

**WINTER SEMESTER – 2022-2023**

**WEB TECHNOLOGIES(SWE1008) – LAB**

**FACULTY :**MAREESWARI V

**SLOT :**L29+L30

**TEAM MEMBERS:**

YOGENDIRAN D (21MIS0055)

PRAVEEN KUMAR P(21MIS0071)

NITHYA SRI S(21MIS0090)

# DESCRIPTION:

Home automation is the use of technology to control various systems and devices in a home, including lighting, security, heating and cooling, entertainment systems, and more. One example of home automation is the use of facial recognition technology to automatically turn lights on and off.

Facial recognition technology uses a camera to capture an image of a person's face and then uses software to analyze the features of the face and match it to a pre-existing database of faces. Once a match is found, the system can trigger a pre-programmed action, such as turning on a light.

To set up this kind of automation in your home, you would need to install a camera with facial recognition software and connect it to your home automation system. You would also need to program the system to recognize the faces of the people who live in your home and to associate each face with a specific action, such as turning on a light when they enter a room.

Once the system is set up and programmed, it can operate automatically, detecting when a person enters a room and turning on the lights for them. This can be particularly useful in areas such as hallways and bathrooms, where people often need to navigate in the dark.

Overall, home automation can help to make your home more convenient and efficient, while also enhancing security and safety. By using facial recognition technology to automatically control your lighting, you can enjoy a more seamless and effortless living experience.

TECHNOLOGIES USED:

* HTML
* CSS
* JAVASCRIPT
* PHP
* MYSQL
* PYTHON

-OPENCV

-FLASK

FEATURES:

* Login and signup page.

-Form Validation

-Data Insertion

-Data retrieval

-Credential validation

* Room occupancy detection.

