Abstract—With the constant developments in smart home technology over the last decade, there is an increased need for innovations in the field. Popular technology companies such as Google and Amazon are working diligently to improve upon their existing products, such as the Google Nest and Amazon Echo, to make them more user friendly and capable of greater tasks. One underdeveloped feature in such existing technologies is the integration of multiple systems, or the ability for different home systems to work together to take on unified functions. This project’s goal was to create a system that provided a feasible method of integration for smart thermostats and smart locks. Using a Raspberry Pi and a platform called If This Then That, a system was created to enhance the Honeywell Total Connect thermostat, while also incorporating it with an improved custom smart lock system. This system utilizes a magnetic reed switch and servo motor to create a mechanism that locks and unlocks a deadbolt. A push-button is also located inside the house for manual locking and unlocking capabilities. Additionally, it uses Bluetooth connection to detect which user is entering or leaving the residence. When the door is unlocked, the smart thermostat computes the average preferred temperatures of all the residents in the house and sets a permanent hold on that average in shared living spaces. This project aims to emphasize efficiency in the home environment.

I. INTRODUCTION

Over the past decade, smart home technology has become increasingly pervasive throughout American households. Smart home technology refers to the use of internet-connected devices in a home environment that enable the remote management of appliances and systems, such as lighting, climate, and entertainment systems. These systems allow users to maintain consolidated control over their connected home devices through smartphones and other remote technology. Some devices have accessibility that extends to voice control and artificial intelligence. The development and proliferation of such technology gives consumers more control over their devices while also substantially reducing human efforts. Additionally, smart technology often saves money and energy and certain smart security devices also provide for stronger home safety [1]. Most of this smart technology is relatively new, but the idea of home automation, the automation of processes in the house and integration of smart devices, dates back several decades.

The world’s first introduction to automating was through X10 home automation in the mid-1970s. X10 was a communication protocol for controlling technology in the house using power lines and radio bursts. However, it was not connected to the internet and could not necessarily be considered “smart technology”, even though it introduced the ability to control parts of the home like light switches and main outlets manually or by a remote control [2]. The first smart home products available on the market still could only carry out simple tasks such as turning devices on and off or setting the temperature. However, over time, these systems have become more widespread and offer a much greater amount of utility. The usage of smart home devices has increased significantly over the past few years. In 2018, approximately 34.8 million homes in the United States had some form of smart home technology. However, estimates predict that by the year 2023 that number will more than double, reaching 70.6 million, or 53.9% of the houses in the country [3]. Not only is there a significant demand for smart home technology, but people are working to create better versions of smart home technology that make home living more convenient.

Currently, standard mechanical locks with keys are not always reliable. Opening these basic locks always require residents to carry a key, which also creates the risk of the key being stolen and the house being broken into. These shortcomings make the advent of smart lock technology important for
homeowners conscious about their home security. Smart locks provide a digital platform that allow users to control their lock from their smartphone, allowing for easier access to the house and relying on digital security. This capability also reduces the concern about losing keys. While the availability of a manual lock and key should not be eliminated because technology has its own vulnerabilities, smart locks still provide value and functionality.

Similarly, basic thermostats lack the desired factor of convenience. With standard thermostats, changing the temperature of the house requires walking to the device on the wall and manually adjusting it. However, this process can be improved through the use of smart thermostats that enable both remote and local monitoring. The thermostat can also follow a pre-programmed schedule that can be modified at any time while also allowing users to set their temperature before arriving home.

Smart locks and smart thermostats are two prime examples of smart home technology that make home living much easier, and exemplify why it is important to keep innovating to improve homeowners’ living experience.

II. BACKGROUND

A. Existing Technologies

The scope of this project was to design a system that integrates multiple existing smart home technologies and improves functionality. Technologies for both smart locks and smart thermostats exist but are sold individually and fail to offer a single unified experience for the user. For instance, Google and Honeywell have released the Google Nest and Honeywell Total Connect Comfort smart thermostats to the public market. The capabilities of these thermostats, however, are limited mainly to temperature control and scheduling. These products can both be controlled through smartphone applications, but still cannot function in a completely unified manner with other smart devices. Smart lock technology exists as well, such as the popular August Wi-Fi Smart Lock. It grants the ability to use a smartphone or a smart-watch as a house key. Some versions can also support the addition of a keypad for other means of control. The door can be unlocked remotely, but the system also recognizes when a user has arrived home using location tracking services and unlocks the door automatically [4]. While these smart locks and others that have been released on the market do function as intended, they do not pair well with other devices to reach their full potential in-home convenience and efficiency.

