Abstract This project implements a WiFi-based smart home system using an ESP32 microcontroller. The system integrates motion, temperature, humidity, and light sensors to automate a relay and servo mechanism and sends alerts via Pushover app when motion is detected in Away mode. A web interface allows toggling between Home and Away modes. The system prioritizes real-time motion detection using FreeRTOS task scheduling.

1. Introduction Smart home systems improve security and energy efficiency by automating device control and providing remote notifications. This work focuses on a compact implementation using ESP32 and commonly availab sensors, emphasizing responsiveness and minimal false alerts.

2. System Architecture and Hardware 2.1 System Architecture The system is organized into three main layers to keep things clear and manageable. The User Interface Layer lets users interact with the system through a simple web page and get motion alerts via Pushover messages. This allows users to check the current status and switch between Home and Away modes remotely. The web interface uses HTTP through the ESPAsyncWebServer library.

The Application Logic Layer is where the main decision-making happens. It reads sensor data, checks if any values cross the set thresholds, and controls the devices based on that. Task scheduling is handled using FreeRTOS, where motion detection is the highest priority, temperature and humidity are in the middle, and light sensing is the lowest. At the base is the Hardware Layer, made up of the ESP32 microcontroller connected to various components: a PIR motion sensor (digital input), AHT20 temperature and humidity sensor (I2C), APDS9960 light sensor (I2C), a relay (digital output), and a servo motor (PWM). These are connected using standard protocols like GPIO and I2C.

2.1 Hardware Components 1. ESP32: Central microcontroller with WiFi support. 2. PIR Sensor: Detects human movement. 3. APDS9960: Measures ambient light. 4. AHT20: Reads temperature and humidity. 5. Relay Module: Controls external devices. 6. Servo Motor: Adjusts blind position based on ambient light level.

2.2 Software Components • Arduino IDE with FreeRTOS • ESPAsyncWebServer for web interface • Pushover app for alerting • Servo and sensor libraries (Adafruit, SparkFun)

3. Implementation 3.1 Sensor Integration Other sensors except motion sensor is activated every 5 seconds. Motion sensor is activated every 5 milliseconds when in "Away" mode. Temperature and humidity activate the relay to switch the ventilator, light intensity controls curtain adjustment via a servo. Motion is processed via interrupt and high-priority task. 3.2 Network and Pushover Setup ESP32 connects to local WiFi and communicates with push notification with the Pushover APP. Messages are sent only when motion is detected, and the system is in Away mode. The system uses the Pushover service to send motion detection alerts to the user's mobile device. A user account was created on the Pushover platform, and a new application was registered to obtain the required User Key and API Token. These credentials are used in the ESP32 firmware to authenticate with the Pushover API. When motion is detected and the system is in "Away" mode as described in the table 1, the ESP32 sends an HTTPS POST request to the Pushover API endpoint with the alert message. This integration allows real-time push notifications, ensuring timely alerts directly to the user's phone. 3.3 Web Server Interface Users access a web page hosted on ESP32 to toggle between Home and Away mode. The page also displays current sensor values and relay state.

3.4 FreeRTOS Task Priority and Control Logic Table 1: Task Priorities of The Seosor Tasks (3 = Highest, 1 = Lowest) Tasks Priority Interval PIR Motion Task 3 Every 5 ms Temp/Humidity 2 Every 5 s Light Sensor 1 Every 5 s

Table 2: Control Logic and Thresholds of Detected Environmental Values

Sensor Values **Threshold Action** Motion detected No threshold (digital) Pushover Notification Alert Temp and Humidity t>=25'C or humidity>=60 Relay is on Temp and Humidity T<25'C or humidity<60 Relay is off Light Sensor 100 lux (for strong daylight ~11AM–2PM) Servo is triggered

4. Results 4.1 System Behaviour The system connects to WiFi successfully and displays its local IP address for user access. Motion alerts are sent via Pushover only when the system is in Away mode and actual human motion is detected. The servo motor responds to ambient light levels by adjusting the blinds—closing when light is too strong and opening when it drops. The relay is activated when either the temperature exceeds a set threshold or humidity rises above the defined limit, helping manage indoor climate conditions effectively.