



JAMAL MOHAMED COLLEGE

ALUMNI ASSOCIATION (Singapore Chapter)

Professional Development Seminar Series

STACK UP YOUR SKILLS

Introduction to Node.js and React/Vuejs

Free IT Workshop

10 February 2019

MR. M KALANDER MOHIDEEN

Assistant Director of Technology
Rajah & Tann Singapore LLP

MR. A BASHEER AHMED

Software Analyst
Manulife Singapore

www.jmcalumni.org.sg

Google

Visual Studio Code

https://code.visualstudio.com

Visual Studio Code Docs Updates Blog API Extensions FAQ

Search Docs

Download

Version 1.31 is now available! Read about the new features and fixes from January.

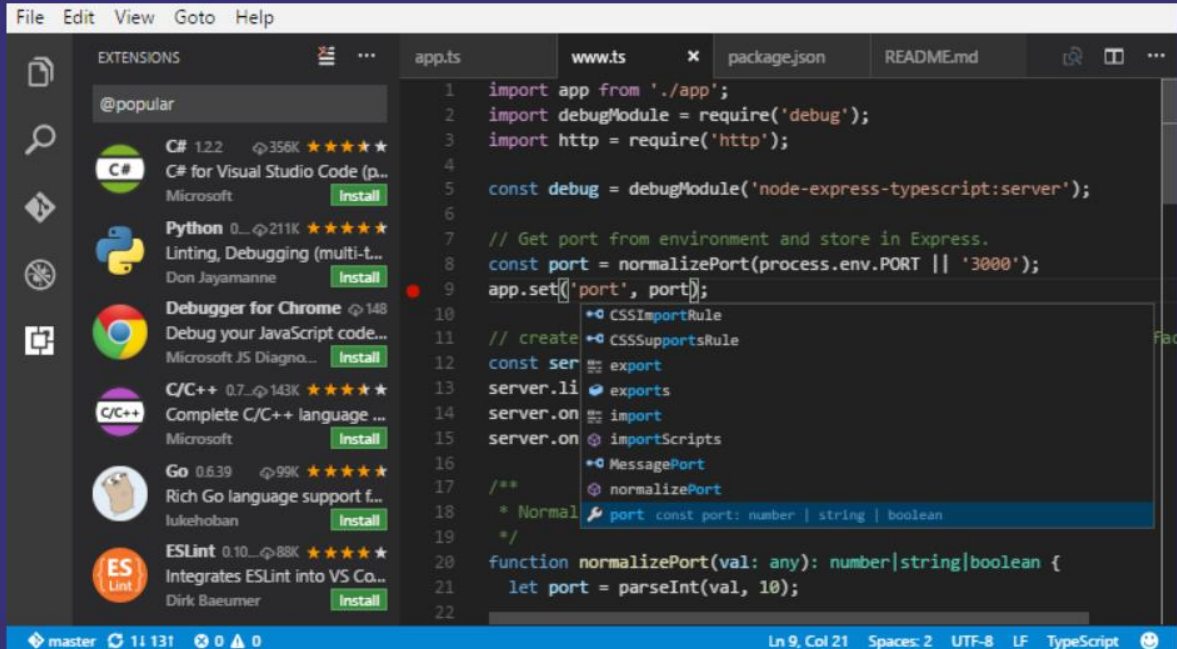
Code editing. Redefined.

Free. Open source. Runs everywhere.

Download for Windows
Stable Build

Other platforms and Insiders Edition

By using VS Code, you agree to its license and privacy statement.



Code



Atom



Sublime Text

Node.js

- Opensource cross platform runtime environment for server side networking application.
- Nodejs runs single-threaded, non-blocking, asynchronously programming , which is very memory efficient.
- Runs on chrome JavaScript runtime - v8 engine , Runs Javascript on our desktop
- Lightweight, Front End and Back End in the same language
- <https://nodejs.org/en/download/>
- Install , Docs, Node –v
- npm stands Node Package Manager (useful things created by others)
- www.azure.com/node

- Asynchronous tasks and callbacks
- Write your own module
- `npm install lodash`
- `npm install -g nodemon`

