Jaypee University of Information Technology

Department of Computer Science and Engineering

Project File

Title – PicStream (Image API Fetcher using ReactJs and Tailwind CSS**)**

Course Code: 18B1WCI575
Course Name: Multimedia Lab

Submitted by (Batch – CS-57):

Name: Aniket Rawat (211315)

Name: Himanshu Sharma (211321)

Submitted to:

Name: Dr. Seema Rani

Table of Contents

Introduction

The Pixabay API-based website is a platform that allows users to search for and access high-quality royalty-free images from Pixabay. The website is designed to be user-friendly and intuitive, with a simple and clean layout that makes it easy for users to search and discover images. The website leverages the power of the Pixabay API, which provides access to over a million images and.

The website is designed to meet the needs of individuals, businesses, and organizations that require high-quality images for personal or commercial use. The website's main feature is the image search function, which allows users to search for images based on keywords. The website uses the Pixabay API to retrieve a list of relevant images, which are displayed in a grid format on the search results page.

The website is fully responsive, meaning it can be accessed from any device, including desktops, laptops, tablets, and smartphones. The design is optimized for fast loading speeds and smooth navigation, ensuring a seamless user experience.

Overall, the Pixabay API-based website is a valuable resource for anyone looking for high-quality, royalty-free images. By leveraging the power of the Pixabay API, the website provides a simple and efficient way for users to discover and access a vast library of stunning images.

Project Description

The Pixabay API-based website is an image search and display platform that allows users to search for and access high-quality royalty-free images from Pixabay. The website is built using the Pixabay API, which provides access to over a million images and videos.

The website is designed to be user-friendly and intuitive, with a simple and clean layout that makes it easy for users to search and discover images. The homepage features a search bar where users can enter their desired search term, as well as a collection of curated images to inspire and guide their search.

When a user enters a search term, the website uses the Pixabay API to retrieve a list of relevant images, which are displayed in a grid format on the search results page.

The website is fully responsive, meaning it can be accessed from any device, including desktops, laptops, tablets, and smartphones. The design is optimized for fast loading speeds and smooth navigation, ensuring a seamless user experience.

Overall, the Pixabay API-based website is a valuable resource for anyone looking for high-quality, royalty-free images for personal or commercial use. By leveraging the power of the Pixabay API, the website provides a simple and efficient way for users to discover and access a vast library of stunning images.

Technology Stack

In this project we will be using ReactJs for the website development purpose and also we will be using tailwind css rather than normal css to make our website more accurate to the designing purpose and we are fetching images from a well known library which is pixabay api library.

Intro to Pixabay API

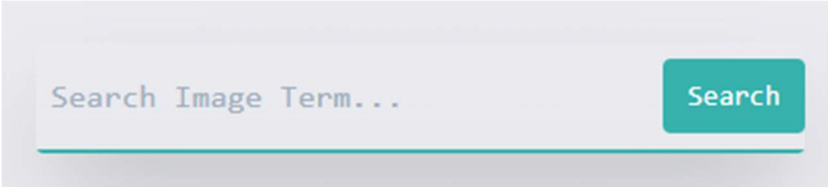
Pixabay is a popular platform for sharing high-quality images, illustrations, and vector graphics. Pixabay offers a free and open API (Application Programming Interface) that developers can use to access and integrate the site's vast library of images into their own projects.

Here are some details about the Pixabay API:

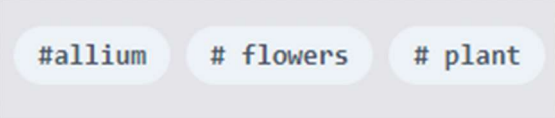
1. Access: The Pixabay API is open and free for anyone to use, with no registration required. You can access the API via HTTP GET requests to the Pixabay server.
2. Authentication: The Pixabay API does not require authentication, but it does require an API key to be included in all requests. You can get an API key by registering on the Pixabay website.
3. Endpoint: The Pixabay API endpoint is https://pixabay.com/api/. All API requests are sent to this endpoint.
4. Search: You can use the Pixabay API to search for images by keyword, category, or even color. The API supports advanced search parameters like image type (photo, illustration, vector), orientation (horizontal, vertical), and minimum/maximum image dimensions.
5. Response format: The Pixabay API returns JSON data in response to all requests. The response includes metadata about each image, such as its URL, dimensions, tags, and user information.
6. Usage limits: The Pixabay API is free to use, but there are usage limits to prevent abuse. The current limit is 5,000 requests per hour, and you should also adhere to the Pixabay terms of service and usage guidelines.

