

# **VIRTUAL EYE - LIFE GUARD FOR SWIMMING POOLS TO DETECT ACTIVE DROWNING**

## **A PROJECT REPORT**

**Submitted by**

<b>JEFRI JEBASON J</b>	<b>19CECS012</b>
<b>NITHESH PAUL AN</b>	<b>19CECS023</b>
<b>PRAKASH G</b>	<b>19CECS027</b>
<b>PRASANNA P</b>	<b>19CECS029</b>

**In partial fulfillment for the award of the degree**

**BACHELOR OF ENGINEERING**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**SNS COLLEGE OF ENGINEERING,**

**COIMBATORE-641035**



<b>CHAPTER NO</b>	<b>TITLE</b>	<b>PAGENO</b>
<b>1</b>	<b>INTRODUCTION</b> a. Project Overview b. Purpose	4
<b>2</b>	<b>LITERATURE SURVEY</b> a. Existing System b. References c. Problem Statement Definition	4
<b>3</b>	<b>IDEATION &amp; PROPOSED SOLUTION</b> a. Empathy Map Canvas b. Ideation & Brainstorming c. Proposed Solution d. Problem Solution Fit	6
<b>4</b>	<b>REQUIREMENT ANALYSIS</b> a. Functional Requirement b. Non-Functional Requirement	9
<b>5</b>	<b>PROJECT DESIGN</b> a. Data-Flow Diagrams b. Solution & Technical Architecture c. User Stories	9
<b>6</b>	<b>PROJECT PLANNING &amp; SCHEDULING</b> a. Sprint Planning ,Schedule &Estimation b. Sprint Delivery Schedule c. Reports From JIRA	12

<b>7</b>	<b>CODING &amp; SOLUTIONING</b> a. Feature 1 b. Feature 2	17
<b>8</b>	<b>TESTING</b> a. Test cases b. User Acceptance Testing	25

<b>9</b>	<b>RESULTS</b> 9.1PerformanceMetrics	27
<b>10</b>	<b>ADVANTAGES &amp; DISADVANTAGES</b> a. Advantages b. Disadvantages	43
<b>11</b>	<b>CONCLUSION</b>	44
<b>12</b>	<b>FUTURE SCOPE</b>	44
<b>13</b>	<b>APPENDIX</b>	44

# **1.INTRODUCTION**

## **1.1 PROJECT 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.

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 then an alert will be generated to attract lifeguards' attention.

## **1.2 PURPOSE**

Designed for whom has to guarantee every day the safety in public and intensive-use pools, AngelEye LifeGuard detects potential drownings and promptly notifies you. It features the latest artificial intelligence technology and adapts to the needs of the user. It's the ultimate drowning detection system for those who demand the ultimate in safety.

## **2. LITERATURE SURVEY**

### **2.1 EXISTING PROBLEM**

[1] One important environment that the need for monitoring systems is crucially sensed is the swimming pool. Each year many people including children are drowned or very close to drowning in the deeps of the swimming pools, and the lifeguards are not trained well enough to handle these problems. This raises the need for having a system that will automatically detect the drowning person and alarm the lifeguards of such danger. Real-time detection of a drowning person in swimming pools is a challenging task that requires an accurate system. The challenge is due to the presence of water ripples, shadows and splashes and therefore detection needs to have high accuracy. SOFTWARE USED : OPEN CV

[2] Lei Fei, Wang Xueli, Chen Dongsheng, proposed a background subtraction method for drowning detection and swimmer identification using visual surveillance in their research paper. This method fails to reflect real background accurately thus restricting model accurate shape detection of moving objects. It also fails to reflect sudden background changes. Ajil Roy, Dr. K. Srinivasan, proposed drowning detection using RFID-based swimming goggles, however, this model also fails to overcome the limitation of accuracy since the water sensor is not placed very close to the mouth and nose. But this model successfully overcomes limitations of video surveillance-based drowning detection systems like the need for high power computing devices. Thus, we suggest two approaches for implementing the drowning detection1. Using ConvLSTM2D layers in the model 2. Using LRCN approach

### **2.2 REFERENCES**

Reference Link :-

[https://www.researchgate.net/publication/357549495\\_Drowning\\_behavior\\_detection\\_in\\_swimming\\_pool\\_based\\_on\\_deep\\_learning](https://www.researchgate.net/publication/357549495_Drowning_behavior_detection_in_swimming_pool_based_on_deep_learning)

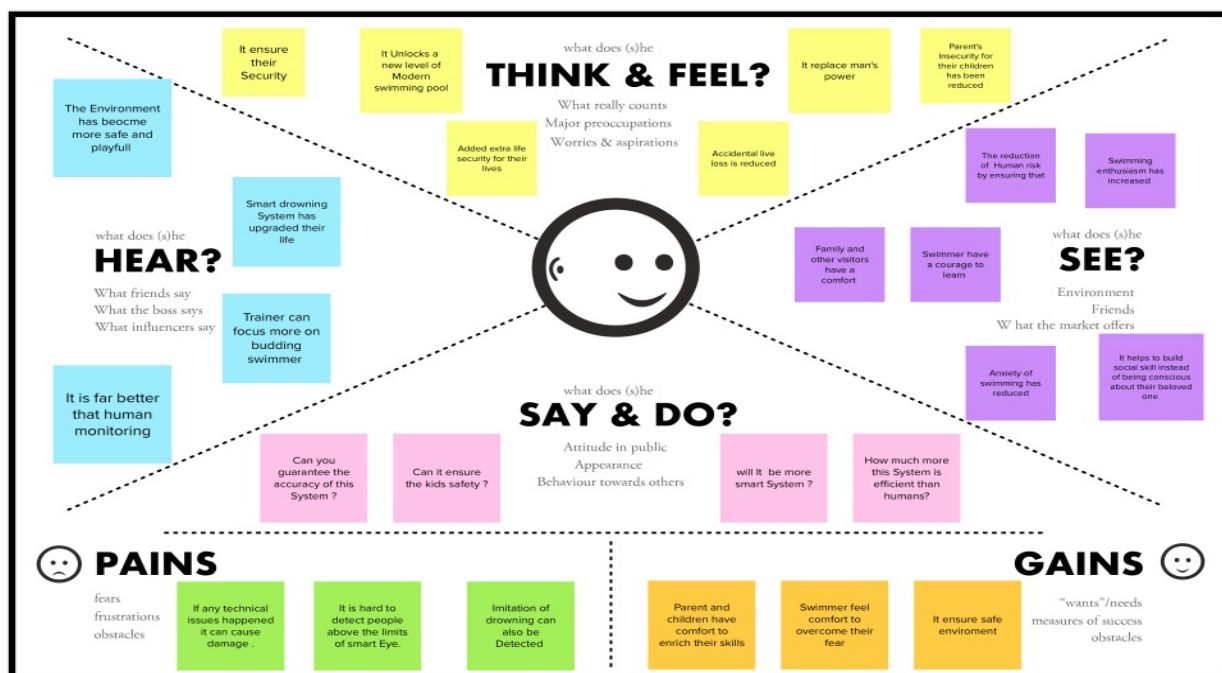
<https://onlinelibrary.wiley.com/doi/full/10.1002/acp.3756>

