EcoSwitch – An Autonomous FCU Maintenance System

Jiawei Liao, Keven DeOliveira, Michelle Thevenin, Samarah Uriarte, and Michael Harkess

Abstract —Manual FCUs in brownstones lead to uncomfortably warm rooms, causing occupants to either turn them off or open a window in an attempt to balance out the temperature. The former is inconvenient and the latter results in a waste of energy. A cost-effective solution that offers multi-party accessibility is required to address this problem. To facilitate a more enjoyable housing experience and provide a centralized maintenance application to Boston University housing administrators, we propose EcoSwitch, an autonomous FCU dial mount with integrated applications which will allow both residents and managers to monitor and adjust FCU temperatures. In addition, it will query local weather information to maintain accurate performance and store relevant information in a highly accessible database, which will allow FCU functionality to be evaluated. EcoSwitch has been optimized for the management of Boston University's campus life and will establish applications with functions that are appropriate for both administrators and students.

Index Terms — Autonomous system, Design, Real-time and embedded systems, Energy efficiency and conservation

● *Jiawei Liao EE' 22. E-mail: liaophie@bu.edu*
● *Keven DeOliveira CE' 22. E-mail: kevend@bu.edu*
● *Michelle Thevenin EE' 22. E-mail: thevenin@bu.edu*
● *Samarah Uriarte CE' 22. E-mail: samcu@bu.edu*
● *Michael Harkess CE' 22. E-mail: mharkess@bu.edu*

1 NEED FOR THIS PROJECT

Many brownstone apartments on Boston University's campus still use FCUs for indoor heating in the winter. This process is not controlled by the occupants in any way. Once FCUs are turned on for the winter, they are often left on for the remainder of the season. This situation leads to an uncomfortably warm room. Occupants typically aim to solve this problem by shutting the FCU off altogether or opening a window to balance the temperature on the inside. Opening a window while an FCU is running is a direct waste of energy and valuable resources.

Building administrators want to monitor room temperatures on a weekly basis to ensure that the FCUs are working properly. At the same time, they want the FCUs to work more efficiently so that energy is not wasted. Brownstone residents want an easy way to control their room temperatures so that it stays within a desirable range. Both students and administrators want a solution that is specific to Boston University so that it is cost effective and optimized for dormitories.

Boston University has a zero waste plan that aims to eliminate all waste by 2030. The plan is to redesign systems, reuse resources, change the culture, and develop sustainable markets. With EcoSwitch, we aim to automate the FCU control system in an effort to be more conscious of Boston University's sustainability goals. Our design supports BU's plan by providing a solution to the waste of energy that occurs as a result of using FCUs.

EcoSwitch will address this issue by allowing students to adjust the temperature of their dormitories according to their liking. EcoSwitch will automatically adjust the FCU to the desired temperature. EcoSwitch will be designed for long-term use and will be friendly to the environment.

2 PROBLEM STATEMENT & DELIVERABLES

Our proposed system design consists of several key components that all contribute to the overall maintenance of brownstone dormitory temperature and humidity levels.

2.1 Problem Statement

By integrating autonomous technology with simple applications for customizable temperature monitoring and management, EcoSwitch solves the problems of students facing uncomfortable dorm conditions and administrators having difficulty in efficiently reducing FCU energy waste. With the goal of achieving zero waste by 2030 [1] and a more general desire to increase the quality of student life on campus, our solution has been optimized for Boston University staff and students to ensure both parties' needs are being met.

The high-level solution that forms our system design is partitioned into two separate sets of user and application interfaces and a centralized server that will process the data received from each EcoSwitch device. This solution addresses the different functionalities students and administrators require. Options for reading in and displaying temperature and humidity levels, filtered between specific individual students' residencies to all of Boston University's brownstones, will be available and chosen depending on the interface. Both parties will be able to control dorm temperatures as well; however, administrators will have the ability to override student input to ensure energy waste is minimized.

