Vanier College

Faculty of Science and Technology

IoT: INTERNET OF THINGS

420-531-VA sec. 00002

Phase 1
CasaConnect

Green Team:

Craig Justin Balibalos

Dinal Patel

Veaceslav Vlas

We, Green Team, certify that this assignment is our own work

I, Craig Justin Balibalos, student ID# 2069192, certify that I have contributed to this deliverable, *Craig Justin Balibalos*

I, Dinal Patel, student ID# 2042827, certify that I have contributed to this deliverable, *Dinal Patel*

I, Veaceslav Vlas, student ID# 1832605, certify that I have contributed to this deliverable, *Veaci*

Table of Contents

Project Description

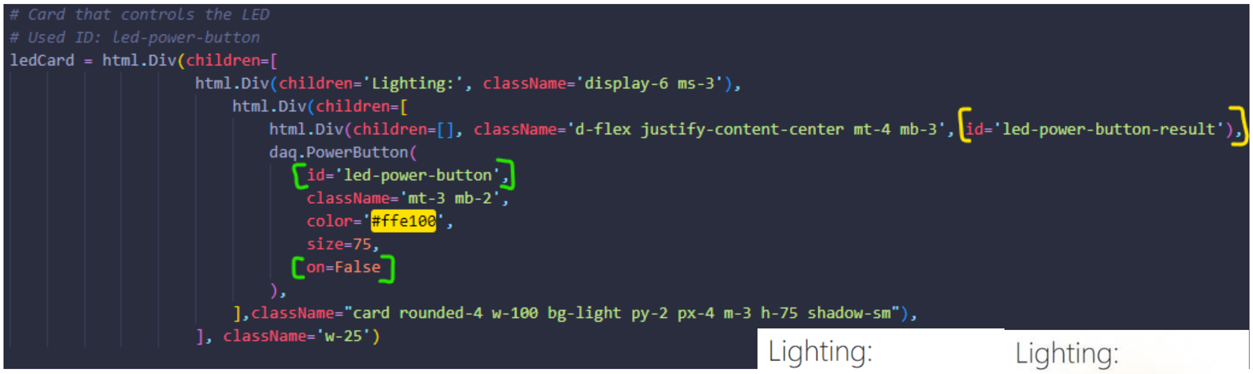
CasaConnect is an IoT-based smart home project as part of our final project in the IoT course at Vanier College. The project consists of multiple phases, with the first phase focused on designing an IoT dashboard structure and data presentation. Using sensors, actuators, motors, and micro-controllers, students capture environmental information, make decisions based on received data, and develop access control and occupancy systems to transfer all data to the cloud or a local server.

In Phase 1, our team is responsible for designing a button as a switch, capture the switch state, transfer the data to an RPi, and present the captured data (switch and LED status) on an IoT dashboard where the user can change the switch status and view the LED status.

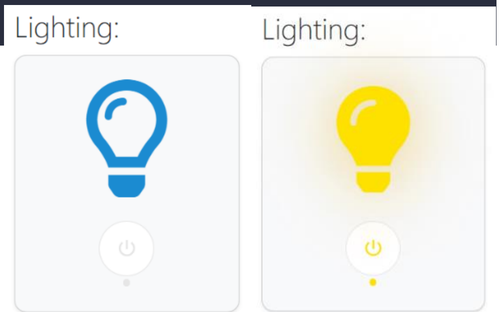
Data Capture

For the CasaConnect project, our group is responsible for designing the data capture component of the IoT dashboard. Our task is to create a button that will serve as a switch, which can capture data in two states: ON and OFF. Dash DAQ will be used as backbone behind the button logic.

UI:



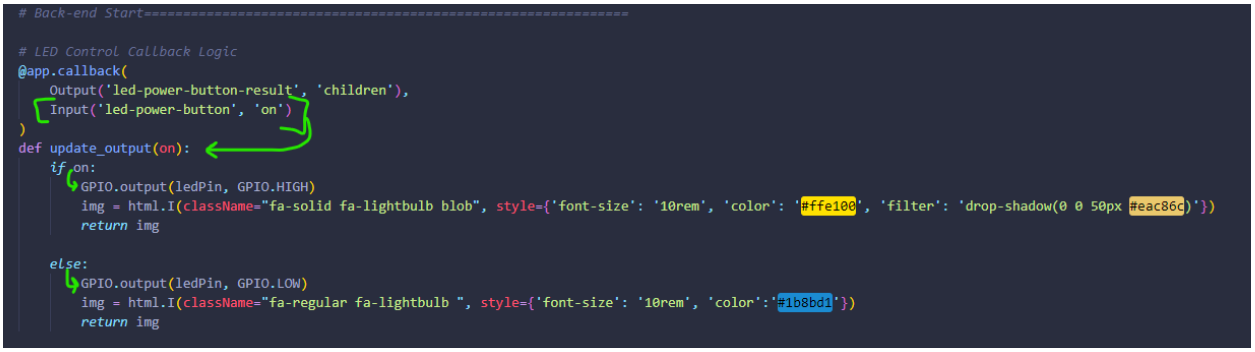
```
# Card that controls the LED
# Used ID: led-power-button
ledCard = html.Div(children=[
                html.Div(children='Lighting:', className='display-6 ms-3'),
                    html.Div(children=[
                        html.Div(children=[], className='d-flex justify-content-center mt-4 mb-3', id='led-power-button-result'),
                        daq.PowerButton(
                            id='led-power-button',
                            className='mt-3 mb-2',
                            color='#ffe100',
                            size=75,
                            on=False
                        ),
                    ],className="card rounded-4 w-100 bg-light py-2 px-4 m-3 h-75 shadow-sm"),
                ], className='w-25')
```



