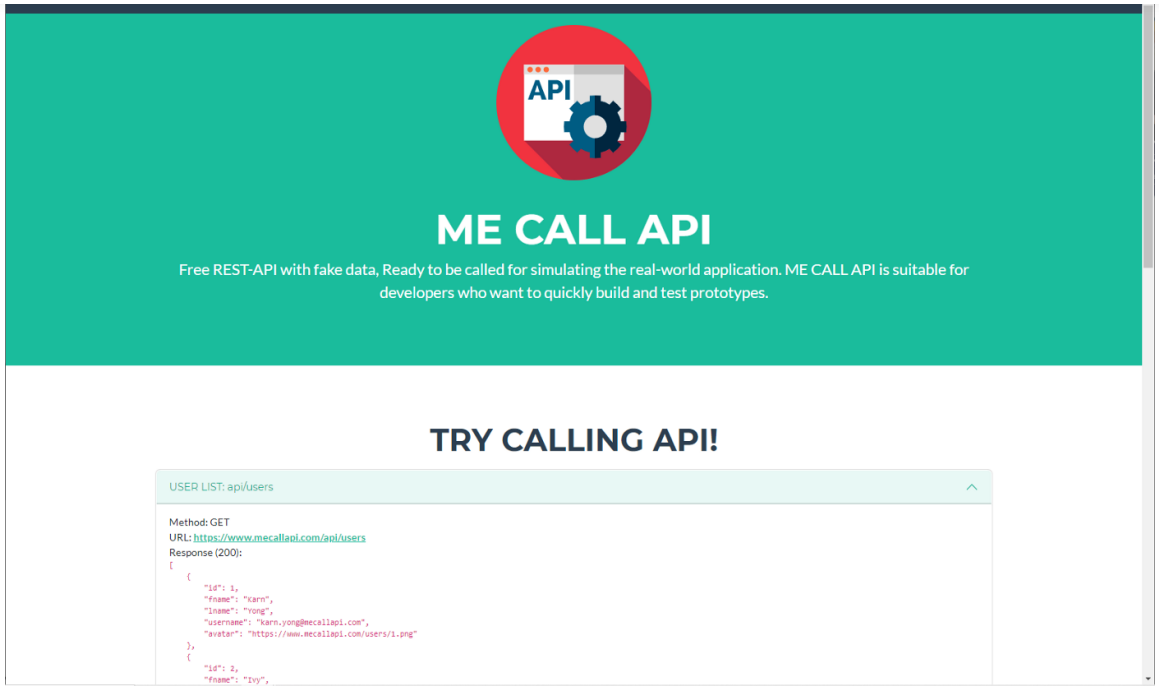
Published in JavaScript in Plain English

