

Elevator Pitch

Hi, I'm Kenneth Vo, a CS student from UT Dallas that graduated back in May of this year. I got into programming to tackle problems on efficiency and optimization, and finding logical solutions. Throughout my coursework, I've built several projects to answer those problems like a chatroom simulation or my capstone project on steganography and cybersecurity. Since I've graduated I've been focused on furthering my own education by getting certified in cloud computing and developing my own portfolio website to display my work! Outside of software development, I like to go to the gym and practice painting as my creative outlet. I'm really excited about working with Cognizant, especially since you guys have goals that align with mine!"

What got me into programming

- When I was younger my interest in programming was solely centered around wanting to create the video games I played, and as I got older it developed into just wanting the technology around me to work as efficiently as possible. My experience at work has shown me how faulty software most retailers use can be with the register program randomly shutting off or the books database being slow and often times crash when you want to help a customer. As much as i still love videogames I really just seek convenience and efficiency in the technology I use, and through research Cognizant I see that our goals align especially with projects like the Cotality and FA case studies mirroring the exact problems I have at my current part time.

Why do you want to work for Cognizant

- reliable business partner with multiple companies (Cotality and its Tax Ecosystem)
- projects with interesting companies like F1 and Football
- efficiency in how you guys manage projects (COTALITY)
- being able to have a proper foundation is important for a company's infrastructure, and being a part of cognizant will allow me to help other companies and cognizant business partners fulfill that requirement for more efficiency and accelerating performance
- very diverse in the projects tackled, from cloud computing migration, to developing payment platforms and __SOMETHING HERE
- your vision aligns with mine: "becoming the preeminent technology services partner"
- having dealt with faulty software and inefficient data centers through my current job as a bookseller, being able to help other companies futureproof and create efficient frameworks for their workflow is my ideal focus when it comes to being a fullstack developer. your goals

as a company are more selfless and admirable than the average fullstack position at a bank or for a generic software product

What are you looking for in your next role?

- a company that embraces innovation and change
- my current part time has software stuck in the early 2000s with faulty servers and slow programs that i myself would love to fix, so being a part of a company that allows me to work on projects focused on creating efficiency work environments for clients is exactly what i'm looking for and cognizant has proven that through their case studies

Challenges faced in my projects

UDP/TCP Server

- this project was a simple simulation of a client and server interacting with each other with some slowdown built in to learn how it affects the relationship between them.
- my main difficulty when working on this project was the idea of simulating slow down
- The project itself is simple in design, it has a server sit alert and wait for data to get sent to it, with the client being able to send messages by having both sides wait for the other side to send each other specific codes called sequence numbers
 - they both talk back at each other and wait for the other to respond before more messages can be received or sent
- the problem i faced was to accurately simulate packet loss without making it feel predictable each time, so to overcome this I had to implement random chance
 - with the basic project already working i gave both the client and server a pack loss percentage and randomly generate a percent each time a message is sent to see if a message is "lost" and skipped being sent

Steganography Capstone

- in my capstone project the main challenge faced was less technical and more about team coordination. since the project was split into multiple milestones and a set end of semester deadline, our group had to meet up multiple times to work and plan out the project itself, and the obstacle we all faced was coordinating meetups with our school and work schedules.
- to overcome this we created a simple chart that detailed all of our availability and the times we were all free was chosen to be the designated project work period each week.
- of course the system wasn't perfect, some weeks a member could miss the meeting due to exams or work shifts that weren't in their control, so to counteract this and keep everyone

up to speed, the member would reach out to see who was free that or the following week to catch them up one on one so that for the next session everything could continue smoothly

- MAYBE ADD HOW THE PROJECT WORKS AND MY CONTRIBUTION TO IT

Issues with a Team or Coworker (Interpersonal conflict)

Part-time

- Outside of my software engineering experience, I've faced interpersonal conflict at my current job at Barnes & Noble. I was working a closing shift where a coworker had to leave early due an exam.
- As part of the team that cleaned the store at the end of the day, I wanted to stay positive and keep the groups morale up to avoid making the day drag longer if we were all upset with more work
- I suggested cleaning up slightly earlier to avoid us staying back later than necessary. Of course not every worker was gonna fully commit to that, but I like going home earlier so I can jog before bed, so me and the coworkers who agreed would start 30 minutes earlier and reorganize the sections of the absent coworker.
- We were able to finish on time like a normal day and the closing group didn't have a sour mood leaving the store, so an experience like that taught me to stay open minded to any sudden changes in a scenario and make the best of it because if we had just given up we would've left 30 minutes later and all been upset at our coworker who couldn't control when her exam was

Write one about a class project

- During a database project I had one a classmate, we had to develop a system with multiple interconnected tables and write complex queries. For our initial test run with our professor, he said the queries were taking too long to execute.
- Both my teammate and I needed to address performance issues without either of us getting defensive or blaming the other, however the project was an end of semester deadline assignment, so both of us were dealing with other projects and exams at the time so tension was high
- Immediately after the professor evaluation, we discussed when we could meet up and just work on it to avoid pushing it back, and settled on just working on it that day and another day a week or two afterwards. During our small meets we tried to split up the problem and tackle different things to optimize our code like different JOIN approaches or changing the query structure.
- After being more on top with touching base and working on the project together instead of separately and sending code to each other, we submitted the project on time and managed to balance our other ongoing assignments alongside it.

Inheritance

- child class inherits properties or behaviors of a parent class using extends
- An example I have of using Inheritance in a project would be in my data systems class we designed a basic banking simulation that had different kinds of bank accounts like savings or checking all inheriting from the bankaccount parent class, so methods like deposit or display and variables like account number or balance would be passed down while savings would have extra variables to determine interest or withdrawal limits

Polymorphism

- Overriding or altering the way a method works in an object
- In the same banking app we would change the withdraw method for each child class of bankaccount, with the savings account having limited withdrawals, while checking could allow for overdrafting.

Encapsulation

- wrapping code and data so that it stays inside each class and isn't exposed globally
- I used encapsulation in my Client-Server project by privatizing certain variables that pertained to their connection. Since this is the logic that handles sending and receiving I wanted the variables and methods to only work together and not be affected by anything outside

Abstraction

- Hiding complex details and exposing only essential features
- For the user interface of the application we used abstraction to simplify accessing information. Since we already had the code written in a different program, the GUI code just needed to mention the method and call it so that it can display the correct information on screen for the user.

Questions for Interviewer

- I looked over your linked in and saw you've been working with Cognizant for almost 5 years and was wondering what aspects of Cognizant keeps your experience there fulfilling
- Under the website's culture and values, Do the right thing was a key value that really stood out to me and I was wondering what that means to you and how does that affect your day to day working at Cognizant?

week 1 core java found

week 2 sql java testing aws

week 3 nosql advanced java

week 4 web dev

do you find a lot of your teams use similar technologies?

fillbuster?

Revature start date - 10/27/2025