Website Features

1. Image Search: The website allows users to search for images based on keywords. Users can enter any term they want, and the website will display a list of relevant images. The search feature is powered by the Pixabay API, which ensures that the results are accurate and up-to-date.



2. Image Filters: The website allows users to filter search results by image type and category. This makes it easier for users to find the exact type of image they are looking for.



3. Responsive Design: The website is fully responsive, meaning it can be accessed from any device, including desktops, laptops, tablets, and smartphones. The design is optimized for fast loading speeds and smooth navigation, ensuring a seamless user experience.
4. API Integration: The website is built using the Pixabay API, which provides access to over a million images and videos. This ensures that the website always has the most up-to-date content, and that users can find exactly what they are looking for.

Code:

```jsx
import React from 'react';
import ReactDOM from 'react-dom/client';
import './assets/main.css';
import App from './App';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

src > JS App.js > ...

```jsx
import React, {useState, useEffect} from "react";
import ImageCard from "./components/ImageCard";
import ImageSearch from "./components/ImageSearch";

function App() {
  const [images, setImages] = useState([]);
  const [isLoading, setIsLoading] = useState(true);
  const [te
    See Real World Examples From GitHub
    Follow link (ctrl + click)

  useEffect
    fetch(`https://pixabay.com/api/?key=${process.env.REACT_APP_PIXABAY_API_KEY}&q=${term}&image_type=photo&pretty=true`)
    .then(res => res.json())
    .then(data => {
      setImages(data.hits);
      setIsLoading(false);
    })
    .catch(err => console.log(err));
  }, [term]);


  return (

    <div className="container mx-auto">
      <h1 className = "text-5xl text-center mx-auto mt-12 font-mono font-bold text-purple-700">Image API Fetcher using React and Tailwind CSS</h1>
      <ImageSearch searchText = {(text) => setTerm(text) } />

      {!isLoading && images.length === 0 && <h1 className = "text-5xl text-center mx-auto mt-32">No Images Found</h1>}

      {isLoading ? <h1 className = "text-6xl text-center mx-auto mt-32">Loading...</h1> : <div className="grid grid-cols-3 gap-4">
          {images.map(image => (
            <ImageCard key = {image.id} image = {image}/>
          ))}
        </div>}
      </div>
    );
  }

  export default App;
```

src > components > JS ImageCard.js > ...

```jsx
import React from 'react';

const ImageCard = ( {image}) => {
    const tags = image.tags.split(',');
  return (
    <div className = "max-w-sm rounded overflow-hidden shadow-2xl bg-slate-600">
      <img src = {image.webformatURL} alt = "" className="w-full" />
      <div className="px-6 py-4">
        <div className="font-bold text-green-700 text-xl mb-2">
          Photo By {image.user}
        </div>
        <ul>
          <li>
            <strong>Views: </strong>
            {image.views}
          </li>
          <li>
            <strong>Downloads: </strong>
            {image.downloads}
          </li>
          <li>
            <strong>Likes: </strong>
            {image.likes}
          </li>
        </ul>
      </div>
      <div className="px-6 py-4">
        {tags.map((tag, index) => (
            <span key = {index} className="inline-block bg-gray-200 rounded-full px-3 py-1 text-sm font-semibold text-gray-700 mr-2">
            #{tag}
          </span>
        ))}
      </div>
    </div>
  )
}

