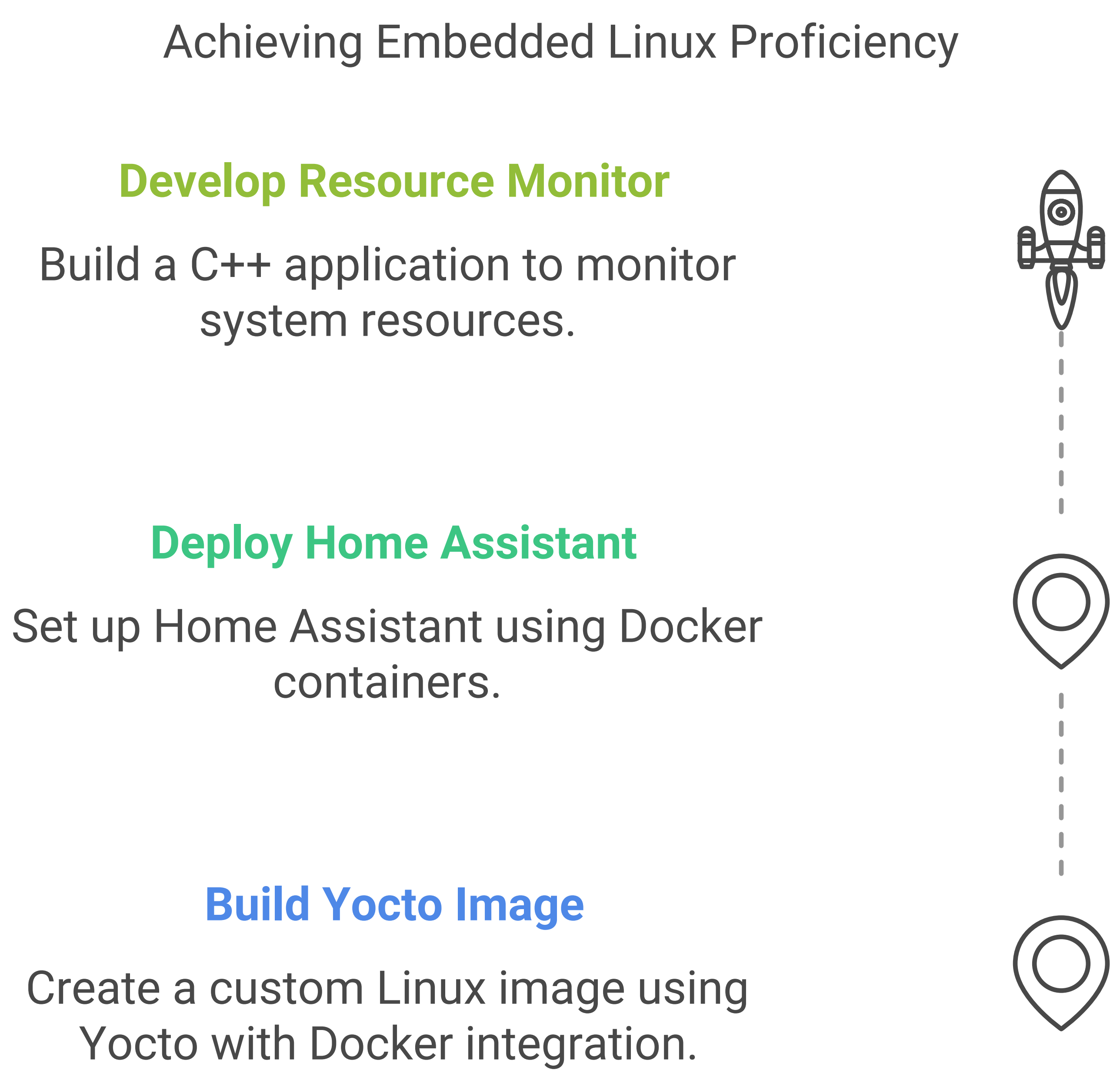
Embedded Linux Trainer Assistant - Technical Assessment Task

Overview

Thank you for your interest in the Embedded Linux Trainer Assistant position. As part of our selection process, we require all candidates to complete a technical assessment to demonstrate their proficiency with embedded Linux technologies. This document outlines the requirements for your assessment task.