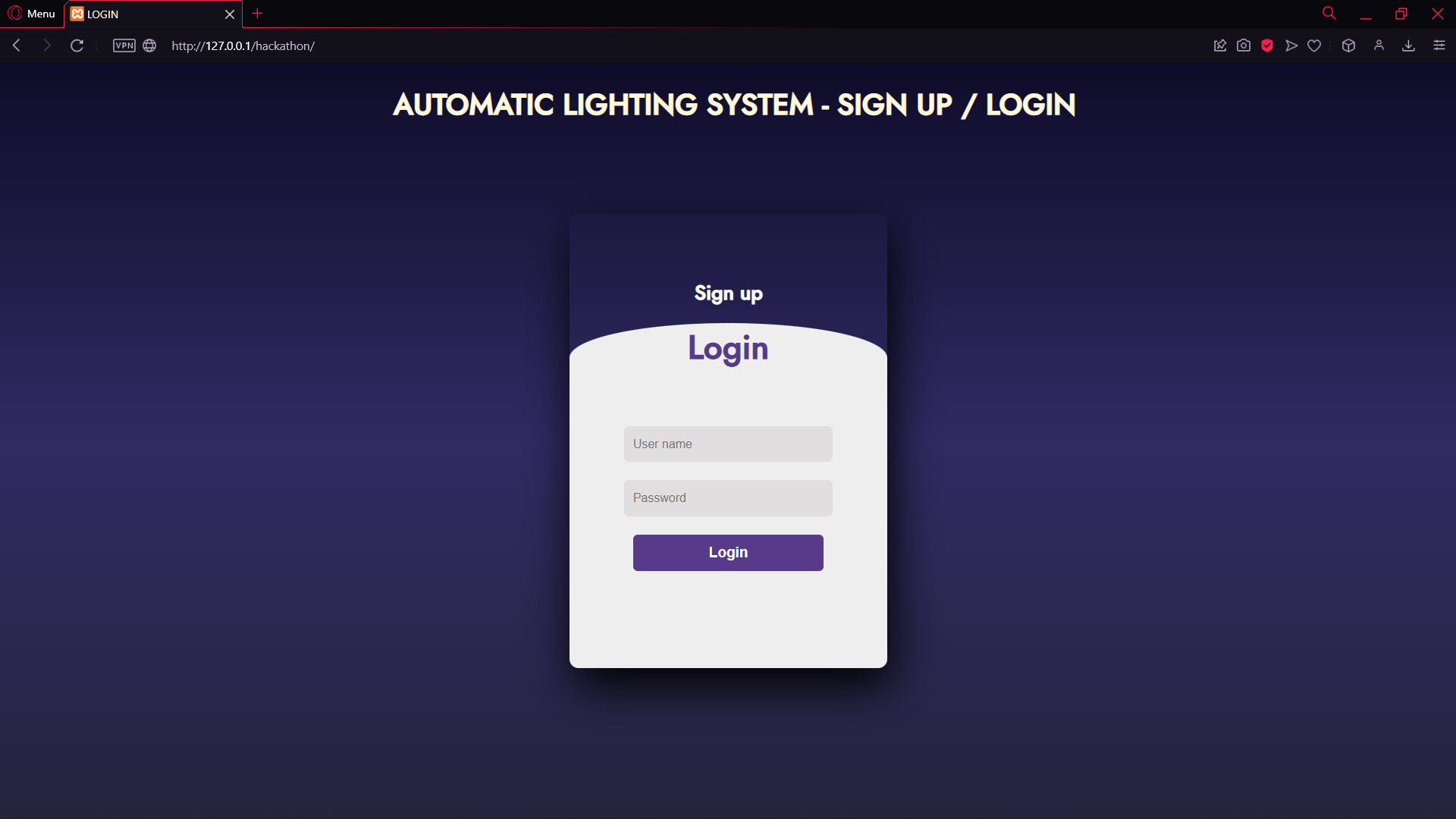
-Face detection

-People detection

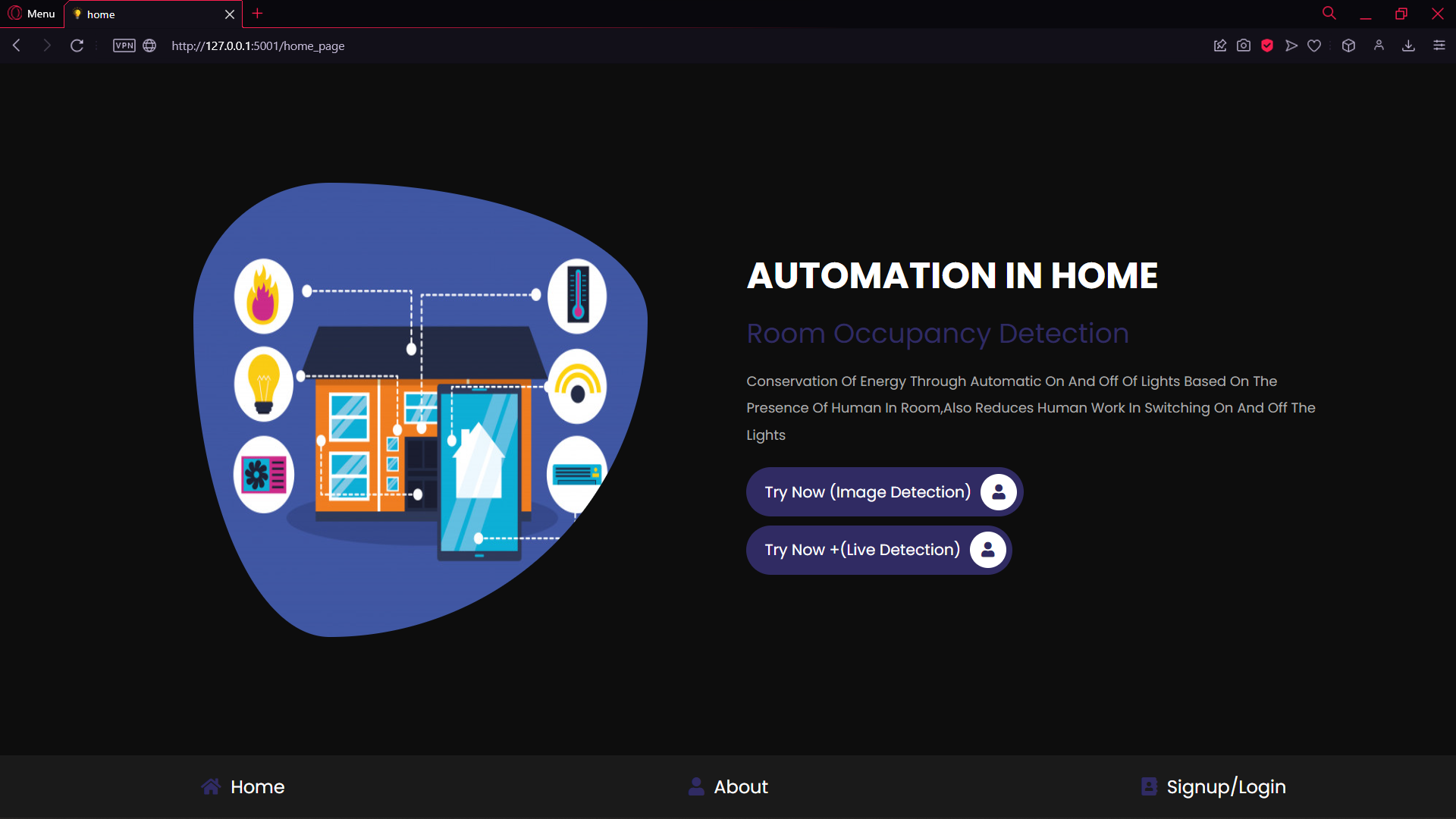
-Control room lights

**Output Screenshots:**

Signup & Login Page:



Home Page:



About Page:

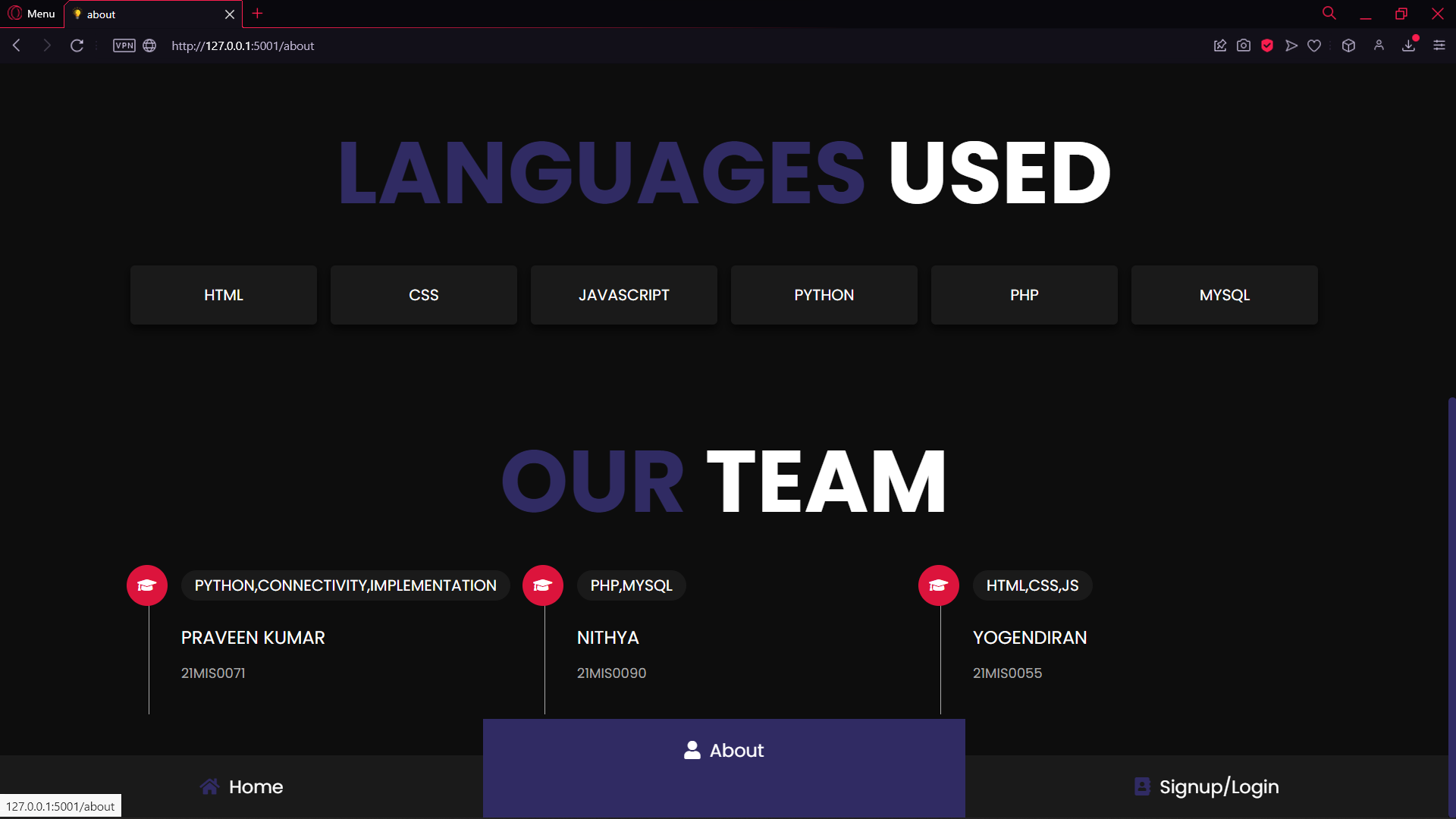
