ECE 4564 - Group 20

Ethan Brooks ethanb98@vt.edu
Chris Honaker chris98@vt.edu
Kishan Parikh kjp2376@vt.edu

Assignment 3

My API

In this project, our job is to set up two Raspberry Pi's, one as a services Pi and one as the LED Pi. Both of the Pis are running zeroconf as their services advertisement to communicate their ip addresses, port number, and LED colors. The LED commands are communicated between the two Pis through the Python request library. The LED pi has control over the GPIO LEDs and can change the colors to red, blue, green, magenta, cyan, yellow, and white. It also supports Pulse Width Modulation (PWM) to allow for LED dimming. The services Pi can do multiple things. It uses the Canvas API to download a file from Canvas, which is also executed through the Python request library. It is also running a MongoDB database to store three usernames and passwords for HTTP Basic Authentication for the flask webpage. It lastly accepts Rest requests from cURL and the HTTP web browser.

The Canvas part is executed utilizing the Python request library. It is first set up by having the user go into their Canvas page and request a user token. Once the token is received, that token is placed in the servicesKeys.py file. The services.py file calls the key in as a ckey variable, and uses OAuth2 to authenticate the key and access the users Canvas page. Once authenticated, the user creates a URL they wish to go to, with course and file information received from the Python request library from the command the TA's run, and the attached ckey token received from servicesKeys.py. Once the URL is constructed, the program searched that course on Canvas for that specific file, and if found, downloads it to the current directory the services.py program is running out of.

For the cURL and HTTP part of the project, we sent the specific curl command that was specified in the powerpoint. Our project also allows for the data to be changed through the http browser with the different data in the url.

For the python request part of the code, we first had to obtain the ip address through zeroconf advertising. Once we obtained the IP address, we used the ip address along with the user name and password to get all of the different data from the flask server.

The mongoDB is used to store 3 usernames and passwords. These usernames and passwords are added to the server when the services pi starts its script. The server is checked whenever a cURL or HTTP request is processed. At the end of the script, the mongoDB server is cleared.

ECE 4564 - Group 20

Ethan Brooks Chris Honaker Kishan Parikh
ethanb98@vt.edu chris98@vt.edu kjp2376@vt.edu

The led.py file operates the LED RPi, which uses zeroconf to display its username to the Services RPi to be called. Once called by the Services RPi, it reads in the LED color to display, either red, green, blue, yellow, cyan, magenta, or white, reads in the color being in the on or off state, as well as the brightness. The LED color is displayed using the RPi.GPIO Python library, and the brightness is controlled by the RPi.GPIO.PWM Python library which controls the pulse width modulation of the GPIO signal. These are combined to control the GPIO LED connection to user specification.

In conclusion, the program works as expected. The TA's input a specific cURL command that is received by the Services RPi, which then tests HTTP Authentication using the MongoDB server running. Once authorized, the program sorts the command between an LED and a Canvas command. If the command is for Canvas, it goes through the authentication and searching process written above, using the course number and file id provided by the Python request library, finds the correct file, and downloads it to the current directory. If the command is for the LED, the services.py sends the command to the LED RPi using the Python request library, which is found after being broadcasted by Zeroconf, and the LED RPi taks the given color and brightness level from the cURL command, and turns on the LED at that specification. All parameters of our project work as expected, and all specifications are met.