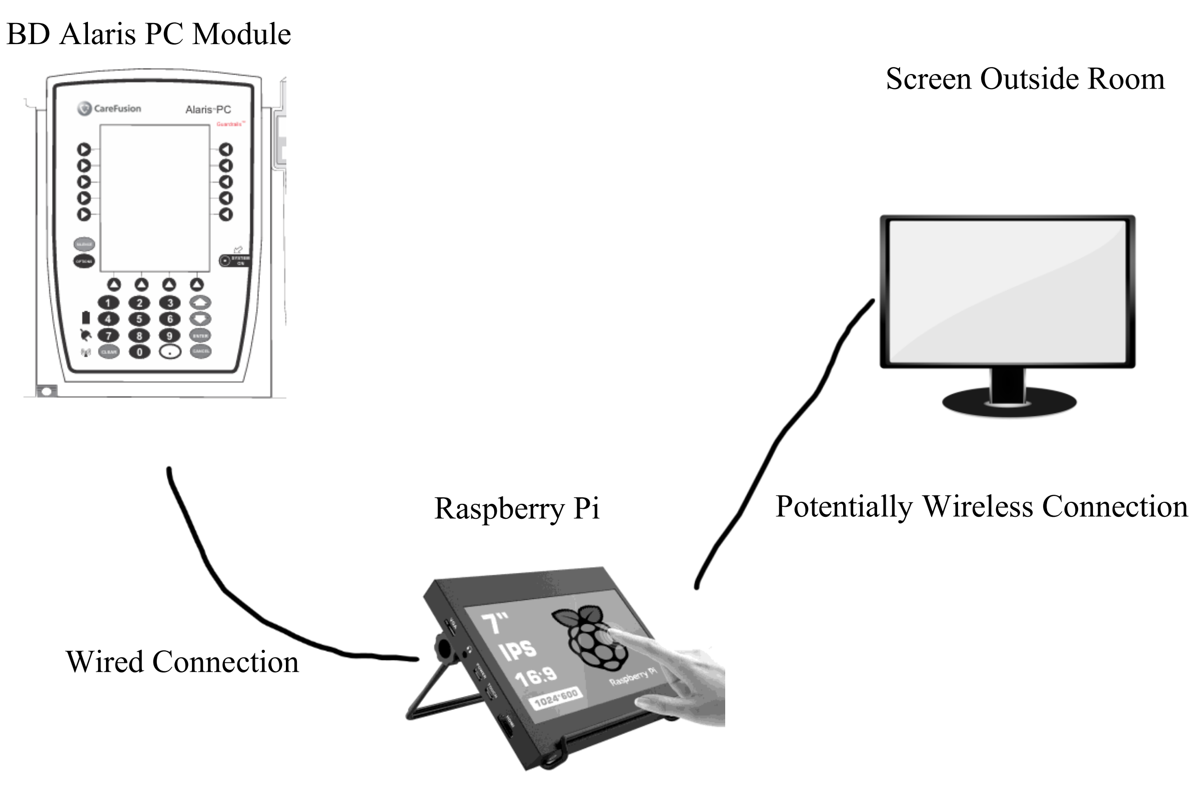
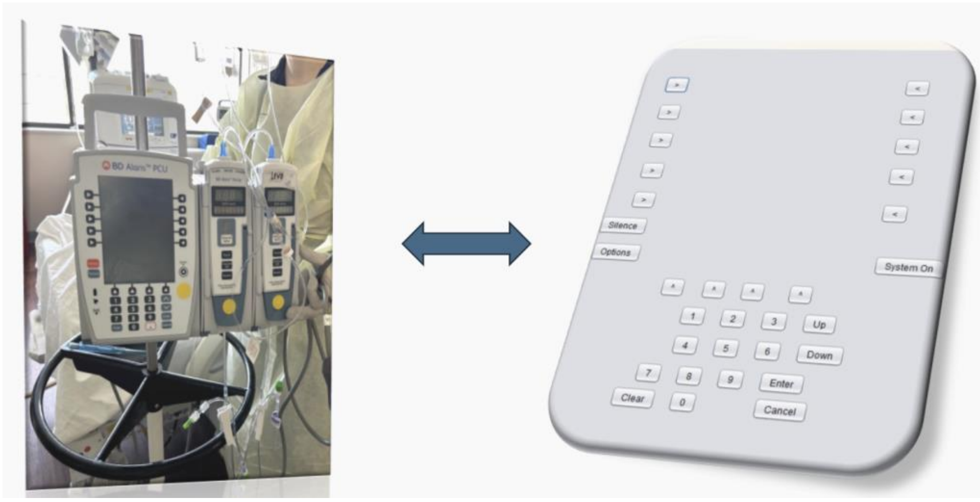
ENHANCING PATIENT SAFETY THROUGH REMOTE-CONTROLLED ROBOTICS FOR INTRAVENOUS ADMINISTRATION

Center for Immersive Learning

Final Report

April 29th, 2025





Jacob Carter
Ross Charlton
Tyler Korz
Rakshik Patturaja
Anthony Ross
Sierra Schmidt
Caden Vanderburg

Dr. Vinciya Pandian, Center for Immersive Learning - vbp5194@psu.edu

No - Intellectual Property Rights Agreement

No - Non-Disclosure Agreement

Executive Summary

The problem our project addresses is the increased risk of infection when nurses need to enter patients' rooms to change IV administration dosages. To fix this problem, our project aims to allow for remote control access to an IV pump that improves efficiency and enhances patient safety. The challenges to this were ensuring that the medical pump did not lose functionality when accessed remotely or wired to a different location and figuring out how the pump itself receives signals from the keypad. This involved reverse engineering to understand the pump itself and then creating our solution based on this understanding.

Prototype Iterations: Three separate designs for the prototype of creating remote access capabilities for the BD Alaris PC Unit (PCU) and subsequent pump modules are described, in order of viability.

The first and most viable option is to remove the front panel of the PC Unit and connect each button to a different pin on a Raspberry Pi or Arduino, which could then be used for both wired and wireless remote capabilities. Upon connecting inspecting each button's waveform with a logic analyzer, we could then mimic the necessary voltages to effectively transfer the PCU's functionality. Dr. Pandian has approved us to move forward with this design.

The second option uses the ethernet port of the PCU to connect to either pre-built control software or developed software. One could connect to the PCU via a secure shell protocol (SSH) and obtain the PCU's host IP address, then execute files and software remotely over the SSH connection. The main issue with this is the potential security measures set in place to prevent this from occurring, which is outside the expertise of our team.

The third option instead relies on the pump modules, removing the PCU from the design entirely. Flow control of the pump modules could be guided by spoofing the signals being sent from the PCU to the pump module. The major downside to this is loss of functionality that is associated with the PCU, such as connection to patient data and drug libraries.

The proposed budget for this project was previously $450. The PCU and pump modules were given to us by Dr. Pandian because the cost of these parts was outside the scope of the budget. Since then, our group has spent $134 on travel costs to Hershey Medical Center, and $272.56 on additional material costs, which totals $406.56. Our group does not plan on spending more money on this project, but additional auxiliary materials may bring our costs up to $450.

Table of Contents

Table of Contents

5.0 Introduction

Intravenous (IV) lines are important tools that allow nurses to deliver fluids, medications, and other therapies directly into the bloodstream, increasing the intake of life-sustaining medications. IV lines that are placed directly into major veins are often called central venous catheters (CVC). Since these veins lead directly to the heart, the medication in these lines gets dispersed quickly throughout the body. A central line can stay in the body for weeks or months as they are sturdier and are often placed deeper in the body than their peripheral line counterparts.[1]

However, one of the largest issues with IV lines are the increased risks of healthcare associated infections (HAIs). Around 3-16% of patients acquire HAIs per day because of IV lines, otherwise known as catheter-related bloodstream infections (CRBSI).[2] These infections are extremely serious, as IV lines lead directly into the bloodstream. An infection site directly on an open wound can spread to vital organs quickly and can prove to be fatal. Around 28,000 people die from central line associated bloodstream infections (CLABSIs) annually.[3] The numbers only grew after the COVID-19 pandemic and, in response, hospital staff began to completely disinfect themselves and wear scrubs before entering the room of any patient with a central line in their bodies. The rise in CLABSI cases was exacerbated by the COVID-19 pandemic. Early in 2020, the CLABSI rate per 1,000 Central Line Days was 1.16. By the end of the year, this figure had skyrocketed to 5.13, marking a dramatic increase in infections.[4] Similar findings have been found in other studies, such as the research by Weiner-Lasttinger et al., who observed a 27% increase in infections from quarter one to quarter two, followed by a nearly 50% increase between the third and fourth quarters in 2020.[5] These statistics highlight the urgent need for solutions that reduce physical contact between healthcare providers and the IV pumps, which are often a vector for contamination.