Some companies that have produced smart lock and smart thermostat technologies have attempted to work together and allow their systems to connect with each other. Honeywell has partnered with August so the app that controls the Honeywell thermostats and other Honeywell products can also activate the August Wi-Fi Smart Lock [4]. This integration is relatively more convenient for the user but is still limited in ability. The app can control both the thermostat and the lock but cannot combine them into one working system due to fundamental differences in the priorities of each of the two products. This project seeks to address this missing factor by enhancing existing smart lock and smart thermostat functions.

B. Project Objective

This project aims to improve upon existing smart home technology by fully integrating a new smart lock system with an existing smart thermostat system. This system’s design serves as a proof of concept to demonstrate the feasibility of this type of unification. Given sufficient time and further development, this project would ideally be ready and integrated for real-world application. While this design is intended as a prototype, the combined system of the smart thermostat and smart lock technologies detailed within this paper would have a variety of combined capabilities. The basic form of control would be through a smartphone application that is compatible with both iOS and Android. Similar to most popular entertainment platforms like Netflix and Hulu, this system would allow users within the same household to create individual profiles. It would recognize which users are home by using Bluetooth to detect each individual’s smartphone. It would also be able to recognize who is arriving at the residence by identifying which user’s profile has unlocked the door and entered.

While our design is intended as a prototype, the final implementation of the system would include a personal preference section under each user’s profile with their ideal temperature conditions. This preference section would be used to specify the user’s preferred temperature and which thermostat zones are associated with the user’s personal bedroom or individual living spaces; any preferences can be edited at any time. Thermostat zones that represent shared living areas of the house, such as the kitchen and living room, would utilize an average of all of the preferred temperatures of the people in the residence at that given time. When an individual leaves the residence, their preferred temperature would be removed from the average, so that it is adjusted to account for only the users that are home. Users would also be able to override the computation of the mean temperature and set it to a specific temperature for a desired amount of time. These capabilities would ensure that all members of the residence are satisfied with the conditions in their personal rooms and can find a higher degree of comfort in public areas of the house, helping to improve the residents’ living experience as a whole.

III. EXPERIMENTAL PROCEDURE

The objective of this project was to design the system, which involved creating a block diagram to show the general setup of devices and the flow of information, a Raspberry Pi schematic to detail how the additional hardware would be controlled and used, software that allowed the Raspberry Pi to lock and unlock the door as well as pass on information, and several If This Then That (IFTTT) applets that would all be utilized in developing the actual system.

A. Design Process

The microprocessor chosen for this project was the Raspberry Pi because it can be easily used to connect with small
sensors and motors. For the smart lock system, a magnetic reed switch was used to detect if the door is open or closed, so that it can determine whether locking or unlocking the door is feasible. Unlocking or locking the deadbolt had no use if the door was not closed. The magnetic reed switch on the door sensed whether the door was locked or unlocked and fed that data to the Raspberry Pi, which locked or unlocked the door. In addition, there needed to be some device to physically turn the deadbolt. The final solution was a servo motor that would be mechanically attached to the existing deadbolt on the interior of the door. This servo received the data from the Raspberry Pi to control when it should lock or unlock by rotating to the appropriate position.

To connect the Raspberry Pi to the smart thermostat and smartphone, the IFTTT service provides the appropriate functionality. IFTTT is a platform that allows for the integration of multiple services through applets to create new and personalized capabilities for existing devices. IFTTT is compatible with several smart thermostats and also commonly used with the Raspberry Pi. IFTTT’s Webhooks service is used to facilitate communication between the Raspberry Pi and the applet. Webhooks is a service that can send and receive requests supported by HTTP to trigger events on associated devices. HTTP, which stands for Hypertext Transfer Protocol, is a communication protocol that allows users and devices to transmit and get data from web resources through requests. POST requests send data to a server and GET requests pull data from a server. For the purposes of this project, the server is the Raspberry Pi.

This design prototype will utilize an existing smart thermostat. The first instinct was to utilize the Google Nest smart thermostat due to its popularity and large consumer base. Although there was documentation found on the IFTTT website suggesting that it was compatible with the Google Nest, further research proved otherwise. Other sources that tracked the developments of the Google Nest showed that Google had terminated its partnership with IFTTT and became solely compatible with the Google Assistant platform. The Honeywell Total Connect Comfort thermostat was the next best option because of its pricing, IFTTT support, and extensive documentation resources.