Both the website and mobile app clients will use sensor information and functions that control the EcoSwitch device to achieve FCU automation and remote adjustment capabilities. To reduce computational redundancies and bandwidth requirements, the website and mobile app will connect to a single server that will have data streamed in and distributed to the appropriate user. After user responses reach the server, it will deliver back calculated temperatures after processing these inputs. Each EcoSwitch device that will be registered with a dormitory will connect to this same server; this architecture allows all EcoSwitch devices to be linked to a centralized hub for ease of data distribution. Furthermore, a database will be maintained that contains temperature and humidity levels, allowing administrators to access past information and assess FCU functionality. The EcoSwitch administrator website will also produce graphs that plot dorm conditions for up to 3 days. Local weather information will be queried for through an API and used to determine individual EcoSwitch device instructions, a feature that guarantees dorm temperatures that more accurately match user input.

The EcoSwitch device will be a compact, battery-powered apparatus that reads in temperature and humidity information and relays it to the centralized server. Comprising a temperature and humidity sensor, microcontroller, and rotary actuator, the simple and cost-effective circuit design fulfills low-cost specifications. Our hardware design also includes an inexpensive attachable component that will fasten to the actuator and can easily be swapped for a new attachment if it becomes damaged. The entire device will be subject to safety constraints which make it suitable for the environment near the FCU's surface.

The EcoSwitch devices will be programmed to collect data every 5 minutes; thus, there is no network strain or reliance on consistent WiFi connection. Computations will be performed in the central server, iterating through each registered EcoSwitch device after taking input from either administrators or students. The calculations will depend on local weather information that will be received through an API and combined with sensor data, as well as any experiment-based observations that are developed on the relationship between FCU surface temperature and the individual dormitory conditions. The EcoSwitch devices will then receive the results of these computations and execute instructions that will turn the actuator accordingly, ensuring the desired temperatures in the dormitories are reached.

2.2 Deliverables

A safely mountable device capable of:
1. Processing data and outputting information through a microcontroller
2. Sensing temperature and humidity levels
3. Attaching to and spinning a swappable attachment that will turn radiator dial knob
4. Withstanding heat from attached/nearby radiator and avoid tampering
5. Consuming low energy from a rechargeable battery

A cross-platform application capable of:
1. Periodically reporting sensed temperature and humidity levels to user
2. Allowing user to set desired range for temperature

An administration website capable of:
1. Monitoring the status of all devices connected to the Boston University housing and network
2. Overriding signals to individual devices
3. Logging and storing temperature and humidity data

3 Visualization

In this section, we will describe the three main aspects of our system that can be meaningfully visualized.

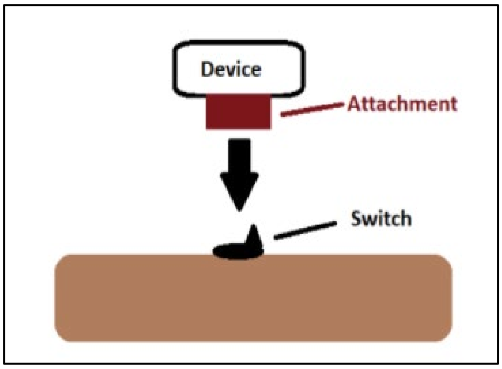

Fig. 1. This is a general depiction of the device in question. All the components, such as the microcontroller and sensors, will be housed in a small shell that will act as the "body" of the device. The attachment will be detachable and swappable for easy maintenance and replaceability.

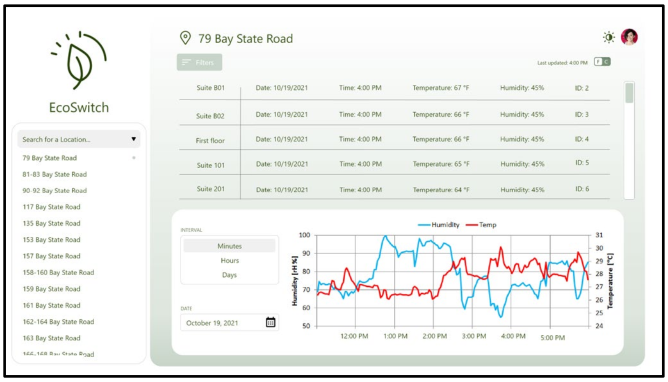

Fig. 2. A mockup of the administrator website UI. This website will allow admins to view past and current temperatures across every room. It will also provide the ability to override student input in dorms as needed.

Fig. 3. The mobile application that students can log into and monitor the temperature of their dormitories with. Users can also adjust the desired temperature range and notify their roommates of any changes.

