

Code in NodeMCU ESP8266 for servomotor controlling:

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
//-----Include the Servo Library
#include <Servo.h>

#include "PageIndex.h"; //--> Include the contents of the User Interface Web page, stored in
the same folder as the .ino file

#define ServoPort D1  //--> Defining Servo Port

//-----Make a wifi name and password as access point
const char* ssid = "NodeMCU_ESP8266";
const char* password = "goodluck";
//-----

Servo myservo; //--> create servo object to control a servo

ESP8266WebServer server(80); //--> Server on port 80

//-----This routine is executed when you open NodeMCU ESP8266
IP Address in browser
void handleRoot() {
  String s = MAIN_page; //Read HTML contents
  server.send(200, "text/html", s); //Send web page
}
//-----

//-----Procedure for handling servo control
void handleServo(){
  String POS = server.arg("servoPOS");
  int pos = POS.toInt();
  myservo.write(pos); //--> Move the servo motor according to the POS value
  delay(15);
  Serial.print("Servo Angle:");
  Serial.println(pos);
  server.send(200, "text/plain", "");
}
//-----
//-----Setup-----
void setup() {
  Serial.begin(115200);
  delay(500);

  myservo.attach(ServoPort); //--> attaches the servo on D1 to the servo object
```

```
WiFi.softAP(ssid, password); //--> Start Making ESP8266 NodeMCU as an access point
```

```
IPAddress apip = WiFi.softAPIP(); //--> Get the IP server
```

```
Serial.print("Connect your wifi laptop/mobile phone to this NodeMCU Access Point : ");
```

```
Serial.println(ssid);
```

```
Serial.print("Visit this IP : ");
```

```
Serial.print(apip); //--> Prints the IP address of the server to be visited
```

```
Serial.println(" in your browser.");
```

```
//-----
```

```
//-----Initialize Webserver
```

```
server.on("/",handleRoot); //--> Routine to handle at root location. This is to display web page.
```

```
server.on("/setPOS",handleServo); //--> Sets servo position from Web request
```

```
server.begin();
```

```
Serial.println("HTTP server started");
```

```
}
```

```
//-----
```

```
//-----Loop-----
```

```
void loop() {
```

```
  server.handleClient();
```

```
}
```

```
//-----
```

Code for HTML Web Server of ESP8266 NodeMCU Wi-Fi:

```
const char MAIN_page[] PROGMEM = R"=====(
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<style>
```

```
  html {
```

```
    font-family: Arial;
```

```
    display: inline-block;
```

```
    margin: 0px auto;
```

```
    text-align: center;
```

```
  }
```

```
  .slidecontainer {
```

```
    width: 100%;
```

```
  }
```

```
  .slider {
```

```
    -webkit-appearance: none;
```

```
    width: 50%;
```

```

height: 15px;
border-radius: 5px;
background: #d3d3d3;
outline: none;
opacity: 0.7;
-webkit-transition: .2s;
transition: opacity .2s;
}

.slider:hover {
  opacity: 1;
}

.slider::-webkit-slider-thumb {
  -webkit-appearance: none;
  appearance: none;
  width: 25px;
  height: 25px;
  border-radius: 50%;
  background: #4CAF50;
  cursor: pointer;
}

.slider::-moz-range-thumb {
  width: 25px;
  height: 25px;
  border-radius: 50%;
  background: #4CAF50;
  cursor: pointer;
}
</style>
</head>
<body>

<h1>NodeMCU ESP8266 / ESP12E Control Servo SG90</h1>
<br><br>
<div class="slidecontainer">
  <input type="range" min="0" max="180" value="50" class="slider" id="myRange">
  <p>Value : <span id="demo"></span></p>
</div>

<script>
function sendData(pos) {
  var xhttp = new XMLHttpRequest();
  xhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      console.log(this.responseText);
    }
  };
  xhttp.open("GET", "sendData.php?pos=" + pos, true);
  xhttp.send();
}
</script>

```

```

    }
  };
  xhttp.open("GET", "setPOS?servoPOS="+pos, true);
  xhttp.send();
}
var slider = document.getElementById("myRange");
var output = document.getElementById("demo");
output.innerHTML = slider.value;

slider.oninput = function() {
  output.innerHTML = this.value;
  sendData(output.innerHTML);
}
</script>

</body>
</html>
)====="";

```

Both the codes has to be loaded into the NodeMCU using Arduino IDE.