export default ImageCard;
```

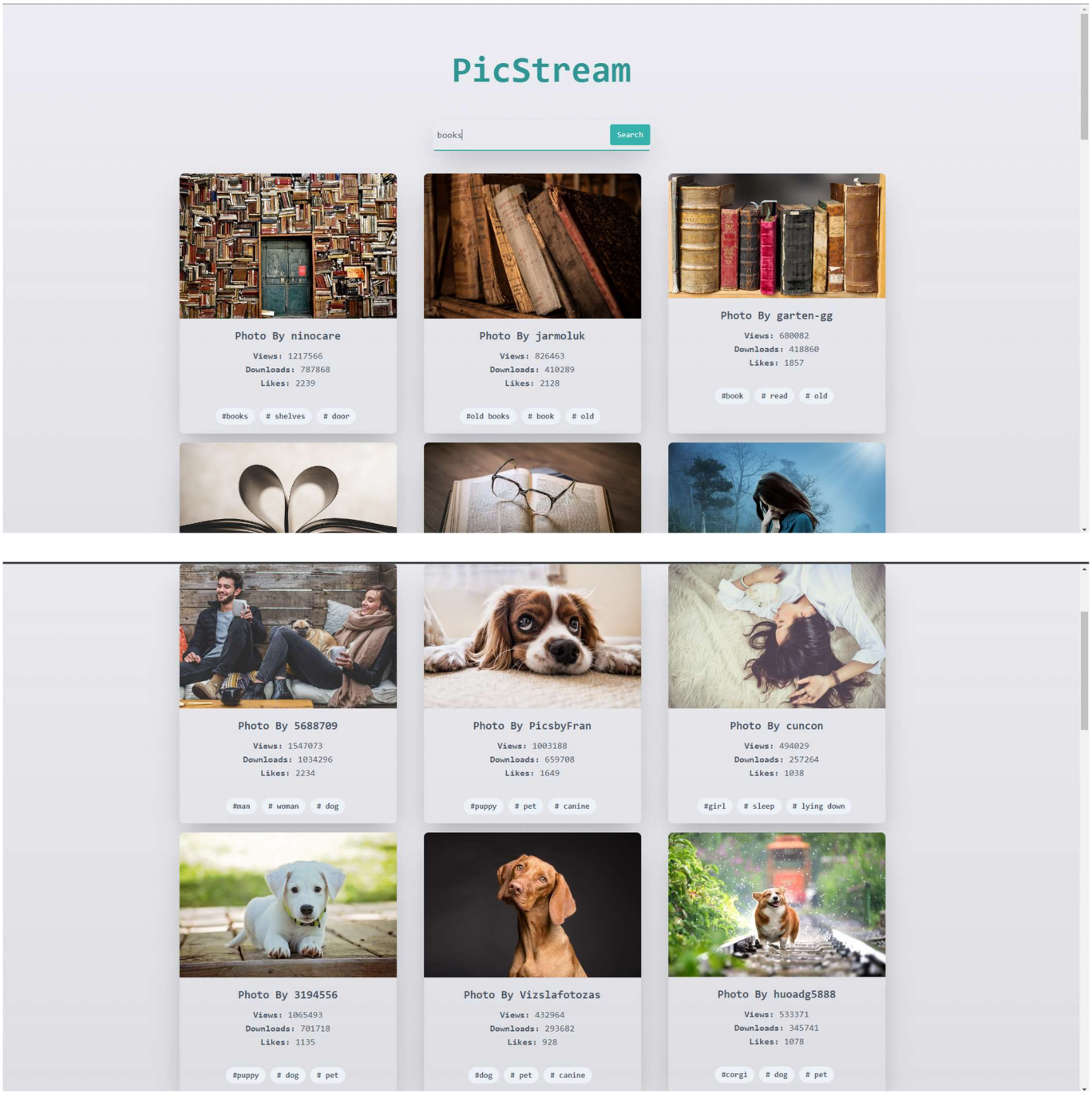
src > components > JS ImageSearch.js > ...

