Project code

```
#include <SoftwareSerial.h>
 SoftwareSerial gsmSerial(7, 8);   // RX, TX to GSM module  const
 int pirPin = 2;     // Connect PIR output to this pin const
int irPin = 3;      // Connect IR sensor output to this pin const
int mq2Pin = A0;    // Connect MQ2 sensor output to this pin const
int s0Pin = 4;      // Connect S0 pin to this pin const
int s1Pin = 5;      // Connect S1 pin to this pin const
int s2Pin = 6;      // Connect S2 pin to this pin const
int s3Pin = 7;      // Connect S3 pin to this pin const
int sensorOutPin = 8; // Connect OUT pin to this pin
void setup()
 {
pinMode(pirPin, INPUT);
 pinMode(irPin, INPUT);
 pinMode(mq2Pin, INPUT);
 pinMode(s0Pin, OUTPUT);
pinMode(s1Pin, OUTPUT);
pinMode(s2Pin, OUTPUT);
pinMode(s3Pin, OUTPUT);
 pinMode(sensorOutPin, INPUT);
  Serial.begin(9600);
 gsmSerial.begin(9600);
  while (gsmSerial.available())
 {
 Serial.println("Initializing GSM module...");
   delay(1000);
  }
gsmSerial.println("AT+CMGF=1");
 delay(1000);
 Serial.println("GSM module ready!");
 }   void loop(){   int pirValue = 3
,gitalRead(pirPin);     // Read PIR sensor output
int irValue = digitalRead(irPin);       // Read IR obstacle sensor output
int mq2Value = analogRead(mq2Pin);      // Read MQ2 gas sensor output
```

```
 digitalWrite(s0Pin, HIGH);  // Choose red LED
digitalWrite(s1Pin, LOW);
  int redFrequency = pulseIn(sensorOutPin, LOW);
 digitalWrite(s2Pin, LOW);   // Choose blue LED
digitalWrite(s3Pin, LOW);
int blueFrequency = pulseIn(sensorOutPin, LOW);      // Check PIR sensor for
activity
 if (pirValue == HIGH)
 {
sendSMS("Activity detected by PIR sensor!");
delay(4000);  // Delay to avoid sending multiple messages
 }      // Check IR sensor for obstacle
 if (irValue == HIGH)
{
  sendSMS("Obstacle detected by IR sensor!");
  delay(4000);
// Delay to avoid sending multiple messages
  }
    // Check MQ2 gas sensor for gas
if (mq2Value > 500)
{
 sendSMS("Gas detected by MQ2 sensor!");
delay(4000); // Delay to avoid sending multiple messages
 }      // Check TCS3200 sensor for blue color
 if (blueFrequency > redFrequency)
 {     sendSMS("Blue color detected by TCS3200 sensor!");
   delay(4000); // Delay to avoid sending multiple messages
    } }   void sendSMS(String message)
 {
gsmSerial.println("AT+CMGS=\"+919490576265\"");
  // Replace "+91xxxxxxxxxx" with receiver's mobile number
  delay(1000);
gsmSerial.println(message);
delay(100);
gsmSerial.println((char)26);
delay(1000);   Serial.println(message);
```

```
}
```

Code For Car Control

```
#define ENA   14          // Enable/speed motors Right        GPIO14(D5)
#define ENB   12          // Enable/speed motors Left         GPIO12(D6)
#define IN_1  15          // L298N in1 motors Right           GPIO15(D8)
#define IN_2  13          // L298N in2 motors Right           GPIO13(D7)
#define IN_3  2           // L298N in3 motors Left            GPIO2(D4)
#define IN_4  0           // L298N in4 motors Left            GPIO0(D3)

#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
String command;             //String to store app command state.
int speedCar = 800;         // 400 - 1023.
int speed_Coeff = 3;
const char* ssid = "Indian Lifehacker wifi car";
ESP8266WebServer server(80);
void setup() {
 pinMode(ENA, OUTPUT);
 pinMode(ENB, OUTPUT);
 pinMode(IN_1, OUTPUT);
 pinMode(IN_2, OUTPUT);
 pinMode(IN_3, OUTPUT);
 pinMode(IN_4, OUTPUT);
  Serial.begin(115200);
// Connecting WiFi
  WiFi.mode(WIFI_AP);
  WiFi.softAP(ssid);
  IPAddress myIP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(myIP);
 // Starting WEB-server
    server.on ( "/", HTTP_handleRoot );
    server.onNotFound ( HTTP_handleRoot );
    server.begin();
}
```

```
void goAhead(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
  }
void goBack(){
    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);
    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
  }
void goRight(){
    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);
    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
  }
void goLeft(){
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
  }
```

```
void goAheadRight()
{
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar/speed_Coeff);
    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
  }

void goAheadLeft()
{
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);
    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar/speed_Coeff);
 }
void goBackRight(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar/speed_Coeff);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
  }

void goBackLeft(){

    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);
```

```
    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar/speed_Coeff);
  }

void stopRobot(){

    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
 }

void loop () {
   server.handleClient();

    command = server.arg("State");
    if (command == "F") goAhead();
    else if (command == "B") goBack();
    else if (command == "L") goLeft();
    else if (command == "R") goRight();
    else if (command == "I") goAheadRight();
    else if (command == "G") goAheadLeft();
    else if (command == "J") goBackRight();
    else if (command == "H") goBackLeft();
    else if (command == "0") speedCar = 400;
    else if (command == "1") speedCar = 470;
    else if (command == "2") speedCar = 540;
    else if (command == "3") speedCar = 610;
    else if (command == "4") speedCar = 680;
    else if (command == "5") speedCar = 750;
    else if (command == "6") speedCar = 820;
    else if (command == "7") speedCar = 890;
```

```
    else if (command == "8") speedCar = 960;
    else if (command == "9") speedCar = 1023;
    else if (command == "S") stopRobot();
}

void HTTP_handleRoot(void) {

if( server.hasArg("State") ){
      Serial.println(server.arg("State"));
  }
  server.send ( 200, "text/html", "" );
  delay(1);
}
```