Throughout this program, I've worked on projects that helped shape my skills and prepared me for a career in software development. From early assignments to my final ePortfolio, I've built, tested, and improved applications while learning how to write efficient, scalable, and well-structured code. This capstone project was an opportunity to refine my abilities, push my understanding of software engineering, algorithms, and databases, and apply real-world programming techniques.

One of the biggest takeaways from my coursework is how important collaboration is in software development. Working with different tools, from GitHub to SQL databases, showed me how essential version control, documentation, and clear communication are when working in a team. Even when developing solo, I made sure to structure my code in a way that would be easy to understand and maintain, as if someone else would be reading and improving it later.

Communication is another key skill I developed. Whether it was through code reviews, written documentation, or presenting my work, I learned how to explain technical concepts in a way that both developers and non-developers can understand. This will be especially useful when working with teams, stakeholders, or clients who may not have a deep technical background but still need to understand what's happening under the hood.

In data structures and algorithms, I focused on optimizing performance, improving how my applications handle large amounts of data, and making sure processes run efficiently. One major improvement in my event-tracking app was making sure it doesn't reload all events when only one is updated. Instead, I modified the way data is handled so that only the changed event updates, making the app more responsive. These kinds of optimizations make a big difference in real-world applications, where speed and efficiency matter.

For software engineering and databases, I transitioned from using raw SQLite queries to Room Database, which made data handling cleaner and more secure. I also added LiveData and ViewModel, improving the way data is stored and displayed without causing unnecessary UI refreshes. These enhancements follow best practices in modern Android development and will help me in future mobile app projects.

Security is another key area I considered, even for a small-scale app. Storing user credentials securely, making sure data is protected, and following good authentication practices are all things that I've learned throughout the program. While my projects weren't security-focused, I still made sure to apply basic security principles to minimize vulnerabilities.

My ePortfolio brings together all of these skills by showcasing my event-tracking app, where I enhanced it in three key areas: software design, algorithms, and databases. Each enhancement helped refine the app and improve my coding abilities. This experience reinforced how important it is to balance clean architecture, efficiency, and usability in real-world projects. Going forward, I plan to continue learning and improving, especially in mobile development and full-stack applications. The projects I worked on in this program gave me a strong foundation, and I'm excited to keep building on it as I move into the next stage of my career.