Modern Web Apps

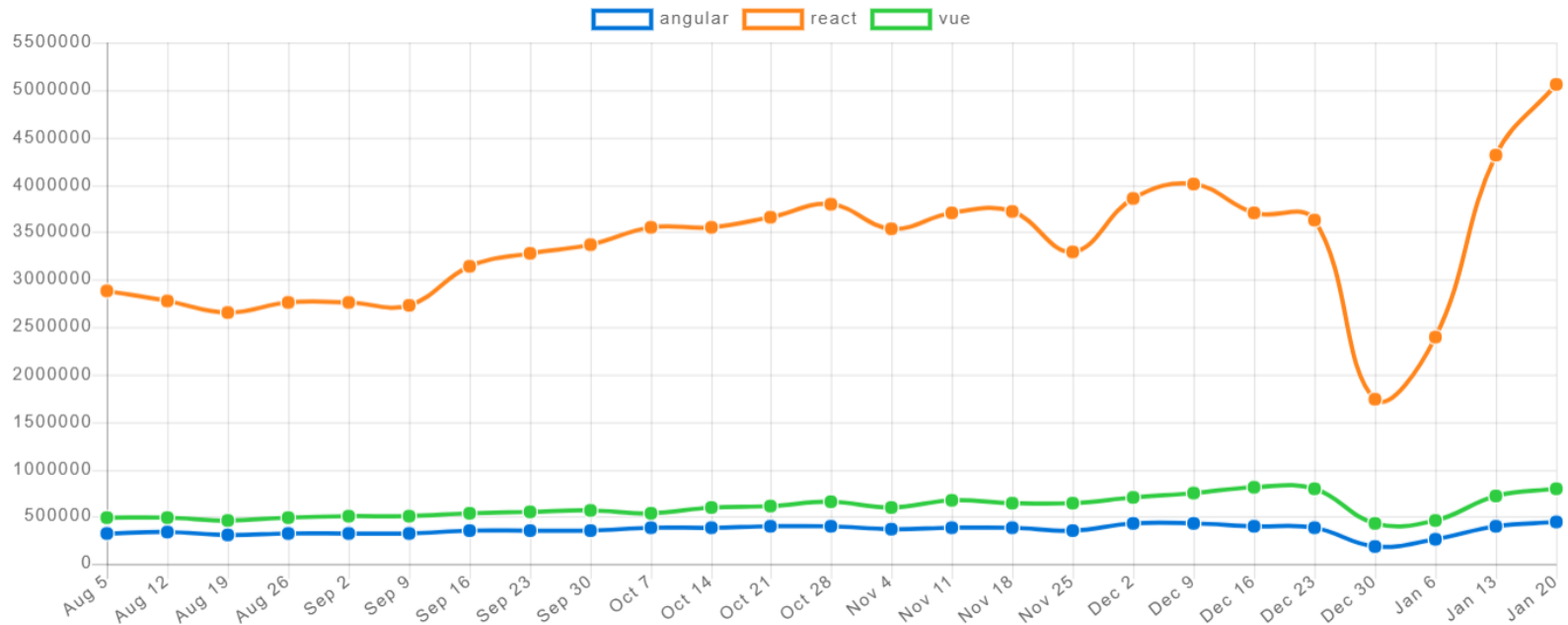
- Front End Components
- Components are typically written in Javascript
- Benefits – Syntax is the same, both front end and backend . Shared Code library.
- Works well with JSON (Javascript Object Notation)

REACT

- JavaScript library for building user interfaces
- It makes really easy to develop responsive web pages
- Developed by Facebook
- Usually called as Framework because of its behaviour /capabilities
- Rival is Angular. New framework getting popular is Vuejs



Downloads in past 6 Months ▾



React Web Development

Amazing examples

- Airbnb
- BBC
- Cloudflare
- Dropbox
- Facebook
- Flipboard
- Instagram
- Khan Academy
- Netflix
- Salesforce
- Uber
- Whatsapp
- Zendesk

[Home](#)[About](#)[Have an account? Log in](#)**Tweets**
15.1K**Following**
2,561**Followers**
8.49M**Likes**
1,449**Lists**
28**Moments**
15[Follow](#)**Microsoft** ✓

@Microsoft

We're on a mission to empower every person and every organization on the planet to achieve more. Support:

[@MicrosoftHelps](#)

Redmond, WA

[news.microsoft.com](#)

Joined September 2009

1,737 Photos and videos

**Tweets****Tweets & replies****Media**

Microsoft Retweeted

**Microsoft Research** @MSFTResearch · 17h

Dr. Cecily Morrison gives us an overview of what she calls the "pillars" of inclusive design and discusses the path to developing Project Torino, an inclusive physical programming language for children with visual impairments: [aka.ms/AA3v6k2](#) #MSRPOdcast

**Empowering people with AI with Dr. Cecily Morrison**[www.microsoft.com](#)

9

49

132

Microsoft Retweeted

harrychum ✓ @harrychum · Jan 22**New to Twitter?**

Sign up now to get your own personalized timeline!

