Measurement System Automation:

Requirements Specification

4/28/25

Version 2

Team Name: Thermo-Gen

Sponsor: Steve Miller

Mentor: Jeevana Swaroop Kalapala

Team Members: Olivia Vester, Kameron Napier, Gareth Carew

THERMO-GEN

Accepted as baseline requirements for the project Thermo-Gen measurement system automation.

Client:
Sign- Date-

Team:
Sign- Date-

Table of Contents

1. Introduction

The industry of material manufacturing, particularly insulating material manufacturing, is both expansive and critical. The industry encompasses a wide range of stakeholders such as government entities like the United States military and NASA, and many private manufacturers. A defining characteristic of this industry is its focus on creating materials that provide thermal insulation/protection for specialized equipment, such as heat-resistant uniforms and space vehicles, those of which are integral in research fields like aerospace research. Although often overlooked by the general public, this industry plays a vital role in allowing for essential activities like space exploration, firefighting, and operating in hazardous environments. The following is an example that can be provided to emphasize the importance of this industry: astronauts rely on advanced insulative materials to survive the extreme conditions they may face in space. Without the protections of the insulative materials, space exploration would not be possible.

Our client and sponsor, HeetShield, operates directly within the insulating materials industry, maintaining contracting with the National Forest Service, the US Marine Corps, NASA, and several other agencies. The company specializes in the in-house development and testing of advanced insulative materials. One such material is their Opacified Fibrous Insulation (aka OFI), which is capable of withstanding excessive heat transfer up to 3000°F. HeetShield conducts its operations from a specialized facility based in Flagstaff, Arizona.

Despite its modest size, Heetshield has demonstrated significant capability, as reflected by its partnerships with major government agencies. The company primarily advances its work through research and development grants that support a focused team of specialists dedicated to material construction and testing. Testing is carried out using a custom-built apparatus that is designed to replicate extreme environmental conditions in a controlled setting. Building on this foundation, we can now explore HeetShield's current needs.

2. Problem Statement

To further this discussion, we now turn our attention to the specific needs identified by our client, HeetShield. As previously mentioned, HeetShield utilizes a custom-built testing apparatus that is designed to meet their unique material testing requirements. While this device satisfies the team's baseline standards for data collection, it presents several operational challenges that impact efficiency and precision:

1. **Manual vacuum adjustment:** this is required to create and maintain the desired test environment pressure.

2. **Voltage regulation:** this must be manually controlled using a transformer which impacts the temperature adjusting in the test environment.

3. **Time-intensive operation:** the system demands constant manual adjustments and monitoring throughout the testing preparation process.

4. **Limited precision:** with manual settings often falling short of the accuracy required, the tests lack optimal environments.

These issues present significant concern as the materials that HeetShield develops are intended for use in extreme environments where precise testing is critical. If the testing apparatus cannot accurately replicate these conditions, the reliability of the materials may be compromised. Returning to the earlier astronaut example, a material that performs adequately under one set of atmospheric conditions may fail under slightly different circumstances; this potentially places lives, like the astronaut's, at risk. Even minor deviations in testing precision can have major consequences for the material's end user. In addition to the concerns about accuracy, the current manual setup is also highly time-consuming. Operators must spend hours fine-tuning the system to achieve the desired conditions for each test. Our team's primary objective is to automate this testing process, eliminating the need for manual adjustments. This change will note only improve consistency and precision but also significantly reduce the time and labor required for test preparation.

