Voice + video heart disease

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Heart Disease Detection</title>

<link rel="stylesheet" href="style.css">

<script src="https://cdn.jsdelivr.net/npm/chart.js"></script>

</head>

<body>

<div class="container">

<h1>Heart Disease Detection App</h1>

<!-- Login Form -->

<form id="loginForm">

<label for="username">Name:</label>

<input type="text" id="username" required>

<label for="password">Password:</label>

<input type="password" id="password" required>

<button type="submit">Login</button>

</form>

<!-- Health Data Form -->

<div id="healthForm" class="hidden">

<h2>Enter Health Data</h2>

<label for="age">Age:</label>

<input type="number" id="age" required>

<label for="height">Height (cm):</label>

<input type="number" id="height" required>

<label for="weight">Weight (kg):</label>

<input type="number" id="weight" required>

<label for="heartRate">Resting Heart Rate:</label>

<input type="number" id="heartRate" required>

<label for="maxHR">Maximum Heart Rate Achieved:</label>

<input type="number" id="maxHR" required>

<label for="bloodPressure">Blood Pressure (mmHg):</label>

<input type="number" id="bloodPressure" required>

<label for="cholesterol">Cholesterol Level (mg/dL):</label>

<input type="number" id="cholesterol" required>

<label for="smoking">Smoking History:</label>

<select id="smoking">

<option value="0">Non-Smoker</option>

<option value="1">Current Smoker</option>

<option value="2">Ex-Smoker</option>

</select>

<label for="familyHistory">Family History of Heart Disease:</label>

<select id="familyHistory">

<option value="0">No</option>

<option value="1">Yes</option>

</select>

<label for="voiceAnalysis">Voice-Based Heart Disease Detection:</label>

<button id="startVoiceRecognition">Start Voice Analysis (5 sec)</button>

<p id="voiceResult"></p>

<!-- Eye Image Capture -->

<div id="eyeCaptureContainer">

<h3>Real-Time Eye Image Capture</h3>

<video id="video" width="320" height="240" autoplay></video>

<button id="captureButton">Capture Eye Image</button>

<canvas id="canvas" width="320" height="240" class="hidden"></canvas>

<img id="capturedImage" class="hidden" />

<p id="eyeAnalysisResult"></p>

</div>

<button onclick="detectHeartDisease()">Detect Heart Disease</button>

<button id="graphButton" class="hidden" onclick="showGraph()">Show Graph</button>

<p id="result"></p>

</div>

<!-- Graph Container -->

<div id="graphContainer" class="hidden">

<canvas id="riskChart" width="400" height="400"></canvas>

</div>

</div>

<script src="script.js"></script>

</body>

</html>

…….

Css

document.getElementById('loginForm').addEventListener('submit', function (e) {

e.preventDefault();

const username = document.getElementById('username').value;

const password = document.getElementById('password').value;

if (password === 'kavita@123#') {

document.getElementById('loginForm').classList.add('hidden');

document.getElementById('healthForm').classList.remove('hidden');

} else {

alert('Incorrect password! Please try again.');

}

});

// ✅ Voice Analysis (Using Audio Frequency)

document.getElementById('startVoiceRecognition').addEventListener('click', async function () {

let audioContext = new (window.AudioContext || window.webkitAudioContext)();

let analyser = audioContext.createAnalyser();

analyser.fftSize = 512;

navigator.mediaDevices.getUserMedia({ audio: true })

.then(stream => {

let source = audioContext.createMediaStreamSource(stream);

source.connect(analyser);

let dataArray = new Uint8Array(analyser.frequencyBinCount);

setTimeout(() => {

analyser.getByteFrequencyData(dataArray);

let avgPitch = dataArray.reduce((a, b) => a + b, 0) / dataArray.length;

let voiceResult = (avgPitch < 80) ? "⚠️ Possible Heart Disease" : "✅ No Heart Disease";

document.getElementById('voiceResult').textContent = voiceResult;

// ✅ Auto-update detection result

detectHeartDisease();

}, 5000);

})

.catch(error => {

console.error("Microphone access error: ", error);

alert("Failed to access the microphone. Please check your permissions.");

});

});

// ✅ Ensure the camera starts properly

const video = document.getElementById('video');

const canvas = document.getElementById('canvas');

const captureButton = document.getElementById('captureButton');

const capturedImage = document.getElementById('capturedImage');

// ✅ Start the camera when the page loads

navigator.mediaDevices.getUserMedia({ video: true })

.then(stream => {

video.srcObject = stream;

video.play(); // Ensure the video starts

})

.catch(error => {

console.error("Camera access error: ", error);

alert("Failed to access the camera. Please check your permissions.");

});

// ✅ Capture and process the eye image

captureButton.addEventListener('click', function () {

const context = canvas.getContext('2d');

// ✅ Capture the current frame from the video

context.drawImage(video, 0, 0, canvas.width, canvas.height);

// ✅ Convert canvas to image and display it

let imageUrl = canvas.toDataURL("image/png");

capturedImage.src = imageUrl;

capturedImage.classList.remove('hidden');

// ✅ Analyze Eye Color for Disease Detection

let imgData = context.getImageData(0, 0, canvas.width, canvas.height);

let colorSum = 0;

for (let i = 0; i < imgData.data.length; i += 4) {

let r = imgData.data[i]; // Red

let g = imgData.data[i + 1]; // Green

let b = imgData.data[i + 2]; // Blue

let avgPixel = (r + g + b) / 3;

colorSum += avgPixel;

}

let avgColor = colorSum / (imgData.data.length / 4); // Normalize avg color

let eyeResult = (avgColor < 100) ? "⚠️ Possible Heart Disease" : "✅ Eye looks normal";

document.getElementById('eyeAnalysisResult').textContent = eyeResult;

// ✅ Auto-update disease detection

detectHeartDisease();

});

// ✅ Heart Disease Detection (Matching Voice & Eye Analysis)

function detectHeartDisease() {

let voiceResult = document.getElementById('voiceResult').textContent;

let eyeResult = document.getElementById('eyeAnalysisResult').textContent;

let resultText = "✅ Your heart health looks normal.";

// ✅ Ensuring both results match

if (voiceResult.includes("Heart Disease") && eyeResult.includes("Heart Disease")) {

resultText = "⚠️ High Risk: Possible Heart Disease";

} else if (voiceResult.includes("Heart Disease") || eyeResult.includes("Heart Disease")) {

resultText = "⚠️ Medium Risk: Possible Heart Issue";

}

document.getElementById('result').textContent = resultText;

}

// ✅ Graph for Health Metrics

function showGraph() {

new Chart(document.getElementById('riskChart').getContext('2d'), {

type: 'bar',

data: {

labels: ["Heart Rate", "Blood Pressure", "Cholesterol"],

datasets: [{ label: "Metrics", data: [80, 120, 200] }]

}

});

}