Engineering experience

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

  Operating Systems: Windows (XP, 7, 10), macOS, Linux (CentOS, Fedora)
  Development Environments: IDEs (VS Code, IntelliJ IDEA, Eclipse, CLion, CodeBlocks, Devcpp), Maven
  Languages: C++ (Proficiency -Expert), Java (Core), Python, C, JavaScript/TypeScript
  Databases: Relational (MySQL, Postgres), Non-Relational (MongoDB)
  Others: REST APIs, Express.js, Angular, Junit

  I have worked full-stack website development (APTIC – mini version of Airbnb) involving technologies like Node.js, Express.js, React.js, MongoDB.

  I have worked with projects leveraging relational database like MySQL. I have used Python, and MySQL to build a Library Management System.

  I have also created a mini project (Robot Race) which involves concept of multithreading, OOPs in Java.

- Would you describe yourself as a high-quality coder? Why?

  Yes, I am a high-quality coder.

  1. My educational background, including a Bachelor's degree in Software Engineering and a Master's degree in Computer Science from a reputed university like UT Dallas, has provided me with a solid theoretical understanding of computer science principles, data structures, algorithms, and software design methodologies. This strong academic foundation serves as a robust base for my coding skills.
  2. I follow coding guidelines defined in the project resulting in lesser bugs, code smell, reusable code and less code duplication.
  3. I try to use best possible data structure depending on the scenario and try to code optimized code with less time complexity algorithm.
  4. My work experience as a Software Engineering Specialist at General Electric Digital has exposed me to industry best practices, Agile methodologies, code reviews, and collaborative development. This real-world experience has further honed my coding skills and taught me

the importance of writing maintainable, scalable, and high-quality code.

- Would you describe yourself as an architect of resilient software? If so, why, and in which sorts of applications?

  Even while I may not yet view myself as a traditional resilient software architect, I have a strong commitment to creating reliable and durable solutions in my work. From the ground up, I created the APTIC project, choosing the front end, back end, and database. The setup, design, development, and basic testing were completed on a local PC; thus, I did not investigate the hosting aspect. Building systems with the capacity to tolerate unanticipated failures, adjust to shifting circumstances, and bounce back quickly from setbacks is my top priority. Even though I'm constantly learning and honing my craft, my commitment to creating robust software comes from my conviction that it's critical to provide trustworthy, high-performing programs that can flourish in demanding, dynamic situations.

- 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 S/W development needs a transparent, and collaborative approach to create software that promotes innovation, community engagement, and drives accessibility and adoption.

  Important factors that are crucial in open-source projects are:

  a) License and legal compliance
  b) Community engagement
  c) documentation and onboarding
  d) funding
  e) Code quality and maintainability
  f) security

- 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?

I don’t have a practical application on working on Linux distribution. Theoretically, I am familiar with OS architecture, database internals.

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

  Performance in Software Engineering is critical as it directly impact user experience and satisfaction. It depends on the combination of design, implementation, and rigorous testing throughout the product lifecycle.

  Firstly, there should be a well-prepared performance metrics can be defined in Software Requirement Specification (SRS) before starting development. It will comprise of these factor(s): throughput, resources utilization, and acceptable response time.

  Secondly, using particular data structure with thorough research on its complexity for searching, inserting for the particular use case and writing optimal algorithm with minimum operation and avoid unnecessary resource allocation and employ best memory management and resource utilization.

  Thirdly, thorough performance testing is essential to validate and optimize the product’s performance characteristics. There are different testing like stress, load to identify potential bottlenecks.

  Lastly, since we cannot give our pc to end user. We should check performance in production env. And monitor it over in real time to ensure reliability and scalability under varying workloads.

  I did something like this in a smaller project to check performance for different type of hashing like java hashset, and cuckoo hashing with varying load of array size (1, 4, 8, 16, 32 million) to check performance.

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

  First there should a particular definition of quality defined before starting the project and should be followed throughout the project life. It can be defined by setting parameters for code reusability, code duplication, code smells, defects etc.

Secondly, continuous improvement should be encouraged - open communication, feedback loop, and promote a mindset of learning and adaption. Follow agile principles and after each sprint, do sprint retrospective and post-mortems can provide insights into areas of improvement and help s/w teams iterate and refine process over the time.

Thirdly, implementing automation testing is highly effective in driving improvement in s/w quality. Developer should do unit testing for the code for current sprint and before pushing the code in the prod all unit tests should be run so that it can be checked that the new code is not breaking the old code. And QA team should have a plan for integration and end-to-end test and static code analysis should be done.

