Assignment 3: Requirements Analysis

Spring 2020: CS4320/7320 Software Engineering

Connor Travers

Problem

The system shall provide 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

Users and User Requirements

The user base shall consist of four main categories:

- **Students**
  - Shall be able to
    - Log-in to the student-side system using their University credentials
    - Navigate between enrolled classes
    - View assignments/grades
    - Submit assignments according to the instructor's restrictions (web link , file upload, text)
    - View the names of students in their section
- **Faculty**
  - Shall be able to
    - Log in to the staff-side system using their University credentials
    - Manage the level of access granted to certain users (i.e. promoting a student to TA/staff status and removing this status)
- **Staff**
  - Shall be able to
    - Log in to the staff-side system using their University credentials
    - Manage student submissions and grades
    - Manage assignments for their sections/courses
    - Change the section a student can access (with no official enrollment powers)
    - Manage the level of access granted to certain users (i.e. promoting a student to TA status and removing this status)
    - Remove a student from a certain section or class
    - Manage their student roster
- **TAs**
  - Shall be able to
    - Perform all actions a student can for sections in which they are not a TA
    - Manage student submissions and grades
    - View the student roster

Functional Requirements

The system's functional requirements can be divided into six main categories, and important data shall be underlined.

- **Log in**
  - The system shall be able to
    - Check a user's input credentials and direct them to their courses dashboard
    - Cache a user's credentials for future ease
- **Navigation**
  - The system shall display
    - A user's enrolled/managed courses and sections
- **Assignment Control**
  - The system shall display
    - All assignments for a section/course
    - An assignment's prompt and details (link, document, text)
  - Users shall be able to
    - Expand an assignment to view assignment properties
  - Staff shall be able to
    - Modify, Create, Remove assignments for a section/course
    - Upload documents as assignments
    - Download all submissions for an assignment
    - Assign a due date for an assignment
    - Manage assignment availability to students
  - TA's shall be able to
    - Download all submissions for an assignment
  - Student shall be able to
    - View the due date and requirements of upcoming assignments
- **Assignment Grading**
  - The system shall
    - Display a student's grade
    - Allow authorized users to download all submissions for an assignment
    - Be able to lock submissions to an assignment at the due date
  - Student shall be able to
    - View the result of graded programming work
    - View reuploaded submissions
  - Staff and TA's shall be able to
    - Assign grades to individual assignment submissions
    - Upload graded programming work
    - Attach comments to specific students' grades
    - View grades and submissions of all students in corresponding sections/courses
- **Course Control**
  - Staff and Faculty shall be able to
    - Add and remove students from a section/course
    - Create/Remove a section/course
    - View details of all students enrolled in a course, including past submissions

Continue…

- **Access Control**
    - The system shall be able to
        - Display appropriate options for the user's level of authorization
    - Staff and Faculty shall be able to
        - Modify a section/course's student roster
        - Modify a user's TA status

System Constraints and Requirements

The system will require

- An internet connection
- A stable internet connection for uploads/downloads
- An internet-capable device with browser function

The system must have

- An upload/download failure rate of less than 1%
- Constraints (non-functional) on the activity or the resultant state of the system
- A page load time of less than 1 second on a 5Mbps internet connection
- A submission failure rate of less than .1%
- The ability to support up to 600 users at one time
- Two-factor authentication or equivalent security measure