3. Solution Vision

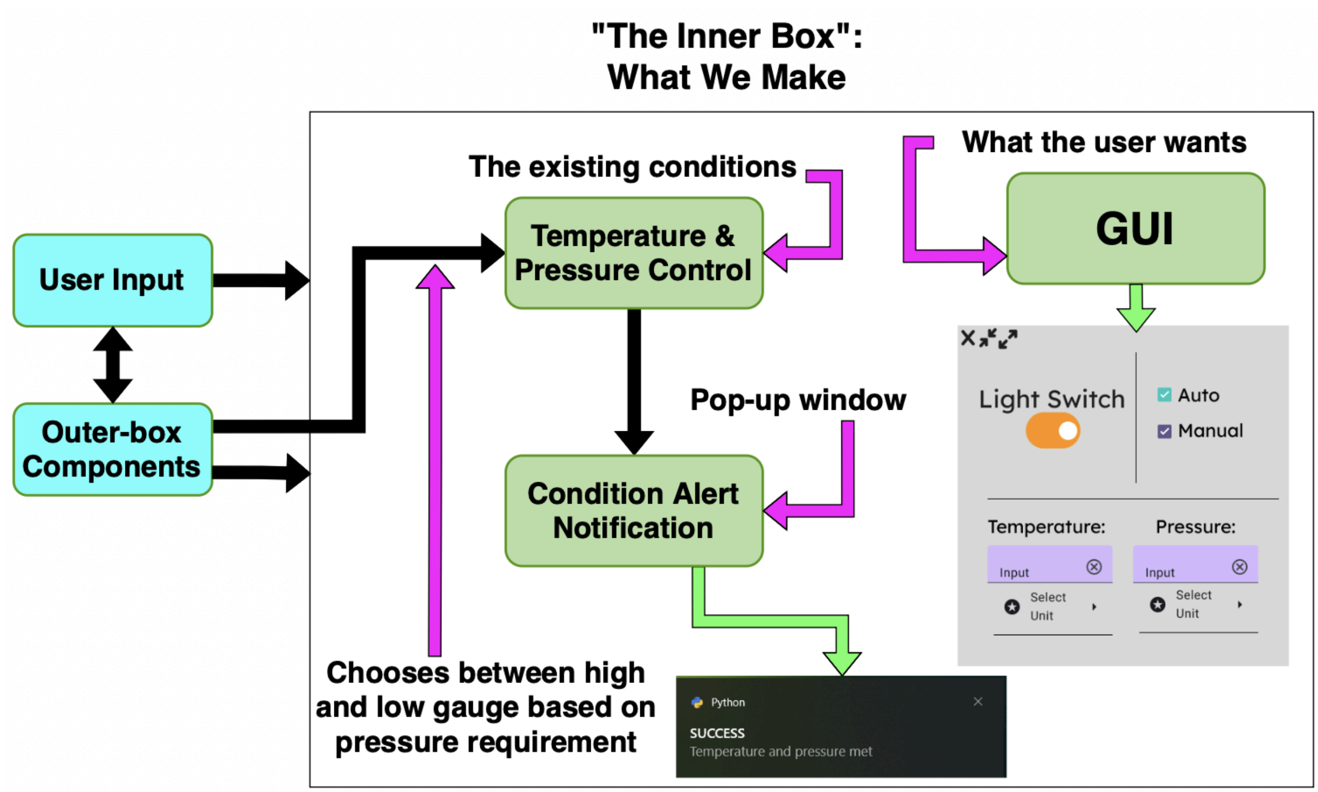
Our solution is centered around a robust software application designed to automate the process of achieving precise environmental conditions for testing. This application will offer a coherent user experience with the following features:

- **Real-time Data Acquisition:** The system reads temperature and pressure data from sensors, using National Instruments' Python API to directly interface with a Data Acquisition (DAC) system for temperature readings. A similar integration for pressure sensor data is still under consideration.
- **User Input and Parameter Validation:** Users can specify their desired temperature and pressure parameters through a user-friendly interface. The software validates input data to ensure it aligns with predefined operational limits, minimizing the risk of errors.
- **Automated System Adjustment:** The software continuously compares the user-defined target values with real-time readings from the DAC. Based on this comparison, the system calculates whether the voltage needs to be increased or

decreased to achieve the desired conditions. This dynamic control is key to maintaining optimal environmental conditions.

- **Communication and Control:** Once the necessary adjustments are determined, the software communicates with a microcontroller via USB, which in turn sends precise commands to the power supply using a 25-pin connector and cable. This communication pathway ensures the system can adjust voltage levels safely and efficiently.
- **Real-time Notifications:** The system continuously monitors data and provides timely alerts via on-screen notifications when the desired temperature and pressure conditions are met, eliminating the need for manual oversight.

This solution aims to automate the process of achieving and maintaining the specified test conditions, reducing the need for manual monitoring and intervention while ensuring safety and precision throughout the testing process. The below graphic 3.1 Figure 1 shows a high-level view of the described system and how each component flows together.



3.1 Figure 1: Display of the inner elements that compose the software that we make.

4. Project Requirements

This section presents a comprehensive overview of the project requirements, which serve as the foundation for all subsequent design and development decisions. The requirements were carefully developed through a series of collaborative meetings with Heetshield, during which we gained a thorough understanding of their objectives, evaluated the current system's limitations, and identified areas for improvement. The resulting set of requirements reflects a balanced consideration of the desired functionalities, operational constraints, and expected behavior of the new system. While this approach ensures a solution that aligns closely with the needs of HeetShield, it also introduces some trade-offs to meeting all the required points like added system complexity and higher resource requirements. Despite the challenges posed by our solution, our approach still aims to cater to our client a deliberate and informed final product.