Bluetooth connection was added as an intermediate between the smartphone and the Raspberry Pi. Bluetooth can be used to detect nearby devices, which would allow the system to recognize when users enter and leave the home.

Based on the initial system design, the only way to unlock the door from inside the house was through the smartphone application, whether it was to leave the house or let a guest in. This would be inconvenient for the users, so there needed to be a method to manually unlock the door from the inside. Instead of making the deadbolt entirely manual, a push-button was added to the system, which would unlock or lock the door when pressed.

B. Software

This project software is written in Python because it is widely supported on the Raspberry Pi and IFTTT. The code is needed to process all the information from the Raspberry Pi’s general-purpose input/output (GPIO) pins. It is also needed to handle the logic associated with determining when to lock or unlock the door. In addition, it would need to check the input for the push button and determine whether the door should unlock. It needs to store data about user preferences and compute the average preferred temperature of the users who are in the house at a given time. If there is at least one user in the house, the code would need to have the Raspberry Pi send a POST request through the Internet to the IFTTT applet, which would recognize the request and trigger the Honeywell thermostat to set the temperature to permanent hold. Due to the limitations of remote development, this project is a proof of concept design. In this design, working code was written that would be necessary to make our design function with limited capabilities. User temperature preferences are directly defined in the code for this prototype version. This limited version of the code is given in the Appendix.

C. System Overview

The block diagram, as seen in Figure 1, provides a high-level overview of the system. The locking servo is used to lock and unlock the deadbolt. The Raspberry Pi can actuate the servo and control its position. The Raspberry Pi also receives input from the magnetic reed switch and the push button. The magnetic reed switch determines whether or not the door is closed, and the push button sends commands to the Raspberry Pi to unlock or lock the door. If the system was fully implemented, the Raspberry Pi would also track how many users are in the house, which it would pass on to the IFTTT applet to determine the optimal temperatures. The applet would then send commands to the Honeywell thermostat system which changes the temperatures of different zones in the house accordingly.

Fig. 1. Block diagram

The schematic for the electronics connected to the Raspberry Pi-based smart lock, as seen in Figure 2, was in constant development alongside the block diagram. The schematic
Fig. 2. Schematic shows how all of the hardware elements for the system are powered and connect to the Raspberry Pi [13] [14]. The components included are the push button, magnetic reed switch, and locking servo.

The next portion of this paper will detail each aspect of the system.

1) **Bluetooth:** Bluetooth is a popular low-powered wireless communication protocol that allows for the sharing of data between different devices. In Bluetooth, short range-radio signals are broadcasted so that devices can identify each other. All devices must be discoverable so that they can be paired to exchange information back and forth. A fully implemented version of the smart lock system would have the Raspberry Pi use Bluetooth to detect paired smartphones that are nearby. Every Bluetooth device, including phones, has a unique 48-bit Bluetooth address. After being initially paired with a Bluetooth sensor, the sensor can recognize the specific address every time it comes into range. This unique address would allow the smart lock system to identify which user is leaving or entering the house. [15] [16].

2) **IFTTT:** IFTTT is a popular web-based service that connects apps and devices from different developers together using applets. This system makes use of IFTTT’s Webhooks service to update the temperature on the Honeywell smart thermostat. Webhooks allows devices to send messages or information to other devices in real-time when a specific trigger event occurs. An applet, as seen in Figures 3 and 4, made with Webhooks can either send a POST request or receive a GET request. For the purposes of the smart lock system, the designated event will be whenever the door is unlocked, indicating that a user has left or entered their home. When this happens, a new average temperature value would be calculated based on the number of users in the house and their individual temperature preferences. The Raspberry Pi would automatically send a POST request to the IFTTT applet, which would send a command to the Honeywell Thermostat with the updated temperature value.

3) **Raspberry Pi:** The Raspberry Pi serves as the controller for the smart lock subsystem. It features a board with 40 GPIO pins, called a 40-pin header, that allows for the integration of various external devices. There are also extra pins for power and ground connections. In the smart lock system, this device connects to the locking servo, magnetic reed switch, and push button. The Raspberry Pi outputs a pulse width modulation (PWM) control signal to actuate the servo and receives information from the magnet reed switch about whether the door is open or closed. When the Raspberry Pi is called on to unlock or lock the door, it runs code that alters the frequency and duration of the pulses passed on to the servo to control the servo’s position. The Raspberry Pi also features and Bluetooth 5 capabilities, which it uses to communicate with the IFTTT applet and the users’ smartphones. The Raspberry Pi 4, the most recent model, has WiFi capabilities and an ethernet port, providing two viable options for connectivity.

