**Report For Runlinc**

**(A case study for AI Voice Recognition)**

By

Afolayan Oluwatimilehin Israel

080-452

**Introduction**

In today's fast-paced digital world, user interaction with technology is evolving to become more natural and intuitive. Voice-activated systems are a significant step in this direction, providing users with the ability to control devices and applications hands-free. This project, developed using the Runlinc platform, focuses on creating an interactive voice button system that can recognize specific phrases and respond accordingly. The goal is to enhance user experience by making interactions more engaging and accessible.

**Background**

Voice-activated technology has become increasingly prevalent with the advent of smart assistants like Amazon's Alexa, Google Assistant, and Apple's Siri. These technologies allow users to perform a wide range of tasks using simple voice commands, from setting reminders to controlling smart home devices. This hands-free interaction is particularly beneficial for accessibility, enabling users with disabilities to interact with technology more easily.

Runlinc is a versatile web-based platform that facilitates the development and deployment of interactive applications. Leveraging the capabilities of Runlinc, this project aims to integrate voice recognition and synthesis to create a set of voice-activated buttons. These buttons can trigger different actions and provide auditory feedback, enhancing the overall user experience.

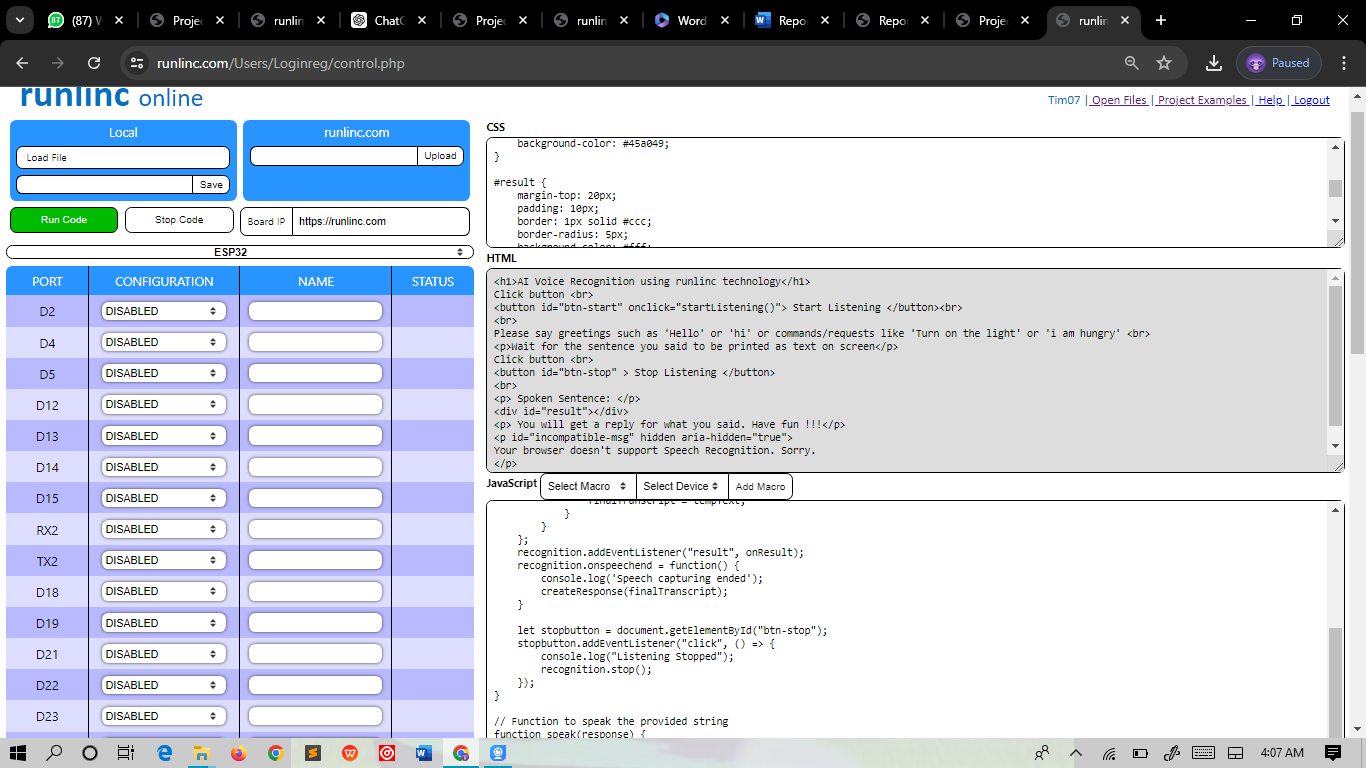
The key features of this project include:

1. **Voice Recognition**: The system listens for specific voice commands and processes them to determine the appropriate response. This feature is implemented using JavaScript and the Web Speech API, which provides speech recognition and synthesis capabilities.
2. **Interactive Feedback**: Visual and auditory feedback are provided to the user in response to recognized commands. This dual-mode feedback ensures that users receive confirmation of their actions, making the system more reliable and user-friendly.
3. **Accessibility**: A primary objective of this project is to improve accessibility for users with disabilities. Voice commands offer an alternative to traditional input methods, making it easier for users with limited mobility or visual impairments to interact with the application.
4. **Custom Responses**: The system can be customized to recognize and respond to a variety of phrases. For instance, if the user says, "I am hungry," the system responds with "How about you go get what to eat, i ain’t getting you nothing." This flexibility allows the application to be tailored to specific user needs and preferences.

**Visual Content**

Screenshots of codes below:

HTML, CSS and JavaScript codes



**HTML**

<h1>AI Voice Recognition using runlinc technology</h1>

Click button <br>

<button id="btn-start" onclick="startListening()"> Start Listening </button><br>

<br>

Please say greetings such as 'Hello' or 'hi' or commands/requests like 'Turn on the light' or 'i am hungry' <br>

<p>Wait for the sentence you said to be printed as text on screen</p>

Click button <br>

<button id="btn-stop" > Stop Listening </button>

<br>

<p> Spoken Sentence: </p>

<div id="result"></div>

<p> You will get a reply for what you said. Have fun !!!</p>

<p id="incompatible-msg" hidden aria-hidden="true">

Your browser doesn't support Speech Recognition. Sorry.

</p>

**CSS**

body {

font-family: Arial, sans-serif;

background-color: #f0f0f0;

color: #333;

margin: 0;

padding: 20px;

}

h1 {

text-align: center;

color: #4CAF50;

}

button {

background-color: #4CAF50;

color: white;

padding: 10px 20px;

border: none;

border-radius: 5px;

cursor: pointer;

font-size: 16px;

margin: 10px 0;

}

button:hover {

background-color: #45a049;

}

#result {

margin-top: 20px;

padding: 10px;

border: 1px solid #ccc;

border-radius: 5px;

background-color: #fff;

min-height: 50px;

}

p {

font-size: 18px;

line-height: 1.5;

}

#incompatible-msg {

color: red;

font-weight: bold;

text-align: center;

}

**JavaScript**

// Main Function to start listening

function startListening() {

console.log("in start");

let result = document.getElementById("result");

let finalTranscript;

// recognition = window.SpeechRecognition();

let speechRecognition = window.SpeechRecognition || window.webkitSpeechRecognition;

let recognition = new speechRecognition();

recognition.continuous = true;

recognition.interimResults = false;

recognition.maxAlternatives = 5;

recognition.start();

const onResult = event => {

result.innerHTML = "";

finalTranscript = "";

for (const res of event.results) {

const tempText = res[0].transcript;

console.log(tempText);

if (res.isFinal) {

const text = document.createTextNode(tempText);

const p = document.createElement("p");

p.appendChild(text);

result.appendChild(p);

finalTranscript = tempText;

}

}

};

recognition.addEventListener("result", onResult);

recognition.onspeechend = function() {

console.log('Speech capturing ended');

createResponse(finalTranscript);

}

let stopbutton = document.getElementById("btn-stop");

stopbutton.addEventListener("click", () => {

console.log("Listening Stopped");

recognition.stop();

});

}

// Function to speak the provided string

function speak(response) {

console.log(response);

const speech = new window.SpeechSynthesisUtterance(response);

speechSynthesis.speak(speech);

}

// Creates response after listening

function createResponse(sentence) {

var greetwords = ['hello', 'hi'];

var commandWords = ['command', 'turn on', 'turn off'];

var hungryWords = ['i am hungry'];

var greetStr = '';

var responseStr = "Command Received: ";

console.log(sentence);

// Check for greeting words

for (var greet of greetwords) {

if (sentence.toLowerCase().indexOf(greet) > -1) {

greetStr = "Hello, what's popping?!";

break;

}

}

// Check for command words

for (var comm of commandWords) {

if (sentence.toLowerCase().indexOf(comm) > -1) {

greetStr = responseStr + sentence;

break;

}

}

// Check for "I am hungry" phrase

for (var hungry of hungryWords) {

if (sentence.toLowerCase().indexOf(hungry) > -1) {

greetStr = "How about you go out and get what to eat from a restaurant nearby.";

break;

}

}

speak(greetStr);

}

**OUTPUT**

