COMSATS University Islamabad (CUI), Lahore Campus

Department of Electrical & Computer Engineering

CSC336 – Web Technologies

Lab Manual v 1.0.1

Lab Resource Person

Muhammad Babar Ali

Theory Resource Person

Muhammad Babar Ali

FALL 2024

Name:		**Registration No:**	
Program:		**Batch:**	
Semester:			

Laboratory 05

To develop web application using vanilla JavaScript and Web APIs

APPLICATION DEVELOPMENT, VANILLA JAVASCRIPT AND WEB APIS

1 Objective

The purpose of the lab is to introduce students to the Web APIs. They will understand the importance of Web APIs which are, by default, exposed by user agents (UA). These pre-built Web APIs in the UA are known as Browser APIs. Students will learn how to use JavaScript to interact with a certain Browser APIs so that its various attributes and features (i.e. properties and methods) can be utilized to enhance functionality of the application and to improve user interface/user experience (UI/UX). Finaly student will use one of the third Paty APIs available on the internet.

2 Introduction to Web APIs

The API stands for the Application Programming Interface. It is a standard protocol or set of rules to communicate between two software components or systems while a Web API is an API for web. The basic purpose of the web API is to:

- extend the functionality of the browser
- simplify complex functions
- provide easy syntax to complex code

2.1 Categorization of Web APIs

Based on the side (fontend or backend) the Web APIs is running on, it can be classified into the following categories:

- Client-side Web APIs
- Server-side Web APIs

The client-side APIs resides in the frontend code of the web application, e.g. Browser APIs. The server-side APIs resides in the backend code of the web application. These server-side APIs can be acceded through an endpoint (AKA Universal Resource Indicator, URI) exposed by the server-side application e.g. https://your-web-app.abc/api/products.

Server APIs which are not part of the backend of the certain application and are available from other servers/websites are called Third Party APIs e.g. Google APIs.

3 The Browser APIs

Some of the useful Browser APIs ↗ are listed below:

- ✓ CSS Object Model (CSSOM) ↗
- ✓ Document Object Model (DOM) ↗
  - HTML DOM API ↗
- ✓ HTML Drag and Drop API ↗
- ✓ Fetch API ↗
- ✓ Geolocation API ↗
- ✓ History API ↗
- ✓ Fullscreen API ↗
- ✓ Keyboard API ↗

4 Lab Activity

4.1 Task 1

Create the login page as shown in Figure 4.1. You will understand HTML DOM API ↗.

Consider the functional requirements mentioned in the next subsection.

4.1.1 Functional Requirements

The web page must adhere to the following functional requirements:

- Validation requirements for username and password
  - The minimum length of 6 characters in username
  - The maximum length of 10 characters in username
  - Considering hyphen '-' as an invalid character in username.
  - Considering hyphen '-' as an invalid character in username.
  - The minimum length of 6 characters in password
  - Containing at least one character symbol, one number, and one character alphabet.
- Add event listers to "Register" button or some otherway to get data from the login form. Add logic to validate it.
- Add event listers to "Change Password" button to set new password. This must now only show password field (hide user name input filed) and "Change" button (instead of "Register" button)
- All interactions performed within JavaScript source file(s) and by using DOM manipulations.

