CHAPTER 1

INTRODUCTION

INTRODUCTION

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

1.1 Project Overview

Although there have been few studies of applied visual search that have examined this domain, lifeguard surveillance is a challenging task that is essential for swimmer safety. In the present study, dynamic, naturalistic stimuli—video clips of cooperative swimmers—were used to compare the search abilities of lifeguards and non-lifeguards with varying set sizes and types of drowning. Lifeguards were more precise and reacted more quickly to targets who were drowning. Additionally, different responses were observed for different types of drowning targets, demonstrating that passive drowning may be less obvious but is nonetheless highly instructive once identified. Passive drowning was detected less frequently but more quickly than active drowning. At an intermediate set-size level, set size effects showed a decline in reaction times, pointing to a potential change in visual search strategies as the array size grows.

1.2 Purpose

Because drowning is the third most common cause of unintentional death, reliable security measures must be developed. Through the use of human action detection, the project's goal is to develop a system that can recognise drowning incidents in swimming pools automatically. The system will receive video that has been captured using live security cameras and will process and categorise it using a drowning detection model. In order for the lifeguards to begin their rescue efforts, the system will break this video up into image frames and apply a model to it. If any early drowning behaviours, such as hand waving, water splashing, or diving, are identified, the system will sound an alarm. The system uses the state-of-the-art

YOLOv5 object detection model to detect persons in each frame and check if they are drowning.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing problem

A victim of drowning cannot yell for help because the water is blocking their mouth, so the process is silent. Gravity will cause the drowning victim's body to fall to the pool floor as soon as they have consumed enough water, gained enough weight, and reached the bottom. After a while, the stomach and lungs will produce bacteria, and the corpse will begin to float on water. Depending on the state of the water, this floating could take several hours or several days. The chances of survival without severe brain or organ damage are higher if the drowning victim is removed from the water within five or six minutes. Existing drowning detection methods include wearable sensor-based systems and visionbased systems. A second division of vision-based technologies is between those that employ underwater cameras and those that employ above-water cameras. The disadvantage of underwater cameras is that they don't capture the initial battle above the water. A crucial concern to take into account in a timecritical emergency is the possibility of a longer rescue time if a drowning incident is not recognised as soon as possible.

The biggest drawback of a wearable-based system is its pain, which may cause younger children to attempt to remove the gadget in order to feel better. However, this idea is unproven.

2.2 References

Title :

Automated Drowning Detection And Security in Swimming Pool.

Authors:

Kanchana A, Kavya G.R, Kavitha C, Soumyashree V, Salila Hedge (Department of Electronics and Communication).

Year :

2019

Description:

Swimming pool surveillance systems plays an essential role in safeguarding the premises. In this project, differential pressure approach is used for detection of drowning incidents in swimming pools at the earliest possible stage. The automated drowning detection system works on the principle of differential pressure. The system contains two fundamental modules: to begin with the wristband consisting of pressure sensors on the transmitter side. Second, the receiver module at the swimming pool territory should wear the wristband. The pressure at underwater is different and greater than the pressure at the air-water interface. The pressure at a particular depth is measured and set as the threshold.

Pros:

- The children’s life is saved during drowning incidents in this swimming pool by lifting the acrylic plate.
- The demo system uses a pressure sensor, which has the advantages of being convenient, economical, and having a straightforward algorithm.

Cons:

- The reflection and refraction of light in air water interference will affect the image quality and drowning man feature in this method does not easily distinguish swimmers and divers obviously.

Title :

A Novel Drowning Detection method for safety of swimmers.

Authors:

Ajil Roy, Dr.K.Srinivasan (Department of Instrumentation andControl Engineering).

Year :

2018

Description:

Effective drowning detection methods are essential for the safety of swimmers. In this paper, a novel type of drowning detection method addressing many limitations of prevailing drowning detectors is proposed. The proposed method ensures detection of drowning and reporting at the earlier stages. The proposed drowning detection method is also a generic solution that suites different water bodies from pools to oceans, and an economically viable method useful for both low and middle income countries. The prototype of the drowning detection method is developed and demonstrated and model of the system is simulated in Proteus design suite. The results of the simulation and hardware experimentation are also reported.

Pros:

- The alarm receivers can be placed at different locations in the water bodies which are having high chance of drowning.
- Another major advantage of this approach unlike other approach is the ease of use in all atmospheric conditions, like rain or wind to day or night.

Cons:

Title :

-
  - The average time a child of age between 5-10 years can hold their breath for 10 seconds underwater.

This feature should be valid only if the GPS connectivity was alive with minimum of 10 minutes before the drowning, as a very old GPS value will give a wrong
location itself.

A Novel Drowning Detection method for safety of swimmers.

Authors:

Ajil Roy, Dr.K.Srinivasan (Department of Instrumentation andControl Engineering).

Year :

2018

Description:

Effective drowning detection methods are essential for the safety of swimmers. In this paper, a novel type of drowning detection method addressing many limitations of prevailing drowning detectors is proposed. The proposed method ensures detection of drowning and reporting at the earlier stages. The proposed drowning detection method is also a generic solution that suites different water bodies from pools to oceans, and an economically viable method useful for both low and middle income countries. The prototype of the drowning detection method is developed and demonstrated and model of the system is simulated in Proteus design suite. The results of the simulation and hardware experimentation are also reported.

Title :

-

Pros:

- The alarm receivers can be placed at different locations in the water bodies which are having high chance of drowning. Another major advantage of this approach unlike other approach is the ease of use in all atmospheric conditions, like rain or wind to day or night.

Cons:

The average time a child of age between 5-10 years can hold their breath for 10 seconds underwater.

- This feature should be valid only if the GPS connectivity was alive with minimum of 10 minutes before the drowning, as a very old GPS value will give a wrong
location itself.

Automated and Intelligent System for Monitoring Swimming PoolSafety Based on the IoT and Transfer Learning.

Author:

AzizAlotaibi (College of Computer and Information Technology).

Year :

2020

Description:

Recently, integrating the Internet of Things (IoT) and computer vision has been utilized in swimming pool automated surveillance systems. Several studies have been proposed to overcome off-time surveillance drowning incidents based on using a

Title :

-

sequence of videos to track human motion and position. This paper proposes an efficient and reliable detection system that utilizes a single image to detect and classify drowning objects, to prevent drowning incidents. The performance of the specialized model is evaluated by using a prototype experiment that achieves higher accuracy, sensitivity, and precision, as compared to other deep learning algorithms. The collected data from different physical devices were processed by using ML techniques, to generate an action value.

Pros:

- This system utilizes the IoT and transfer learning to provide an intelligent and automated solution for off-time monitoring swimming pool safety.

  A specialized transfer-learning-based model utilizing a model pre-trained on "ImageNet", which can extract the most useful and complex features of the captured image to differentiate between humans, animals, and other objects.

Cons:

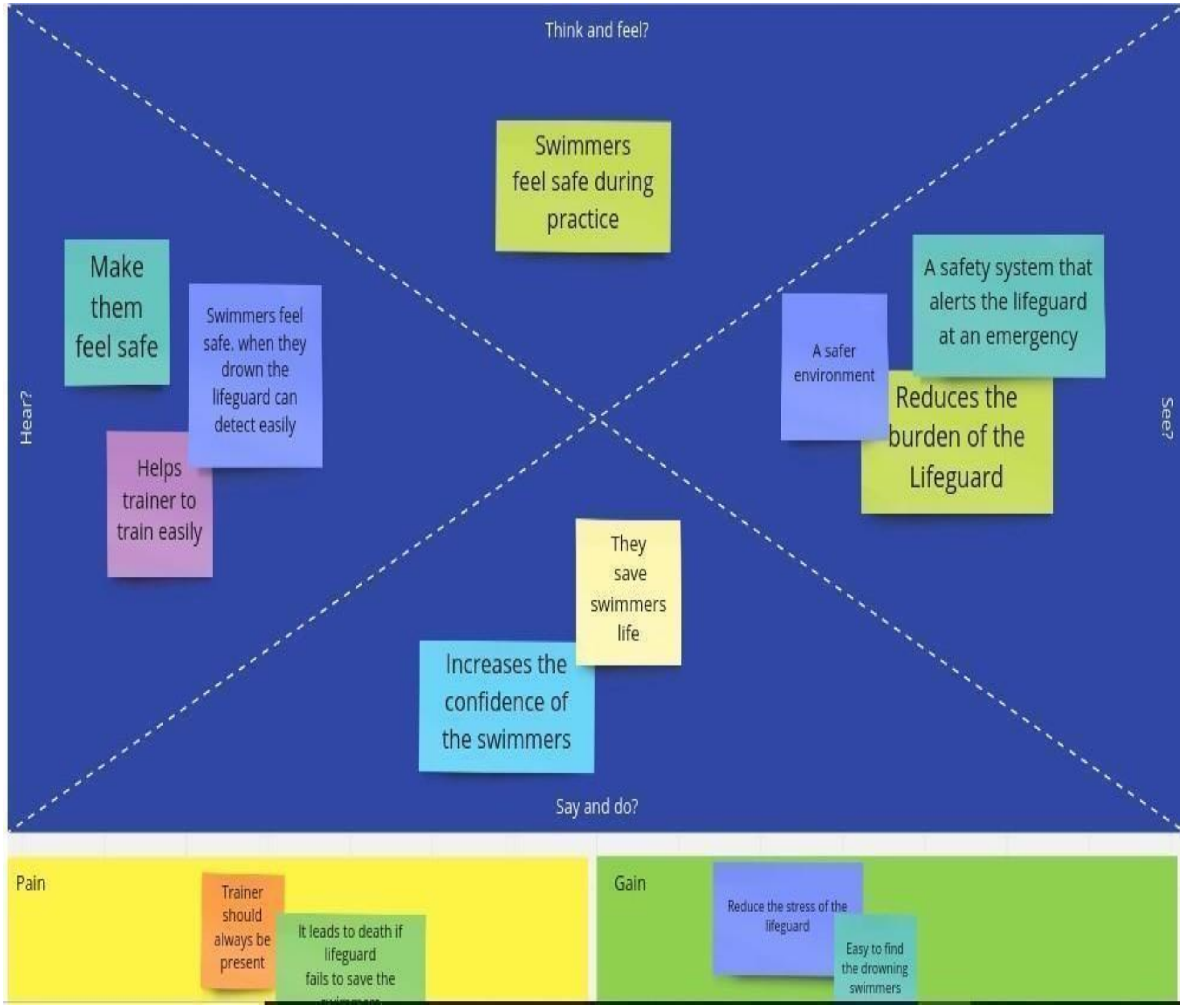
- A generative adversarial network should be applied to generate synthesis data, in order to increase the size of the training dataset.
- More classes should be added to explore and investigate the efficiency.

1.3 Problem Statement Definition

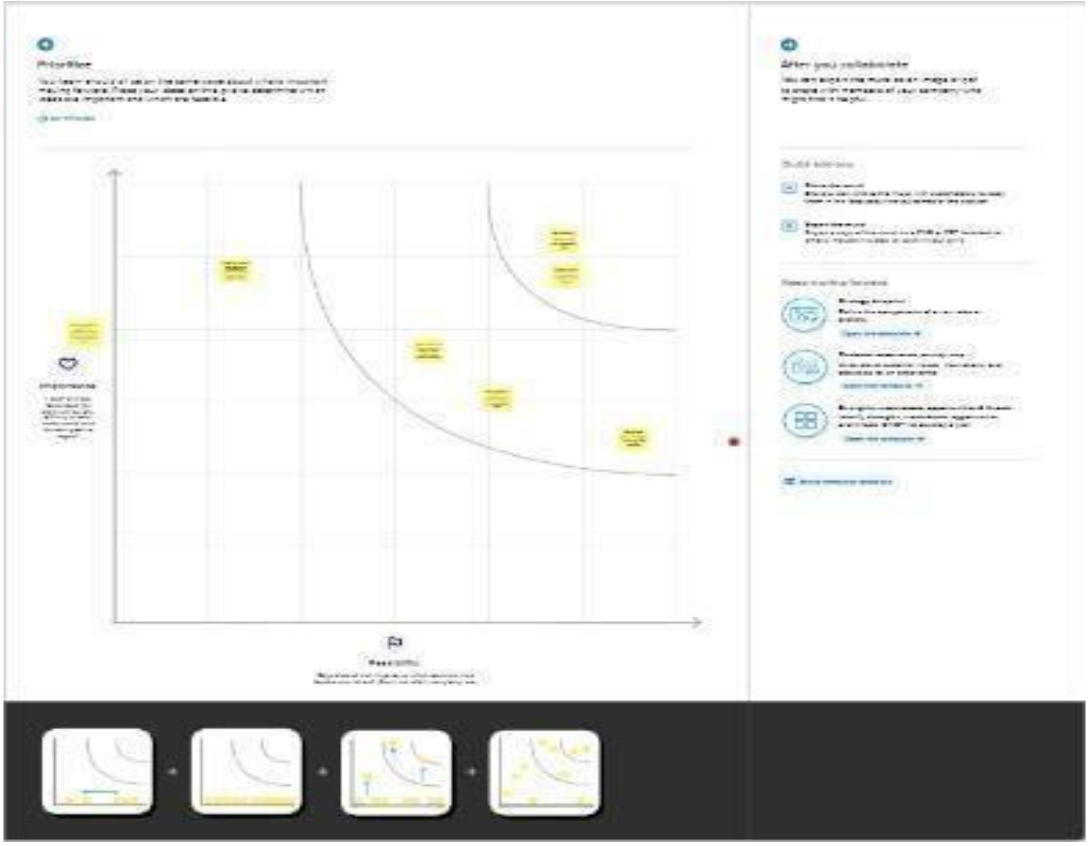
Safety is the top priority in all swimming areas. Due to their technical characteristics, such as underwater cameras, and methodological aspects, like the requirement for human engagement in the rescue mission, the present solutions supposed to handle the issue of maintaining safety at swimming pools have serious issues. The effective reduction of drowning and assurance of pool safety can be achieved through the implementation of an automated visual-based monitoring system. This study proposes a ground-breaking system that triggers an alarm to help to drown victims after quickly identifying them. It can identify a drowning person in three phases using the state-of-the-art object detection model called YOLOv5. Then the detections are processed in real-time to check for drowning swimmers and trigger an alarm.

CHAPTER 3 IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas


Think and feel?
Swimmers feel safe during practice
Hear?
Make them feel safe
Swimmers feel safe, when they drown the lifeguard can detect easily
Helps trainer to train easily
See?
A safety system that alerts the lifeguard at an emergency
A safer environment
Reduces the burden of the Lifeguard
They save swimmers life
Increases the confidence of the swimmers
Say and do?
Pain
Trainer should always be present
It leads to death if lifeguard fails to save the
Gain
Reduce the stress of the lifeguard
Easy to find the drowning swimmers

3.2 Ideation & Brainstorming



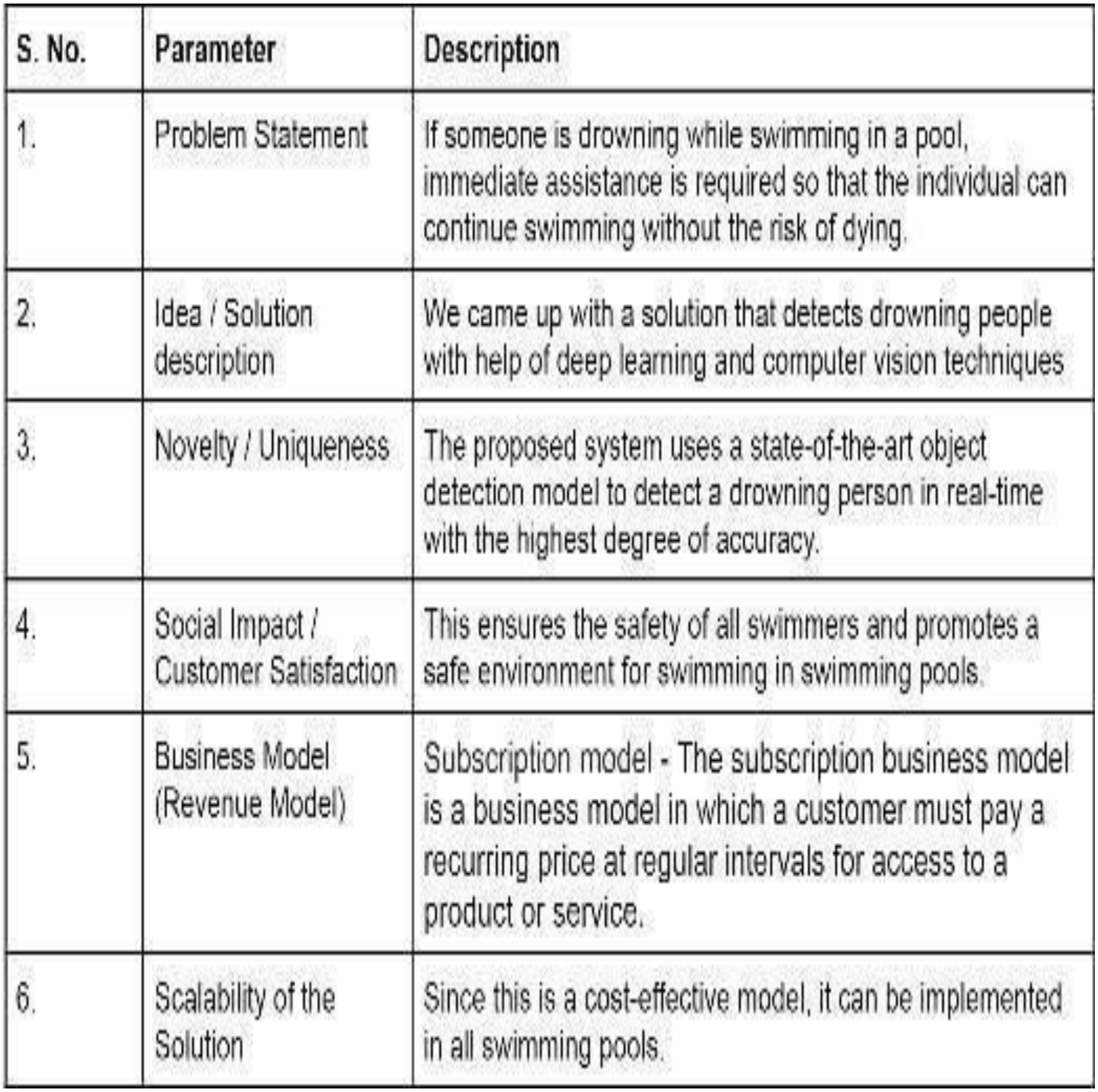
Prioritize

[illegible]

After you collaborate

[illegible]

3.3 Proposed Solution



S. No.	Parameter	Description
1.	Problem Statement	If someone is drowning while swimming in a pool, immediate assistance is required so that the individual can continue swimming without the risk of dying.
2.	Idea / Solution description	We came up with a solution that detects drowning people with help of deep learning and computer vision techniques
3.	Novelty / Uniqueness	The proposed system uses a state-of-the-art object detection model to detect a drowning person in real-time with the highest degree of accuracy.
4.	Social Impact / Customer Satisfaction	This ensures the safety of all swimmers and promotes a safe environment for swimming in swimming pools.
5.	Business Model (Revenue Model)	Subscription model - The subscription business model is a business model in which a customer must pay a recurring price at regular intervals for access to a product or service.
6.	Scalability of the Solution	Since this is a cost-effective model, it can be implemented in all swimming pools.

Problem Solution Fit:

The Problem-Solution Fit simply means that you have found a problem with your customer and that the solution you have realized for it actually solves the customer's problem. It helps entrepreneurs, marketers and corporate innovators identify behavioral patterns and recognize what would work and why.

