# Reflection on Programming Experience

Project Title: Accessible Campus Navigation System

Team Size: 6 members

Technologies Used: React Native, GitHub, Figma, Miro, Adobe XD

Over the past few weeks, working on this project has been a mix of challenges, learning, and growth. Our goal was clear from the beginning: to create an app that helps students with disabilities navigate campus more easily. While the idea sounded straightforward, turning it into a real and useful system was much more complex — and more rewarding — than I expected.

## Learning by Doing

This project really pushed me out of my comfort zone. I worked mostly on the front-end using React Native, which I had some experience with, but never to this depth. I got the chance to build components from scratch, experiment with layouts, and make sure everything was accessible. I learned how to use React Hook Form to handle inputs and validations properly, and how to style components using Styled Components while keeping accessibility in mind.

What stood out the most was how much we focused on making the app user-friendly for everyone — not just in looks, but in actual functionality. Features like voice input, screen reader support, and color contrast options weren’t just 'nice to have' — they were essential. This changed how I approached UI design and gave me a much better understanding of what inclusive design really means.

## Problems, Fixes & Small Victories

Like with any real project, things didn’t always go smoothly. One major issue was the GPS functionality for obstacle reporting. It didn’t always detect the user’s location, especially when permissions weren’t granted or signals were weak. After a few failed attempts, we decided to add a manual location picker as a backup. It wasn’t part of the original plan, but it ended up making the app more reliable and flexible.

Another frustrating moment came when we were trying to build out the route customization feature. Users needed to be able to choose paths that avoided stairs or had gentler slopes, but our system kept crashing when people entered invalid inputs. Eventually, we added better error handling and validation, and the feature started working smoothly. These small victories added up and gave us confidence.

## Collaboration & Tools

We used GitHub throughout the project to manage our code. At first, there were a few hiccups — merge conflicts, people forgetting to push their changes — but over time we got into a good rhythm. It taught me how important version control is in team projects, and I’m much more comfortable using Git now.

For the design side, we created wireframes using Figma and mapped out how users would move through the app with Miro. These tools helped us stay aligned as a team, especially when we couldn’t always meet in person. We also had regular team meetings where we discussed progress and challenges, which really helped keep everyone focused and on the same page.

## Takeaways

This project was different from anything I’ve worked on before because it wasn’t just about making something cool — it was about solving a real problem for real people. It made me think more carefully about design decisions, data privacy (we used AES-256 encryption for that), and how to handle user input with care.

I also learned a lot about communication — both in terms of working with teammates and thinking about how users interact with the app. It’s easy to forget that not everyone experiences a campus the same way, and this project reminded me of the importance of listening and designing with empathy.

## Final Thoughts

Looking back, I’m proud of what we’ve accomplished so far. We started with a big idea and turned it into something practical and meaningful. The experience helped me grow as a programmer, a teammate, and most importantly, as someone who cares about making tech more inclusive.