PRO-XY

Data Monitoring Dashboard

FrontEnd & User manual Documentation

Team Members

Jailash Natarajan

Jawahar Ravi Venugopalan

Sivaranjani Soundarraj

Srimanikandan Ravikumar

Vaishnavi Suvetha Srinivasan

Introduction

React version 18.2.0 is used for FrontEnd application. React has been designed from the start for gradual adoption and React makes it painless to create interactive UIs. Design simple views for each state in your application and React will efficiently update and render just the right components when your data changes. Along with Reactjs we have Tailwind CSS for styling and showing as Responsive web design.

React components implement a render() method that takes input data and returns what to display. This example uses an XML-like syntax called JSX. Input data that is passed into the component can be accessed by render() via this.props.

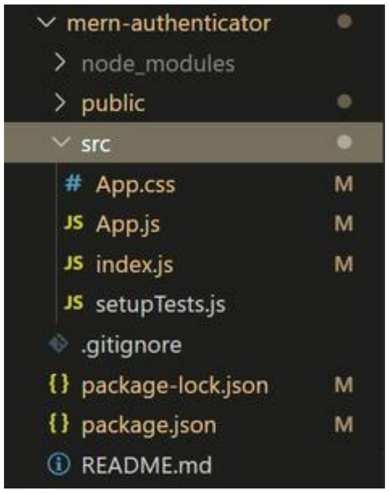
React expects that the rendered content is identical between the server and the client. It can patch up differences in text content, but you should treat mismatches as bugs and fix them. In development mode, React warns about mismatches during hydration. There are no guarantees that attribute differences will be patched up in case of mismatches. This is important for performance reasons because in most apps, mismatches are rare, and so validating all markup would be prohibitively expensive.

If a single element's attribute or text content is unavoidably different between the server and the client (for example, a timestamp), you may silence the warning by adding suppressHydrationWarning={true} to the element. It only works one level deep and is intended to be an escape hatch. Don't overuse it. Unless it's text content, React still won't attempt to patch it up, so it may remain inconsistent until future updates.

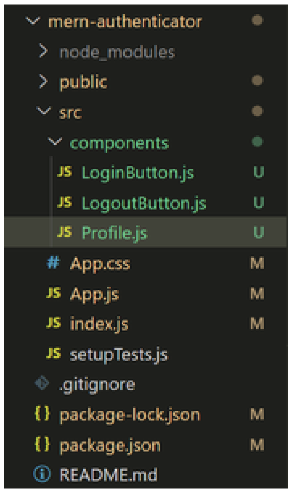
Authentication

User authentication is the process of securing user information through the use of some parameters like username, password, and more. This helps the user to access his perks and features on a particular website. Such authentication is used almost everywhere today. Some examples would be banking, booking tickets, online video streaming platforms, etc. We will be using React JS, a Javascript frontend library, and Auth0, a framework that will help us in making our authentication user-friendly as well as secure.

Project Structure: Your Folder structure should look something like this



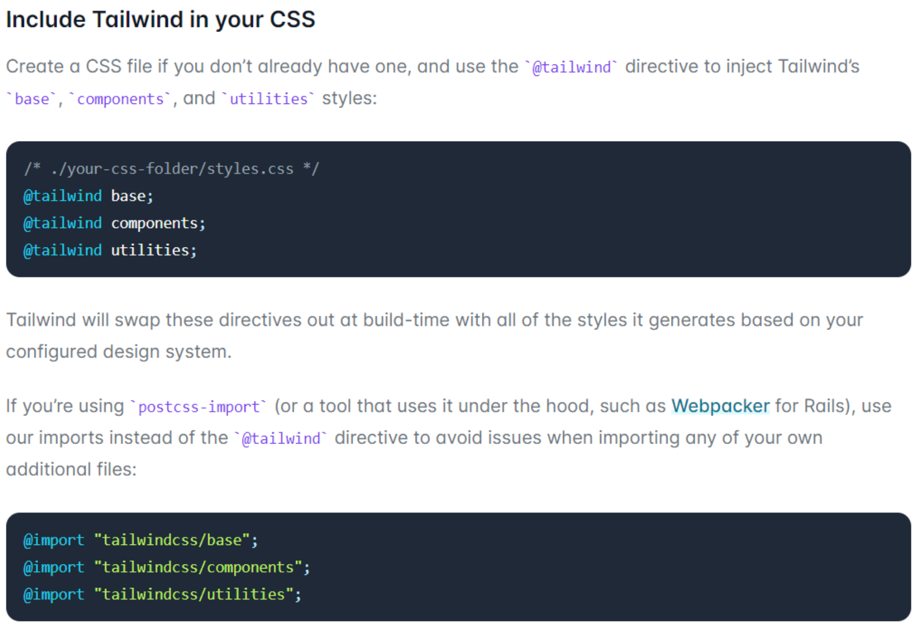
```
mern-authenticator
  > node_modules
  > public
  v src
      App.css          M
      App.js           M
      index.js         M
      setupTests.js
    .gitignore
    package-lock.json  M
    package.json       M
    README.md
```



```
mern-authenticator
  > node_modules
  > public
  v src
    v components
        LoginButton.js     U
        LogoutButton.js    U
        Profile.js         U
      App.css              M
      App.js               M
      index.js             M
      setupTests.js
    .gitignore
    package-lock.json      M
    package.json           M
    README.md
```

Tailwind CSS:

Tailwind CSS is a highly customizable, low-level CSS framework that gives you all of the building blocks you need to build bespoke designs without any annoying opinionated styles you have to override.



