

# MILESTONE EXAM - 2



## Project 1: Building PWSkills Blog using HTML, CSS, and JavaScript

**Objective:** In this assignment, you will create a blog application called "PWSkills Blog" using HTML, CSS, and JavaScript. The application will allow users to add their own blogs and read any existing blog posts. To achieve this, all blogs' data will be stored in the browser's LocalStorage.

### Requirements:

- The blog application should have two main pages: the home page and the blog page.
- On the home page, users should be able to see a list of all existing blog posts with their titles and brief summaries.
- Users should be able to click on a blog post to view its full content on a separate page.
- On header There should be "+" icon to add new blog by clicking on that a modal should open
- The add blog modal should have a form with fields to input the blog title, blog poster, description and blog content.
- Upon submitting the add blog form, the new blog post should be added to the list of existing blogs on the home page, and the data should be stored in the LocalStorage.

Ensure that the application has a clean and visually appealing design using CSS.

**Note:** it's not mandatory to make it responsive but in case you want to do so Feel Free to do so.

### 1. Homepage

#### PWSkills Blog

+



Blog Title: Demo Article

Blog Description: lorem ipsum  
lorem ipsum sum ip

[Read](#)



PwSkills - 4 Year Residential Program

Building Bridges to Success with Our Accomplished Mentors

[Read](#)

