```
#include <ESP8266WiFi.h>
#define IN1a 16
#define IN2a 5
#define IN3a 4
#define IN4a 0
#define IN1b 2
#define IN2b 14
#define IN3b 12
#define IN4b 13
int delayTime = 2;
WiFiClient client;
WiFiServer server(80);
const char* ssid = "YOUR_WiFi_NAME";
const char* password = "WiFi_PASSWORD";
String  data = "";
void setup() {
  pinMode(IN1a, OUTPUT);
  pinMode(IN2a, OUTPUT);
  pinMode(IN3a, OUTPUT);
  pinMode(IN4a, OUTPUT);
  pinMode(IN1b, OUTPUT);
  pinMode(IN2b, OUTPUT);
  pinMode(IN3b, OUTPUT);
  pinMode(IN4b, OUTPUT);
  Serial.begin(115200);
  connectWiFi();
  server.begin();
}
void loop() {
  client = server.available();
  if (!client) return;
  data = checkClient ();
  if (data == "forward") {
    for (int steps = 0; steps < 400; steps++) {
      Serial.println("FORWARD");
      forwardMotorA();
      forwardMotorB();
    }
  }
  else if (data == "left") {
    for (int steps = 0; steps < 400; steps++) {
      Serial.println("LEFT");
      backwardMotorA();
      forwardMotorB();
    }
  }
  else if (data == "right") {
    for (int steps = 0; steps < 500; steps++) {
      Serial.println("RIGHT");
      forwardMotorA();
      backwardMotorB();
    }
  }
  else if (data == "reverse") {
    for (int steps = 0; steps < 500; steps++) {
      Serial.println("BACKWARD");
      backwardMotorA();
      backwardMotorB();
    }
  }
  else if (data == "stop") {
    Serial.println("STOP");
    stopMotors();
  }
}
void connectWiFi()
{
  Serial.println("Connecting to WIFI");
  WiFi.begin(ssid, password);
  while ((!(WiFi.status() == WL_CONNECTED)))
  {
    delay(300);
    Serial.print("..");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("NodeMCU Local IP is : ");
  Serial.print((WiFi.localIP()));
  Serial.print("");
  Serial.println("");
}
String checkClient (void)
{
  while (!client.available()) delay(1);
  String request = client.readStringUntil('\r');
  request.remove(0, 5);
  request.remove(request.length() - 9, 9);
  return request;
}
void forwardMotorA(void) {
  digitalWrite(IN4a, HIGH);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, HIGH);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, HIGH);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, HIGH);
  delay(delayTime);
}
void forwardMotorB(void) {
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, HIGH);
  delay(delayTime);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, HIGH);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, HIGH);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
  digitalWrite(IN4b, HIGH);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
}
void backwardMotorA(void) {
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, HIGH);
  delay(delayTime);
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, HIGH);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, HIGH);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
  digitalWrite(IN4a, HIGH);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, LOW);
  delay(delayTime);
}
void backwardMotorB(void) {
  digitalWrite(IN4b, HIGH);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, HIGH);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, HIGH);
  digitalWrite(IN1b, LOW);
  delay(delayTime);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, HIGH);
  delay(delayTime);
}
void stopMotors(void) {
  digitalWrite(IN4a, LOW);
  digitalWrite(IN3a, LOW);
  digitalWrite(IN2a, LOW);
  digitalWrite(IN1a, LOW);
  digitalWrite(IN4b, LOW);
  digitalWrite(IN3b, LOW);
  digitalWrite(IN2b, LOW);
  digitalWrite(IN1b, LOW);
}
```