

Sachin Patel

## **A Reflective Journey: Navigating Your Cumulative Experience at Iowa State University**

At Iowa State University, the most I've learned about design, problem-solving, and teamwork was from the many hands-on labs and projects given in the engineering courses. Every semester in the computer engineering curriculum, I have taken one or more classes with a lab or project incorporated into the course. One of my favorites of these courses is CPRE 288: Embedded Systems, where we program a Roomba's sensors and motors to perform various tasks in the lab. I had the best time problem-solving and cooperating with teammates in this class. The course gives you enough direction for the labs and plenty of room to experiment. Then, for the final project, instead of holding your hand, you are given a mission that your Roomba must complete, which in our case was navigating through an obstacle course, and it is up to you to implement all of what you learned in class and previous labs to program the Roomba to complete the mission. Based on my internships, this accurately reflects what happens in the field: you are given a problem, and it is up to you to figure out how to solve it. Teamwork was essential in this class, as there is a lot to implement, and it is necessary to split up tasks effectively. This class also made me comfortable to take the leadership position in group projects, which I've done for courses after this one, and the leadership skills I've learned from this will be an asset to me in the field.

Another class that prepared me well for the field was COM S 309, where we worked in groups to design and build a full-stack application. I have grown to enjoy the software engineering side of my major more than the electrical, and this class taught me effective project management and team collaboration in the software engineering space. We learned how to use project management tools and methodologies used in the field and modern tools, programming languages, and frameworks that many

companies use. Here, teamwork is not optional; every team member has a feature to implement for the application before the milestone deadlines. Of all the three internships I had throughout college, this class was the most informational about how project design and development is in the field. The course also taught me about the ethics of a project or development cycle and how it's important for a software engineer to follow them since it could cost the company if the code of ethics is broken.

There are, of course, many other courses that I've taken that have been useful to me. I found it important to work hard in all of them, even if I wasn't enjoying the class, because there might come a time when I need the knowledge or skills developed in the course. Also, every class can improve my problem-solving skills, which are important in the field. There may be a problem I have no interest in solving, but it may become a larger issue in the future. For example, I, like many software engineers, don't like to document my code. But it is important to do so because other developers may try to add to your code in the future, or you may come back to your code a while later. If you don't have any documentation, it will be hard for anyone to understand how the program works.

If I ever needed help for a course beyond the classroom, I would communicate with my peers taking the same course or go to office hours. Having a peer network is important in college because many of us are trying to reach the same goal, and not many people know how to achieve every step to reach the goal. Outside of courses, Iowa State offers a variety of resources and extracurriculars to expand a student's knowledge beyond what is taught in class. I have been a part of a few student organizations where I have developed projects that have taught me many technical and soft skills that I wouldn't have learned in class. I have been a member of the Cardinal Space Mining Club for my entire time at Iowa State and have learned a lot about designing and developing robotic systems. I have learned about systems engineering, which is important for project management and designing systems. I learned how to do risk analysis, interfacing, and creating a solid design before starting development. I

also had the opportunity to be the controls lead for the club and gained a lot of leadership experience. This club has reinforced my love for robotics and is why I would like to continue in robotics after college.

If I were to do college over again, I would change a few things. I would have gotten involved in more non-academic clubs and activities and not stressed over my grades as much. From my experience, having a really high GPA won't necessarily impress employers that much; it is just important to keep it above a certain threshold. Experiences cultivated outside of class will give more things to discuss during an interview than having a high GPA. Also, people may not get to do some things after college, like studying abroad, intramural sports, or interesting clubs.

Recently, I've been learning more about machine learning. Machine learning is a large field right now, especially with the introduction of Chat GPT and other online AI services. Machine learning is also heavily used in other applications like self-driving cars and autonomous robotics. Given that I see myself in the field of robotics, AI/ML is a great thing to learn, so I took COM S 474: intro to Machine Learning and chose a senior design project that revolved around machine learning. At first, machine learning was a difficult topic since it was more theoretical. However, understanding the theory behind machine learning concepts made it much easier to implement for an application. It is very important to understand the theory because it's easy to rush to make a machine learning implementation with the vast number of easy-to-use resources that can give you a machine learning algorithm in barely any code, but the underlying mechanics behind the algorithm are abstracted away. One example where I transformed a theoretical concept into a real implementation was for our first homework assignment. We had to implement the perceptron learning algorithm to recognize the difference between a hand-drawn one and five. Once I understood the theory behind the algorithm from the lectures, it was easy to implement in code.

After college, I will work in fields I am most passionate about since I will be more determined to learn new things in that field. I will have more opportunities to specialize in what I am interested in. To acquire new knowledge, I will use the research techniques I have learned in class and student organizations. Unfortunately, I won't have a professor to lecture me on the topic, but there are plenty of online resources that will, and I can also call for help from a mentor or subject matter expert.