**REPORT**

I enjoy software development, although I didn't have prior experience with Python for software development. However, after taking this course, I have learned a lot, including how to use pytest, coverage, and how to create test cases, among other topics. You also introduced me to environments, Docker (which I had no previous knowledge of), and GitHub Actions. This course has been very practical, and I have gained significant knowledge. I also learned how to identify errors and bugs and how to approach solving them. Additionally, you taught me about logging and other important features.

The introduction to Docker was a game-changer. I had no prior experience with containerization, but through this project, I’ve gained a solid understanding of how Docker can simplify deployment and create a consistent environment across different systems. Using Docker to containerize this project, along with setting up GitHub Actions for CI/CD, has opened my eyes to the importance of automation and best practices for deployment in modern software development workflows.

***About Project:-***

This project, built with FastAPI and PostgreSQL, focuses on user management functionality, including registration, authentication, password management, and profile management. The core features include email verification, role-based access control, and strong password validation during user registration, updates, and resets. I implemented a comprehensive solution to handle unique nickname generation, password validation, and professional status upgrades, which allow managers and admins to update users' profiles and promote them to professional status. A key enhancement in this project was improving user profile management, enabling users to update their profile fields, and allowing managers or admins to upgrade a user to professional status. Furthermore, I ensured that only users with verified emails could update their profiles, and I added proper URL validation for profile-related links like social media profiles.

Additionally, the project addressed real-world challenges such as preventing nickname collisions by generating unique nicknames, implementing enhanced password validation for stronger security, and managing different user roles effectively. The ability to handle these complex functionalities, while ensuring a seamless user experience, was a significant achievement. This project also integrates automated testing to ensure the system works as intended, which I gained experience with through pytest and coverage.

Through this project, I gained hands-on experience with important software development tools and best practices. I learned how to structure and test my code with pytest and coverage, ensuring robust test coverage and reliable functionality. Dockerization and GitHub Actions for CI/CD integration were key takeaways, allowing me to streamline deployment and automate testing processes. I also tackled several real-world challenges like managing user roles, preventing nickname collisions, and enhancing password security, all while refining my debugging and error-handling skills.

This project has greatly improved my software development abilities, and I feel more confident in my skills, particularly in testing, deployment, and handling complex user scenarios. The learning process has been incredibly practical and has deepened my understanding of Python development, making this project a crucial milestone in my software development journey.