CIS 344 GROUP PROJECT GUIDELINES

FALL 2024

PROF. YANILDA PERALTA

LEHMAN COLLEGE

1. Project Selection

- **First Step**: Each group must choose a project topic from the provided [Google Sheet](#).

2. Group Collaboration

- **Teamwork**: Collaborate effectively with your group members, dividing tasks and maintaining regular check-ins to ensure the progress of the project.

- **Communication Tools**: Utilize group communication tools (e.g., Slack, Discord, or group emails) for effective collaboration and issue tracking.

- **Project Management**: Implement a project management approach (e.g., kanban board, Trello, or Jira) to organize tasks and deadlines.

3. Application Requirements

- **Framework**: Build your application using PHP for server-side processing, HTML/CSS/JavaScript for the front-end, and MySQL as the database.

- **Frontend**: Ensure that the user interface is user-friendly, responsive, and visually appealing.

- **Backend Processing**: Use PHP for backend processing to handle form submissions, data processing, and interactions with the database.

- **Database**: Design your MySQL database with optimized structures and use **joins** for data retrieval.

4. Documentation

- **Project Report**: Document the entire development process, including:

  o **Project Plan**: Describe your project scope, objectives, and initial design plans.

  o **Technical Details**: Include a step-by-step overview of the development process, code snippets, and the logic behind key features.

  o **Database Design**: Explain your database schema and the use of **joins** for data retrieval.

  o **Security Measures**: Detail the security features and best practices implemented in your PHP application.

- **User Guide**: Create a simple guide explaining how to use the application, with screenshots and step-by-step instructions.

- **GitHub Repository**: Upload your code, including all PHP files, HTML/CSS/JavaScript, and SQL queries, to a public GitHub repository.

  o **Repository Structure**: Maintain a clear folder structure and include a README.md file that outlines how to run and test the application.

o **Commit History**: Ensure that your commit history reflects your development process, with descriptive commit messages.

5. Core Focus Areas

- **Use of Joins**: Implement and optimize SQL joins in your database queries for efficient data retrieval. This will be a significant part of your project evaluation.

- **Security Enhancements**: Apply security best practices, such as:

  o Data sanitization and validation to prevent SQL injection.

  o Use of prepared statements or PDO (PHP Data Objects) for secure database interactions.

  o Secure handling of user input and session management to mitigate common web vulnerabilities (e.g., XSS, CSRF).

6. Grading Criteria

- **Functionality**: The PHP application must meet all functional requirements and run without errors. 30%

- **Design**: The user interface should be clear, attractive, and responsive. 10%

- **Code Quality**: Code should be well-structured, commented, and maintainable. 10%

- **Database Management**: Proper database schema design and the use of **joins** for data retrieval. 10%

- **Security**: Implementation of security measures will play a crucial role in grading. 10%

- **Documentation and Reporting**: Quality of the project report and documentation. 10%

- **GitHub Submission**: Clear and complete code submission in a public GitHub repository with an organized structure. 20%

7. Submission Guidelines

- **Report Submission**: Submit the project report as a PDF via Blackboard by 11.21.24.

- **GitHub Link**: Include the public GitHub repository link in your report and submit it.

- **One submission per team**: Only one submission per team, by one member of the team.

8. Important Notes

- Ensure all group members contribute actively to the project and maintain a record of contributions for potential evaluation.

- Reach out to me if you encounter technical difficulties or have questions regarding the project.