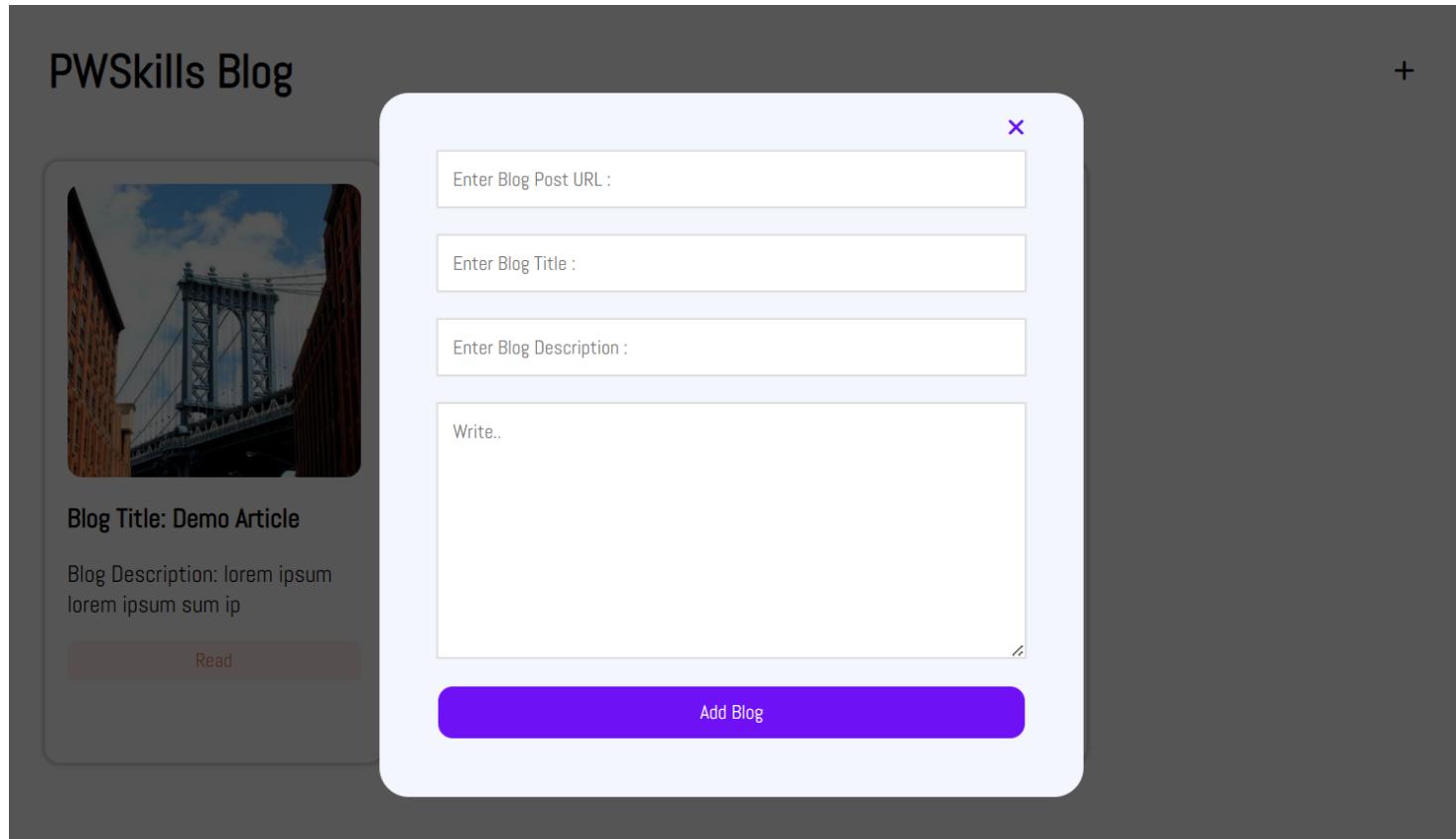


How did I build Sharestory Platform Within 7 days

in this blog, I'll be sharing how did I actually came up with idea

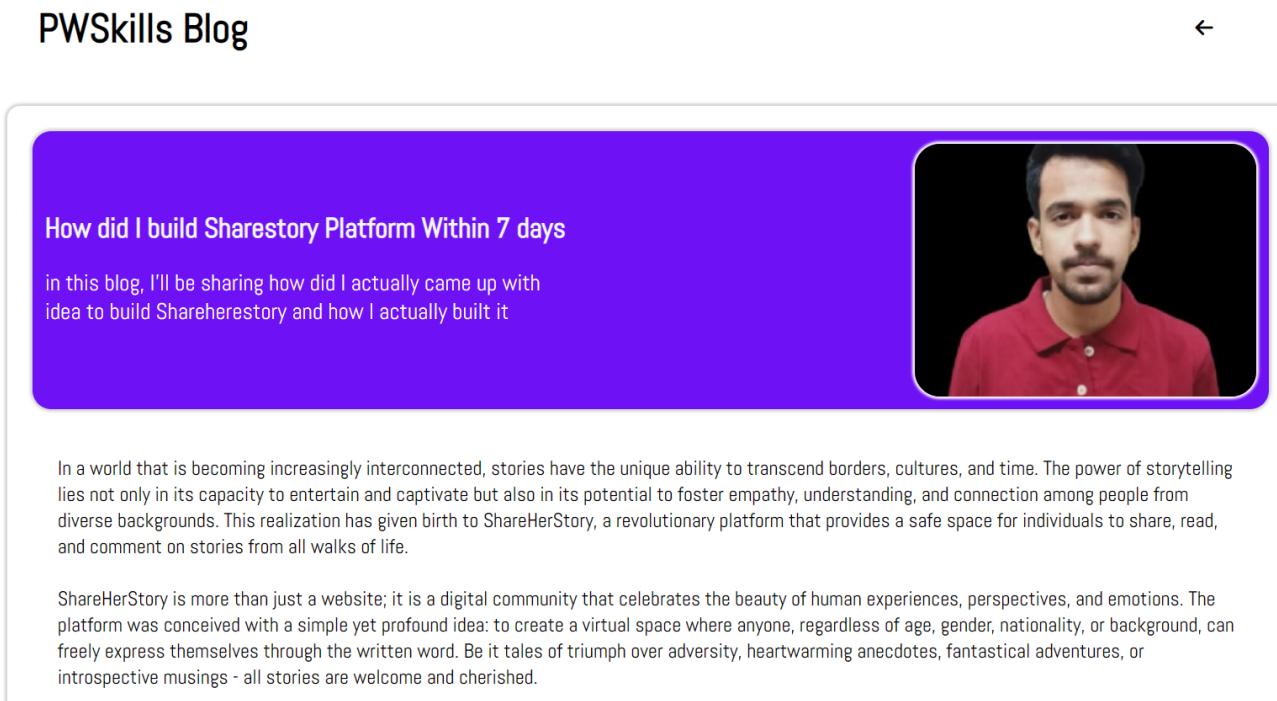
[Read](#)