Fourthly, emphasize collaboration and teamwork. Cross-functional collaboration between end-user, product owner, developers, and tester are very important. I remember my time at GE where all the stakeholder's meetup biweekly, it helped in early identifying and resolution in bugs, UI feedback and promotes knowledge sharing and ensure that quality is everyone's responsibility throughout the development lifecycle.

Multipass related technologies

- Describe your experience with C++. Which standard(s) are you familiar with? What are some features you use regularly?

  I learned C++ in my semester of my bachelor (in Software Engineering). My motivation to learn C++ was genuine love for programming, and I knew Programming in C++ will be going to taught in the upcoming semester but since there are lot of concepts in each programming language. Therefore, all concepts cannot be covered in the class syllabus. It will be beneficial for me to get it self-started. So, I started learning it from cplusplus.com,

GeeksForGeeks, etc. and practicing problems in GeeksForGeeks and LeetCode. I mostly use C++14, C++17 version of CPP. Although I read new concepts like smart pointers and try to implement few programs in these concepts also. I find it very enriching in implementing various projects, for problem solving.

LeetCode: https://leetcode.com/dark_loser/

(I removed my name from leetcode profile)

Some features that I use of C++ are rich data structures like unordered_set, unordered_map, Standard Template Library (vector, queue, priority_queue, algorithms(sort, binary_search), etc), OOPs, templates, memory management(unique_ptr, shared_ptr), operator overloading.

- How do you measure and minimize the resource footprint of your C++ applications?
    1. I will use efficient data structures (DS) and always try to write optimal algorithm to achieve the task.
       Efficient DS: Choose the appropriate container from Standard Template Library (STL) based on the particular task.
    2. Enabling compiler optimization flags (-01, -02, -03) to tell the compiler to optimize generated code for size and speed.
    3. Reducing the unnecessary calculations, operation, unnecessary logging(cout), and redundant data processing.
    4. Running the application on the platform where it will be use like local machine, or lab computer or any other cloud server (AWS, GCP, MS Azure) and take notes on usage (RAM (memory), disk, etc) and do the required changes according to the platform CPU architecture, RAM constraints, and I/O.
    5. Lastly, I always try to monitor my code continuously and optimize application through its development lifecycle and review and address any potential bottleneck and delete/remove unused code.

- What hypervisors are you familiar with on Windows and Linux? Describe your experience with these.

  I have used Virtual Box and CentOS virtualization.

CentOS: I was doing an academic internship at NUS where in my windows laptop, I was using CentOS virtualization to develop Hadoop Clusters and Cloudera Manager (Version 5.3) for Big Data and Hadoop System Administration with HDFS, HIVE, Sparks, Yarn, and HBase as part of the internship.

Virtual Box: This I was using it one of labs session in Computer Hardware and Troubleshooting as part of testing the applications in different platforms.

- Describe your development experience on Linux, Windows, and/or macOS.

  I have all the above three OS over the time and posses' expertise on macOS and Windows. I have used Fedora, Ubuntu, CentOS flavors of Linux. I have used Windows (XP, 7, 10) and switched to macOS since 2022. Before that I used windows since 2012.

  I have used different IDEs in these platforms:

  Windows: IntelliJ, CLion, VSCode, CodeBlocks, DevCpp, PyCharm, WebStrom.

  macOS: IntelliJ, CLion, VSCode, PyCharm.

  I have used Github extensively for personal projects, collaboration in school projects, and when I was working with GE.

  Personal and School Projects: It helps in a way that I can get help from my school friends in team project if I get stuck at somewhere and also, I can access my projects code/snippets everywhere with a PC and internet.

  Work Experience (General Electric): I was working as a Software Engineer at GE. So, in each sprint, I used to take user stories and/or bugs depending on the bandwidth. So, I used to take latest code from main/master branch and create a branch for user story and subbranches from branch for each smaller task and push the code with the branch for reviews from fellow developers and finally merging all the sub-branches with the branch.

  Testing: I have worked with Junit (Java), Jasmine, Karma (Angular) for testing in Windows, macOS development environment. I found testing the frontend more challenging than the backend testing.

Build Systems: I have used Maven, CMake as part of my application development. I find Maven more friendly build tool that any other build tool.

