ENPM 809W – Introduction Secure Software Engineering Project (DRAFT)

Project Description:

The goal of this project is to plan, design and implement a simple software application that meets the given (by you) requirements and specifications. This will be done in three phases.

Important Project Deliverables and Due Dates:

Deliverable	Due Date and Time
Project Phase 1	*September 21, 2021 by 11:59 PM*
Project Phase 2	*November 9, 2021 by 11:59 PM*
Project Phase 3	*November 30, 2021 by 11:59 PM*

Grading Criteria

Project Phase	Grading Criteria
Project Phase 1	20% of the project grade. Depends on: • Richness of the security features of the application. Does the application idea implement sufficient security features given below? • Completeness of the SRS specification, Design specification (including any diagrams), Use/Misuse cases. • Timely submission.
Project Phase 2	50% of the project grade. Depends on: • How well you implemented the specified requirements from the Project Phase 1. • Whether you implemented all the security requirements you outlined in the Project Phase 1. • Timely submission.
Project Phase 3	30% of the project grade. Depends on: • The quality of findings from the code review. • The quality of proposed fixes based on the findings. • Timely submission.

Project Phase 1

In Project Phase 1 you will be becoming familiar with the Git source code control system and providing the requirements and design for

Project Phase 1 Part 1: An exercise using Git

Git Source Code Control

- Sign in to https://code.umd.edu/
- Please ensure you have access to the ENPM809WProject-<Your Directory ID> project.

- The first phase of the project involves designing the software you are going to build.

Project Phase 1 Part 2: Application Requirements & Design

Application Design

You are being asked to consult for a customer organization to design, implement and deliver a secure enterprise class application. You can decide what that enterprise application will do and can be very simplistic.

- Your application is required to, and must implement all of the following security features:
  - A database to store sensitive organizational and application data.
  - Users with email identifiers are used to authenticate to the application.
  - Roles and authorizations within the application map to the roles within the customer organization that authorize the user for different features of the application.
- Your application must implement at least two (2) of the following security features:
  - Cryptography, since the customer organization is worried about privacy.
  - Session Management, since the customer organization would like to track users over the course of their session.
  - Logging and error handling, since the customer organization collects those as part of a dashboard.
- Your application must accept input and allow users to administer the application by implementing at least one of the following features:
  - An application programming interface, OR
  - A web interface, OR
  - A Graphical User Interface (GUI) on the platform of your choice
- Your application must also have the following:
  - Tests that ensure application functionality works as designed.
- Your application must be implemented either using C#, Java or PHP.

Submission Criteria

- Create a Microsoft Word document titled 'Software Requirements Specification – ApplicationName.docx' where ApplicationName is the name of your software application.
- Please fill out the document with the software requirements specification (SRS) for your application and include your Name and UMD Email in the title page.
- If you need help with Software Requirements Specifications, please read:
  - https://medium.com/trailblazer-of-salesforce/software-requirements-specification-srs-document-fd9ab103b18
  - https://jelvix.com/blog/software-requirements-specification
- Your document on the SRS must include the following:
  - System Features and Requirements (Describe the environment this system runs on)
  - Functional Requirements Specification (Uses by actors)
  - Feature Functional Requirements (Feature use cases)
  - A section on Security Requirements that addresses security requirements for the application and requires the security features listed in Application Design section above in order to function.
  - Use cases that explain how the features of the application work.
  - Misuse cases that involve the potential misuse of the features listed above.
  - An optional threat model showing threats to the application.

- Create another Microsoft Word document titled 'Design – ApplicationName.docx' where ApplicationName is the name of your software application.
- Please write up a design for how you plan to implement the above software requirements and must include the following:
    - The application design around the use cases and specifications in the SRS.
    - Database design (tables and relationships)
    - If applicable, the application GUI screens based on the specified Use Cases in the SRS.
- If applicable, you may use any of the following applications to help you wireframe or mockup the GUI screens:
    - Mockplus (Free): https://www.mockplus.com/free-wireframing-tool/
    - Wireframe.cc (Free): https://wireframe.cc/
- Add **both** documents to the git repository (referenced and used in Project Phase 1 Part 1 above) under **Phase1** ***branch*** and commit the changes, and push them to https://code.umd.edu. Please submit a text file containing your UMD student id and the commit ID (on two separate lines) of the latest commit you want graded.

Project Phase 2: Project Implementation

In Project Phase 2, you will be implementing the chosen design and building it to the requirements and specifications you provided in your SRS document.

You will also be required to submit documentation on how the software can be installed (any required libraries or platforms that need to be installed), any compilation steps necessary, and a User Guide on how a user might go about using the application.

Submission Criteria

After implementing your code, please add it/implement the code under the **Phase2** ***branch***, commit the changes and push them to https://code.umd.edu. Please include any documentation as part of this same branch as well.

Please submit a text file containing your UMD student id and the commit ID (on two separate lines) of the latest commit you want graded.

Project Phase 3: Code Review

In Project Phase 3, you will be assigned a partner, with whom you will trade your project (by giving access to your code repository) and you will perform a Secure Code Review of your partner's code base. You will be required to submit a document detailing your findings using a given template and format.