4) **Locking servo:** The locking servo is a standard size high torque servo that controls a deadbolt in the door. This system is compatible with the Tower-Pro MG-995 servo. It has three contacts that can be connected via wires to the Raspberry Pi: a positive end connected to a voltage pin, a negative end connected to a ground pin, and a third that acts as a control input utilizing PWM to control the position of the servo [17]. The servo can move in a range of +/− 60 degrees. For the purpose of the lock, the Raspberry Pi will only actuate the servo to −60 and +60 degree positions [18]. The control wire passes pulses with varying width to the servo to move...
it to either the locked or unlocked position. The total period for each pulse is 20 ms for all positions, however, a pulse duration of 1 ms will move the servo to the left position and a pulse duration of 2 ms will move the servo to the right position [19]. In an actual hardware prototype, a servo would be mechanically connected to the knob of an existing deadbolt on the interior of the door. This would allow the servo to move the deadbolt to locked and unlocked positions.

5) Magnetic Reed Switch: The magnetic reed switch is a simple proximity switch that is optimal for low voltage devices such as the Raspberry Pi [20]. It consists of two parts: one is a wired element that connects directly to the Raspberry Pi and is attached to the narrow side of the door under the deadbolt. The other is a magnetic piece that is placed in the door frame and rests directly opposite of the first piece when the door is closed. It detects whether the door is open or closed to determine if the deadbolt can be locked or unlocked. The switch can be designed as a normally open (NO) or normally closed (NC) switch, meaning that when the wired end experiences a magnetic field, the reeds, or the wired elements, will either be forced together or apart. For the purposes of this smart home system, the magnetic reed switch used is NO [21]. When the door is closed, the circuit will close and the Raspberry Pi will sense a 3.3V logic high on the associated GPIO pin, through the switch. Otherwise, the switch will open the circuit and the Raspberry Pi will sense a 0V logic low on the GPIO pin, due to the 10KOhm pull-down resistor R1 as seen in 2.

6) Push Button: The push button is a manual control mechanism for the smart lock to improve the practicality of the system. If a user finds it easier to manually unlock the door while exiting the house, the push button is a faster mechanism. The button is a momentary NO push button that is connected to a GPIO pin on the Raspberry Pi. When a user presses the button, the GPIO pin senses a 3.3V logic high. Otherwise, the 10KOhm pull-down resistor R2 makes the GPIO sense a 0V logic low. Once the Raspberry Pi receives a logic high, it will actuate the servo to unlock or lock the door depending on its current state.

7) Honeywell Smart Thermostat: Honeywell produces a variety of smart thermostat and home environment control systems. Their products feature a variety of WiFi-enabled devices that can control features such as heating, air conditioning, ventilation, humidification, and dehumidification. Honeywell also produces the Total Connect Comfort, an online portal app that allows users to control their home thermostat remotely via WiFi. The Total Connect Comfort is compatible with the IFTTT web-service. This allows for connection with the Raspberry Pi through pre-existing or custom-made IFTTT applets, granting users ease-of-access. The smart home at its most basic level requires a thermostat and a heating, ventilation, and air conditioning (HVAC) system in the user’s residence for temperature control. This system would allow the user to control and monitor a zone in their house and potentially set up additional zones in the future. A multi-zone household would require an HVAC system that supports zoning, meaning independent temperature control of multiple areas in the house. For a fully developed system that includes multiple zones, the Honeywell Lyric T5 is the ideal model for the smart home due to its pricing and utility. However, the system is not dependent on the multi-zone feature.

IV. CONCLUSION

Currently, there is a wide array of smart home products on the market. With development in this industry, more companies, such as Google or August, are creating devices run by their own individualized systems. While this expands the capability of smart home technology, as homeowners install more smart products, they must operate with more systems to control these various devices. Integration between various products is an area of development in the industry, which has led to services like IFTTT, which seeks to create a centralized platform for controlling all components of a smart home. This project aims to utilize IFTTT to combine smart lock and smart thermostat technologies. This smart lock and thermostat prototype system serves as a proof of concept that custom-built systems can be integrated with existing smart home technology.

A. Shortcomings

Due to the in-person restrictions, no hardware was available for prototyping this system. Therefore, the current model does not provide a way to track users in real-time via Bluetooth. The current Bluetooth capabilities of the Raspberry Pi require
manual action by the user to pair their smartphone and subsequently connect to the Raspberry Pi. This restriction makes active monitoring of not users in the house impractical for real-life application. Similarly, a user profile system was not integrated into the system as intended, in part because of these Bluetooth limitations. In addition, the current system assumes that a resident is always entering or leaving the house wherever the door is unlocked or locked, which is not always the case. In the future, this system would need to be modified to be practical for actual use.