## 2.3 PROJECT STATEMENT DEFINITION

We provided a method to robust human tracking and semantic event detection within the context of video surveillance system capable of automatically detecting drowning incidents in a swimming pool. In the current work, an effective background detection that incorporates prior knowledge using HSV color space and contour detection enables swimmers to be reliably detected and tracked despite the significant presence of water ripples. The system has been tested on several instances of simulated water conditions such as water reflection, lightening condition and false alarms. Our algorithm was able to detect all the drowning conditions along with the exact position of the drowning person in the swimming pool and had an average detection delay of 1.53 seconds, which is relatively low compared to the needed rescue time for a lifeguard operation. Our results show that the proposed method can be used as a reliable multimedia video-based surveillance system.

## 3.IDEATHON & PROPOSED SOLUTION

### 3.1 EMPATHY MAP CANVAS



## 3.2 IDEATHON AND BRAINSTROMING

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 fit the paper (pinch to stretch) over to start drawing!

**NITHESH PAUL AN**

Smart Alerting System.

Effective Supervision System.

Number of active swimmers.

Detect Budding swimmers

**PRAKASH G**

Technical Issue detection

Pool water Quality Detector

Enhance Pool surveillance System

Detect the child who are below restricted age

**JEFRI JEBASON J**

Insecurity of kids

Chance of (Hypoxic brain injury)

Quick Emergency alert

Proper swim suit detection

**PRASANNA P**

System to monitor kids and old people

Water Impurity Detection

Send Quick alert to near by Trainers.

Playful drowning can't be detected

3

**Group ideas**  
Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, 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

Detect Budding swimmers

Pool water Quality Detector

Smart Alerting System.

Playful drowning can't be detected

Chance of (Hypoxic brain injury)

Proper swim suit detection

Insecurity of kids

Send Quick alert to near by Trainers.

Detect the child who are below restricted age

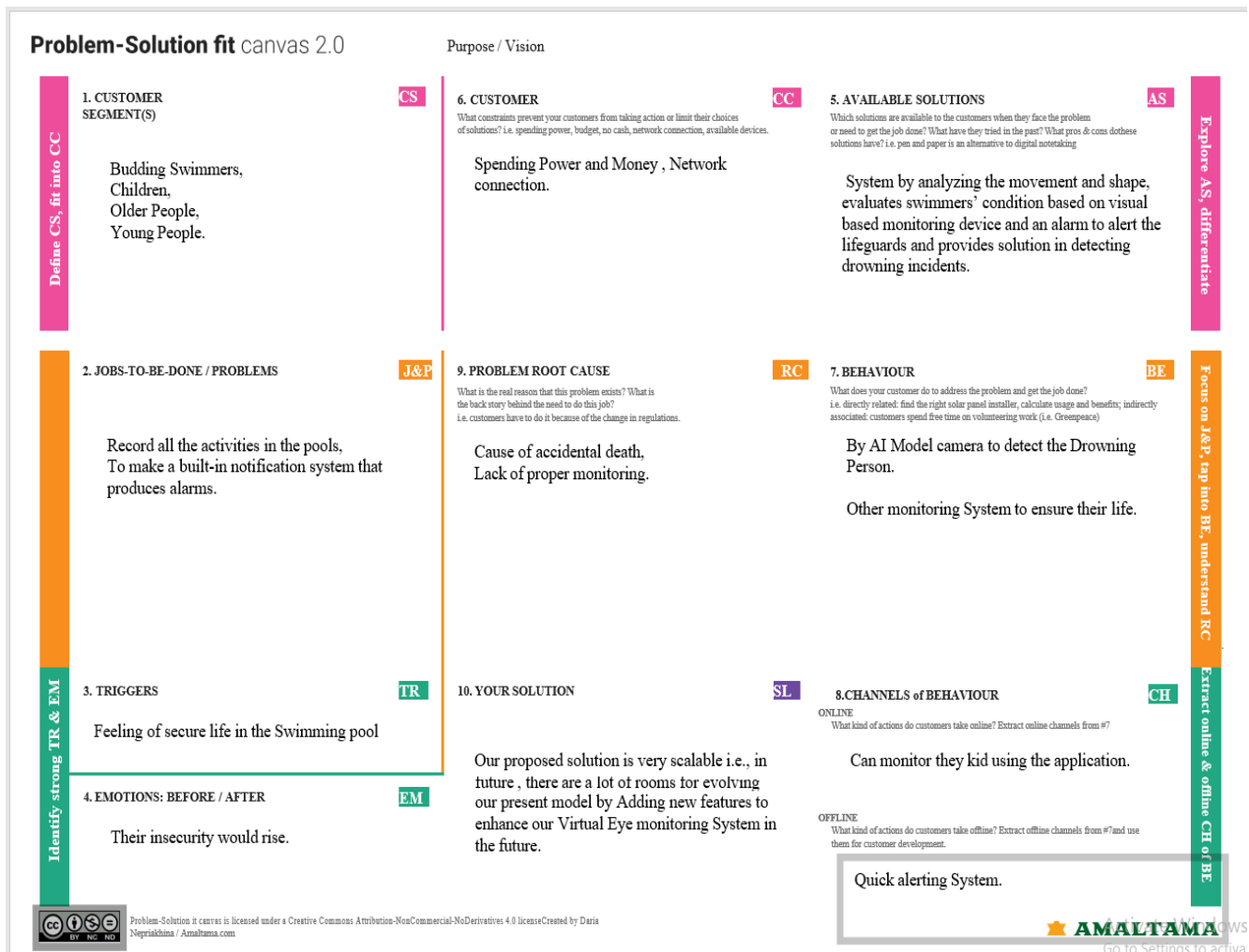
Page No : 7

### 3.3 PROPOSED SOLUTION

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	This system can also be able to record all the activities in the pools and to classify critical situations from normal ones in order to keep track of what happened. The built-in notification system produces alarms within 10 seconds on smartwatches, phones, flashing lights and other configurable devices.
2.	Idea / Solution description	This system by analyzing the movement and shape, evaluates swimmers' condition based on visual based monitoring device and an alarm to alert the lifeguards and provides solution in detecting drowning incidents.
3.	Novelty / Uniqueness	Virtual eye has developed a novel idea of alerting the ambulance and another life guard if there is any delay in saving the person to death.
4.	Social Impact / Customer Satisfaction	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 pool to save human life.
5.	Business Model (Revenue Model)	Our solution, once developed well, has enough possibility to become a good product to
6.	Scalability of the Solution	Our proposed solution is very scalable i.e., in future, there are a lot of rooms for evolving our present model by adding new features to enhance our system in the future.



