IOT&Wisdom ESP32 algorithm

1. Understand how s-m.com.sa/test/ convert ur audio to text (site experience only).
2. Writing a wisdom ESP32 operating algorithm.

1- Understand how s-m.com.sa/test/ convert ur audio to text (site experience only):

<file:///Users/sarahalghanmi/Desktop/ss.html>

<!doctype html>

<head>

<style>

body {

font-family: arial;

text-align:center;

background-color:black;

color: white;

}

button {

padding:10px;

background-color:pink;

border: 0px;

cursor:pointer;

border-radius: 5px;

}

#output {

background-color:pink;

padding:10px;

width: 100%;

margin-top:20px;

line-height:30px;

}

.hide {

display:none;

}

.show {

display:block;

}

</style>

<title>JavaScript Speech to Text</title>

</head>

<body>

<p>Speak something...</p>

<p><button type="button" onclick="runSpeechRecognition()">Speech to Text</button> &nbsp; <span id="action"></span></p>

<div id="output" class="hide"></div>

<script>

/\* JS comes here \*/

function runSpeechRecognition() {

var output = document.getElementById("output");

var action = document.getElementById("action");

var SpeechRecognition = SpeechRecognition || webkitSpeechRecognition;

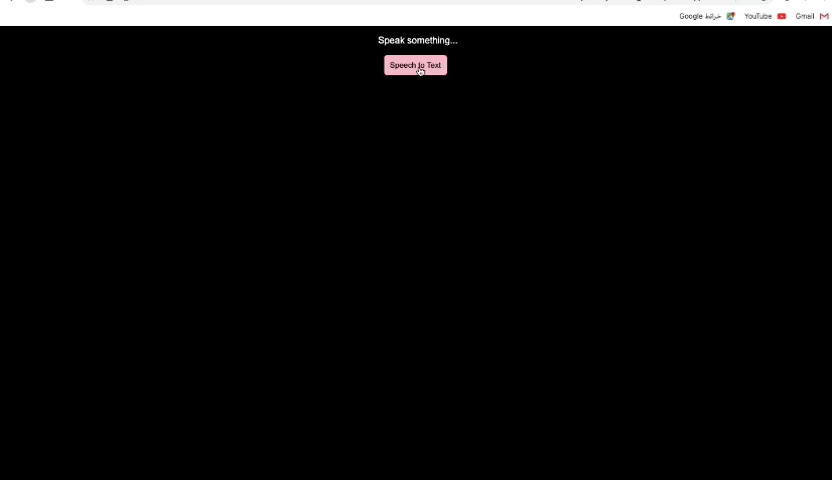
var recognition = new SpeechRecognition();

recognition.onstart = function() {

action.innerHTML = "<small>listening, please speak...</small>";

};

recognition.onspeechend = function() {

 action.innerHTML = "<small>stopped listening, hope you are done...</small>";

recognition.stop();

}

recognition.onresult = function(event) {

var transcript = event.results[0][0].transcript;

output.innerHTML = "<br/> <b>" + transcript ;

output.classList.remove("hide");

};

recognition.start();