Purpose:

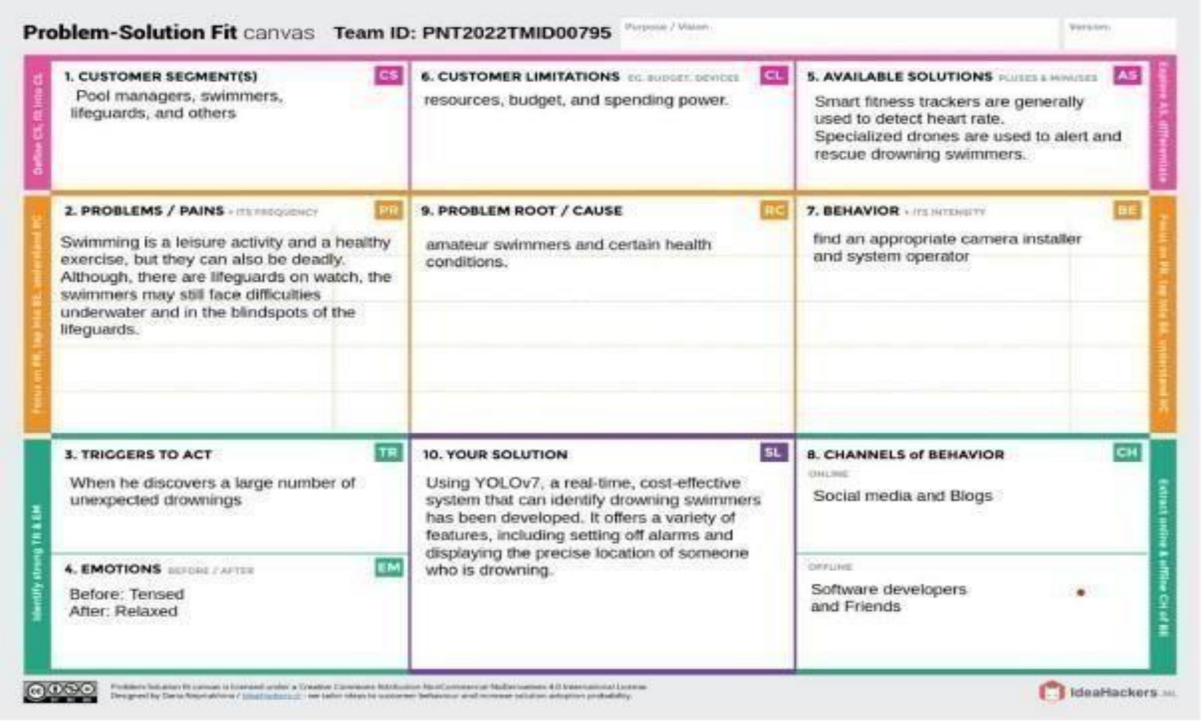
❑ Solve complex problems in a way that fits the state of your customers.

❑ Succeed faster and increase your solution adoption by tapping into existing mediums and channels of behavior.

❑ Sharpen your communication and marketing strategy with the right triggers and messaging.

❑ Increase touch-points with your company by finding the right problem-behavior fit and building trust by solving frequent annoyances, or urgent or costly problems.

❑ Understand the existing situation in order to improve it for your target group**.**



Problem-Solution Fit canvas **Team ID: PNT2022TMID00795** Purpose / Vision Version:

1. CUSTOMER SEGMENT(S) CS Pool managers, swimmers, lifeguards, and others	**6. CUSTOMER LIMITATIONS** EG. BUDGET, DEVICES CL resources, budget, and spending power.	**5. AVAILABLE SOLUTIONS** PLUSES & MINUSES AS Smart fitness trackers are generally used to detect heart rate. Specialized drones are used to alert and rescue drowning swimmers.
2. PROBLEMS / PAINS + ITS FREQUENCY PR Swimming is a leisure activity and a healthy exercise, but they can also be deadly. Although, there are lifeguards on watch, the swimmers may still face difficulties underwater and in the blindspots of the lifeguards.	**9. PROBLEM ROOT / CAUSE** RC amateur swimmers and certain health conditions.	**7. BEHAVIOR** + ITS INTENSITY BE find an appropriate camera installer and system operator
3. TRIGGERS TO ACT TR When he discovers a large number of unexpected drownings **4. EMOTIONS** BEFORE / AFTER EM Before: Tensed After: Relaxed	**10. YOUR SOLUTION** SL Using YOLOv7, a real-time, cost-effective system that can identify drowning swimmers has been developed. It offers a variety of features, including setting off alarms and displaying the precise location of someone who is drowning.	**8. CHANNELS of BEHAVIOR** CH ONLINE Social media and Blogs OFFLINE Software developers and Friends

Define CS, fit into CL
Explore AS, differentiate
Focus on PR, tap into BE, understand RC
Focus on PR, tap into BE, understand RC
Identify strong TR & EM
Extract online & offline CH of BE
IdeaHackers.nl

CHAPTER 4REQUIREMENT ANALYSIS

4.1. Functional requirements

Following are the functional requirements of the proposed solution

Fr No	**Functional Requirements**	**Sub Requirement**
1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIN
2	User Confirmation	Confirmation via Email Confirmation via OTP
3	User Profile	User information Bank details
4	Database	Car database Customer database

5	Features and Technology	Accuracy , efficiency of drowning detection system etc.
6	Feedback	Feedback through Form Feedback through Gmail Feedback through LinkedIN

4.2 Non-Functional requirements

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

FR No.	**Non-Functional Requirement**	**Description**
NFR 1	Usability	Great UI (user interface), Quick adaptation of user.
NFR 2	Security	Aware of fraud and scams, Protect your password and account personal details.
NFR 3	Reliability	Rate of occurrence of failure is less. Failure free.
NFR 4	Performance	Perform value and correct prediction value, The landing page must support several users must provide 5 second or less response time.

NFR 5	Availability	Uninterrupted services must be available all time except the time of server updation.
NFR 6	Scalability	that can handle any amount of data and perform many computations in a cost-effective and timesaving way to instantly serve millions of users residing at global locations.

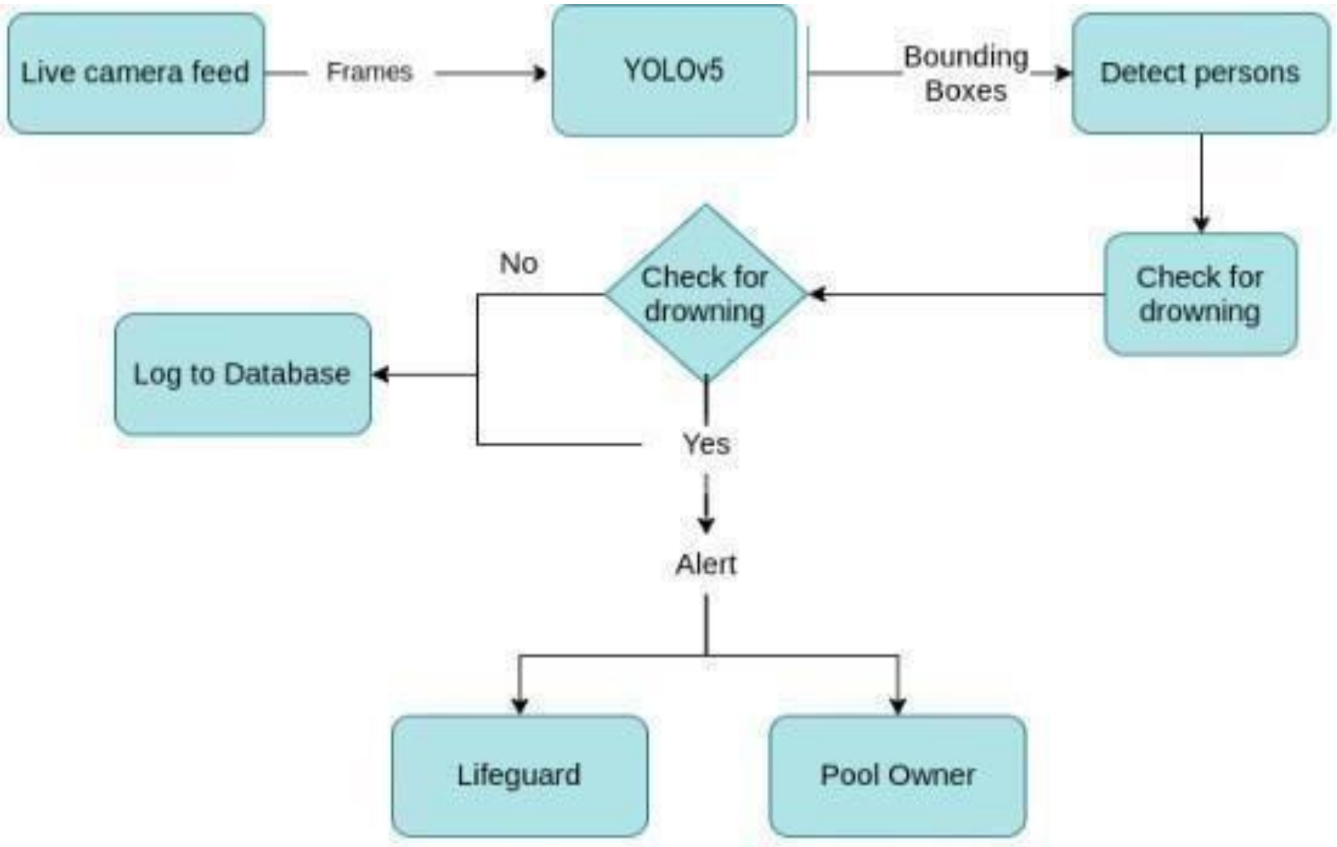
CHAPTER 5

PROJECT DESIGN

5.1 Data Flow Diagrams

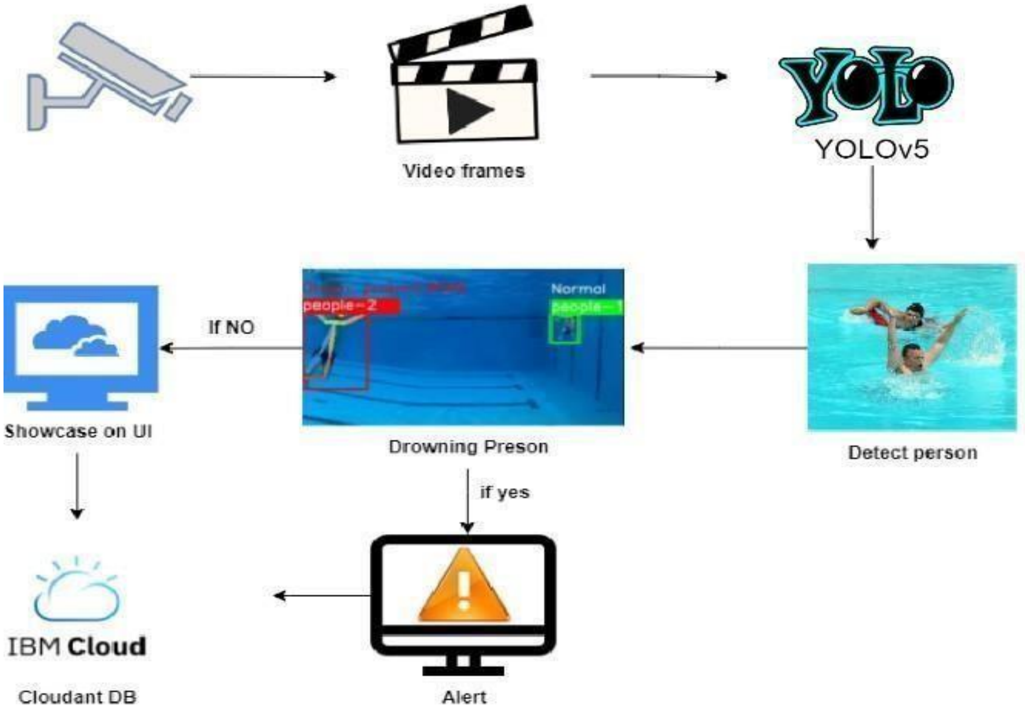
A Data Flow Diagram (DFD) is a traditional visual representation of the information flows within a system. A neat and clear DFD can depict the right amount of the system requirement graphically.

It shows how data enters and leaves the system, what changes the information, and where data is stored.

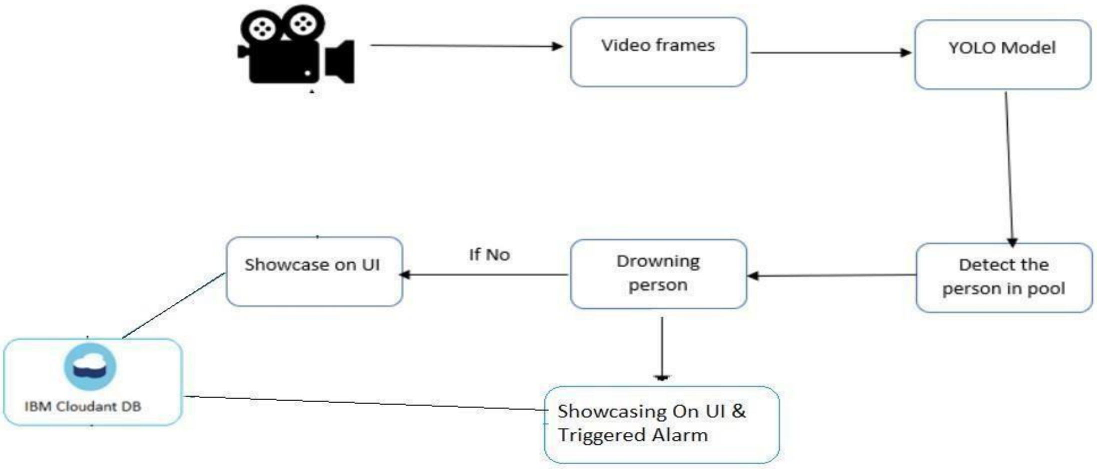


5.2 Solution & Technical Architecture

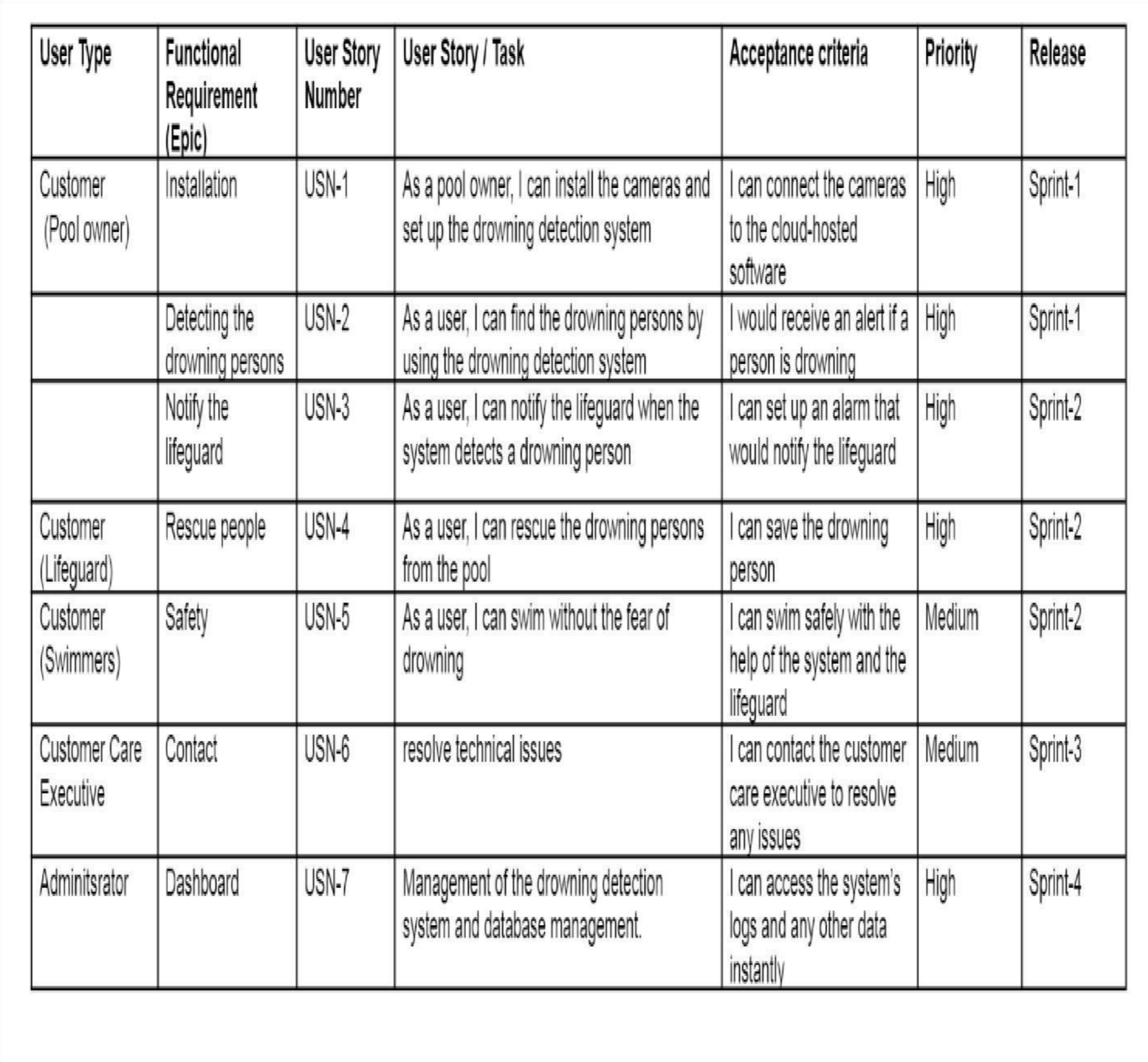
5.2.1 Solution Architecture:



5.2.2 Technical Architecture:



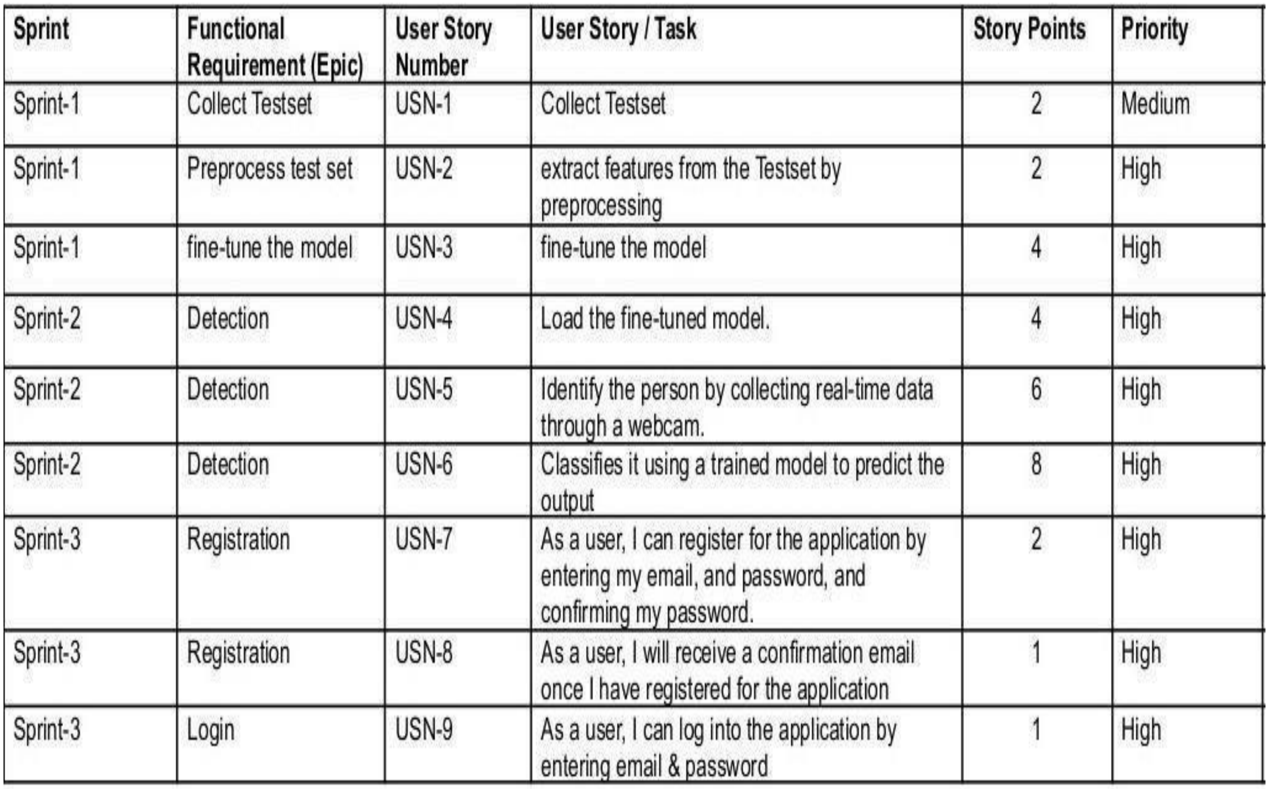
5.3 User Stories



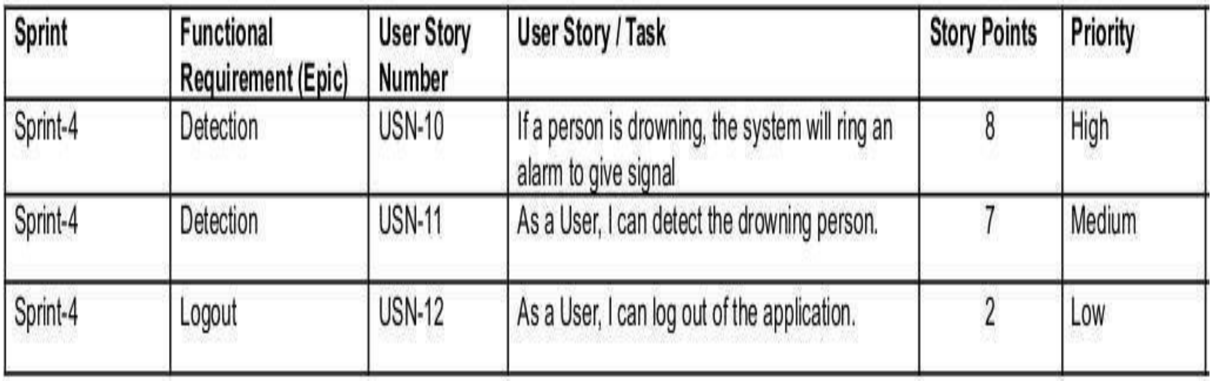
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Pool owner)	Installation	USN-1	As a pool owner, I can install the cameras and set up the drowning detection system	I can connect the cameras to the cloud-hosted software	High	Sprint-1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