## 3.4 PROBLEM FIT SOLUTION



## 4. REQUIREMENT ANALYSIS

### 4.1 FUNCTIONAL REQUIREMENTS

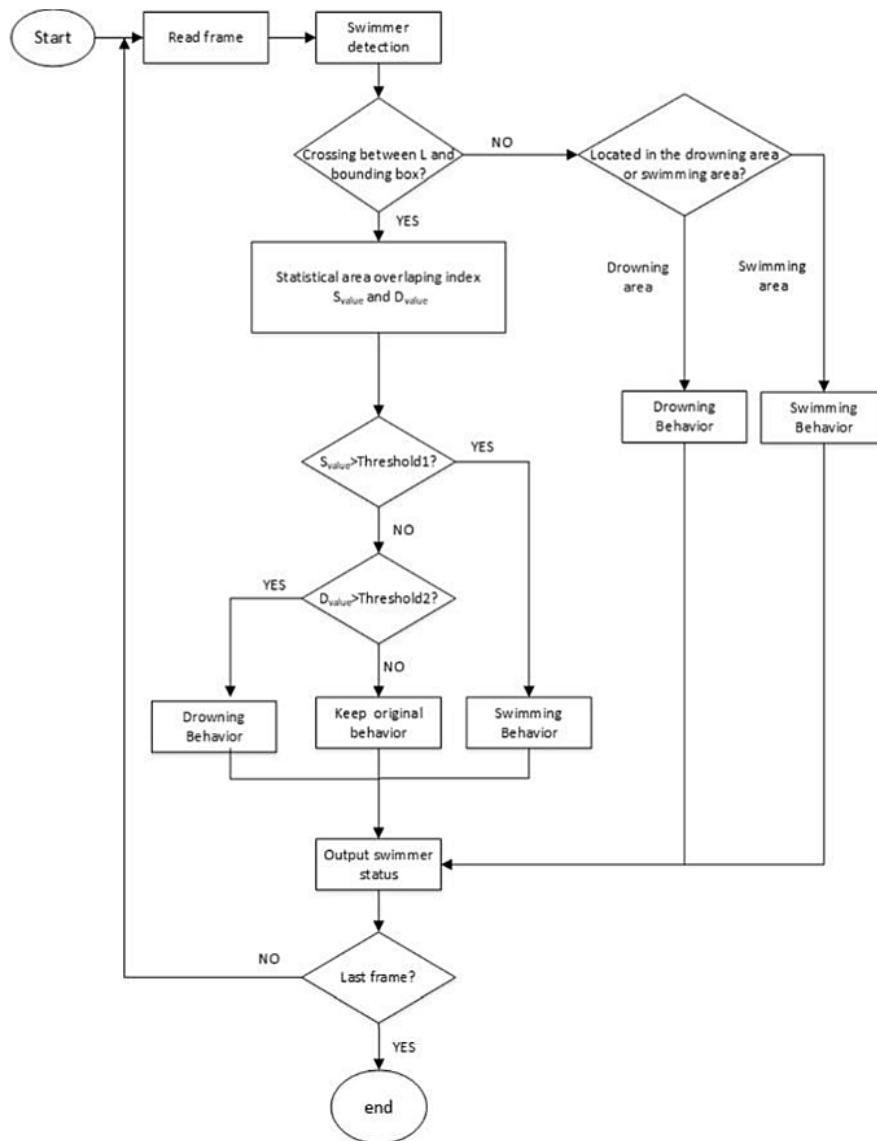
FR No.	Functional Requirement(Epic)	Sub Requirement(Story/Sub-Task)
FR-1	Installation	Needed to be fixed under the water without creating any disturbance to the people in the swimming pool.
FR-2	Deduction	Either horrified or in unconscious
FR-3	Audio	Ask for help or stay quiet if the person is unconscious
FR-4	Support	Take swim tubes or take the help of rescuer
FR-5	PriorAlert	Send alert message to the lifeguard

### 4.2 NON - FUNCTIONAL REQUIREMENTS

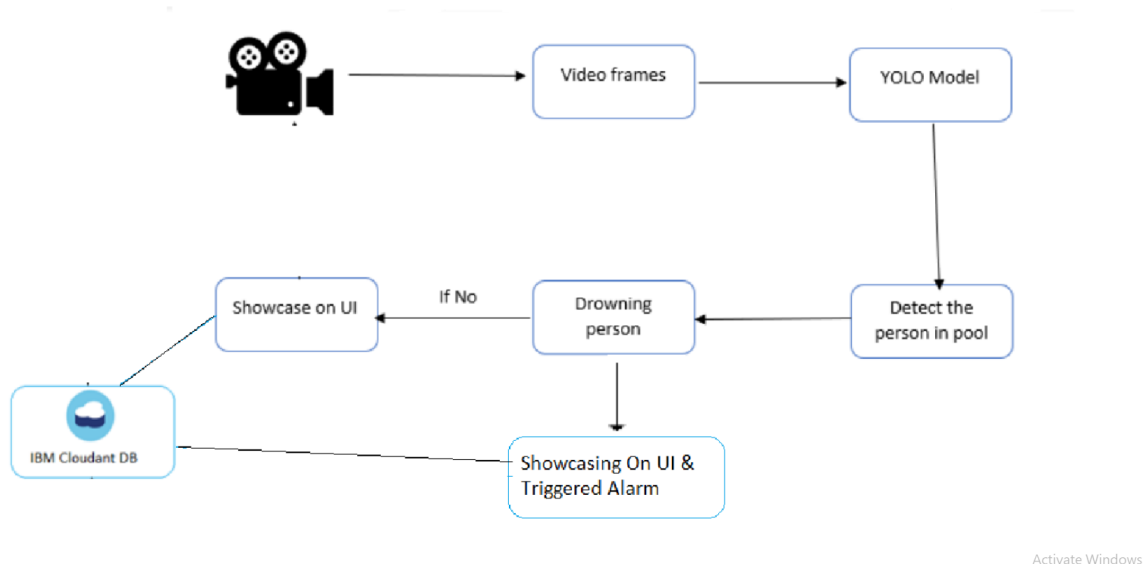
FRNo.	Non-Functional Requirement	Description
NFR-1	Usability	To ensure the safety of each and every person present in the pool. A Lifeguard should be present all the time in the pool.

## 5. PROJECT DESIGNS

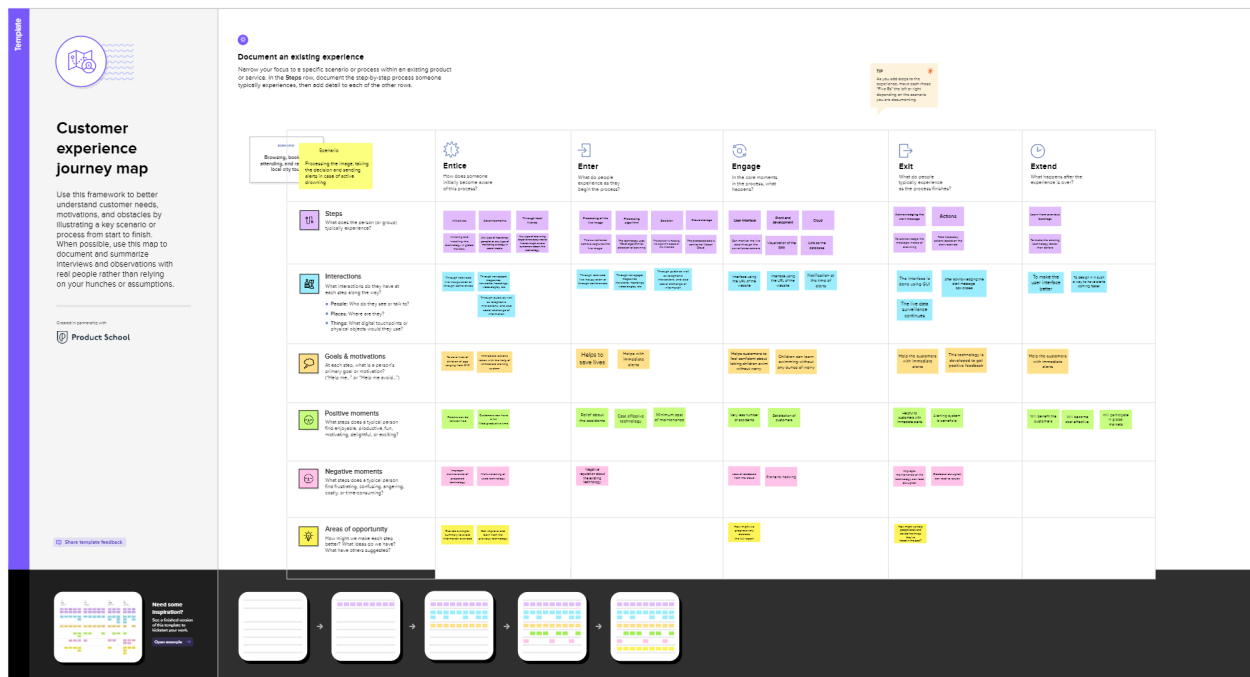
### 5.1 DATA FLOW DIAGRAMS



## 5.2 SOLUTION AND TECHNICAL ARCHITECTURE



## 5.3 USER STORIES



## 6. PROJECT PLANNING AND SCHEDULING

### 6.1 Sprint Planning & Estimation

Sprint	Functional Requirement(Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	I can register for the application by entering my phone number.	1	High	Prasanna P
		USN-2	I will receive confirmation OTP once I have registered for the application.	2	Low	Prasanna P

		USN-3	I can also register for the application through Phone Number/Email	2	Medium	Prasanna P
	Log in	USN-4	I can login into the application by entering email or phone number & password.	1	High	Jefri Jebason J
		USN-5	In prediction page, the data uploaded will help the user to detect the drowning movements	2	Medium	Jefri Jebason J

Sprint-1	Dataset collection	USN-6	The dataset collected will give high accuracy on the drowning details of the person.	2	High	Jefri Jebason J
Sprint-2	Data Pre-processing	USN-7	The dataset is extracted and is used to train the model.	4	High	Prakash G
	Train the model	USN-8	We will train the model.	8	High	Prakash G
		USN-9	We will test the model.	6	High	Prakash G
Sprint-3	Detection	USN-10	The tested model will be loaded.	3	High	Nithesh Paul
		USN-11	To identify the person by collecting real-time data.	5	Medium	Nithesh Paul
		USN-12	The data collected at present is checked with the pre-fed data.	8	High	Nithesh Paul