Achieving Embedded Linux Proficiency

Develop Resource Monitor

Build a C++ application to monitor system resources.

Deploy Home Assistant

Set up Home Assistant using Docker containers.

Build Yocto Image

Create a custom Linux image using Yocto with Docker integration.

Task Requirements

You are required to complete the following technical implementation and provide comprehensive documentation of your process:

1. Build a Custom Yocto Image with Docker Integration

- Create a custom Yocto image with Docker engine integrated
- Document your layer configuration, recipes, and build process
- Ensure the image boots properly on your target hardware or in QEMU

2. Deploy Home Assistant via Docker Container

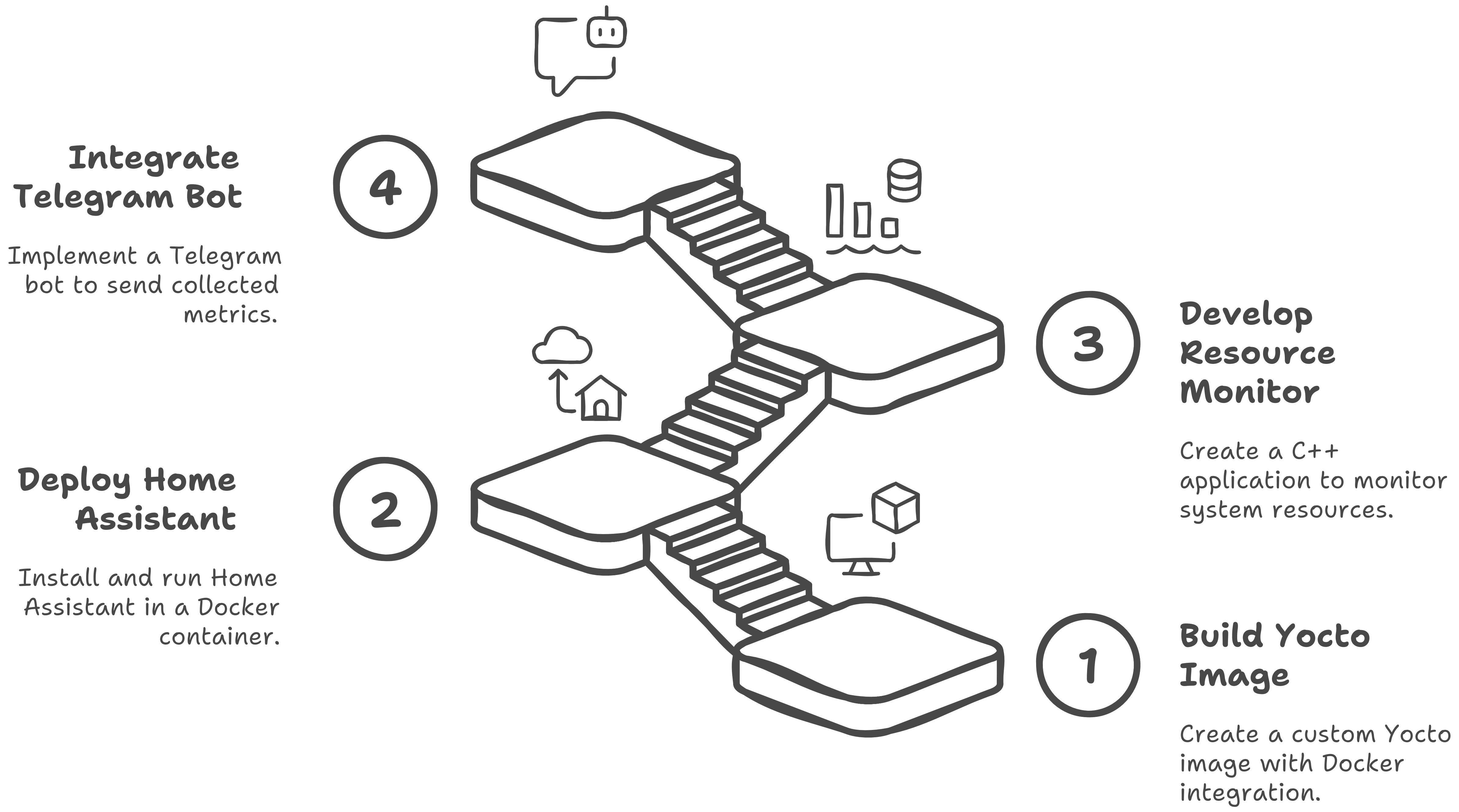
- Install and run the Home Assistant container (homeassistant/home-assistant) on your Yocto image
- Create a systemd service to ensure the container automatically starts on boot
- Document the Docker configuration and any optimizations made for embedded deployment