4 Competing Technologies

Companies around the world have developed mechanisms to regulate heating systems for the purpose of reducing energy waste and providing a convenient method to alter the temperature of a room. The most significant products that provide such a service are the Bosch Smart Radiator, the Eve Thermo, and the Haverland Smartwave.

4.1 Bosch Smart Radiator

The UK company Bosch, developed a TRV (thermostatic radiator valve) that can be controlled through an internet connection. This TRV has the capability of being controlled with a voice assistant like Siri and can also be controlled by the smart home application that Bosch provides for their smart home products. This TRV also has buttons so that the temperature of the radiator can be controlled manually, and it carries a display that shows the temperature that the TRV is set to. There are also other features that are advertised such as a battery warning, a child safety lock, and a whisper mode. Bosch sells this device for approximately £55 [3]. This device is interoperable with other Bosch smart home devices such as their thermostats and door/window sensors. At that price point, Bosch not only provides a method to change the temperature from a web interface but an entire ecosystem. This ecosystem maintains the temperature automatically based on the information gathered by other Bosch smart home products and the user's desired temperature.

4.2 Eve Thermo

The German company Eve has also developed a TRV that is capable of Internet of Things (IoT) functionality. Their product, marketed as "Eve Thermo," is similar to the Bosch Smart Radiator. The Eve Thermo has a mobile application that is provided by Eve; however, it is only available on iOS. The TRV itself has an LED display that shows the set temperature, as well as tactile buttons that allow users to change the temperature manually, without the use of an internet connection. This device is being sold for £60 [4]. The Eve Thermo is compatible with other products sold by Eve, such as the Eve Door & Window, through the mobile application Eve provides. To achieve compatibility, all of Eve's devices support Bluetooth and Thread for communication. For this compatibility to exist, the user must have an Apple TV (4th generation) or the HomePod mini, along with a mobile iOS device equipped with the Eve app. The mobile application provides a way of not only changing the temperature but also setting schedules so that a user can set different temperatures for different days in advance [5]. Thus, similarly to the Bosch Smart Radiator, Eve provides an ecosystem for their TRV, but that ecosystem is more limited due to the devices that can access the ecosystem. It provides a way for the user to plan out what they want for the temperature days in advance.

4.3 Haverland Smartwave

UK company Haverland have created an electric heater with similar "smart" capabilities to both the Eve Thermo and the Bosch Smart radiator. Their product named the "Haverland Smartwave" is a thin electric radiator (75 mm thick) that can be mounted on a wall. On the device, there are buttons (to change the temperature) and a LED display for the current temperature of the room. An important feature is that the Smartwave can detect sharp changes in temperature without the use of other devices, allowing it to change the heat output automatically. By default, the Smartwave does not have IoT capability and the user must purchase a "Smartbox" accessory separately to have IoT features [6]. However once the SmartBox accessory is installed, the Smartwave can be controlled on a mobile app (iOS and Android) and with Amazon Alexa. Currently, the Smartwave is being sold for £300 for the cheapest model, not including the SmartBox which will cost an additional £55 [7]. The mobile app for the optional SmartBox allows the user to schedule specific heating for different days, similar to the Eve Thermo. As an extension the app has a "learning" feature, where the app adjusts to a user's desired temperature based on previous usage [8]. The app also allows the user to view power consumption in correlation with the room temperature over a period of time (day, month and year). Our product will also record temperature data over a set time frame for a room although not on such a scale and detail as the Smartwave.

4.4 EcoSwitch

Our product is similar to our competitors in that we will allow the temperature of a room to be determined by user input on a web interface. There will also be data collection for the temperature of a room over a predetermined time period similar to the Smartwave and a low battery warning to alert the user like the Bosch Smart Radiator. While our competitor's products require a specific heating system to be integrated into (e.g radiator heating), our product will be compatible with most heating systems that have a control unit. To achieve this, our product will be modular, with an attachment that will change depending on the design of the heating system's control unit.

5 Engineering Requirements

This section breaks down the engineering requirements per associated subsystem.