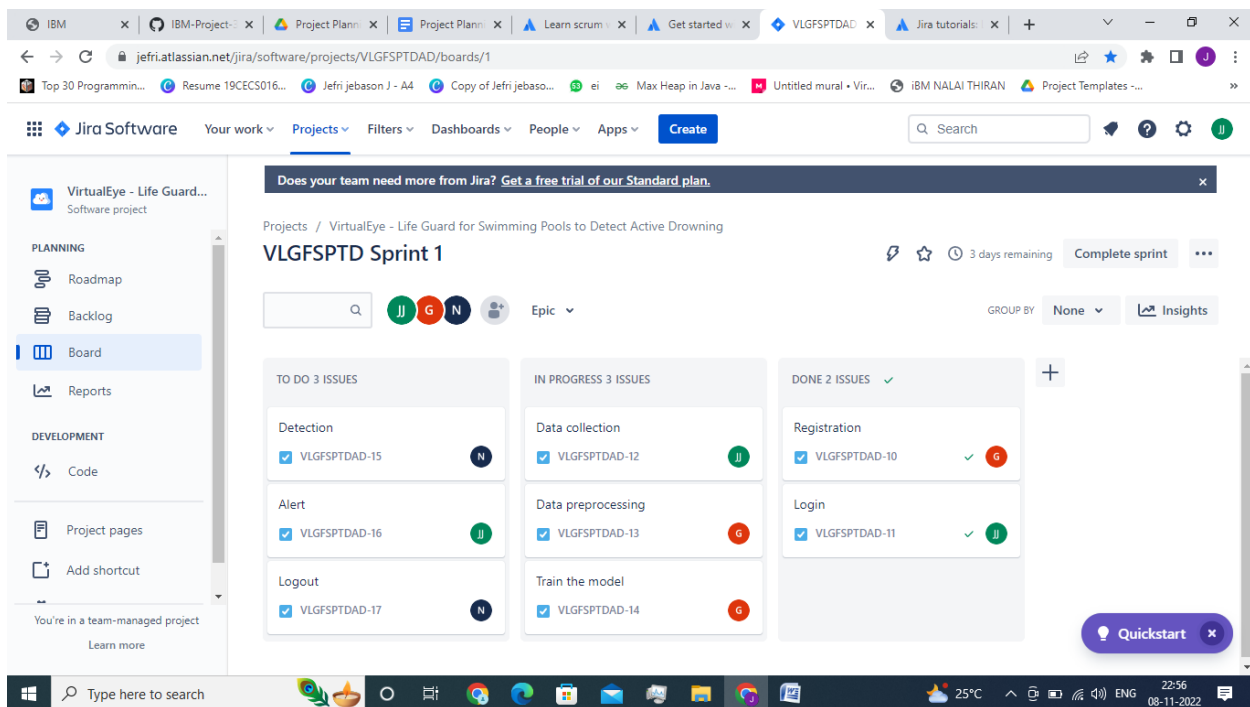
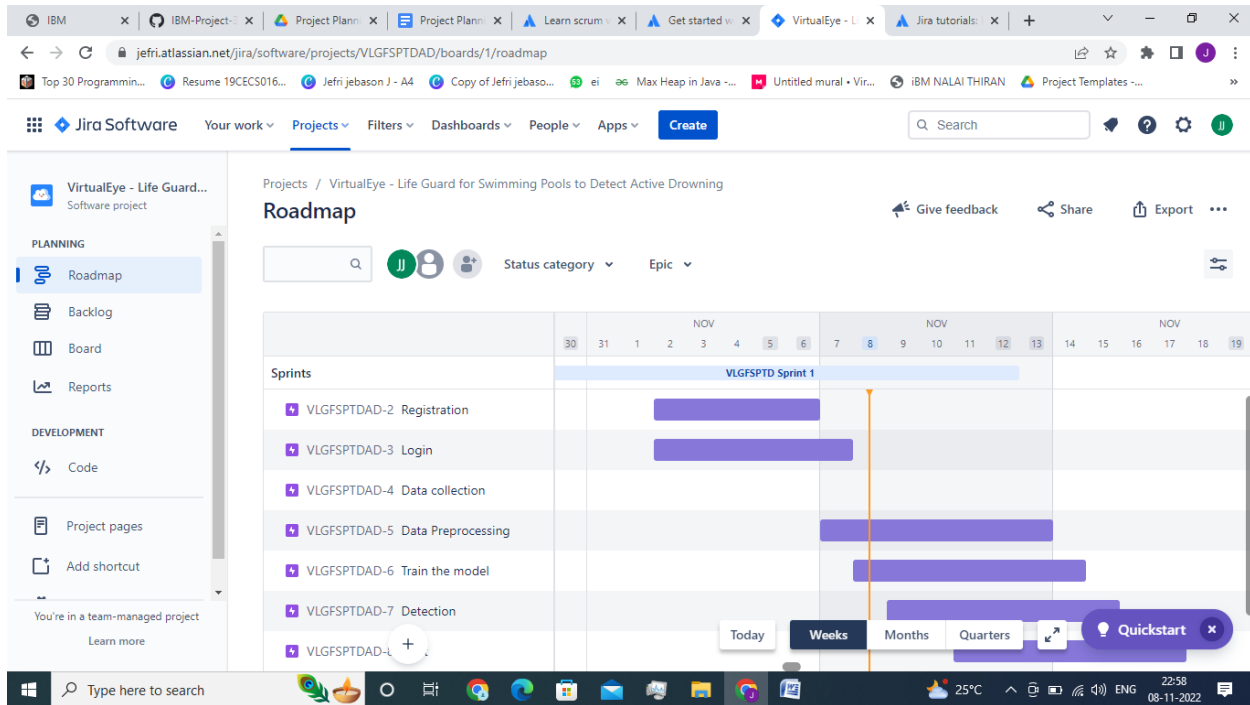
Sprint-4	Alert	USN-13	When the abnormal movement is detected the system will ring an alarm to notify the lifeguard to rescue the person.	7	High	Prakash G
		USN-14	We will be able to detect the drowning person.	3	Medium	Prakash G

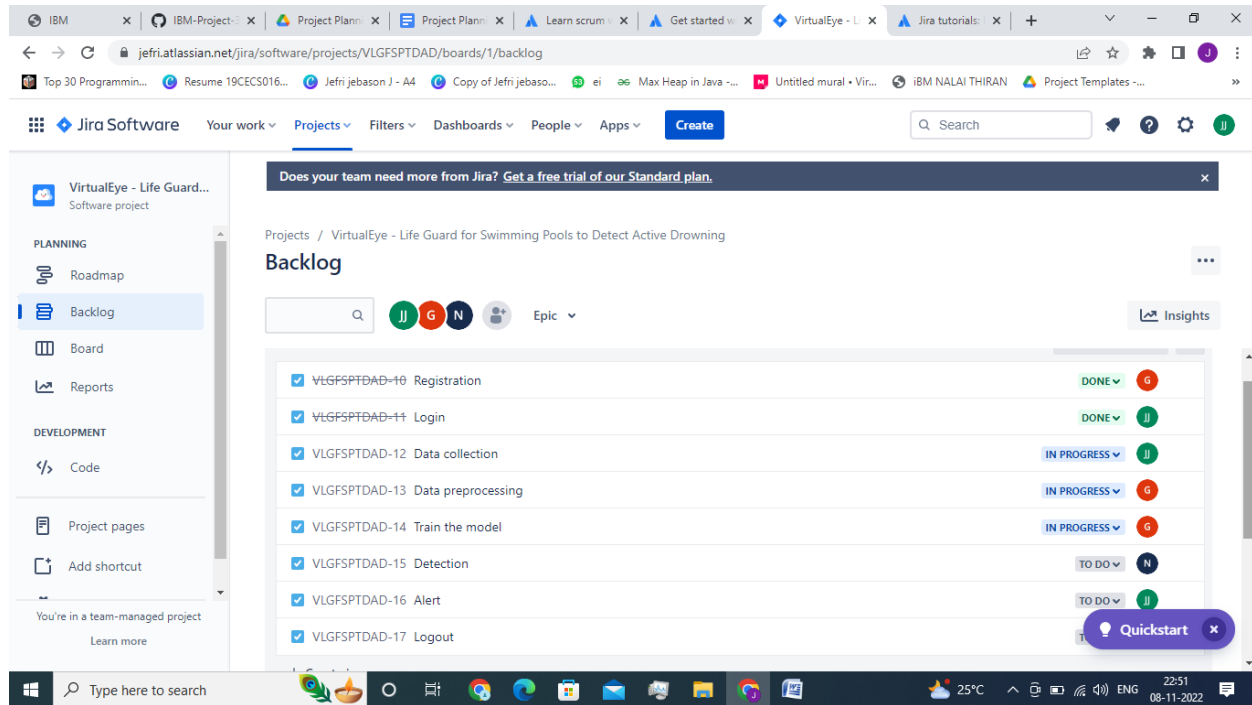
## 6.2 Sprint Delivery Schedule

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	10	6 Days	24 Oct 2022	29 Oct 2022	20	05 Nov 2022
Sprint-2	18	6 Days	31 Oct 2022	05 Nov 2022	20	10 Nov 2022
Sprint-3	16	6 Days	07 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	12	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

