As a capstone project for Web Development, we're going to combine the various front-end and back-end techniques we've learned over the course to produce a modern application. Specifically, we'll be building our own version of Canvas. You are advised to refer to Canvas for any design and layout questions for this project.

## **Functional Requirements**

- As a user, I should be able to login with username and password
- As a user, I should be able to click on "Forget Password" button to reset my password by answering three security questions correctly.
- As a user, I should be able to sign up by providing the following information
  - Name
  - Email (must by an email)
  - ID (must by a number)
  - Password (5 chars in length, at least 1 number and 1 symbol)
  - confirm password (it must match the password)
  - Account type: Student or Teacher
  - Three security questions (of your choice)
    - Example, question: What is your favorite movie?
    - Answer: free text to be entered by the student
- As a user, upon login to the application, I should see side navigation menu with the following options:
  - My Account
  - Dashboard (default, the page to display after successful login)
  - Courses
  - Settings (for admin users only)
  - Logout
- As a user, I should be able to logout anytime. The user will be redirected to the login page.

- As a user, I should be able to click on My Account to view my personal information such as name, email, student ID.
- As a user, I should see "Edit Profile" button to edit personal information such as name, email, student ID.
- As a user, I should see "Change Password" button to change my password. The website must ask the user for:
  - Current password (it must match the current password)
  - New password (5 chars in length, at least 1 number and 1 symbol)
  - Confirm password (it must match the new password)
- As a user, I should see "Change Security Questions" button to edit my security questions, the user must type in the current password first before allowing to edit the security questions.
- As an admin, I should be able to click on "Settings" inside navigation menu to see the list of registered users, active and inactive users.
  - Admin should be able to filter by user's status, active and inactive
  - Admin should be able to search for a user by typing their name or email
- As an admin, I should be able to click on any user from the list above and perform the following actions:
  - Approve or reject certain registered users (i.e. no one should be able to login without admin's approval, once admins approve, the user's account will become active; otherwise, it's inactive)
  - Attach registered users (student or teacher) to their appropriate classes (it may be one class or multiple classes)
  - Deactivate user.
- As a student, on dashboard page, I should be able to see my activities as follows:
  - "To Do" items: assignments that are due in the next 3 days from current date.
  - "Upcoming" items: assignments that are due after 3 days from current date.
  - "Past" items: to show past due assignments

- As a teacher, on dashboard page, I should be able to see the assignments that needs grading (i.e. past due assignments only)
- For each assignment on the dashboard, the user should see the name of the assignment, the class name, number of points, and the due date.
- As a teacher or student, I should be able to click on "Courses" to see a list of my courses.
- As a user, in "Courses" section, I should be able to the following secondary navigation menu options:
  - Announcements
  - Assignments
  - Grades
- As a student, I should be able view all announcements in the course
- As a student, I should be able to view my grades (grade for each assignment in this course)
- As a student, I should be able to view all available assignments and click on each assignment and submit the assignment by providing my answers in text area.
- As a teacher, I should be able to view all announcements in the course. Also, I should be able to create a new announcement by providing the text area and submit button.
- As a teacher, I should be able to see all students' grades in a list and submit grades for each student for each assignment.
- As a teacher, I should be able to view all available assignments. Also, I should be able to create a new assignment by providing assignment's name, number of points, description in text area.
- As an admin, I should be able to click on "Courses" to see all courses in the system.
- As an admin, in "Courses" section, I should be able to see "Create Course" button to create a course with the following information such as: name, description, capacity, choose the assigned teacher from dropdown list.
- As an admin, I should be able to view following information on the dashboard page:
  - Number of active students in the system

- o   Number of courses in the system
- o   Number of active teachers in the system

## Non-Functional Requirements

- As a user, I should be able to view the website on mobile, tablet and desktop (i.e. the website must by responsive)

- As a user, I should be able to navigate the website without refreshing the entire website (i.e. the website must be a single-page web application. serve a single HTML request on load and do not refresh the page)

- You are required to use flexbox or grid view system to build the skeleton of the website

- All communications with the server must by through an Ajax and the server must accept and receive API requests in JSON format. Content like scripts, stylesheets, and images, files, etc. can be served statically (i.e. no API requests needed)

- All data in this application must be stored in a storage system (e.g. database such as SQLite database)

## API (server) Requirements

The following endpoints are needed for this project:

- General API for any user
    - o   Login
    - o   Logout
    - o   Forget password
    - o   Update logged in user information
    - o   Change password
    - o   Sign up
    - o   Update security questions
    - o   Get logged in user information

- Student role related APIs (student must be able to access general user APIs)
    - o   View all announcements
    - o   View announcement's details
    - o   View my grades for a course

- View all assignments for a course
- View assignment's details
- Submit assignment
- Viewing "to do", "past due", and "upcoming" assignments on dashboard

- Teacher role related APIs (teacher must be able to access general user APIs)
    - student should NOT be able to access teacher's APIs (i.e. unauthorized response must be returned)
    - View all announcements
    - View announcement's details
    - Create an announcement
    - View all students' grades for a course for all assignments
    - Update all students' grades for a course for certain assignment
    - View all assignments for a course
    - Create an assignment for a course
    - View assignments that ready for grading on dashboard

- Admin role related APIs (admin must be able to access general user APIs)
    - student or teacher should NOT be able to access admin APIs (i.e. unauthorized response must be returned)
    - View registered users' information
    - Update registered users' information
    - Approve or reject users
    - Create a new course
    - View all courses
        - View all announcements in a course
        - View assignments in a course
        - View all grades in a course
    - Get dashboard information for admins