4.1 Functional Requirements

4.1.1 Read the temperature

The system is designed to acquire temperature data using National Instruments' Python API. To facilitate this, the software must first establish a reliable connection to the data acquisition (DAC) hardware. This process entails loading the necessary drivers and libraries, initializing the DAC, and implementing error handling to manage connection issues and retries. Comprehensive error handling mechanisms will be integrated to address common issues such as hardware disconnection, driver incompatibility, or port conflicts. In the event of a connection failure, the system will be configured to perform automatic retries, generate log error messages for diagnostic purposes, and alert the user if manual intervention is required to restore functionality.

The system will be configured to perform periodic data acquisition reads from the DAC hardware, retrieving temperature measurements at defined intervals. Each data point is immediately timestamped upon capture to ensure a precise record of when the reading occurred. The timestamped data is then either stored in memory or written to a local database or file for further analysis, trend monitoring, or integration into the control logic. By maintaining a continuous, time-synchronized stream of temperature data, the system enhances the reliability of feedback mechanisms and supports precise environmental regulation based on real-world conditions.

4.1.2 Read the pressure

The specific requirements for monitoring the pressure are currently in the process of being defined and refined. This ongoing development involves careful analysis by both the team and the client to ensure that the monitoring capabilities will meet both technical specifications and operational needs.

4.1.3 Read user inputs

The software system includes a user-friendly graphical interface (GUI) that enables end users to define desired environmental parameters by a method for end users to define the desired environmental conditions by specifying target temperature and pressure values. The GUI features clearly labeled input fields designed for intuitive interaction, allowing users to enter numeric values directly. To uphold data integrity and ensure system reliability, the interface will incorporate real-time validation mechanisms. These mechanisms verify that user inputs are in acceptable formats and to verify that values fall within predefined operational thresholds, which are determined based on the capabilities of the underlying hardware and the physical limitations of the system. In the cases where a user may enter invalid data, the interface provides immediate feedback through a warning message or visual cue that prompts the user to revise their input accordingly. Once validated, the input values are parsed and securely stored internally within the application's memory.

4.1.4 Compare inputs to readings

The system continuously monitors and compares the user-defined target values for temperature and pressure against real-time data acquired from the DAC. This ongoing comparison is fundamental to maintaining environmental conditions within specified parameters and enables the system to respond promptly to fluctuations. To execute this process, the software calculates the difference between each target value and its corresponding measured value. These differences are evaluated against predefined tolerance thresholds to determine whether action is necessary. An integral part of this mechanism is the stability check, which ensures that temperature and pressure readings not only fall within acceptable tolerance ranges but also remains stable over a predefined duration. This stability check mitigates the risk of false positives that could result from transient sensor spikes, momentary fluctuations, or noise in the signal, thereby enhancing the reliability and robustness of the system's control response.

Once the system determines that corrective action is necessary, it calculates both the magnitude and direction of the required voltage adjustment. If the measured temperature consistently falls below the target value by more than a specified margin of tolerance, the system initiates an increase in the voltage to raise the temperature. Conversely if the measured temperature reading consistently exceeds the target temperature, the system initiates a decrease in the voltage to lower the temperature. This feedback control mechanism enables the system to operate dynamically, applying small, frequent adjustments to maintain conditions as close as possible to the desired points inputted. In addition, safeguards are implemented to account for sensor noise, transient readings, or failure conditions, thereby preventing false adjustments or unsafe operations.

4.1.5 Send necessary changes

After determining the appropriate control action—whether to increase, decrease, or maintain the output voltage—the system transmits the appropriate adjustment commands from the host computer to a microcontroller via a USB connection. The connection between the microcontroller and the power supply is via a 25-pin connector and cable. This connection facilitates the precise and secure transmission of control signals required to adjust the voltage output in accordance with system demands, ensuring accurate and safe regulation of the environmental conditions.

Upon receiving the commands, the microcontroller parses the data and generates the corresponding electrical signals or control voltages required by the power supply interface. These signals may involve setting specific voltage levels, toggling control pins, or adjusting pulse-width modulation (PWM) outputs depending on the design of the hardware control system.