B. Future Improvements

With more time, many additional features that can be added to the current system. Currently, the temperature control on the thermostat relies exclusively on commands from the IFTTT applet. Although this method works, a future goal would be to create an entirely original user interface that handles user inputs. While the IFTTT applet is highly adaptive to various smart devices, this specialization could make it difficult for some users. A simpler or more user-friendly interface that could send commands to the system in addition to the existing applet could improve the experience for users.

To improve active monitoring of who is in the house at any point in time, motion capture or infrared technology could be implemented into the system. These sensors could send information about where users are located to determine which zones of the house should be prioritized for heating and cooling. This active monitoring would improve the users’ experiences with the environment control and it would also improve the energy efficiency of the thermostat. It might be possible in the future to integrate the geofencing abilities of the Honeywell Home App with the Honeywell thermostat and this project’s system to monitor who is in the house or entering the house. Geofencing is a virtual perimeter that, in this case, would surround an individual home property and alert the system once a user has entered. This data would then enable the system to adjust the environment based on users’ preferred temperatures.

APPENDIX

The following code is intended to be downloaded onto the Raspberry Pi in order to operate the integrated smart lock and smart thermostat system.

```python
import RPi.GPIO as GPIO
import time
import requests

GPIO.setmode(GPIO.BOARD)
sensor = 17
button = 22
pinOut = 12

GPIO.setup(sensor, GPIO.IN)
GPIO.setup(button, GPIO.IN)
GPIO.setup(pinOut, GPIO.OUT)

isLocked = False

class User:
    def __init__(self, name):
        self.name = name
        self.if_home = True

user1 = User("User1")
user1.temp = 71
user2 = User("User2")
user2.temp = 75
user3 = User("User3")
user3.temp = 68

#determine average
def updateTemp():
    averageTemp = 0
    people = 0
    if user1.if_home:
        averageTemp += user1.temp
        people += 1
    if user2.if_home:
        averageTemp += user2.temp
        people += 1
    if user3.if_home:
        averageTemp += user3.temp
        people += 1
    average = averageTemp / people
    return int(averageTemp)

def isClosed():
    #magnetic_switch_live
    if (GPIO.input(17) == 1):
        return True
    else:
        return False

def lock():
    while True:
        #position right
        #period 20 ms
        #duration 2 ms
        GPIO.output(12, 0)
        time.sleep(0.018)
        GPIO.output(12, 1)
        time.sleep(0.002)

def unlock():
    while True:
        #position left
        #period 20 ms
        #duration 1 ms
        GPIO.output(12, 0)
        time.sleep(0.019)
        GPIO.output(12, 1)
        time.sleep(0.001)

    current_state = 0
    previous_state = 0

    while True:
        previous_state = current_state
        current_state = GPIO.input(sensor)

        #update temperature
        if updateTemp() == 0:
            pass
        elif current_state == 1 and
            previous_state == 0:
            r = requests.post(  
                'https://maker.ifttt.com/
                trigger/Door_Unlocked_Cool
                /with/key/{key_number}',
                params={"value1": updateTemp()})
```
previousState = 1

#Check button input
if GPIO.input(button) == 1:
    if isLocked:
        unlock()
    else:
        while GPIO.input(sensor) == 0:
            time.sleep(1)
        lock()

ACKNOWLEDGMENTS

The authors of this paper wish to express our deep appreciation of various people for their invaluable contributions to this project: Residential Teaching Assistant and Project Liaison Reshma Koppapa, for her constant encouragement and helpful insights that greatly improved the quality of this project; Project Mentor Daniel Hogan, for his expertise on the subject matter and patient guidance; Dean Jean Patrick Antoine, the Director of GSET, and Dean Ilene Rosen, the Director Emeritus of GSET, for their continued support and encouragement of the scholars; and Head Counselor Rajas Karagikar and Research Coordinator Benjamin Lee, for their leadership throughout this year's program. We extend our thank you to the various sponsors that have made this project possible: Rutgers University, Rutgers School of Engineering, and the State of New Jersey, for facilitating a fantastic educational experience for all students; Lockheed Martin and the New Jersey Space Grant Consortium for promoting and funding our academic ventures; and NJ GSET Alumni, for their active support and dedication.

REFERENCES