

ENSF-381: Assignment 3

Winter 2025

Department of Electrical and Software Engineering Schulich

School of Engineering

Due: March 14, 2025, at 11:59 PM

Enhance the Learning Management System (LMS) built in Assignments 1 and 2 by adding interactivity and dynamic features using JavaScript. Focus on DOM manipulation, form validations, API integration and real-time updates to create a responsive user experience.

Administrative Details

- The submission should be a **single compressed file (.zip)** that includes the complete source code and resources (e.g., images), ready for execution without modifications.
- Name the submitted file as: `Assignment3_ENSF381_Section#_Group#`
Example: `Assignment3_ENSF381_L02_Group01.zip`
- Submit ONLY ONE copy.
- Include group members' names and UCIDs at the top of the `index.html` file as comments.

Specifications:

Implement the following features across the LMS pages:

Instructions:

- If you did not complete Assignment 02, a reference solution will be available on March 04, 2025, in D2L to help you proceed with this assignment.
 - Concepts which will be needed but are not discussed in class: `window.location.href`, `input.checked`, `setInterval`
-

1. Course View Page (`course_view.html`) - 20 Marks

Requirements:

a) Dynamic Enrollment:

- Clicking "Accept" or "Decline" on pending courses updates the UI without reloading the page as follows: accepted courses move to the "Enrolled" section; declined courses are removed.

b) Hover Effects:

- Highlight course tiles on hover (e.g., change background color to `#ffffff`).
-

2. Attempt Quiz Page (attempt_quiz.html) - 30 Marks

Requirements:

a) Quiz Timer:

- Add an html element to display a 10-minute countdown timer in MM:SS format. [Use `setInterval` and `element.textContent` to implement the timer.]
- Automatically redirect to `leaderboard.html` when time expires. [Use `window.location.href` for automatically redirecting to other pages.]

b) Answer Validation:

- Validate radio/checkbox selections and show alerts for incomplete answers. [Use `input.checked`]

c) Score Calculation:

- Track correct answers and display a final score on the page.
 - Assume Q1's correct answer is "Hyper Text Markup Language". Correct answer gives +50 marks. Incorrect answer results in 0 marks. [Use `input.checked`]
 - Assume Q2's correct are both checkboxes. For each selected checkbox, the student gets +25 marks. [Use `input.checked`]
 - The final score displayed is out of 100.
-


3. Login Page (login.html) - 20 Marks

Requirements:

a) API Validation:

- Fetch user data from <https://jsonplaceholder.typicode.com/users>
- Validate credentials against the API's username and email fields.
 - Username entered must match with any one of the usernames returned by the API.
 - Password entered must match with the corresponding users email address as returned by the API.
- Display success/error messages in a styled dynamic box.
- In case of success, display the message for 2 seconds and then automatically load `course_view.html`. [Use `window.location.href` for automatically redirecting to other pages.]

- b) Update the Login page to include a link for loading a Sign Up page for new users.



LMS - Learning Management System

Homepage

LMS Login


Username:

Password:

Login

[Forgot Password?](#)
[Don't have an account? Sign Up](#)

Login successful! Redirecting...



LMS - Learning Management System

Homepage

LMS Login

Username:

Password:

Login

[Forgot Password?](#)
[Don't have an account? Sign Up](#)

Invalid username or password!

file:///Users/Novarun/Dropbox/ENSF-381/Assignments/Assignment 03/Assignment 03 Solution/course_view.html

4. Signup Page (signup.html) - 30 Marks

Requirements:

a) Create *signup.html*:

- Include a form with fields for username, password, confirm password, and email.
- Add a "Signup" button.
- Style the "Signup" button with a contrasting color of #4CAF50 (green), padding of 10px, border radius of 5px, margin of 10px, and opacity of 0.5.
- Add a hover effect by changing the background color to #45A049 (a slightly darker shade of green) and opacity of 1.0.
- The page should be in the same theme and layout as the Login page that was developed as part of Assignment 02.

b) *Form Validation*:

i) Username:

- Must be between 3 and 20 characters long.
- Allowed characters: alphanumeric characters (letters A-Z, numbers 0-9), hyphens (-), and underscores (_).
- Must start with a letter.
- Cannot contain spaces or special characters other than hyphens and underscores.

ii) Password:

- Must be at least 8 characters long.
- Must contain at least one uppercase letter, one lowercase letter, one number, and one special character.
- Allowed special characters: !@#\$%^&*()-_+=[]{}|;:",".<>?/^~.
- Cannot contain spaces.

iii) Confirm Password:

- Must match the password entered in the password field.

iv) Email:

- Must be a valid email address format (e.g., username@example.com).
- Cannot contain spaces.
- Must contain an "@" symbol followed by a domain name (.com, .net, .io).

c) When the user clicks on the "Sign up" button, perform the validation and:

- Display error messages* in a dynamic box for invalid inputs.
- On successful signup*, redirect the user to the Login Page.



Sign Up

Username:

234jhsdkjdhf

Password:

Confirm Password:

Email:

qosaoeir@sidhf

Sign Up

Invalid username (Reason:...)

Invalid password (Reason:...)

Passwords do not match

Invalid email (Reason:...)



Sign Up

Username:

john doe12

Password:

Confirm Password:

Email:

jdoe12@gmail.com

Sign Up

Signup successful! Redirecting to login...

[Already have an account? Login here](#)