

Axis Requirements

Team members

Michael Blythe
Tyler Gilbert
JaNiah Harris
Mason Krause

Purpose

The purpose of Axis is to provide a platform for professors to manage standards based grading for multiple courses, eliminating inefficient and redundant communication between teacher and student, and reducing student confusion while enrolled in a course with standards based grading.

Requirements

1. Login

Name: login_display

Type: Functional

Summary: The login screen should have two text fields. One text field for the username and another for the password.

Name: login_button

Type: Functional

Summary: It should have a "login" button that takes the user to their dashboard if they enter a valid username and password. If the user enters an invalid username or password it should display an error message. If the user presses the login button and one of the text fields is empty, then it should display an error message.

Name: forgot_password_button

Type: Functional

Summary: Should take the user to the forgot password screen

Name: new_user_signup_button

Type: Functional

Summary: It should take the user to the registration screen

2. Registration

Name: registration_display

Type: Functional

Summary: The registration screen should be displayed when the user presses the “sign up” link on the login page. It should display text fields for first name, last name, email, password, confirm password, student organization, and a dropdown for school type.

Name: signup_button

Type: Functional

Summary: It should display a sign up button that sends the user to their dashboard. If the user presses the “sign up” button and one or more of the text fields is empty, then it should display an error message.

Name: username_error

Type: Functional

Summary: If the username entered in the username text field is not available it should display a “username not available” error message.

Name: password_error

Type: Functional

Summary: If the text entered in the password and confirm password text fields do not match, then it should display an error message that says, “the passwords do not match”.

Name: student_organization_textfield

Type: Functional

Summary: As the user enters their school’s name, it should display a dropdown of schools with that name

Name: school_type_dropdown

Type: Functional

Summary: displays a dropdown with the options: middle school, high school, college.

3. Teacher Dashboard

Name: section_management

Type: Functional

Summary: displays a list of selectable, in-progress sections. When selected pulls up the section_view. Teachers can also create new sections, or edit section points.

Name: section_view

Type: Functional

Summary: displays the course code and section code, course name, and a selectable list of students enrolled in that section. A student can be selected to pull up the student_view. A teacher can also send an enrollment invite to a student using the student's email address.

Name: student_view

Type: Functional

Summary: displays the students name, and a selectable list of checkpoints/testpoints for that section. The teacher can select a point to update the status. There should also be a button to unenroll the student from the section.

Name: course_management

Type: Functional

Summary: displays a list of the teachers' courses. A teacher can select a course to pull up the course_view. Teachers can also create new courses.

Name: course_view

Type: Functional

Summary: displays the course code, course name, and a list of the topics that make up that course. A teacher can select a topic to pull up the topic_view.

Name: topic_view

Type: Functional

Summary: displays the topic name and a list of the concepts that make up that topic.

Name: semester_management

Type: Functional

Summary: displays a list of previous semesters that the teacher has used the app. A teacher can select a semester to pull up the semester_view. Teachers can also create new semesters.

Name: semester_view

Type: Functional

Summary: displays the section_management for the sections that occurred during the selected semester.

4. Student Dashboard

Name: student_dashboard_display

Type: Functional

Summary: The student dashboard should display a list of courses the student is currently enrolled in. Each course should include the course name, course code, and instructor(s).

Name: add_course

Type: Functional

Summary: Students should be able to manually add a course to their dashboard for tracking grades, assignments, and progress.

Name: drop_course

Type: Functional

Summary: Students should be able to remove a course from their dashboard if they no longer wish to track its grades or assignments.

Name: view_assignments

Type: Functional

Summary: The student should be able to view all assignments including past attempts and future assignments due.

Name: view_grades

Type: Functional

Summary: The student should be able to view grades for past completed assignments as well as an estimated grade if no more assignments are completed at the current point.

Name: load_times

Type: Nonfunctional

Summary: The Display and relevant components should display to user seamlessly by being under 2 seconds.

Name: data_encryption

Type: Nonfunctional

Summary: All student-related data, including personal information, grades, and messages, must be encrypted both in transit and at rest using Firebase Security Rules and Firebase Authentication to ensure data privacy and protection.

Name: auth_timeout

Type: Nonfunctional

Summary: If a student closes the app or is inactive for a period of time(10 mins) then the system should automatically log them out.