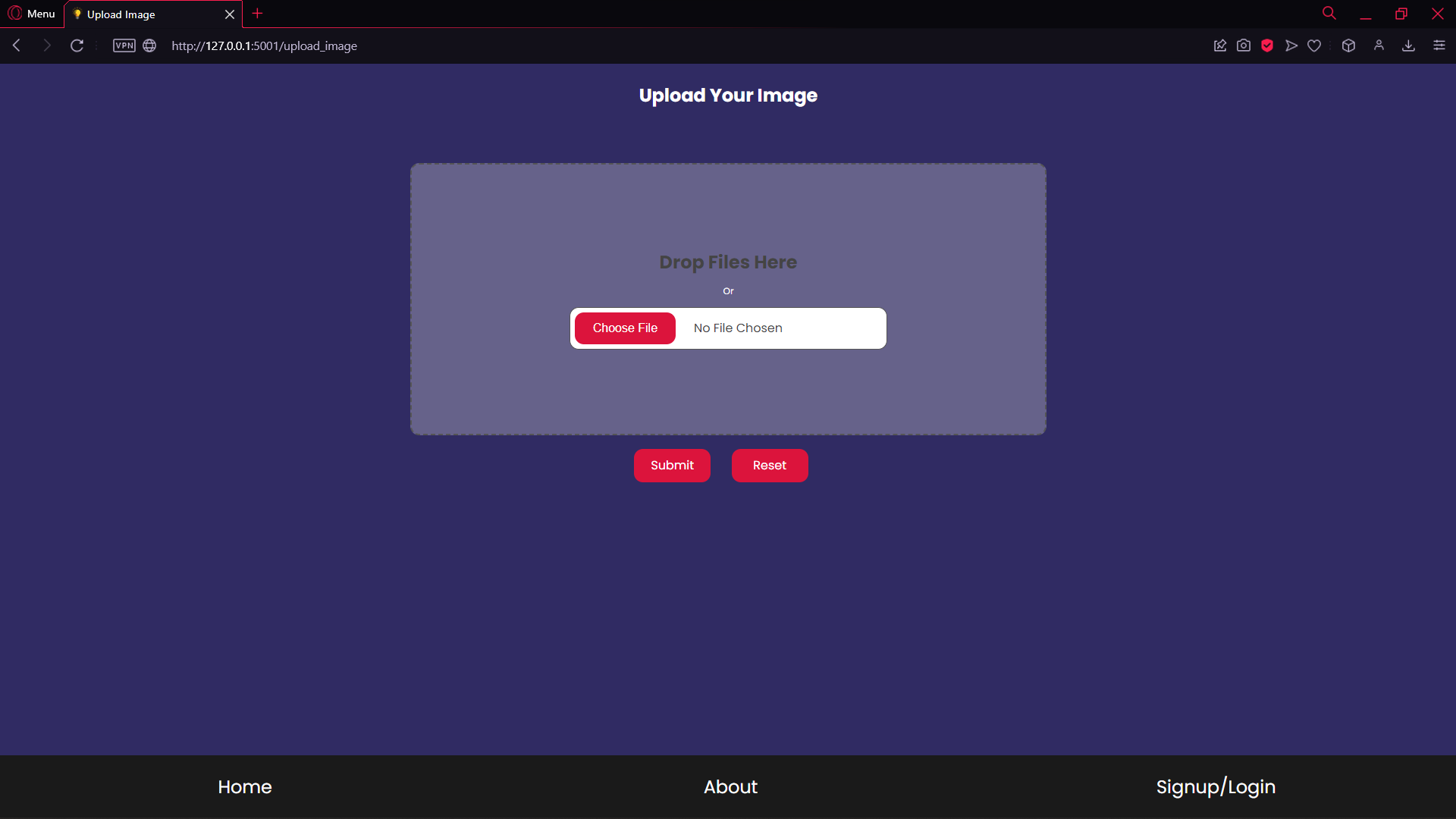
