Engineering Design 4WBB0 2023-2024			
Self-study assignment report for meeting no:			
Name:	**Student ID:**	**Group number:**	**Hours spent:**
Hazel Lim	1789279	51	4.5 of 17 hrs/wk

Assignment type:

- ☐ A. Administrative tasks
- ☐ B. Building, assembling, manufacturing
- ☐ C. Drawing, sketching, 3D modelling, Concepting
- ☐ D. Modelling, analyzing, performing calculations
- ☒ E. Programming, installing electronics
- ☐ F. Testing (entire design or components)
- ☐ G. Writing or editing the report
- ☐ H. Other:

Description of the assignment (describes tasks and expectations):

Make the interface for the app

Description of the work and the results (should be a clear and complete account of the hours spent):

Over the weekend I have been looking up how to create an app and here's the summary of what I've concluded:

- The app will only need front-end programming, as it will only provide a simple function which directly interacts with users. No back-end is needed.
- As I only know Python for programming language, I'll be using Kivy from the pip package, which is mainly a game development platform but it is suitable to create a mobile app for both android and iOs.
- The app works over wifi, connected with the API key provided from the Arduino board.

I have recognized that the concept of the app is quite similar to that of Philips Hue app, which interacts with the API from the Bridge, the slide bar on the app interacts directly with the brightness of the light. The Hue Bridge is the central hub that connects and controls the light bulbs, which in our case, is similar to the Arduino board. I therefore started to research on Philips Hue app code, with a help of chat gpt, I have studied what the draft of the code should probably look like. Below is the code that I took reference of:

```
import requests

# Replace these values with your own Bridge IP address and API token
bridge_ip = "your_bridge_ip"
api_token = "your_api_token"

# Define the base URL for Hue API requests
base_url = f"http://{bridge_ip}/api/{api_token}"

# Function to send an HTTP GET request to the Hue Bridge
def get_request(endpoint):
    url = f"{base_url}/{endpoint}"
    response = requests.get(url)
    return response.json()

# Function to send an HTTP PUT request to the Hue Bridge
def put_request(endpoint, data):
    url = f"{base_url}/{endpoint}"
    response = requests.put(url, json=data)
    return response.json()

# Example 1: Get the state of a specific light
light_id = 1
light_state = get_request(f"lights/{light_id}")
print(f"State of Light {light_id}: {light_state}")

# Example 2: Turn a light on
```

```
light_id = 2

light_state_on = {"on": True}

response = put_request(f"lights/{light_id}/state", light_state_on)

print(f"Turned on Light {light_id}: {response}")


# Example 3: Change the brightness and color of a light

light_id = 3

light_state_changes = {

   "bri": 150,     # Brightness (0-254)

   "hue": 25000,   # Hue (0-65535)

   "sat": 200,     # Saturation (0-254)

}

response = put_request(f"lights/{light_id}/state", light_state_changes)

print(f"Changed Light {light_id} state: {response}")
```

From this, I will need an IP address of the Arduino board, API key and a base URL to work on. The rest of the code I will need to add will be the visualization of the app.