Introduction

As I approach the end of my undergraduate journey at Iowa State University, I reflect on the transformative experiences that have shaped me into the engineer I am today. When I arrived in Ames in August 2022, I was eager but uncertain about what computer engineering meant beyond textbooks and lectures. My education has extended beyond the classroom walls, encompassing technical mastery, professional development, ethical reasoning, and a deep appreciation for lifelong learning. This reflection captures the essence of my growth through coursework, industry experience, personal projects, and co-curricular involvement, all of which have prepared me to tackle complex engineering challenges in an interconnected world.

Understanding Engineering's Broader Impact

My coursework and experiences at Iowa State have provided balanced exposure to engineering practice's societal, economic, global, and environmental dimensions. Through courses like Software Project Management and Construction of User Interfaces, I learned that engineering solutions must serve diverse stakeholders with varying needs, abilities, and cultural contexts. My minor in Spanish has further enhanced my ability to communicate across cultural boundaries, recognizing that technology increasingly serves global markets where linguistic and cultural sensitivity matter.

The economic implications of engineering decisions became particularly clear during my apprenticeship at Source Allies. I automated WiFi voucher generation and upgraded AWS RDS databases, reducing monthly operational costs by 50%. This experience taught me that elegant technical solutions must also make business sense. Cost efficiency, scalability, and maintainability are fundamental design considerations. My work demonstrated how thoughtful engineering can directly impact an organization's bottom line while improving employee productivity, as evidenced by saving 6+ hours per week of manual IT work.

Environmental considerations have also shaped my engineering perspective. Through volunteer work with Students Helping Our Rescue Animals and discussions in various technical courses, I've become increasingly aware of cloud computing and data centers' energy consumption and carbon footprint. When designing the Project Helper AI Chatbot with AWS Bedrock and Lambda functions, I considered serverless architecture for its scalability and potential to reduce idle resource consumption compared to always-on server infrastructure.

Perhaps most importantly, my experiences have heightened my awareness of technology's societal impact. The Konnect social media application project forced my team to grapple with questions about data privacy, user safety, and the ethical implications of real-time messaging platforms. As we implemented user authentication and settings, we discussed how design choices affect user autonomy and privacy. These conversations, combined with coursework in embedded systems and operating systems, have instilled in me a responsibility to consider how my technical decisions affect real people's lives, security, and wellbeing.

Designing Systems and Solving Engineering Problems

My ability to design systems, recognize engineering challenges, and formulate solutions has developed significantly through both academic projects and professional experience. The progression from basic programming in Com S 227 to complex system design in CprE 381 (Computer Organization & Design) and CprE 308 (Operating Systems) provided the foundational knowledge necessary to tackle real-world problems.

One concrete example of this growth is my Project Helper AI Chatbot. I identified a genuine pain point, developers' difficulty understanding unfamiliar codebases, and designed a comprehensive solution. I architected a full-stack system integrating AWS Bedrock with LangChain for natural language processing, designed a PostgreSQL database with GitHub webhooks for repository synchronization, and implemented a RAG (Retrieval-Augmented Generation) system with vector embeddings. This project required me to formulate the problem clearly, design a multi-component system addressing that problem, and solve numerous sub-problems, including data synchronization, context retrieval, and response generation.

At Source Allies, I encountered and solved multiple engineering problems that required systems thinking. When tasked with migrating case studies to a GraphQL architecture, I recognized that the challenge wasn't merely technical migration but creating a unified metadata access layer that would serve multiple client applications efficiently. I designed and implemented an AWS Lambda subgraph that integrated seamlessly with the existing federation, demonstrating my ability to work within complex distributed systems while considering scalability, maintainability, and developer experience.

The FinAntes personal finance application exemplifies my growth in full-stack system design. I architected a RESTful API with JWT authentication, designed a MongoDB schema optimizing for transactional data and analytical queries, and created a tiered subscription system with role-based access control. This required understanding the interplay between front-end requirements, back-end logic, database design, and security considerations.

The Konnect Android application further demonstrated my problem-solving abilities. Beyond writing code, I resolved over 80% of reported bugs through systematic debugging, optimized API calls to reduce response times by 50%. I coordinated my three teammates' UI/UX and API integration efforts. This experience taught me that solving engineering problems often requires technical skills, leadership, communication, and the ability to see patterns across seemingly unrelated issues.

Leveraging External Resources and Independent Learning

My education has taught me that an engineer's most valuable skill is learning independently and applying knowledge from diverse sources. I've consistently demonstrated this ability by going beyond classroom requirements and seeking external resources to deepen my understanding and solve complex problems.

Earning my AWS Certified Developer - Associate certification exemplifies this commitment to independent learning. Rather than waiting for coursework to cover cloud technologies comprehensively, I proactively studied AWS services, architecture patterns, and best practices through official documentation, online courses, and hands-on experimentation. This certification wasn't required for any course, but has proven invaluable in my apprenticeship and personal projects.

I've regularly consulted professional journals, technical documentation, and expert communities throughout my projects. When implementing the RAG system for my AI chatbot, I extensively studied recent research papers on vector embeddings and retrieval strategies, consulted LangChain's documentation, and engaged with the developer community through GitHub discussions and Stack Overflow. The initial course material provided foundational knowledge, but building a production-quality system required synthesizing information from multiple authoritative sources.

My apprenticeship at Source Allies provided exposure to industry experts and real-world problem-solving approaches that differed significantly from academic contexts. I learned about infrastructure as code through Terraform, absorbed best practices for serverless architecture design, and observed how experienced engineers approach system optimization. I actively sought mentorship from senior developers, asking questions that went beyond my immediate tasks to understand architectural decisions and design tradeoffs.

The coursework in Software Systems for Big Data Analytics particularly encouraged external exploration. I supplemented lectures by reading industry case studies from companies like Netflix, Spotify, and Amazon, understanding how they handle scale challenges that dwarf academic examples. I also explored open-source projects to see how theoretical concepts from Design and Analysis of Algorithms manifest in production codebases.

My involvement in organizing Source Allies' internal hackathon required me to research event planning, budgeting, and logistics. These skills were not taught in engineering courses but are essential for professional success. I consulted with event coordinators, reviewed case studies of successful hackathons, and adapted best practices to our specific context, demonstrating that engineering education extends beyond technical domains.

Engagement Beyond the Classroom

My involvement in co-curricular and extracurricular activities has been instrumental in developing professional skills, expanding my network, and understanding the broader context of engineering practice. These experiences have complemented my academic learning and prepared me for professional engineering work's collaborative, multifaceted nature.

Serving as Treasurer for Wallace Hall taught me financial responsibility, organizational leadership, and the importance of transparent communication. These skills are directly applied to managing engineering projects and resources. Balancing budgets, planning expenditures, and presenting financial reports to residents required attention to detail and clear communication, both critical engineering competencies.

My role as Hackathon Organizer at Source Allies represented a significant leadership opportunity. I coordinated logistics for the entire event, managed budgets, recruited participants, and facilitated team formation. This experience taught me about project management in a way that complemented my Software Project Management coursework, but with real stakes and real-time problem-solving. I learned to anticipate challenges, adapt quickly when plans changed, and create an environment where creativity and collaboration flourished.

Volunteering with Students Helping Our Rescue Animals provided a perspective on technology's potential for social good. Organizing fundraising events and animal care activities reminded me that engineering skills can serve broader humanitarian purposes. This experience has influenced my interest in

leveraging technology for non-profit organizations and social impact initiatives, recognizing that engineering expertise contributes to community wellbeing.

My apprenticeship experience itself served as crucial co-curricular learning. Working in a professional agile environment with Scrum, Extreme Programming, and Kanban methodologies provided practical context for concepts introduced in coursework. I participated in daily standups, sprint planning, retrospectives, and pair programming sessions. I experienced firsthand how collaborative software development differs from academic group projects. The apprenticeship also exposed me to professional tools, workflows, and standards that don't typically appear in classroom settings but are essential for industry success.

These activities collectively developed my professional identity beyond technical competence. They taught me communication skills across diverse audiences (from residence hall residents to C-suite executives at Source Allies), ethical reasoning in real-world contexts, and the importance of building professional relationships and networks. Each experience reinforced that engineering is fundamentally a collaborative, human-centered discipline.

