Overview

Panely React is a multipurpose admin dashboard template based on React NextJS and Bootstrap. This template has a huge collection of essential and reusable UI components for you to design interfaces and powerful web applications. This template provides many widgets and support for light and dark themes. It is a fully clean and responsive dashboard template design for all types of devices and works on all major web browsers.

We built this template to be easy to customize. Every widgets and components are modular, reusable and customizable. The layout and navigation menu is easy to customize. Out of the box, this template is built-in with Redux and Redux Saga. All styles are fully developed with SCSS, you can easily change this template style.

Get starting

1. First, you must have Node.js installed on your computer
2. Extract the downloaded package from the marketplace
3. Start the console and go to `/template/` directory from the package that was extracted
4. Install all dependencies needed by running `yarn install`
5. This template requires Firebase for authentication, you must set the Firebase configuration. Get your Firebase configuration on your Firebase console page at **Project Settings -> General -> Your Apps** and copy the configuration and replace into `/template/src/config/firebase.config.jsx`
6. After that, you must set Firebase Admin key. Go to Project **Settings -> Service Accounts** on Firebase console page and generate the key and replace into `/template/src/config/firebase-service-account.json`
7. Start the development server by running `yarn dev` and keep the console open and open the link that displayed on the console. Press `CTRL+C` to stop the server

Directory structure

Panely React comes with a simple and organized directory structure for easy to understand and maintainability. Look at the treeview below to find the basic template directory structure.

Template directory `/template/`

- **public/** - contains static assets
- **src/** - contains source codes
  - **components/** - contains modular components
  - **config/** - contains configuration file
  - **modules/** - contains library codes
  - **pages/** - contains page components
  - **store/** - contains all redux files
  - **styles/** - contains all stylesheet files
- **jsconfig.json** - compiler configuration
- **package.json** - general package information

Development tools

This template uses default NextJS build tool without custom configuration. Below are all the build tool commands.

Command	Description
`yarn dev`	Start development server
`yarn build`	Build codes for production purpose
`yarn start`	Serve the production codes

Page customization

In NextJS, a page is a React Component exported from a javascript file in the `/template/src/pages/` directory. Each page is associated with a route based on its file name.

Example: If you create `/template/src/pages/about.js` that exports a React component like below, it will be accessible at `/about`.

For more information and examples look at the official documentation.

Static file serving

NextJS can serve static files, like images, under a folder called `public` in the root directory. Files inside `public` can then be referenced by your code starting from the base URL (`/`).

For example, if you add an image to `/template/public/me.png`, you can access it in `/me.png` url. This folder is also useful for `robots.txt`, `favicon.ico`, Google Site Verification, and any other static files (including `.html`)!

For more references look at the documentation

Layout

You can wrap the page component with `withLayout` function to apply the layout component. This template provides 2 types of layout: `default` and `blank` layouts. The blank layout will remove all navigation components. To change layout type, you can use the second parameter of the wrapper function `withLayout(Page, "default|blank")`.

To configure the layout components you can set the configuration in `/template/src/config/page.config.jsx`.

Navigation menu

For customizing the side navigation menu, you can change the configuration in `/template/src/config/menu.config.jsx`. Look at the codes to learn how the navigation menu is working.

Authentication

This template uses Firebase for authenticating and you must set Firebase configuration. Wrap your page component with `withAuth` to make protected page that can access only for authorized users. If the users are not authorized, the page will redirect to login page.

Redux

By default, this template comes with Redux and Redux Saga. You can find all redux source codes in `/template/src/store/`, that folder contains all reducers, actions and saga middlewares.

Stylesheet

This template uses SCSS for handling the CSS efficiently. You can find all SCSS files in `/template/src/styles/`. We remove several Bootstrap components because we replace them with our custom components.

You can easily customize the color palette and other options by editing the variables in `/template/src/styles/core/_variables.scss`.

Credits

- Apex Charts
- Feather
- Firebase
- Font Awesome
- Isomorphic Unfetch
- NextJS
- Next Absolute URL
- Next Redux Wrapper
- Nookies
- NProgress
- Quill
- React
- React Hook Form
- React Slick
- React Sticky Node
- React Transition Group
- Redux
- Redux DevTools Extension
- Redux Saga
- Simplebar
- SortableJS
- Sweet Alert 2
- Popper
- Yup