Team Project

Software Engineering (Fall 2023)

Demo & Presentation Schedule

- **Demo 1**: 10/31/2023 (for first iteration)
- **Demo 2**: 11/21/2023 (for second iteration)
- **Demo 3**: 12/12/2023 (for third iteration)

Objectives

- The goal of this team project is to collaboratively develop a comprehensive healthcare information system using the `iTrust2` framework.
- Through this project, students will gain practical experience in various aspects of software engineering, including requirements analysis, design, coding, testing, and documentation.

Team Project Setup Guidelines

1. GitLab Account Creation:
   - Register a GitLab account for your team project using your team name at our server.
2. Private Project Creation:
   - Establish a private project named `iTrust2` on GitLab. Ensure that each team member receives a proper invitation with the necessary permissions.
   - Utilize this project for managing all aspects of your work, including implementations, documentation, progress reports, and team meeting minutes.
3. `README.md` Initialization:
   - Create a `README.md` file and provide initial details about your team, including team members with their GitLab IDs and contact information, their experiences, regular meeting times, etc.
   - This file should serve as a central hub for reporting the progress of your team project.
4. Team Contract Submission:
   - Fill out, sign, and upload your completed `Team Contract` to the GitLab project.

iTrust2 Project Setup Guidelines

1. Download iTrust2 Project:
   - Obtain the latest version of the `iTrust2` project from the `main` branch at https://github.com/ncsu-csc326/iTrust2.
2. Read Requirements Documentation:
   - Thoroughly review the documentation available at iTrust2 Requirements for a comprehensive understanding of project specifications.
3. Build the Project Locally:
   - Follow the instructions in Developer's Guide to set up, test, and run `iTrust2` on your local machine.

Project Requirements and Guidelines

1. Use Case Implementation:
    - Implement a total of 6 Use Cases (UC) throughout the three iterations, including 5 specified UCs and 1 newly proposed UC. Reference the attached 8 UC files, and select 5 UCs within the given iterations.
        - **Mandatory 1**: Complete UC specification and propose a new UC by the end of iteration 1.
        - **Mandatory 2**: Finalize all implementations and testing by the conclusion of iteration 3.
2. Regular Commit:
    - Regularly commit to the project to demonstrate consistent progress. Avoid last-minute implementations to prevent a 20% deduction.
3. Peer Evaluation:
    - Each member's contribution will be assessed through peer evaluation. Active participation is encouraged; avoid being a free-rider.
4. JUnit Test Coverage:
    - Maintain JUnit test coverage of over 80% to ensure comprehensive testing.
5. `README.md` Maintenance:
    - Keep the `README.md` file up-to-date, as it will be a key evaluation point for your team project.
6. Weekly Meetings:
    - Hold regular weekly meetings, and upload meeting minutes as child pages for documentation.
7. Leadership Rotation:
    - Assign team roles with (1) a team leader, (2) a planning leader, and (3) a QA leader. Rotate these roles at each iteration.
8. Team Coordination:
    - Coordinate specific schedules within the team for effective collaboration.

Evaluation Criteria for Iteration 1

1. Web Page Management (`README.md`):
    - Thoroughness and accuracy in maintaining the `README.md` file.
2. New Use Case (UC) Proposal:
    - Quality and innovation in proposing a new Use Case for the project.
3. User Story Development:
    - Effective development and articulation of user stories.
4. Git Commit:
    - Consistency and regularity in Git commits to track progress.
5. Regular Team Meeting:
    - Active participation and contribution during regular team meetings.
6. Role Allocation:
    - Clear and fair allocation of roles within the team.