## 2. Add Blog Modal



The screenshot shows a modal window titled "PWSkills Blog" with a background image of a bridge. The modal contains fields for "Enter Blog Post URL:", "Enter Blog Title:", "Enter Blog Description:", and a text area labeled "Write..". A purple "Add Blog" button is at the bottom. The modal has a close button "X" in the top right corner.

## 3. Read Blog Page



The screenshot shows a blog post titled "How did I build Sharestory Platform Within 7 days". The post starts with the text: "in this blog, I'll be sharing how did I actually came up with idea to build Shareherestory and how I actually built it". To the right is a portrait of a man with a beard. Below the post, there is a paragraph about the power of storytelling and a larger text block about the ShareHerStory platform.

How did I build Sharestory Platform Within 7 days

in this blog, I'll be sharing how did I actually came up with idea to build Shareherestory and how I actually built it

In a world that is becoming increasingly interconnected, stories have the unique ability to transcend borders, cultures, and time. The power of storytelling lies not only in its capacity to entertain and captivate but also in its potential to foster empathy, understanding, and connection among people from diverse backgrounds. This realization has given birth to ShareHerStory, a revolutionary platform that provides a safe space for individuals to share, read, and comment on stories from all walks of life.

ShareHerStory is more than just a website; it is a digital community that celebrates the beauty of human experiences, perspectives, and emotions. The platform was conceived with a simple yet profound idea: to create a virtual space where anyone, regardless of age, gender, nationality, or background, can freely express themselves through the written word. Be it tales of triumph over adversity, heartwarming anecdotes, fantastical adventures, or introspective musings - all stories are welcome and cherished.

## Project 2: Build Simple Movie Search Web App Using HTML, CSS & JavaScript Only

### Objective:

Your assignment is to create a movie search application that allows users to search for any movie using an input box. Upon receiving the user's input, the application should fetch movie data from the OMDB API for the specific query and display the relevant movie details.