Karn Yongsiriwit Follow

Jun 20, 2021 · 4 min read · Listen

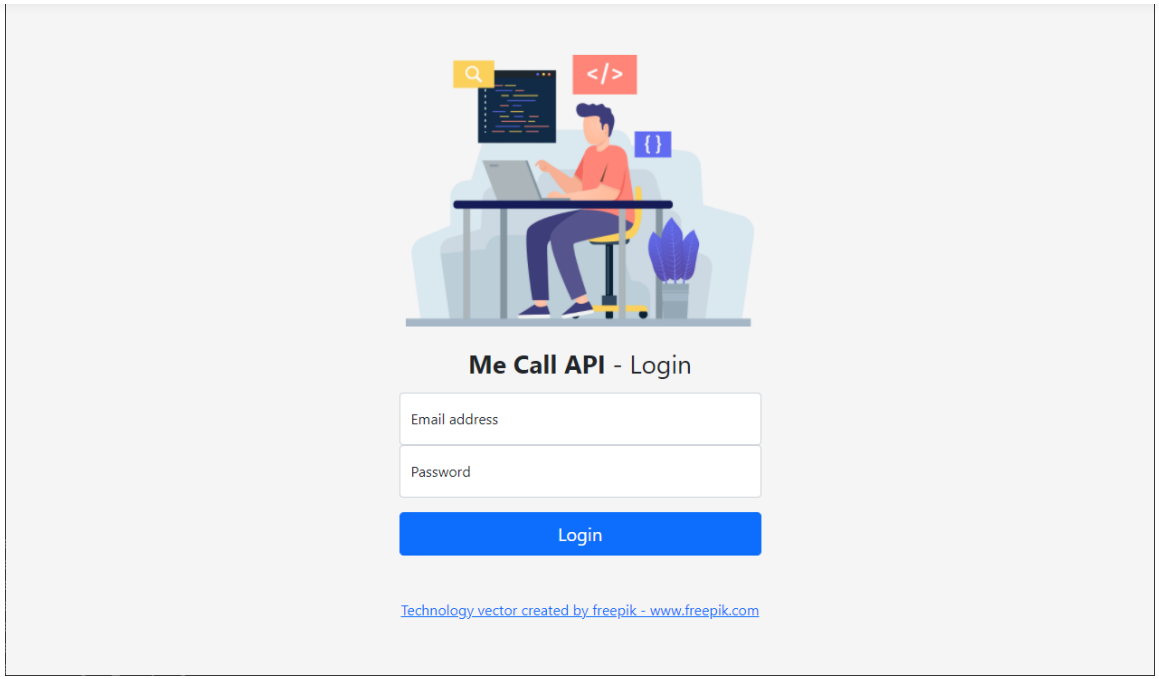
Let's Build a Website Login Page with HTML, CSS, JavaScript and an External API

This article will give you a basic tutorial for creating a website login page with HTML, CSS, JavaScript (Bootstrap 5) and an external API. The login will perform by the use of an external **API** from MeCallAPI.com (my website :D). If you want to try a mockup API for CRUD and authentication operations, feel free to check on the website.



https://www.mecallapi.com/

You can also check an example of the login page similar to what we want to achieve in this article here: https://www.mecallapi.com/login/



https://www.mecallapi.com/login/

Software Installation

It's only required a Text Editor/IDE (VS Code, Notepad, etc.) and a web browser (Chrome, Firefox, Edge, etc.) to do this tutorial!

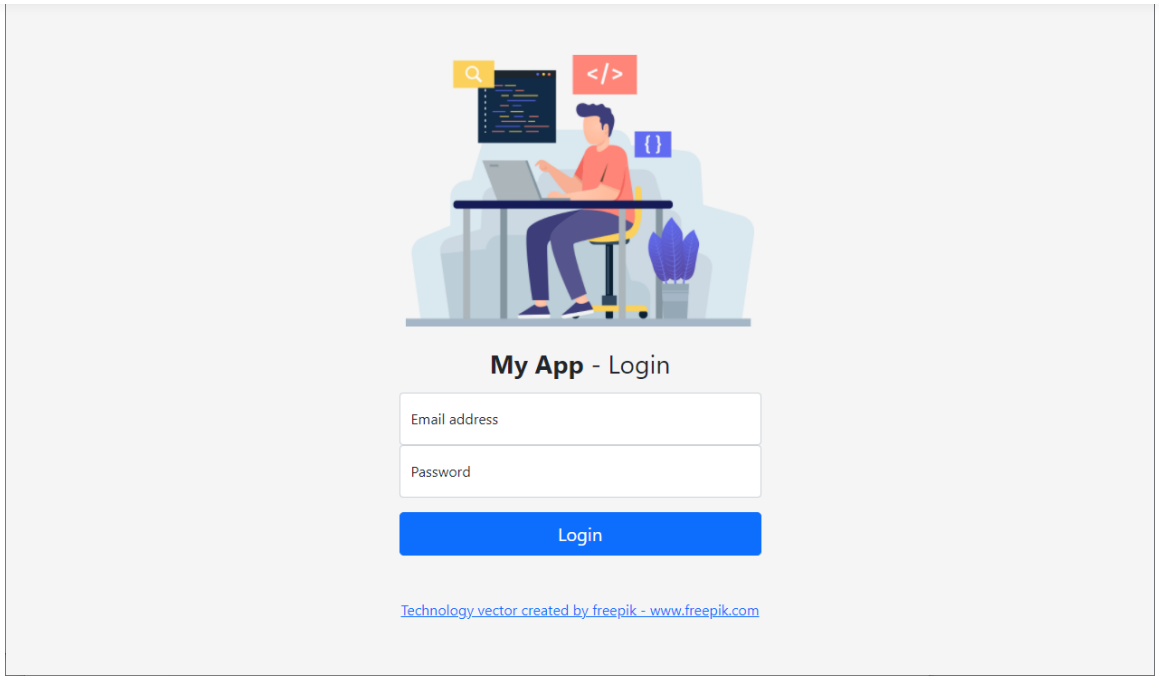
Let's Code! (HTML and CSS)