In addition, for healthcare providers, the need to support a sterile environment—especially during the increase in emergency room usage due to COVID-19—requires frequent and time-consuming procedures, such as entering and exiting patient rooms while adhering to stringent personal protective equipment (PPE) protocols. These necessary precautions are both labor-intensive and a time drain for healthcare workers. A study conducted at Johns Hopkins Hospital by Balazs Vagvolgyi et al. revealed that, on average, it takes a healthcare professional 170 seconds to wear the proper PPE and an added 73 seconds to safely remove it.[6] While necessary for infection control, these protocols result in considerable time delays, reducing the overall efficiency of hospital operations.

Given this context, two of the most dangerous complications that can arise from IV fluid administration is CLABSIs, which can often lead to sepsis and healthcare providers' inability to change anything relating to the IV without entering the patient's room. To mitigate both the health risks to patients and the operational inefficiencies hospitals face, it is essential to implement a remote IV dosage system. Such a system would end the need for healthcare providers to enter patient rooms for the sole purpose of adjusting IV dosages after first setup. This would reduce the risks of healthcare provider-caused infections, such as CLABSIs, by minimizing physical contact with IV equipment. Additionally, the time saved from not having to repeatedly wear and take off PPE would translate into better use of hospital staff hours and a reduction in overall hospital costs. By implementing a remote IV dosage regulator, hospitals could both safeguard patient health and improve the efficiency of care. This technological solution would offer a dual benefit: reducing HAIs while simultaneously freeing up time and resources that can be better distributed to direct patient care.

6.0 Design Features and Relation to Technical Specifications

The design of Prototype 1 focuses on creating an efficient, safe, and reliable system for remote IV fluid regulation using the PCU and Raspberry Pi with Wi-Fi connectivity. This system meets the technical specifications by providing real-time flow control, feedback, and interface capabilities.

System Overview

The system consists of four main components:

1. **BD Alaris PC Unit**: The BD Alaris PCU serves as the central control unit for the IV pump system. It manages the operation of the pump modules and ensures accurate delivery of fluids and medications, as well as being able to access patient data wirelessly via servers. Our Raspberry Pi is connected to this to allow for remote capabilities of pump modulation.
2. **BD Alaris Pump Module**: The pump module is responsible for the actual administration of IV fluids. It receives commands from the PCU and adjusts the flow rate accordingly.

Figure 1: PCU with front panel taken off. Relevant connection shown on right.

3. **Raspberry Pi Hub with Screen**: Replacing the earlier plan to use software on a nurse's PC, the new design incorporates a Raspberry Pi with an integrated touchscreen. The Pi serves as the central interface, enabling the nurse to control all four pumps in real-time without needing an external computer.
4. **Subsystem Communication**: Communication between the PCU and Raspberry Pi is initially metered via a wired connection. This setup reduces the need for physical interaction with the patient and required dressing changes, aligning with healthcare goals of minimizing contamination risks.

Subsystems and Interfaces

1. **Pump Control Subsystem**: The subsystem includes PCU, pump modules, and Raspberry Pi that work in conjunction to administer correct dosages of medication to patients.
2. **User Interface Subsystem**: The Raspberry Pi 4 with its touchscreen serves as the control interface. The interface displays real-time feedback from each pump and allows the nurse to adjust the flow rate using simple input methods such as sliders and buttons. The display also shows error messages or alerts in case of disconnections or flow interruptions.
3. **Communication and Power Interface**: The system is powered via USB or battery, ensuring that it remains portable and reliable during operation.

Achieving Technical Specifications

The technical specifications require precise control of fluid flow, minimal latency, reliable wireless communication, and user-friendly operation. Here’s how the design meets these goals:

- **Precise Flow Control**: The pump modules have already passed clinical trials and commercialization, so the flow control remains the same precision regardless of changes in where the controls are coming from.
- **Low Latency**: The PCU and Raspberry Pi are programmed to handle real-time control commands, ensuring minimal delays between user input and response.
- **Reliable Communication**: Wi-Fi communication is robust and allows for seamless control from the Raspberry Pi hub to the PCU. Error handling mechanisms ensure that disconnections or communication failures are managed without disrupting the patient’s IV flow.
- **User-Friendly Interface**: The Raspberry Pi’s touchscreen simplifies the control process, making it intuitive for healthcare professionals to adjust settings without needing extensive technical knowledge.