### Requirements:

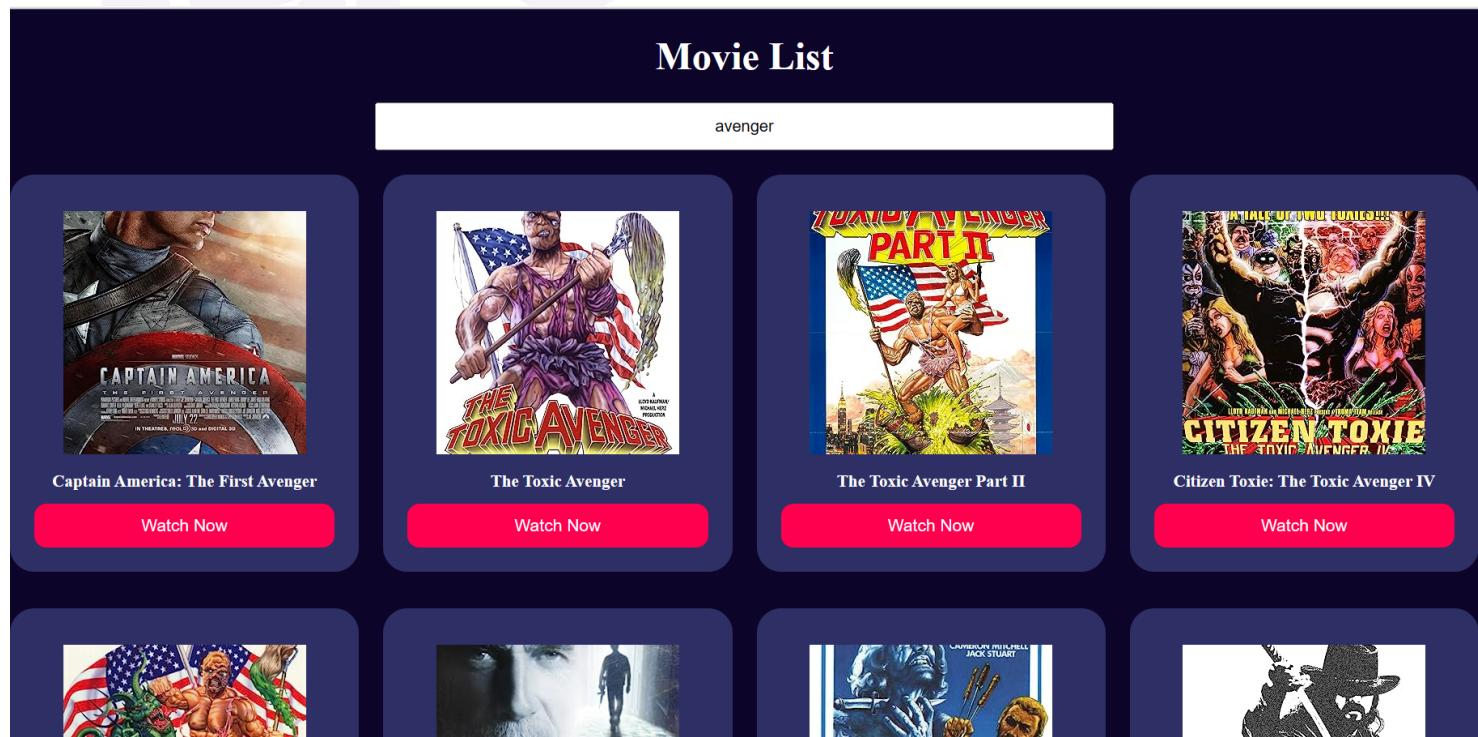
- The application should have a simple and user-friendly interface built using HTML and CSS.
- You are required to use JavaScript to handle user interactions and fetch data from the OMDB API.
- Display the fetched movie details such as movie title, poster image on the application interface.
- Handle edge cases gracefully, such as when a movie is not found or the user provides invalid input.

### Instructions:

- Register on the OMDB API website to obtain your API key. The API key will be required to make requests to the OMDB API.
- Set up a new HTML file and name it "index.html".
- Create a CSS file named "style.css" to style the application.
- Create a JavaScript file named "script.js" to implement the functionality.
- Use the fetch API in JavaScript to request movie data from the OMDB API using the user's input as the search query.
- Display the fetched movie details in a structured and visually appealing manner on the HTML page.

### OMDB (API) Document:

- Ad in this project You'd have to use an api so make sure to get your doubts clear by going through the documentation of the api itself
- Link : <https://www.omdbapi.com/>



A screenshot of a movie search application titled "Movie List". The search bar at the top contains the text "avenger". Below the search bar, there are four movie cards displayed in a row. Each card includes a movie poster, the title, and a "Watch Now" button. There are also two additional movie cards partially visible below the first row.