Lastly, I am willing to adapt to new technologies, tools, and best practices across different operating systems. I try to stay updated with the latest developments in Software Development by learning and building smaller projects using newer features or new technology.

Describe your level of understanding of networking, especially virtual networking on Windows, macOS and Linux.

I have an understanding of concepts such as IP addressing, subnetting, routing, switching, and also, I have an experience of networking in distributed environment as part of Adv. Operating Systems course. I did not much practical exposure to the networking concepts though theoretically I understands them.

Education

We consider academic results in high school and university for all roles, regardless of seniority. In every discipline, from engineering to marketing to operations and sales, we intensely value colleagues who are able to puzzle through difficult problems and find the optimal path forward.

- How did you rank in your final year of high school in mathematics? Were you a top student? On what basis would you say that?

  I was the one of the top performing students in Mathematics all my life. I have the full command of the topics like relation, sets, and function, vectors, 3-D geometry, probability, differentiation, integration, trigonometry, statistics, etc. I used to score mostly in top 1% in unit tests that were conducted each quarterly and in the three major exams (quarterly, half-yearly and final) through the year. I used to practice a lot of questions with the textbook and practice from the supplement to go extra mile.

- How did you rank in your final year of high school, in your home language? Were you a top student? On what basis would you say that?

  In language (English) course also, I was one of the top performing candidates in the school. I was in the top 5% of the class. There were 2 literature books, as well different essays, application as part of the course. I used to like the

essay's part the most. I think the story here is little different, I was always the top student in mathematics but in English this was not the case. It was about the continuous hard-work, I still remember my family decided to buy English newspaper so that my and my siblings can improve English day-by-day. I used to read newspaper from the last (sports section) paper as I am a Sports fan, and I am biggest Cricket buff and lastly my English teacher also very helpful. She used to help with my doubts in grammatical parts.

- Please state your high school graduation results or university entrance results and explain the grading system used. For example, in the US, you might give your SAT or ACT scores. In Germany, you might give your scores out of a grading system of 1-5, with 1 being the best.

  I had the 4^{th} rank in the class in the high school graduation in around a class of 50-60 students. My school followed CBSE (India)

- Can you make a case that you are in the top 5% in your academic year, or top 1%, or even higher? If so, please outline that case. Refer where possible to standardised testing results at regional or national level, or university entrance results. Please explain any specific grading system used.

  I scored a 4.0/4.0 in the last fall 2023 and 3.7/4.0 in the last Spring 2023. So, it can be considered a pretty good grade overall. But I cannot know other people grade here so I cannot guess that I was in top 1%, top 5%, or not.

  Standardization here at the University of Texas at Dallas, Texas: marks are relative with the highest scorer in most of the class, but some professor will have absolute grading.

- What sort of high school student were you? Outside of class, what were your interests and hobbies? What would your high school peers remember you for?

  During my high school years, I prioritized my studies and maintained proficiency across all subjects. However, I also pursued extracurricular activities with enthusiasm. Cricket was a significant passion of mine, initially focusing on bowling before transitioning to becoming a proficient batter. Additionally, I found enjoyment in strategic pursuits like chess and exploring culinary arts through cooking.

  My peers would likely remember me as someone who embraced a well-rounded high school experience. We shared memorable moments of group

studies, cricket matches, outings to movies and trips, as well as indulging in evening snacks or occasionally starting our days with breakfast together. These shared experiences fostered strong bonds and lasting memories among us. I believe those were days that I always remember the most.

- Which university and degree did you choose? What other universities did you consider, and why did you select that one?

  I chose SRM University due to its diverse student community and it has a reputed Computer Science Department. It attracts students from all the states and people speaking different languages, and I chose Software Engineering because of my interest developed in programming because I joined a summer school where I learnt C programming. I consider VIT University also due to the same reasons, but I got scholarship in SRM University, so I joined SRM University.

- Overall, what was your degree result and how did that reflect on your ability? Please help us understand the grading system for your results.

  I got 87.77 % overall where I mostly scored exceptional marks in coding, mathematics, and problem-solving courses. But at the same time, I think I scored less marks in courses where there was lot of memorizations like Software Project Management, Information Storage Management, Requirement Engineering etc. which reduced my overall percentage.

  Grading System: My university followed absolute grading not the relative grading. So, my 87.77 % means 87.77/100 not relative to the highest mark of a student. It is a measure such that I score 87.77 in a scale of 0-100.

- During all of your education years, from high school to university, can you describe any achievements that were truly exceptional?