7.0 Code Overview

The code used on the Raspberry Pi will model the functionality described by Alaris PCUs and pump modules. The full code is attached in the teams, but a screenshot is included below to briefly show the structure. The code will include modules for:

- **User Interface:** Handling input from the touchscreen and displaying real-time data.
- **Communication:** Managing communication between the Raspberry Pi and PCU.

```python
def create_button(self, parent, text, command, row, column):
    """
    Helper method to create a button with standard styling
    """
    button = ttk.Button(parent, text=text, command=command)
    button.grid(row=row, column=column)
    return button

def on_close(self):
    """
    Clean up resources when the window is closed
    """
    self.ribbon.shutdown()
    self.destroy()

def toggle_pins(self, pin1, pin2):
    """
    Helper method to toggle a pair of pins on for 200ms then off
    """
    def toggle():
        self.ribbon.set_pin(pin1, True)
        self.ribbon.set_pin(pin2, True)
        time.sleep(0.2)  # 200ms
        self.ribbon.set_pin(pin1, False)
        self.ribbon.set_pin(pin2, False)
```

8.0 Material and Component Selection Process

Dr. Pandian provided our team with Alaris PCU's and pump modules like those used in the ICU of Hershey Medical Center. Raspberry Pis were obtained from the previous semester's team working under the same project name. Wires and soldering equipment were obtained and used from Penn State's Engineering Design and Innovation and Electrical Engineering West buildings. Ribbon cable and connectors were ordered and acquired through DigiKey distribution company. Two additional front case covers with assembly keypads were ordered through PartsSource website.

9.0 Design Analysis

The design analysis evaluates the performance, reliability, and overall effectiveness of the remote IV fluid regulation system. This section focuses on key aspects such as accuracy of flow control, response time, system robustness, and user feedback.

5.1 Accuracy of Flow Control

Our objective is to ensure that the pump modules deliver the correct amount of fluid as specified by the user. Our method consists of conducting a series of tests to measure the flow rate accuracy under various conditions and comparing the measured flow rates with the expected values to determine the system's precision. The result should demonstrate high accuracy in delivering specified fluid volumes.

5.2 Response Time

Our objective is to measure the latency between user input on the Raspberry Pi and the corresponding adjustment in the flow rate. Our methods for doing so will be recording the time taken for the system to respond to user commands and adjust the flow rate accordingly and performing tests under different network conditions to evaluate the impact on response time. The result should exhibit minimal latency, with response times consistently below the threshold required for in-room control.

5.3 Overall Effectiveness

Our objective is to evaluate the overall effectiveness of the system in achieving its goals of enhancing patient safety and improving hospital efficiency. We will do so by calculating the time saved by nurses no longer having to perform dressing changes, as well as monitoring CLABSI rates in patients with and without the remote-controlled IV pumps.

10.0 Test Procedure Plan

The test procedure plan outlines the steps necessary to validate the performance and reliability of the remote IV fluid regulation system. The plan includes functional testing and integration testing to ensure the system meets all technical specifications and user requirements.

1. **Initial Setup:** Assemble the system components, including the BD Alaris PC Unit, pump modules, Raspberry Pi, and necessary wiring. Ensure all the connections are secure and the system is powered on. Verify that the Raspberry Pi interface is functioning correctly and can communicate with the pump modules.
2. **Baseline Measurements:** Record the initial performance of the system under standard conditions. Measure the flow rate accuracy of the pump modules and compare it to the expected values. Monitor the response time between use input on the Raspberry Pi and the corresponding adjustment in the flow rate.
3. **Functional Testing:** Test each feature of the system individually to ensure proper operation. Verify that the Raspberry Pi interface can control the flow rate, start and stop the pump, and display real-time feedback. Check for any error messages or alerts and ensure they are displayed correctly.
4. **Integration Testing:** Validate that the system can handle multiple pump modules simultaneously without performance degradation.
5. **Stress Testing:** Evaluate the system's performance under extreme conditions, such as high flow rates or prolonged operation. Identify any potential weaknesses or failure points in the system. Ensure that the system can recover from errors or disconnections without compromising patient safety.