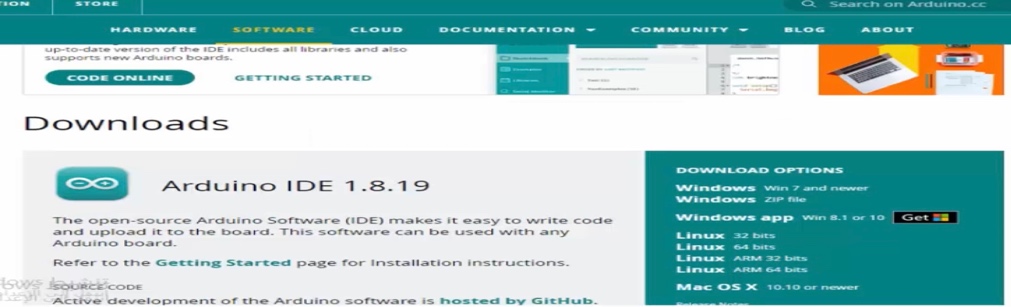
}

</script>

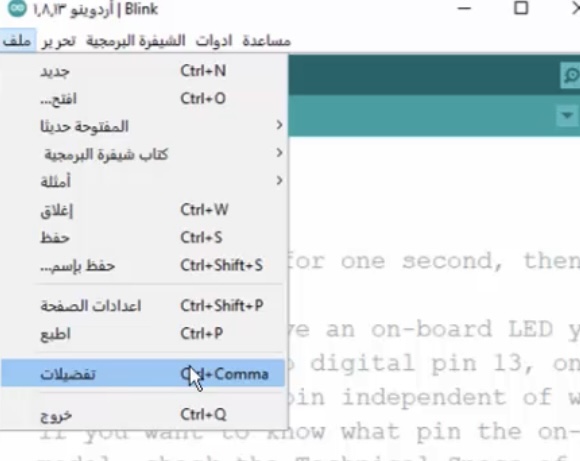
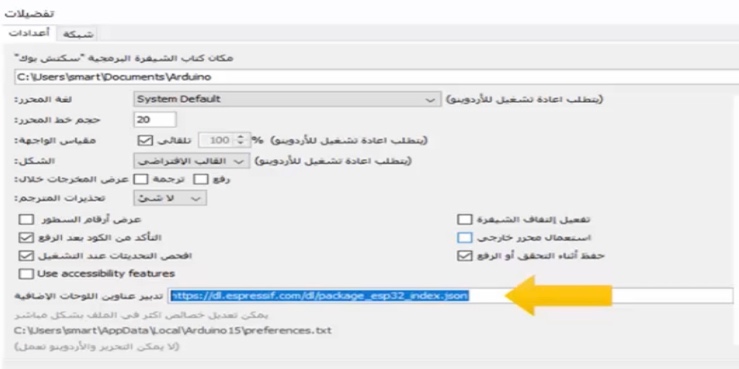
</body>

</html>

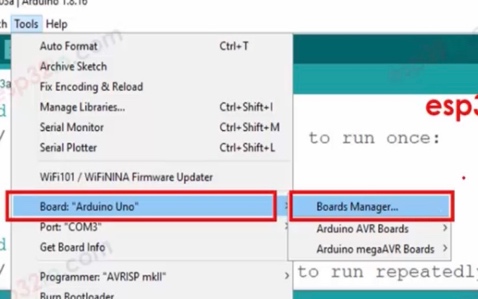
1. Writing a wisdom ESP32 operating algorithm:
2. Download Arduino IDE from the Arduino website.



1. To see the ESP32 piece, click on File at the top, click on Preferences, add this line https://dl.espressif.com/dl/package\_esp32\_index.json, and click OK.

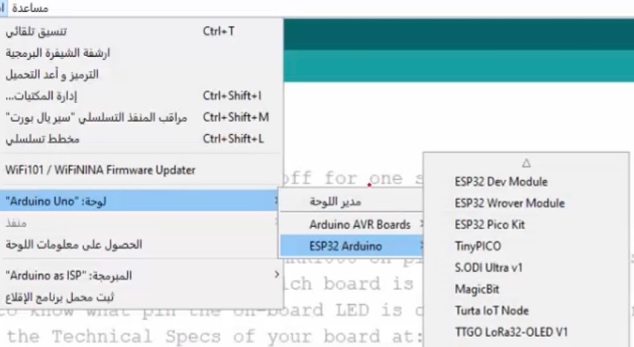


1. Click Tools then choose Arduino uno, then click Board Manager, then a screen will appear we are writing ESP32, then click enter and download.





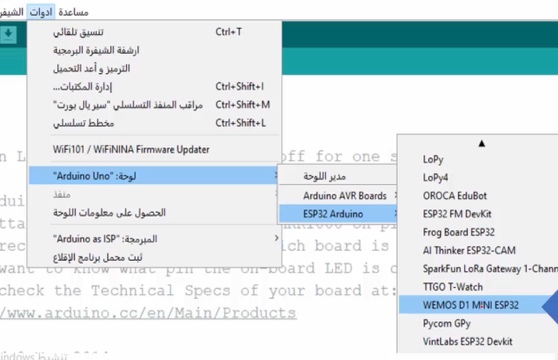
1. Go to the board and find ESP32 Arduino.



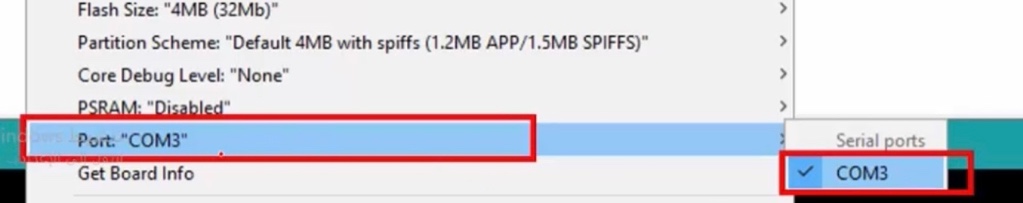
1. ESP32 piece, is connected to the USB wire.



1. To connect, we click on the Arduino uno tools, then choose ESP32 Arduino, then choose this type of WEMOS D1 MINI ESP32 controller.



1. Then we go to tools and connect to a piece by going to port "com3" and then clicking on "com3"



1. To turn the lights on and off, we go to the file, then choose examples, then 01.Basics, then blink. After that, the blink example will be opened, we click on the start button and the code will run.

