Open Source Smart Home

Zachary Kauffman’s 490H Project

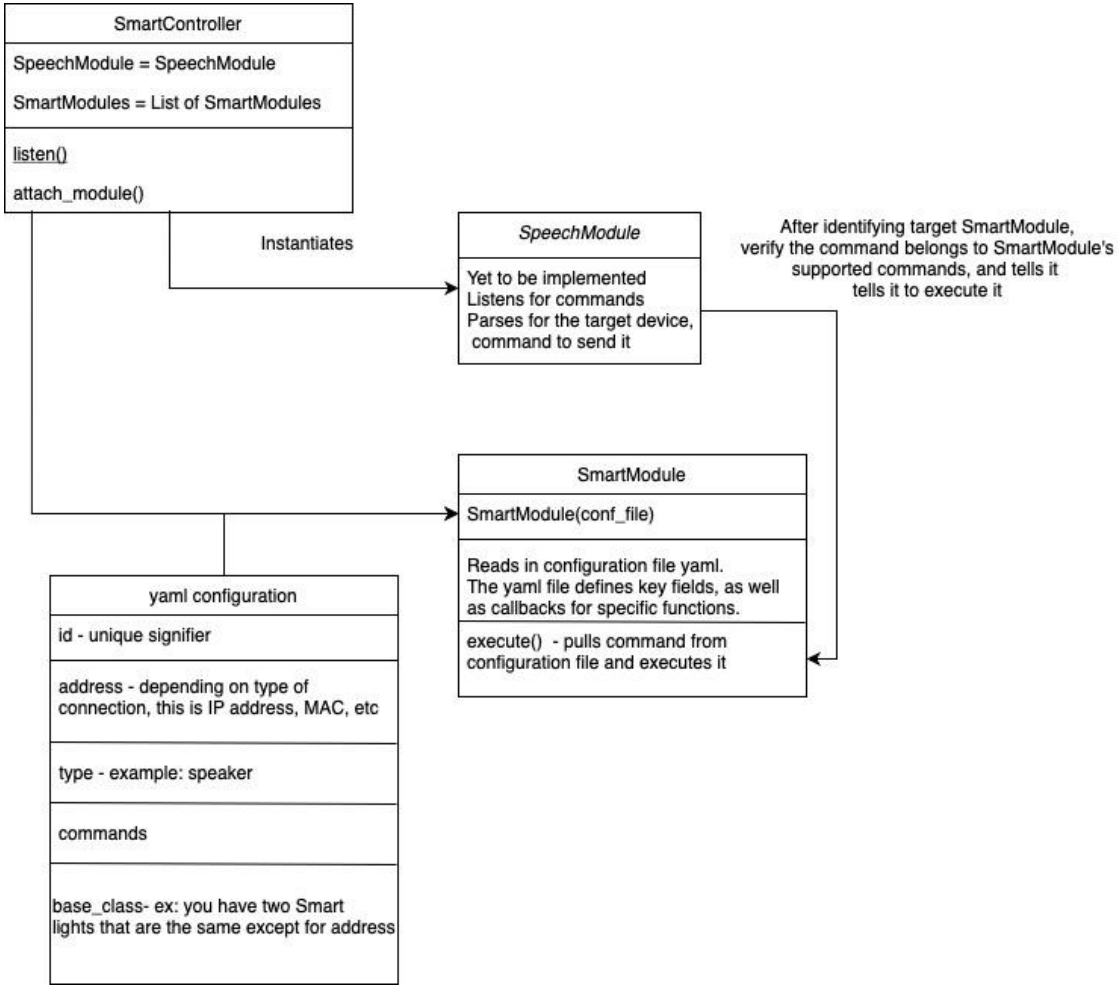
Goal

- Smart Homes are on the rise, from Google Home to Amazon Alexa.
- Using these smart devices hands over insane amounts of data over to big tech companies, regardless of what they may say. See examination by Sogeti (https://labs.sogeti.com/google-home-spying/)
- What if we could replace the spyware offered by a Google Home with an open-source, private alternative without sacrificing the simplicity?

Project

To achieve my project's stated goal, I am going to utilize a raspberry pi connected to various IoT sensors and smart devices. The raspberry pi is going to be running some AI software (something like MyCroft) with which I am going define common actions corresponding to voice commands. This is going to involve diving deep into mycroft's development APIs and writing Python code.

The end project will aim to make pairing new devices easy and offer the simple format of an off-the-shelf smart home device. This will involve a specified, easy to modify, configuration structure that will allow users to define their own devices.


SmartController
SpeechModule = SpeechModule
SmartModules = List of SmartModules
listen()
attach_module()
Instantiates
SpeechModule
Yet to be implemented
Listens for commands
Parses for the target device,
command to send it
After identifying target SmartModule,
verify the command belongs to SmartModule's
supported commands, and tells it
tells it to execute it
SmartModule
SmartModule(conf_file)
Reads in configuration file yaml.
The yaml file defines key fields, as well
as callbacks for specific functions.
execute() - pulls command from
configuration file and executes it
yaml configuration
id - unique signifier
address - depending on type of
connection, this is IP address, MAC, etc
type - example: speaker
commands
base_class- ex: you have two Smart
lights that are the same except for address

Biweekly Progress - Aug 23 - Sept 3

- Basic Control Loop
  - “Listen”
  - Parse each command for a command and a target
    - At its most basic level: know that “turn on bedroom light” means “tell the bedroom light to turn on”
    - Collect valid targets from a list of activated smart modules
- Configuration yaml file structure for smart devices
- Voice Tests
  - Started testing with pre-existing speech recognition code that is NOT open source, this was for preliminary tests of the microphone quality
  - Microphone I already had for the Pi was not sufficient for recognizing commands from somewhere else in the room