HTML documents are designed to be displayed in a web browser. There are more than a hundred of HTML elements you can choose to create an HTML file.

Let's start from creating **index.html**. I will explain a bit about what is

- **login.css**, an extra CSS (Cascading Style Sheets) to style your **login.html** in addition from the Bootstrap 5 (line 12).
- A login form with the inputs for username and password and a button (line 15–32).
- **Sweetalert**, a JavaScript library for easily creating nice popups (line 35).
- **login.js**, JavaScript file using in **login.html** to call a login **API** (line 34).

Create **login.css**

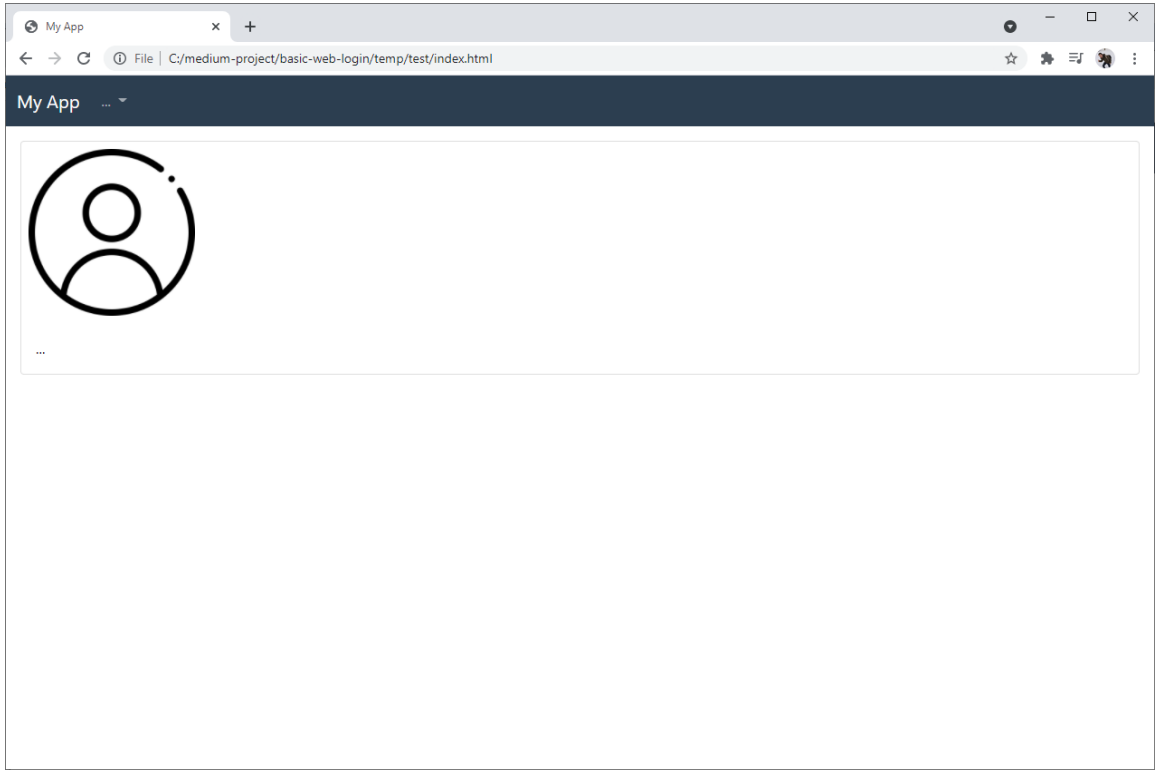


login.html on a web browser

Create **index.html** to show the information of the currently logged user. Therefore, at the end of this tutorial, this page will be only accessible when logged in.

