reactjs + redux

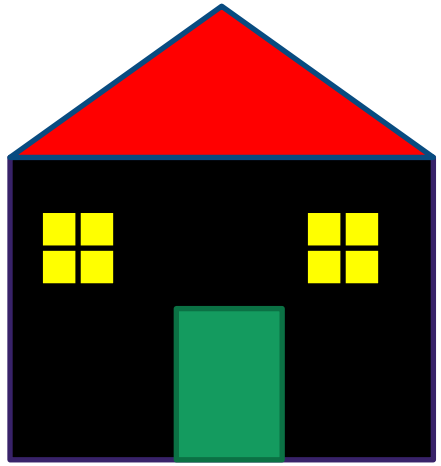
Arjun Raghavan Manathanath
Bangalore, 14-12-2018

MERCK

React JS

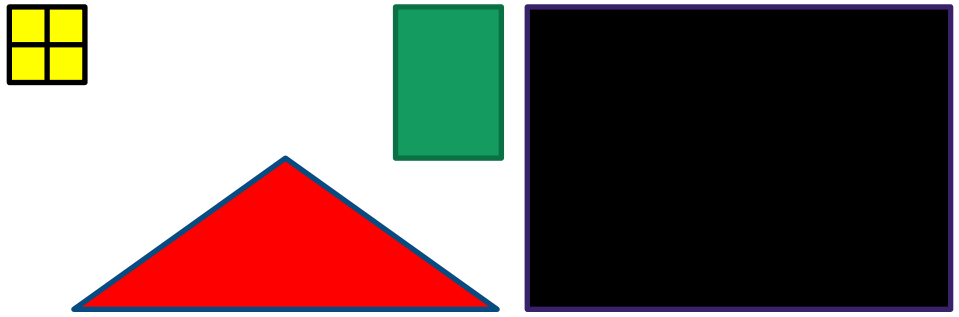
Introduction

- React is a JS library for building user interfaces.
- Renders UI and respond to events.
- Building block of React : React Elements
- Predictable and easier to debug



React Component

- Small
- Reusable
- Composable
- Unit Testable
- Return a React Element



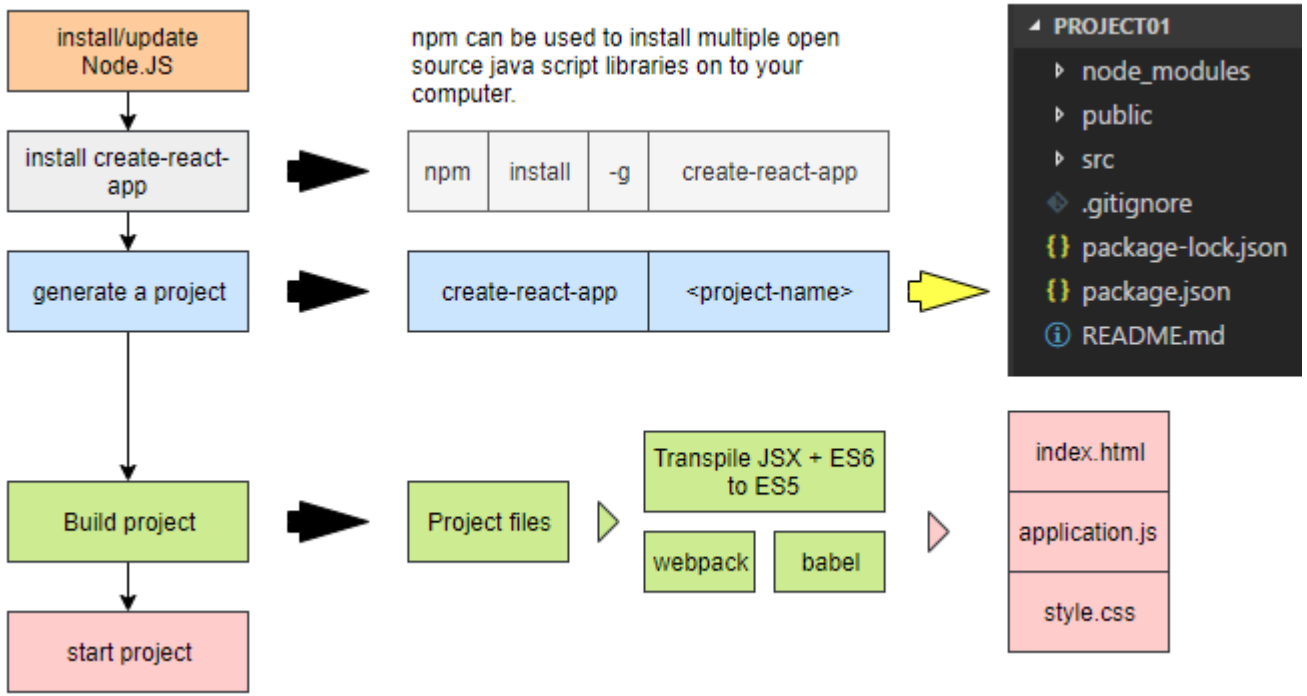
React Application

- Made up of multiple components

Hello React!