5.1 Hardware

1. The device must be able to detect the room temperature and humidity levels with the DHT11 digital sensor whose range for relative humidity is 20-80%; for temperature is 0-50°C with ±2°C accuracy.[2]

2. The ESP32 microprocessor which can use the Arduino IDE as the development environment must be able to read and stream data to the clients. Also, the program is stored here and it determines whether to turn on/off the actuator that connects to a programmable motor.
3. The swappable attachment should be in plastic and be able to cap the whole FCU dial nicely. Also, it should be attached with an actuator and allow a motor to give momentum to it. Moreover, this attachment must be easily removed and reattached to another FCU dial.
4. The device must support the removal of at least one sensor and its replacement with another sensor that uses the same communication protocol.
5. The actuator is the 28BYJ-48 Stepper Motor with ULN2003 driver which has 300 gf. cm pull in torque which is enough for turning the FCU dial back and forth.[9] The actuator must be installed inside the device and firmly attached to the attachment. Also, it is compatible with the ESP32 microprocessor. This motor set only costs $10, and easy installation makes the installation fee cheap.

5.2 Software

1. The device must be able to communicate with the user app via wifi connection or Bluetooth connection if wifi is disabled.
2. The cross-platform mobile app or full-stack web app must show the current room temperature and humidity. Also, it must allow the user to decide the temperature in a specific range.
3. An administrator website is needed to check the status of each device for maintenance purposes. Also, if the user changes the room temperature too many times in a short time, a warning message must be sent out to the user. If the user continues this kind of action, the administrator website must be able to overwrite the device and send out a report to the user.
4. The program designed in the microprocessor must be able to read the collected data and compare it with the database to set up a base value for maintaining the room temperature in a specific range.

5.3 Data Coll ection and Transmission

1. The digital sensor must send the room temperature and humidity data every five minutes. Also, it transfers the collected data to the microprocessor for comparison with the data in the database.
2. The input/request data of the app must be transferred through API via Node.js and React Native that acts as back end and front end servers respectively. The SQL database transfers data through servers to the app if that is a request or stored back to the database if there is an update.
3. The SQL database must be updated every day so that the the program can update its base value for maintaining room temperature.
4. The device must be able to transmit data wirelessly with a range of at least 1km and data rate of at least 250 kbps.

5.4 Power, cost and safety

1. The cost of this device should not exceed $100 because most of the compatible products are around that range. Also, its size should be similar to an FCU dial and light to prevent the dial from coming off.
2. The device should be powered by rechargeable batteries and the app should remind users of the battery level.
3. The device should be safe and stable (e.g., tamper-proof), and it should mount near the FCU.

6 Appendix A References

[1] *Zero waste plan*. Zero Waste Plan | Sustainability. (n.d.). Retrieved October 22, 2021, from https://www.bu.edu/sustainability/projects/zero-waste-plan/.

[2] Industries, A. (n.d.). *DHT11 basic temperature-humidity sensor + extras*. adafruit industries blog RSS. Retrieved October 22, 2021, from https://www.adafruit.com/product/386.

[3] *Smart Radiator Thermostat*. Bosch Smart Home. (n.d.). Retrieved October 22, 2021, from https://www.bosch-smarthome.com/uk/en/products/devices/radiator-thermostat/.

[4] *Eve Thermo*. evehome.com. (n.d.). Retrieved October 22, 2021, from https://www.evehome.com/en/eve-thermo.

[5] Rowley, E. (2021, January 7). *The best smart radiator valves 2021*. Tech Advisor. Retrieved October 22, 2021, from https://www.techadvisor.com/test-centre/digital-home/best-smart-radiator-valves-3693855/.

[6] *Haverland SmartWave self-programming electric radiators*. Haverland SmartWave Self-Programming Electric Radiator. (n.d.). Retrieved October 22, 2021, from https://www.electricradiatorsdirect.co.uk/haverland-smart-wave-electric-radiators-white/.

[7] *Haverland SmartWave Electric Radiators - SmartBox WIFI Hub*. Haverland SmartWave Electric Radiator SmartBox Controller. (n.d.). Retrieved October 22, 2021, from https://www.electricradiatorsdirect.co.uk/haverland-smart-box-wifi-controller/.

[8] *Haverland app - step by step - electricradiatorsdirect.co.uk*. (n.d.). Retrieved October 22, 2021, from https://www.electricradiatorsdirect.co.uk/media/pdf/Smartbox_Haverland_App.pdf?_t=1536576385.

[9] Components101 - 28BYJ-48 - 5V Stepper Motor.Retrieved October 22, 2021, from https://components101.com/motors/28byj-48-stepper-motor

Client Signature: