University of Jeddah
College of Computer Science and Engineering
Department of Software Engineering

CCSW321 – WEB DEVELOPMENT

Assignment #2: Web Services, APIs, AJAX, and DOM

Assignment Marks: 5 Points

Submission Deadline: November 26, 2023

Assignment

Choose an external API and build a web page that communicates with the API using AJAX requests and displays the data returned in the webpage. The outcome should demonstrate your ability to retrieve data from the API, handle responses asynchronously, and dynamically update the DOM to display the retrieved data.

Objectives

The purpose of this assignment is to learn about web services, APIs, AJAX (Asynchronous JavaScript and XML), and DOM (Document Object Model) manipulation. Through this assignment, students will gain hands-on experience in working with external APIs, making AJAX requests, and dynamically manipulating the DOM to create interactive web applications.

Instructions/Requirements

Task Requirements:

- Select a **unique and distinct API** of your choice, ensuring that your API selection and implementation differ from your classmates. **Put effort** into choosing an API that aligns with your interests and **consider creative ways** to **present** the retrieved data on the web page.
- You can create a single-page website or a multi-page website.
- You must use **at least three different event types**. Follow this link for a full list of possible events: https://www.w3schools.com/jsref/dom_obj_event.asp
  - E.g., single click, double click, keyboard keyup.
- The web application should successfully integrate with the chosen external API and retrieve data using **AJAX** to handle all communication with the API.
- The **DOM** should be **dynamically updated** to display the retrieved data in an organized and user-friendly manner (i.e**., no page refresh** is required).

- Your code must include at least **a single dynamic creation** and **removal** of DOM elements. For example, create a button to clear all API data, and another button to fetch the latest API data, or a button to pull the next random API data.
- Your code must contain **form elements** that **allow** the **user** to **interact** with the API. For example, if you're using a weather API, ask the user to specify the city.
- Your code must **display at least 5 useful API data information** on the page. For example, if you're using the weather API, then from the API data, pull the 1)city name, 2)wind speed, 3)temperature, 4)feels like, 5)weather description.
- The web application should demonstrate proper error handling, providing informative messages to the user when necessary.
- The user interface should be visually appealing and responsive across different devices.
- This an individual assignment. No collaboration between students is allowed.
- The code should be well-documented with clear comments explaining the functionality and logic of each section.

Where to find APIs?

To find a suitable API for your assignment, you can search for "free public API" on search engines like Google. Additionally, you can refer to this GitHub repository that contains a wide range of free APIs suitable for software and web development.

https://github.com/public-apis/public-apis

Since the list contains numerous APIs, **you should ensure that your API selection is distinct from your classmates**. Even if you and another student selected the same API, you must present the information and data retrieved from the API differently. **Any submissions that closely match each other will be significantly penalized!**

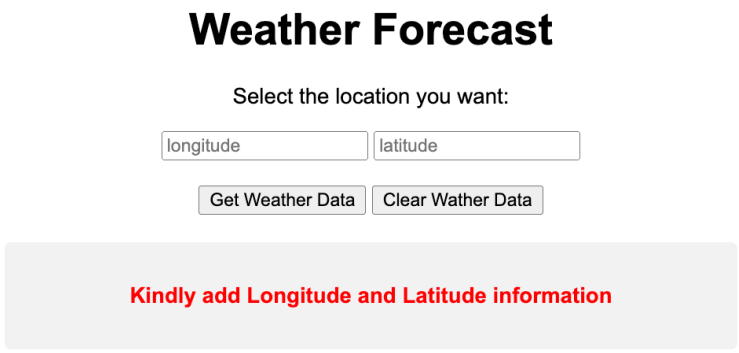
Submission Instructions:

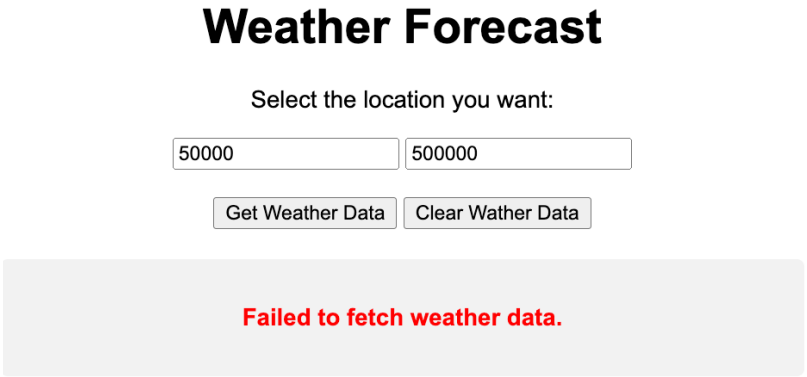
You should submit the following:

- A zipped file with a fully functional version of your assignment code.
  - The zipped file name must contain your name and ID.
- A PDF file with images of all page(s), code, available interactivity (e.g., show what happens when a button is clicked), and any error handling that was done (e.g., show the error messages). It should clearly show the implemented JavaScript behavior.

Example Submission

The following is an example for a submission that pulls the weather data based on the longitude and latitude specified.





Assessment Sheet:

Grades will be determined according to the following:

Assessment (5 Points)	Grade
CLO 2.1 (3.5 points)	
Structure: HTML, CSS, JS, and media are separated and organized as specified with clear separation between front-end and back-end code.	0.5
Content: The used API and the presentation of the page is unique and different from other students in classroom	1
API: The user can modify the parameters of the API through a provided form	0.5
API: The communication with the API is done through AJAX.	0.5
DOM: At least a single element creation and removal exist and the page content is dynamically updated (no page refresh needed)	1
CLO 3.2 (1.5 points)	
Interactivity: The JS code offers at least three different event types	0.5
Interactivity: The user is shown appropriate messages for both valid and invalid submissions.	0.5
API: The page displays 5 API elements	0.5
Common Issues	
Student choice of API and design of page is closely matching a classmate (minimum effort shown)	-2
Student did **NOT** follow submission instructions (PDF + zip)	-2
The submitted code does **NOT** seem to be the student's own work. (e.g., seems to be an online premade template, or seems to be generated from an online tool)	-2.5