MEDTRONIC TALENT ACQUISITION QUESTIONNAIRE

Candidate Name:	Praneeth Reddy Nagilla
Candidate Phone #	+1 210-255-3362
Recruiter:	
Manager:	Bemin Ghobreal
Date of Screen:	
Position:	Platform Software Engineer – Surgical Robotics

General Questions:

1. **How did you hear about this opportunity?**
   I found this opportunity on LinkedIn.

2. **Why are you considering this opportunity with Medtronic?**
   I'm passionate about creating intelligent, safety-critical systems that make healthcare smarter and more precise, and Medtronic's work in surgical robotics perfectly demonstrates that mission. My experience developing secure, real-time healthcare platforms, as well as the NavIC-based Reefer Monitoring System, which focuses on precision tracking and fault-tolerant design using C++, Python, and TCP/IP, is closely aligned with Medtronic's mission of delivering next-generation medical devices through innovation and AI.

3. **Please describe your 3 strengths related to this position, as well as 3 areas of development related to this position.**

   Strengths:

   - Demonstrated proficiency in C and C++ development on STM32 microcontrollers, including real-time system behavior, process synchronization, and module communication.
   - Demonstrated experience integrating communication protocols (UART, I^2C, and TCP/IP) in my NavIC-based Reefer Monitoring System project, resulting in reliable sensor data interchange and real-time tracking.
   - Extensive knowledge of Linux operating system principles, including multi-threaded programming, debugging, and performance analysis in distributed systems.

   Development needs:

   - Expanding hands-on experience with QNX and real-time operating systems to strengthen embedded platform development exposure.

- Gaining deeper experience in system validation and integration testing for complex embedded applications.

- Enhancing proficiency in advanced debugging and performance tuning across Linux and C++-based applications.

4. **In your previous experience, what have you enjoyed most?**

   I've particularly enjoyed working on projects that involved building reliable, performance-oriented systems, such as developing secure data pipelines and backend services for Centene's healthcare platforms and optimizing transactional systems within TCS's banking division. I also thoroughly enjoyed my NavIC-based Reefer Monitoring System project, where I programmed STM32 microcontrollers in C and C++ to enable real-time sensor communication and fault-tolerant data recording. Working on such technically complex and purpose-driven systems has fueled my passion for creating software that seamlessly interacts with hardware and delivers meaningful, real-world impact.

   Least?
   I wouldn't say there were any truly "least enjoyed" moments, but at times, frequent requirement changes or deployment challenges made delivery more complex. However, those situations helped me strengthen my adaptability, collaboration, and problem-solving skills under pressure—qualities I believe are essential in dynamic engineering environments.

5. **Please explain the reasons for leaving each of your previous employers:**
   Birlasoft: Completed a 6-month contract internship as part of early professional experience.

   TCS: Left the company to pursue a master's degree in computer science in the U.S.

   Centene Corporation: Contract position is nearing completion, and I'm looking for a long-term full-time opportunity.

6. **What is the <u>minimum</u> salary (base salary) you will consider for this role?**
   I'm looking for a base salary of around $150,000, though I'm open to negotiation based on the project scope and responsibilities defined for this position, as well as overall project budget considerations.

7. **If appropriate, can you please explain what type of relocation assistance you would be looking for?**
   I do not need relocation assistance and will manage the relocation at my own expense.

8. **Will you need sponsorship now or in the future to be employed by Medtronic?**
   I am currently on OPT, valid from July 2025 to July 2028, and do not require any sponsorship at this time.

FUNCTIONAL QUESTIONS:

1. **Please provide your proficiency level on a scale of (1-5, 5 being an expert) with the following computer and scripting languages:**

- **C: 3**
- **C++: 4**
- **Bash: 4**
- **Python: 4**
- **PowerShell: 3**
- **Other computer and scripting languages that you are proficient in: Java (4), JavaScript (3), SQL (4)**

2. **Please describe your experience with C++ Software Development?**

RESPONSE: I have hands-on experience developing and integrating software components in C and C++, both at the application and embedded system levels. During my NavIC-based Reefer Monitoring System project, I programmed STM32 microcontrollers in C++ to handle real-time data acquisition, inter-sensor communication (UART/I²C), and fault-tolerant system behavior. At Centene and TCS, I developed and maintained backend modules and APIs, focusing on reliability and performance, using C++ for data processing and system logic. I have experience with object-oriented design, multi-threading, memory management, and debugging with GDB and Valgrind. My background in embedded and system-level development has strengthened my ability to write efficient, maintainable, and high-performance C++ code for complex, real-world applications.

3. **Various Operating System Experience (Windows, Linux, QNX, Other):**
   a. **Describe the breadth of your experience. Please list development tools and environments used, debugging techniques and tools (hardware and software based) used, type of code developed, etc. Please include your experience with one or more OS and level of expertise.**

RESPONSE: I have experience developing applications, backend, and embedded software across both Windows and Linux environments. My embedded development primarily focused on STM32 microcontrollers using STM32CubeIDE with C and C++, where I implemented real-time sensor data acquisition and managed serial communication through UART and I²C. This work reflects the same real-time scheduling, concurrency, and deterministic design principles used in operating systems such as QNX.

On Linux, I gained extensive experience with Bash scripting, process monitoring, and debugging using tools such as GDB, Valgrind, and Wireshark for in-depth network analysis. On Windows, I worked with Visual Studio, Eclipse, and Postman for development, testing, and API validation. My debugging methodology included breakpoint analysis, step-by-step execution, structured log tracing, and serial console monitoring—particularly effective for diagnosing timing, synchronization, and communication issues in embedded and distributed systems.

I have a strong understanding of multi-threaded execution, memory management, and system performance optimization on both Windows and Linux. I am eager to apply this experience to QNX-based real-time systems, such as those used in Medtronic's surgical robotics platforms.

4. **Networking Experience:**
   a. **Describe your experience with networking protocol. Please include your experience and any debugging techniques / tools used.**

RESPONSE: I have hands-on experience with network communication protocols in embedded and enterprise environments. At Centene, I managed secure data exchange across distributed systems, integrated services via REST APIs over TCP/IP, and validated network performance for healthcare data pipelines. At TCS, I supported banking applications that required stable, encrypted client-server communication and resolved connectivity and latency issues within internal networks.

In my Reefer Monitoring System project, I implemented UART and I²C communication between STM32 microcontrollers and sensors for real-time environmental monitoring, ensuring reliable data transfer. I have used Wireshark, Postman, and serial analyzers to debug and validate packet flow, latency, and data integrity. I am confident working with both transport and application layer protocols and troubleshooting connectivity, synchronization, and performance issues in complex systems.

5. **Cybersecurity Experience:**
    a. **Describe your experience implementing/coding Cybersecurity concepts. Please include any relevant experience / knowledge key and certificate management / chain of trust / secure boot / cryptography / authentication / code signing / threat assessment / OS Hardening / other.**

RESPONSE: Although I have not implemented low-level cybersecurity features such as secure boot or certificate chain management, I have developed a strong security foundation through my experience in healthcare and banking. At Centene, I managed authentication and authorization using OAuth2 and JWT to secure APIs for sensitive healthcare data. I also applied best practices for encrypted data transfer with TLS and SSL, and resolved security issues at both the application and network levels.

I have a working knowledge of OS hardening, data encryption, and key-based authentication. I am eager to expand my practical experience in secure communication, threat assessment, and device-level security, particularly in safety-critical fields such as medical robotics.

6. **Please describe your approach of debugging applications and/or driver related issues. Please add any relevant tools / techniques leveraged.**

RESPONSE: I approach debugging by isolating the issue, reproducing it consistently, and analyzing system logs, call stacks, and runtime behavior. For embedded systems, such as the NavIC-based Reefer Monitoring System, I used STM32CubeIDE, serial monitoring, and step debugging to trace communication faults and timing mismatches between sensors and microcontrollers.

At Centene and TCS, I used application logs, Wireshark for network analysis, and Postman or cURL for API testing to diagnose latency, connection, and data integrity issues. I also use GDB, Valgrind, and Linux performance counters to identify memory leaks, thread synchronization problems, and runtime bottlenecks.

My process relies on structured logging, controlled reproduction, and environment comparison to identify root causes and resolve issues with minimal side effects.

7. **Other Computer and Work Experiences:**
    a. **Do you have any additional computer and work experience you would wish to share that sets you apart and makes you a unique consideration for this position?**

RESPONSE: My experience bridges enterprise-scale software engineering and embedded system development. At Centene, I developed healthcare platforms focused on secure data handling, backend reliability, and privacy compliance, implementing authentication protocols and encrypted communication. At TCS, I supported banking systems requiring fault-tolerant design and real-time data exchange.

Additionally, I developed a NavIC-based Reefer Monitoring System using STM32 microcontrollers in C and C++, implementing real-time UART and I²C communication and enabling debugging to ensure stable performance.

I published this research on Springer under the title "Development and Performance Evaluation of NavIC-Based Reefer Monitoring System" — available at: https://doi.org/10.1007/978-981-97-0562-7_15

This blend of enterprise reliability, network communication, and research-driven embedded development equips me to contribute effectively to Medtronic's platform software across complex multi-OS environments, supporting its mission to deliver innovative and safe surgical robotics systems.