Logic:

Default state of button will be **FALSE**, so it will make sure we have and the correct icon colors showing.

On Switch click the switch will change state to **TRUE** and change color and style of the icon.

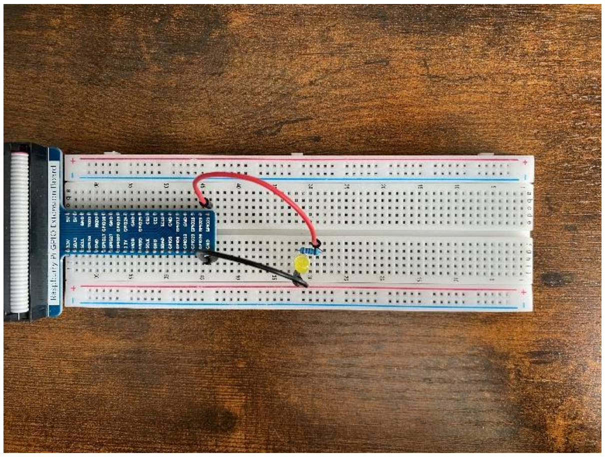


```
# Back-end Start============================================================

# LED Control Callback Logic
@app.callback(
    Output('led-power-button-result', 'children'),
    Input('led-power-button', 'on')
)
def update_output(on):
    if on:
        GPIO.output(ledPin, GPIO.HIGH)
        img = html.I(className="fa-solid fa-lightbulb blob", style={'font-size': '10rem', 'color': '#ffe100', 'filter': 'drop-shadow(0 0 50px #eac86c)'})
        return img

    else:
        GPIO.output(ledPin, GPIO.LOW)
        img = html.I(className="fa-regular fa-lightbulb ", style={'font-size': '10rem', 'color':'#1b8bd1'})
        return img
```

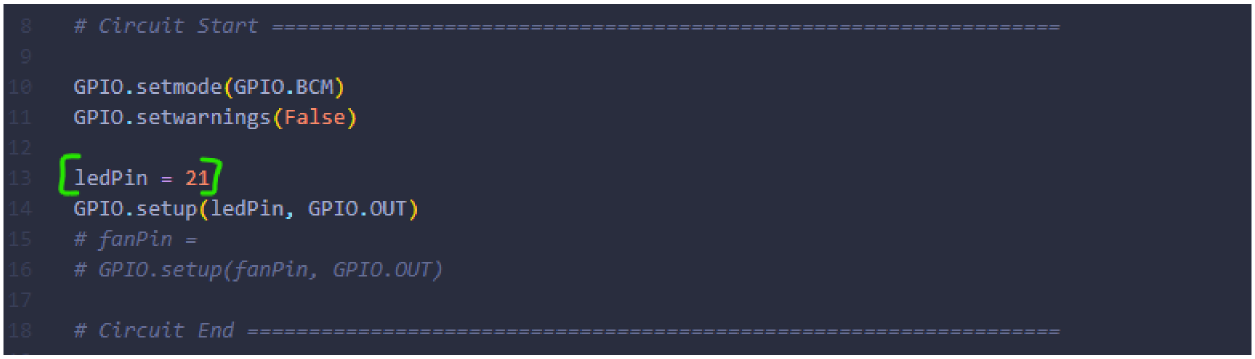
Data Communication

To accomplish this step, we will be using various components such as an LED, a resistor, wires, a breadboard, and a Raspberry-Pi. Once the button is designed and implemented, we will ensure that the captured data (switch state) is transferred to the RPi for further processing and integration into the IoT dashboard.

Circuit:

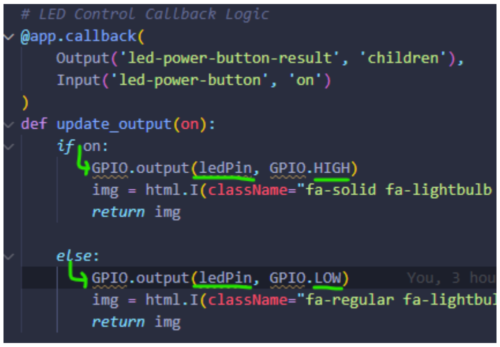


Script for RPi Communication:



```
# Circuit Start ==================================================================

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)

ledPin = 21
GPIO.setup(ledPin, GPIO.OUT)
# fanPin =
# GPIO.setup(fanPin, GPIO.OUT)

# Circuit End ====================================================================
```