[Sign up](#)**You may also like** · Refresh**Windows** ✓

@Windows

**Google** ✓

@Google

**Intel** ✓

@intel

**Bill Gates** ✓

@BillGates

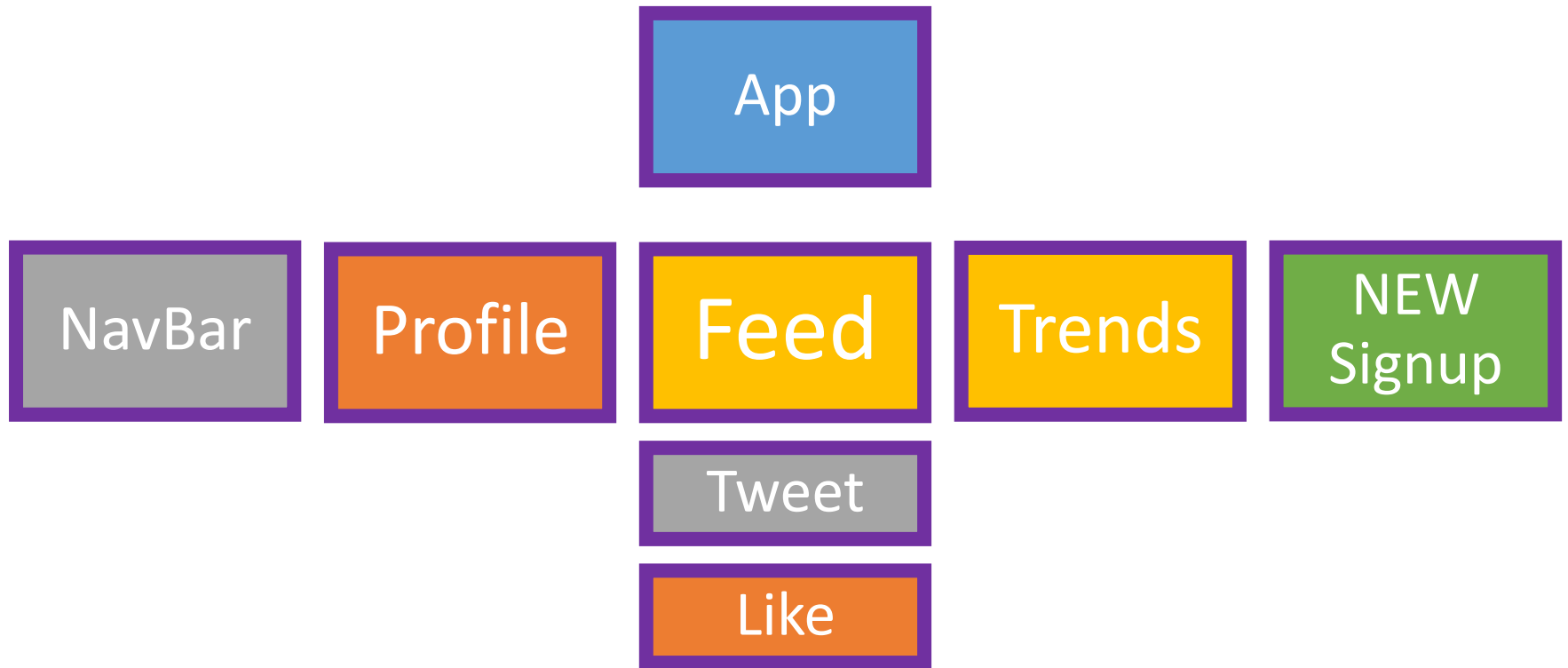
**Twitter** ✓

@Twitter

Worldwide trends

#YerliMobilUygulamaELLAM

Twitter Page - Components



```
class Tweet {
```

Tweet

```
state = { };
```

```
render () {  