Include Tailwind in your CSS

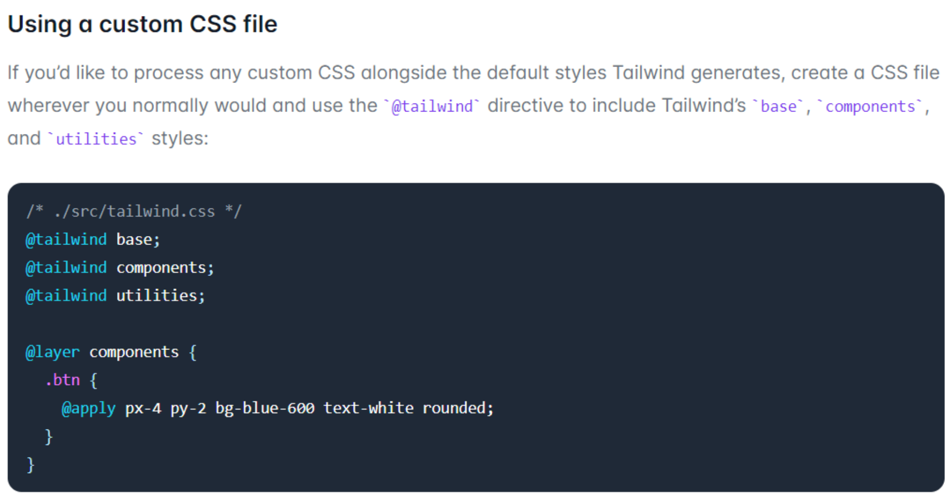
Create a CSS file if you don't already have one, and use the `@tailwind` directive to inject Tailwind's `base`, `components`, and `utilities` styles:

```css
/* ./your-css-folder/styles.css */
@tailwind base;
@tailwind components;
@tailwind utilities;
```

Tailwind will swap these directives out at build-time with all of the styles it generates based on your configured design system.

If you're using `postcss-import` (or a tool that uses it under the hood, such as Webpacker for Rails), use our imports instead of the `@tailwind` directive to avoid issues when importing any of your own additional files:

```css
@import "tailwindcss/base";
@import "tailwindcss/components";
@import "tailwindcss/utilities";
```



Using a custom CSS file

If you'd like to process any custom CSS alongside the default styles Tailwind generates, create a CSS file wherever you normally would and use the `@tailwind` directive to include Tailwind's `base`, `components`, and `utilities` styles:

```css
/* ./src/tailwind.css */
@tailwind base;
@tailwind components;
@tailwind utilities;

@layer components {
  .btn {
    @apply px-4 py-2 bg-blue-600 text-white rounded;
  }
}
```

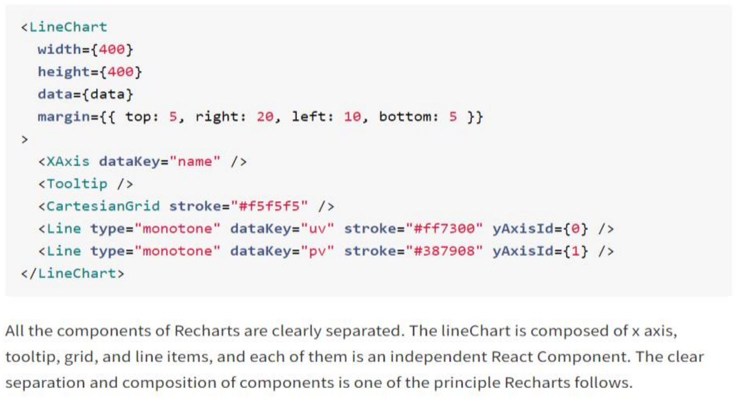
Recharts:

Recharts is a redifned chart library built with **React** and **D3** (Data Driven Documents) . Recharts version used in this project is 2.10.2

The main purpose of this library is to help you to write charts in React applications without any pain. Main principles of Recharts are:

- **Simply** deploy with React components.
- **Native** SVG support, lightweight depending only on some D3 submodules.
- **Declarative** components, components of charts are purely presentational.

