**CS 499 Module Three Milestone Two Narrative**

Andrew Emilio DiStefano

SNHU

Professor Neil Kalinowski

May 23, 2025

1. **Briefly describe the artifact. What is it? When was it created?**

The artifact which I am using for Enhancement One is my final project from the CS-350: Emerging Systems Architectures and Technologies course which I took last term at SNHU (submitted April, 2025). The original system functions as a smart thermostat which enters either a heating or cooling state depending on temperature readings compared to the target temperature set by the user. The system runs on a Raspberry Pi 4 with a circuit made up of buttons, LED lights, and a LCD screen which are used to collect and communicate input and output data to and from the user and the environment. The system is a state machine written in Python. Data collected by the system is not stored in non-volatile memory - all temperature and temperature change data is forgotten when the system terminates.

1. **Justify the inclusion of the artifact in your ePortfolio. Why did you select this item? What specific components of the artifact showcase your skills and abilities in software development? How was the artifact improved?**

This artifact is not only highly relevant to my career goals, but its enhancement also provides an excellent opportunity to meet the stated outcomes for this course while showcasing the skills which I have acquired throughout my participation in the computer science program at SNHU. I have chosen to enhance this artifact by adding code which will provide the system with the added functionality of determining soil humidity and entering either a *watering* or *drying* state based on comparison between the target humidity level (set by the user) and the soil humidity detected by the sensors.

This enhancement will turn the system from a smart thermostat into a complete smart greenhouse system capable of controlling the climate in a closed (indoor) agricultural environment for optimal production. The integration of these changes to the circuit, along with my code’s use of the circuit components and Raspberry Pi’s GPIO functionality, will show my proficiency in a variety of skills associated with the robotics industry. The arrangement of the enhanced circuit shows my ability to design and build low voltage electrical circuits. The addition of code which interacts with added system components displays my ability to write code to control the interaction of the components of those circuits. The ability to design and program state machines for embedded systems showcases my ability to use industry standard system designs to create systems which provide real-world industry-specific solutions.

1. **Did you meet the course outcomes you planned to meet with this enhancement in Module One? Do you have any updates to your outcome-coverage plans?**

As of the time of writing, I have made the vast majority of the necessary changes to meet the objectives stated in Module One. I have implemented the required functionality for the system to interpret soil humidity sensor readings. I have also completely rewired the original circuit, adding new buttons and lights to previously unused GPIO pins, and I have added the code necessary to communicate with these components as necessary in order to meet the system requirements which I described in Module One. The only items which I have yet to configure are the implementation of the second display screen and the connection of the actual soil humidity sensors to the system. These components are set to arrive next week, and as soon as I have these components I will integrate them into the circuit. Other than these two small details, the changes necessary to complete Enhancement One are complete! The system is functioning properly and I do not have any additional questions or areas of particular interest for improvement, but I am certainly open to any potential suggestions from my professor!

The changes which I have made to this artifact so far certainly meet the outcomes which I had planned to meet in Module One. As stated in my Module One Assignment submission, I had planned to meet outcomes *2*, *3*, and *4* with this enhancement. My comprehensive commenting and other documentation (such as keeping a record of all associated dependencies) show my dedication to designing, developing, and delivering professional-quality oral, written, and visual communications that are coherent, technically sound, and appropriately adapted to specific audiences and contexts. My implementation of a smart green-house system able to control the climate within a closed agricultural environment demonstrates my ability to use well-founded and innovative techniques, skills, and tools in computing practices for the purpose of implementing computer solutions that deliver value and accomplish industry-specific goals. My decision to write a separate state machine class in order to run each of them simultaneously using Python’s built-in *threading* library (which is actually an approximation of multithreading since true CPU multithreading is not possible given the limitations of the Python programming language) showcases my ability to design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution while managing the trade-offs involved in design choices.

While I will still be implementing all of the changes stated in my enhancement plan in Module One, there are a few minor updates which I have decided to make to my enhancement plan after having moved forward with development. The aforementioned inclusion of two separate state machines differs from my original plan to add states to the existing state machine which instruct the system to trigger watering and drying behavior while also triggering heating and cooling behavior. While adding watering and drying states to the existing machine would technically be possible, the implementation of two separate state machine classes (one for air temperature control and another for soil humidity control) greatly simplifies the interaction of the required states and functions while allowing for the use of easily interpreted multithreading (one thread for one state machine, another thread for the other).

1. **Reflect on the process of enhancing and modifying the artifact. What did you learn as you were creating it and improving it? What challenges did you face?**

While the process of modifying this artifact was relatively straightforward, the process certainly brought new possibilities to light. Before deciding to write two separate state machine classes, as I was adding new functions and component declarations to the *TemperatureMachine* class, I realized that the number of states would need to grow exponentially rather than doubling. For example, rather than having to simply add one state for drying and another watering, I would have to define state behavior for each combination of actions. These would include a state for cooling and drying, a state for cooling and watering, a state for heating and drying, a state for heating and watering, a state for only heating, a state for only cooling, a state for only drying,. And a state for only watering. Rather than defining functionality for these eight distinct non-off states, my inclusion of two separate classes for air temperature control and soil humidity control meant defining only two non-off states for temperature control and two non-off states for humidity control.