Create-React-App

- https://github.com/facebook/create-react-app



- npx if npm > 5.1

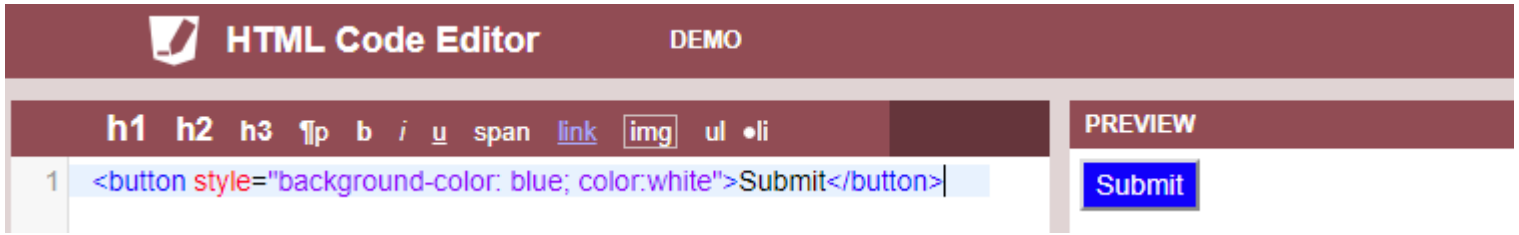
```
index.js
1  import React from 'react';
2
3  //To render a react component to a DOM
4  import ReactDOM from 'react-dom';
5
6  //App component
7  const App = function () {
8    return <div>Hello React!</div>;
9  }
10
11 ReactDOM.render(<App />, document.getElementById('root'));
```

Hello React!

- Import statements
- React & ReactDOM
- Dev Tools
- JSX

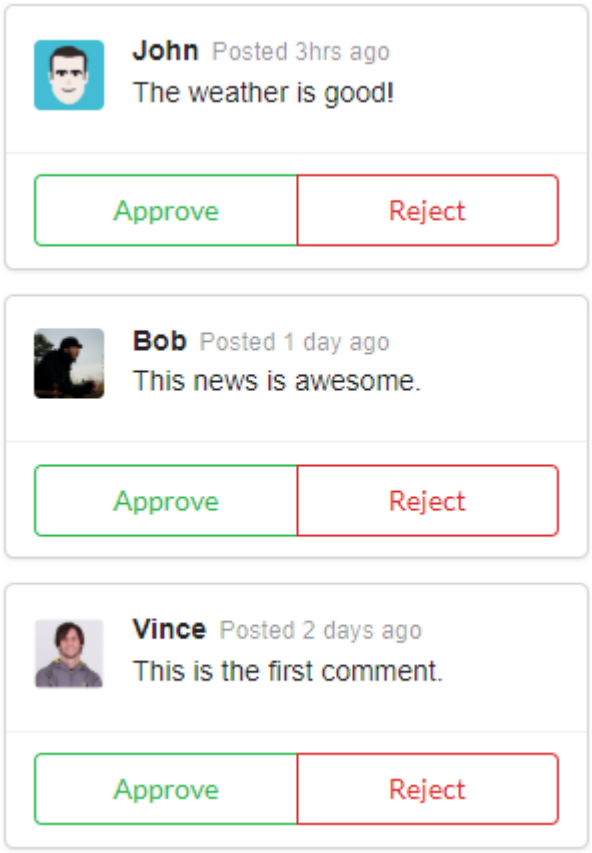
JSX

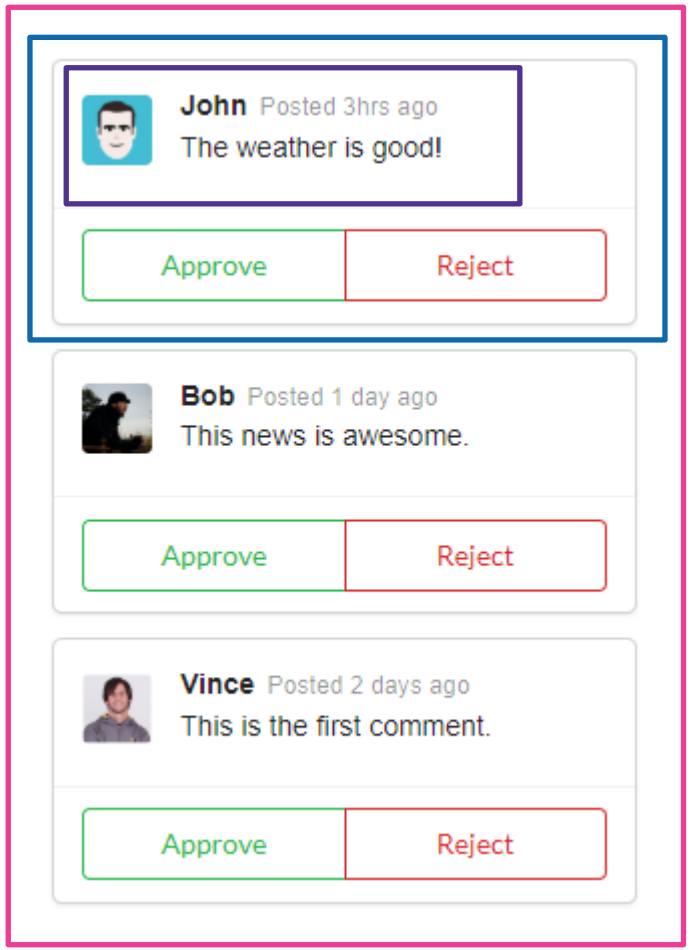
- HTML that gets produced and loaded on to the DOM
- Clean and concise
- Comes with full power of javascript
- Opening tag is on the same line as that of return statement or use ()
- JSX can reference javascript variables
- Uses camelCase property naming conventions
  - class > className
  - tabindex > tabIndex
- Optional




```
//App component
const App = function(){
  return   <button style={{backgroundColor: 'blue', color:'white'}}>Submit</button>;
}
```