  One instance which I find exceptional is mathematics and programming course score in undergrad.

  I have scores of 99, 99, 85, 98, 96 in mathematics courses and 100 (C), 96 (C++), 98 (Java), 90(Python) in programming courses.

  Score of 85 in one of the mathematics courses is because I fell ill in one of the mid-term exams of the course.

- What leadership roles did you take on during your education? Did you conceive of, and drive to completion, any initiatives outside of your required classwork?

  I take various roles over the academic journey; I was group captain for 2 years in school. My group was yellow and won various challenges and championships during my leadership, but I think a leader is as good as their team and I was blessed with various good team players. I was a leading a team in cleaning nearby campus area in my undergrad first year, and a class representative for 2 semesters in my undergrad.

Context

- Outline your thoughts on the mission of Canonical. What is it about the company's purpose and goals which is most appealing to you? What do you see as risky or unappealing?

  As a recent graduate in Computer Science, I find Canonical's mission intriguing and appealing due to its focus on open-source s/w and its commitement to making technology, software accessible to everyone.

  I used Ubuntu and came to know very recently that it is developed at Canonical. Since, it is an open-source and readily available I was able to get it and use it without knowing about its manufacturer/developers. I believe in the Canonical values dedication to open-source principles as a Software Engineer and by providing free and accessible OS like Ubuntu, Canonical empowers small and medium org. and most importantly individuals regardless of financial resources to leverage technology efficiently.

  Now coming to risky part, one concern that I feel company's sustainability of the business model. I believe there will be some products or service for revenue generation but at the same time it contradicts the founding motto of the company. There will be always a challenge to maintain this in the present and the future. And looking at the website and Wikipedia page, Canonical operates in a highly competitive and rapidly evolving industry dominated by tech giants, so there will always seriously hard challenges in terms of market-penetration, maintaining relevance among existing users and reaching to future and potential users and standing out among competitors.

  In summary, Canonical's mission to develop and promote open-source software like Ubuntu is appealing to me as it aligns with my values as a

software engineer. However, there are risks and challenges associated with sustaining this mission in a competitive industry, which would require strategic planning and adaptation from the company.

- Who are Canonical's key competitors, and how should Canonical set about winning?

  Canonical will always face competition with corporate in operating systems, servers, cloud computing resources.

  Microsoft: Windows occupies big chunk in OS market and compete with cloud computing space.

  RedHat: It also distributes Linux (CentOS, RHEL), so it is a direct competitor of Canonical in Linux distribution.

  Amazon Web Services: Anyone entering cloud space will face and find it hard to compete with Amazon in cloud computing service.

  The steps that can be taken by Canonical to compete with rivals:

  1. Focus and highlight on ease of use, flexibility, and security on its Ubuntu product.
  2. Foster Ecosystem partnerships: As seen on the website, Canonical has partnered with silicon giants like Nvidia, Intel, AMD, arm, etc. and hardware with HP, Dell, Lenovo. Collaborating with industry leaders and integrating Ubuntu with complementary products and services can enhance its appeal to users and drive adoption.
  3. Continue to engage with the open-source community and foster collaboration and contributions from developers, enthusiasts, and users.

- Why do you most want to work for Canonical?

  I am particularly drawn to the prospect of working for Canonical due to the abundance of opportunities it offers for professional growth and development in a rapidly evolving industry. Being part of a company developing technologies like cloud computing, containers, presents invaluable learning experiences. Additionally, Canonical's renowned collaborative culture aligns perfectly with my values of teamwork and knowledge-sharing, promising a supportive and enriching work environment. Moreover, the global impact of Canonical's projects, with Ubuntu being utilized by millions worldwide,

presents a unique opportunity to contribute to initiatives with substantial societal and economic implications on a global scale. Canonical's commitment to the open-source principles and community aligns with my passion for collaborative development.

- What would you most want to change about Canonical?

  I think Canonical is doing excellent work in innovating technology, providing products and services to individuals, small, and medium org. in free of cost. But when considering potential areas for improvement at Canonical, one aspect that could be addressed is enhancing diversity and inclusion within the organization. While Canonical has made strides in fostering a collaborative culture, there may still be opportunities to further promote diversity in terms of gender, ethnicity, and background among its workforces. Implementing initiatives to actively recruit and support individuals from underrepresented groups, creating inclusive policies and practices, and providing resources for diversity training and awareness could help cultivate a more diverse and inclusive workplace culture at Canonical.