

Structure

File structure

Once you have everything setup (if you haven't go back to the Installation (/app/docs/installation) page and finish the installation) you'll notice a file structure similar to this in your `src` folder:

```
sass
|-- <project-name>
|   |-- colors.scss
|   |-- font-faces.scss
|   |-- main.scss
|   |-- theme.scss
|   |-- pages
|   |   `-- _pages.scss
|   |-- print
|   |   `-- _print.scss
|   |-- theme
|   |   |-- _colors.scss
|   |   |-- _theme-mixin.scss
|   |   |-- components
|   |   |   |-- _breadcrumbs.scss
|   |   |   |-- _button.scss
|   |   |   |-- _components.scss
|   |   |   |-- _datepicker.scss
|   |   |   |-- _dropdown.scss
|   |   |   |-- _forms.scss
|   |   |   |-- _ion-rangeSlider.scss
|   |   |   |-- _ion-tabs.scss
|   |   |   |-- _link.scss
|   |   |   |-- _messenger.scss
|   |   |   |-- _pagination.scss
|   |   |   |-- _panel.scss
|   |   |   |-- _select2.scss
|   |   |   `-- _vex.scss
|   |   `-- sections
|   |       |-- _body.scss
|   |       |-- _footer.scss
|   |       |-- _header.scss
|   |       |-- _sections.scss
|   |       `-- _sidebar.scss
```

colors.scss

This file contains a list of color combinations that are provided by default along with Rubix. You can add/remove from this list. We have provided a handy mixin called `generateColors($name, $color, $hover-color)` that colors different components like lists, navs, buttons, menus, timeline etc.

You can also override the foreground color of any element by prefixing the color name with **fg-** (ex: **fg-deepred**) and background color by prefixing color name with **bg-** (ex: **bg-darkgreen45**).

Similary we have also provided classnames for borders and different states (like hover/focus/active etc). Please refer to the demo JSX files for usage of these different color combinations on various components. Certain components have a `bsStyle` property (ex: Button) which takes the color name directly and generates the appropriate classname behind the scenes. Also, please refer to the `sass/app/theme/_colors.scss` file for the color mixin definition.

font-faces.scss

This file contains a list of web icon fonts that come packaged with Rubix. You can add/remove fonts from this list. If you are going to add a new icon font make sure you add a font partial in **src/sass/app/fonts**. Also make sure you have dropped a corresponding TTF file into the dropbox located at: **public/fonts/dropbox/app**

As an example lets take the case of Climacons font. The TTF file dropped into **public/fonts/dropbox/app** is **Climacons-Font.ttf**. The corresponding partial **src/sass/app/fonts/_climacons.scss** contains a reference to this filename in the font-family:

```
.climacon:before{
  font-family: 'Climacons-Font';
  ...
}
```

If you are using text web fonts like Lato/Open Sans there is no need to create a separate partial file and can directly reference it in your stylesheets after dropping the required TTF file into your project's font dropbox.

main.scss

This file contains a list of all third party plugin styles which come packaged with Rubix that you can add/remove from. To provide page specific styling please create partials in **src/sass/app/pages** and require them in the **src/sass/app/pages/_pages.scss** partial. Similarly include your print styles in the print partial located at **src/sass/app/print/_print.scss**

theme.scss

This file a mixin called `theme-maker($name, $list)` which takes two parameters: a name and a list of default settings. This mixin includes another mixin called `theme-mixin($name, $list)`. The theme-maker mixin is used to for creating individual themes with its own styles while theme-mixin is used for setting global property values for styles across themes. As is evident in the file we have provided 6 different themes with a base color: default (orange), green, blue, purple, brown and cyan. You can remove themes that you don't like and add the ones you do. You can even modify the themes to whatever color combination you desire.

© 2014 SketchPixy Creative - v3.0.0