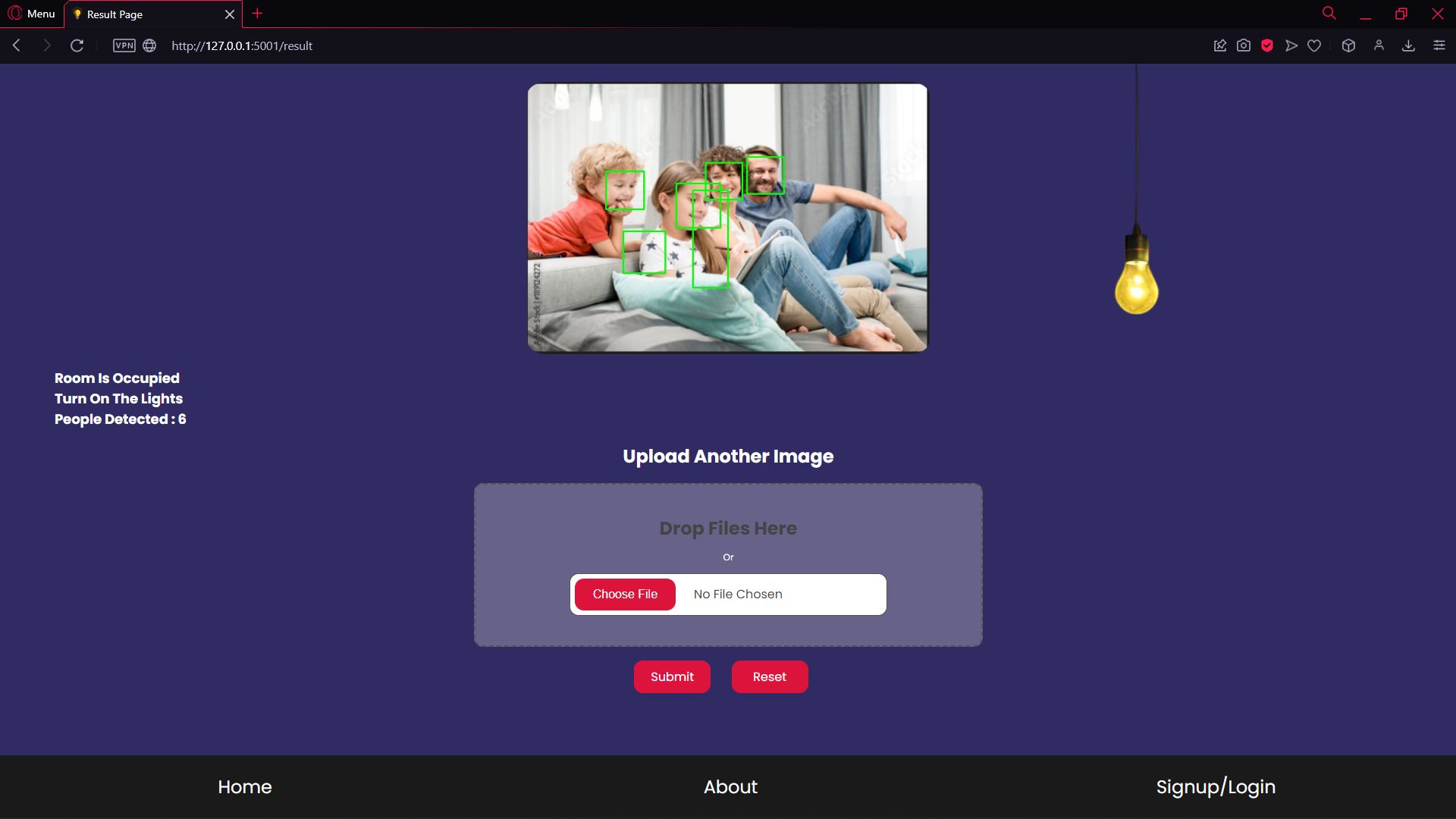