Comments App





- Applying styles
- Passing Props
- Using libraries
- Passing Children
- Component Reuse
- Component Composition

React Basics

- Event Handling
- Class Component
- States
- State without constructor
- Lifecycle methods
- axios
- Keys
- Callbacks
- Understanding *this*
- Array.map()

App Examples

- Event Handling
- Country Flag
- Todo

Virtual DOM

On every update

- React builds a new virtual DOM subtree
- Diffs it with the old one
- Computes the minimal set of DOM mutations and puts them in a queue
- Batch executes all updates

Redux

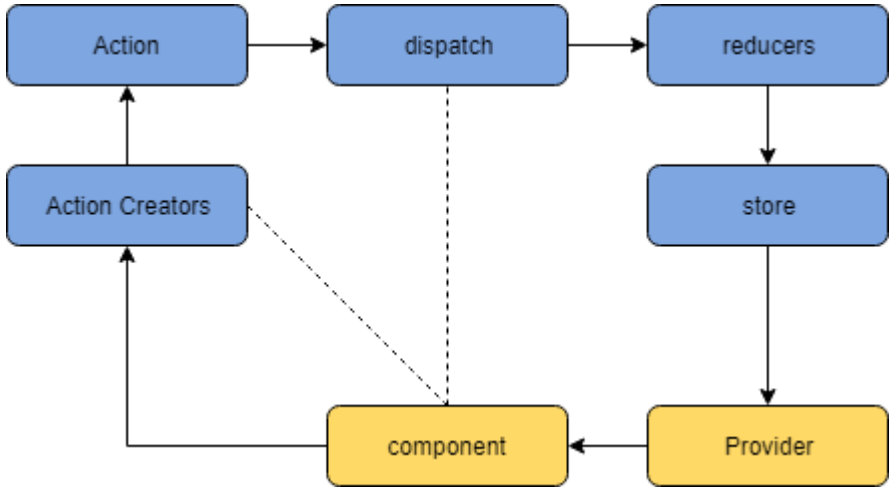
- Redux is a predictable state container for JavaScript apps.
- Can use Redux together with React, or with any other view library.
- Low memory footprint (2kB, including dependencies)
- Has a large ecosystem of addons available.

Three Principles

- Single source of truth
- State is read-only
- Changes are made with pure functions

Actions

- Actions are payloads of information that send data from your application to your store.
- They are the *only* source of information for the store.
- Actions are plain JavaScript objects.
- Actions must have a type property that indicates the type of action being performed.



Action Creators

- Action creators are functions that create actions.
- Action creators simply return an action

Dispatch

- to actually initiate a dispatch, pass the result to the dispatch() function
- The dispatch() function can be accessed directly from the store as store.dispatch(), or can be accessed it using a helper like react-redux's connect().
- bindActionCreators() automatically bind many action creators to a dispatch() function.

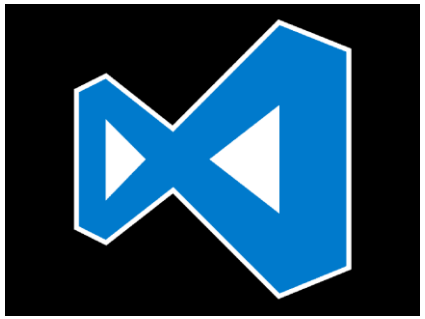
Reducers

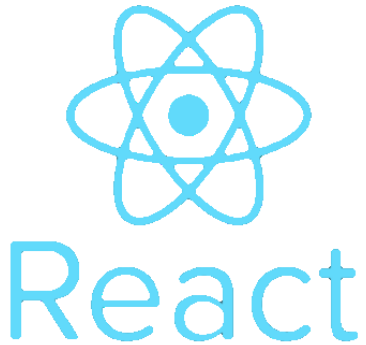
Reducers specify how the application's state changes in response to actions sent to the store

Q&A

MERCK

References





- https://reactjs.org/
- https://github.com/arjun-merck/reactjs
- https://www.udemy.com/
- https://www.w3schools.com/
- https://semantic-ui.com/
- https://babeljs.io/repl
- https://redux.js.org/
- https://codepen.io/
- https://restcountries.eu/

GitHub

MERCK

THANK YOU !!!

MERCK