3. Develop a System Resource Monitor in C++

- Create a C++ application that measures and logs:
    - CPU utilization
    - Memory consumption (used, free)

    - Disk usage of the Docker container
- Configure the application to collect metrics every 30 minutes
- Implement a Telegram bot integration to send collected metrics
- Ensure the monitoring application starts automatically with the system

Achieving Embedded System Monitoring



1 **Build Yocto Image**

Create a custom Yocto image with Docker integration.

2 **Deploy Home Assistant**

Install and run Home Assistant in a Docker container.

3 **Develop Resource Monitor**

Create a C++ application to monitor system resources.

4 **Integrate Telegram Bot**

Implement a Telegram bot to send collected metrics.

Submission Requirements

Your submission on github must include:

1. **Source Code**
    - Complete Yocto layer configuration
    - Systemd service file for Docker container
    - C++ monitoring application with Telegram integration
    - Any additional scripts or configuration files
2. **Documentation**
    - Step-by-step build instructions for the Yocto image
    - Docker deployment instructions
    - Architecture overview and design decisions
    - Installation and configuration guide
    - Troubleshooting section addressing common issues
3. **Demo**
    - Video demonstration of the working system (5-10 minutes)
    - Screenshots of the Telegram bot reporting system metrics

Evaluation Criteria

Your submission will be evaluated based on:

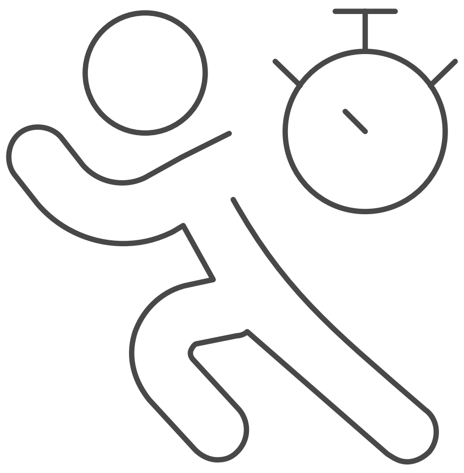
1. **Functionality** - Does the solution meet all requirements?
2. **Code Quality** - Is the code well-structured, commented, and maintainable?
3. **Problem-Solving** - How well were technical challenges identified and resolved?

Technical Hints

- For Yocto, consider using the meta-virtualization layer which includes Docker support
- Use the Telegram Bot API for sending metrics (either directly or via a library)
- Feel free to use Ai
- Explore the Linux /proc filesystem or system libraries for resource monitoring

Submission Deadline

Please submit your completed assessment before 23 May 2025. send a video demo or screenshots and your github repo to eng.moatasem.9@gmail.com with your name and mobile number use “applied for Embedded Linux Mentors for 2025“ as a subject for Email



Beat the Clock!