}
```

```
}
```

React

Virtual DOM

React Element

App

NavBar

Profile

Feed

Trends

NEW
Signup

Tweet

Like

Real DOM

DOM Element

App

NavBar

Profile

Feed

Trends

NEW
Signup

Tweet

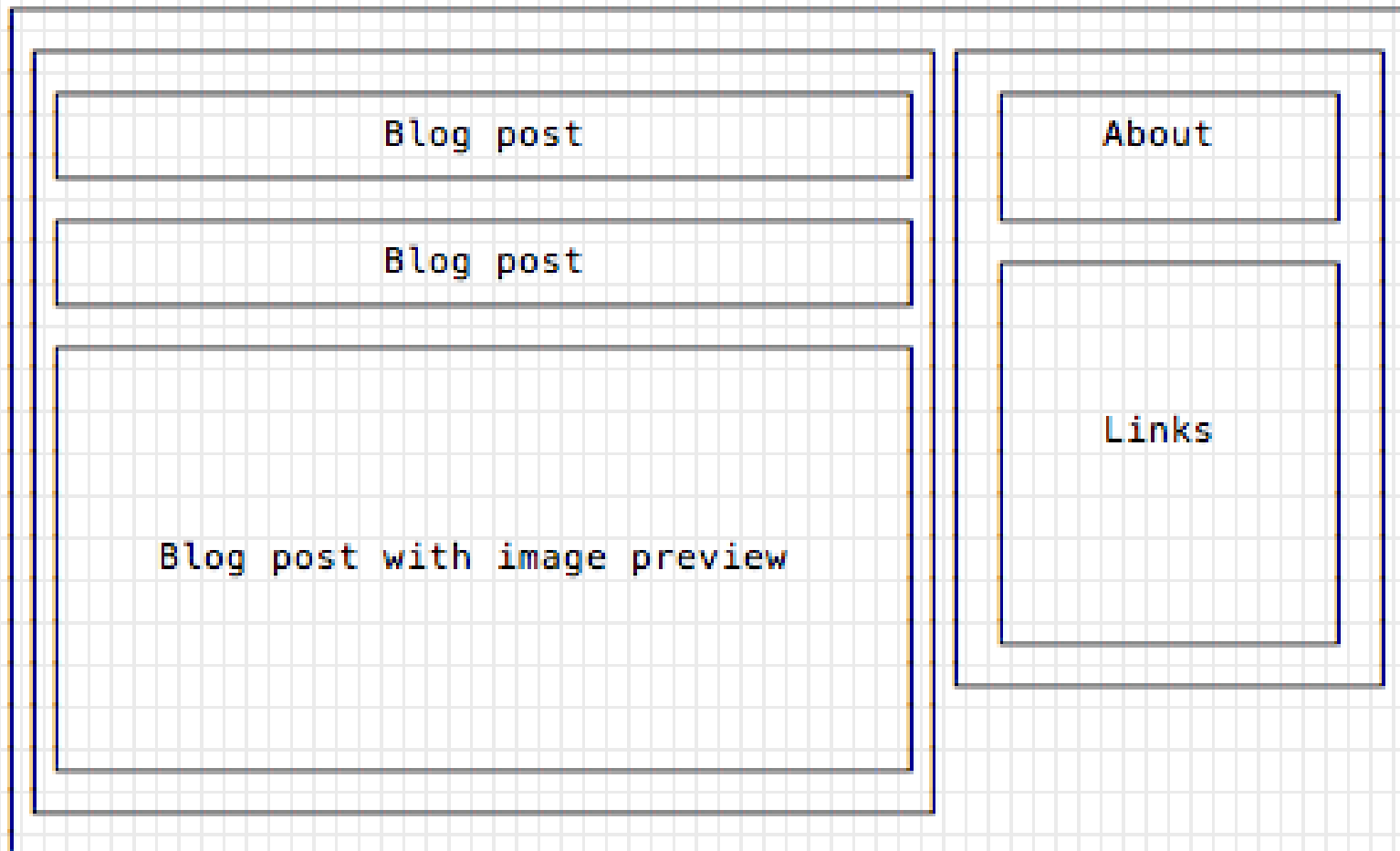
Like

```
const element = document.querySelector();  
element.classList.add();  
element.addEventListener();
```