## 6.3 Reports from JIRA







## 7. CODING AND SOLUTIONING :

### 7.1 FEATURE 1

```
import os
from cloudant.client import Cloudant
from flask import Flask, flash, redirect, render_template,
request, url_for, Response
from werkzeug.utils import secure_filename
from detect import detect
UPLOAD_FOLDER = "static/uploads/"
RESULTS_FOLDER = "static/results/"
app = Flask(__name__)
```

```

app.secret_key = "secret-key"
app.config["UPLOAD_FOLDER"] = UPLOAD_FOLDER
API_KEY = "api key"
USERNAME = "username"
databaseName = "virtual_eye"
client = Cloudant.iam(USERNAME, API_KEY, connect=True)
@app.route("/")
def index():
    return render_template("index.html", static_folder="static")
@app.route("/register", methods=["GET", "POST"])
def register():
    if request.method == "POST":
        # Get the form data
        36
        try:
            email = request.form["email"]
            password = request.form["password"]
            # Create a database using an initialized client
            my_database = client.create_database(databaseName)
            # Check that the database doesn't already exist
            if my_database.exists():
                print(f"'{databaseName}' successfully created.")
            # Create a JSON document
            json_document = {
                "_id": email,
                "email": email,
                "password": password,
            }
            if email in my_database:
                return render_template("register.html",
                    msg="Email already exists")
            else:

```

```

# Create a document using the Database API
new_document =
my_database.create_document(json_document)
return render_template(
"register.html", msg="Account created
successfully!"
)
except Exception as e:
return render_template(
"register.html", msg="Something went wrong!
Please try again"
)
if request.method == "GET":
return render_template("register.html")
37
@app.route("/login", methods=["GET", "POST"])
def login():
if request.method == "POST":
email = request.form["email"]
password = request.form["password"]
my_database = client[databaseName]
# Check that the database exists
if email in my_database and
my_database[email]["password"] == password:
return redirect(url_for("predict"))
else:
return render_template("login.html", msg="Invalid
credentials!")
if request.method == "GET":
return render_template("login.html")
@app.route("/predict", methods=["GET", "POST"])
def predict():

```

```

if request.method == "POST":
    if "file" not in request.files:
        flash("No file part")
        return redirect(request.url)
    file = request.files["file"]
    if file.filename == "":
        flash("No video selected for uploading")
        return redirect(request.url)
    else:
        filename = secure_filename(file.filename)
        file.save(os.path.join(app.config["UPLOAD_FOLDER"],
        filename))
        return render_template(
        "predict.html",
        msg="Video uploaded successfully",
        filename=filename,
        38
        )
    if request.method == "GET":
        return render_template("predict.html")
    @app.route("/response/<string:filename>", methods=["GET",
    "POST"])
    def response(filename):
        print(filename)
        return Response(
        detect(
        os.path.join(app.config["UPLOAD_FOLDER"], filename),
        ),
        mimetype="multipart/x-mixed-replace; boundary=frame",
        )
    @app.route("/logout", methods=["GET"])
    def logout():

```

```
return render_template("logout.html")
if __name__ == "__main__":
    app.run(debug=True)
```

## 7.2 FEATURE 2

```
import os
from cloudant.client import Cloudant
from flask import Flask, flash, redirect, render_template,
request, url_for, Response
from werkzeug.utils import secure_filename
from detect import detect
UPLOAD_FOLDER = "static/uploads/"
RESULTS_FOLDER = "static/results/"
app = Flask(__name__)
app.secret_key = "secret-key"
app.config["UPLOAD_FOLDER"] = UPLOAD_FOLDER
API_KEY = "api key"
USERNAME = "username"
databaseName = "virtual_eye"
client = Cloudant.iam(USERNAME, API_KEY, connect=True)
@app.route("/")
def index():
    return render_template("index.html", static_folder="static")
@app.route("/register", methods=["GET", "POST"])
def register():
    if request.method == "POST":
        # Get the form data
        try:
            email = request.form["email"]
            password = request.form["password"]
```

```

# Create a database using an initialized client
my_database = client.create_database(databaseName)
# Check that the database doesn't already exist
if my_database.exists():
    print(f"'{databaseName}' successfully created.")
# Create a JSON document
json_document = {
    "_id": email,
    "email": email,
    "password": password,
}
if email in my_database:
    return render_template("register.html",
        msg="Email already exists")
else:
    # Create a document using the Database API
    new_document =
    my_database.create_document(json_document)
    return render_template(
        "register.html", msg="Account created
        successfully!"
    )
except Exception as e:
    return render_template(
        "register.html", msg="Something went wrong!
        Please try again"
    )
if request.method == "GET":
    return render_template("register.html")
@app.route("/login", methods=["GET", "POST"])
def login():
    if request.method == "POST":

```

```

email = request.form["email"]
password = request.form["password"]
my_database = client[databaseName]
# Check that the database exists
if email in my_database and
my_database[email]["password"] == password:
return redirect(url_for("predict"))
else:
return render_template("login.html", msg="Invalid
credentials!")
if request.method == "GET":
return render_template("login.html")
@app.route("/predict", methods=["GET", "POST"])
def predict():
if request.method == "POST":
if "file" not in request.files:
flash("No file part")
return redirect(request.url)
file = request.files["file"]
if file.filename == "":
flash("No video selected for uploading")
return redirect(request.url)
else:
filename = secure_filename(file.filename)
file.save(os.path.join(app.config["UPLOAD_FOLDER"],
filename))
return render_template(
"predict.html",
msg="Video uploaded successfully",
filename=filename,
)
if request.method == "GET":

```



```
return render_template("predict.html")
@app.route("/response/<string:filename>", methods=["GET",
"POST"])
def response(filename):
    print(filename)
    return Response(
        detect(
            os.path.join(app.config["UPLOAD_FOLDER"], filename),
        ),
        mimetype="multipart/x-mixed-replace; boundary=frame",
    )
@app.route("/logout", methods=["GET"])
def logout():
    return render_template("logout.html")
if __name__ == "__main__":
    app.run(debug=True)
```

### 7.3 DATABASE SCHEMA

Cloudant Dashboard

af09b4f9-a789-4b0d-9d5f-e4ad2ab21006-bluemix.cloudant.com/dashboard.html







Top 30 Programmin... Resume 19CECS016... Jefri jebason J - A4 Copy of Jefri jebaso... ei Max Heap in Java ~... Untitled mural • Vir... IBM NALAI THIRAN Project Templates ~...