CHAPTER 6 PROJECT PLANNING AND SCHEDULING

6.1 Sprint Planning & Estimation

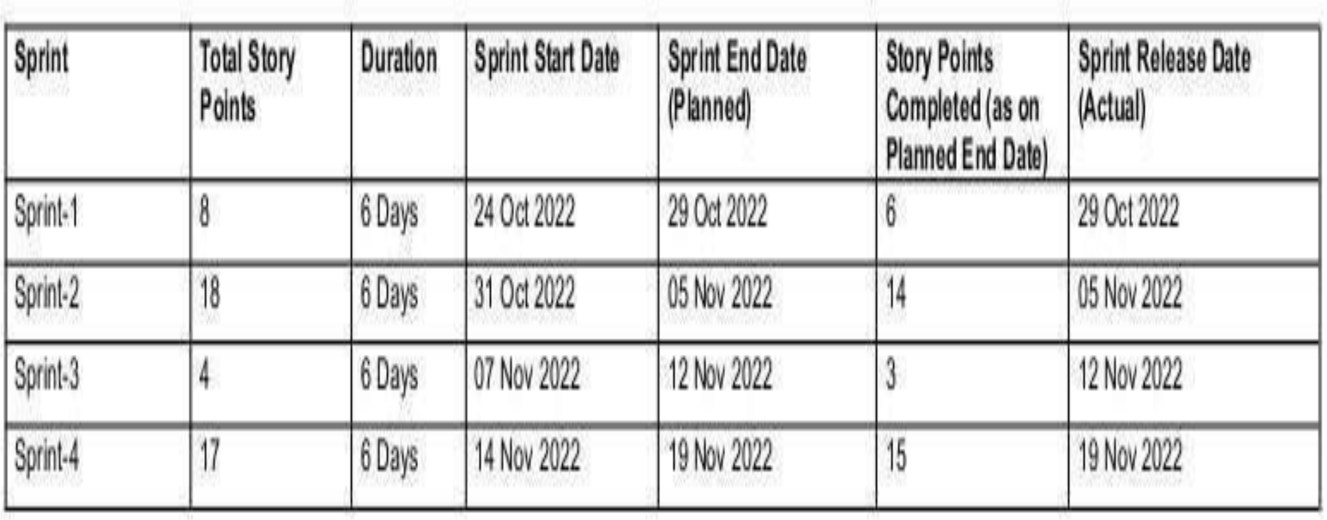


Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-1	Collect Testset	USN-1	Collect Testset	2	Medium
Sprint-1	Preprocess test set	USN-2	extract features from the Testset by preprocessing	2	High
Sprint-1	fine-tune the model	USN-3	fine-tune the model	4	High
Sprint-2	Detection	USN-4	Load the fine-tuned model.	4	High
Sprint-2	Detection	USN-5	Identify the person by collecting real-time data through a webcam.	6	High
Sprint-2	Detection	USN-6	Classifies it using a trained model to predict the output	8	High
Sprint-3	Registration	USN-7	As a user, I can register for the application by entering my email, and password, and confirming my password.	2	High
Sprint-3	Registration	USN-8	As a user, I will receive a confirmation email once I have registered for the application	1	High
Sprint-3	Login	USN-9	As a user, I can log into the application by entering email & password	1	High



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority
Sprint-4	Detection	USN-10	If a person is drowning, the system will ring an alarm to give signal	8	High
Sprint-4	Detection	USN-11	As a User, I can detect the drowning person.	7	Medium
Sprint-4	Logout	USN-12	As a User, I can log out of the application.	2	Low

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	8	6 Days	24 Oct 2022	29 Oct 2022	6	29 Oct 2022
Sprint-2	18	6 Days	31 Oct 2022	05 Nov 2022	14	05 Nov 2022
Sprint-3	4	6 Days	07 Nov 2022	12 Nov 2022	3	12 Nov 2022
Sprint-4	17	6 Days	14 Nov 2022	19 Nov 2022	15	19 Nov 2022

Velocity:

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

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

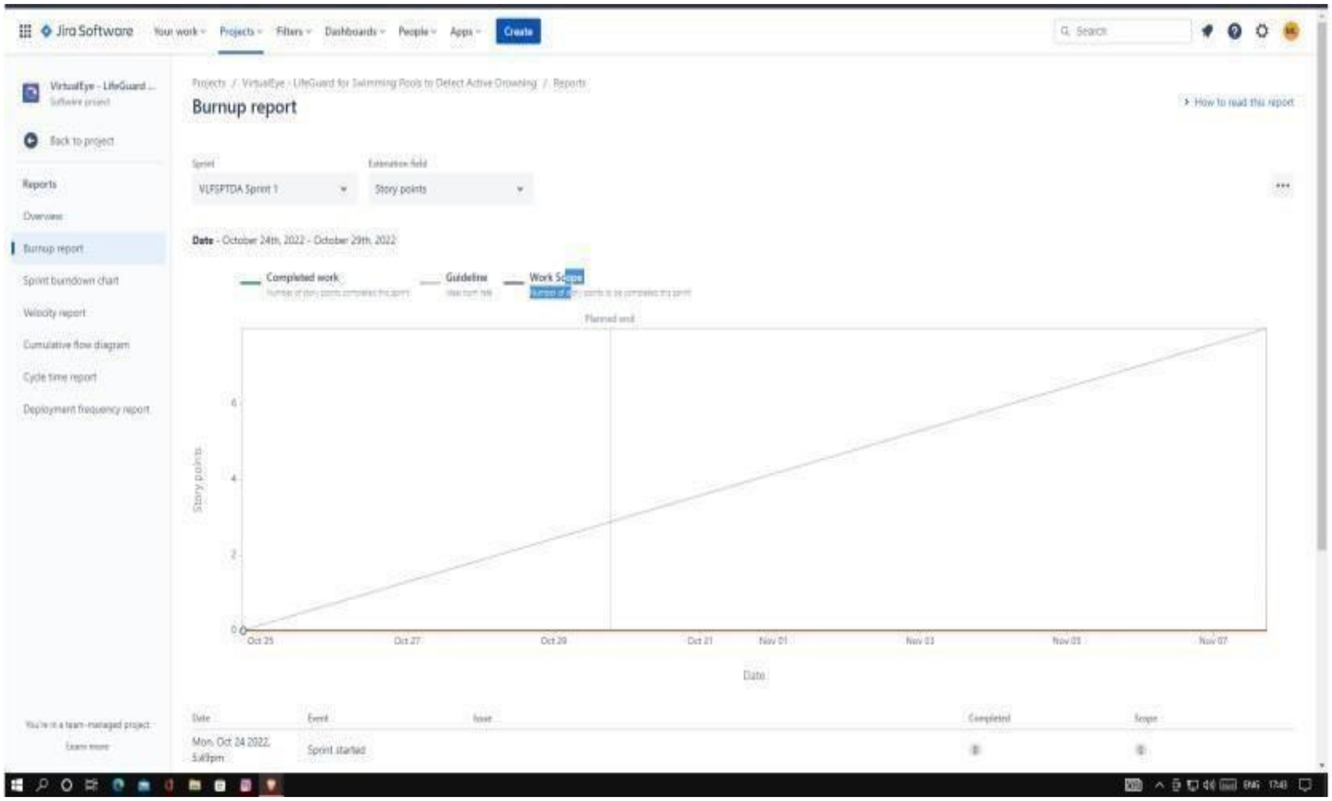
For Sprint-1 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 8 / 6 = 1.33

For Sprint-2 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 18 / 6 = 3

For Sprint-3 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 4 / 6 = 0.66

For Sprint-4 the Average Velocity (AV) is: AV = Sprint Duration / velocity = 17 / 15 = 1.13 **Burndown Chart:**

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



Projects / VirtualEye - LifeGuard for Swimming Pools to Detect Active Drowning / Reports

Burnup report

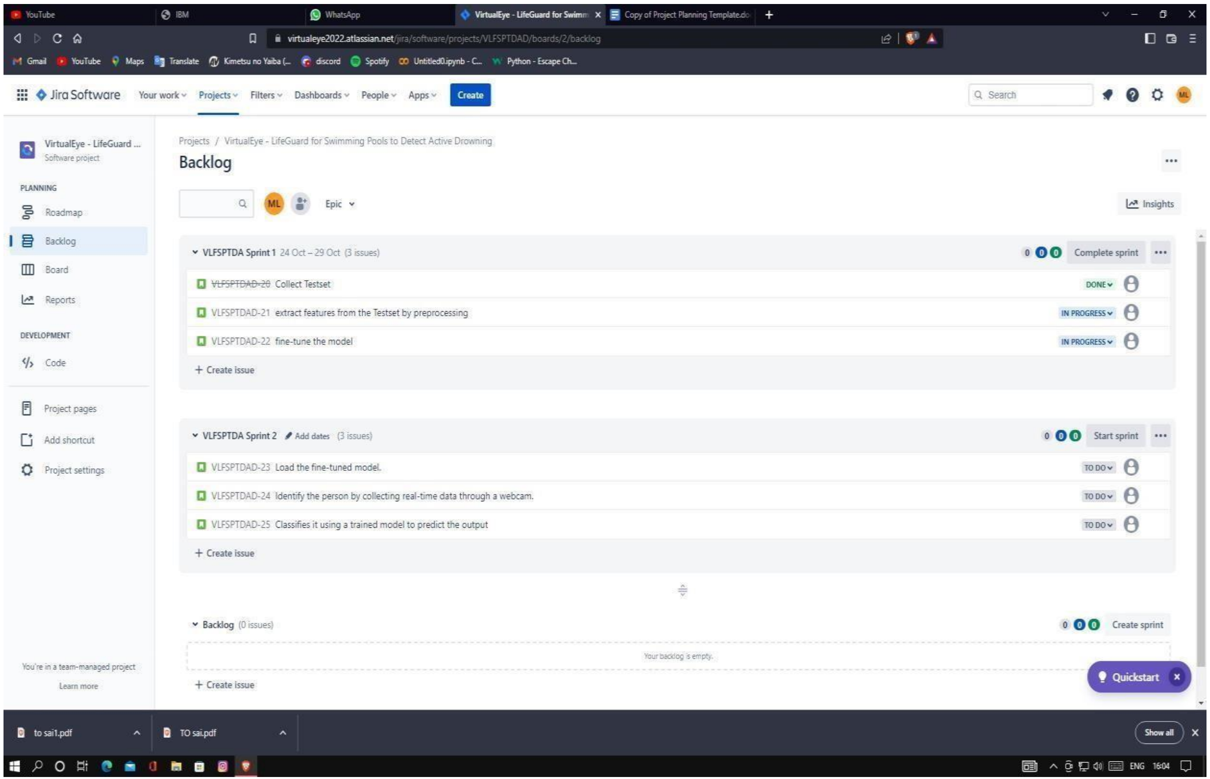
Sprint: VLFSPTDA Sprint 1

Estimation field: Story points

Date - October 24th, 2022 - October 29th, 2022

Date	Event	Issue	Completed	Scope
Mon, Oct 24 2022, 5:49pm	Sprint started		0	0

6.3 Reports from JIRA



Projects / VirtualEye - LifeGuard for Swimming Pools to Detect Active Drowning

Backlog

VLFSPTDA Sprint 1 24 Oct – 29 Oct (3 issues)

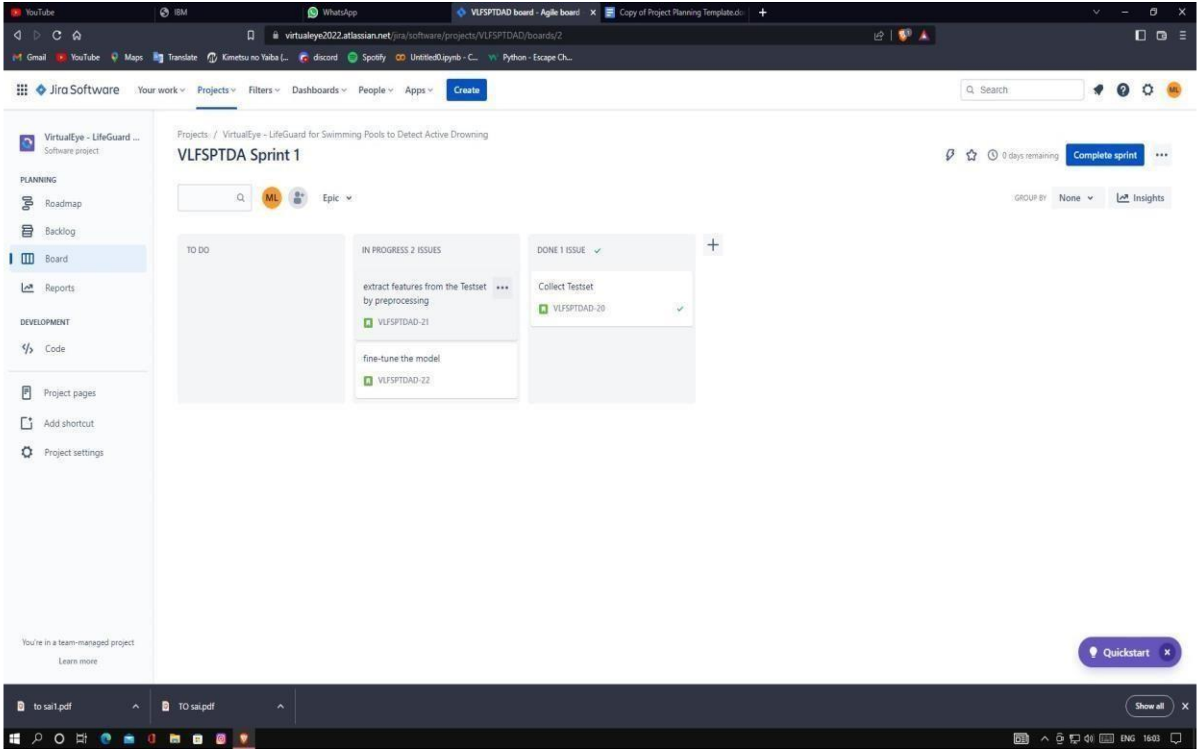
- ~~VLFSPTDAD-20~~ Collect Testset DONE
- VLFSPTDAD-21 extract features from the Testset by preprocessing IN PROGRESS
- VLFSPTDAD-22 fine-tune the model IN PROGRESS

VLFSPTDA Sprint 2 Add dates (3 issues)

- VLFSPTDAD-23 Load the fine-tuned model. TO DO
- VLFSPTDAD-24 Identify the person by collecting real-time data through a webcam. TO DO
- VLFSPTDAD-25 Classifies it using a trained model to predict the output TO DO

Backlog (0 issues)

Your backlog is empty.

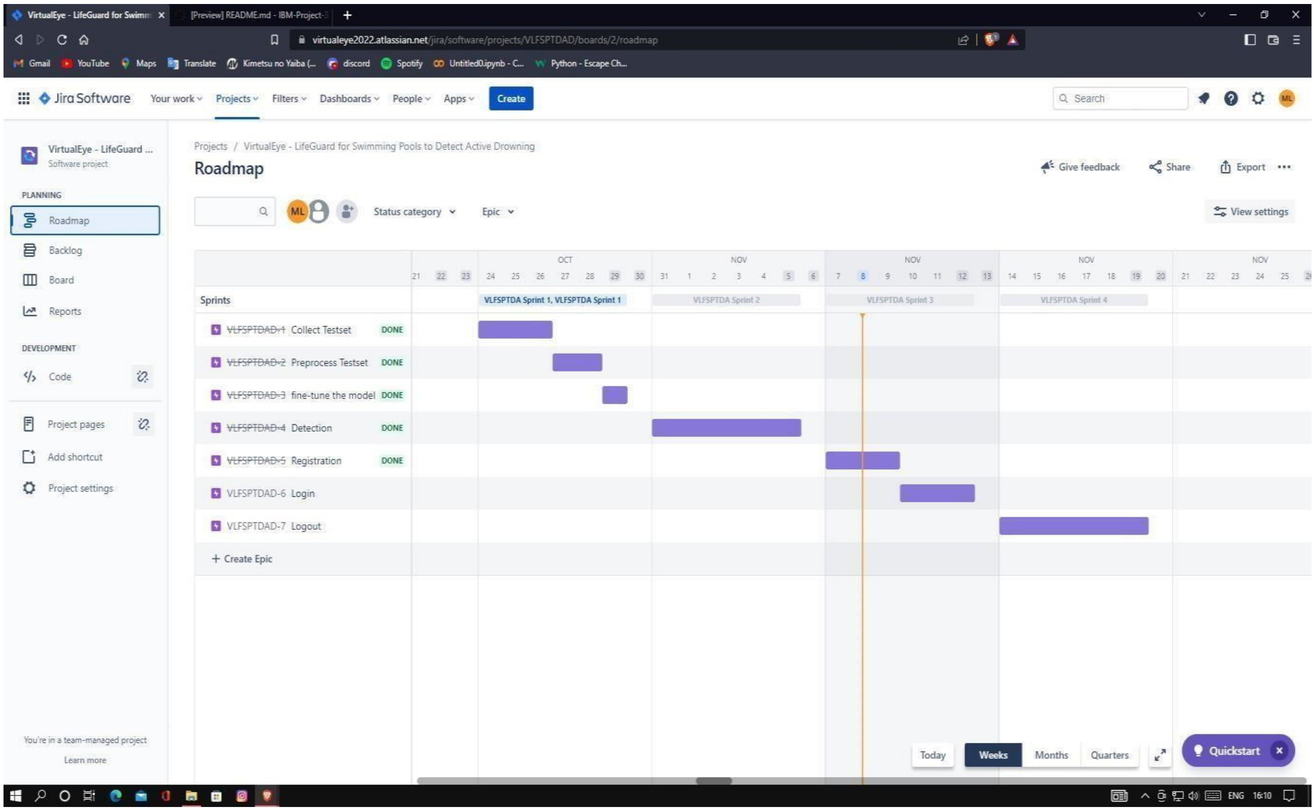


Projects / VirtualEye - LifeGuard for Swimming Pools to Detect Active Drowning

VLFSPTDA Sprint 1

0 days remaining

TO DO	IN PROGRESS 2 ISSUES	DONE 1 ISSUE
	extract features from the Testset by preprocessing VLFSPTDAD-21 fine-tune the model VLFSPTDAD-22	Collect Testset VLFSPTDAD-20



Projects / VirtualEye - LifeGuard for Swimming Pools to Detect Active Drowning

Roadmap

Sprints

- ~~VLFSPTDAD-1~~ Collect Testset DONE
- ~~VLFSPTDAD-2~~ Preprocess Testset DONE
- ~~VLFSPTDAD-3~~ fine-tune the model DONE
- ~~VLFSPTDAD-4~~ Detection DONE
- ~~VLFSPTDAD-5~~ Registration DONE
- VLFSPTDAD-6 Login
- VLFSPTDAD-7 Logout