11.0 Project Management Update

7.1 Project schedule

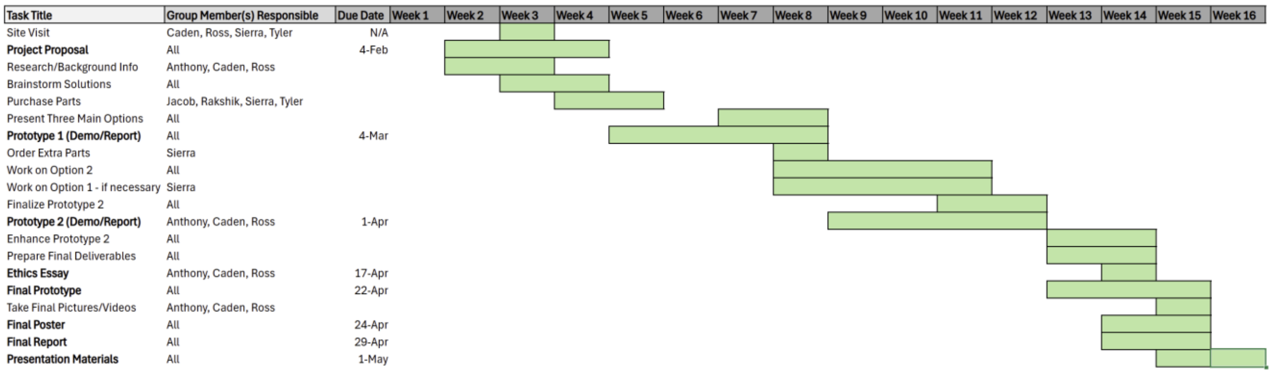


Figure 2: Project Gantt Chart

This Gantt Chart's updated milestones give more granularity in the completion of the project. The schedule is divided into major sections based on the brainstorming, prototyping, and final deliverables phases. Overall, our work was on schedule with deliverables, with some minor delays in a few assignments.

7.2 Risk Plan and Safety

The risk plan and safety section addresses potential risks associated with the project and outlines strategies to mitigate these risks. Ensuring the safety of both patients and healthcare providers is a top priority.

1. **Component Failures:**
   - **Risk:** Critical components such as the pump modules or Raspberry Pi may fail during operation.
   - **Mitigation:** Maintain a stock of backup components and perform regular maintenance checks to identify potential issues early.
2. **Data Security:**
   - **Risk:** Unauthorized access to patient data or control systems could compromise patient safety.
   - **Mitigation:** Start with a wired approach, then worry about encryption, secure communication protocols, and access controls later.
3. **Communication Failures:**
   - **Risk:** Wireless communication between the Raspberry Pi and pump modules may be disrupted, leading to loss of control.
   - **Mitigation:** Implement error handling mechanisms to detect and recover from communication failures. Ensure that the system can operate in a fail-safe mode if communication is lost.

4. **User Errors:**
   - **Risk:** Healthcare providers may make mistakes while using the system, leading to incorrect IV dosages.
   - **Mitigation:** Design an intuitive user interface with clear instructions and safeguards to prevent accidental errors. Provide training for healthcare providers on how to use the system effectively.
5. **Power Supply Issues:**
   - **Risk:** Power interruptions or fluctuations could disrupt the operation of the system.
   - **Mitigation:** Use a reliable power supply with backup options such as batteries or uninterruptible power supplies (UPS) to ensure continuous operation.

7.3 Issues

The issues section addresses the current problems our project has faced, along with potential solutions. They are listed below by importance to the project on creation of this document:

1. **PC Unit Error/Failure (CRITICAL, UNSOLVED):** On March 31, the team went in to work on the PC Unit cable extension pathway. The previous week, the PC Unit was fully functional. Upon attempting to turn on the PCU, the team experienced the following error:

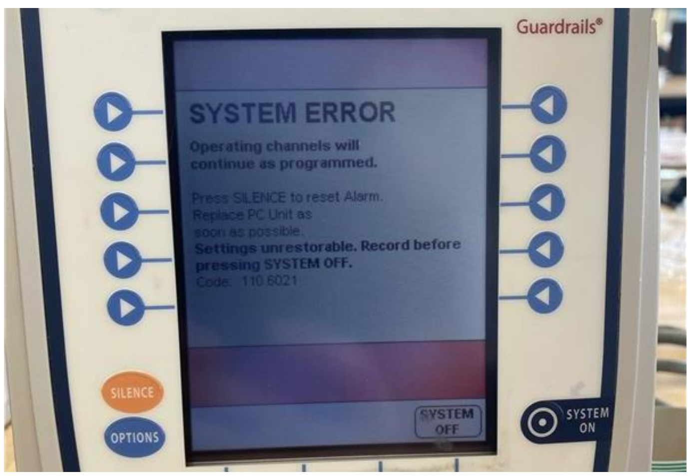


Figure 3: Error Display Screen

The error code 1106021 is related to keypad defects. We currently have three screen/keypad combinations, none of which work. The original screen which came with the PCU does not recognize connection when plugged in, and the PCU cannot turn on as a result. The PCU does recognize connection for one replacement screen, but when turned on afterwards, the error occurs. The third screen is unusable, as the header for the 20-pin connector was taken off for the cable extension pathway. Without a working PCU, the final deliverables cannot be achieved, making this problem of critical importance.

