Smart Home Automation with Alexa: Complete Step-by-Step Guide

---

1. Project Overview

This project enables voice control of home appliances (light, fan, door) using an ESP8266-based NodeMCU and Amazon Alexa. We use the fauxmoESP library to emulate Wemo-compatible devices so Alexa can discover and control them without external cloud dependencies.

Key Capabilities:

- Voice commands: “Alexa, turn on light” / “turn off fan” / “open door”
- Local network operation (no cloud subscriptions)
- Modular design: easily add more devices

---

2. Components & Tools

Component	Description
NodeMCU (ESP8266)	Main microcontroller with Wi-Fi
2-Channel Relay Module	Controls AC/DC loads (light, fan, door actuator)
Amazon Echo Dot/Show	Alexa voice assistant
Jumper Wires	For signal connections
USB Cable & Adapter	Power for NodeMCU
Breadboard (optional)	Prototyping base
Computer with Arduino IDE	Code upload and serial monitor

Software Libraries:

- fauxmoESP (Wemo emulation)
- ESP8266WiFi (Wi-Fi connectivity)
- Arduino core for ESP8266

---

3. Circuit Connections

1. **Power Supply:**
    - Connect NodeMCU VIN (5 V) to USB 5 V via adapter
    - NodeMCU GND ↔ Relay Module GND
2. **Relay Inputs:**
    - Relay IN1 ↔ NodeMCU D1 (GPIO5) — controls **Light**
    - Relay IN2 ↔ NodeMCU D2 (GPIO4) — controls **Fan**
    - Relay IN3 ↔ NodeMCU D3 (GPIO0) — controls **Door actuator**
3. **Relay Outputs:**
    - Relay COM1/NO1 ↔ Light load terminal
    - Relay COM2/NO2 ↔ Fan load terminal
    - Relay COM3/NO3 ↔ Door actuator motor or solenoid
4. **Ground & Neutral:**
    - Ensure relay module GND and NodeMCU GND are common
    - Connect AC neutral through COM pins; switched via NO
5. **Optional Breadboard Layout:**
    - Place relay module on breadboard
    - Route NodeMCU jumper wires neatly

---

4. Software Setup

4.1. Install Arduino IDE & Board Support

1. Open **Arduino IDE**
2. Go to **File > Preferences**
3. Add URL: http://arduino.esp8266.com/stable/package_esp8266com_index.json
4. Open **Tools > Board > Boards Manager**, install **esp8266** package

4.2. Library Installation

- In Arduino IDE: **Sketch > Include Library > Manage Libraries**
- Search and install **fauxmoESP**
- Ensure **ESP8266WiFi** is available (built-in)

4.3. Source Code

Paste the following into a new Arduino sketch:

```
#include <ESP8266WiFi.h>
#include "fauxmoESP.h"

#define RELAY_LIGHT D1
#define RELAY_FAN   D2
#define RELAY_DOOR  D3

const char* ssid     = "YOUR_WIFI_SSID";
const char* password = "YOUR_WIFI_PASSWORD";
fauxmoESP fauxmo;

void setup() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) delay(500);
  fauxmo.createServer(true);
  fauxmo.setPort(80);
  fauxmo.enable(true);
  pinMode(RELAY_LIGHT, OUTPUT);
  pinMode(RELAY_FAN,   OUTPUT);
  pinMode(RELAY_DOOR,  OUTPUT);
  digitalWrite(RELAY_LIGHT, LOW);
  digitalWrite(RELAY_FAN,   LOW);
  digitalWrite(RELAY_DOOR,  LOW);
  fauxmo.addDevice("light");
```

```
 fauxmo.addDevice("fan");
 fauxmo.addDevice("door");
 fauxmo.onSetState([](unsigned char id, const char* name, bool state, unsigned char val) {
  if (!strcmp(name, "light")) digitalWrite(RELAY_LIGHT, state);
  if (!strcmp(name, "fan"))   digitalWrite(RELAY_FAN,   state);
  if (!strcmp(name, "door") && state) {
    digitalWrite(RELAY_DOOR, HIGH);
    delay(1000);
    digitalWrite(RELAY_DOOR, LOW);
  }
 });
}

void loop() {
 fauxmo.handle();
}
```

4.4. Configuration

- Replace YOUR_WIFI_SSID and YOUR_WIFI_PASSWORD with your network credentials
- Verify **Board > NodeMCU 1.0 (ESP-12E Module)** and correct **Port**
- Upload the sketch

---

5. Alexa Integration

1. **Discover Devices:**
    - Open the **Alexa app**
    - Go to **Smart Home > Devices > Discover**
2. Alexa will find **light**, **fan**, and **door** devices
3. Test commands:
    - “Alexa, turn on light”

- “Alexa, turn off fan”
- “Alexa, open door”

---

6. Testing & Troubleshooting

- **No devices found:** Ensure NodeMCU is on same Wi-Fi as Echo
- **Relay not switching:** Check wiring and pin definitions
- **Alexa errors:** Power-cycle Echo and NodeMCU

---

7. Future Enhancements

- Add temperature/humidity sensors and new Alexa intents
- Integrate with home dashboards (MQTT, Home Assistant)
- Use secure sockets (HTTPS) for remote control