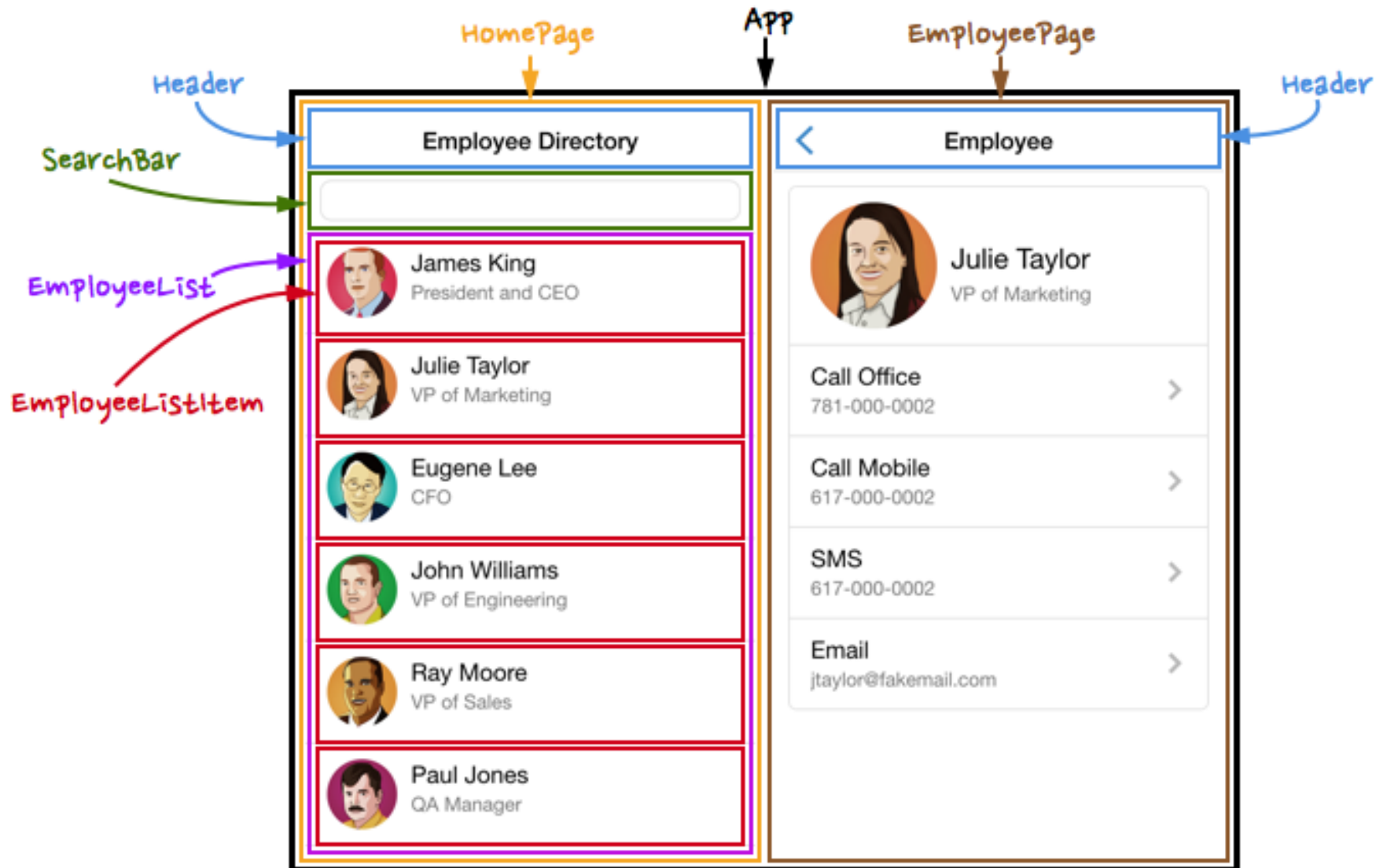
Components

BLOG POSTS

A blog posts listing page



Components EMPLOYEE DIRECTORY



Components

TRAIN NETWORK

TfL Tube Tracker

<Network />

Bakerloo Line

Stations

<Line />

Go

Central Line

Stations

<Line />

Go

Circle Line

Stations

<Line />

Go

District Line

Stations

<Line />

Go

Hammersmith & City Line

Stations

<Line />

Go

Jubilee Line

Stations

<Line />

Go

Metropolitan Line

Stations

<Line />

Go

Station Name

<Predictions />

Platform 1

<DepartureBoard />

| Destination | Due | Current location |
|-----------------|------------|-------------------|
| White City | 0:00 | At Platform |
| Ealing Broadway | <Trains /> | Holland Park |
| West Ruislip | 4:30 | Notting Hill Gate |
| Ealing Broadway | 6:00 | Queensway |

Platform 2

<DepartureBoard />

| Destination | Due | Current location |
|-----------------|------------|-------------------|
| White City | 0:00 | At Platform |
| Ealing Broadway | <Trains /> | Holland Park |
| West Ruislip | 4:30 | Notting Hill Gate |
| Ealing Broadway | 6:00 | Queensway |

React

reacts to **state** changes

Key Takeaways

- How to actually get started
- JSX
- Components
- Props
- State
- render()
- Virtual Dom
- Functional Components

