Senior Design
ENG EC 463

Team 18 Final Prototype Testing Report

To: Professor Pisano

From: Greener Living

Team: 18

Date: 4/8/21

Subject: Team 18 Final Prototype Testing Report

1.0 Introduction

Compared to our second prototype testing, we were able to implement our hardware components together. We were also able to implement our hardware locally so that the web application can interact with the sensors and database. In our demo, we showed our sensors writing data to our database in real time, hosted by our Raspberry Pi. We also showed the website's functions, which include display graphs and selectable time scale, as well as other additional features.

2.0 Equipment

2.1.0 Hardware

2.1.1 Raspberry Pi Zero W

It records the measurements taken by the temperature and humidity sensor and the energy plug in an Influx database, and visualizes them locally using Grafana. All data recorded is updated in real time and that is reflected in the graph visualization.

2.1.2 DHT22 Temperature and Humidity Sensor

It measures the surrounding temperature with ±0.5°C accuracy and humidity with 2-5% accuracy. It is connected to a Raspberry Pi Zero W which records the measurements taken.

2.1.3 Etekcity ESW15 WiFi Energy Monitoring Plug

It measures the power usage of any device connected to it at an outlet. It is used to measure the power consumption of any device the client wants to know more about

2.1.4 IoTaWatt Open WiFi Energy Monitor

It measures a circuit's wattage, voltage, etc. using a current transformer. It is used to measure the energy consumption of appliances connected to a circuit breaker.

2.1.5 ET200 Combination AC Line Splitter and GFCI Receptacle Tester

It allows appliances with plugs to be measured using a current transformer, It is used to test the IoTaWatt due to lack of circuit breaker to install into.

2.2.0 Software:

2.2.1 Python scripts

First collects temperature and humidity data, and stores them in an influx database in the raspberry pi. The second does the same but with the energy monitoring plug.

2.2.2 InfluxDB

Database server that store information gathered from our sensors

2.2.2 Grafana

Allows us to visualize data from our database to be rendered onto our web application

2.2.2 Web Application

A React project created using Javascript, HTML, and CSS, which is the platform where our client can view all the information.