Create **index.css**

Open **index.html** on a web browser to see the result:



index.html on a web browser

Create **login.js** to call an **API** for login based on JWT (JSON Web Token)
Standard provided by MeCallAPI.com.
API URL: https://www.mecallapi.com/api/login
Method: POST
Sample body (JSON):

```
{
    "username": "karn.yong@mecallapi.com",
    "password": "mecallapi"
}
```

Sample Response (JSON):

```
{
    "status": "ok",
    "message": "Logged in",
    "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cC..."
}
```

The **accessToken** from the response represents the authorization of a user. Therefore, we will check whether the user is logged in by this **accessToken**.

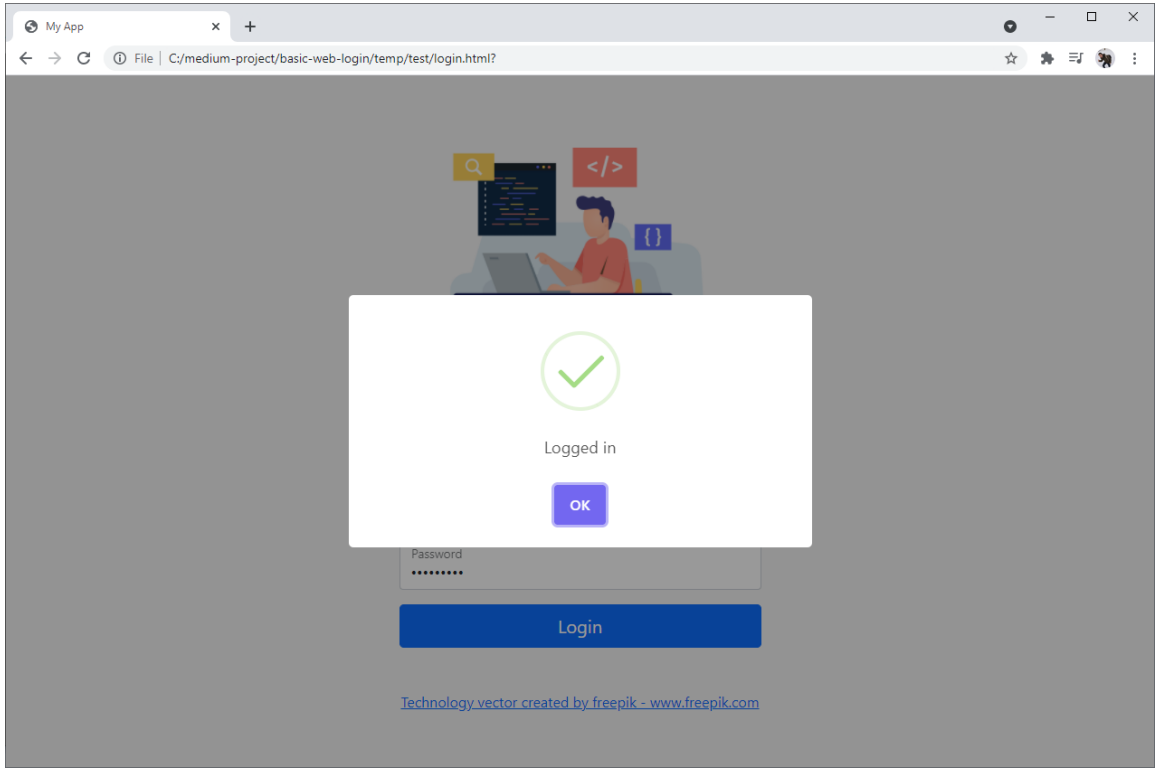
- In the JavaScript (line 1–4), we will first get the **jwt item** from **localStorage (**The localStorage allow to save key/value pairs in a web browser). If **jwt** has value which means that a user is logged in, the web browser will load index.html.
- We create a function **login** which will execute when user click **Login**

click ok in the popup, the web browser will load index.html.