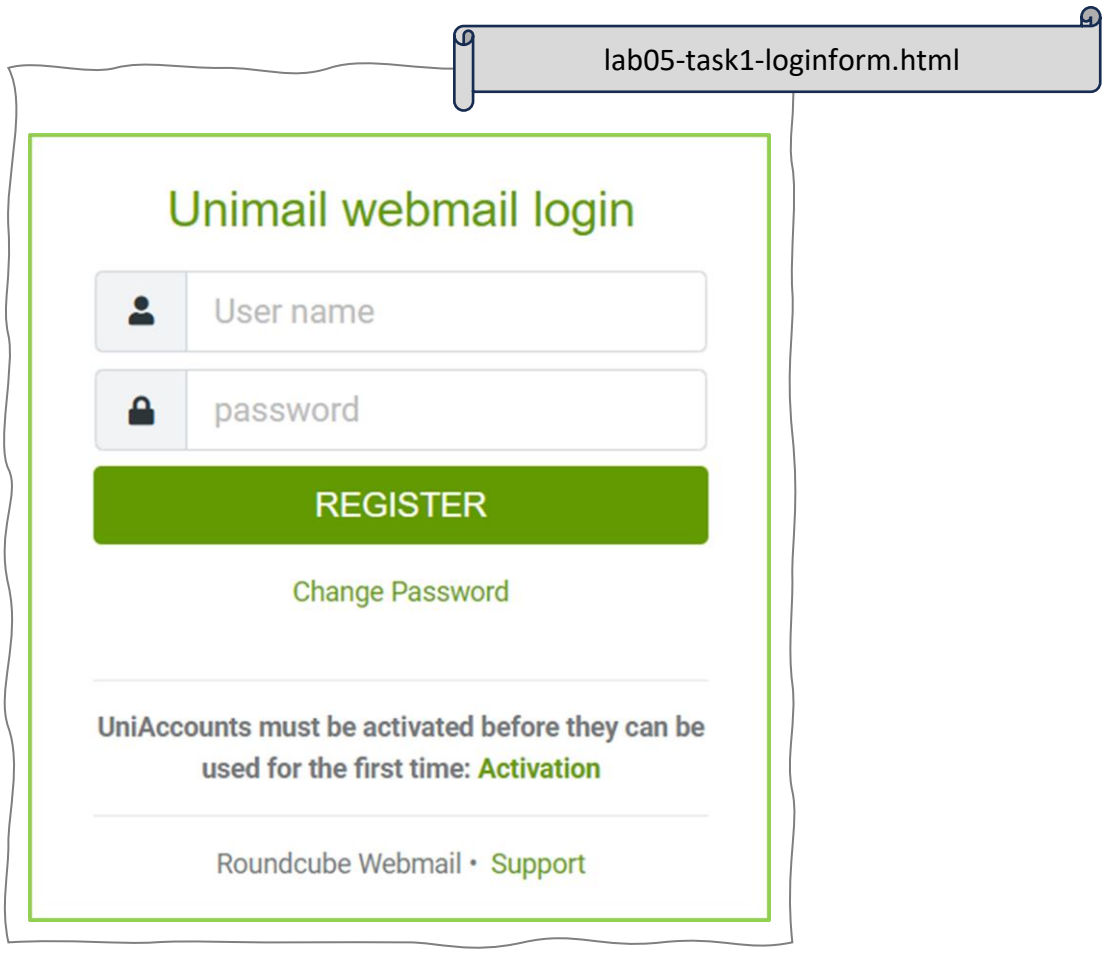

Figure 4.1: Login Form

5 Home Activity

5.1 Work 1

You are interested in web application, namely “Todo Application” to manage your daily activities. The desired user interface (UI) of the page is given in Figure 5.1.

You will understand HTML DOM API ↗ and HTML Drag and Drop API ↗

To achieve this, however, you must design an app meeting the functional and user interface/user experience (UI/UX) requirements mentioned in the next subsection.

5.1.1 Functional Requirements

The web application must adhere to the following functional requirements:

- All tasks must be stored in an array constant named “TASK_STORE”.
- Design an object structure to store each task so that tracking the state of the task is easy.
- The drag and drop must update the dragged tasks accordingly in “TASK_STORE”.
- The three lists display tasks, in accordance with “TASK_STORE”, as described:
    - The “Pending Tasks” list must show all tasks which are neither done/complete nor deleted.
    - The “Completed Tasks” list must show all tasks which are done/complete.
    - The “Purged Tasks” list must show all tasks which are deleted.
- The four buttons control the status of a task as described and must update the respective task accordingly in “TASK_STORE”:
    - Press “progressing” button if the task has been started and is currently in progress.
    - Press “done” button if the task is finish/done/complete.
    - Press “restore” button if the purged task must be restored to the last list where the task was, before deletion.
- The “Purge Tasks” button deletes all tasks which are currently in progress and must update the respective task accordingly in “TASK_STORE”.
- Do not use inline event attributes to register events or to style an element. All interactions performed within JavaScript source file(s) and by using DOM manipulations.