My diverse background in enterprise reliability, network communication, and embedded development provides a strong foundation for building and troubleshooting platform-level software in complex multi-OS environments, including Windows, Linux, and QNX.

8. **Please highlight one project from your past experience that you are proud of?**

   a. **Tell us the project details.**
   RESPONSE: The project involved developing a NavIC-based Reefer Monitoring System designed to track and regulate temperature, humidity, and gas levels in refrigerated containers in real time. It integrated STM32 microcontrollers, multiple environmental sensors, and a NavIC receiver (S2525F8GL1) to provide precise location and timing data. The system continuously acquired and processed sensor readings and securely transmitted them to a monitoring console to enable real-time fault detection and corrective actions.

   b. **What computer language was used in the development?**

   RESPONSE The development was primarily done in C and C++, using STM32CubeIDE for embedded programming, debugging, and firmware deployment. The microcontroller firmware was developed for STM32F030F4P6 (slave) and STM32F100RB (master) units, both based on the ARM Cortex-M0 architecture, programmed through the ST-LINK/V2 debugger and in-circuit programmer

   c. **What was your contribution to the project?**

   RESPONSE: I was involved in the project end-to-end, starting from concept and hardware design to firmware programming and testing. I designed the overall system architecture, selected suitable microcontrollers (STM32F030F4P6 and STM32F100RB) and sensors, and decided to integrate a NavIC receiver (S2525F8GL1) instead of a traditional GPS module for improved regional accuracy. I personally worked on PCB design and routing, creating circuit

layouts using design software and performing component soldering, power regulation, and sensor calibration.

The most challenging and rewarding part was programming both microcontrollers in C/C++ for real-time data synchronization, error handling, and precision management. I implemented and debugged I²C and UART communication, optimized sensor data acquisition, and developed algorithms for fault detection and correction. I also handled data validation and serial debugging to ensure reliable operation across multiple modules and conditions.

d. What SW/ HW interfaces you used (communication protocol, OS, GPIO..etc)

RESPONSE: The system used multiple hardware and software interfaces to enable synchronized data flow between sensors, controllers, and the NavIC module. The two microcontrollers—STM32F030F4P6 (slave) and STM32F100RB (master)—communicated via I²C protocol, while the NavIC receiver (S2525F8GL1) was connected to the master through UART. GPIO pins were used for interfacing sensors, LCD, and keypad controls, and analog inputs handled signals from modules like MQ135 (smoke) and ACS712 (current).

The embedded firmware was developed using STM32CubeIDE with bare-metal C/C++ programming (no RTOS). I used GNSS Viewer and serial data analyzers for debugging and NMEA-0183 parsing for NavIC communication. Power management was achieved through 3.3V and 5V regulators (LD1117 and LM2576T), and the overall PCB routing was optimized for signal integrity and EMI reduction during communication.

e. If you have an opportunity to improve or redesign the device/project what would you do differently?

RESPONSE: If I had the opportunity to redesign the system, I would focus on improving scalability, precision, and connectivity. I'd replace the existing sensors with higher-accuracy digital modules using I²C/SPI for faster sampling and add wireless communication (Wi-Fi, LTE, or MQTT) to enable cloud-based real-time monitoring and analytics. I would also integrate a lightweight RTOS (FreeRTOS or QNX Neutrino) to better manage multitasking between sensor data acquisition, NavIC synchronization, and display updates.

Additionally, I'd incorporate data encryption and authentication layers for secure transmission and adopt a four-layer PCB design to minimize EMI and improve signal stability. Finally, I'd enhance fault detection through error-handling algorithms and predictive data analysis to make the system more reliable and closer to industrial or medical-grade standards.

2) Have you worked on Medical Device products- Class III?
If not, what is the industry experience you have that would be most closely related?

While I have not worked directly with Class III medical devices, my background closely aligns with the technical and regulatory standards of such environments. At Centene, I developed and supported healthcare data systems that required strict HIPAA compliance, secure data management, and high system reliability. This experience aligns with the quality, safety, and validation standards essential to medical device development.

Additionally, I designed and published a NavIC-based Reefer Monitoring System through Springer, which involved developing a real-time embedded monitoring system using STM32 microcontrollers in C and C++. This project required precision, timing accuracy, and reliability, all of which are central to Class III device software.

My combined experience in healthcare software and embedded system development provides a strong foundation to contribute effectively to safety-critical medical software at Medtronic.

Thank you for taking the time to complete these questions. It helps us identify the position(s) that will most complement your skills and work experience.