Movie Title	Poster Image	Action
Captain America: The First Avenger		Watch Now
The Toxic Avenger		Watch Now
The Toxic Avenger Part II		Watch Now
Citizen Toxie: The Toxic Avenger IV		Watch Now
(Partially Visible)		

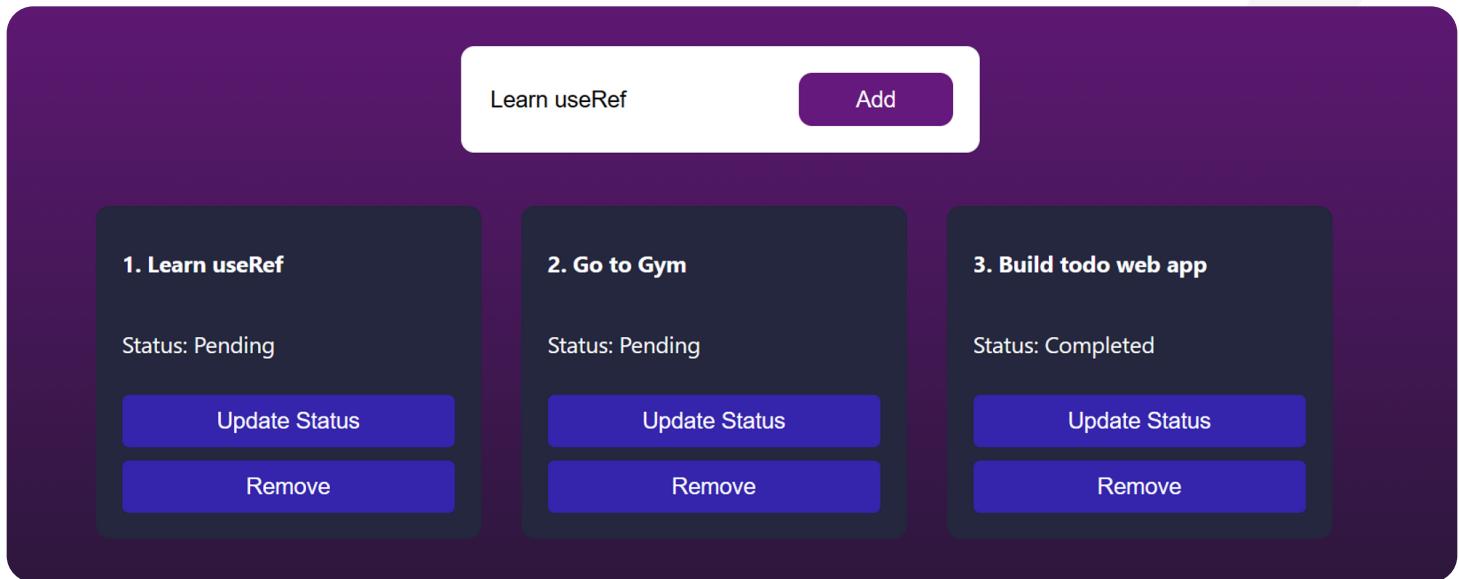
## Project 3: Create a Simple Todo Web App with following features Using React

### Requirements:

- The blog application should have one main pages: the home page On the home page, users should be able to see a list of all existing todo with their titles and status
- An input button where users can type their tasks. On clicking the submit button, the entered task should be displayed in the UI.
- Users should be able to click on a update todo status to update status of todo pending to completed & completed to pending
- In todo card in footer there should be button to remove todo on clicking that todo should be removed from ui
- Before title there should be todo number as shown in the image

Ensure that the application has a clean and visually appealing design

**Note:** it's not mandatory to make it responsive but in case you want to do so Feel Free to do so.



## How to Submit Assignment

Dear Students, Congratulations on embarking on this exciting coding project! In order to effectively organize and showcase your work,

We kindly request you to **create a repository** that aligns with the following structure. **Each project should have its dedicated folder within the repository**, encapsulating all relevant code assets and resources.

To ensure accessibility and ease of navigation, please remember to make your repository public.

Moreover, we encourage you to demonstrate your commitment to clarity and transparency by crafting a descriptive and informative README file for each project.

This **README** file should serve as a comprehensive guide to understanding the **functionalities of your project**. By clearly outlining the purpose, features, and potential benefits of your work, you will be providing valuable context to any interested parties who explore your repository.

**MileStone Repo**

**Project 1**

**Project 2**

**Project 3**