Project Peer Review

Group: S-Dawg

Authors: Kevin Stevens, Janna Timmer, Matthew Coker

General Idea

Dr. Siva, our client/customer, wants a website that allows students to review their peers projects and assignments as well as simple or basic integration with Desire 2 Learn.

Participants

* Students - Will have a basic student view, that allows them to review their peers with a subjective score and written description of why they rated it as such
* Administrators - Will have an administrator view and manage privileges among both Students and Professors
* Professors - Will have a faculty view and manage their specific class section, able to see all of students reviews, scores, and has access to tools to analyze trends and history.

Frontend Mechanics

* When the Instructor uploads the class list the class list is shuffled, and the student is assigned the next 3 in the list, and it loops around at the end of the list.
* No two students grade each other
* And each student’s project is reviewed 3 times

Backend Mechanics

* mySQL Workbench may be our ticket for managing the database aspect of our system to run, managing the scores and reviews attributed to each student, per class, per professor.
* One way function that stores hash codes of passwords to exercise some level of basic security

D2L Integration

* D2L has a function to export grades into a text or CSV format, that CSV format will ultimately be the way the system can streamline the process of transferring data without excessive manual input.

Liability Issues

* FERPA, releasing student information, who has access

Functional Requirements

* 2. System must be a website and must store data
* 1. Students must have access to review 3 random other students’ project submissions
* 2. Two students must not be able to review each other
* 1. Professors must be able to view all reviews
* 1. Administrator must be able to authorize professors
* 2. Administrators must be able to add/create new sections
* 3. Professors must be able to export CSV files of review grades
* 3. Professors must be able to import CSV files into the system

Non-Functional Requirements \*Maximum Priority Starts at 1

* 1. System must not violate FERPA
* 5. Website must have user-Friendly UI
* 3. Secure Log-In of Passwords and such
* 4. Must load in under 5 minutes
* 7. Access the Database without it exploding
* 2. Data-Storing with text/CSV files
* 6. Writing in HTML/CSS