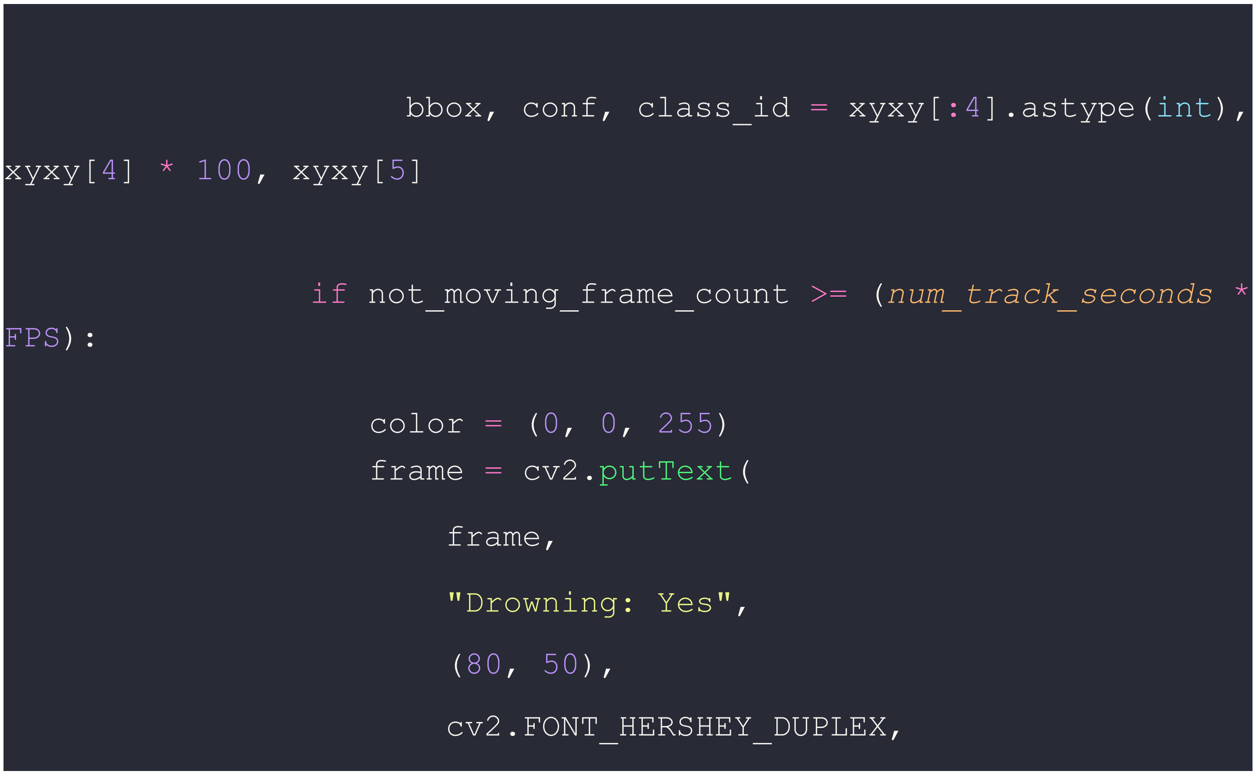
+ Create Epic

CHAPTER 7
CODING & SOLUTIONING

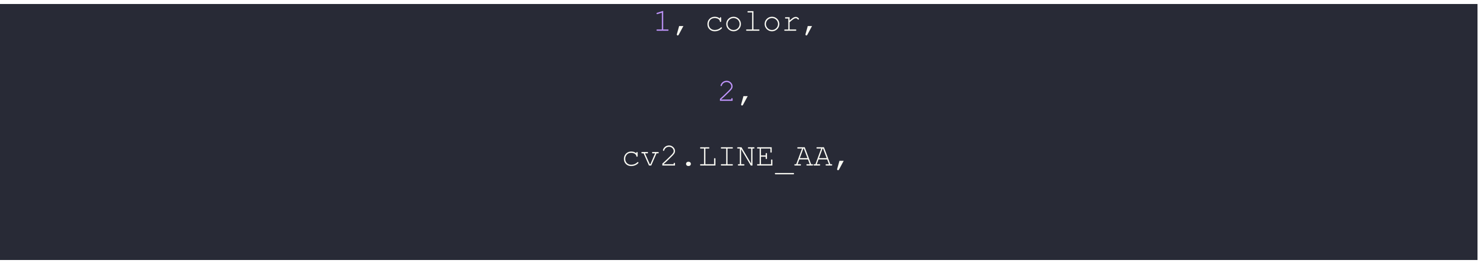
7.1 Feature 1

The person detection feature is implemented using YOLOv5 pre-trained model.

```
import cv2 import torch
from tqdm.auto import tqdm



device = torch.device("cuda" if torch.cuda.is_available() else
"cpu") model
= (
     torch.hub.load("ultralytics/yolov5",  "yolov5s",
pretrained=True).eval().to(device)
)



def  detect(source_path,   num_track_seconds=5,
save_path="static/output.mp4"):

   cap = cv2.VideoCapture(source_path) FPS
   = cap.get(cv2.CAP_PROP_FPS)
   total_frames =
   cap.get(cv2.CAP_PROP_FRAME_COUNT) print("FPS: ", FPS)
   print("Total Frames: ", total_frames)


   imageWidth = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
   imageHeight = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))


   writer = cv2.VideoWriter( save_path,
       cv2.VideoWriter_fourcc("m", "p",
       "4", "v"),
       FPS,
       (imageWidth, imageHeight),
   )
   prev_center = None

   not_moving_frame_count = 0


   for frame_num in tqdm(range(int(total_frames))):

       success, frame = cap.read() if

       success:

           with torch.inference_mode():

               results = model(frame)


           xyxys = results.xyxy[0].cpu().numpy() for

           xyxy in xyxys:

                   center = ((xyxy[0] + xyxy[2]) // 2, (xyxy[1] +
xyxy[3]) // 2)

               # check if the detected object is a person if

               xyxy[-1] == 0 and prev_center is not None:

                   # check for no movement if (

                   abs(prev_center[0] - center[0]) < 20 and

                   abs(prev_center[1] - center[1]) < 20

                   ):

                       not_moving_frame_count += 1


               prev_center = center
```



```
                    bbox, conf, class_id = xyxy[:4].astype(int),
xyxy[4] * 100, xyxy[5]

                if not_moving_frame_count >= (num_track_seconds *
FPS):

                   color = (0, 0, 255)
                   frame = cv2.putText(

                       frame,

                       "Drowning: Yes",

                       (80, 50),

                       cv2.FONT_HERSHEY_DUPLEX,
```



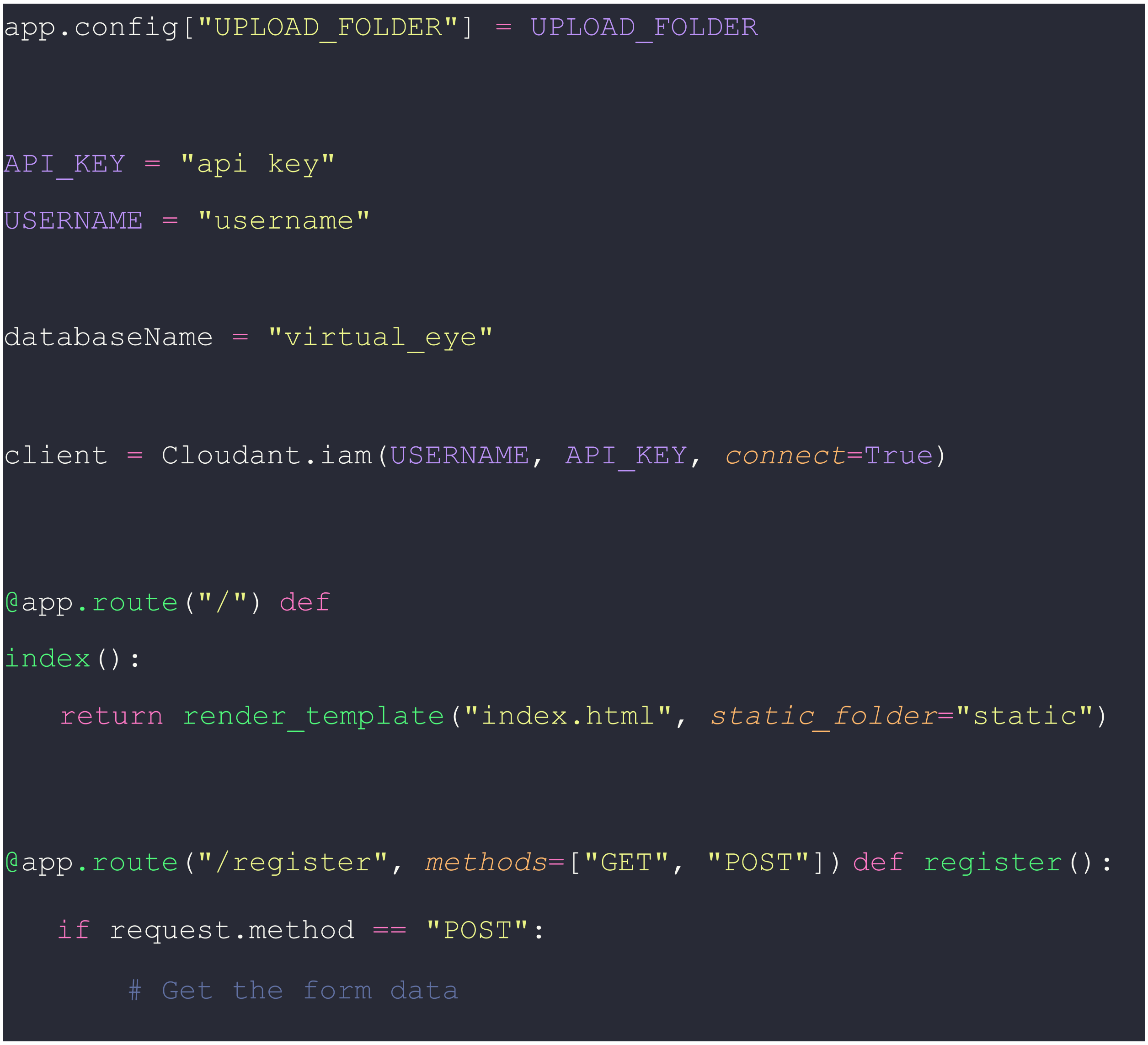
```
                                  1, color,

                                     2,

                                cv2.LINE_AA,
                   )


               else:
                   color = (0, 255, 0) frame
                   = cv2.putText( frame,
                       "Drowning: No",
                       (80, 50), cv2.FONT_HERSHEY_DUPLEX,
                       1, color,



                       2,


                       cv2.LINE_AA,
                   )
                       out_frame = cv2.rectangle(frame, bbox[:2],
bbox[2:], color, 2)
               out_frame = cv2.putText( out_frame,
                   f"conf: {conf:.2f}",bbox[:2],
                   cv2.FONT_HERSHEY_DUPLEX,
                   0.6, color,
                   2, cv2.LINE_AA,
               )
               center_pt = list(map(int, center)) out_frame =
                  cv2.circle(out_frame, center_pt, 3,
color, -1)

               ret, buffer = cv2.imencode(".jpg", out_frame)
```

```
               out_frame_bytes = buffer.tobytes()

               yield (

                   b"--frame\r\n"
                            b"Content-Type: image/jpeg\r\n\r\n" +
out_frame_bytes + b"\r\n"
               )

           writer.write(out_frame)
```

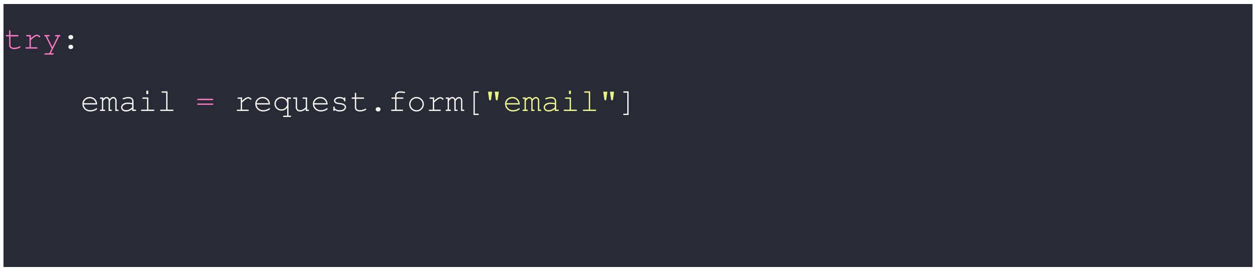
```
   # release resources cap.release()

   cv2.destroyAllWindows()



if __name__ == "__main__": detect("swim.mp4")

   detect("standby.mp4")
```

7.2 Feature 2

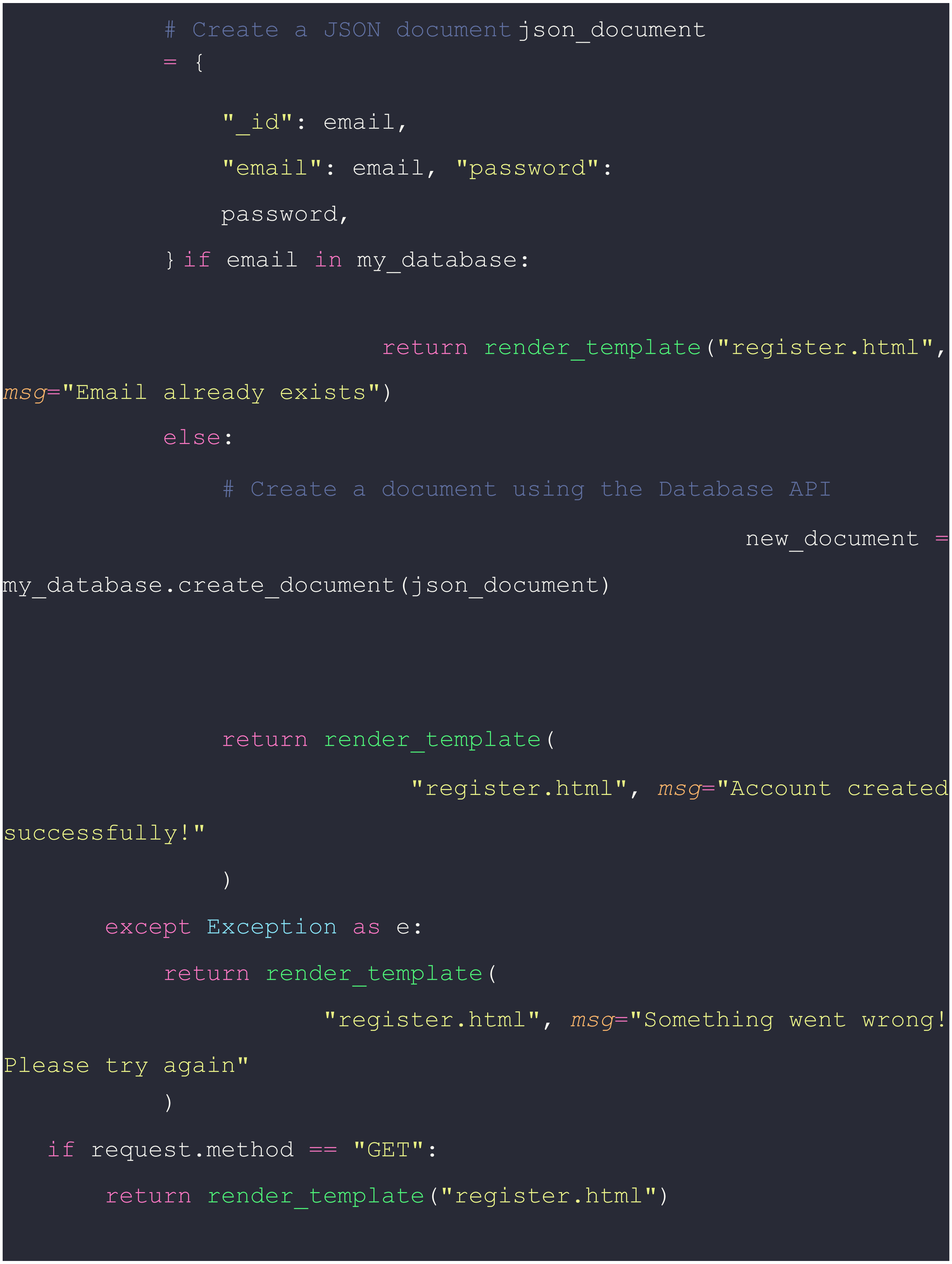
```
import os


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
```

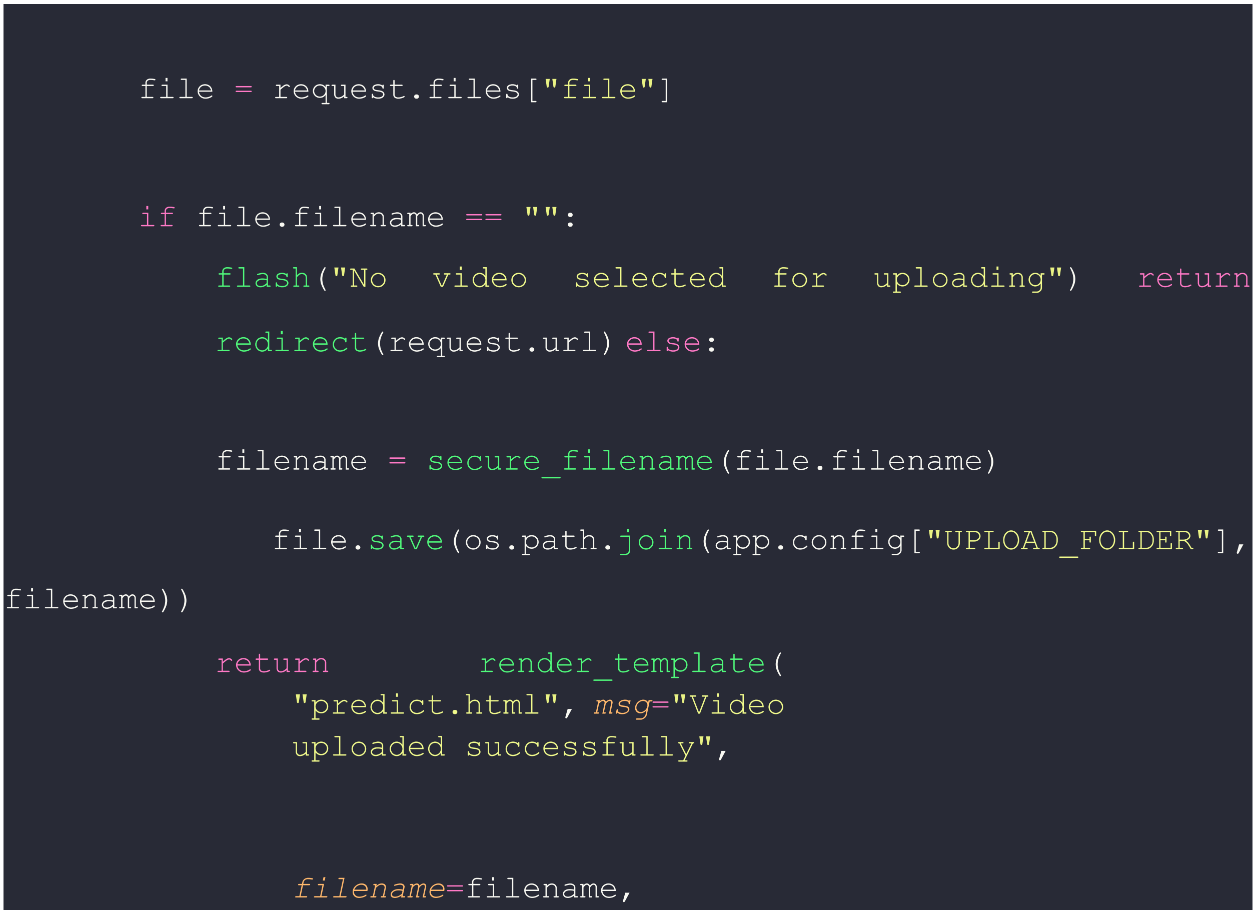


```
try:
    email = request.form["email"]
    password = request.form["password"]

    # 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.")
```

