

# Publication

---



**Cory House**

@housecor | reactjsconsulting.com



# The Plan



**Publish documentation**

**Prepare package.json and Readme**

**Automated build and deployment**

**Inspiration**

**Challenge**



Demo



Automated doc deploy to GitHub Pages



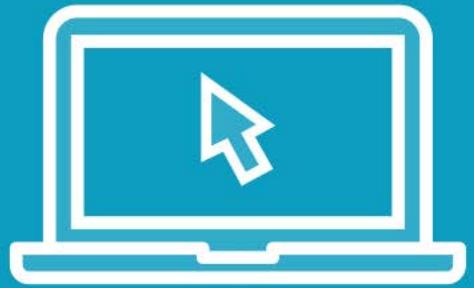
# What Belongs in the Readme?



**Project summary**  
**Install instructions**  
**Link to docs**



Demo



**GitHub readme**



# Initial npm Configuration

---





find packages



sign up or log in



# Build amazing things

npm is the package manager for JavaScript and the world's largest software registry. Discover packages of reusable code — and assemble them in powerful new ways.

[Sign up for npm](#)

## npm Orgs is powerful collaboration

- Encourage code discovery and re-use within teams
- Publish and control access to your own namespace
- Manage public and private code with the same workflow

[Sign up for Orgs](#)or, [Learn more about Orgs](#)

# Set Up npm

```
npm set init.author.name "Your Name"
```

```
npm set init.author.email "you@example.com"
```

```
npm set init.author.url "http://yourblog.com"
```

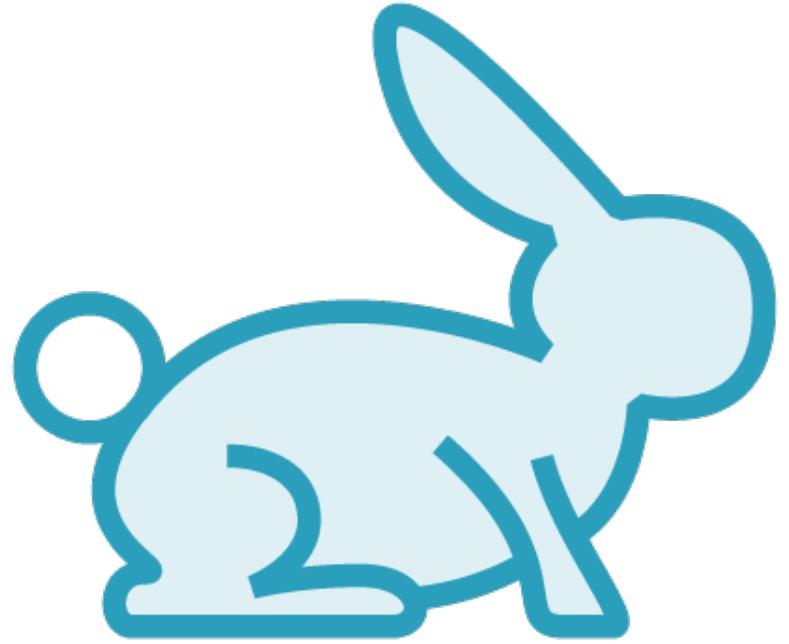
```
npm adduser
```



# npm Scripts for Production Build

---





## Build React in production mode

- Better performance
- Reduced bundle size



# Build React in Production Mode

```
plugins: [
  new webpack.DefinePlugin({
    "process.env.NODE_ENV": JSON.stringify("production")
  })

//...
```



Demo



**Set up automated build npm scripts**



# Preparing package.json for Publish

---



```
{  
  "name": "reusable-react-components",  
  "version": "1.0.0",  
  "description": "Demo for Reusable React Components",  
  "main": "index.js",  
  "scripts": {  
    "test": "echo \\\"Error: no test specified\\\" && exit 1",  
    "storybook": "start-storybook -p 9001 -c .storybook"  
  },  
  "keywords": [  
    "React"  
  ],  
  "author": "Cory House",  
  "license": "MIT",  
  "devDependencies": {  
    "@kadira/storybook": "^2.35.1"  
  },  
  "dependencies": {  
    "react": "^15.4.1",  
    "react-dom": "^15.4.1"  
  }  
}
```

# package.json



The way you declare dependencies in package.json impacts *your users*.



# Dependency Types

	Meaning	User experience
<b>dependencies</b> npm install -S	Always required	Always installed
<b>devDependencies</b> npm install -D	Required for development	Not installed
<b>peerDependencies</b> Enter manually	Expect consumer to install	User needs to install
<b>optionalDependencies</b> npm install -O	Install if available	Is installed if available



Place dev dependencies in...  
devDependencies



```
devDependencies: {  
},  
  
dependencies: {  
}
```

**npm install -D**

**npm install -S**

Only the packages listed  
here are installed when  
your package consumers  
type `npm install`.



Demo



**Prepare package.json for publish**



# Publish npm Package

---



# Pre-Publish Steps

- **Assure linting passes**
- **Assure tests pass**
- **Remove and recreate the build/lib/dist folder**

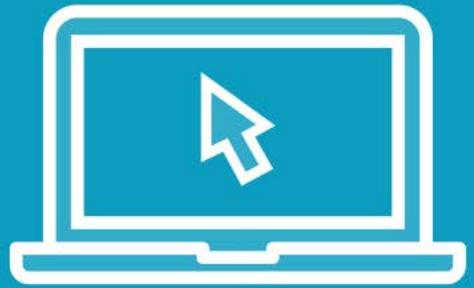




**Prepublish runs on npm install too**



Demo



Consume npm package



Demo



Compare import approaches



# Decision:

Pick a direct import approach



# Import Approaches

## 1. Named import

```
import {Label} from 'ps-react';
```

## 2. Import from /lib

```
import Label from 'ps-react/lib/Label';
```

## 3. Import from package root

```
import Label from 'ps-react/Label';
```



Currently supported

Uh oh! 😥

Our docs show this...