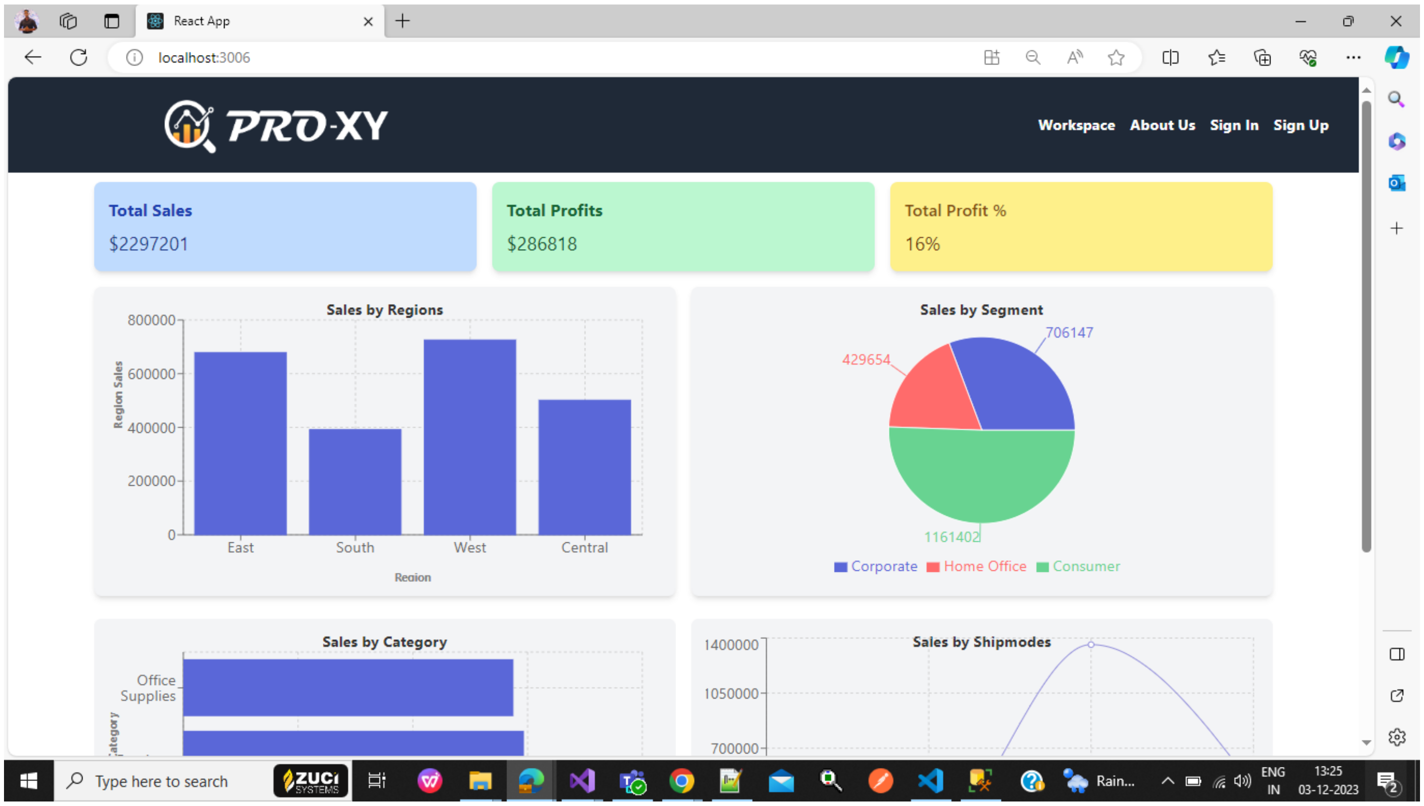
Creating documentation for a Web API is crucial for developers and users who want to understand how to interact with your API. Below is a template and guidelines for documenting a Web API:



```jsx
<LineChart
  width={400}
  height={400}
  data={data}
  margin={{ top: 5, right: 20, left: 10, bottom: 5 }}
>
  <XAxis dataKey="name" />
  <Tooltip />
  <CartesianGrid stroke="#f5f5f5" />
  <Line type="monotone" dataKey="uv" stroke="#ff7300" yAxisId={0} />
  <Line type="monotone" dataKey="pv" stroke="#387908" yAxisId={1} />
</LineChart>
```

All the components of Recharts are clearly separated. The lineChart is composed of x axis, tooltip, grid, and line items, and each of them is an independent React Component. The clear separation and composition of components is one of the principle Recharts follows.

Frontend Application PRO-XY webapp using React, Recharts and TailwindCSS


React App
localhost:3006
PRO-XY
Workspace
About Us
Sign In
Sign Up
Total Sales
$2297201
Total Profits
$286818
Total Profit %
16%
Sales by Regions
Region Sales
Region
East
South
West
Central
800000
600000
400000
200000
0
Sales by Segment
706147
429654
1161402
Corporate
Home Office
Consumer
Sales by Category
Office Supplies
Category
Sales by Shipmodes
1400000
1050000
700000
Type here to search
Rain...
ENG IN
13:25
03-12-2023