Written interview for Software Engineer (Python/Linux/Packaging) at Canonical

Education

Back in high school mathematics and science was my favorite. I specialized in all the three sciences, biology, chemistry and physics. The best part was the fact that physics and chemistry used math logic, and had so many calculations, and so this created a lot of interest to me. I attained a B+ in Mathematics and C+ In Physics, Chemistry and Biology.

I chose the University of Nairobi where I joined to pursue a Bachelor of Commerce. I chose this university because it is the best and highly ranked university in Africa. The degree was of my interest because I was and still am good at Math. Later I specialized in Finance as my certificate indicates.
Later I joined ALX Africa where I took a 12 months 70 hours a week software engineering course. I chose this school because they had everything that was necessary to make me a good software engineer. In my last sprint I specialized in Back end.

What I enjoyed most in university was using the computer lab. There I got a chance to learn more things and it is due to this interest that I got to learn about software engineering and I promised myself to pursue it after my graduation. I also enjoyed my Saturdays because I got a chance to watch football and rugby which are some of my favorite sports.

In university I attended part time classes. This gave me enough time during the day to gain some experience. Outside of class I worked as a junior accountant assistant. Saturdays I was in school and Sundays I used them as my time to spend with family.

Upon my graduation from University of Nairobi I attained first class honors with cluster points of 82 %. This was exceptional to me because I did the course part time. Also I did so well in software engineering and attained points of 112%.

Context

After successfully completing the software engineering course I can say that I have experience with developer relations. I have worked on developing applications like HbnB which is a clone of AirBnb.

I have experience in building command line applications. Am competent with the C programming language and I used it to develop Printf that works just like the usual Printf. I have also developed a Simple shell which is a command line interpreter and a clone of the Unix shell.

I am familiar with the basics of build tools like make and makefile in C language to simplify and organize my code for compilation, GitHub to save my code for reference use and Puppet for automation of codes used regularly.

I have experience with cross compilation. I used cross compilation when I was working on the dynamic libraries in C language. These Libraries were shared and allowed different programs to run at the same time.

I have worked on mostly command line applications and web applications, but I never got a chance to create a Snap or Charm.

I have been using Ubuntu 20 LTS for 2 years now as the deb package for Linux. I am familiar with most if not all commands and installations and maintenance.

Am experienced with using container-on-demand Docker to perform all my debugging tasks.

What I think the major challenges are in distributing software reliability to a wide audience across multiple platforms are the fact that some people are afraid of chance and might not want to try the new software. Another is that the software keeps depreciating and needing an upgrade.

I mostly want to work for Canonical because I have the skill set required and I would be pleased to be part of the engineering team so as to advance in my career.

Engineering Experience

I have worked on web applications, command line applications, debugging, working on a remote server and low level programming. I have been using Linux as the operating system to work on my projects. My development environments have been Sublime, VS Code, Emacs and Virtual Machines. The Languages are Python, C, Javascript, Monty, and Puppet.

I am good at most of the languages that I studied but Python remains to be my favorite and the one I used oftenly. I have been using it for 2 years now to work on most of my web applications and also learning most concepts like sorting algorithms and user authentication.

The project that I am most proud of is AirBnB Clone. I say this because despite doing it as pair programming, I got a chance to develop something from scratch that is working like an original application. Also this is the project that took the most time as compared to the others that I did.

My thoughts on quality in software development is that the quality should be taken seriously as the most critical part of the software. The most effective practice to drive improvement in quality is testing often to ensure no loopholes of default. And oftenly reviewing its functionality.

Documentation is key in software projects. Documentation should be done as the project is being done. Documenting what has happened, how and where to do it as it happens makes it easy to do so since us humans tend to forget. This will help the people that will use the software or even during debugging in case of default. Some great examples of open source docs are Python Docs, Apache web server and Gobby.