Emergency Alert Button

Motivation

My wife is 75, weak and unstable. I am her 24/7 caregiver but must leave her alone occasionally; groceries mostly. This causes me anxiety and I shortened my trips as much as possible. So I decided to make for her a wearable button that, when activated, would call my cell phone, just in-case she had trouble during my absence.

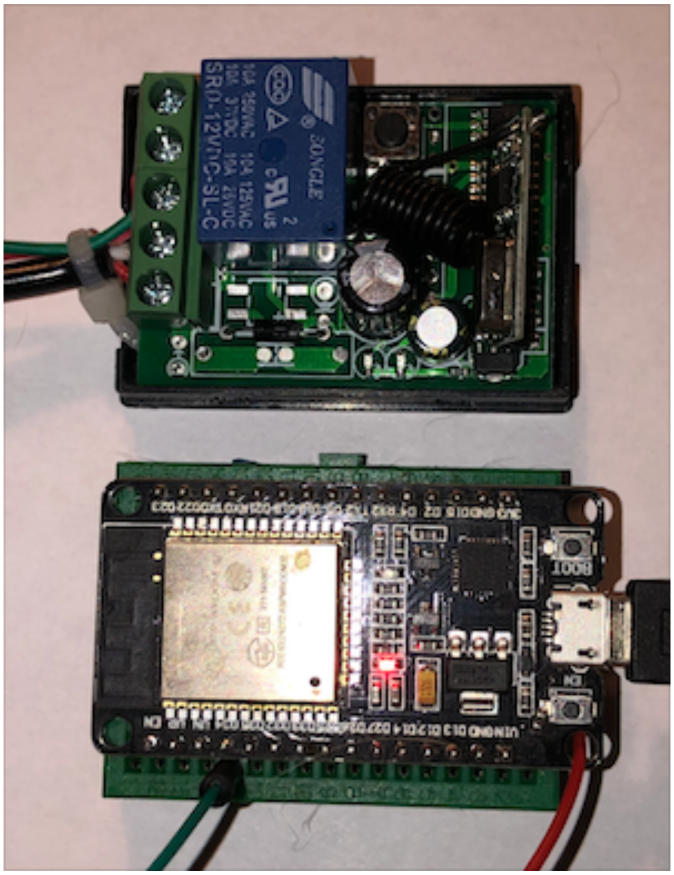
Research led to a few commercial solutions. But they seemed costly with recurring monthly charges. Some other DIY solutions looked bulky and overly complicated, except for one - "The Smallest WiFi Button in the World" https://www.youtube.com/watch?v=lmVK5cGVrpQ. I was inspired by Bitluni's work but I have no access to a 3-D printer. So I decided to use a very simple, cheap hybrid solution using one of my extra ESP32s, although the ESP8266 works just as well. This ESP is the core of a '*base station*' configuration that processes the button press events when received. The physical button and receiver is a commercial, paired 433MHz single channel switch by eMylo;

Compact and inexpensive. The cell phone call is generated via IFTTT's Webhooks app which takes the ESP's button press event packaged as an HTTP GET request and passes it to Woopla, a third party application that makes the actual cell phone call.

IFTTT and Woopla accounts are free and required. Woopla charges my account less than $0.04 per call. I can handle 4¢ per call given that I hope to never get a call at all, except during testing.

I tested using Email, SMS Messaging and even a simple Notification, also free. Each of these methods provide a single beep/vibration upon arrival to my cell phone. But I have bad hearing and mostly do not ware my hearing aids. I opted for the cell phone call with an obnoxious ring tone that does stop ringing/vibrating until I hear it and answer it. Tada! Mission accomplished. Help message received by me.

The eMylo receiver is a compact 433MHz receiver with learning. Button pairing is simple. And the reach is in-house and nearby yard. Perfect for my use case. Only one input from the ESP, 3v3, is switched HIGH/LOW to the ESP 34 input port as an analog value. (Red - 3v3 power and green is the input pin #34.



Truly a simple solution to end my anxiety when out of the house. It is simple to implement with an ESP32 or ESP8266 connected to your local WiFi and internet. It is inexpensive at something about $20 USD.

— Stuff Required ———

- nodeMCU - ESP32s or ESP8266 E12 w/ Arduino Framework
- eMylo R121A 433MHz Switch
- couple of wires
- DC 12V Power Source for the switch, laying around perhaps.