It is important to note that this stage of the project involves hardware design, signal conditioning, and electrical interfacing tasks that are outside the core skill set of the student software development team. To bridge this gap, the team is working closely with electrical engineering subject matter experts who are providing guidance on selecting appropriate components, designing safe signal pathways, and ensuring that the microcontroller-to-power-supply communication is robust, electrically safe, and compliant with industry standards.

4.1.6 Notifications

Once the system verifies that the temperature and pressure readings have reached the user-specified target values, it triggers a notification to alert the user. This process involves continuously monitoring and comparing the current values to the values input by the user as well as verifying that the readings remain stable and within the acceptable range for a specified duration. Once these conditions are satisfied, the system generates an on-screen Windows-based notification to inform the user that the desired environmental conditions have been reached.

4.2 Performance (non-functional) Requirements

This section outlines the non-functional requirements that define the expected performance, environmental constraints, and system dependencies. These requirements are critical to ensuring the responsiveness, accuracy, reliability and interoperability of the system in a real world lab environment. All requirements are formulated based on discussions with HeetShield, an understanding of available hardware, and the operational conditions of their experimental environments. The resulting constraints aim to balance speed, precision, and stability, while recognizing limitations inherent in the physical system and hardware interfaces. The software stack relies on Python 3.10+, various python libraries for hardware communication and GUI rendering, and Windows based system services for notification.

4.2.1 Control Responsiveness

The system must ensure that control commands sent from the Python based software interface to the microcontroller are executed within one second. This includes both the communication and processing steps involved in applying voltage adjustments to the power supply. Specifically, the command must be transmitted from the host PC to the microcontroller in under 250 milliseconds. Once received, the microcontroller is required to process the command and output the corresponding control signal to the power supply within an additional 750 milliseconds. Furthermore, voltage adjustments initiated by these commands must begin affecting the power supply within that same one second window. To support this timing, the connection between the microcontroller and the power supply must remain stable and free of electrical noise or signal loss throughout the entirety of operation.

4.2.2 Data Gathering Accuracy and Speed

Accurate and timely data acquisition is essential for system performance. Thermocouple readings, which are collected through LabVIEW and a DAC system, must be sampled and processed at a rate of at least 8 times a second. Each temperature reading must fall within a 5% margin of error to ensure measurement integrity. The system must also maintain a packet retention rate of at least 99.5%, meaning that no more than 0.5% of data packets may be dropped during acquisition or transmission. Pressure readings must update at a minimum interval of once every two seconds and must also remain within a 5% error margin, with the same strict packet retention requirements. These thresholds ensure the environmental monitoring components of the system can support the accuracy and reliability needed for automated feedback control.

4.2.3 Parameter Stability and Threshold Detection

To declare the test environment as stable, the system must verify that both temperature and pressure readings remain within the desired threshold for a continuous duration of at least 30 seconds. This stability requirement is enforced by analyzing 8 or more thermocouple data points, which are collected at the specified sampling rate. The system must confirm that these values remain within the acceptable bounds for the entire 30 second window. Any fluctuation or drift during this time will reset the stability timer, requiring the system to revalidate environmental conditions before signaling test completion. This mechanism protects against transient spikes or unstable readings that could lead to inaccurate conclusions or premature alerts.

4.2.4 Alert Performance

Once the system has confirmed that stable environmental conditions have been achieved, a notification must be issued to the user within three seconds. This alert will appear as a Windows based system notification and must clearly describe whether the target environmental parameters have been met or if conditionals remain out of specification. This ensures users are informed promptly and can take appropriate action or begin subsequent test steps without unnecessary delay. The notification system also helps eliminate the need for constant manual monitoring of test status, improving overall efficiency.

4.2.5 System Assumptions and Environment

The system is designed specifically for operation in a controlled lab environment. It assumes use of a Windows based PC equipped with LabVIEW 2020 or later, a DAC, and required Python bindings. The microcontroller must be USB-connected or another connection that will accomplish the required tasks to the host PC and thermocouples must be interfaced through a DAC capable of supporting a minimum data rate of 8 temperature points a second. The lab environment must maintain an ambient temperature range between 15°C and 25°C and provide a stable AC power source converted to SC as needed for system components. To ensure proper signal quality and prevent data corruption, all electrical connections must be shielded and housed within grounded enclosures. These environmental and hardware constraints are necessary to meet the outlined timing, accuracy, and responsiveness requirements.

