

M2: Project Description and Requirements

Project Description:

OpenEdu is a web e-learning platform

OpenEdu aims to be a central location with all the tools a teacher/professor would need to host a class online. There are a plethora of features such as online submissions of documents, the ability to take quizzes written by the teacher, and the ability to communicate between students and teachers/instructors.

User Requirements:

- Sign In/Login to ensure security
- Users should be able to easily navigate through the site.
 - Switching between courses
 - Finding assignments
 - Looking at grades
- The system should offer classes
 - Instructors should be able to have full control of these classes
 - Students can join courses or be added to them
- Instructors can add assignments and quizzes that students can complete
 - Instructors can grade these and add feedback
- Students and Instructors need to be able to message each other
 - In private
 - And as a class
- Instructors need to be able to upload course related materials.
- All users need to be able to update information about themselves
- Instructors need to be able to make announcements to share important information with the class

Functional Requirements:

As an e-learning platform, OpenEdu will be able to offer the following to users:

Security

- Create an account
- A system to login to an account

Navigation

- Student and instructors can switch between the courses that they are enrolled in

Courses

- Student can browse the currently active courses
- Students should be able to request to join courses
- Instructors can Accept/Decline join requests
- Instructors can manually add users to courses
- Instructors can create new courses
- Everyone can view the people that are currently enrolled in the course

Assignments/Quizzes

- Instructors can create assignments and quizzes
- Students can take quizzes
- Quizzes can consist of question that are:
 - Multiple choice
 - Short answer
 - Long answer
- Assignments can be automatically or manually graded

Chat/Discussions

- Students can direct message the professor or other students
- Students can discuss class topics or ask questions on a discussion board (similar to piazza)

Files

- Instructors can upload course related files that students will be able to read
 - PDF file viewing
 - MP4 videos
 - Embedded YouTube videos

Administration

- Professors can customize their course page
- All users need to update information about themselves
 - Passwords
 - Name
 - Profile pictures

Optional (May not get to this)

- Implement a flashcard feature similar to Quizlet
 - Create flashcards
 - Modify flashcards

Non-functional Requirements:

OpenEdu should be:

- Make the UI visually appealing and more clean.
 - i.e Avoid the need to include unnecessary clutter
- Finished by the deadline of Apr 12
- Secure (courses should be only accessible by those accepted by the given professor)
- All website related data is stored in a MySQL database.
- Client written in HTML, CSS, and JavaScript
- Server written with JSP
- Variables must use camelCase.
- Proper git usage.

User Stories:

Users:

- As a user, I want to be able to login/sign up, so that I can view my account.
- As a user, I want to be able to update my details, so that information about me is up to date.

Admin:

- As an admin, I want to be able to create courses, so that students can see available courses.
- As an admin, I want to be able to accept students for courses, so that they can view the course content.

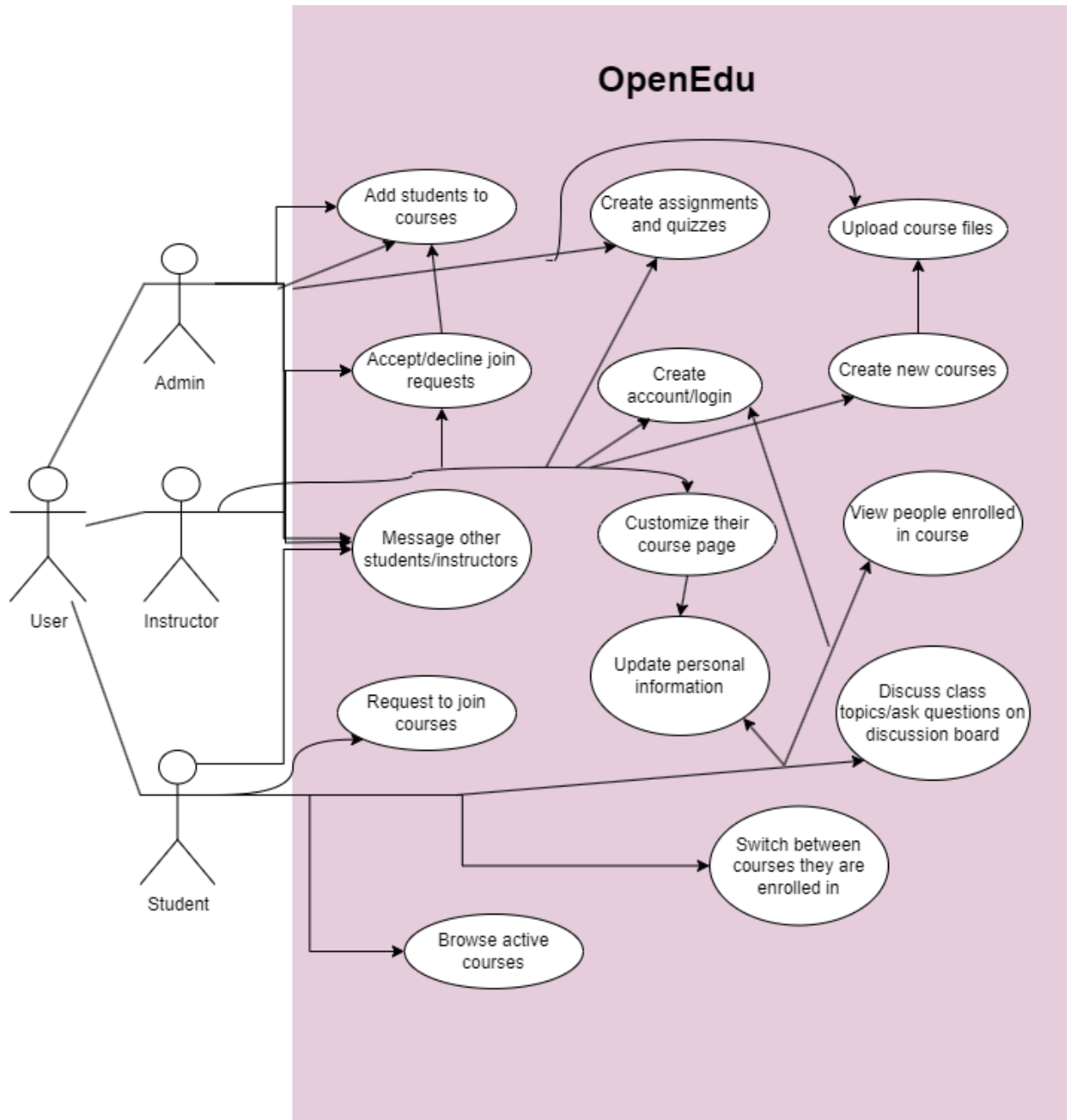
Teacher:

- As a teacher, I want to be able to create different assignments (quizzes and essays), so that I can assess student learning.
- As a teacher, I want to be able to grade assignments, so that students can access their grades and feedback.

Student:

- As a student, I want to be able to view currently available courses, so that I can register for courses I'm interested in.
- As a student, I want to be able to submit assignments I have completed, so that my teacher can grade them.
- As a student, I want to be able to see my grades, so that I can know how I'm doing in my classes.

Use Case Diagram (Updated):



Use Case Diagram (Outdated):