Embracing Lifelong Learning and Professional Development

Perhaps the most crucial realization from my Iowa State experience is that my undergraduate degree represents my engineering education's beginning, not the culmination. The field of computer engineering evolves rapidly. The technologies I learned freshman year have been superseded, and the tools I used daily in my apprenticeship didn't exist when I started college. This reality has instilled a deep appreciation for continuous learning and professional development.

My short-term goals include securing a full-time software engineering position in New York City, where I can apply and expand my cloud computing and AI/ML expertise. I'm particularly interested in roles involving scalable system design and intelligent applications, building on my Project Helper AI Chatbot work. However, achieving these goals requires staying current with emerging technologies, frameworks, and best practices.

Long-term, I aspire to technical leadership roles where I can architect complex systems and mentor junior engineers. Achieving this vision will require continuously updating my skills, pursuing graduate education or specialized certifications, and deepening my understanding of system design patterns, distributed computing, and emerging paradigms like edge computing and quantum computing.

The rapid evolution of AI and machine learning particularly excites me. My work with AWS Bedrock and LangChain represents only a surface-level understanding of these powerful technologies. I'm committed to deepening my expertise through online courses, research papers, and hands-on experimentation. The democratization of AI through cloud services means that staying competitive requires understanding how to use these tools and when to apply them, their limitations, and their ethical implications.

My AWS certification taught me the value of structured learning paths and industry-recognized credentials. I plan to pursue additional certifications in cloud architecture, machine learning, and cybersecurity. I recognize that these credentials validate my expertise and provide frameworks for systematic skill development. More importantly, the process of certification study exposes me to industry best practices and standards that might not emerge through project work alone.

I've also learned that professional growth requires diverse learning strategies. Technical blogs, podcasts, conference talks, open-source contributions, and side projects all serve different learning

purposes. Reading about new technologies provides breadth; implementing them in projects develops depth. Attending meetups and conferences (which I plan to do more of post-graduation) exposes me to how other engineers approach similar problems and helps me understand industry trends before they become mainstream.

The responsibility for my own learning has become clear through my Iowa State experience. Professors and courses provide structure and guidance, but genuine mastery requires self-directed exploration. My most meaningful learning has often occurred while debugging challenging issues at 2 AM, exploring documentation for libraries not covered in class, or discussing architectural tradeoffs with peers and mentors. This realization empowers me. I'm not dependent on formal education to grow as an engineer.

Finally, I recognize that technical skills alone don't ensure professional success. Communication, collaboration, ethical reasoning, and business acumen require continuous development. I plan to seek opportunities to present at technical conferences, write technical blog posts explaining complex concepts, and engage in mentorship. All these activities will deepen my understanding while contributing to the broader engineering community.

Conclusion

Reflecting on my four years at Iowa State University, I'm struck by how much my understanding of computer engineering has evolved. I arrived viewing engineering primarily as problem-solving through code; I'm leaving with a holistic appreciation for engineering as a discipline that balances technical excellence with ethical responsibility, economic viability, environmental consciousness, and societal impact.

The curriculum provided essential technical competence, from foundational mathematics to advanced topics like embedded systems and algorithm analysis. However, my most transformative growth occurred at the intersections, where classroom theory met industry practice during my apprenticeship, individual projects demanded skills across multiple technical domains, and co-curricular activities developed leadership and communication abilities that complement technical expertise.

As I prepare for graduation and a career in software engineering, I carry forward not just knowledge of specific technologies or frameworks, which will inevitably become obsolete, but rather a mindset: intellectual curiosity, commitment to continuous learning, appreciation for diverse perspectives, and understanding that excellent engineering serves human needs within complex constraints. These enduring lessons from Iowa State will guide my professional journey, enabling me to adapt to technological change, tackle increasingly complex challenges, and contribute meaningfully to the engineering profession and society.

The journey from freshman orientation in Engr 101 to this senior portfolio has been challenging, rewarding, and transformative. I'm grateful for the educators who challenged me, the peers who collaborated with me, the mentors who guided me, and the opportunities Iowa State provided to grow both technically and personally. As I move forward, I do so with confidence in my abilities, humility about how much remains to learn, and excitement about contributing to technology's role in shaping a better future.