

Education:

- **How did you fare in high school mathematics, physical sciences, and computing? Which were your strengths and which were most enjoyable? How did you rank, competitively, in them?**

In high school, I excelled in mathematics, physical sciences, and computing. While computing, as taught in my school, primarily focused on Microsoft Office applications and didn't particularly attract my interest, I ranked first in the physics electricity and electronic modules exams across all six classes during my final year. This outstanding performance in physics, specifically in the electricity and electronics modules, further solidified my passion for the subject. However, it wasn't until university, when I was introduced to algorithms and logic circuits, that my love for computing truly blossomed. The fusion of computing and electronics, coupled with my strong foundation in physics, captivated my interest and motivated me to pursue a career in the field. The intricate connection between the two disciplines became evident as I delved into the world of algorithms and logic circuits, which allowed me to apply my problem-solving and logical reasoning skills to computational challenges.

- **Which degree and university did you choose, and why?**

I chose to pursue a degree in Computer Science at the Faculty of Sciences and Technologies based on my strong performance in electronics and logic during high school, which sparked my interest in the field.

- **Which university courses did you perform best at? How did you rank in your degree?**

I performed best in the Java programming module and the Computer Networks course and ranked the first-grade in my bachelor's degree during the final year.

- **Outside of class, what were your interests and where did you spend your time?**

Outside of class, my primary interest was spending quality time with my children. As a mom, I cherished playing video games with them and engaging in various activities to nurture their growth and happiness. Additionally, I dedicated time to managing household tasks, ensuring a balanced and fulfilling family life. This experience has taught me valuable time management and multitasking skills.

- **What did you achieve at university that you consider exceptional?**

At university, I achieved exceptional academic performance by obtaining the first grade in my bachelor's degree during my final year. This accomplishment represents my dedication and hard work.

Engineering experience:

- **What kinds of software projects have you worked on before? Which operating systems, development environments, languages, and databases?**

Java Desktop Application:

Operating System: Windows

Development Environment: NetBeans IDE

Language: Java

Database: MySQL

Description: Developed a desktop application for bank accounts and transactions management using Java. The application allowed users to perform various banking operations, such as account creation, deposits, withdrawals, and transaction history tracking.

COVID Tracker:

Operating System: Windows

Development Environment: Visual Studio Code (VSCode)

Languages: React.js (JavaScript), Redux

Description: Developed a COVID tracker application using React.js and Redux. The application consumed existing APIs to fetch and display real-time COVID statistics, such as confirmed cases, deaths, and recoveries. The data was visualized in an interactive and user-friendly manner.

Open Source Educational Social Network Feature:

Operating System: Ubuntu

Development Environment: Visual Studio Code (VSCode)

Languages: JavaScript, Python

Frameworks: Django-rest-framework, Django, React.js, Redux

Database: PostgreSQL

Description: Created a complete feature for an existing open-source educational social network. The feature enabled educators to create and share activities with students. It involved frontend development using React.js and Redux, backend development using Django and Django REST Framework, and PostgreSQL as the database for storing activity data.

Chrome Extension for Microverse Review Requests:

Operating System: Windows

Development Environment: Visual Studio Code

Languages: React.js (JavaScript)

Description: Built a Chrome extension to track new review requests on the Microverse dashboard. The extension continuously refreshed the dashboard, played an audio track indicating new requests, and generated a pop-up window to bring the dashboard to the forefront. This automated the process of finding and catching review requests, improving code reviewers' efficiency and productivity.

also contributed to many projects built using React and Ruby On Rails and Django

- **Would you describe yourself as a strong coder? Why?**

Yes, I would describe myself as a strong coder. I have the ability to solve problems and come up with creative solutions. Additionally, I consistently receive positive feedback from colleagues and peers during pair programming sessions, which reinforces my confidence in my coding skills.

- **Outline your thoughts on open-source software development. What is important to get right in open-source projects? What open-source projects have you worked on? Have you been an open source maintainer, on which projects, and what was your role?**

Open-source software development is a collaborative approach that encourages transparency, community involvement, and the sharing of source code. I believe open-source projects provide a unique opportunity for collective knowledge sharing and innovation. Key aspects to get right in open-source projects include clear documentation, active community engagement, maintaining good code quality, and having a structured process for contributions and issue tracking.

While I haven't been an open-source maintainer, I have actively contributed to open-source projects. One notable example is my involvement as a Full-stack web

developer in creating a new feature for an open-source educational social network. This experience allowed me to collaborate with the community, contribute code, and participate in discussions to improve the project. My role involved developing the backend functionality and integrating it with the front end, ensuring seamless user interactions and robust performance.

[Unstructured studio zubhub project](#)

- **How comprehensive would you say your knowledge of a Linux distribution is, from the kernel up? How familiar are you with low-level system architecture, runtimes and Linux distro packaging? How have you gained this knowledge?**

While I am not an expert in low-level system architecture or Linux kernel internals, I do have a solid working knowledge of Linux distributions, particularly Ubuntu. I am proficient in using Ubuntu for development purposes, which includes installing and managing software packages, navigating the file system, and using basic command-line tools.