### Databases

Database name

Create Database {} JSON

Your Databases

Name	Size	# of Docs	Partitioned	Actions
login	0 bytes	0	No	  
my_database	33 bytes	1	No	  

Showing 1-2 of 2 databases. Databases per page: 20

Log Out

---

Cloudant Dashboard - database

af09b4f9-a789-4b0d-9d5f-e4ad2ab21006-bluemix.cloudant.com/dashboard.html#database/my\_database/\_all\_docs

Top 30 Programmin... Resume 19CECS016... Jefri jebason J - A4 Copy of Jefri jebaso... ei Max Heap in Java ~... Untitled mural • Vir... IBM NALAI THIRAN Project Templates ~...

### my\_database

Document ID

Options {} JSON

All Documents

Query

Permissions

Changes

Design Documents

Create Document

	id	key	value
<input type="checkbox"/>	jebason500@gmail.com	jebason500@gmail.com	{"rev": "1-480e7510adb9e3e6ec996..."}

Showing document 1 - 1. Documents per page: 20

Log Out

AQI 132

## 8. TESTING

## 8.1 TEST CASES

			Date: 2019-09-22										
			Team ID: PTH-0012 (MIDWEST)										
			Project Name: Virtual Eye - Life Guard for Swimming Pools to Detect Active Drowning										
			Maximum Marks: 4 Marks										
Test case ID	Feature Type	Component	Test Scenario	Pre-conditions	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation (Y/N)	SLD ID	Executed By
HomePage_TC_001	Functional	Home Page	Verify user is able to see the home page or not.		1. Enter URL and click go 2. Verify whether the user is able to see the home page.	Enter URL and click go	User able to see the home page	Working as expected	Pass	N/A	N	--	Aruna R.
			Verify the UI elements in Home Page.		1. Enter URL and click go 3. Verify the UI elements in Home Page.		Application should show below UI elements	Working as expected					Smashika R.
HomePage_TC_002	UI	Home Page			1. Enter URL and click go	Enter URL and click go			pass	N/A	N	--	Eurapana R.
RegisterPage_TC_003	Functional	Register Page	A Register page is able to well input the user data.		1. Enter URL and click go 2. Verify the UI elements in Home Page 3. Click the register button	Click on sign-up home page	Application should show "Incorrect email or password" validation message.	Working as expected	pass	N/A	N	--	Karupana R.
LoginPage_TC_004			Verify user is able to redirect to detect page or not.		1. Enter URL and click go 2. Click on detect button 3. Verify whether the user is redirected to detect page or not.	Click on sign-in home page	Application should show "Incorrect email or password" validation message.	Working as expected					Jasmin R.
	Functional	Sign Page			1. Enter URL and click go 2. Verify the UI elements in Predict Page.	Click on predict button and redirect to predict page	Application should show below UI elements (Dropdown List, detect button).	Working as expected	pass	N/A	N	--	Karupana R., Jasmin R.
PredictPage_TC_005	UI	Predict page	Verify the UI elements in Predict Page.		1. Enter URL and click go 2. Click on Predict button	Clicking on not	Application should show detecting video	Working as expected	pass	N/A	N	--	Aruna R., Smashika R.
PredictPage_TC_006	Functional	Predict page	Verify user is able to select the dropdown value or not.		1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user is redirected to predict page or not. 4. Verify user is able to select the dropdown value or not.	Predicting the video	Application should show the updated video	Working as expected	pass	N/A	N	--	Aruna R., Smashika R.
PredictPage_TC_007	Functional	Predict page	Verify the video		1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user is redirected to predict page or not. 4. Verify user is able to select the dropdown value or not. 5. Verify the video.		Application should show the predicted output.	Working as expected	pass	N/A	N	--	Aruna R., Smashika R., Jasmin R., Eurapana R.
PredictPage_TC_008	Functional	Predict page	Verify whether the video is predicted correctly or not.		1. Enter URL and click go 2. Click on Predict button 3. Verify whether the user is redirected to predict page or not. 4. Verify user is able to select the dropdown value or not. 5. Verify whether the video is predicted correctly or not.	Click the Detect button.		Working as expected	pass	N/A	N	--	

## 8.2 User Acceptance Testing

### Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Virtual Eye - Life Guard for Swimming Pools to Detect Active Drowning] project at the time of the release to User Acceptance Testing (UAT).

### Defect Analysis

This reports how is 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	6	5	21
Duplicate	4	0	3	0	7
External	1	2	0	1	4
Fixed	14	1	3	8	26
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	4	2	0	6
Totals	26	11	18	19	67

## Test Case Analysis

This report shows the number of testcases that have passed,failed,anduntested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	5	0	0	5
Client Application	30	0	0	30
Security	2	0	0	2
Outsource Shipping	1	0	0	1
Exception Reporting	7	0	0	7
Final Report Output	9	0	0	9
Version Control	1	0	0	1

## 9. RESULTS

### 9.1 PERFORMANCE METRICS

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

```
<head>
```

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

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

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

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

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

```

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

</script>

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

</script>

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

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

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

            </ul>

```

```

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

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

```

```

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

</div>

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

<h3>
    <span id="result"> </span>

</h3>

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

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

</html>

```

**Index.html**

```

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

  <!--Bootstrap -->
    <link rel="stylesheet"
      href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/boo
strap.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-
KJ3o2DKtIkVYIK3UENzmM7KChRr/rE9/Qpg6aAZGJwFDMVNA/GpG
FF93hXpG5KkN" crossorigin="anonymous"></script>
    <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/
popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPsk
vXusvfa0b4Q" crossorigin="anonymous"></script>
    <script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootst
rap.min.js" integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5
+76PVCmYl" crossorigin="anonymous"></script>

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

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

```

```

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

```

```

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

```

```

<div class="body">
<div class="left">
    <h2>Problem:</h2>
    <p>

```

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

```

</p>

```

</div>

<div class="left">

<h2>Solution:</h2>

<p>

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

</p>

</div>

</div>

<div class="bottom">

<p><b>

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.

