

Requirement Analysis

School system for students and faculty to manage their classes.

Jeremy Breese

Need: A method for easily submitting and collecting programming assignments.

Purpose: To create a system which allows students in programming classes to submit programming work. Additionally, the TAs must be able to collect assignments. Finally, the instructors need to be capable of managing the course, its sections, TAs, and assignments.

Client Base: CS Department – faculty, staff, and students

1. Types of users

- Professors of the class.
- Administration, including the registrars office, advisors, and any other official will have access.
- TAs should have access but limited functionality compared to professors.
- Students will have the most limited functionality of all of the other users.

2. Activities

- Professors
 - Professors will have the ability to create a new assignment.
 - They will be able to specify how many points available.
 - They will be able to specify the due date.
 - They will be able to assign an optional IP whitelist for submissions for the case of a lab.
 - **Data**
 - This will create a new entry in the database for the assignment.
 - **Constraints**
 - An assignment with the same name cannot exist.
 - Points, due date, and IP are not required to create the insertion.
 - Professors will have the ability to delete an assignment.
 - **Data**
 - This will find the assignment entry and remove it.
 - **Constraints**
 - The assignment has to exist.
 - Professors and administration will be able to view submissions for any assignment.
 - **Data**
 - This will query all data associated with an assignment.
 - **Constraints**
 - The assignment must exist.
 - The professor and TAs can create sections within a class
 - **Data**

- This will create a new section that users can be associated with, this is a column in the user table.
 - **Constraints**
 - A section with the same cannot exist already.
 - Professors and TAs can assign other users to a section, and TAs can assign themselves to a section.
 - **Data**
 - This will look up the user and then add data to the section column.
 - **Constraints**
 - The specified section must exist.
 - The specified user must exist.
 - Professors and TAs will be able to upload documents to the class document.
 - **Data**
 - This will upload a new document to the database for the entire class.
 - **Constraints**
 - There must be a class to associate it with.
 - TAs and professors will be able to grade assignments
 - **Data**
 - This will fill out the grade column to an assignment.
 - **Constraints**
 - The user must exist.
 - The assignment must exist.
 - Professors, Administration and TAs will be able to view grades by student.
 - They can also see class grade as a whole.
 - Individual assignments are also able to be viewed.
 - **Data**
 - This will lookup a specified student by name or ID and retrieve their grade.
 - **Constraints**
 - The student must exist and the student should have a grade in the class.
 - View entire class grades by assignment
 - View mean, median, high and low score
 - **Data**
 - This will query all grades associated with an assignment.
 - **Constraints**
 - The specified assignment must exist.
- Administration
- Administration will be able to view grades by student. This will return all assignment grades for the specified student.
 - **Data**
 - The query will accept a student name or ID and search for all assignments and grades for them.
 - **Constraints**
 - The student must exist and have grades associated with them.
 - Administration can view grades for whole class.
 - **Data**

- The administrator can enter a class number and see a list of students with grades.
- **Constraints**
 - The class must exist, have students, and assignments.
- Administration will be able to view enrollment list for class.
 - **Data**
 - This will run a query and pull in names for anyone associated with the class.
 - **Constraints**
 - The class must exist.
- TAs
 - TAs are able to open assignment submission.
 - **Data**
 - The assignment will be marked as open in the status column.
 - **Constraints**
 - The assignment must exist.
 - TAs are able to close assignment submission.
 - **Data**
 - The assignment will be marked as closed in the status column.
 - **Constraints**
 - The assignment must exist and be currently open.
 - TAs are able to view each student's assignment in an in browser preview.
 - **Data**
 - The assignment is found in the database with the student ID and then displayed.
 - **Constraints**
 - Both the assignment and the student must exist in the database.
 - TAs are able to assign a grade to each assignment.
 - **Data**
 - The assignment's grade column is filled out.
 - **Constraints**
 - The assignment must exist, as well as the student.
 - TAs are able to edit a grade to each assignment
 - **Data**
 - The assignment's grade column is refilled out.
 - **Constraints**
 - The assignment must exist, and be graded already.
- Students
 - Students are able to submit an assignment.
 - **Data**
 - The assignment is added to the submissions table, and filled out with time, date, name, and IP address if applicable.
 - **Constraints**
 - The assignment has to be created by the professor.

- Students are able to resubmit an assignment if the assignment is still open.
 - **Data**
 - The assignment is added to the submissions table and marked as the most current submission if the assignment is still open for submission.
 - **Constraints**
 - The assignment has to exist and still be open.
- Students are able to view grades for assignments.
 - **Data**
 - This will query their assignments and return the grades with them
 - **Constraints**
 - The student must be signed into their account and have grades associated to their assignments.
- Students are able to see their grade for the class
 - **Data**
 - This will search for their grade in each class they are enrolled in.
 - **Constraints**
 - Student must be signed in and be enrolled in classes.

3. System Requirements

- A server with large amounts of storage, and enough computing power to quickly return queries is required. The server must also run a webserver on it so that everyone is able to access the data. Because there is sensitive information on the server there should also be a up to date security system.