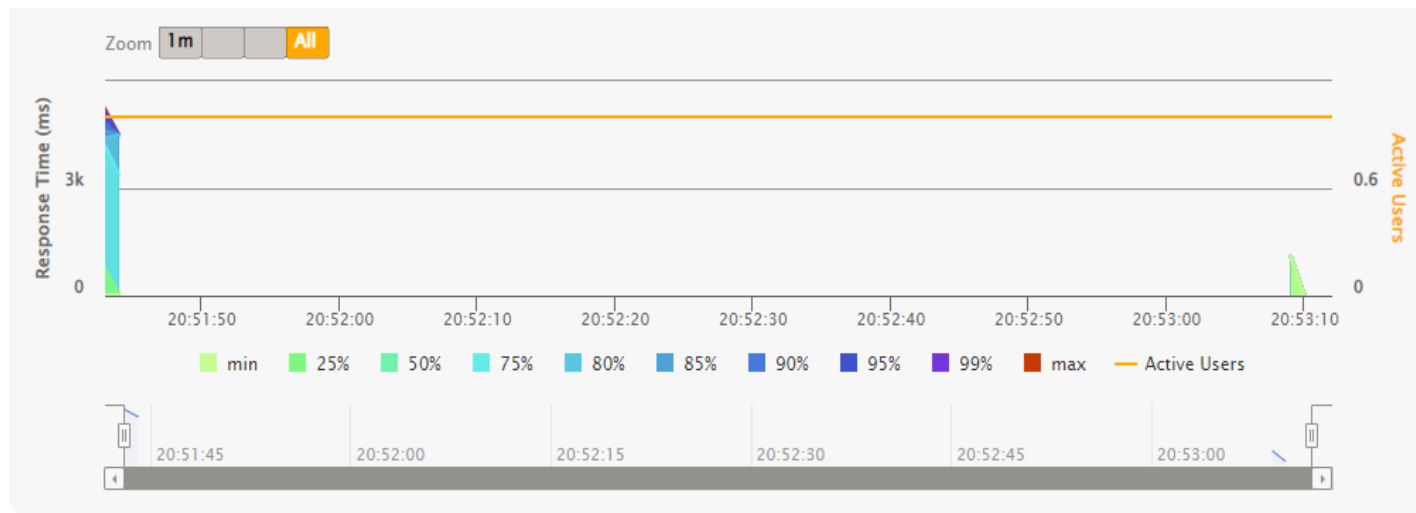
[illegible]

request_12	1 1 0 0%	0.011	997	997	997	997	997	997	997	0
request_13	1 1 0 0%	0.011	863	863	863	863	863	863	863	0
eye_image_generic_555.jpg	1 1 0 0%	0.011	934	934	934	934	934	934	934	0
css?family=Montserrat	1 1 0 0%	0.011	100	100	100	100	100	100	100	0
request_11	1 1 0 0%	0.011	203	203	203	203	203	203	203	0
request_15	1 1 0 0%	0.011	2162	2162	2162	2162	2162	2162	2162	0
request_19	1 1 0 0%	0.011	6	6	6	6	6	6	6	0
request_23	1 1 0 0%	0.011	2231	2231	2231	2231	2231	2231	2231	0
request_23 Redirect 1	1 1 0 0%	0.011	7	7	7	7	7	7	7	0
JScript.js	2 0 2 100%	0.022	14	107	153	190	197	199	107	93
request_24	1 1 0 0%	0.011	202	202	202	202	202	202	202	0
request_26	1 1 0 0%	0.011	199	199	199	199	199	199	199	0
request_25	1 1 0 0%	0.011	200	200	200	200	200	200	200	0
request_31	1 1 0 0%	0.011	1085	1085	1085	1085	1085	1085	1085	0
request_32	1 1 0 0%	0.011	10	10	10	10	10	10	10	0
request_34	1 1 0 0%	0.011	11	11	11	11	11	11	11	0
request_33	1 1 0 0%	0.011	12	12	12	12	12	12	12	0
request_35	1 1 0 0%	0.011	13	13	13	13	13	13	13	0
request_36	1 1 0 0%	0.011	6	6	6	6	6	6	6	0
request_37	1 1 0 0%	0.011	26	26	26	26	26	26	26	0
request_38	1 1 0 0%	0.011	23	23	23	23	23	23	23	0
request_39	1 1 0 0%	0.011	26	26	26	26	26	26	26	0

Errors

Error	Count	Percentage
status.find.in([200, 209], 304), found 404	12	80 %
status.find.in([200, 209], 304), found 400	3	20 %





10.ADVANTAGES & DISADVANTAGES

ADVANTAGES

- AI-powered cameras can track and monitor all swimmers, no matter where they are in the pool, and regardless of ripples, glare, and splashes in the water.
- Timely alerts empower lifeguards to intervene with a swimmer in distress, before a drowning occurs.
- System setup is short and simple, and does not require closing the facility.
- Seamless for swimmers and staff.
- User-friendly for lifeguards.
- AI-powered cameras also provide pool managers with water surface analytics data to help them improve the pool experience for everyone, for instance, by alleviating crowding during popular hours, and ensuring efficient staffing and maintenance activities.

DISADVANTAGES

- Many factors have a negative impact on successful target detection in basic studies of visual search, including crowding, target-distractor similarity and attentional set. These factors are also likely to play a negative role in the visual search of lifeguards.
- For instance, crowding is typically defined as an effect that limits perception of objects, features when surrounded by neighbouring distractors. The ability to recognize and respond to crowded targets is dramatically reduced during visual search
- The negative impacts of crowding overlap considerably with the related concept of visual clutter. As the number of items in a search area increases, the space between items becomes smaller and this limits the searcher's attention to smaller areas
- This phenomenon of crowding has obvious relevance to lifeguarding, for example, with increased numbers of swimmers, physical space within the zone of supervision will become visually cluttered, causing delayed reaction times in visual searches.

11. CONCLUSIONS

Consistently numerous people, including kids, are suffocated or near suffocating in the deeps of the swimming pools, and the lifeguards are not prepared all around to deal with these issues. In this manner raises the necessities for having a framework that will thus recognize the suffocating people and alert the lifeguards at such hazard. It can be installed in International standardized schools where classes are held for training kids.

12. FUTURE SCOPE

- We aim at designing and bringing in further renditions that would result in developing a more precise, accurate and more reliable in totality.
- Our AI design improvements would include a wider set of features including more advanced pattern analysis, quicker response times, faster processing and more inventive protocols
- We would also be incorporating further hardware advancements including water quality monitoring, foreign solvent detection to make sure that none is effected by any allergens within the pool that might trigger adverse health effects in the swimmers that even possibly result in drowning

13. APPENDIX

SOURCE CODE LINK: [Code Link](#)

Project Demo Link: [Video Link 1](#), [Video Link 2](#)

GitHub Link: [github link](#)