Components Control Logic:



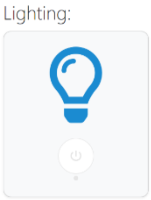
```
# LED Control Callback Logic
@app.callback(
    Output('led-power-button-result', 'children'),
    Input('led-power-button', 'on')
)
def update_output(on):
    if on:
        GPIO.output(ledPin, GPIO.HIGH)
        img = html.I(className="fa-solid fa-lightbulb
        return img

    else:
        GPIO.output(ledPin, GPIO.LOW)        You, 3 hou
        img = html.I(className="fa-regular fa-lightbul
        return img
```

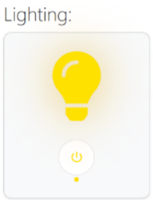
Data Presentation

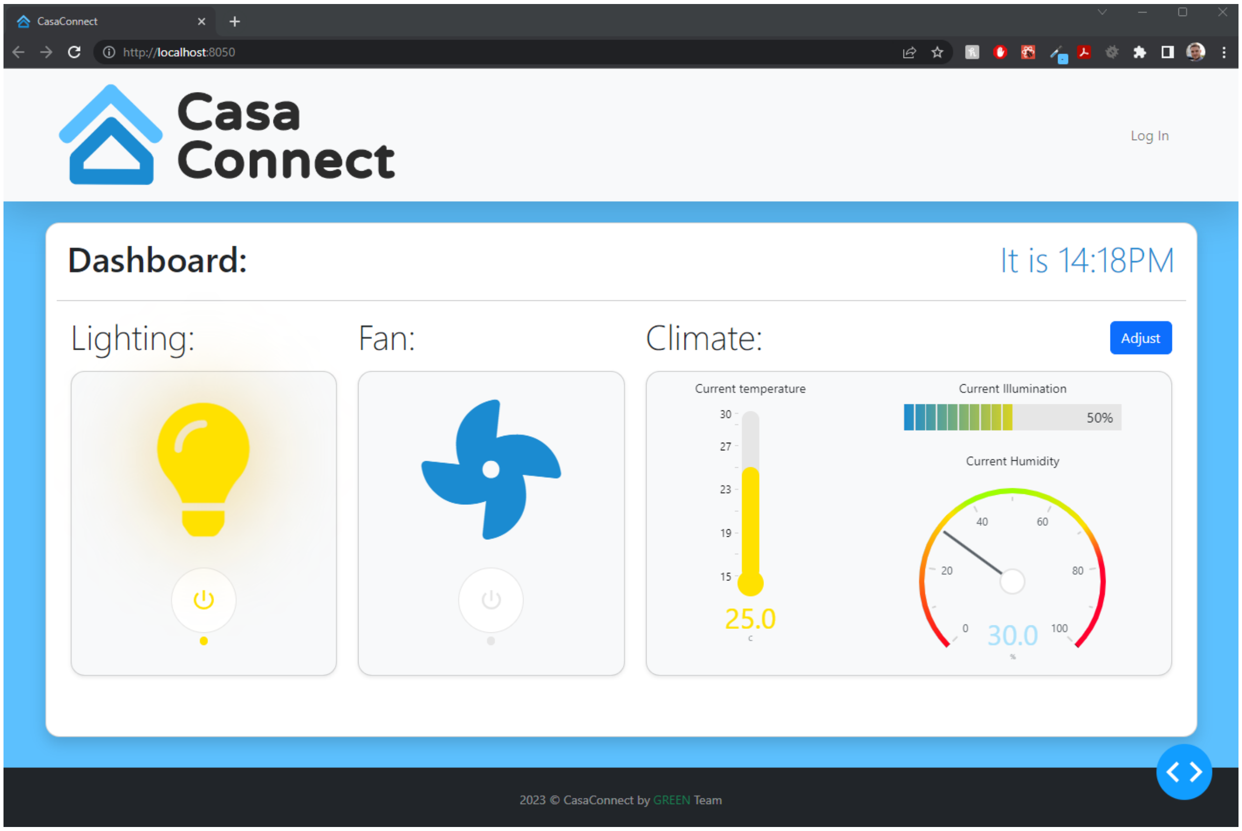
We created an IoT dashboard that displays the captured data, which includes the switch and LED status. The user should be able to change the switch status from ON to OFF (vice versa) and the light icon on the webpage will show the status of the LED.

We ensured that the data is presented in a user-friendly way and that the dashboard is easy to use. Our goal is to create an intuitive and visually appealing interface that will enable users to interact with the smart home system.

We went the extra mile and added some branding (logo, favicon...) and created the mock-up of the future features to be implemented, to gain some time in future phases.







Dashboard:

It is 14:18PM

2023 © CasaConnect by GREEN Team

Links

Dash DAQ
https://dash.plotly.com/dash-daq/

FontAwesome Icons (Free Kit)
https://fontawesome.com/search?o=r&m=free

Green Team's GitHub Repository (Private)
https://github.com/vlasslavic/iot-final-project