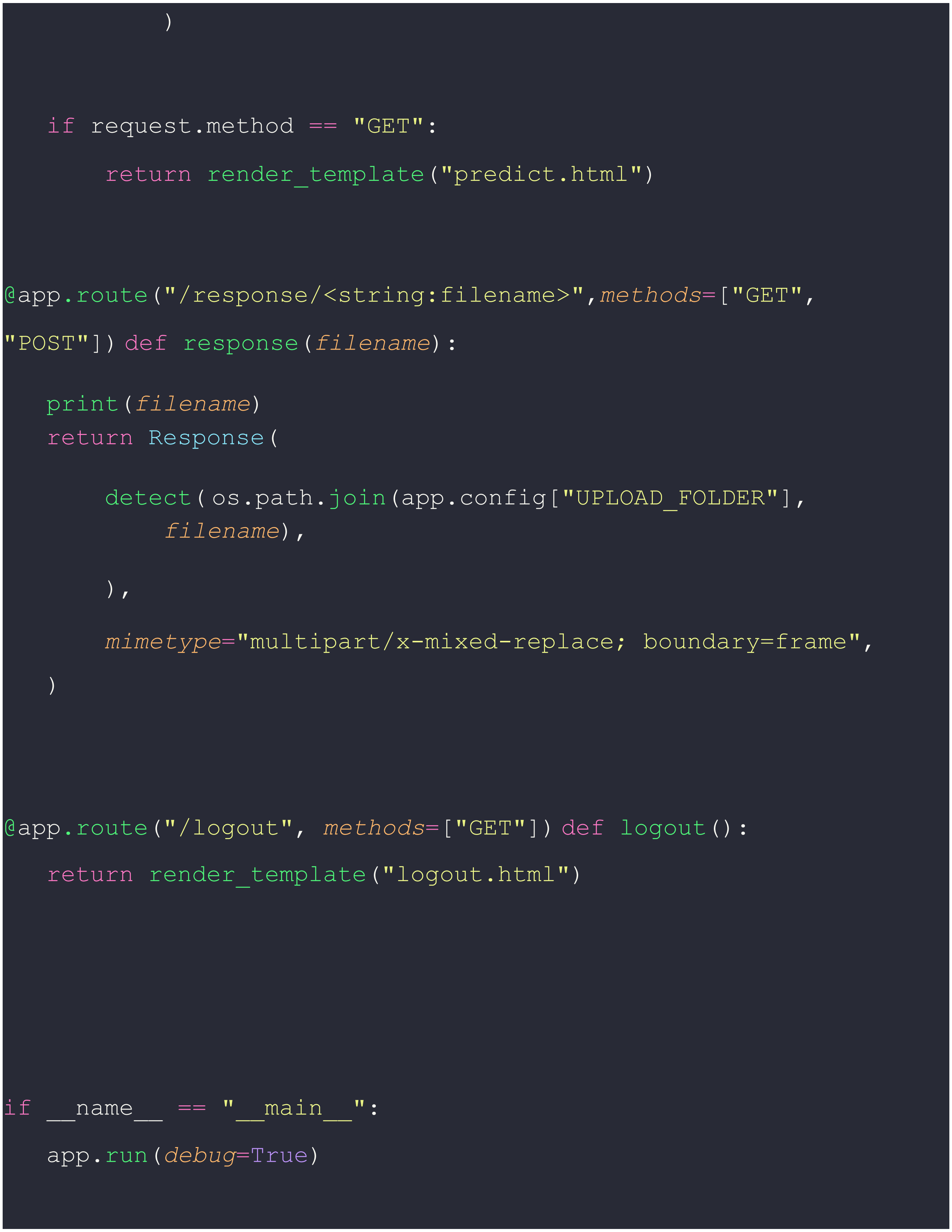


```
           # Create a JSON document json_document
           = {

               "_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,
```

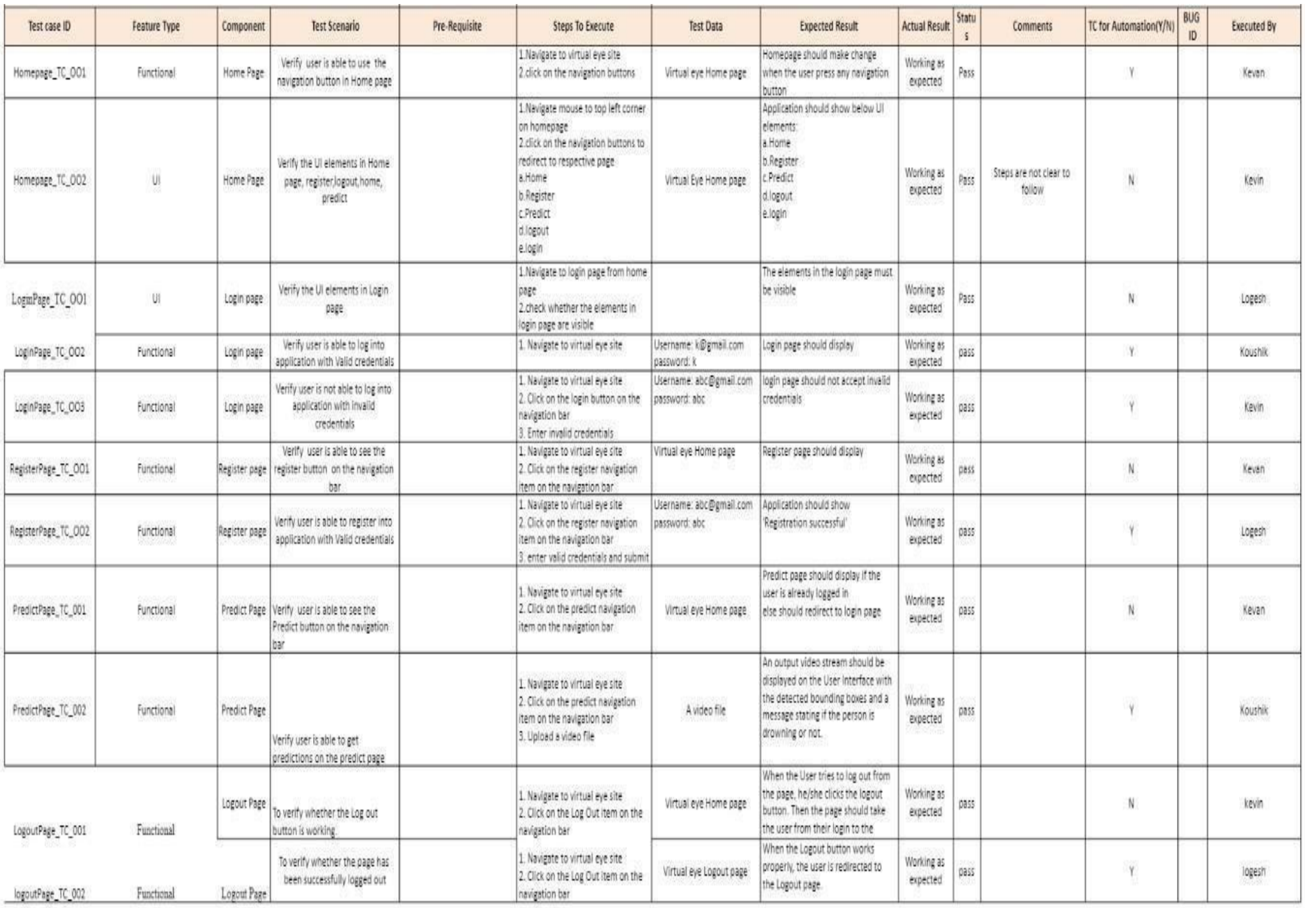


```
          )


   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)
```

CHAPTER 8

TESTING

8.1 Test Cases

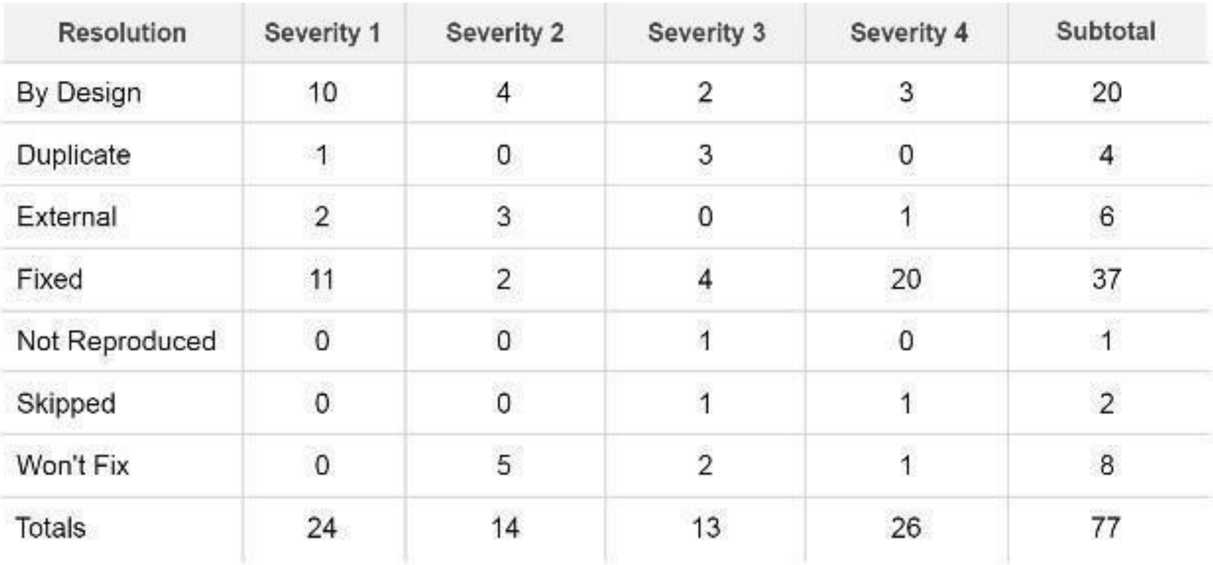


Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result	Status	Comments	TC for Automation(Y/N)	BUG ID	Executed By
Homepage_TC_OO1	Functional	Home Page	Verify user is able to use the navigation button in Home page		1.Navigate to virtual eye site 2.click on the navigation buttons	Virtual eye Home page	Homepage should make change when the user press any navigation button	Working as expected	Pass		Y		Kevan
Homepage_TC_OO2	UI	Home Page	Verify the UI elements in Home page, register,logout,home, predict		1.Navigate mouse to top left corner on homepage 2.click on the navigation buttons to redirect to respective page a.Home b.Register c.Predict d.logout e.login	Virtual Eye Home page	Application should show below UI elements: a.Home b.Register c.Predict d.logout e.login	Working as expected	Pass	Steps are not clear to follow	N		Kevin
LoginPage_TC_OO1	UI	Login page	Verify the UI elements in Login page		1.Navigate to login page from home page 2.check whether the elements in login page are visible		The elements in the login page must be visible	Working as expected	Pass		N		Logesh
LoginPage_TC_OO2	Functional	Login page	Verify user is able to log into application with Valid credentials		1. Navigate to virtual eye site	Username: k@gmail.com password: k	Login page should display	Working as expected	pass		Y		Koushik
LoginPage_TC_OO3	Functional	Login page	Verify user is not able to log into application with invalid credentials		1. Navigate to virtual eye site 2. Click on the login button on the navigation bar 3. Enter invalid credentials	Username: abc@gmail.com password: abc	login page should not accept invalid credentials	Working as expected	pass		Y		Kevin
RegisterPage_TC_OO1	Functional	Register page	Verify user is able to see the register button on the navigation bar		1. Navigate to virtual eye site 2. Click on the register navigation item on the navigation bar	Virtual eye Home page	Register page should display	Working as expected	pass		N		Kevan
RegisterPage_TC_OO2	Functional	Register page	Verify user is able to register into application with Valid credentials		1. Navigate to virtual eye site 2. Click on the register navigation item on the navigation bar 3. enter valid credentials and submit	Username: abc@gmail.com password: abc	Application should show 'Registration successful'	Working as expected	pass		Y		Logesh
PredictPage_TC_001	Functional	Predict Page	Verify user is able to see the Predict button on the navigation bar		1. Navigate to virtual eye site 2. Click on the predict navigation item on the navigation bar	Virtual eye Home page	Predict page should display if the user is already logged in else should redirect to login page	Working as expected	pass		N		Kevan
PredictPage_TC_002	Functional	Predict Page	Verify user is able to get predictions on the predict page		1. Navigate to virtual eye site 2. Click on the predict navigation item on the navigation bar 3. Upload a video file	A video file	An output video stream should be displayed on the User Interface with the detected bounding boxes and a message stating if the person is drowning or not.	Working as expected	pass		Y		Koushik
LogoutPage_TC_001	Functional	Logout Page	To verify whether the Log out button is working.		1. Navigate to virtual eye site 2. Click on the Log Out item on the navigation bar	Virtual eye Home page	When the User tries to log out from the page, he/she clicks the logout button. Then the page should take the user from their login to the	Working as expected	pass		N		kevin
logoutPage_TC_002	Functional	Logout Page	To verify whether the page has been successfully logged out		1. Navigate to virtual eye site 2. Click on the Log Out item on the navigation bar	Virtual eye Logout page	When the Logout button works properly, the user is redirected to the Logout page.	Working as expected	pass		Y		logesh

8.2 User Acceptance Testing

8.2.1 Defect Analysis

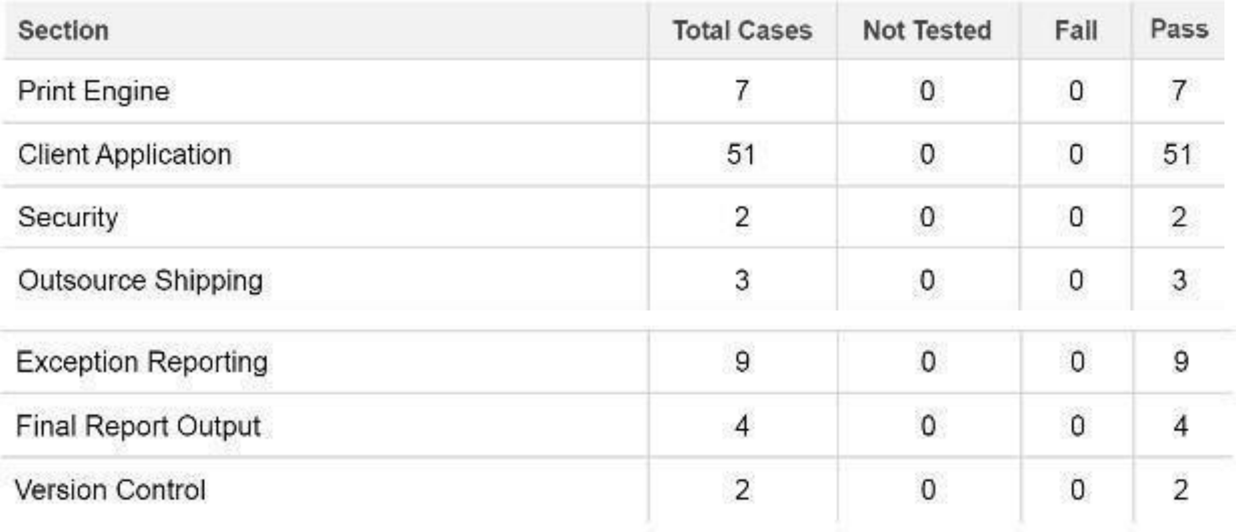
This report shows the number of resolved or closed bugs at each severity level, and how they were resolved.



Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	4	2	3	20
Duplicate	1	0	3	0	4
External	2	3	0	1	6
Fixed	11	2	4	20	37
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

2. Test Case Analysis

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

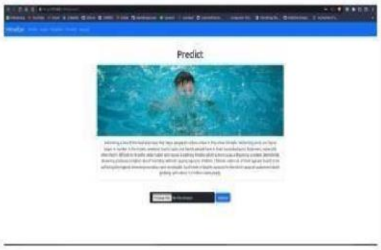
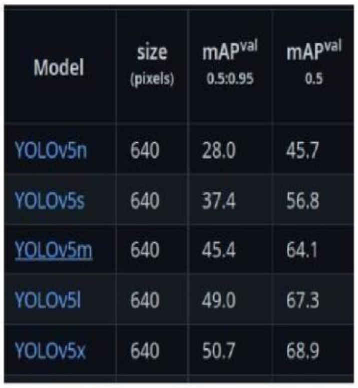
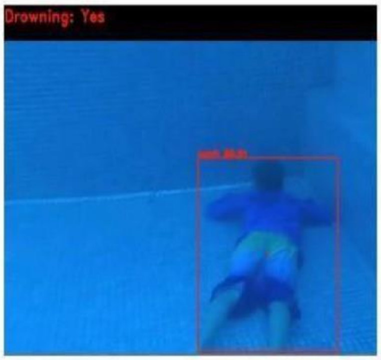


Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	51	0	0	51
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

CHAPTER 9RESULTS

9.1 PERFORMANCE METRICS

Model Performance Testing

S.No.	Parameter	Values	Screenshot
1.	Model Summary	-	
2.	Accuracy	Training Accuracy - 60 Validation Accuracy - 56	 Model | size (pixels) | mAP^{val} 0.5:0.95 | mAP^{val} 0.5 YOLOv5n | 640 | 28.0 | 45.7 YOLOv5s | 640 | 37.4 | 56.8 YOLOv5m | 640 | 45.4 | 64.1 YOLOv5l | 640 | 49.0 | 67.3 YOLOv5x | 640 | 50.7 | 68.9
3.	Confidence Score (Only Yolo Projects)	Class Detected - Person Confidence Score - 83%	

CHAPTER 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.

CHAPTER 11CONCLUSION

CONCLUSION

The results will determine the class names for a batch of frames from the videos provided as input. The action in the swimming pool will be detected as the projected class name with the highest probability. As the confidence variable, the predicted class name with the highest likelihood might be displayed. After the project is completed successfully, the video monitoring and drowning detection system can be used. If someone is discovered drowning, an alert will sound.

Drowning prevention procedures can be implemented as a result of the system's early warnings.

CHAPTER 12FUTURE SCOPE

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.

