```
//This program is the ESP8266 side of Robo_Car project.

#include<ESP8266WiFi.h>
#include<ESP8266mDNS.h>
#include<WiFiClient.h>
#include <Servo.h>

Servo armServo;
Servo tweezerServo;
int Speed=6;

const char* ssid = "JioFi3_2875D4";
const char* password = "deepudeepthi";
String
cmds[]={"arm_up","arm_down","tweezers_tighten","tweezers_loosen","move_forward","m
ove_backward","move_right","move_left","stop"};

// TCP server at port 80 will respond to HTTP requests
WiFiServer server(80);

void setup(void)
{

 Serial.begin(115200);
 armServo.attach(D7);
 tweezerServo.attach(D8);
 tweezerServo.write(0);

  // Connect to WiFi network
 WiFi.begin(ssid, password);

  // Wait for connection
 while (WiFi.status() != WL_CONNECTED) {
   delay(500);
     }

 if (!MDNS.begin("esp8266")) {
   while(1) {
     delay(1000);
   }
  }
 server.begin();

 MDNS.addService("http", "tcp", 80);
}

void loop(void)
{
```

```
 // Check if a client has connected
 WiFiClient client = server.available();
 if (!client) {
   return;
 }

 // Wait for data from client to become available
 while(client.connected() && !client.available()){
   delay(1);
 }

 // Read the first line of HTTP request
 String req = client.readStringUntil('\r');

 // First line of HTTP request looks like "GET /path HTTP/1.1"
 // Retrieve the "/path" part by finding the spaces
 int addr_start = req.indexOf(' ');
 int addr_end = req.indexOf(' ', addr_start + 1);
 if (addr_start == -1 || addr_end == -1) {

   return;
 }
 req = req.substring(addr_start+2 , addr_end);
  client.flush();

 String s;
 if (elementinarray(req,10,cmds)>0)
 {

  executecommand(req);
  s = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n<!DOCTYPE
HTML>\r\n<html>OK";
    s += "</html>\r\n\r\n";

 }
 else
 {
   s = "HTTP/1.1 404 Not Found\r\n\r\n";
    }
 client.print(s);

 }

int elementinarray(String element,int arraysize,String Array[]){
 for(int i=0;i<=(arraysize-1);i+=1){
  if((element.indexOf("speed_changev"))>-1){
     return 2;
   }
  if (Array[i]==element){
```

```
    return 1;
   }
  }
  return 0;
}
void executecommand(String cmd){

   if(cmd==cmds[0])
        {

         Serial.print('a');
                  }
   else if (cmd==cmds[1])
        {
         Serial.print('b');

                  }
   else if(cmd==cmds[2])
        {
         Serial.print('c');
                  }
   else if (cmd==cmds[3])
        {
           Serial.print('d');
          }

   else if(cmd==cmds[4]){
        Serial.print('e');
    }
   else if(cmd==cmds[5]){
        Serial.print('f');
    }
   else if(cmd==cmds[7]){
        Serial.print('g');
    }
   else if(cmd==cmds[6]){
      Serial.print('h');
    }
   else if(cmd==cmds[9]){
    Serial.print('i');
    }
   else if(cmd.indexOf(cmds[8])>-1){

   int i1=cmd.indexOf("v");
   int i2=cmd.length();
   String val=cmd.substring(i1+1,i2);
       Speed=val.toInt();
    }
```

}