```jsx
import React, {useState} from 'react';

const ImageSearch = ({searchText}) => {
    const [text, setText] = useState('');

    const onSubmit = (e) => {
        e.preventDefault();
        searchText(text)
    }

  return (
    <div className = "max-w-sm rounded overflow-hidden my-10 mx-auto shadow-2xl ">
        <form onSubmit={onSubmit} className='w-full max-w-sm'>
            <div className='flex items-center border-b border-b-2 border-teal-500 py-2'>
                <input onChange={e => setText(e.target.value)} className='appearance-none bg-transparent border-none w-full text-gray-700 mr-3 py-1 px-2 leading-tight focus:outline-n
                <button className='flex-shrink-0 bg-teal-500 hover:bg-teal-700 border-teal-500 hover:border-teal-700 text-sm border-4 text-white py-1 px-2 rounded' type = "s
                    Search
                </button>
            </div>
        </form>
    </div>
  )
}

export default ImageSearch;
```

public > index.html > ...

```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <link rel="icon" href="%PUBLIC_URL%/new.png" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />
    <meta name="theme-color" content="#000000" />
    <meta
      name="description"
      content="Web site created using create-react-app"
    />
    <link rel="apple-touch-icon" href="%PUBLIC_URL%/logo192.png" />
    <!--
      manifest.json provides metadata used when your web app is installed on a
      user's mobile device or desktop. See https://developers.google.com/web/fundamentals/web-app-manifest/
    -->
    <link rel="manifest" href="%PUBLIC_URL%/manifest.json" />
    <!--
      Notice the use of %PUBLIC_URL% in the tags above.
      It will be replaced with the URL of the `public` folder during the build.
      Only files inside the `public` folder can be referenced from the HTML.

      Unlike "/favicon.ico" or "favicon.ico", "%PUBLIC_URL%/favicon.ico" will
      work correctly both with client-side routing and a non-root public URL.
      Learn how to configure a non-root public URL by running `npm run build`.
    -->
    <title>Image Gallery</title>
  </head>
  <body>
    <noscript>You need to enable JavaScript to run this app.</noscript>
    <div id="root"></div>
    <!--
      This HTML file is a template.
      If you open it directly in the browser, you will see an empty page.

      You can add webfonts, meta tags, or analytics to this file.
      The build step will place the bundled scripts into the <body> tag.

      To begin the development, run `npm start` or `yarn start`.
      To create a production bundle, use `npm run build` or `yarn build`.
    -->
  </body>
</html>

```

Output:


PicStream
books
Search
Photo By ninocare
Views: 1217566
Downloads: 787868
Likes: 2239
#books
shelves
door
Photo By jarmoluk
Views: 826463
Downloads: 410289
Likes: 2128
#old books
book
old
Photo By garten-gg
Views: 680082
Downloads: 418860
Likes: 1857
#book
read
old
Photo By 5688709
Views: 1547073
Downloads: 1034296
Likes: 2234
#man
woman
dog
Photo By PicsbyFran
Views: 1003188
Downloads: 659708
Likes: 1649
#puppy
pet
canine
Photo By cuncon
Views: 494029
Downloads: 257264
Likes: 1038
#girl
sleep
lying down
Photo By 3194556
Views: 1065493
Downloads: 701718
Likes: 1135
#puppy
dog
pet
Photo By Vizslafotozas
Views: 432964
Downloads: 293682
Likes: 928
#dog
pet
canine
Photo By huoadg5888
Views: 533371
Downloads: 345741
Likes: 1078
#corgi
dog
pet

Conclusion

The Pixabay API-based website is a user-friendly and intuitive platform that provides easy access to high-quality, royalty-free images from Pixabay. The website is built using the Pixabay API, which provides access to over a million images and videos. The website's main feature is the image search function, which allows users to search for images based on keywords.

The Pixabay API-based website is a valuable resource for individuals, businesses, and organizations that require high-quality images for personal or commercial use. By leveraging the power of the Pixabay API, the website provides a simple and efficient way for users to discover and access a vast library of stunning images. The website is constantly updated with new content, ensuring that users have access to the most up-to-date and relevant images.

Overall, the Pixabay API-based website is an excellent example of how API integration can be used to create a user-friendly and efficient platform. The website provides a valuable service to users and demonstrates the power of using APIs to enhance the functionality of websites and applications.