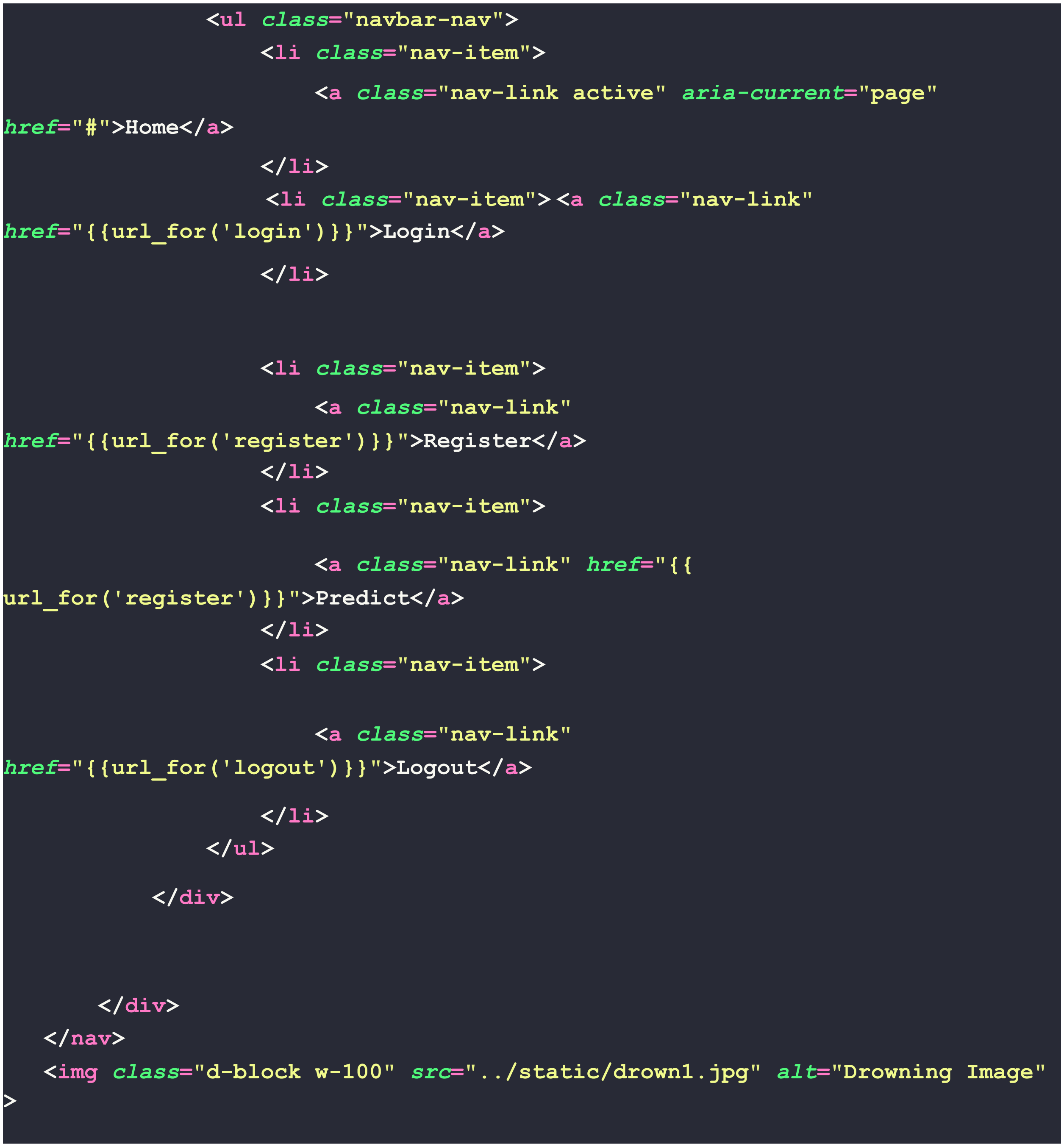
CHAPTER 13 APPENDIX

13.1 SOURCE CODE index.html

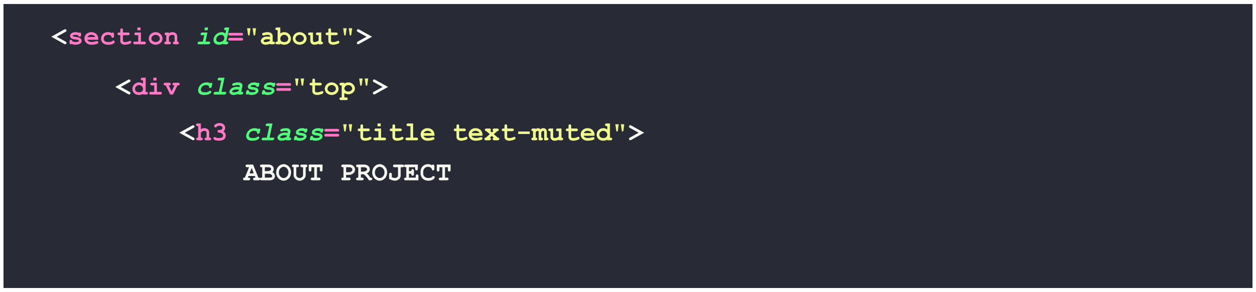
```
<!DOCTYPE html>
<html lang="en">


<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">
   <title>VirtualEye - Home</title>
   <link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"


integrity="sha384-
Gn5384xqQ1aoWXA+058RXPxPg6fy4IWvTNh0E263XmFcJlSAwiGgFAW/dAiS
6JXm" crossorigin="anonymous">
   <script src="https://code.jquery.com/jquery-3.2.1.slim.min.js"


integrity="sha384-
KJ3o2DKtIkvYIK3UENzmM7KCkRr/rE9/Qpg6aAZGJwFDMVNA/GpGFF93hXpG
5KkN"
       crossorigin="anonymous"></script> <script




src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.j
s "



integrity="sha384-
ApNbgh9B+Y1QKtv3Rn7W3mgPxhU9K/ScQsAP7hUibX39j7fakFPskvXusvfa
0b4Q"
       crossorigin="anonymous"></script>
   <script
src="https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"
```

```
integrity="sha384-
JZR6Spejh4U02d8jOt6vLEHfe/JQGiRRSQQxSfFWpi1MquVdAyjUar5+76PV
CmYl"
       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=
s
wap" rel="stylesheet">
```

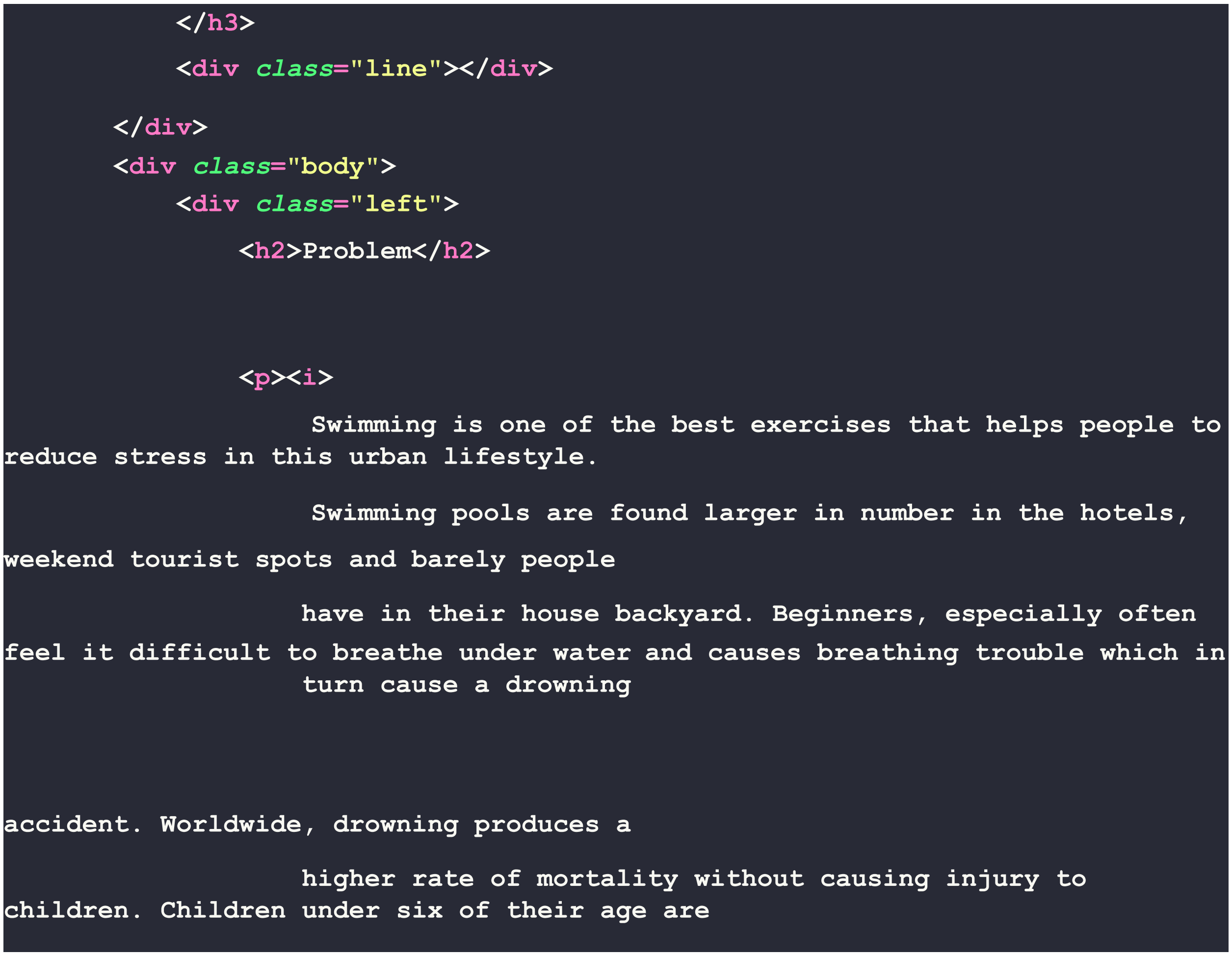
```
   <link rel="stylesheet" href="../static/style.css"></head>


<body>


   <nav class="navbar navbar-expand-lg navbar-dark bg-primary">

       <div class="container-fluid">

           <a class="navbar-brand" href="#">VirtualEye</a>
           <button class="navbar-toggler" type="button"
data-bs-toggle="collapse" data-bs-target="#navbarNav"

                aria-controls="navbarNav" aria-expanded="false"

aria-label="Toggle navigation">
               <span class="navbar-toggler-icon"></span>
           </button>

           <div class="collapse navbar-collapse" id="navbarNav">
```



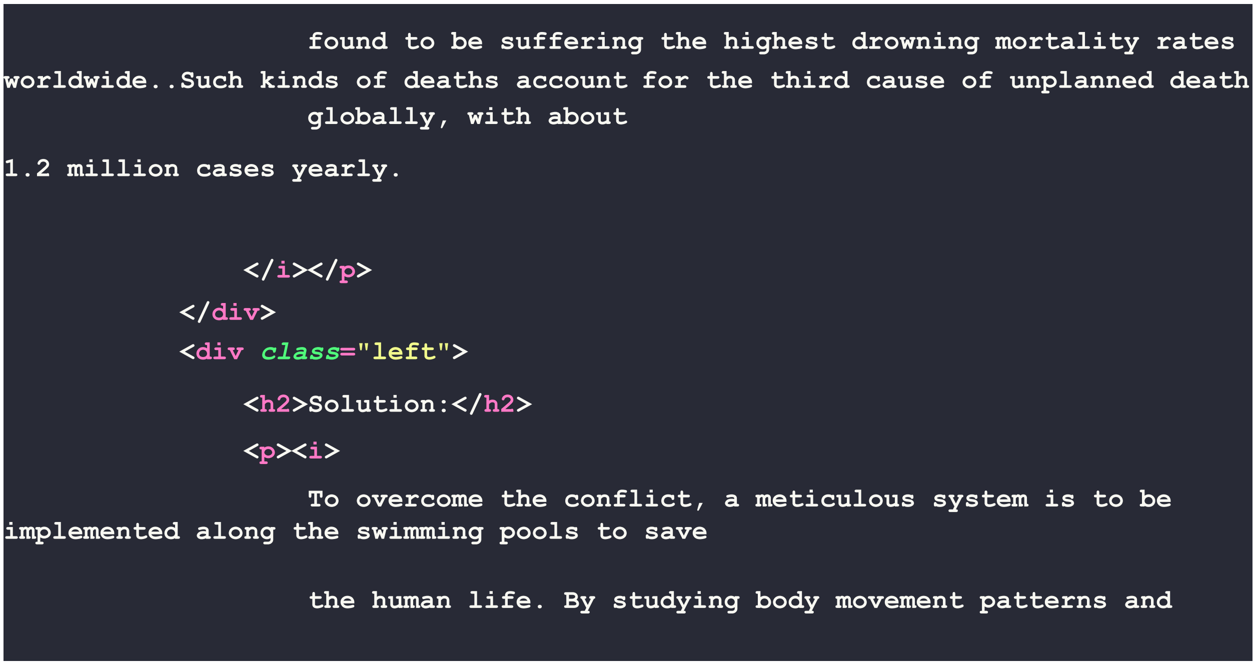
```
               <ul class="navbar-nav">
                   <li class="nav-item">

                       <a class="nav-link active" aria-current="page"
href="#">Home</a>

                   </li>
                    <li class="nav-item"><a class="nav-link"
href="{{url_for('login')}}">Login</a>

                   </li>


                   <li class="nav-item">

                       <a class="nav-link"
href="{{url_for('register')}}">Register</a>
                   </li>
                   <li class="nav-item">

                       <a class="nav-link" href="{{
url_for('register')}}">Predict</a>
                   </li>
                   <li class="nav-item">


                       <a class="nav-link"
href="{{url_for('logout')}}">Logout</a>

                   </li>
               </ul>

           </div>



       </div>
   </nav>
   <img class="d-block w-100" src="../static/drown1.jpg" alt="Drowning Image"
>
```



```
   <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><i>
                     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.


                </i></p>
            </div>
            <div class="left">
                <h2>Solution:</h2>
                <p><i>
                    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.


               </i></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 © 2021. All Rights Reserved</p> </section>
</body>


</html>

```

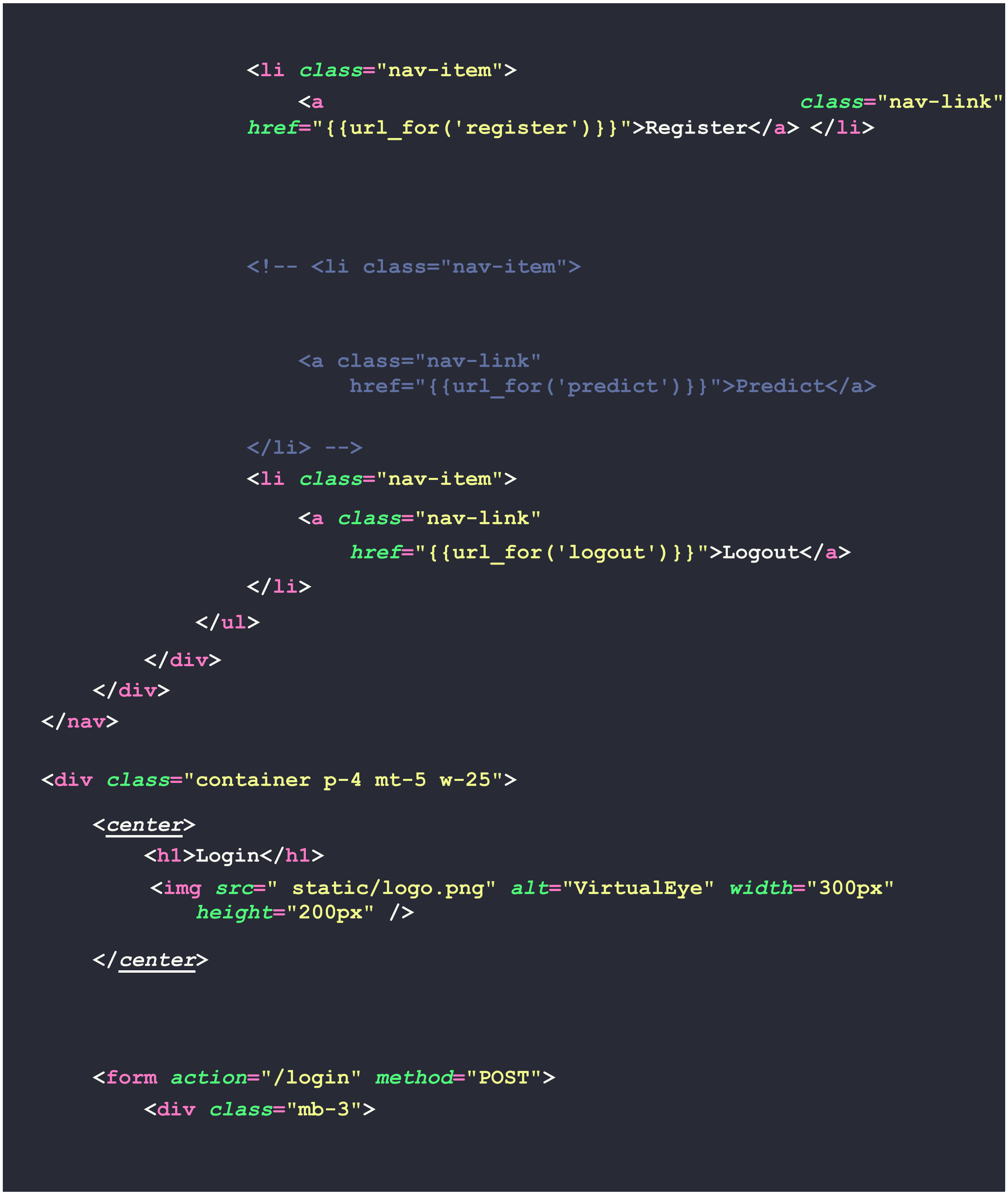
login.html

```
<!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">
   <title>VirtualEye - Login</title>
   <link


href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css
"


          rel="stylesheet"


integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1
WTRi"
          crossorigin="anonymous">
   <script


src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.mi
n

```

```
.js"
```

```
integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Q bsw3"

       crossorigin="anonymous"></script>


</head>



<body>
   <nav class="navbar navbar-expand-lg navbar-dark bg-primary">

       <div class="container-fluid">

           <a class="navbar-brand" href="#">VirtualEye</a>

           <button class="navbar-toggler" type="button"
               databstoggle="collapse" data-bs-target="#navbarNav"
               ariacontrols="navbarNav" aria-expanded="false"


               aria-label="Toggle navigation">




               <span class="navbar-toggler-icon"></span>
           </button>

           <div class="collapse navbar-collapse" id="navbarNav">

               <ul class="navbar-nav">

                   <li class="nav-item">

                       <a class="nav-link active" aria-current="page"
                           href="{{url_for('index')}}">Home</a>
```

```
                   </li>
                   <li class="nav-item">
                       <a class="nav-link active" href="#">Login</a>



                   </li>
```

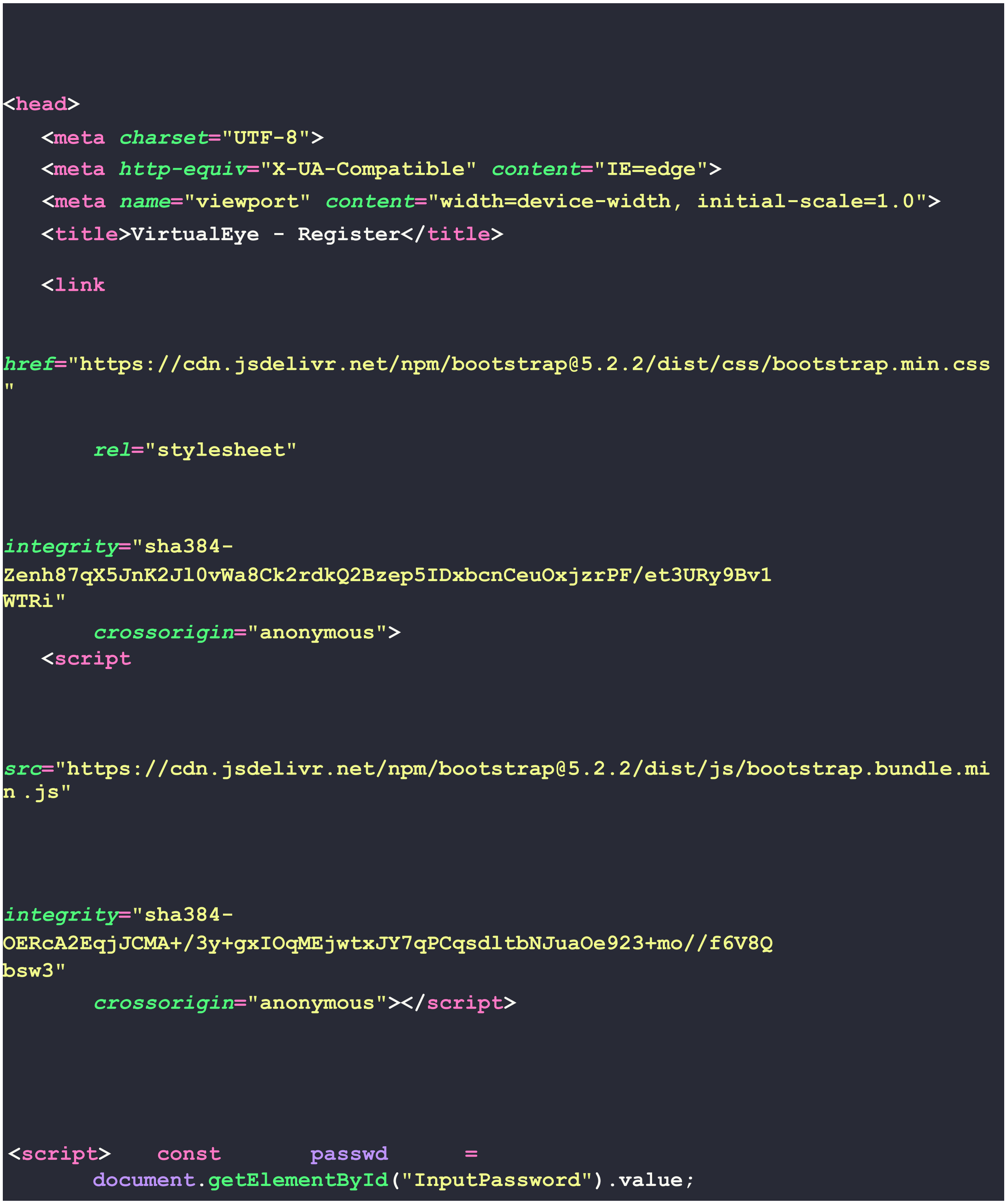


```
                <li class="nav-item">
                    <a                                 class="nav-link"
                href="{{url_for('register')}}">Register</a> </li>



                <!-- <li class="nav-item">


                    <a class="nav-link"
                        href="{{url_for('predict')}}">Predict</a>

                </li> -->
                <li class="nav-item">
                    <a class="nav-link"
                        href="{{url_for('logout')}}">Logout</a>
                </li>
            </ul>
        </div>
    </div>
</nav>

<div class="container p-4 mt-5 w-25">

    <center>
        <h1>Login</h1>
         <img src=" static/logo.png" alt="VirtualEye" width="300px"
            height="200px" />

    </center>



    <form action="/login" method="POST">
        <div class="mb-3">
```

