Course Reflection Essay

CSCI 441 VA: Software Engineering

Jul 15, 2025

Author: Conner Erckert

The major technical difficulties my team encountered centered around connecting the frontend to the backend of our program. While I had previously taken courses focused separately on frontend and backend development, I had not yet learned how to integrate the two. We decided to use React for a dynamic, scalable frontend, but since we had never used React before, there was a great deal of trial and error involved in getting it functional. Another significant challenge was creating unit, integration, and documentation tests. We had not worked with these before, so we had to learn not only how to write and run the tests but also where to organize them in the project's file structure. Managing the package.json files and ensuring the correct dependencies were installed and configured was an additional hurdle.

Despite these challenges, I felt that my previous courses, especially those involving API development, had prepared me reasonably well for this course. However, I would strongly recommend taking this course during a full semester rather than an accelerated summer session. The project requires substantial time and effort, and a longer timeline would have allowed for a deeper understanding of the material.

While this course did not directly teach how to combine frontend and backend technologies, it did provide methodologies for structuring software projects. Techniques such as design patterns helped us better plan and understand the connections between components. Additionally, the UML notation and diagramming techniques we learned improved my understanding of the program and its required functionalities. However, not all diagram types felt necessary, some seemed to present the same information in slightly different formats, which added clutter to the documentation rather than clarity.

I also came to appreciate the difference between working on a predefined project versus creating your own. A predefined project provides a clear direction and scope, making it easier to plan and execute. In contrast, designing our own project allowed for more freedom and creativity. I personally enjoyed this freedom, as it gave us the opportunity to explore new ideas and propose novel approaches to solving a chosen problem without strict limitations.

The biggest challenge of working on a team-based software project was coordinating the backend functionality with the frontend requirements. Much of my debugging time was spent aligning frontend API calls with the backend's data formats and adjusting the backend to provide new CRUD operations as the frontend evolved. Occasionally, there were communication gaps, sometimes the frontend requested functionality that the backend either didn't support or didn't need to support. These issues were particularly common during the data model design phase, and they reinforced the importance of clear communication between team members.

Despite these challenges, working as a team had significant benefits. The division of labor allowed us to be more efficient; for example, while one team member focused on learning and implementing React, the other continued backend development. This was especially valuable given the tight summer schedule. Having multiple perspectives also made debugging easier, when one of us hit a roadblock, the other could step in with a fresh approach. The differing viewpoints of a frontend-focused and backend-focused developer often led to faster problem resolution. One area where more upfront knowledge would have been helpful was in understanding how to connect frontend and backend servers, particularly with the package.json configurations in Express. We weren't initially aware of all the required files and dependencies and had to figure it out as we went along.

Overall, this course significantly changed how I approach software projects. I now have a much deeper understanding of the processes behind both frontend and backend development, as well as the broader software engineering workflow. I gained an appreciation for the importance of documentation, iterative development, and building projects step by step. While the initial reporting requirements felt intimidating, working as a team and tackling the project piece by piece made the process manageable, even within the shortened summer semester. Although I don't feel I fully mastered every topic. I now feel far better prepared for future projects and software engineering challenges, both academically and professionally.