CSS Frameworks and You

How CSS frameworks can help you with your React projects.
And your Front-End Development in general

Mario Altagracia Jimenez

About me

- Former Software Engineer at Socially Elite Pro

- Volunteer Software Engineer for Austin Public Library. Formally volunteered for Open Austin

- Sometimes build SaaS and online businesses - APIplaza.co | petwarezinc.com

- Tech blogger: https://medium.com/@7adam7e

- Sometimes create music. Sometimes do photography

What the presentation is about:

- CSS frameworks. What are they all about?

- How you can use them in most projects?

- Multiple CSS framework examples

- Getting you into using a framework via a step by step guide

What I hope that you get from it

- Improve your Front-End Development

- Attract more users, traffic, or sales to your service

- Stop spending a lot of time center aligning website elements

What are CSS frameworks?

- Ready to use code rules and parameters used for Front-End Dev Work

- Comes with its own styling, layouts, and font styling for each text element

- Lets you easily create responsive grids, forms, buttons, and other elements for a website

- Provides consistent styling for your Front-End Dev work

- Great for adding responsive design to your project

Where you can implement them:

- Small web applications

- Legacy apps

- P.W.A

- SaaS

- Sales pages / landing pages

- Personal websites or client sites

- Codebases with separate CSS libraries referenced

Pros of using frameworks

- Extremely small libraries. Some take only **2 kilobytes** of data to install

- Cut down Dev time

- Ease up testing and user experiences on multiple platforms (smartphones, desktops, tablets, etc)

- Easy to update and change

- They work on any type of JS/React/JS framework project

- Free to use, try and install

- Framework documentation and guides are always available

Multi platform example

- [Example video coming soon]

Cons of CSS Frameworks

- CSS that's already installed in projects loses priority

- Sometimes rigid for Front-End Development. The CSS rules that the framework become the main rules

- It can turn into a mess of code if you don't take good care and organize everything

A few sample frameworks
There's many options to choose from:

Frameworks are all available on npm.js

Free and easy to install on any React project

Some example websites from frameworks

- https://bulma.io/expo/

- https://icons.getbootstrap.com/

- http://getskeleton.com/examples/landing/

-

Demo project

- URL: https://github.com/tjdev7/reactatx-css-frameworks-samples

- GitHub Commands:

- https://github.com/tjdev7/reactatx-css-frameworks-samples.git

- git@github.com:tjdev7/reactatx-css-frameworks-samples.git

- gh repo clone tjdev7/reactatx-css-frameworks-samples

- Shortened URL:

- QR code:

My small advice

- Read the documentation. Follow the walkthrough guides before doing anything

- Use a separate Git branch / container / directory to test things out

- Experiment with everything before pushing to main/master

Conclusion

- Hope you liked it

- Hope it helped you out personally

- Hope that you make some great work with it

-

Reach out to me:

- Github: https://github.com/tjdev7/

- Website: https://tjdev7.co/

- Twitter/X: https://twitter.com/Tjdev7

- LinkedIn: https://www.linkedin.com/in/mariojimenez1/

Additional resources

- https://mui.com/

- https://getbootstrap.com/

- https://tailwindcss.com/

- https://get.foundation/

- https://bulma.io/

- http://getskeleton.com/

- https://getuikit.com/

Npm links

- https://www.npmjs.com/package/bootstrap

- https://www.npmjs.com/package/tailwindcss

- https://www.npmjs.com/package/uikit

- https://www.npmjs.com/package/bulma

- https://www.npmjs.com/package/@mui/material