# Import Approaches

## 1. Named import

```
import {Label} from 'ps-react';
```

## 2. Import from /lib

```
import Label from 'ps-react/lib/Label';
```

## 3. Import from package root

```
import Label from 'ps-react/Label';
```

Pick one.



## 2. Import from /lib

```
import Label from 'ps-react/lib/Label';
```

- ✓ Imports single component
- ✓ No index.js at root needed
- ✓ Most common
- ✓ Simple build setup
- ✗ More typing

## 3. Import from package root

```
import Label from 'ps-react/Label';
```

- ✓ Imports single component
- ✓ Concise - No /lib in path
- ✓ No index.js at root needed
- ✗ Custom build
  - Generate package.json
  - Copy assets
  - No .npmignore/files array
  - Run "npm publish" from /lib



# Step 1: Add /lib to Webpack Alias



The image shows a screenshot of the Visual Studio Code interface. On the left, the Explorer sidebar lists several files and folders: build, config (which is expanded), jest, componentData.js, env.js, paths.js, polyfills.js, webpack.config.dev.js (which is selected and highlighted in blue), and webpack.config.prod.js. On the right, the main editor area shows the content of the webpack.config.dev.js file. The file contains configuration for Webpack, including reasonable defaults for file extensions and an alias section. The alias section currently includes 'react-native' and 'ps-react'. A red arrow originates from the text 'Change to ps-react/lib' located below the code and points to the 'ps-react' entry in the alias block, which is highlighted with a yellow box.

```
12 // @WebpackPath('node_modules'),
13 // These are the reasonable defaults supported by the Node ecosystem.
14 // We also include JSX as a common component filename extension, though
15 // some tools, although we do not recommend using it, see:
16 // https://github.com/facebookincubator/create-react-app/issues/253#issuecomment-265890594
17 extensions: ['.js', '.json', '.jsx', ''],
18 alias: {
19     // Support React Native Web
20     // https://www.smashingmagazine.com/2016/08/a-glimpse-into-the-future-of-react-native-web/
21     'react-native': 'react-native-web',
22     'ps-react': path.resolve(__dirname, '../src/components')
23 }
```

Change to ps-react/lib



# Step 2: Update Imports in Examples

The image shows a screenshot of the Visual Studio Code interface. On the left, the Explorer sidebar displays a project structure with 'components', 'docs', 'examples', and 'Eyelcon' folders. Below these, 'Example.js' is selected and highlighted with a blue bar. In the main editor area, the file 'Example.js' contains the following code:

```
1 import React from 'react';
2 import EyeIcon from 'ps-react/EyeIcon';
3
4 export default function EyeIconExample() {
5   return <EyeIcon />;
6 }
```

A red arrow originates from the text "Change to ps-react/lib/Eyelcon" located at the bottom right and points to the import statement "import EyeIcon from 'ps-react/EyeIcon';". The word "EyeIcon" is also highlighted in red.

Change to `ps-react/lib/Eyelcon`





Warning: Minor optimization ahead

# Import Approaches

## 1. Named import

```
import {Label} from 'ps-react';
```

## 2. Import from /lib

```
import Label from 'ps-react/lib/Label';
```

## 3. Import from package root

```
import Label from 'ps-react/Label';
```

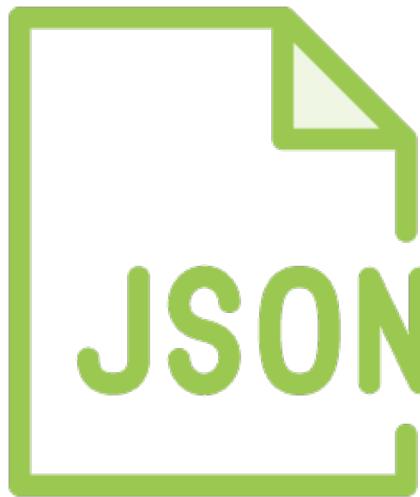
So, I have to  
type /lib on  
every import...



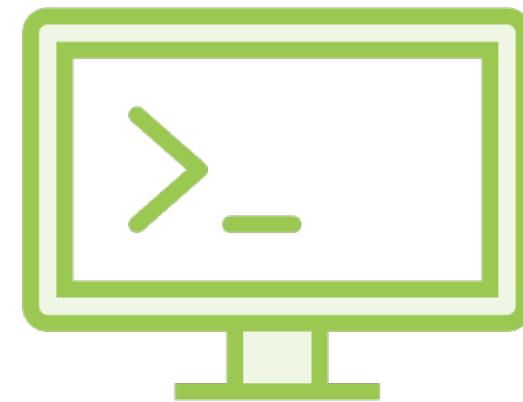
# Here's the Plan



**Step 1**  
Copy files



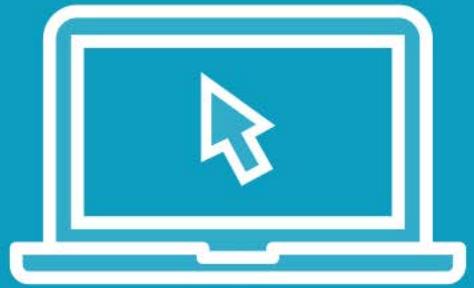
**Step 2**  
Generate package.json



**Step 3**  
Run npm publish in /lib



Demo



Add short import support



# Semantic Versioning

---



1.0.0

~~Major Minor Patch~~

Breaking Feature Fix





- 
- 0.1.0 Initial beta release**
  - 1.0.0 Initial public release**
  - 1.0.1 Bug Fix**
  - 1.1.1 Add a new feature**
  - 2.0.0 Breaking change**





1. Make code change
2. Commit
3. **npm version major|minor|patch**



Demo



**Demo – Semantic versioning with npm**



material-ui/CONTRIBUTING.md x Icon Library Search | Pluralsight x Cory

GitHub, Inc. [US] | https://github.com/callemall/material-ui/blob/master/CONTRIBUTING.md

# Contributing

If you're reading this, you're awesome! Thank you for helping us make this project great and being a part of the Material-UI community. Here are a few guidelines that will help you along the way.

## Asking Questions

For how-to questions and other non-issues, please use [StackOverflow](#) or [Gitter](#) chat instead. You can add a StackOverflow tag called "material-ui" that you can use to tag your questions.

## Opening an Issue

If you think you have found a bug, or have a new feature idea, please start by making sure it hasn't been reported or fixed. You can search through existing issues and PRs to see if someone has reported or fixed it.

Next, create a new issue that briefly explains the problem, and provides a bit of background information about what triggered it, and steps to reproduce it.

For code issues please include:

- Material-UI version

## CONTRIBUTING.md

- Development Guide
- Coding standards
- Issues
- Pull Requests
- How to run the build
- Branching structure
- How to generate docs

material-ui/CONTRIBUTING.md

GitHub, Inc. [US] | https://github.com/callemall/material-ui/blob/master/CONTRIBUTING.md

This repository Search Pull requests Issues Gist

callemall / material-ui Watch 975 Star 25,400 Fork 4,549

Code Issues 718 Pull requests 12 Projects 1 Wiki Pulse Graphs

Branch: master material-ui / CONTRIBUTING.md Find file Copy path

mbrookes [CONTRIBUTING] Update with git clone --depth 1 + other tweaks (#6627) 30ce9fc 21 days ago

4 contributors

102 lines (62 sloc) 5.33 KB Raw Blame History

# Contributing

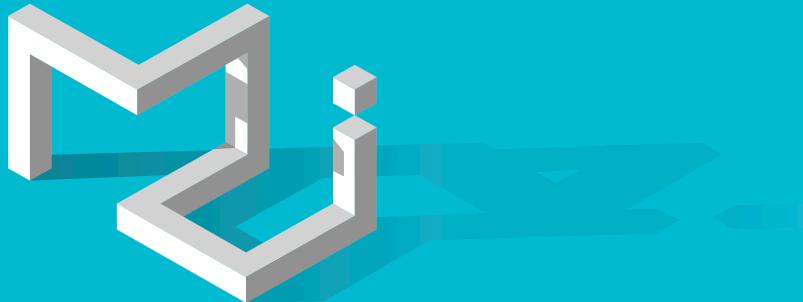
If you're reading this, you're awesome! Thank you for helping us make this project great and being a part of the Material-UI community. Here are a few guidelines that will help you along the way.

## Asking Questions

# Inspiration

---

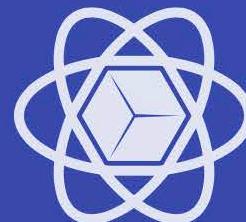




# Material-UI

A Set of React Components that Implement  
Google's Material Design

DEMO

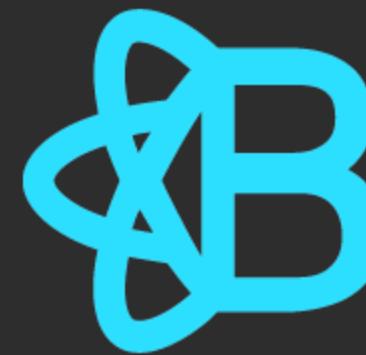


# React Toolbox

Bootstrap your application with beautiful Material Design Components

[Installation](#) • [Components](#) • [Github](#)

React Toolbox is a set of React components that implements Google Material Design specification. It's built on top of some the trendiest proposals like CSS Modules (written in SASS), Webpack and ES6. The library harmoniously integrates with your Webpack workflow



The most popular front-end framework, rebuilt for React.

- 📢 We are actively working to reach a 1.0.0 release, and we would love your help to get there.
- ☑ Check out the [1.0.0 roadmap](#) and [contributing guidelines](#) to see where you can help out.
- 👉 A great place to start is any [issue](#) with a [help-wanted](#) label.
- ✓ We are open to pull requests that address bugs, improve documentation, enhance accessibility, add test coverage, or bring us closer to feature parity with [Bootstrap](#).
- 👤 We actively seek to invite frequent pull request authors to join the organization. ↗

⚠️ The project is under active development, and APIs will change.

# React + Foundation

FOUNDATION AS REACT COMPONENTS

Get started

## Button

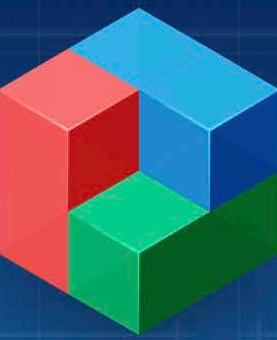
### Basics

```
<div className="button-basics-example">
  <Link>Learn More</Link>
  <Link>View All Features</Link>
  <Button color={Colors.SUCCESS}>Save</Button>
  <Button color={Colors.ALERT}>Delete</Button>
</div>
```

Learn More   View All Features   Save   Delete

### Sizing

```
<div className="button-sizes-example">
  <Link size={Sizes.TINY}>So Tiny</Link>
  <Link size={Sizes.SMALL}>So Small</Link>
  <Link>So Basic</Link>
  <Link size={Sizes.LARGE}>So Large</Link>
  <Link isExpanded>Such Expand</Link>
```



# Blueprint

A React UI toolkit for the web

[DOCS](#) • [GITHUB](#)

Input

Default

Primary

Success

Warning

Danger

[blueprintjs.com](http://blueprintjs.com)



ANT DESIGN

Search Components...

Home

## Specification

## Components

## Pattern

Practice

Resource

中文

# ANT DESIGN

## A UI Design Language

## Introduce

## Get Started



9983

ant.design

Fabric

Get started

Styles

Components

Resources

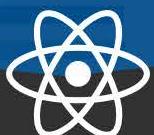
Blog

# Office UI Fabric

The official front-end framework for building experiences  
that fit seamlessly into Office and Office 365.

[Get started](#)

Fabric Core >=5.1.0-0 <6.0.0-0 and Fabric React >=2.23.0 <3.0.0



Built with React

Fabric's robust, up-to-date

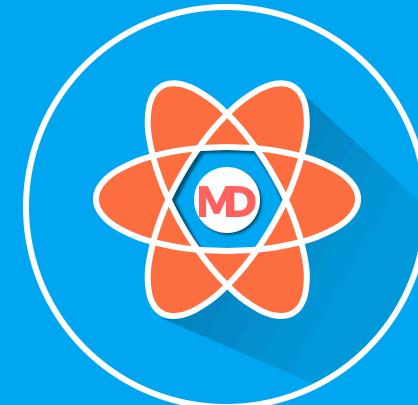
[dev.office.com/fabric](https://dev.office.com/fabric)

[Home](#)[Getting Started](#)[Customization](#)[Discover More](#)[Components](#)

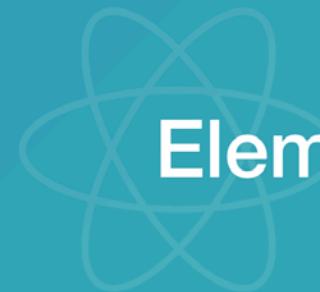
## References

[React](#)[Material Design](#)[Material Icons](#)[Contrast Checker](#)

# react-md

[VIEW DEMO](#)

This project's goal is to be able to create a fully accessible material design styled website using React Components and Sass. With the separation of styles in Sass instead of inline styles, it should hopefully be easy to create custom components with the existing styles.



# Elemental UI

A UI Toolkit for React.js Websites and Apps

[Follow @ElementalUI on Twitter](#)[View the GitHub Project](#)

## Project Status

- Currently under development, initially for use in [KeystoneJS](#)
- We are experimenting with Component APIs
- Potentially, we'll transition from stylesheets to more inline styles, and would love feedback

---

### DEMOS

---

[CSS](#)[Buttons](#)[Forms](#)[Spinner](#)[Modal](#)[Misc](#)



# Rebass

v0.3.4

57 Configurable React Stateless Functional UI Components

[build](#) passing [Tweet](#) [Star](#) 2,047

[View Demo](#)



Slack - it's teamwork, but  
simpler, more pleasant and  
more productive.  
ads via Carbon



# Ship Better UI, Faster

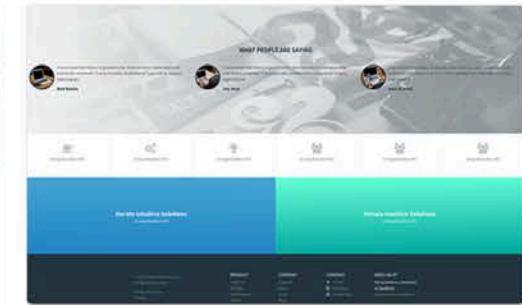
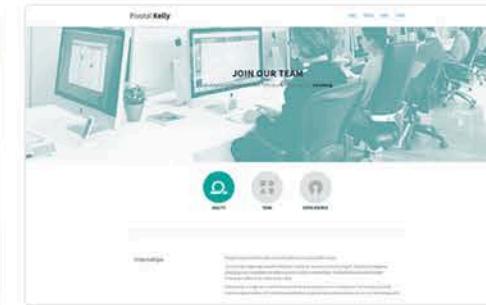
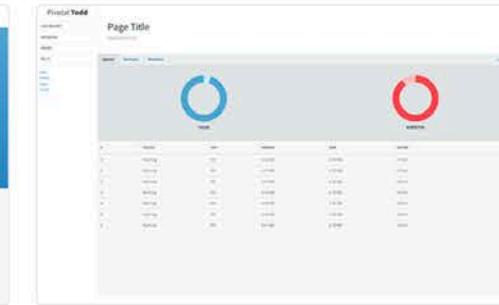
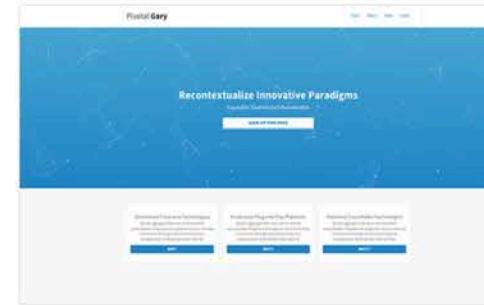
Pivotal UI is a collection of React and CSS components styled for the Pivotal brand.

[GET STARTED](#)

[React](#)    [CSS](#)

Alerts  
Autocomplete  
Back To Top  
Buttons  
Collapse  
Copy To Clipboard  
Dividers  
Dropdowns  
Ellipsis  
Expander  
Forms  
Grids  
Hoverable  
Iconography  
Images  
Labels  
Lists  
Media  
Modals  
Notifications  
Pagination  
Panels  
Panes  
Portals  
Print  
Responsive Utilities

## Examples Using Pivotal UI



This documentation generated using [Hologram](#)

[cf-builder-card](#)[cf-builder-pagination](#)[cf-builder-table](#)[cf-component-callout](#)[cf-component-card](#)[cf-component-checkbox](#)[cf-component-code](#)[cf-component-copyable-textarea](#)[cf-component-dropdown](#)[cf-component-dynamic-content](#)[cf-component-flex](#)[cf-component-form](#)[cf-component-heading](#)[cf-component-icon](#)[cf-component-input](#)

## cf-builder-card

### This is a Card

This is the description of a card.

[Click me!](#)[One](#)[Two](#)[Three](#)[Four](#)[API](#) [Help](#)

#### EXAMPLE CODE

```
const React = require('react');
const { render } = require('react-dom');
const { createStore, combineReducers } = require('redux');
const { Provider } = require('react-redux');
const {
  CardBuilder,
  cardReducer
}
```

**Getting Started****Introduction**

Usage



GitHub



CHANGELOG

Start typing...

**Elements**

Button

Container

Divider

Flag

Header

Icon

Image

Input

Label

List

Loader

Rail

Reveal

Segment

Step

**Collections**

Breadcrumb

Form

Grid

Menu

Message

Table

**Views**

Card

Comment

Feed

"



## Semantic-UI-React

The official Semantic-UI-React integration.

### Introduction

Semantic-UI-React is the official React integration for [Semantic UI](#).

- ✓ jQuery Free
- ✓ Declarative API
- ✓ Augmentation
- ✓ Shorthand Props
- ✓ Sub Components
- ✓ Auto Controlled State

Installation instructions are provided in the [Usage](#) section.

### jQuery Free

jQuery is a DOM manipulation library. It reads from and writes to the DOM. React uses a virtual DOM (a JavaScript representation of the real DOM). React only writes patch updates to the DOM, but *never reads* from it.

It is not feasible to keep real DOM manipulations in sync with React's virtual DOM. Because of this, all jQuery functionality has been re-implemented in React.

### Declarative API

Declarative APIs provide for robust features and prop validation.



# grommet

focus on the essential experience





## Components

## ActionBar

ActionBar

 [Share](#)  [Edit](#)  [Pin](#)  [Download](#)

BarChart

Button

## ActionButton

Card

Checkbox

Column

ContentEditable

DatePicker

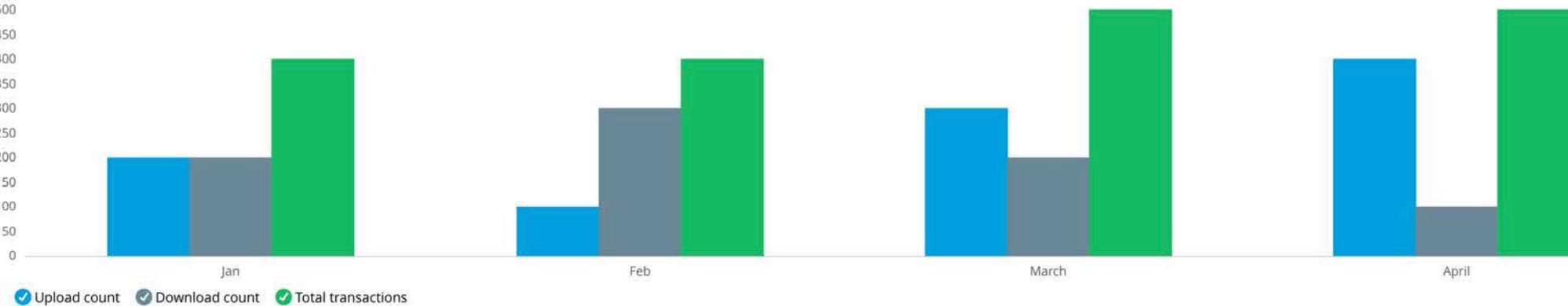
## BarChart

Detail

## BarChart

DonutChart

DropdownMenu



DropdownToggle

EmptyResult

FileUploader

Filter

FilterBar

Flex

Form

GridLayout

Heading

Hero



# Belle

Configurable React Components with great UX

Star 1,785

## Getting Started

Components

Button

Card

ComboBox

DatePicker

Select

Option

Placeholder

Separator

Rating

Spinner

TextInput

Toggle

Choice

Guides

Introducing Belle

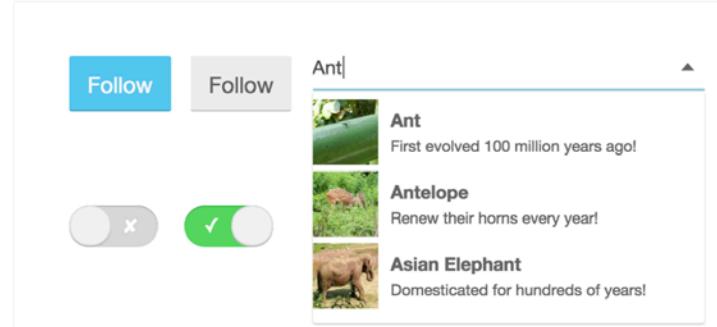
Form Components

Belle provides you with a set of React components like Toggle, ComboBox, Rating, TextInput, Button, Card, Select and soon many more.

All of the components are optimized to work both on **mobile & desktop devices**.

The styles are highly customizable on two levels. You can **configure the base styles** of all the components as well as **modify each one of them individually**.

## Overview



Once upon a time there was a  
Princess who lived in a kingdom  
far, far away ...



[nikgraf.github.io/belle](http://nikgraf.github.io/belle)

[Getting Started](#)[Example Layouts](#)[CSS/JS](#)[React](#)[Angular](#)[HTML Email](#)[WebComponents](#)[Design Files](#)

MUI is a lightweight CSS framework that follows  
Google's Material Design guidelines

[DOWNLOAD v0.9.9](#)[VIEW ON GITHUB](#)[New - Angular Support!](#)

## Quickstart

You can copy-paste this boilerplate HTML to get started using MUI:

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <!-- load MUI -->
    <link href="//cdn.muicss.com/mui-0.9.9/css/mui.min.css" rel="stylesheet" type="text/css" />
    <script src="//cdn.muicss.com/mui-0.9.9/js/mui.min.js"></script>
  </head>
  <body>
    <!-- example content -->
    <div class="mui-container">
      <div class="mui-panel">
        <h1>My Title</h1>
        <button class="mui-btn mui-btn--primary mui-btn--raised">My Button</button>
      </div>
    </div>
  </body>
</html>
```



## Getting Started

[Installation](#)

### macOS

[Box](#)

[Button](#)

[Checkbox](#)

[Dialog](#)

[Label](#)

[Link](#)

[List View](#)

[Pin](#)

[Progress Circle](#)

[Radio](#)

[Search Field](#)

[Segmented Control](#)

[Text](#)

[Text Input](#)

[Title Bar](#)

[Toolbar](#)

[Toolbar Nav](#)

[View](#)

[Window](#)

### Windows 10

[Button](#)

[Checkbox](#)

[Label](#)

[Master/Details View](#)

[Nav Pane](#)

[Progress Circle](#)

[Radio](#)

[Text](#)

[Text Input](#)

[Title Bar](#)

[View](#)

[Window](#)

## Documentation

[EDIT PAGE](#) [SUBMIT BUG](#)

**react-desktop** is a JavaScript library built on top of [Facebook's React](#) library, which aims to bring a native desktop experience to the web, featuring many macOS Sierra and Windows 10 components. **react-desktop** works perfectly with [NW.js](#) and [Electron.js](#), but can be used in any JavaScript powered project!

### Table of Contents

#### 1. Getting Started

- [Installation](#)

#### 2. macOS

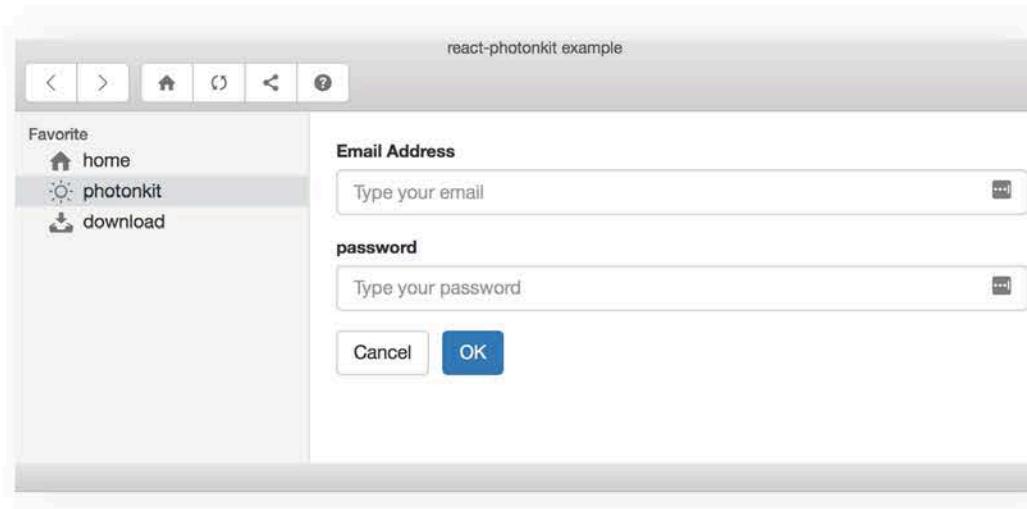
- [Box](#)
- [Button](#)
- [Checkbox](#)
- [Dialog](#)
- [Label](#)
- [Link](#)
- [List View](#)
- [Pin](#)
- [Progress Circle](#)
- [Radio](#)
- [Search Field](#)
- [Segmented Control](#)
- [Text](#)
- [Text Input](#)
- [Title Bar](#)
- [Toolbar](#)
- [Toolbar Nav](#)
- [View](#)
- [Window](#)

#### 3. Windows 10



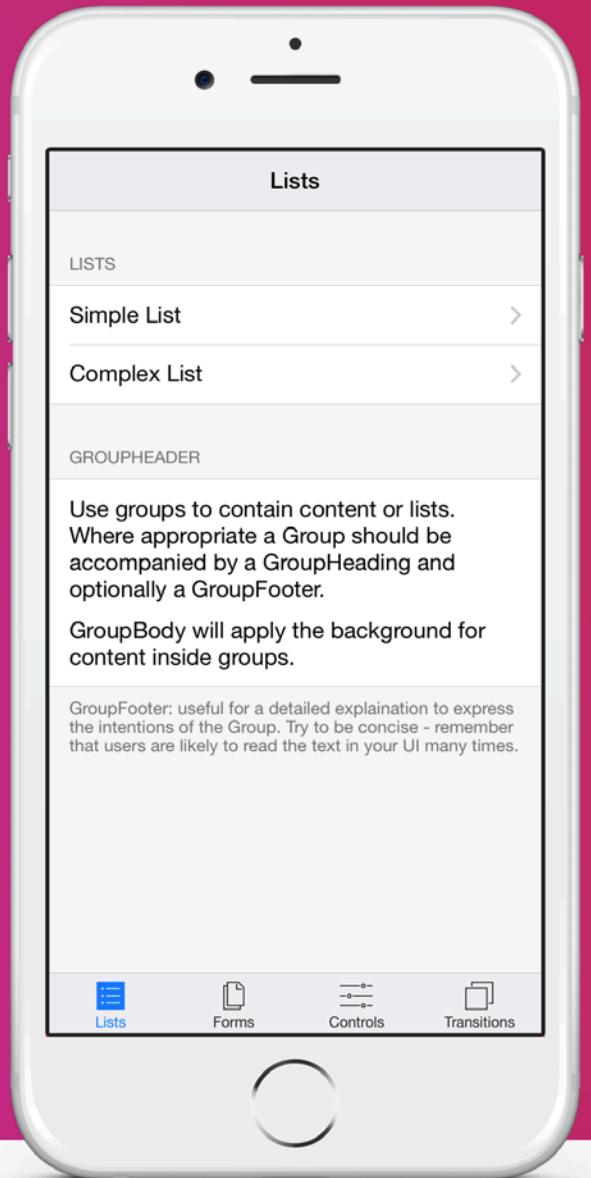
# react-photonkit

Photon components built with React

[Download](#)[See all Components](#)

Code licensed under the MIT License and the docs are licensed under CC BY 3.0

[react-photonkit.github.io](https://react-photonkit.github.io)



# TouchstoneJS

React.js powered UI framework for developing beautiful hybrid mobile apps.

Docs, examples and getting started guide coming soon.  
Get notified when we launch

Email address

Notify me

Follow @touchstonejs

Star on GitHub

Current version 0.4.0 · Free and open source (MIT)

[touchstonejs.io](http://touchstonejs.io)

# Onsen UI for React

Create beautiful high-quality hybrid mobile apps the fastest way  
with React and Onsen UI Framework.



## Get Started

with Onsen UI for React

Also available for...



Latest 2.1.0 released 23 days ago. See [all versions](#). Distributed free and open source (Apache 2.0) via [GitHub](#)

 Follow @Onsen\_UI

 Star on GitHub

## Beautifully made with Android and iOS components

More than 100 components are specially made for Material and Flat design, with [automatic platform detection](#)



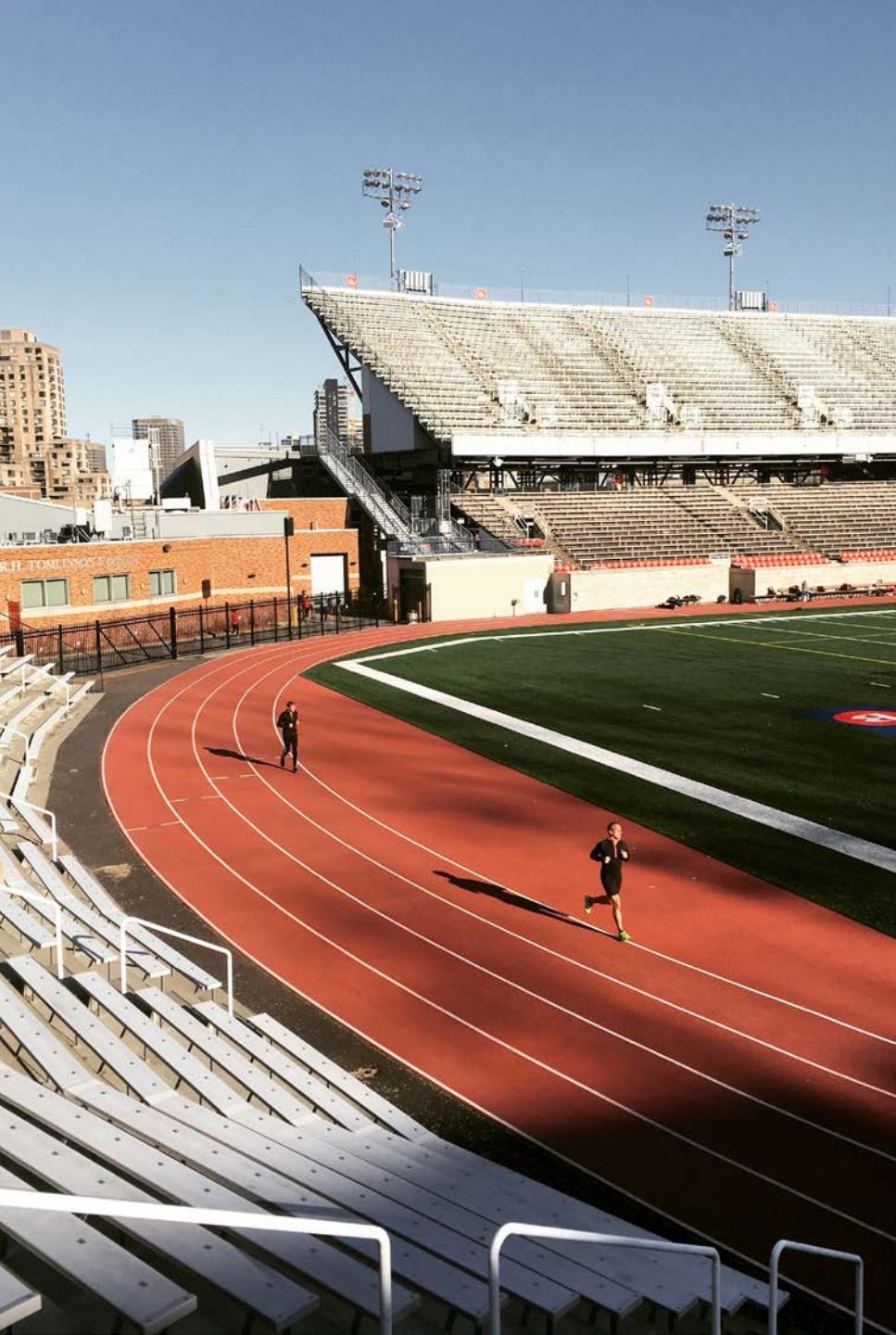
onsen.io/react

## Components

- [React Component List](#)
- [JS.coach - A catalog of React components](#)
- [iReact: Fantastic React Components and Libraries!](#)
- [Khan Academy React Components](#)
- [VistarMedia Components](#)
- [React Topcoat UI components](#)
- [react-material - Material design components written with React.js and React Style](#)
- [material-ui - A CSS Framework and a Set of React Components for Material Design](#)
- [react-google-login-component](#)
- [react-facebook-login-component](#)
- [react-component - A Collection of React UI Components](#)
- [Ant Design - An enterprise-class UI design language and React-based implementation](#)
- [UXCore - A React UI Component Set living for enterprise backend application](#)
- [react-desktop - OS X and Windows UI components built with React](#)
- [react-aim - Determine the cursor aim for triggering mouse events](#)
- [react-touch - React photo viewer for mobile](#)
- [griddle - A simple grid component for use with React](#)
- [react-grid - Data grid for React](#)
- [React-iCheck - iCheck components built with React. Highly customizable checkboxes and radio buttons](#)
- [react-absolute-grid - An absolutely positioned, animated, filterable, sortable, drag and droppable, ES6 grid for React](#)
- [react-async - Async rendering for React components](#)
- [react-async-input - Inputs that behave properly with async set\[State,Props\] calls](#)
- [react-countup - A React component wrapper around CountUp.js](#)
- [react-mediaswitch - Choose your DOM based on media queries](#)
- [react-frame-component - Render your React app to an iFrame](#)
- [react-gravatar - React component for rendering a gravatar profile image](#)
- [markdown-react - React Render for Standard Markdown](#)
- [react-component-gallery — React component for creating an evenly spaced components](#)
- [react-markdown-textarea - Component to create textareas for entering Markdown](#)
- [react-markdown-area - Github flavored markdown editor with live preview](#)
- [react-textarea-autosize - Textarea component for React which grows with content](#)
- [react-spinner - Zero configuration loading spinner](#)
- [react-treeview - Easy, light, flexible tree view made with React](#)

Opportunity:  
Use these as a foundation





**Publish a reusable component!**



ps-react/README.md at master

GitHub, Inc. [US] | <https://github.com/coryhouse/ps-react/blob/master/README.md>

# Pluralsight React Components

A library of React components created in "Creating Reusable React Components" on Pluralsight.

## Install

