SmartSpace: A Comprehensive System for Home Automation and Monitoring

Product Manual

Contents

1 Introduction

The Smart Home Automation System is designed to provide seamless control and monitoring of home appliances and environmental conditions. This product manual outlines the technical specifications, installation procedures, configuration options, and troubleshooting tips for effective deployment and operation. Our system integrates advanced technology to enhance your home's functionality, security, and energy efficiency.

2 Product Specifications

- **Product Name:** Smart Home Automation System
- **Microcontroller:** ESP32
- **Connectivity:** Wi-Fi (802.11 b/g/n) and Bluetooth
- **Supported Sensors:**
  - DHT11 (Temperature and Humidity)
  - Soil Moisture Sensor
  - Magnetic Reed Switch
  - PIR Motion Sensor
- **Relay Channels:** 4-Channel Relay Module
- **Power Supply:** 5V DC (USB or external power adapter)
- **Operating Temperature:** -20°C to 85°C
- **Dimensions:** 100mm x 70mm x 25mm
- **Weight:** 150g
- **Web Interface:** Responsive design for mobile and desktop access
- **Data Logging:** Historical data storage for temperature, humidity, and moisture levels

3 Components Overview

- **ESP32 Module:** The central processing unit that manages connectivity and controls the system. It supports both Wi-Fi and Bluetooth for flexible communication.
- **DHT11 Sensor:** Monitors temperature and humidity, providing real-time data with a measurement range of 0-50°C and 20-90% humidity.
- **Soil Moisture Sensor:** Measures soil moisture levels to aid in irrigation management, with a range of 0-100% moisture.
- **Magnetic Reed Switch:** Detects open/closed status of doors or windows for security monitoring, with a response time of ¡ 1ms.
- **PIR Motion Sensor:** Detects motion within a range of 5-7 meters, enhancing security features.
- **4-Channel Relay:** Controls up to four appliances remotely via the web interface, rated for 10A at 250V AC.

4 Installation Instructions

4.1 Hardware Installation

- **ESP32 Microcontroller:**
  - Connect the ESP32 to a power source (5V).
  - Ensure it is positioned in a location with good Wi-Fi coverage.
- **DHT11 Sensor:**
  - Connect the DHT11 sensor to GPIO pin 4 of the ESP32.
  - Ensure proper power connections (VCC and GND).
- **Soil Moisture Sensor:**
  - Connect the soil moisture sensor to GPIO pin 32.
  - Verify power connections (VCC and GND).
- **Magnetic Reed Switch:**
  - Connect the magnetic reed switch to GPIO pin 5.

  - Ensure it is properly powered.

- **PIR Motion Sensor:**
  - Connect the PIR motion sensor to GPIO pin 12.
  - Ensure power connections are secure.

- **4-Channel Relay:**
  - Connect the relay module to GPIO pins 26, 27, 14, and 12 for appliance control.

– Ensure proper power connections to the relay module.

Installation Diagram:

4.2 Software Installation

- **Arduino IDE Setup:**
  - Install the Arduino IDE on your computer.
  - Add the ESP32 board to the Arduino IDE by following the instructions on the ESP32 GitHub page.
- **Firmware Upload:**
  - Download the firmware code provided for the Smart Home Automation System.
  - Open the code in Arduino IDE, select the correct board and port, and upload the code to the ESP32.
- **Web Interface Deployment:**
  - Host the web interface files (HTML, CSS, JavaScript) on a local server.
  - Use tools like Ngrok to create a secure tunnel to your local server, making it accessible online.

4.3 Configuration

- **Network Configuration:**
  - Modify the config.h file in the firmware to include your Wi-Fi SSID and password.
  - Ensure the ESP32 is set to connect to your local network.
- **Web Interface Configuration:**

    - Access the web interface by navigating to the assigned IP address in your browser.
    - Configure device settings, including appliance control, sensor thresholds, and notification preferences.
- **User Accounts:**
    - Create user accounts for family members or authorized users to access the system.
    - Set permissions for each user based on their role (admin, user, guest).

5 Technical Details

- **Programming Language:** C/C++ (for ESP32 firmware)
- **Web Technologies:** HTML, CSS, JavaScript (for the web interface)
- **Data Communication Protocol:** HTTP/HTTPS for web communication
- **Database:** SQLite for local data storage and retrieval
- **Security Features:** AES encryption for data transmission and user authentication

6 Troubleshooting

6.1 Common Issues and Solutions

- **Issue:** ESP32 not connecting to Wi-Fi.
    - **Solution:** Check Wi-Fi credentials in the firmware code. Ensure the ESP32 is within range of the Wi-Fi router. Restart the router if necessary.
- **Issue:** Sensors not providing data.
    - **Solution:** Verify sensor connections and ensure they are powered correctly. Check for any physical damage to the sensors.
- **Issue:** Relay not responding to commands.
    - **Solution:** Check the relay wiring and ensure the correct GPIO pins are used in the firmware. Test the relay with a multimeter.
- **Issue:** Web interface not loading.

* **Solution:** Ensure the local server is running and accessible. Check firewall settings that may block access.

6.2 Diagnostic Tools

- **Serial Monitor:** Use the Arduino IDE's Serial Monitor to view debug messages and sensor readings in real-time.
- **Network Scanner:** Utilize network scanning tools to verify the ESP32's IP address and connectivity status.

7 Safety Information

- Ensure all electrical connections are secure to prevent short circuits.
- Do not expose the components to water or extreme temperatures.
- Use only the specified power supply to avoid damage.
- Follow local electrical codes and regulations during installation.

8 Support and Contact Information

For technical support, please contact:

- **Email:** vardan.tuteja2021@vitstudent.ac.in