4.3 Environmental Requirements

As with any student or industry project, various constraints such as space, time, materials, and cost limit what is feasible. In the case of our project, the primary limitations arise from HeetShield’s use of partially digitalized tools and the high cost of new components.

Specifically, the main project constraints are:

1. **High cost of components:** many new parts exceed prices up to $5,000, making affordability a significant challenge.

2. **Limited physical space:** The workspace surrounding the testing apparatus is not very large so the product we develop must take up minimal space.

3. **Extreme operating temperatures:** High temperatures can be emitted from the testing apparatus so the physical implements but be durable and protected in this environment.

4. **Partial digitization:** HeetShield currently works with tools that are not fully digitally integrated, requiring careful selection of compatible systems.

Despite these challenges, our system will fulfill the following key functions:

- Allow the operator to digitally specify the conditions of the testing environment.
- Interface with condition-altering machinery by reading data and adjusting settings accordingly.

We are actively collaborating with our client to identify parts that meet both the technical and environmental requirements while remaining within the budget HeetShield is willing to work with. Although finding the right components has been difficult, we and our client are aligned on the system's objectives and continue to work together to find the best-fit solutions.

5. Potential Risks

Several potential risks could impact the safety, reliability, and overall success of our system once deployed in HeetShield's testing environment. These risks can be categorized as follows:

1. **Sensor Failure**
    a. **Likelihood:** Medium
    b. **Severity:** High
    c. **Description:** Failure of critical sensors, such as thermocouples or pressure sensors, during testing could result in inaccurate or missing data. This, in turn, could cause the system to apply incorrect control actions which could lead to damage of test materials or pose safety hazards to laboratory personnel.

2. **Over-Pressure or Over-Temperature Conditions**
    a. **Likelihood:** Medium
    b. **Severity:** High

c. **Description:** Improper readings, control system failures, or faulty sensor calibration could result in scenarios with over-temperature or over-pressure. Such conditions could lead to the creation of unsafe testing environments, risk of equipment damage, and even injury to laboratory personnel.

3. **Communication Failures**
    a. **Likelihood:** Medium
    b. **Severity:** High
    c. **Description:** Disruptions of communication between LabVIEW interface, Python backend, arduino microcontroller, or power supply could prevent the system from executing safety-critical operations, such as adjusting voltage or alerting the operator. These failures pose a serious risk to the safe operation of the testing system.

4. **User Error**
    a. **Likelihood:** Medium
    b. **Severity:** Medium
    c. **Description:** Incorrect input of target parameters by operators could result in unsafe operating conditions if not properly validated by the system. Real-time input checking and clear feedback from the system are essential to mitigate this risk.

5. **Software Bugs**
    a. **Likelihood:** Medium
    b. **Severity:** Medium
    c. **Description:** Software defects could cause incorrect behavior during control or monitoring operations. While generally less dangerous than hardware failures, software issues can result in wasted time, invalid test results, or failure to meet testing requirements, thereby reducing the system's reliability and credibility.

6. **Power Failure**
    a. **Likelihood:** Low
    b. **Severity:** High
    c. **Description:** An unexpected loss of power could abruptly interrupt active testing, leading to potential damage to the system or equipment, loss of valuable test data, or the inability to return the system to a safe state.

These identified risks highlight the importance of incorporating robust fail-safes, thorough input validation, reliable communication protocols, and clear technician feedback mechanisms into the final system design to enhance overall functionality and safety.

6. Project Plan

Our project is currently aligned with the requirements outlined in the course description. We have established team standards and an initial inventory, developed a strong conceptual understanding of the project's scope. We have also created a thorough explanation of our technological feasibility, and with this inclusion of this document, have outlined detailed system requirements specifications. The team follows a structured schedule that maps out the stages of each particular assignment while accounting for all the key deadlines and check-in points.