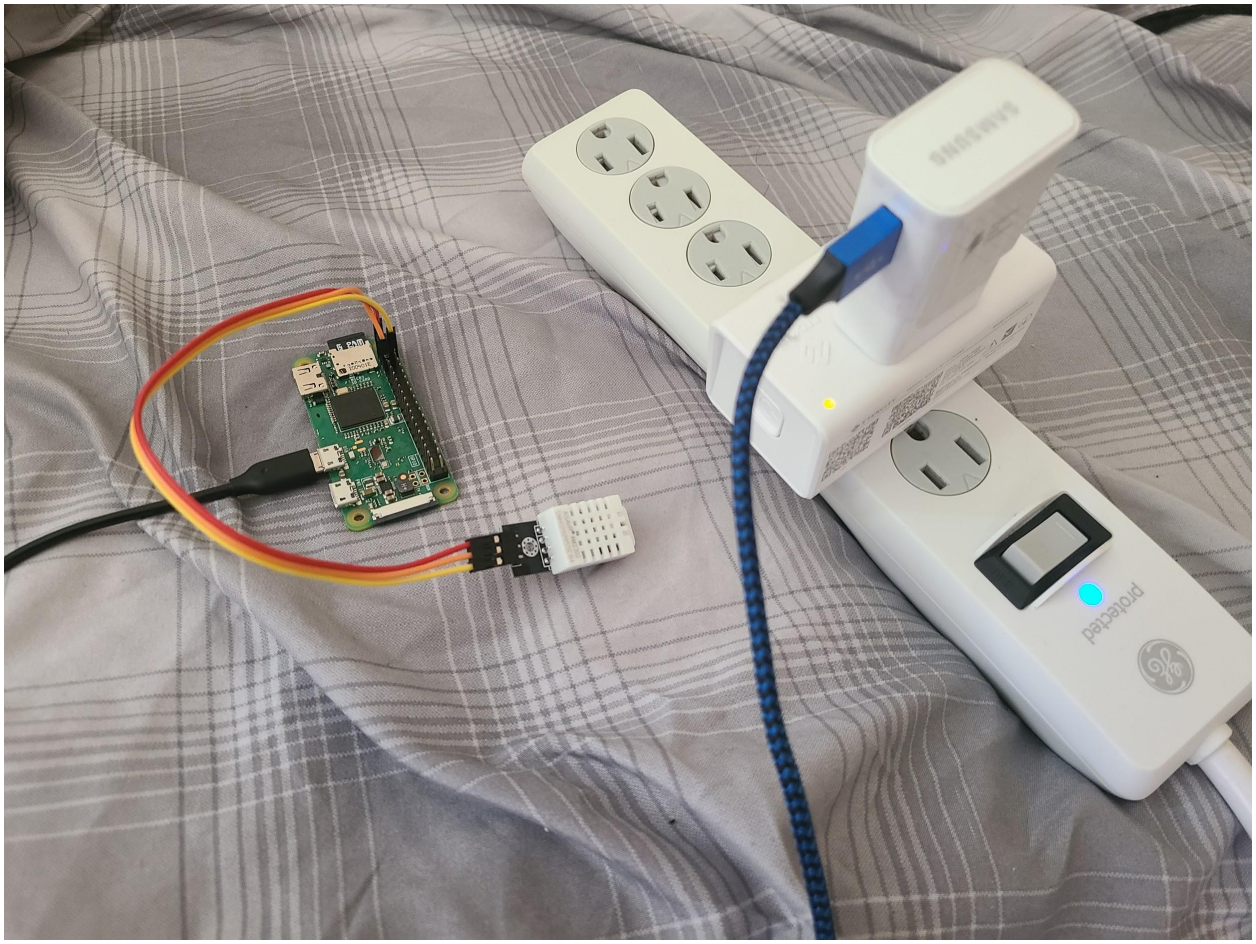

Figure 1: The DHT22 sensor connected to the Raspberry Pi Zero W and the ESW15 energy plug connected to a phone charger.

3.0 Setup

1. The Raspberry Pi Zero W and the energy plug are connected to the same WiFi network.
2. The DHT22 sensor is properly connected to the Raspberry Pi. The energy plug is connected to a device that is actively consuming power.
3. Two scripts, which store the data measured by each of the two devices in an Influx database, are ready to use.
4. The local instances of InfluxDB and Grafana are enabled on the Raspberry pi. The local instance of Grafana is checked to see that all previously created dashboards and graphs are in order.
5. The IoTaWatt is connected to the WiFi network and writes data to the correct Raspberry Pi. The database is then checked to make sure data from the sensors is being received.

4.0 Testing Procedure

1. Run the two scripts and show that they are taking measurements live using print statements. Show that changes in the environment (like breathing on the DHT22 sensor) can result in a change in the measurements (the humidity recorded increases).
2. Run a query on the two databases of the two sensor devices to show that collected data is being stored properly.
3. Access Grafana's local interface
    a. Show the integration of the local database and Grafana
    b. Show that the power, temperature and humidity are being collected and the graphs are updating in real time
    c. Show customization of the panels
    d. Show time range manipulations
4. Show website designs and current state.
    a. Show the different tabs: The home tab which is where the user starts. The room tab which displays the temperature and humidity tab. The cost tab which shows the cost of the energy measured. And the devices tab which allows the user to select a device and see the graphs of energy usage.
    b. Show dropdown menu and selections of each device.
    c. Show button group with time scales (hourly, daily, weekly)
    d. Show correct graphs being displayed depending on the room/appliance and the time scale
    e. Show data is updated in real time

5.0 Measurements Taken

1. Data collected by the two sensor devices is properly stored in a database.
2. Data collected by the IoTaWatt is properly stored in a database.
3. Breathing hot air on the DHT22 sensor gives higher temperature humidity values showing that it is recording data and responding properly.
4. The energy plug is taking power usage in real time and the recorded power is shown.
5. The local instance of Grafana is displaying the stored data (temperature, humidity, power) in three separate graphs, which get updated in real time to show the current measurements being taken. Additionally, these graphs can be displayed on different timescales such as hourly, daily and weekly.
6. The website layout was responsive with the different tabs operational. The devices tab was shown to have a functional drop down menu that allowed the user to swap between devices, and the energy usage graph changes the data displayed accordingly.

7. Grafana was shown to be integrated onto the website, and graphs were shown to update in real time to reflect the current measurements being taken.
8. The cost tab was shown to calculate the cost of energy that was used by the devices being measured with the IotaWatt and energy plugs.

6.0 Conclusion

Our final prototype testing was successful. The sensors were able to be properly stored in the database. We were also able to show real time functionality in regards to the web application and the graphs. Overall, we were able to implement our hardware and software components together nicely.

From our second prototype testing, we wanted to finalize our hardware so that we could work on completing the web application. We were able to set up all of our sensors and servers on one local network, as well as add the core functionalities of our web application.

Unfortunately, due to a lack of communication from our client, we were forced to focus on one utility (electricity) instead of the intended three (electricity, gas, solar).

Overall, we were able to put together a web application that shows the energy usage of appliances and implement a time scale - the main feature that our client had wanted. Looking forward, we hope to add/finish additional features that would complement the information displayed, such as a notification center for unusual activities, a section about carbon emission, and a quick overview of all the information combined on a home page.