Vyorius Test

Task Title: Digital Thermostat Implementation

Task Description:
As an embedded software engineer intern, your task is to design and implement a digital thermostat system. Although you won't have access to the actual hardware, you should be able to demonstrate your knowledge and skills in embedded systems development.

Requirements:

Programming Language: Use C/C++ for implementing the simulation.

Temperature Simulation: Simulate temperature readings from different sensors. You can generate random temperature values within a specified range or implement a simple algorithm to simulate temperature variations.

User Interaction: Implement a user interface that allows the user to set the desired temperature and observe the current temperature readings. The interface should include buttons or input fields to adjust the temperature setting.

Control Logic: Implement a control algorithm that adjusts a virtual heating or cooling system based on the simulated temperature readings and the desired temperature set by the user. The control logic should aim to maintain the temperature as close to the desired setting as possible.

Feedback and Display: Provide a visual display or feedback mechanism to indicate the current temperature, desired temperature, and the state of the heating or cooling system (on/off).

Code Organization: Use appropriate software design principles and modular code organization to ensure readability, maintainability, and extensibility of the codebase.

Deliverables:

Source Code: Provide the complete source code of your digital thermostat system implementation.

README Documentation: Include a README file explaining the architecture, design decisions, and any additional instructions on how to build and run the simulation.

Optional: If you have experience with unit testing, you can include a suite of unit tests to verify the correctness of your implementation.

Evaluation Criteria:
Your implementation will be evaluated based on the following criteria:

Correctness: The digital thermostat system should accurately simulate temperature readings, adjust the virtual heating or cooling system based on the desired temperature, and provide appropriate visual feedback.
Code Quality: The code should be well-structured, readable, and follow best practices for embedded systems development.
Modularity and Reusability: The code should be modular and reusable, allowing for easy modification or extension of the system.
User Interface: The user interface should be intuitive and provide the necessary functionality to set the desired temperature and observe temperature readings and control actions.

Documentation: The README file should provide clear instructions and explanations for building and running the simulation.

Note: Please submit your solution within the given timeframe, and feel free to ask any questions if you need further clarification on the task requirements. Good luck with the task!