Looking ahead, we as a team have identified four major milestones that will guide our work flow in the coming months:

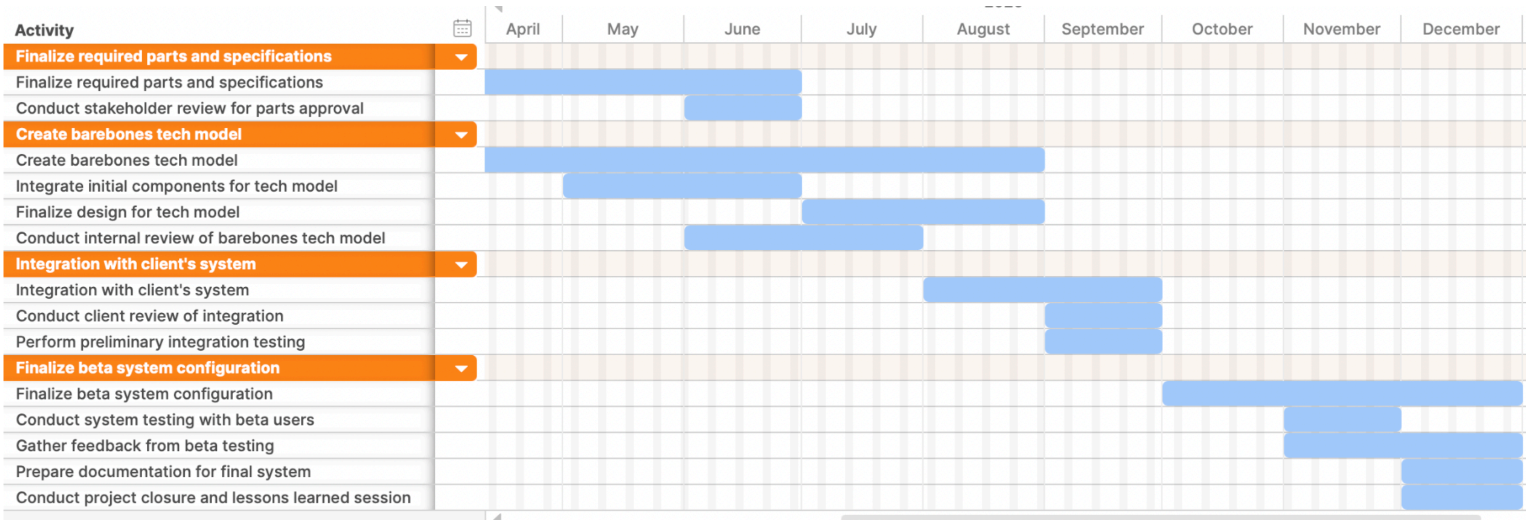
1. **Technological Demonstration:** We will showcase two foundational computing functions within our software. These functions will include the implementation of pop-up window alerts designed to inform the system operator when test conditions are met within the apparatus, as well as the initial version of the software's graphical user interface (GUI). This will take place April through May of 2025.

2. **Mechanical Component Inventory:** We will finalize a comprehensive list of all mechanical parts required for the system, with the assistance of our client, and determine the most effective method for integrating and reading their digital data. This will take place April through August of 2025.

3. **Planning and Development:** Our team will develop a high-level pseudocode outline that will define the core structure and functions of the software we are building. We will begin our programming based on our pseudcode's structure with the goal of creating a full functional version of the application. This will take place June through September of 2025.

4. **Beta Product Release and Client Integration:** A working beta version of our system will be developed, incorporating all defined goals and required system features. Part of this beta release will include our system being integrated with the client's system to prove compatibility. This will take place September through December of 2025.

While these milestones are currently broad, they will be further refined as we approach the next semester. The team will continue to use this current milestone outline along with a detailed team calendar to manage progress, all of which will be updated to reflect evolving requirements and deadlines. The below graphic 6.1 figure 1 shows a visual outline of the described milestones.



6.1 Figure 1: Gantt Chart of the team's milestone planning

6. Conclusion

HeetShield's innovative insulating materials, such as their Opacified Fibrous Insulation (OFI), play a crucial role in protecting lives and equipment in extreme environments from space exploration to firefighting applications. However, their current manual testing process introduces inefficiencies and potential inaccuracies, limiting their ability to validate materials with the precision and speed required.

This document has outlined our comprehensive solution: an automated software system designed to enhance HeetShield's testing capabilities. By integrating real time sensor monitoring, dynamic voltage and pressure adjustments, and user notifications, our system eliminates manual intervention, reduces testing time, and ensures greater consistency in replicating extreme conditions. We have detailed the functional and performance requirements, addressed environmental and budgetary constraints, and identified key risks to ensure a safe and reliable implementation.

With a clear development plan and structured milestones in place, we are confident that this automation solution will significantly improve HeetShield's testing efficiency, allowing them to advance their material innovations faster and with greater accuracy. As we move forward, we remain committed to delivering a robust, user friendly system that meets the high standards required for critical insulation testing ultimately supporting HeetShield’s mission to push the boundaries of material performance in extreme environments.