```
<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>Document</title>
</head>
<style>
    body {
        font-family: 'Share Tech', sans-serif;
        font-size: 17px;
        color: white;
        display: flex;
        justify-content: center;
        align-items: center;
        margin: 0;
        width: 100vw;
        height: 100vh;
        text-shadow: 8px 8px 10px
```

```
#0000008c;
        background-color: #343a40;
        background-image: url("data:image/
svg+xml,%3Csvg xmlns='http://
www.w3.org/2000/svg' width='28'
height='49' viewBox='0 0 28 49'%3E%3Cg
fill-rule='evenodd'%3E%3Cg id='hexagons'
fill='%239C92AC' fill-opacity='0.25' fill-
rule='nonzero'%3E%3Cpath d='M13.99
9.25l13 7.5v15l-13 7.5L1
31.75v-15l12.99-7.5zM3 17.9v12.7l10.99
6.34 11-6.35V17.9l-11-6.34L3 17.9zM0
15l12.98-7.5V0h-2v6.35L0 12.69v2.3zm0
18.5L12.98 41v8h-2v-6.85L0
35.81v-2.3zM15 0v7.5L27.99
15H28v-2.31h-.01L17 6.35V0h-2zm0
49v-8l12.99-7.5H28v2.31h-.01L17
42.15V49h-2z'/%3E%3C/g%3E%3C/
g%3E%3C/svg%3E"), linear-gradient(to
right top, #343a40, #2b2c31, #211f22,
#151314, #000000);
    }
```

```
    h1 {
        text-align: right;
        margin: 20px;

    }

    textarea {
        text-align: right;
        width: 50%;
        height: 200px;
        padding: 12px 20px;
        box-sizing: border-box;
        border: 2px solid #ccc;
        border-radius: 4px;
        background-color: #f8f8f8;
        font-size: 16px;
        resize: none;

    }
</style>

<body>
```

```
    <h1>ملتقط الصوت</h1>
    <textarea type="text" id="speechToText"
placeholder="... اضغط هنا ثم تحدث "
onclick="record()"></textarea>
    <button
onclick="connectSerial()">اتصال</button>

    <script>
        var port, textEncoder,
writableStreamClosed, writer;
        async function connectSerial() {
            try {
                // Prompt user to select any serial
port.
                port = await
navigator.serial.requestPort();
                await port.open({ baudRate:
9600 });

                textEncoder = new
TextEncoderStream();
                writableStreamClosed =
```

```
textEncoder.readable.pipeTo(port.writable);

                writer =
textEncoder.writable.getWriter();
                listenToPort();
            } catch {
                alert("Serial Connection Failed");
            }
        }


        function record() {
            var recognition = new
webkitSpeechRecognition();
            recognition.lang = "ar";

            recognition.onresult = function
(event) {

                var a =
document.getElementById('speechToText').v
alue = event.results[0][0].transcript;
```

```
            if (a == "يمين." || a=="يمين") {
                console.log(a)
                sendSerialLine();

            }else if(a == "يسار." || a=="يسار") {
                console.log(a)
                sendSerialLineB();

            }

        }
        recognition.start();

    }


document.querySelector('button').addEvent
Listener('click', async () => {
        const port = await
navigator.serial.requestPort();
        await port.open({ baudRate: 9600 });
```

```
    });


    async function listenToPort() {
        const textDecoder = new
TextDecoderStream();
        const readableStreamClosed =
port.readable.pipeTo(textDecoder.writable);
        const reader =
textDecoder.readable.getReader();
        // Listen to data coming from the
serial device.
        while (true) {
            const { value, done } = await
reader.read();
            if (done) {
                // Allow the serial port to be
closed later.
```

```
                reader.releaseLock();
                break;
            }
            // value is a string.
            appendToTerminal(value);
        }


    }
    async function sendSerialLine()
{dataToSend = 'A'
        dataToSend = dataToSend + "\r\n";

        await writer.write(dataToSend);
    }
    async function sendSerialLineB() {
        dataToSend = 'B'
        dataToSend = dataToSend + "\r\n";

        await writer.write(dataToSend);
    }
```

```
    </script>

</body>

</html>
```

```
#include <Servo.h>

volatile char v;
Servo servo_7;

void setup(){
  v = 0;
  Serial.begin(9600);
  servo_7.attach(7);
}

void loop(){
  v = Serial.read();
  if (Serial.available() > 0) {
    if (v == 'A') {
      servo_7.write(0);
      delay(100);

    }
    if (v == 'B') {
      servo_7.write(180);
      delay(100);
```

```
        }

    }

}
```