**How to Implement Dark/Light Themes in a Next.js App Using Context Hook & TailwindCSS**

Initialize a Next.js application and then implement dark/light theme toggle for it using context hook and TailwindCSS

[[](https://hossein13m.medium.com/?source=post_page-----336558dd4579--------------------------------)](https://hossein13m.medium.com/?source=post_page-----336558dd4579--------------------------------)

[[JavaScript in Plain English](https://javascript.plainenglish.io/?source=post_page-----336558dd4579--------------------------------)](https://javascript.plainenglish.io/?source=post_page-----336558dd4579--------------------------------)

[Hossein Mousavi](https://hossein13m.medium.com/?source=post_page-----336558dd4579--------------------------------)

·

Follow

Published in

[JavaScript in Plain English](https://javascript.plainenglish.io/?source=post_page-----336558dd4579--------------------------------)

·

6 min read

·

Jun 8, 2022

468

One of the most requested features of users from any proper web application is switching between light and dark modes. In this article, we will build a template Next.js application, and with the help of TailwindCSS, we will implement a toggle mechanism between dark and light themes. I have used React's hooks for this template, such as the *useContext, useState, and useEffect*.

So without any further ado, let's get started.



Partial image of Carina Nebula

***TL;DR,****You can find the project’s*[*GitHub repository*](https://github.com/hossein13m/nextjs-tailwind-dark-theme)*. It is a template repository, and you can use it to create a new project based on it. You could give it a ⭐ if you like! You can also check out a live demo of the project*[*here*](https://nextjs-tailwind.hmousavi.dev/)*.*

**Table of Contents:**

* Initializing a Next.js Application
* Changing the project's structure (*optional*)
* Adding TailwindCSS to the project and configuring it
* Implement the layout of the project (code available in the GitHub repo)
* Using the useContext hook to change themes widely in the application and storing the preferred theme in the localStorage
* Add our themeContext to the main component to use it widely in the application
* Creating a navbar with a button to toggle between themes

**Initializing a Next.js application and Adding TailwindCSS to Project**

The first step would be initializing a new Next.js application. For that, ensure you have installed *node.js* on your machine and then use the following command:

npx create-next-app@latest --typescript  
# or  
yarn create next-app --typescript  
# or  
pnpm create next-app --typescript

The --typescript option provides that you will benefit from TypeScript in your project, which I highly recommend you use.

**Project Structure (Optional)**

I personally prefer to include my project's files and folders in a separate directory called ***src****.*But this step is entirely up to you whether you want to do it or not. The only difference is when we want to configure the tailwind configuration file and when we want to address our files which we will see in a second. For this, I will create a new directory and transfer *pages* and *styles* folders there. The project's structure will change throughout the development process as well. For example, we will create a new directory called *components*for storing our components or another one for storing our *context* called *store.*All the following directories will be held inside the *src* folder.



Photo by [Evgeniy Surzhan](https://unsplash.com/@followtherabbit?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com/?utm_source=medium&utm_medium=referral" \t "_blank)

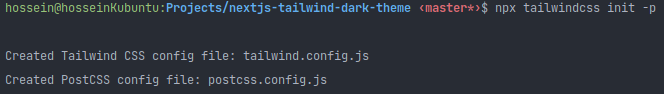
**Adding TailwindCSS to project**

After the initialization finishes, we can add TailwindCSS to our project. First, we need to install three packages using the following command:

npm install -D tailwindcss postcss autoprefixer

Since we use the -D option, all of the following packages will be installed as **devDependencies**. The next step would be creating two configuration files for *tailwind* and *postcss*.

npx tailwindcss init -p



As you can see, two configuration files will be made once we run the above command!

We do not need to make any changes to the postcss.config.js file for our purpose, and we will leave it that way.  
On the other hand, the *tailwind.config.js* file is responsible for configuring Tailwind options in our application. We will dive deeper into it in another article, but for the sake of this article, we just need to adjust a few things to implement the dark/light theme. The following configuration file is the one we need in our application.

The key takeaways from the configuration file are the following:

* If you have changed the project's structure, you need to include the ***src*** in the content array at the beginning of your path.
* You need to add the fifth line to the configuration and explicitly indicate the dark mode by using the ***class***. The project will use a dark theme if we add the ***dark*** class to our HTML body.

Now you need to **import Tailwind into the main CSS** file of your project. Add the following lines of codes inside your ***global.css*** file.

// import tailwind to global.css file  
@tailwind base;  
@tailwind components;  
@tailwind utilities;

**Creating the layout of the project**

We just added a navbar inside our layout with a button to toggle between themes for simplicity. We will store both our *layout* and *header*inside the*components*directory.  
I am not going through the layout implementation as it is an HTML and CSS task. You can find it in the GitHub repository. But I will dive into the implementation of the header which we have a button to toggle the theme in a few seconds.

