Getting Started with Angular

Using an Existing Project

If you have an existing Angular project and would like to integrate Brightlayer UI, you can follow the manual integration instructions below.

Manual Brightlayer UI Integration

Before taking the following steps, you'll want to make sure that you have an existing application.

Including the Component Libraries

Angular Material

Angular projects should make use of the Angular Material component library. Our Brightlayer UI themes are designed to work with these components.

```sh
npm install --save @angular/material // or yarn add @angular/material
```

For detailed documentation about the various components available in Angular Material, as well as sample code and API reference, refer to the Angular Material Documentation.

Brightlayer UI Components

Additionally, we have a supplemental library of components designed specifically for Brightlayer UI applications. If you wish to use these components, you can install them by running:

```sh
npm install --save @brightlayer-ui/angular-components // or yarn add @brightlayer-ui/angular-components
```

Adding Brightlayer UI Themes

By default, your application will be themed with the Google Material theme. To use the Brightlayer UI theme, first install it:

```sh
npm install --save @brightlayer-ui/angular-themes // or yarn add @brightlayer-ui/angular-themes
```

Installing the theme package will automatically include the Open Sans web font package as well as the Brightlayer UI colors package. In order to use the Brightlayer UI themes and font (Open Sans), you will need to modify your angular.json file "styles" entries (there is one under "build" and one under "test") to include the Brightlayer UI themes and Open Sans reference:

```json
"styles": [ "src/styles.scss", "./node_modules/@brightlayer-ui/angular-themes/theme.scss", "./node_modules/@brightlayer-ui/angular-themes/open-sans.scss", ],
```

Applying the Theme

You can apply the theme to your application by adding the proper class to your application's body element:

```html
// Default Theme

// Dark Theme
```

> If you do not specify a theme class, your application will use the default Material

> *theme.*

You can access the fonts using mat-typography, e.g.:

```html

Bold body copy
```

You can access theme colors as well, e.g.:

```html My App ```

More details about applying themes in your application can be found in [Theming your Angular Material app](#).

Additional Information

Code Formatting / Style

Brightlayer UI provides packages for code standards and style using ESLint and Prettier. These packages are used internally by the Brightlayer UI team, but they can also be used in your projects. You can find instructions for adding these packages to your project on [GitHub](#).

Browser Support

Brightlayer UI supports the same browsers as Angular and Angular Material. For specifics, please check out their support pages:

- [Angular CLI](#)
- [Angular Material](#)

> *With the end-of-life of Internet Explorer 11, Brightlayer UI will no longer be supporting IE11 as of 2021.*

License Information

Angular and Angular Material are both available under the MIT License.