</b></p>

</div>

</section>

<section id="footer">

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

<div class="social">

<a href="#" target="\_blank"><i class="fab fa-2x fa-twitter-square"></i></a>

<a href="#" target="\_blank">

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

<a href="#">

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

```
</a>
</div>
</section>
</body>
</html>
```

## Logout.html

```
<!DOCTYPE html>
<html >

<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">

    <title>Virtual Eye</title>
    <link href='https://fonts.googleapis.com/css?family=Pacifico'
rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Arimo'
rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Hind:300'
rel='stylesheet' type='text/css'>
    <link href='https://fonts.googleapis.com/css?family=Open+Sans+Condensed:300'
rel='stylesheet' type='text/css'>

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

    <style> .header
{
```

```

        top:0; margin:0px;
        left: 0px; right:
        0px; position:
        fixed;
        background-color: #28272c;
        color: white; box-shadow: 0px 8px 4px
        grey; overflow: hidden; padding-
        left:20px; font-family: 'Josefin
        Sans'; font-size: 2vw;
        width: 100%;
        height:8%;
        text-align: center;
    }
    .topnav { overflow:
    hidden;
    background-color: #333;
}

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

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

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

.topnav-right {
float: right;
padding-right:1
00px;

```

```
}
```

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

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

```
.main{  
margin-top:100px; textalign:center;  
form { margin-  
left:400px;marginright: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;  
marginbottom:8  
px; 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/boo
tstrap.min.css" integrity="sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGg
FAW/dAiS6JXm" crossorigin="anonymous">
<script src="https://code.jquery.com/jquery-3.2.1.slim.min.js" integrity="sha384-
KJ3o2DKtIkVYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpG
FF93hXpG5KkN" crossorigin="anonymous"></script>
<script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/
popper.min.js" integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPsk
vXusvfa0b4Q" crossorigin="anonymous"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootst
rap.min.js" integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5
+76PVCmYl" crossorigin="anonymous"></script>

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

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

```

```

        <li><a
href="{ { url_for('logout') } }">Logout</a></li>
        <!-- <li><a href="#about">About</a></li>
        <li><a href="#services">Services</a></li> -->

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

<div    class="body">    <div
class="left">
    <p>
        Swimming is one of the best exercises that helps people to reduce
stress in this urban lifestyle. Swimming pools are found larger in number
in the hotels, weekend tourist spots and barelypeople have in their house
backyard. Beginners, especially oftenfeel 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 besuffering
the highest drowning mortality rates worldwide..Such kinds of deaths
account for the third cause of unplanned death globally, with about 1.2
million cases yearly.

    </p>
</div>
<div class="left">

```

```

        <div class="prediction-input">
         </br>
        <form id="form" action="/result" method="post"
        enctype="multipart/form-data">

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

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

</section>

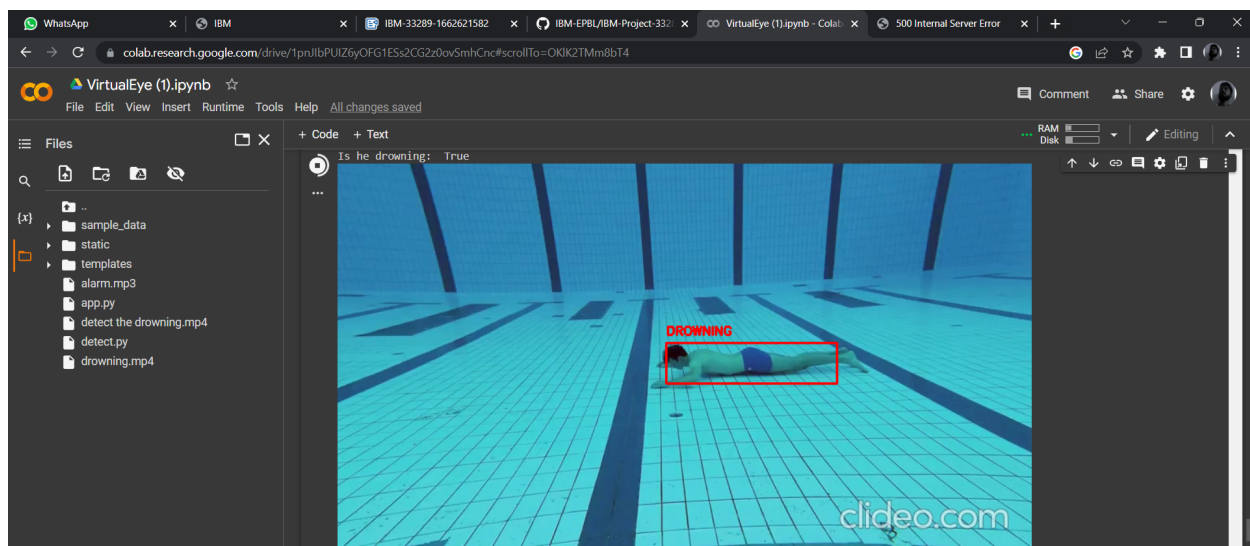
<br><br>

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

</section>
</body>

</html>

```



## **10.ADVANTAGES AND DISADVANTAGES**

### **10.1 ADVANTAGES**

- The system will monitor everyone in the pool and if it notices someone isn't moving for a certain period of time, it will trigger alarms and send alerts to the lifeguards monitoring device.
- To continuously monitor the pool, our software closely integrates with the cameras already in place.
- The built-in notification system produces alarms within 10 seconds on the lifeguard's monitoring device.
- The proposed system can work in real-time on edge devices, making rescue operations effortless.

### **10.2 DISADVANTAGES**

- Underwater live cameras are exorbitant.
- Swimming pools may have potential blind spots due to their size and shape.
- Risk that such systems can create a false sense of security for lifeguards.
- Concerns over inconsistent levels of reliability of systems and situations where glare, and high occupancy activity rates can cause false alarms.

## **11.CONCLUSION**

In this paper, we proposed a method for efficient drowning detection. With the help of the Yolo V3 model we have detected the person and their drowning condition if a person does not move or moved slowly for 10 sec then the alarm is sent to the lifeguard. This system may be further extended for future scope.

## **12.FUTURE SCOPE**

Finetuning the YOLOv5 model would also result in better and more efficient predictions. Better datasets, current approaches, and technologies with great processing power, along with high-quality surveillance cameras, will assist to increase the accuracy of drowning detection and can even be employed under bad conditions. After all of these requirements are met, this method can be used to detect drowning on seashores.

## **13.APPENDIX**

**GITHUB LINK : <https://github.com/IBM-EPBL/IBM-Project-33289-1660218119>**

**YOUTUBE LINK : <https://youtu.be/LV5vgeXyPQg>**