**CS 350 Final Project**

Naim Lindsay

Southern New Hampshire University

CS 360

Professor Sina Asadi

October 19, 2025

**SysTec Smart Thermostat Prototype Report**

This project demonstrates the design and implementation of a working smart thermostat prototype developed for SysTec using a Raspberry Pi 4B. The goal was to create a system capable of reading temperature data, displaying real-time information, and controlling indicator lights to simulate heating and cooling behavior. The prototype also sends periodic data over UART to represent cloud communication.

The system integrates several hardware components that work together under a finite state machine model. An AHT20 temperature and humidity sensor connected through the I²C interface provides environmental readings. A 16x2 LCD display connected through GPIO pins presents the current time, temperature, humidity, and thermostat state. Three pushbuttons allow the user to interact with the system: the yellow button cycles between operating modes (OFF, HEAT, and COOL), while the red and blue buttons adjust the target temperature up or down. Two LEDs visually represent system states - the red LED fades in and out while heating, the blue LED fades in and out while cooling, and both LEDs remain off when the thermostat is idle.

The system architecture is built on a Python-based state machine. Each mode transition triggers specific entry and exits actions that define how the LEDs behave and how temperature control logic operates. The thermostat continuously monitors sensor input, updates the LCD display in real time, and transmits a CSV-formatted data line through the UART interface approximately every thirty seconds. The Python script was developed with modular design principles, using separate methods for state entry, temperature updates, LED control, and user input.

For the prototype, the Raspberry Pi 4B was chosen as the primary hardware platform. Compared to alternatives like the Microchip PIC32 and NXP/Freescale Kinetis microcontrollers, the Raspberry Pi provides the most flexibility for rapid prototyping. It includes built-in Wi-Fi and Bluetooth for future cloud integration, has significantly more processing power and memory, and supports high-level development in Python. While microcontrollers are ideal for low-power embedded systems, the Raspberry Pi’s operating system and GPIO libraries make it far easier to prototype and iterate quickly during early design stages.

In conclusion, the Raspberry Pi-based smart thermostat successfully demonstrates SysTec’s concept for a connected, intelligent temperature control system. It incorporates real-time sensing, visual feedback, and user interaction while maintaining an organized state-machine architecture. Future improvements could include adding cloud synchronization, web or mobile app control, and energy optimization algorithms. Overall, this prototype provides a strong foundation for future development of a commercial smart thermostat product.