IS601 Reflection Document

This course has been both rewarding and challenging. I gained practical expertise with code design concepts and industry tools like Docker, GitHub Actions, and FastAPI, WSL2, RestAPI with QRcode API. I was able to progressively develop the abilities required for the midterm and final projects thanks to the well-organized weekly tasks. I am grateful to Professor Keith for creating such an amazing and challenging course. I had little familiarity with Python when I started the class, so I felt nervous by the new tools and techniques. But with regular practice, I was able to better control the learning process. The projects were especially rewarding, evolving from simple tasks to complex ones like multi-core processing and integrating tools such as Faker and MinIO. Even though I faced challenges like debugging and setting up workflows, the experience was incredibly fulfilling.

The course also included collaborative elements that gave me valuable insights into real-world software development. I gained experience with tools for team collaboration and learned about QA roles. These aspects of the course helped improve both my technical skills and my ability to work well in teams. I learned from this course is the importance of perseverance and adaptability. Tasks like setting up MinIO in GitHub Actions or debugging command-line utilities pushed me outside my comfort zone. These challenges taught me to approach problems step-by-step and appreciate the learning process. The feeling of accomplishment after successfully implementing a solution was unmatched.

**Final Term Project**: The User Management System project was a key part of my learning journey. It provided a comprehensive experience that closely resembled real-world software development. I focused on enhancing user profile management by integrating the Profile Picture Upload feature with Minio. I also improved system functionality by adding validation checks, handling edge cases, and ensuring smooth user interactions.

This project helped me understand backend development, database management, and third-party tool integration. I also developed better debugging and problem-solving skills while collaborating with a team. By following best coding practices and using tools like pytest for testing, I made sure the features were reliable and scalable. Overall, this project enhanced my ability to deliver efficient and user-friendly solutions.

The profile picture upload feature is designed to boost user engagement by allowing users to personalize their accounts. Using Minio, a distributed object storage system, ensures secure and efficient management of profile pictures. The integration of this feature demonstrates my ability to use third-party tools effectively while prioritizing security, scalability, and a smooth user experience.

**Links:**

1. **GitHub:** [GitHub Repository](https://github.com/nisha2110/IS601_final_user_management)
2. **DockerHub**: [DockerHub Repository](https://hub.docker.com/repository/docker/nishi2110/is601_final_api/general)
3. **QA Issues:**

1. Fix issue in Docker File to allow build: [[Issue-1 link](https://github.com/nisha2110/IS601_final_user_management/issues/1)]

2. Profile picture URL validation: [[Issue-2 Link]](https://github.com/nisha2110/IS601_final_user_management/issues/3)

3. User ID being passed as None in the user verify email has been resolved: [[Issue-3 Link]](https://github.com/nisha2110/IS601_final_user_management/issues/5)

4. Nickname Assign and Uniqueness in User Registration: [[Issue-4 Link]](https://github.com/nisha2110/IS601_final_user_management/issues/7)

5. Password Validation in User Registration: [[Issue-5 Link]](https://github.com/nisha2110/IS601_final_user_management/issues/9)

1. **Closed Issue:** [[Link Here]](https://github.com/nisha2110/IS601_final_user_management/issues?q=is%3Aissue+is%3Aclosed)
2. **Feature:** **Profile Picture Upload with Minio:**
   * + 1. [Install and config minIO](https://github.com/nisha2110/IS601_final_user_management/issues/13)
       2. [Code integration](https://github.com/nisha2110/IS601_final_user_management/issues/15)
       3. [Profile picture URL fetch and retrieve minIO](https://github.com/nisha2110/IS601_final_user_management/issues/27)
3. **New 10 Test Cases Links:** 
   * + 1. [Test- Picture invalid type](https://github.com/nisha2110/IS601_final_user_management/issues/17)
       2. [Test- picture upload & retrieve successful](https://github.com/nisha2110/IS601_final_user_management/issues/19)
       3. [Test edge case handling for file uploads](https://github.com/nisha2110/IS601_final_user_management/issues/21)
       4. [Test error and exception handling for file operations](https://github.com/nisha2110/IS601_final_user_management/issues/23)

We can guarantee a reliable and easy-to-use profile image upload feature while preserving the security and dependability of the MinIO storage backend by putting these test cases into practice.