Image Upload Page:

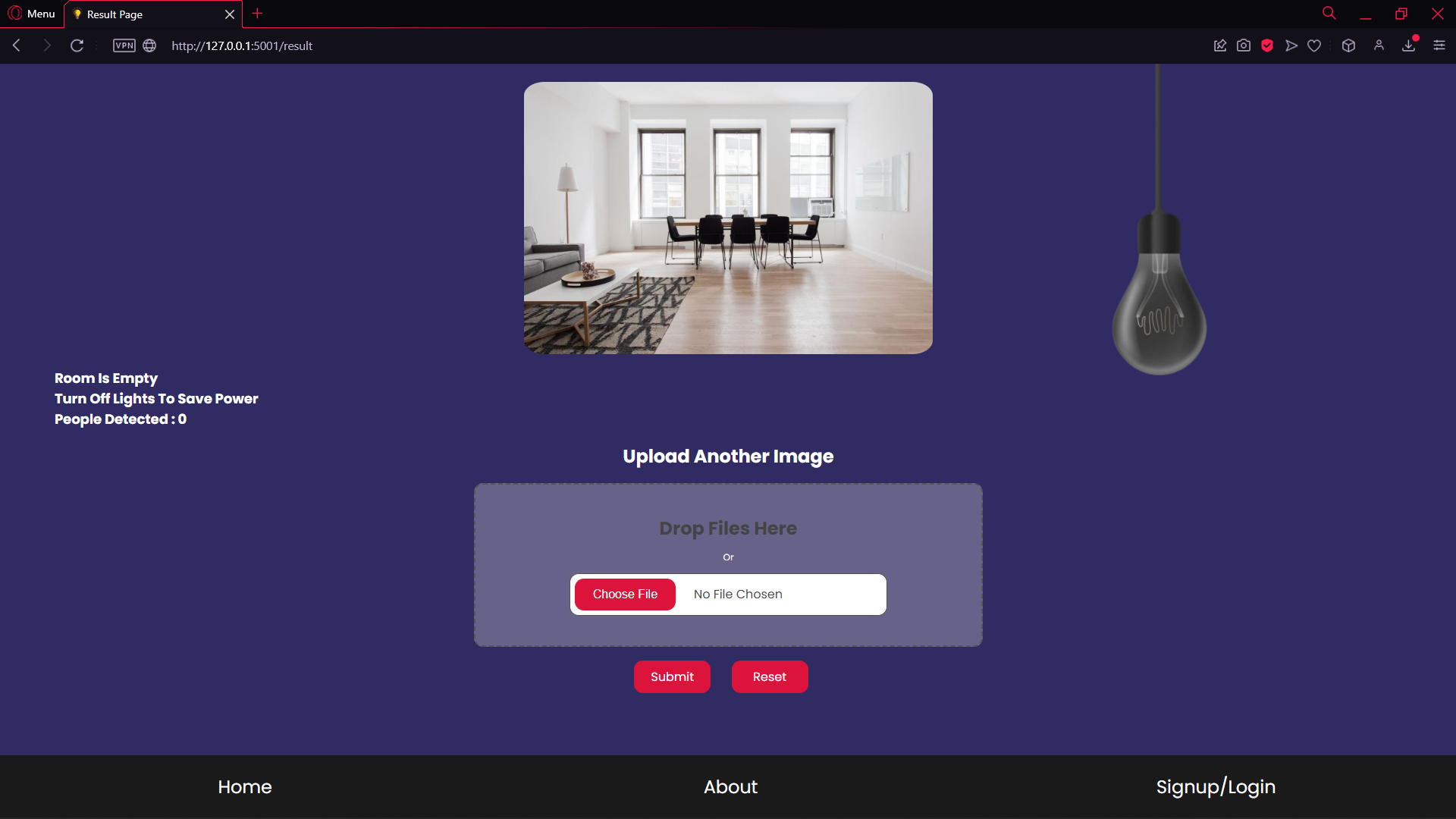


Result Page:

Occupied Room:



Empty Room:



**Sample code:**

Signup.php (Form validation & Database Insertion):

<script>

  const userNameInput = document.querySelector('input[name="txt"]');

  // add an event listener for input change

  userNameInput.addEventListener('input', () => {

    // get the user name value

    const userName = userNameInput.value;

    // check if the user name is at least 5 characters long

    if (userName.length < 5) {

      userNameInput.setCustomValidity('User name must be at least 5 characters long');

    } else {

      // check if the user name contains numbers or special characters

      const regex = /^[a-zA-Z]+$/;

      if (!regex.test(userName)) {

        userNameInput.setCustomValidity('User name cannot contain numbers or special characters');

      } else {

        userNameInput.setCustomValidity('');

      }

    }

  });

  // prevent form submission if there are invalid fields

  const signUpForm = document.querySelector('.signup form');

  signUpForm.addEventListener('submit', (event) => {

    if (!signUpForm.checkValidity()) {

      event.preventDefault();

    }

  });

  const passwordInput = document.querySelector('input[name="pswd"]');

  // add an event listener for input change

  passwordInput.addEventListener('input', () => {

    // get the password value

    const password = passwordInput.value;

    // check if the password meets the requirements

    const regex = /^(?=.\*\d)(?=.\*[!@#$%^&\*])(?=.\*[A-Z]).{6,}$/;

    if (!regex.test(password)) {

      passwordInput.setCustomValidity('Password must contain at least one uppercase letter, one special character, and be at least 6 characters long');

    } else {

      passwordInput.setCustomValidity('');

    }

  });

  // prevent form submission if there are invalid fields

  signUpForm.addEventListener('submit', (event) => {

    if (!signUpForm.checkValidity()) {

      event.preventDefault();

    }

  });

</script>

<?php

$sname= "localhost";

$unmae= "root";

$password = "";

$db\_name = "auto\_lights";

$conn = mysqli\_connect($sname, $unmae, $password, $db\_name);

//print\_r($\_POST);

$username = $\_POST["txt"];

$email = $\_POST["email"];

$password = $\_POST["pswd"];