[**Link To GitHub Repository**](https://github.com/hossein13m/nextjs-tailwind-dark-theme)[**Link To Live Demo**](https://nextjs-tailwind.hmousavi.dev/)



Photo by [Lautaro Andreani](https://unsplash.com/@lautaroandreani?utm_source=medium&utm_medium=referral) on [Unsplash](https://unsplash.com/?utm_source=medium&utm_medium=referral" \t "_blank)

**The context for toggling theme**

Create another directory called *store* and then create a .tsx file with the following contents:

The Following file toggle theme, add dark to the body of your class so tailwind can work with it. It also stores your preferred theme on the localStorage, so you won't miss it on reload. The initialThemeHandler function sets proper settings based on whether or not the localStorage is empty. By default, we want our application to be in dark mode.

Now that we have written our context, let's get access to it on our component and use it.

**Add Context to the main component.**

Since we want to benefit from the ***context*** of our whole application, we need to add it to \_app.tsx file like the following:

Please note that our layout should be a child of our custom hook since we want it to work with our context.

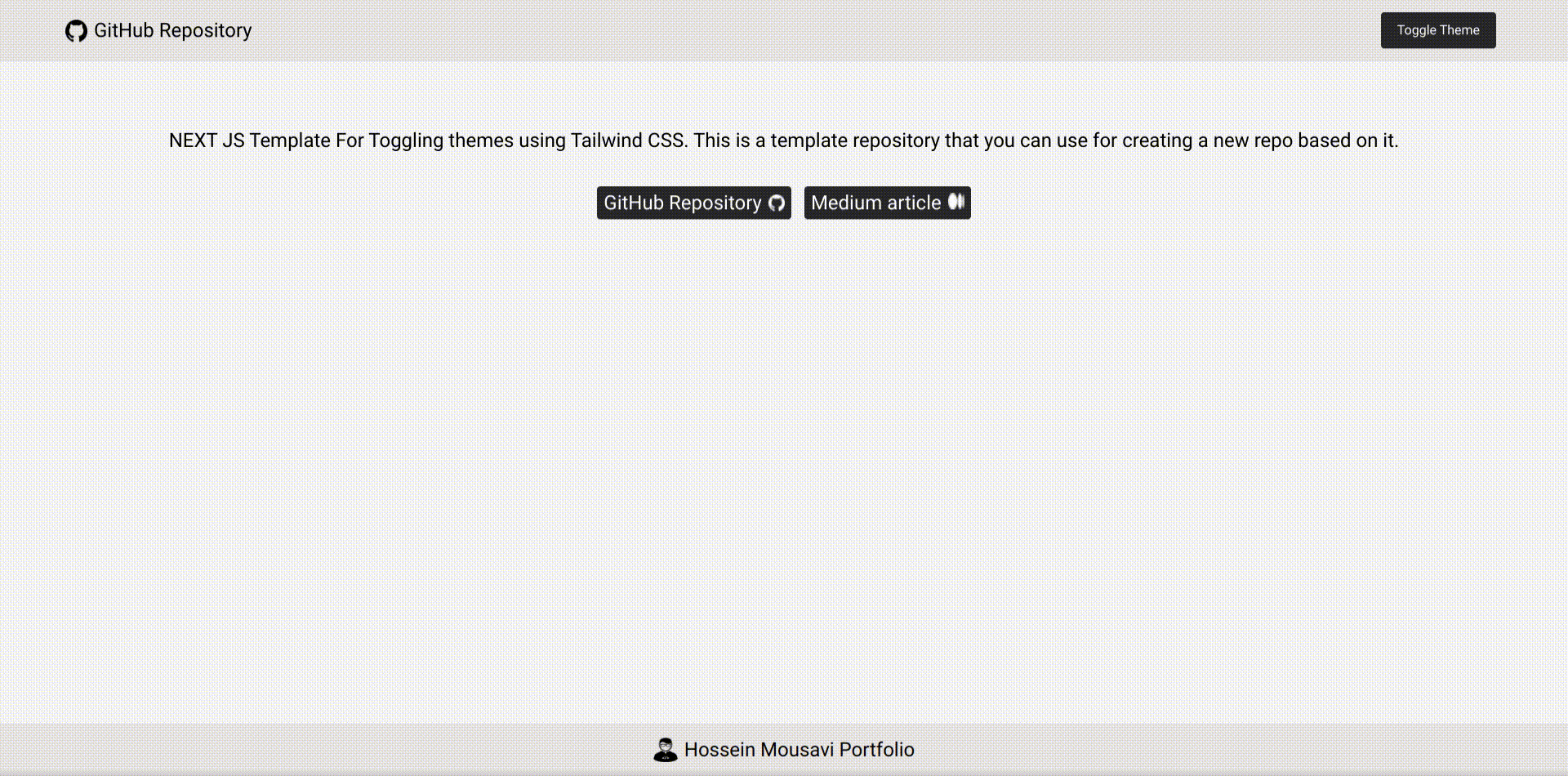
**Toggle Theme on the header component**

We want to implement a button to toggle between themes, but this button will not be in the same component where our context code is located. We need to import the ***useContext*** hook in our header component, which takes our context as an argument. And on the function responsible for the click event of the toggle button, we need to call our custom context to change the theme. The explanation might seem confusing, but let's look at the code, and it will be crystal clear for you!