Currently, the team is ordering another replacement screen, along with contacting the technical staff at Hershey Medical Center to troubleshoot the issue. If troubleshooting here fails, the team will then contact the BD and CareFusion technical staff for further troubleshooting. If both these fail, the team will redeem the warranty for the PCU and use a new one with added care that no internal components are damaged when working.

2. **Low Battery (NONCRITICAL, UNSOLVED):** Since arrival of the PCU, there is a low battery error. When plugged in, the PCU is still fully functional, but the battery does not charge, regardless of whether the PCU is turned on or off when plugged in.

   The team currently have no plans to fix this, as it is a noncritical issue. However, it's important that this issue is addressed in the future to mitigate problems during potential power outages at hospitals. One potential solution is to replace the battery, which is available on BD Alaris' component sales site.

8.0 Future Considerations

Moving forward with future projects, additional continuity testing (with a working screen) is needed to get the remaining 6 buttons. Of the 3 screens, only 1 is partially functioning; an additional screen may be needed to get the remaining buttons. After obtaining the remaining 6 buttons, a team could use optocouplers to mediate responses between the Pi and PCU. The Pi would send a voltage to ground through one side of the optocoupler, and the other side would be connected to two pins of the 20-pin header to close the circuit, analogous to how the PCU screen and buttons function. For example, one side would send a voltage from the Pi to ground, while the other side would have wires connected to pin 1 and 2 to press the "System On" button. Future projects also need to be able to find a header to effectively communicate to the PCU without having wires touch inside the 20-pin header. To do this, a future team could cut one of the ribbon cables from a non-working screen and solder wires onto the ribbon cable corresponding to each pin. For display options, future teams could invest in a camera to be able to view the display remotely. VNC viewer can be used to access the Pi remotely, enhancing wireless capabilities.

9.0 References

1. Cleveland Clinic. "What Is a Central Venous Catheter?" Cleveland Clinic, 28 July 2022, my.clevelandclinic.org/health/treatments/23927-central-venous-catheter.

2. Cardinal Health. "Alaris® PC Unit and Pump Module Technical Service Manual", Models 8000, 8015, & 8100, Cardinal Health, pp. [insert relevant pages], March (2007).

3. Fletcher, Stephen. "Catheter-Related Bloodstream Infection." Continuing Education in Anaesthesia Critical Care & Pain, vol. 5, no. 2, Apr. 2005, pp. 49–51, https://doi.org/10.1093/bjaceaccp/mki011.

4. Haddadin Y, Annamaraju P, Regunath H. Central Line–Associated Blood Stream Infections. [Updated 2022 Nov 26]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2024 Jan-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK430891/

5. Buetti N, Marschall J, Drees M, Fakih MG, Hadaway L, Maragakis LL, Monsees E, Novosad S, O'Grady NP, Rupp ME, Wolf J, Yokoe D, Mermel LA. Strategies to prevent central line-associated bloodstream infections in acute-care hospitals: 2022

6. Weiner-Lastinger LM, Pattabiraman V, Konnor RY, Patel PR, Wong E, Xu SY, Smith B, Edwards JR, Dudeck MA. The impact of coronavirus disease 2019 (COVID-19) on healthcare-associated infections in 2020: A summary of data reported to the National Healthcare Safety Network. Infect Control Hosp Epidemiol. 2022

7. Vagvolgyi, Balazs P., et al. "ractp." Frontiers in Robotics and AI, vol. 8, 24 June 2021, https://doi.org/10.3389/frobt.2021.612964.

Appendix B: Budget Table

Budget Item	**Proposed**	**Interim**	**Actual**
Travel	**$125**	**$134**	**$0 (no reimbursement form filed)**
Equipment	**$0**	**$2,820**	**$0 (PCU and modules were not included in budget)**
Materials	**$325**	**$325.19**	**$325.19**
Total Project Costs	**$500**	**$5,279.19**	**$325.19**