

Teaching Statement: Critical Vulnerability as Pedagogy in Computer Science

Computer Science is often marketed as a field of pure logic a world of objective truths, elegant algorithms, and meritocratic success. However, after thirteen years in the industry, I have seen firsthand that code is never neutral. Every technical decision we make carries social, ethical, and political consequences. My interest in Computer Science is driven by its immense capacity for innovation, but my passion for teaching is driven by the need to dismantle the epistemic mastery that often masks deep-seated patterns of exclusion within the discipline.

I do not view the classroom as a site for the mere transmission of data from expert to novice. Instead, I aim to transform the CS environment into a space of experiential understanding. For many students particularly those from marginalized backgrounds entering a high-tech space can trigger a profound cultural dissonance. They often feel to leave their identities at the door in favor of a dominant persona. My goal is to create a welcoming environment where students can navigate these spaces without losing themselves, bringing their full identities, racial, religious, social, and ethical, into the technical process.

In my classroom, I'd consciously step away from the role of the traditional, all-knowing lecturer. Instead, I'd position myself as a facilitator of learning. This shift is grounded in the **4-Stage Model of Intergroup Dialogue (IGD)**. I view the relationship between learner and instructor as a collaborative partnership where both parties are responsible for validating each other's wisdom. I acknowledge that while I may hold technical expertise, my students hold lived expertise that is equally vital to the classroom ecosystem.

To support this, I'd employ **Critical Dialogic Practice (CDP)**. This is not just group work; it is a pedagogical commitment to radical listening and the sharing of personal narratives. By integrating CDP into technical curricula, I'd allow students to see their peers as whole human beings rather than just competitors for a grade. This approach is particularly personal to me. As a member of a nano-minority, I am acutely sensitive to how demographic factors shape a student's sense of belonging. I know what it feels like to be the only person in the room who looks like me, and I know the cognitive load that comes with wondering if you truly belong in a high-stakes engineering environment.

To address these power dynamics head-on, I'd use tools like the **Social Identity Wheel** and **Community Agreements** during the very first week of a course. We'd don't just talk about code; we'd talk about how we will treat one another. We'd establish the classroom as a non-affirming space for dominant oppressive narratives, ensuring that every student feels safe enough to take the intellectual risks necessary for growth.

My 13 years in industry have taught me that technical skill is only half the battle. A brilliant engineer who cannot communicate relationally or reflect on the impact of their work is a liability to society. Therefore, my primary goal is for students to develop both technical proficiency and a critical consciousness. For any given project, for example, designing an algorithm or building a database, we don't stop at the technical implementation. We engage in a dialogic debrief that explores the logic and theoretical constraints and the human impact and the relational responsibility involved in the work. We might ask: Who is excluded by this data model? Whose

voice was missing in the design phase? How did our team's internal power dynamics influence the final product?

I'd practice **critical vulnerability as praxis**, meaning I'd be open with my students about my own pedagogical choices and the biases I carry. I'd acknowledge that I am not a neutral observer; I am a participant in their journey. This vulnerability models for them that mastery does not mean perfection or certainty. It means having the reflexivity to realize when a move technical or social needs to be adjusted.

Assessment in my courses goes beyond the binary of working vs. broken code. I'd utilize **Reflective Structured Dialogue** and iterative feedback loops that mimic professional reflexivity. Assignments often include reflective papers on positionality, where students analyze how their own background influences their approach to problem-solving. These assessments are a rigorous measurement of deeper meaning and experiential learning. They reflect the reality of modern engineering as a collaborative, ethical practice where the soft skills of empathy and compassion are actually the hardest skills to master.

I'd also use assessment results to interrogate my own interpretive constraints. If a group of students is struggling, I don't look first at their lack of effort; I look at my own teaching moves. I'd use student feedback regarding the pace of their lives and their sense of support to adjust our classroom trajectory in real-time. This is **Strategic Questioning** in action using inquiry not to trap a student in a wrong answer, but to uncover the experiential concerns that might be blocking their path to success.

My foundation is built on thirteen years of industry engineering, but my future is in the classroom. I am currently refining my ability to be a **multi-partial facilitator**. This means I'd not strive for neutrality which often defaults to supporting the status quo but rather I am partial to everyone in the room. I'd work to balance social power by amplifying voices that are typically silenced and gently challenging those that dominate.

My ultimate aim is to prepare students to resist the exclusionary professional networking norms they will encounter in their careers. I want them to leave my classroom not just as proficient coders, but as empathetic leaders who understand that their relational accountability is just as important as their technical output. By focusing on a deep sense of belonging and critical consciousness, I hope to foster an environment where the next generation of engineers sees technology as a tool for liberation rather than a mask for power.