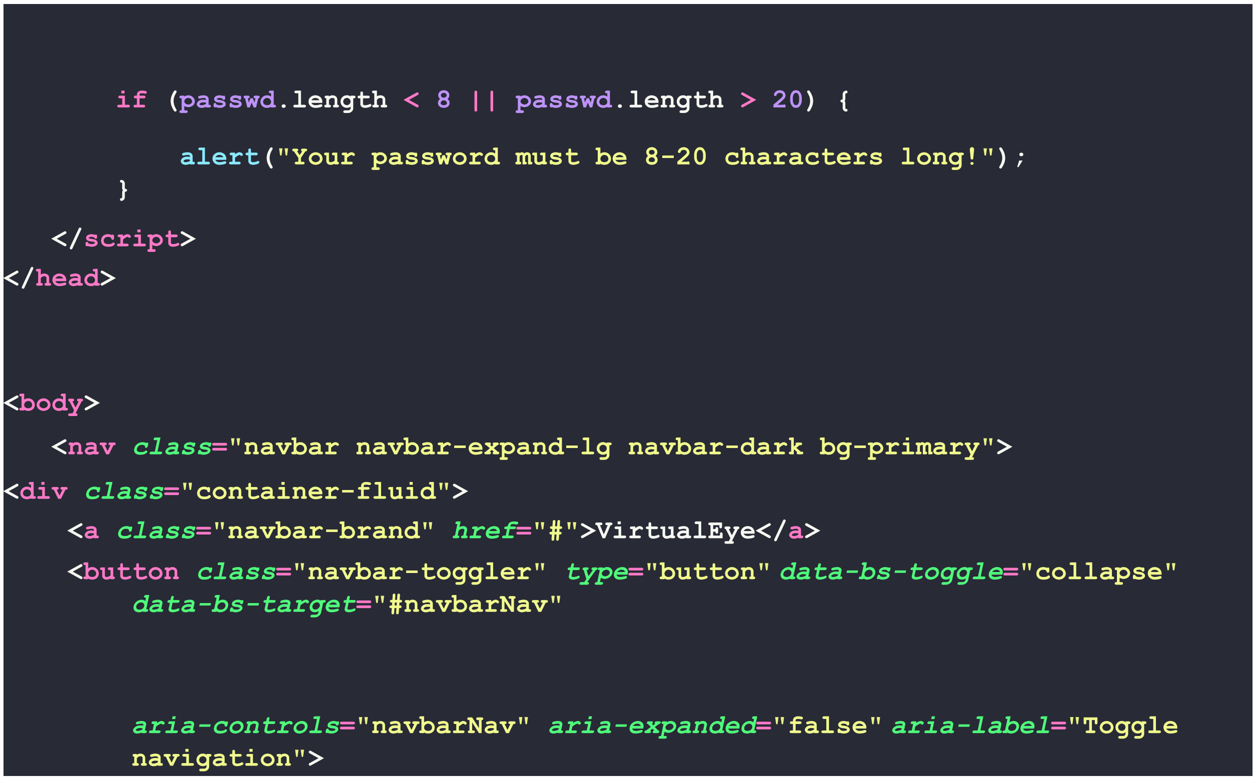
```
                 <label for="InputEmail" class="form-label">Email

address</label>
               <input name="email" type="email" class="form-control"
                   id="InputEmail" aria-describedby="emailHelp"
                   placeholder="Email">

                 <div id="emailHelp" class="form-text">We'll never share your

                   email with anyone
                   else.</div>
           </div>



           <div class="mb-3">

                <label for="InputPassword" class="form-label">Password</label>

               <input name="password" type="password" class="form-
           control" id="InputPassword" placeholder="Password"> </div>



           <button type="submit" class="btn btn-primary">Submit</button>
            <button type="reset" class="btn btn-light">Reset</button>
           <p  class="mt-3"  style="color:red;  font-size:  20px;">{{msg}}</p>
       </form>



   </div>
</body>




</html>
```

register.html

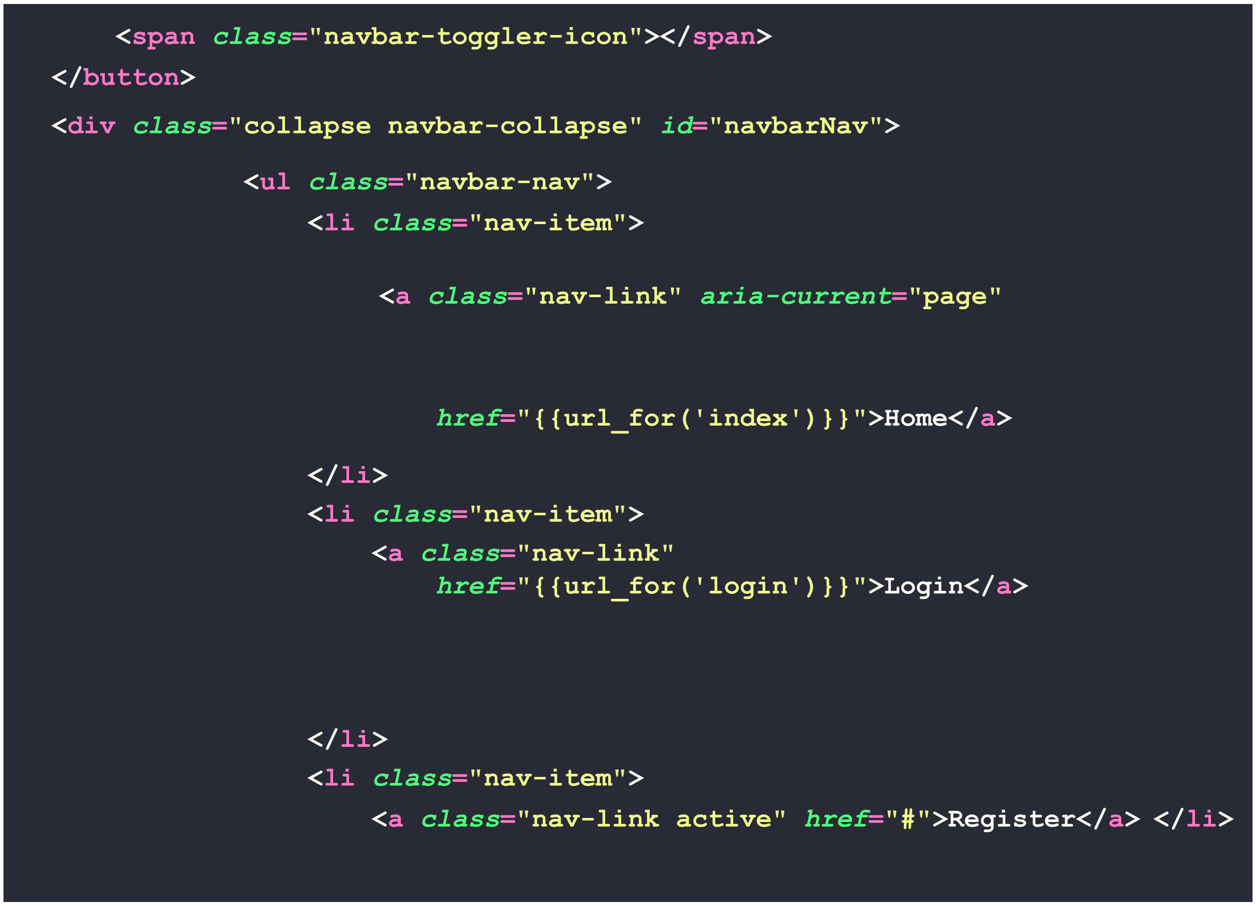
```
<!DOCTYPE html>

<html lang="en">
```



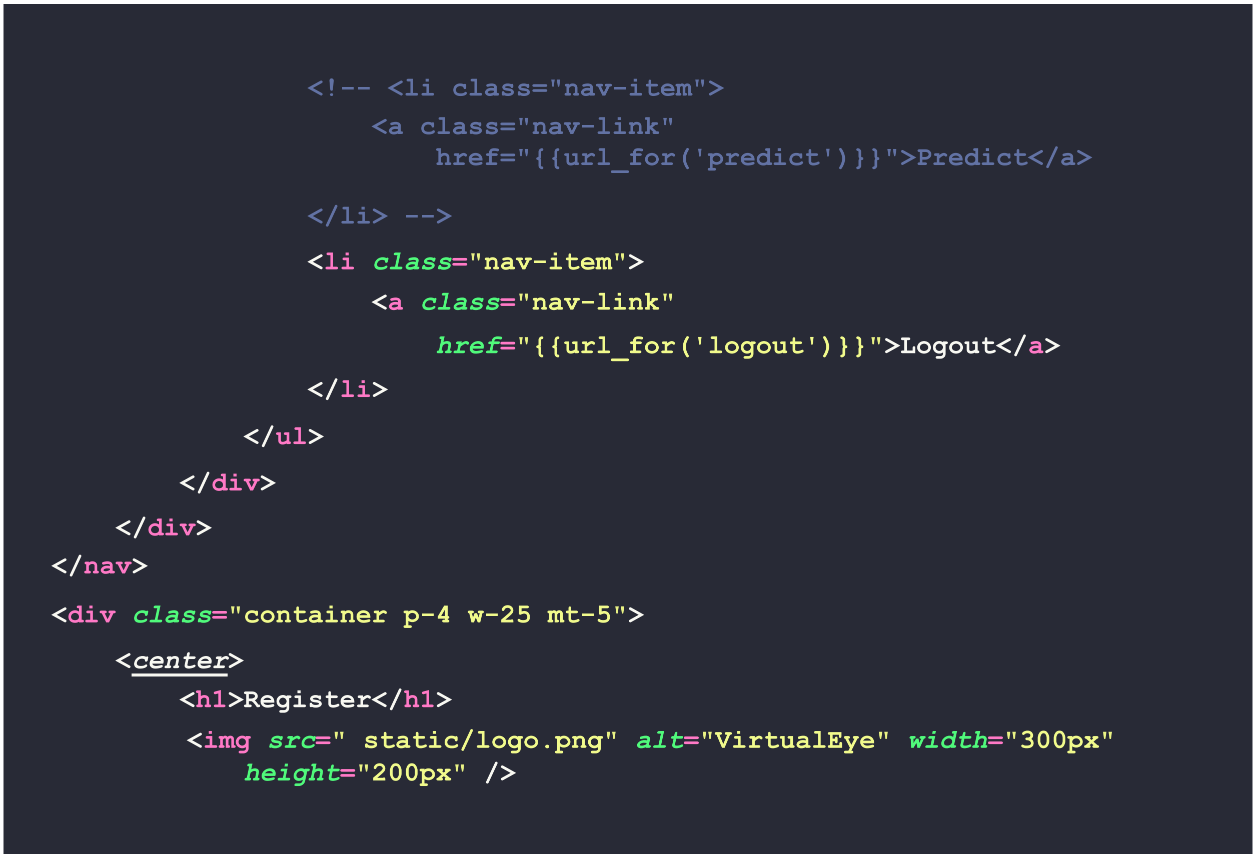
```
<head>
   <meta charset="UTF-8">
   <meta http-equiv="X-UA-Compatible" content="IE=edge">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>VirtualEye - Register</title>

   <link


href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css
"

       rel="stylesheet"


integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1
WTRi"
       crossorigin="anonymous">
   <script



src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.mi
n .js"




integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Q
bsw3"
       crossorigin="anonymous"></script>





 <script>    const       passwd      =
       document.getElementById("InputPassword").value;
```



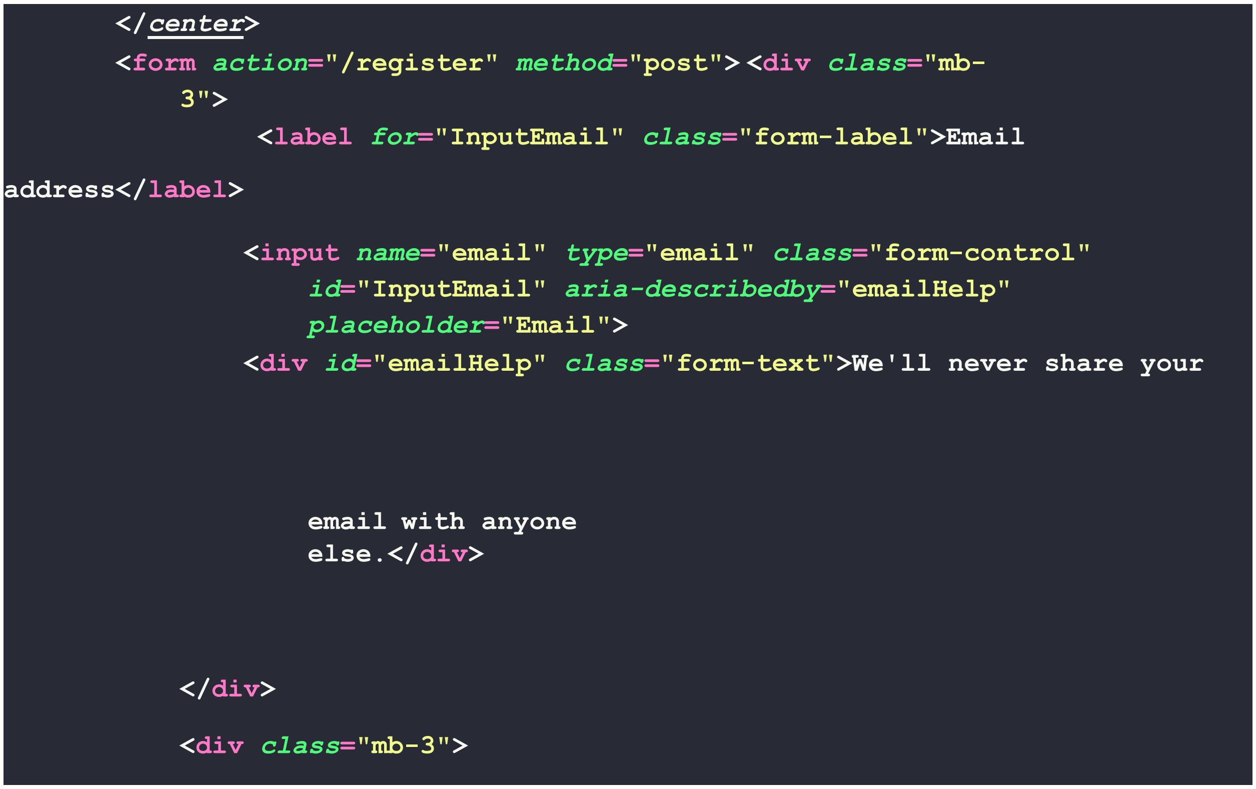
```
        if (passwd.length < 8 || passwd.length > 20) {

            alert("Your password must be 8-20 characters long!");
        }

    </script>
</head>



<body>
    <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
<div class="container-fluid">
     <a class="navbar-brand" href="#">VirtualEye</a>
     <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
         data-bs-target="#navbarNav"


         aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle
         navigation">
```



```
        <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarNav">
                <ul class="navbar-nav">
                    <li class="nav-item">

                         <a class="nav-link" aria-current="page"


                              href="{{url_for('index')}}">Home</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link"
                            href="{{url_for('login')}}">Login</a>



                    </li>
                    <li class="nav-item">
                        <a class="nav-link active" href="#">Register</a> </li>
```



```
                  <!-- <li class="nav-item">
                      <a class="nav-link"
                          href="{{url_for('predict')}}">Predict</a>

                  </li> -->
                  <li class="nav-item">
                      <a class="nav-link"
                          href="{{url_for('logout')}}">Logout</a>
                  </li>
              </ul>
          </div>
      </div>
  </nav>
  <div class="container p-4 w-25 mt-5">
      <center>
          <h1>Register</h1>
           <img src=" static/logo.png" alt="VirtualEye" width="300px"
              height="200px" />
```

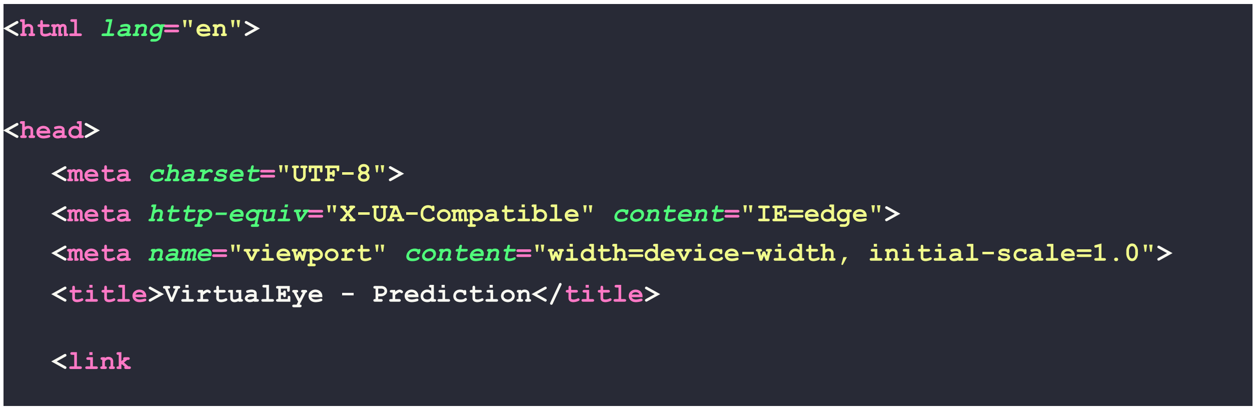


```
      </center>
      <form action="/register" method="post"><div class="mb-
          3">
               <label for="InputEmail" class="form-label">Email
address</label>

              <input name="email" type="email" class="form-control"
                  id="InputEmail" aria-describedby="emailHelp"
                  placeholder="Email">
              <div id="emailHelp" class="form-text">We'll never share your



                  email with anyone
                  else.</div>



          </div>

          <div class="mb-3">
```