As you can take a look, I have used Tailwind classes heavily to customize the button's style. The dark:<className> is the syntax that Tailwind recognizes and uses the class after dark: if we append a class to the body of our page called dark.

**The Final Output**

The final output would be something like this:



I have provided a GitHub repository and published the final production using [Netlify](https://netlify.com/), which you can find their link below:

[**GitHub Repository**](https://github.com/hossein13m/nextjs-tailwind-dark-theme)**|**[**Live Demo**](https://nextjs-tailwind.hmousavi.dev/)

**Share with your friends! Clap 👏 as max as 50 times and give a star ⭐ to its GitHub repository**

Please do not hesitate to share your thoughts and ideas with me. You can reach me on [Twitter](https://twitter.com/Hossein13M) or find me on other social media by visiting my portfolio.

**[Hossein Mousavi - Software Developer](https://www.hmousavi.dev/?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[Hi there! 👋🏻 Discover Hossein Mousavi's Space and Find Out My Works. Feel Free to Get In Touch With Me and More!](https://www.hmousavi.dev/?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[www.hmousavi.dev](https://www.hmousavi.dev/?source=post_page-----336558dd4579--------------------------------" \t "_blank)

**Read more from me:**

**[A Detailed Look at filter(), map(), and reduce() in JavaScript](https://javascript.plainenglish.io/take-a-good-look-at-filter-map-and-reduce-in-javascript-18d72f483ada?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[A deep dive into the world of filter(), map(), and reduce() methods in JavaScript.](https://javascript.plainenglish.io/take-a-good-look-at-filter-map-and-reduce-in-javascript-18d72f483ada?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[javascript.plainenglish.io](https://javascript.plainenglish.io/take-a-good-look-at-filter-map-and-reduce-in-javascript-18d72f483ada?source=post_page-----336558dd4579--------------------------------" \t "_blank)

**[API-First Approach Complete Guide](https://itnext.io/a-complete-guide-to-the-api-first-approach-ecd796dd0f10?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[One of the many approaches you might take to develop an application through your software development career is…](https://itnext.io/a-complete-guide-to-the-api-first-approach-ecd796dd0f10?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[itnext.io](https://itnext.io/a-complete-guide-to-the-api-first-approach-ecd796dd0f10?source=post_page-----336558dd4579--------------------------------" \t "_blank)

**[Multiple Interceptors in Angular](https://medium.com/codex/multiple-interceptors-in-angular-e0880b2f7d91?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[One of the amazing features that Angular provides is the interceptors, but what does an Interceptor do, and can we…](https://medium.com/codex/multiple-interceptors-in-angular-e0880b2f7d91?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[medium.com](https://medium.com/codex/multiple-interceptors-in-angular-e0880b2f7d91?source=post_page-----336558dd4579--------------------------------" \t "_blank)

**[Husky 6 Lint (prettier + eslint) and commitlint for JavaScript Projects](https://medium.com/angular-in-depth/husky-6-lint-prettier-eslint-and-commitlint-for-javascript-project-d7174d44735a?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[Programming is a teamwork job so we must assure that our codebase is clean and usable for everyone in the team with the…](https://medium.com/angular-in-depth/husky-6-lint-prettier-eslint-and-commitlint-for-javascript-project-d7174d44735a?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[medium.com](https://medium.com/angular-in-depth/husky-6-lint-prettier-eslint-and-commitlint-for-javascript-project-d7174d44735a?source=post_page-----336558dd4579--------------------------------" \t "_blank)

**[Angular forms (Reactive Form) including Angular Material and Custom Validator](https://medium.com/angular-in-depth/angular-forms-reactive-form-including-angular-material-and-custom-validator-9ef324cc3b08?source=post_page-----336558dd4579--------------------------------" \t "_blank)**

[Forms are major parts of every Angular project and in this article, we want to implement a Reactive Angular form with a…](https://medium.com/angular-in-depth/angular-forms-reactive-form-including-angular-material-and-custom-validator-9ef324cc3b08?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[medium.com](https://medium.com/angular-in-depth/angular-forms-reactive-form-including-angular-material-and-custom-validator-9ef324cc3b08?source=post_page-----336558dd4579--------------------------------" \t "_blank)

[Nextjs](https://medium.com/tag/nextjs?source=post_page-----336558dd4579---------------nextjs-----------------)

[React](https://medium.com/tag/react?source=post_page-----336558dd4579---------------react-----------------)

[JavaScript](https://medium.com/tag/javascript?source=post_page-----336558dd4579---------------javascript-----------------)

[Tailwind Css](https://medium.com/tag/tailwind-css?source=post_page-----336558dd4579---------------tailwind_css-----------------)

[Context](https://medium.com/tag/context?source=post_page-----336558dd4579---------------context-----------------)