5.1.2 UI/UX Requirements

The web application must adhere to the following UI/UX requirements:

- On each click on the “progressing” button of a certain task in the “Pending Tasks” list, the font color of the text must toggle between “black” and “green”.
- The display of the list (contents) under any of the “Pending Tasks”, “Completed Tasks” or “Purged Tasks” lists can be toggled (hidden/shown) by clicking respective heading.
- Add drag and drop feature using `HTML Drag and Drop API` ↗
    - User can drag an item to shift it and drop it to any other lists.

lab05-work1-todoapp.html

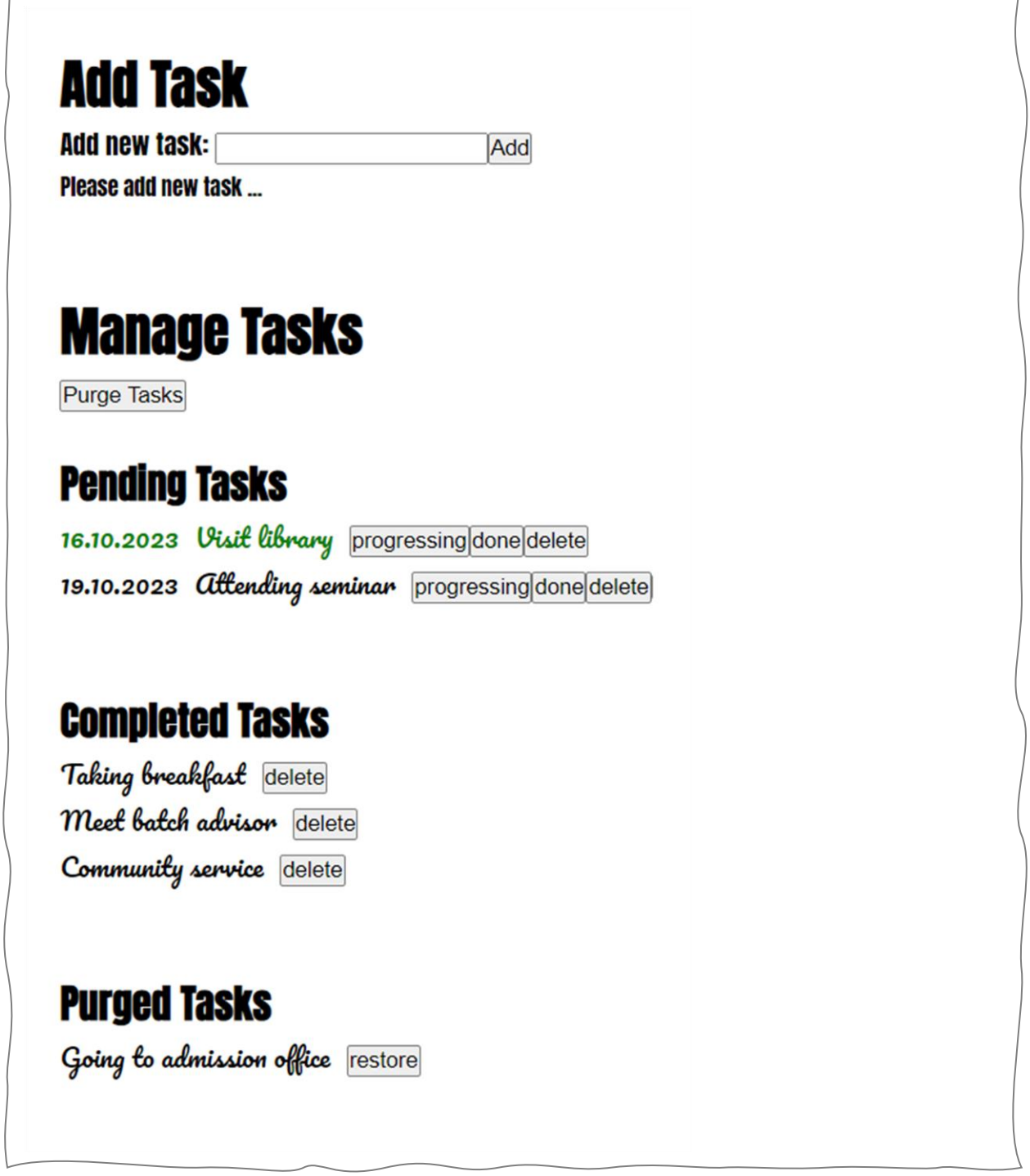

Figure 5.1: Todo application

6 Rubric for Lab Assessment

The student performance for the assigned task during the lab session was:			
Excellent	The student completed assigned tasks without any help from the instructor and showed the results appropriately.	4	
Good	The student completed assigned tasks with minimal help from the instructor and showed the results appropriately.	3	
Average	The student could not complete all assigned tasks and showed partial results.	2	
Worst	The student did not complete assigned tasks.	1	

Instructor Signature:_____ **Date:** _____