```
npm install ps-react
```

## Docs

[Component documentation](#)

## Component Hall of Fame

Here's a list of components built by your fellow viewers!

Submit a pull request to add your link here... :)



What do you want to learn?

Cory

housecor@gmail.com



...



# Creating Reusable React Components

by Cory House

Learn how to design, create, and publish reusable React components that you can share with your team, your company, or the world.

▶ Resume Course

Bookmark

Add to Channel

Live mentoring

Table of contents

Description

Transcript

Exercise files

Discussion

Recommended

This course is part of:  React Path

Chime in here!

Expand all

▶ Course Overview



1m 36s



Share course



Course author



Cory House

Cory is an independent consultant with over 15 years of experience in software development. He is a Microsoft MVP, ASP Insider, and a member of the Telerik developer experts program.



# EXPLORE REACT, TOGETHER.

TELL ME MORE

## Courses By Cory House

### **Building a JavaScript Development Environment**

Beginner    5h 19m    10 Nov 2016

### **Building Applications with React and Redux in ES6**

Intermediate    6h 13m    20 May 2016

### **Building Applications with React and Flux**

Intermediate    5h 8m    13 Aug 2015

### **HTML5 Web Component Fundamentals**

Beginner    5h 3m    9 Jan 2015

### **Becoming an Outlier: Reprogramming the Developer Mind**

Intermediate    2h 33m    24 Apr 2014

### **Architecting Applications for the Real World in .NET**

Intermediate    2h 52m    7 Jan 2014

### **Clean Code: Writing Code for Humans**

Intermediate    3h 10m    7 Oct 2013



# That's a Wrap!



**Published to npm**

**Generated rich documentation**

**Inspiration**