// Prepare and execute the SQL query

$sql = "INSERT INTO user (s\_no, username, email, password) VALUES (0,'$username', '$email', '$password')";

if (mysqli\_query($conn, $sql)) {

    echo "<script>

  function redirect() {

    var confirmed = confirm('Account Created, Click yes to go to Login Page');

    if (confirmed) {

      window.location.href = 'http://127.0.0.1/hackathon/';

    }

  }

  redirect();

</script>

";

  } else {

    echo "<script>alert('Account NOT Created ')</script>";

    echo  mysqli\_error($conn);

  }

?>

<?php

session\_start();

include "db\_connection.php";

if (isset($\_POST['txt']) && isset($\_POST['pswd'])) {

    function validate($data){

       $data = trim($data);

       $data = stripslashes($data);

       $data = htmlspecialchars($data);

       return $data;

    }

    $uname = validate($\_POST['txt']);

    $pass = validate($\_POST['pswd']);

    if(1==1){

        $sql = "SELECT \* FROM user WHERE username='$uname' AND password='$pass'";

        $result = mysqli\_query($conn, $sql);

        if (mysqli\_num\_rows($result) > 0 ) {

            $row = mysqli\_fetch\_assoc($result);

            if ($row['username'] === $uname && $row['password'] === $pass) {

                echo "Logged in!";

                $\_SESSION['username'] = $row['username'];

                $\_SESSION['password'] = $row['password'];

                header("Location: http://127.0.0.1:5001/home\_page");

                exit();

            }else{

                echo "<script>alert('Password is required')</script>";

                exit();

            }

        }else{

        echo "<script>

        function redirect() {

            alert('Incorrect USERID/PASSWORD Click yes to go to Login Page');

            window.location.href = 'http://127.0.0.1/hackathon/';

        }

        redirect();

        </script>

        ";

            exit();

        }

    }

}else{

    header("Location: http://127.0.0.1/hackathon/");

    exit();

}

Detect People:

import cv2

def detect\_():

   # Reading the Image

   image = cv2.imread("static\\uploaded\_files\\image.jpg")

   height, width, channels = image.shape

   aspect\_ratio = width / height

   new\_width=700

   new\_height = int(new\_width / aspect\_ratio)

   image = cv2.resize(image, (new\_width, new\_height))

   # initialize the HOG descriptor

   hog = cv2.HOGDescriptor()

   hog.setSVMDetector(cv2.HOGDescriptor\_getDefaultPeopleDetector())

   # detect humans in input image

   (humans, \_) = hog.detectMultiScale(image, winStride=(10, 10),

   padding=(32, 32), scale=1.1)

   face\_cascade = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

   # Convert the image to grayscale

   gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

   # Detect faces in the image

   faces = face\_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)

   # loop over all detected humans

   for (x, y, w, h) in humans:

      pad\_w, pad\_h = int(0.15 \* w), int(0.01 \* h)

      cv2.rectangle(image, (x + pad\_w, y + pad\_h), (x + w - pad\_w, y + h - pad\_h), (0, 255, 0), 2)

   # Draw rectangles around the detected faces

   for (x, y, w, h) in faces:

      cv2.rectangle(image, (x, y), (x+w, y+h), (0, 255, 0), 2)

   # display the output image

   cv2.imwrite("static\\uploaded\_files\\out\_image.jpg", image)

   MESSAGE = ""

   im\_pth=""

   # getting no. of human detected

   if len(humans)>0 or len(faces)>0:

      print("Room is occupied")

      MESSAGE += "Room is occupied<br>Turn on the Lights"

      im\_pth="BULBON.png"

   else:

      print("Room is empty")

      MESSAGE += "Room is empty<br>Turn off Lights to save power "

      im\_pth="BULBOFF.png"

   print('Human Detected : ', len(humans)+len(faces))

   MESSAGE += ('<br>People Detected : '+ str(len(humans)+len(faces)))

   return MESSAGE,im\_pth