Requirements

Basic understanding of Javascript required

- Objects, Arrays, Conditionals
 - Classes – Object Oriented Programming
 - Destructuring
 - Higher order array fn: forEach, map, filter
 - Arrow functions
 - Fetch API & Promises
- ES6 Modules



SETUP

- Node – Npm
- Understanding Object Oriented Programming
- Familiarity with Functional Programming
- Created Dynamic website before (using jQuery, Angular etc)

```

import { Component, OnInit } from '@angular/core';
import { Component } from 'react';
import PropTypes from 'prop-types';
import { connect } from 'react-redux';
import PostForm from './PostForm';
import PostFeed from './PostFeed';
import Spinner from '../common/Spinner';
import { getPosts, addPost } from '../../actions/postActions';

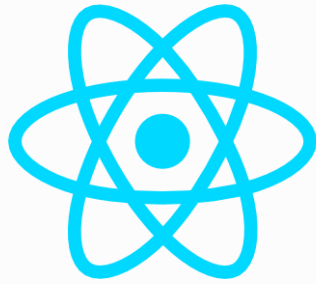
class Posts extends Component {
  componentDidMount() {
    this.props.getPosts();
  }

  render() {
    const { posts, loading } = this.props.post;
    let postContent;

    if (posts === null || loading) {
      postContent = <Spinner />;
    } else {
      postContent = <PostFeed posts={posts} />;
    }

    return (
      <div className="feed">
        <input type="text" v-model="position" required>

```



REACT

> create-react-app
Boilerplate



webpack



ESLint

facebook

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

BABEL

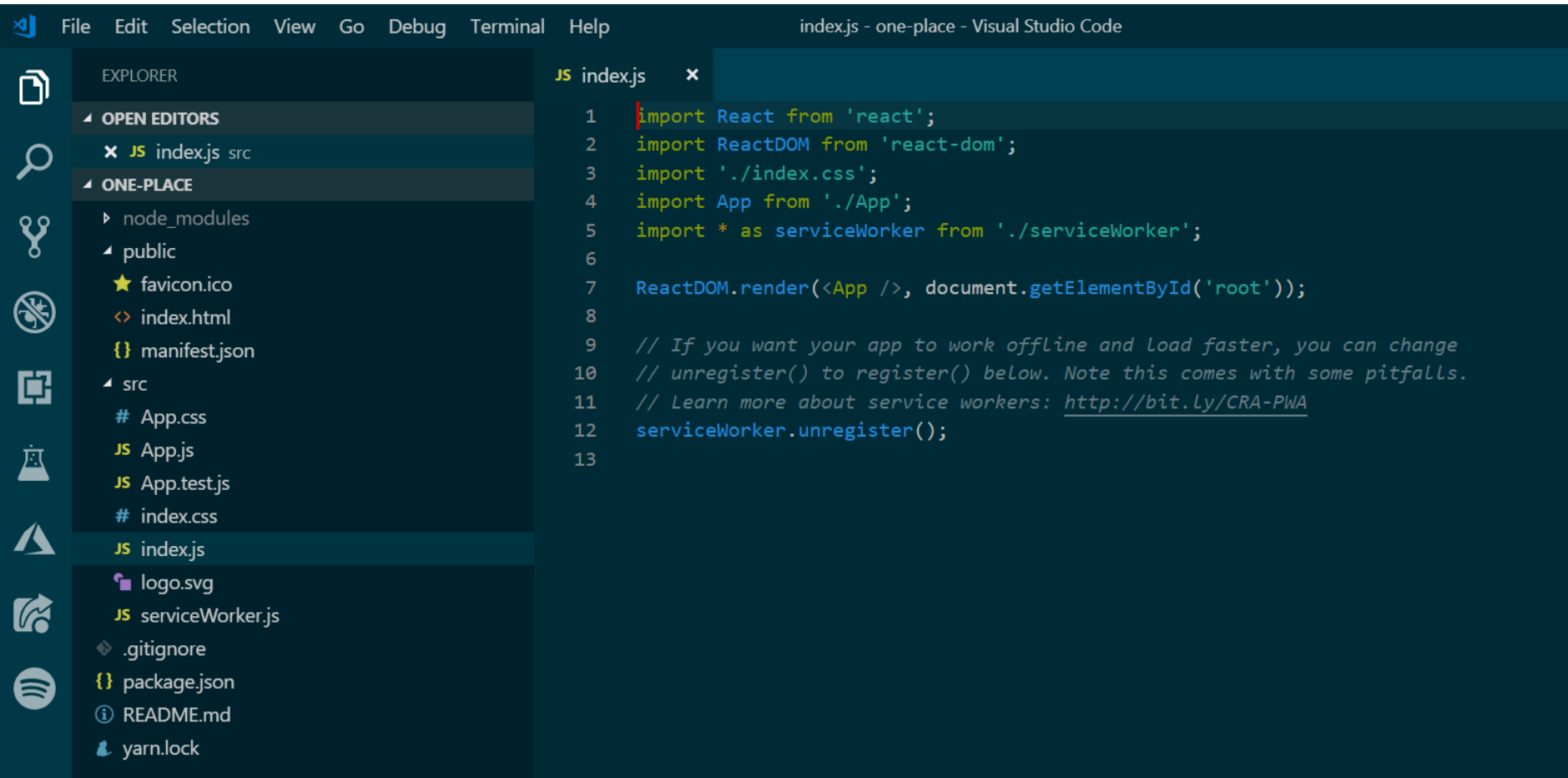
Setup

- Node.js
- npm or yarn [dependency manager]
- `npm install -g create-react-app`

My first app

- `cd ~/code`
- `create-react-app oneplace`
- `cd oneplace`
- `npm start` (starts the local web server, and go to browser to inspect web applications)
- Goto `localhost:3000`
- Congratulations!! (what is inside the folder)

Structure - Boilerplate



The screenshot displays the Visual Studio Code interface. The Explorer sidebar on the left shows the project structure:

- EXPLORER
 - OPEN EDITORS
 - JS index.js src
 - ONE-PLACE
 - node_modules
 - public
 - ★ favicon.ico
 - <> index.html
 - { } manifest.json
 - src
 - # App.css
 - JS App.js
 - JS App.test.js
 - # index.css
 - JS index.js
 - logo.svg
 - JS serviceWorker.js
 - .gitignore
 - { } package.json
 - i README.md
 - 👤 yarn.lock

The main editor window shows the code for `index.js`:

```
1 import React from 'react';
2 import ReactDOM from 'react-dom';
3 import './index.css';
4 import App from './App';
5 import * as serviceWorker from './serviceWorker';
6
7 ReactDOM.render(<App />, document.getElementById('root'));
8
9 // If you want your app to work offline and load faster, you can change
10 // unregister() to register() below. Note this comes with some pitfalls.
11 // Learn more about service workers: http://bit.ly/CRA-PWA
12 serviceWorker.unregister();
13
```


First Component

```
import React, { Component } from 'react';

class Hello extends Component {
  render() {
    return <h1> Hello </h1>;
  }
}

export default Hello;
```

- Import React
- Define Component
- Implement render
- Export component

Inside components folder

src/components/Hello.jsx

src/components/Hello.css

Render the hello component

- Index.js

```
import Hello from './components/Hello';  
ReactDOM.render(<App />, document.getElementById('root'));
```

props

```
import React, { Component } from 'react';

class Hello extends Component {
  render() {
    return <h1> Hello {this.props.name} </h1>;
  }
}

export default Hello;
```

Planning

CSS

HTML

JavaScript Components

← → ↻ ⓘ localhost:3000

Search ...