Open **login.html** on a web browser, therefore input:

the same.

Click **Login** button and you will see this result:



User information with API (JavaScript)

Create **index.js** to request the **API** for retrieving the information of the currently logged user.

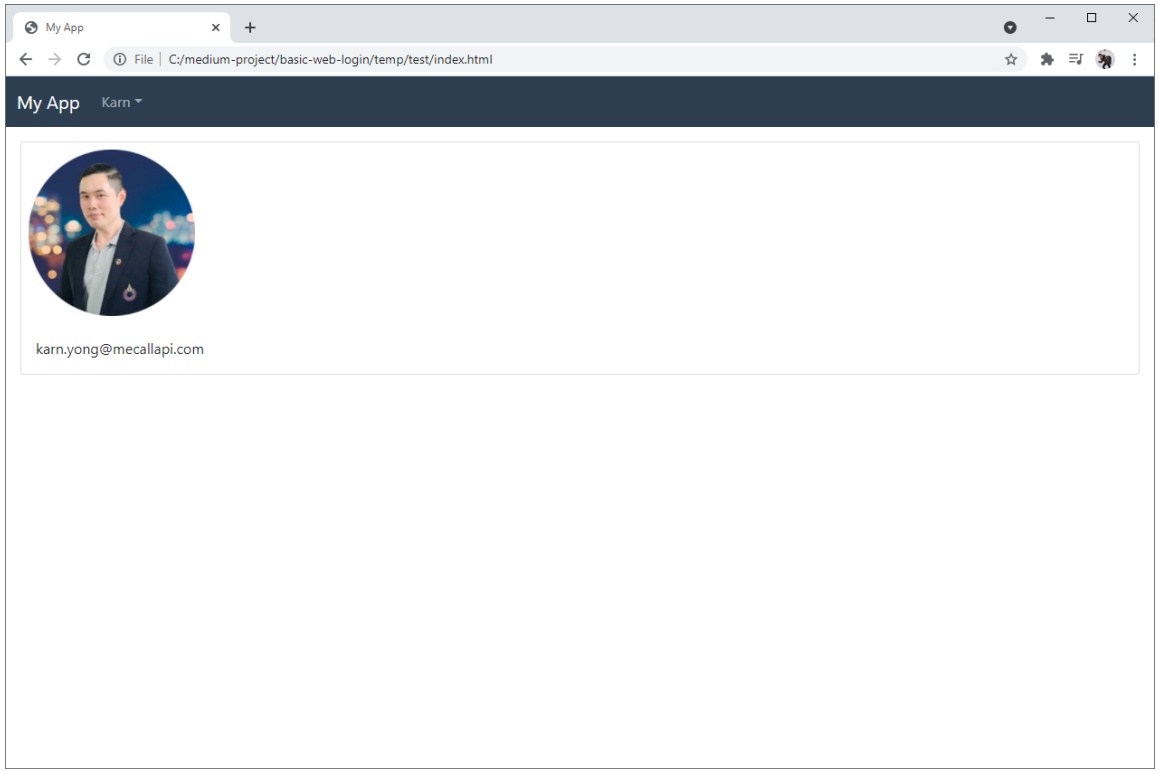
API URL: https://www.mecallapi.com/api/auth/user

Method: GET

*The API request header needs to have the value of access token of the user as

- We create a **loadUser** function to retrieve and display the currently logged user information in index.html**.** Basically, we use **XMLHttpRequest** to call the API with **jwt** in the Authorization header (Bearer). Note that this function will be called when loading index.html.
- We create a **logout** function to remove **jwt** from **localStorage** then load login.html on the web browser. This will be called when click **Logout** button.

The result:



index.html after logged in

Conclusion

That's it for building a basic login page using just HTML, CSS, JavaScript and an API. As now most of the applications are driven with API, therefore I will try to cover more about the API in the upcoming articles. Stay tuned. :D

More content at ***plainenglish.io***