```cpp
#include <ESP8266WiFi.h>
#include <Servo.h>

const char* ssid = "AcquicBotX";
const char* password = "12345678";

WiFiServer server(80);

// Pins
#define MQ135 A0
#define IR_SENSOR D1
#define SERVO_PROPULSION D2
#define SERVO_WING D3
#define L298N_IN1 D4
#define L298N_IN2 D5
#define RELAY_PUMP D6
#define TRIG D7
#define ECHO D8
#define BATTERY_PIN A0

Servo propulsionServo;
Servo wingServo;
```

```cpp
int gasThreshold = 400;
int batteryLevel = 0;

void setup() {
  Serial.begin(115200);
  WiFi.softAP(ssid, password);
  server.begin();

  pinMode(IR_SENSOR, INPUT);
  pinMode(L298N_IN1, OUTPUT);
  pinMode(L298N_IN2, OUTPUT);
  pinMode(RELAY_PUMP, OUTPUT);
  pinMode(TRIG, OUTPUT);
  pinMode(ECHO, INPUT);


propulsionServo.attach(SERVO_PROPULSION);
  wingServo.attach(SERVO_WING);

  digitalWrite(RELAY_PUMP, LOW);
}
```

```cpp
long readUltrasonicDistance() {
  digitalWrite(TRIG, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG, LOW);
  return pulseIn(ECHO, HIGH) / 58;
}

void loop() {
  WiFiClient client = server.available();
  if (client) {
    String request =
client.readStringUntil('\r');
    client.flush();

    if (request.indexOf("/forward") != -1) {
      propulsionServo.write(0);
    } else if (request.indexOf("/backward") !=
-1) {
      propulsionServo.write(180);
    } else if (request.indexOf("/left") != -1) {
```

```cpp
    wingServo.write(0);
  } else if (request.indexOf("/right") != -1) {
    wingServo.write(180);
  } else if (request.indexOf("/stop") != -1) {
    propulsionServo.write(90);
  } else if (request.indexOf("/feed") != -1) {
    digitalWrite(L298N_IN1, HIGH);
    digitalWrite(L298N_IN2, LOW);
    delay(1000);
    digitalWrite(L298N_IN1, LOW);
    digitalWrite(L298N_IN2, LOW);
  } else if (request.indexOf("/pump") != -1) {
    digitalWrite(RELAY_PUMP, HIGH);
    delay(2000);
    digitalWrite(RELAY_PUMP, LOW);
  }

  int gasValue = analogRead(MQ135);
  int oilDetected =
digitalRead(IR_SENSOR) == LOW;
  long distance = readUltrasonicDistance();
  batteryLevel =
map(analogRead(BATTERY_PIN), 0, 1023, 0,
```

```
  100);

  String response = "<!DOCTYPE
html><html><head><title>Acquic Bot X</
title></head><body>";
  response += "<h2>Acquic Bot X
Dashboard</h2>";
  response += "<p><a href=\"/
forward\">Forward</a> | <a href=\"/
backward\">Backward</a> | ";
  response += "<a href=\"/left\">Left</a> |
<a href=\"/right\">Right</a> | <a href=\"/
stop\">Stop</a></p>";
  response += "<p><a href=\"/feed\">Feed
Fish</a> | <a href=\"/pump\">Pump Water</
a></p>";
  response += "<p>Gas Level: " +
String(gasValue) + "</p>";
  response += "<p>Oil Detected: " +
String(oilDetected ? "Yes" : "No") + "</p>";
  response += "<p>Battery: " +
String(batteryLevel) + "%</p>";
  response += "<p>Distance (cm): " +
```

```
    String(distance) + "</p>";
    response += "</body></html>";

    client.print("HTTP/1.1 200
OK\r\nContent-Type: text/html\r\n\r\n");
    client.print(response);
    client.stop();

    if (gasValue > gasThreshold) {
      Serial.println("BLE ALERT: High gas
levels detected!");
    }
  }
}
```