I am familiar with the basic concepts of system architecture and runtimes, and I have experience using package managers like apt to install and update software. I have gained this knowledge through my hands-on experience using Linux for web development tasks, such as setting up development environments, and troubleshooting issues.

Though my experience with Linux is more focused on practical usage rather than deep system-level expertise, I am eager to learn and deepen my understanding of Linux distributions and low-level system architecture as needed for the role.

- **How comprehensive would you say your knowledge of networking is? How have you gained this knowledge?**

My knowledge of networking is primarily theoretical, based on my studies in computer networks at university, where I excelled in the course. While I had a strong understanding of key networking concepts such as the OSI model, TCP/IP protocols, routing, and network security, I didn't have the opportunity to apply this knowledge in a practical, hands-on environment. I am eager to gain practical experience and apply my theoretical knowledge to real-world scenarios.

- **Outline your thoughts on performance in software engineering. How do you ensure that your product is fast?**

Performance in software engineering is crucial because it directly affects the user experience and overall efficiency. To ensure that my product is fast, I focus on several key practices:

Writing clean, efficient code. For example, instead of using nested loops which can lead to high time complexity, I prefer using algorithms with better performance

characteristics, utilizing asynchronous processing to handle tasks that can run in the background, and ensuring database queries are optimized by using indexing.

- **Outline your thoughts on quality in software development. What practices are most effective in software teams to drive improvements in quality?**

Quality in software development is crucial for delivering reliable, efficient, and maintainable products. To ensure high quality, effective practices include regular code reviews, automated testing to ensure new changes don't affect existing functionalities, continuous integration and deployment, regular refactoring for a continuous optimisation of the code, and gathering user feedback to ensure that the software meets user needs and performs well.

- **How extensive is your experience of Python software engineering? How do you test Python applications? Outline the applications that you have led in Python, and your takeaways from that experience.**

I have built and contributed to multiple projects using Python combined with different frameworks like Django, Django Rest Framework, and FastAPI. My testing approach includes creating unit and integration tests to ensure the reliability and performance of the applications. I led the development of a full new feature for an open-source educational social network from conception to final delivery, by creating models, serializers, and REST APIs to enable the creation and editing of educational activities, including videos, images, multiple creators, and multiple steps each with images and other data. I also created endpoints for bookmarking, liking, and publishing activities. In my current role, we use Python to develop middleware solutions to automate business operations, effectively bridging AI technology and social media platforms. From these experiences, I have learned the importance of clear communication, progressive delivery, and conducting thorough code reviews to ensure we are building the expected outcome. Additionally, having a good understanding of the big picture helps us build reusable code.

Context

- **Have you ever found a problem on your computer? How did you investigate and fix it?**

I faced a critical issue where my system's root filesystem was almost entirely full, despite having a large disk and having installed a very limited number of software. I used `df -h /` to confirm the disk space issue and `du -sh /*` to identify which directories were consuming the most space. I discovered that Docker logs were the primary culprit due to improper configuration. To resolve this, I manually stopped Docker processes, uninstalled docker.io, and removed its data. Following the official Docker documentation, I reinstalled Docker and configured it correctly to prevent excessive log storage by setting up log rotation. This significantly freed up disk space and resolved the issue, ensuring it would not reoccur.

What do you consider the main engineering challenges to allow remote testing of a hardware product?

Engineers might need to physically interact with the device, such as pressing buttons which can't be done remotely. also Replicating specific environmental conditions remotely, such as temperature, humidity can be challenging. also may be internet latency if some interactions needs immediate feedback or reaction.

- **What are manual vs automated tests useful, appropriate for? What makes a good automated test procedure? What sort of tests cannot be automated?**

Manual Tests:

Useful for evaluating the user experience that Needs human judgment to assess user experience and interface design which might require subjective judgment.

and appropriate for small projects with limited repetitive tasks, and unstable projects with frequent changes

Automated Tests:

Useful to Ensures that new code changes do not adversely affect existing functionalities. and to repeatedly simulate multiple users.

and appropriate for large and stable projects where tests need to be run repeatedly and frequently.

What Makes a Good Automated Test Procedure:

Design tests that can be reused and that are easy to update when the application changes. also provide good clear and detailed messages to easily identify issues.

Tests That Cannot Be Automated:

Testing visual aspects of an application which might require subjective opinion. and testing complex scenarios that can take more time for building tests than to be manually tested.

- **Outline your thoughts on the mission of Canonical. What is it about the company's purpose and goals which is most appealing to you?**

Canonical's mission of developing and promoting open source products, particularly Ubuntu, offers appealing opportunities for developers, it's a chance to engage with a large community of users, gather valuable feedback, and adapt the product based on their needs. The collaborative nature of open source fosters innovation and allows for the exploration of creative solutions.

- **Why do you most want to work for Canonical?**

Working for Canonical can provide opportunities for professional growth and skill development through exposure to diverse projects, and international teams of experts . Additionally, as a company with a global presence, Canonical offers a distributed workforce and remote work opportunities,