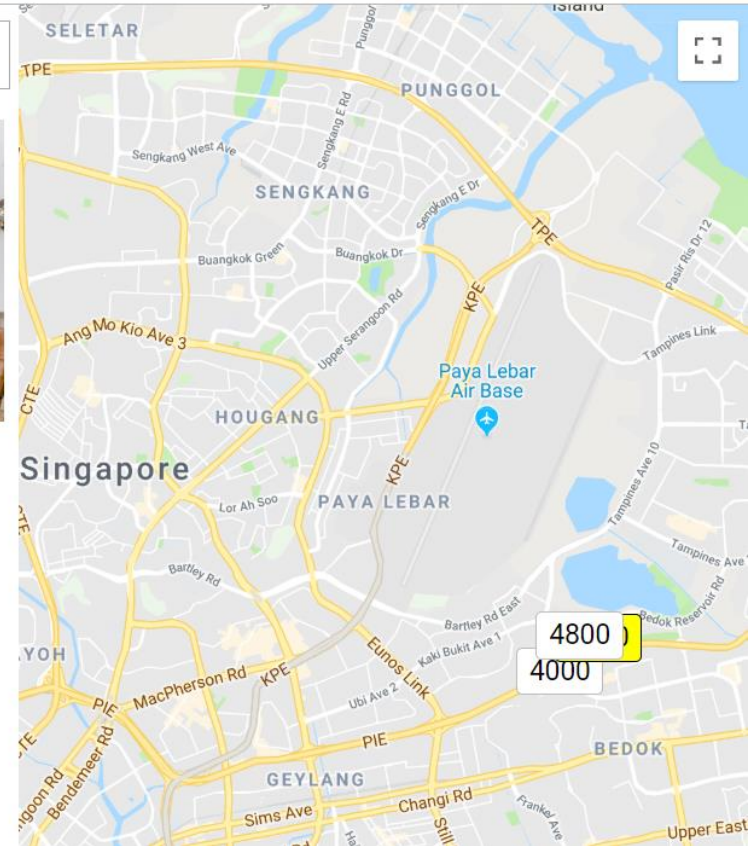
Bedok SGD4000



Changi Beach SGD3900



Marina Bay SGD4800



Flat Component



Marina Bay SGD4800

```
<div class="flat" >  
<div class="flat-picture" />  
<div class="flat-text"></div>  
</div>
```

Flexbox

```
<div class="app">  
<div class="main">  
<div class="search"></div>  
<div class="flats"></div>  
</div>  
<div class="map"></div>  
</div>
```

React State

Constructor

- class App extends Component {

```
  constructor(props) {  
    super(props);  
  
    this.state = {  
      flats: [] ,  
    };  
  }
```

Data Source

<https://raw.githubusercontent.com/TechnoAtSG/flat-data/master/flat-data.json>

```
[{
  "id": 1,
  "place": "Bedok",
  "rent": 288,
  "lat": 1.335007,
  "lng": 103.914533,
  "url": "https://images.unsplash.com/photo-1512917774020-18bb8230272e?ixlib=rb-1.2.1&auto=compress&cs=tinysrgb&h=300&w=300&format=webp",
  "price": 4000,
  "currency": "SGD"
}, {
  "id": 2,
  "place": "Changi Beach",
  "rent": 588,
  "lat": 1.390000,
  "lng": 103.980000,
  "url": "https://images.unsplash.com/photo-1512917774020-18bb8230272e?ixlib=rb-1.2.1&auto=compress&cs=tinysrgb&h=300&w=300&format=webp",
  "price": 4000,
  "currency": "SGD"
}]
```

Ajax

```
componentDidMount() {  
  fetch('https://raw.githubusercontent.com/TechnoA1')  
    .then(response => response.json())  
    .then(data => this.setState({  
      flats: data  
    }));  
}
```

componentDidMount

- `console.log('DidMount');`

```
componentDidMount() {  
  fetch('https://raw.githubusercontent.com/  
TechnoAtSG/flat-data/master/flat-data.json')  
    .then(response => response.json())  
    .then(data => this.setState({  
      flats: data,  
      allFlats: data,  
      selectedFlat: data[0],  
    }));  
}
```

Map & Marker(s)

- google-map-react

import GoogleMapReact from 'google-map-react';

```
<GoogleMapReact bootstrapURLKeys={{ key: "AIzaSyDNdhxwSwms0Fgv9hx5fc16iK3P2rDCRdw" }}
  center={center} zoom={16}
>
  {this.state.flats.map(flat => (
    <Marker key={flat.id}
      text={flat.price}
      lat={flat.lat}
      lng={flat.lng}
      selected = {flat === this.state.selectedFlat} />
  ))}
</GoogleMapReact>
```

Events

```
onClick={this.handleClick}>
```

```
handleClick = () =>{  
    this.props.selectFlat(this.props.flat);  
}
```

```
selectFlat = (flat) => {  
    this.setState({selectedFlat: flat})  
}
```

Search

```
<input type="text" placeholder="Search ..."  
  value={this.state.search}  
  onChange={this.handleSearch} />
```

```
handleSearch = (event) => {  
  this.setState({  
    search : event.target.value,  
    flats: this.state.allFlats.filter((flat)=> new RegExp(event.target.value, "i").exec(flat.pla  
  });  
}
```

JSX

- Mix JavaScript with HTMLs
- And returns as HTML, display properly
- Babel

Components

- Fundamental building blocks for React
- Have internal state and external props
- Can be nested – inside each other, across multiple files.

State

- Internal Information about a component
- Starts Initial state and change it with `this.setState()`

render()

- Called whenever the state changes
- Here, we decide what the user should see based on the state
- Render, does not reloads everything,
- Virtual DOM – add/delete to the page

Q&A