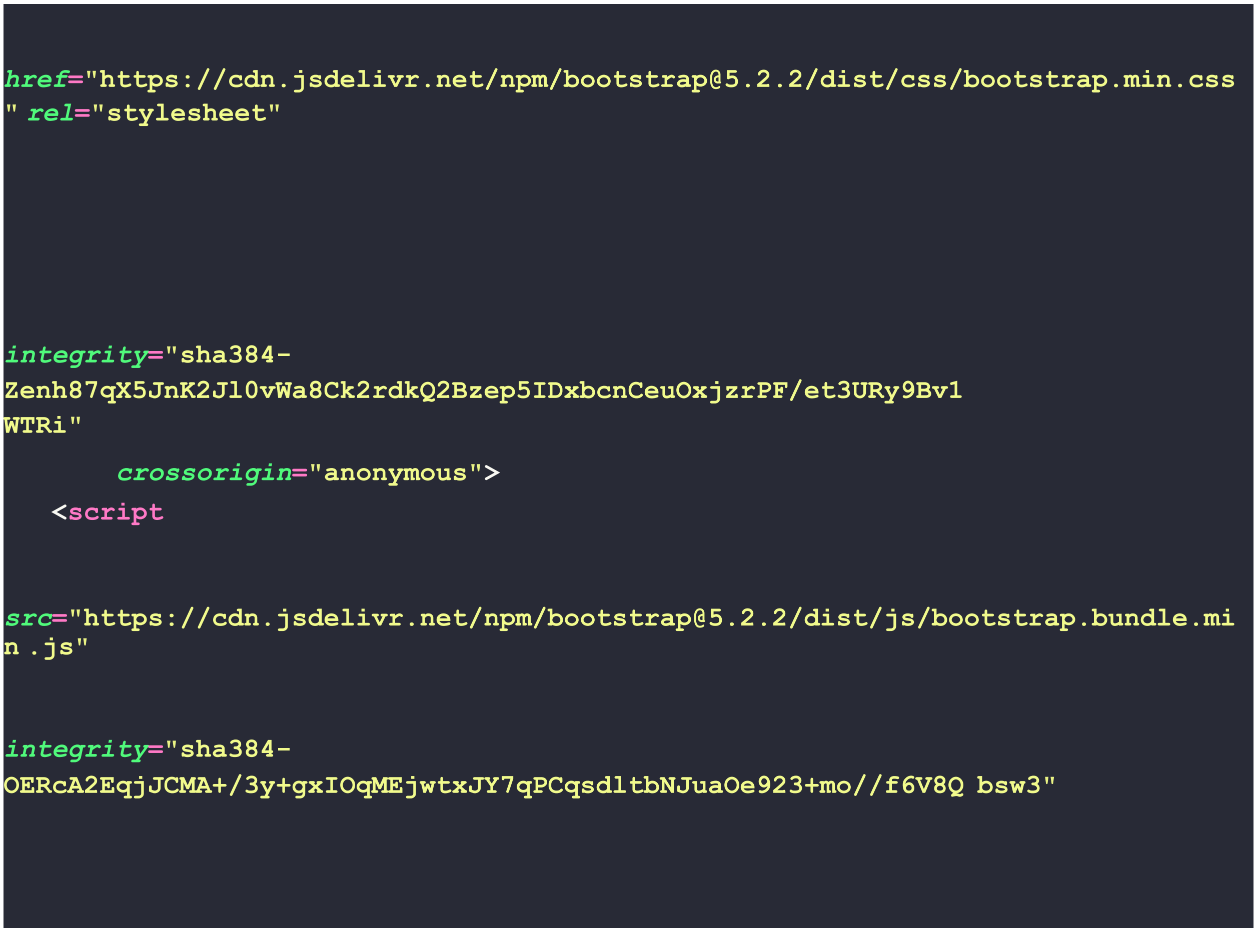
```
               <label for="InputPassword" class="form-label">Password</label>

               <input name="password" type="password" class="form-control"
                   id="InputPassword" placeholder="Password">



               <div id="passwordHelpBlock" class="form-text">
                   Your password must be 8-20 characters long, contain
                   letters and numbers, and must not contain spaces or emoji.



               </div>
           </div>
           <button type="submit" class="btn btn-primary">Submit</button>

           <button type="reset" class="btn btn-light">Reset</button>

           <br><br>

           <div id="passwordHelpBlock" class="form-text">

               Already a registered user?

           </div>

           <p class="text-primary" style="font-weight: bold; font-size:
18px;">

               {{msg}}</p>
              <a href="{{url_for('login')}}" class="btn btn-success">Login</a>
       </form>

   </div>


</body>


</html>
```

predict.html

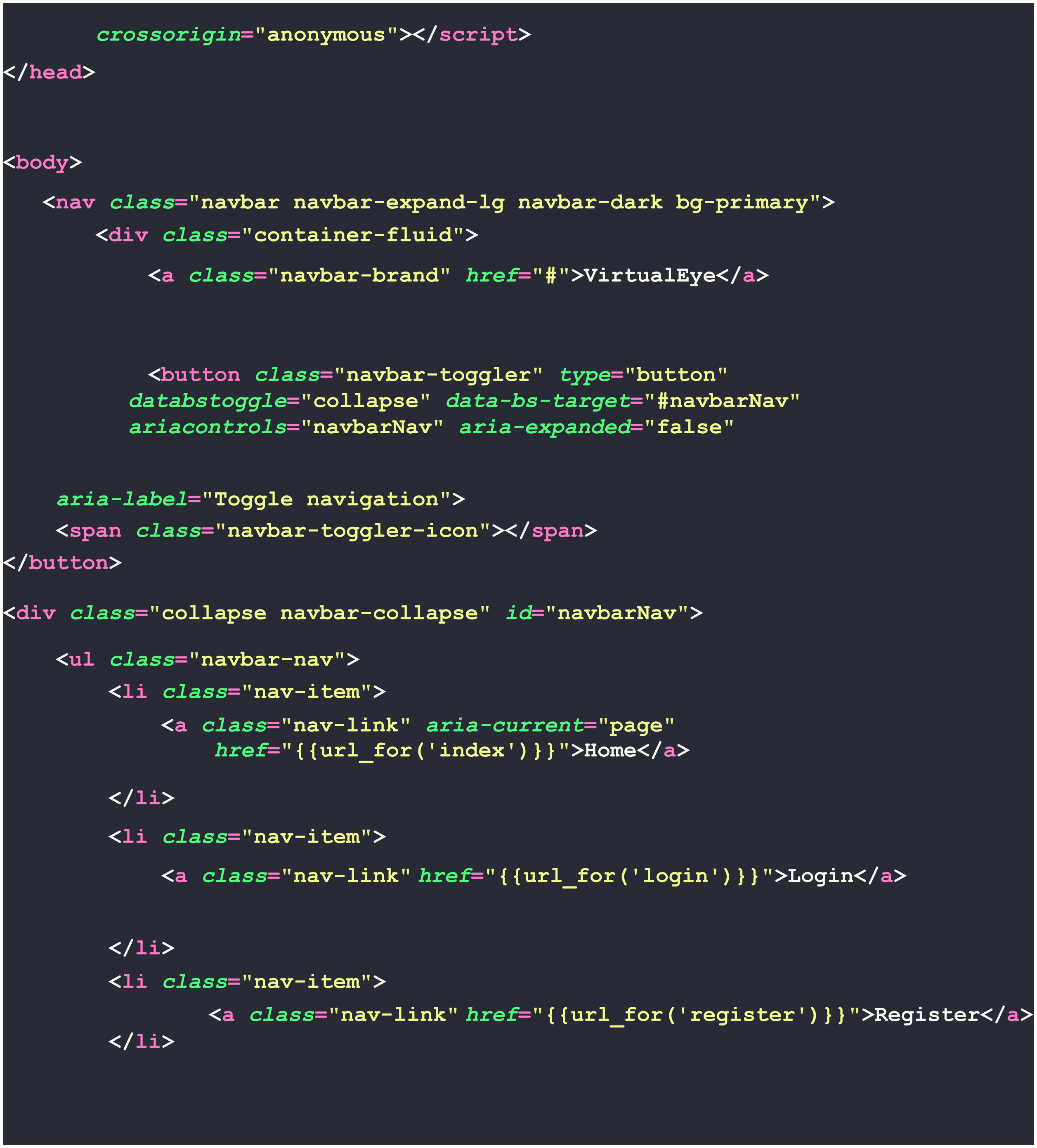
```
<!DOCTYPE html>
```



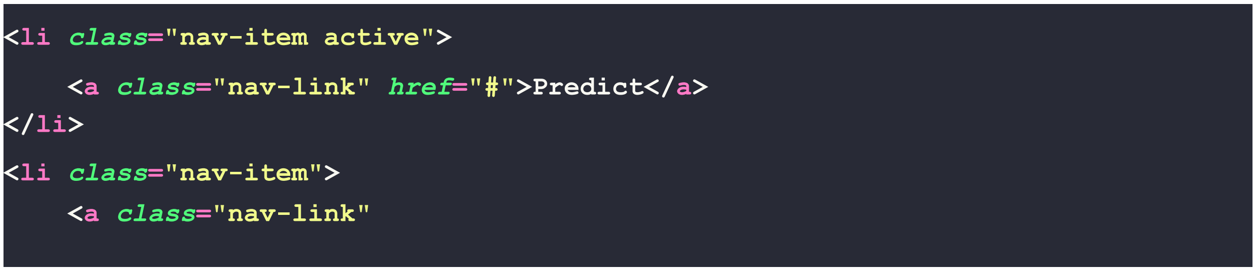
```
<html lang="en">


<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">
   <title>VirtualEye - Prediction</title>

   <link
```



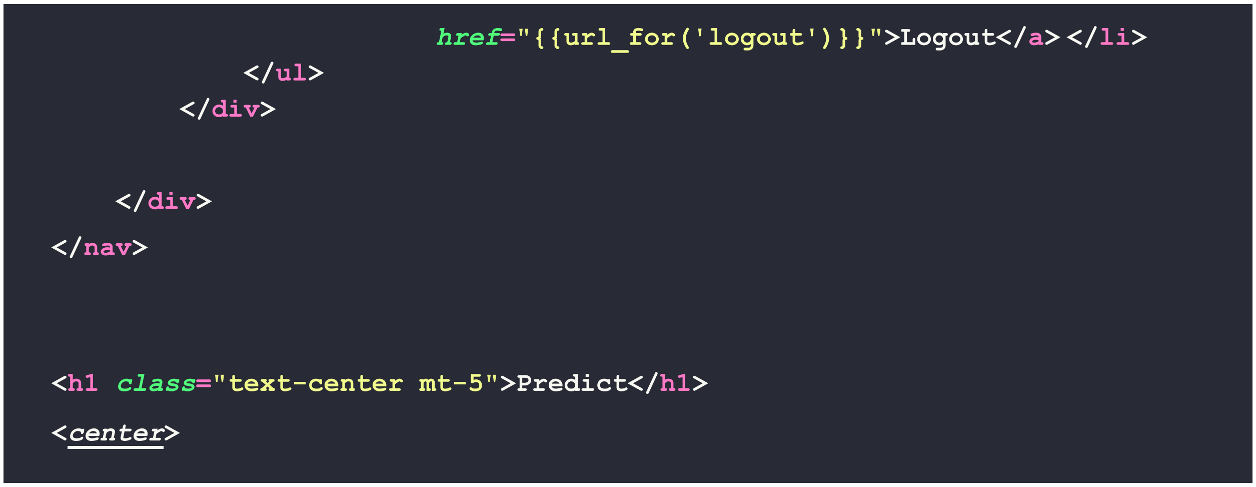
```
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css
" rel="stylesheet"






integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1
WTRi"
       crossorigin="anonymous">
   <script


src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.mi
n .js"


integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Q bsw3"
```



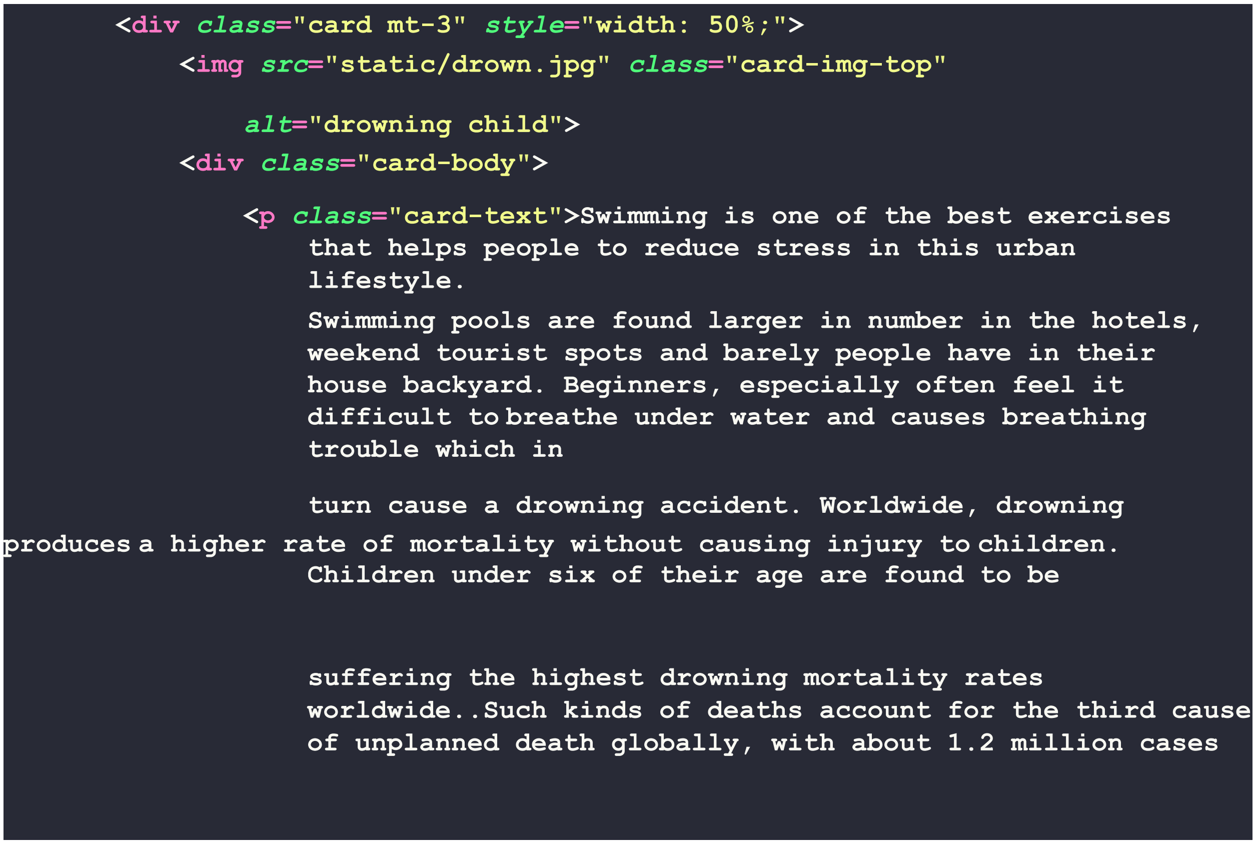
```
       crossorigin="anonymous"></script>
</head>


<body>
   <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
       <div class="container-fluid">
           <a class="navbar-brand" href="#">VirtualEye</a>


           <button class="navbar-toggler" type="button"
          databstoggle="collapse" data-bs-target="#navbarNav"
          ariacontrols="navbarNav" aria-expanded="false"

    aria-label="Toggle navigation">
    <span class="navbar-toggler-icon"></span>
</button>

<div class="collapse navbar-collapse" id="navbarNav">

    <ul class="navbar-nav">
        <li class="nav-item">
            <a class="nav-link" aria-current="page"
                href="{{url_for('index')}}">Home</a>

        </li>
        <li class="nav-item">

            <a class="nav-link" href="{{url_for('login')}}">Login</a>


        </li>
        <li class="nav-item">
                <a class="nav-link" href="{{url_for('register')}}">Register</a>
        </li>
```



```
<li class="nav-item active">

    <a class="nav-link" href="#">Predict</a>
</li>
<li class="nav-item">
    <a class="nav-link"
```



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


      </div>
  </nav>



  <h1 class="text-center mt-5">Predict</h1>
  <center>
```



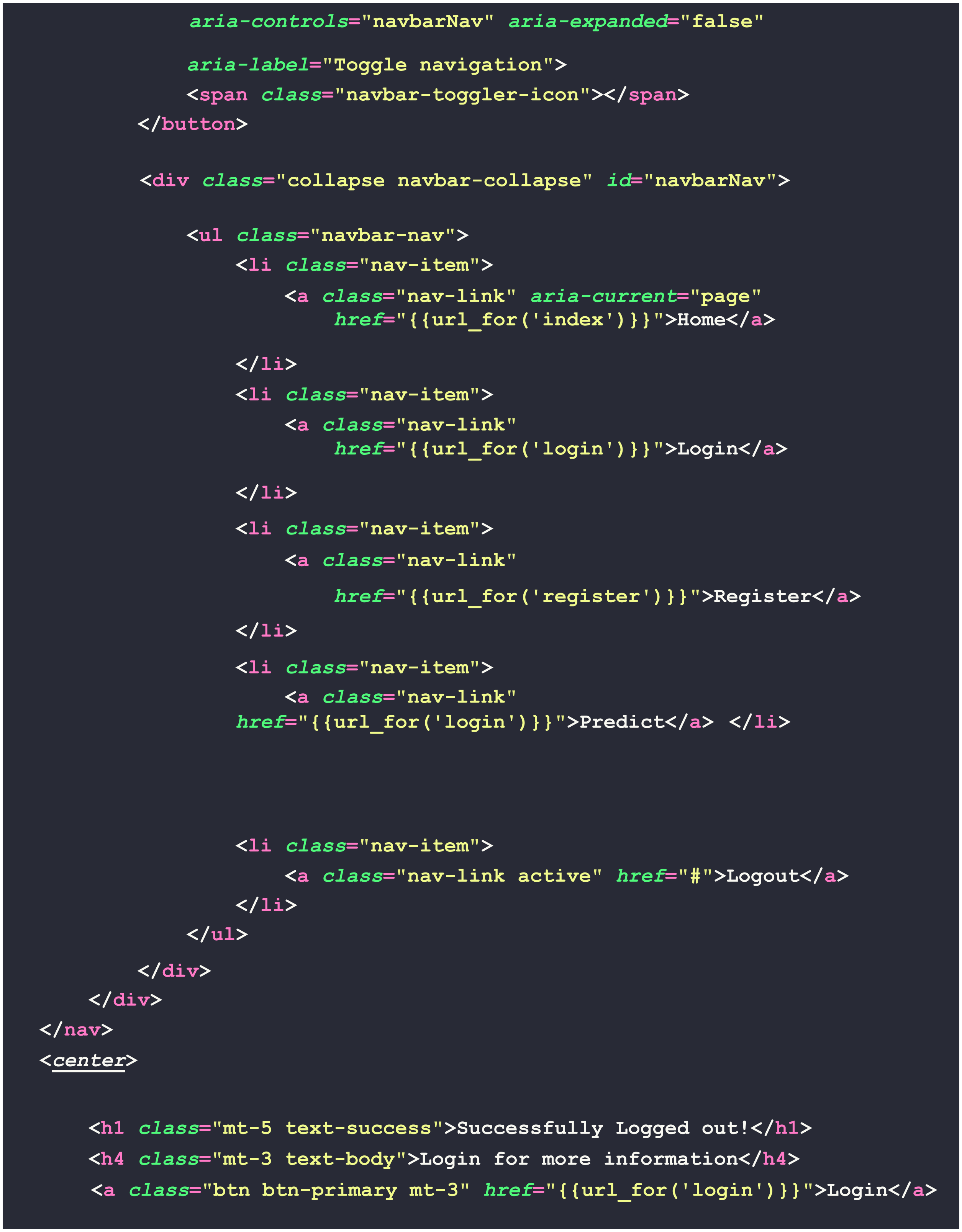
```
      <div class="card mt-3" style="width: 50%;">
          <img src="static/drown.jpg" class="card-img-top"

              alt="drowning child">
          <div class="card-body">
              <p class="card-text">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>
    </center>
    <form method="post" action="/predict" enctype="multipart/form-data">
        <div class="container text-center mt-5">
            <input class="btn btn-dark" type="file" name="file"
                autocomplete="off" required>
            <input class="btn btn-primary" type="submit" value="Upload">
        </div>
    </form>
    <center>

        {% if filename %}
        <div class="container m-5">
            <img src="{{ url_for('response', filename=filename) }}" width="50%"
                height="400px" />
        </div>
        {% endif %}
    </center>
</body>



</html>
```

logout.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">
   <title>VirtualEye - Logout</title>


   <link


href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/css/bootstrap.min.css
"
       rel="stylesheet"


integrity="sha384-
Zenh87qX5JnK2Jl0vWa8Ck2rdkQ2Bzep5IDxbcnCeuOxjzrPF/et3URy9Bv1
WTRi"
       crossorigin="anonymous">

   <script


src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.2/dist/js/bootstrap.bundle.mi
n



.js"


integrity="sha384-
OERcA2EqjJCMA+/3y+gxIOqMEjwtxJY7qPCqsdltbNJuaOe923+mo//f6V8Q bsw3"
       crossorigin="anonymous"></script>
</head>
```

```
<body>

   <nav class="navbar navbar-expand-lg navbar-dark bg-primary">
       <div class="container-fluid">
           <a class="navbar-brand" href="#">VirtualEye</a>
           <button class="navbar-toggler" type="button"
               databstoggle="collapse" data-bs-target="#navbarNav"
```



```
            aria-controls="navbarNav" aria-expanded="false"
            aria-label="Toggle navigation">
            <span class="navbar-toggler-icon"></span>
        </button>

        <div class="collapse navbar-collapse" id="navbarNav">

            <ul class="navbar-nav">
                <li class="nav-item">
                    <a class="nav-link" aria-current="page"
                        href="{{url_for('index')}}">Home</a>

                </li>
                <li class="nav-item">
                    <a class="nav-link"
                        href="{{url_for('login')}}">Login</a>

                </li>

                <li class="nav-item">
                    <a class="nav-link"

                        href="{{url_for('register')}}">Register</a>
                </li>

                <li class="nav-item">
                    <a class="nav-link"
                href="{{url_for('login')}}">Predict</a> </li>




                <li class="nav-item">
                    <a class="nav-link active" href="#">Logout</a>
                </li>
            </ul>

        </div>
    </div>
</nav>
<center>


    <h1 class="mt-5 text-success">Successfully Logged out!</h1>
    <h4 class="mt-3 text-body">Login for more information</h4>
    <a class="btn btn-primary mt-3" href="{{url_for('login')}}">Login</a>
```

```
   </center>
</body>


</html>
```

13.2 GitHub & Project Demo Link

GitHub link: https://github.com/IBM-EPBL/IBM-Project-54843-1662538756.git

Project demo link: https://youtu.be/uiwF3lQH4OE