



Andrew Miracle

STRUCTURING A JAVASCRIPT PROJECT



INTRODUCTION

I AM ANDREW MIRACLE

Code validator on indorse.io and Founder of Tecmie, a Software Consulting Company.
Currently an EIT @MESTAFrica

Twitter: @koolamusic

Website: www.andrewmiracle.com

**What is the first thing
you do when you
have a project idea ?**

Structuring a Javascript Project

First Steps

Craft the perfect name

Buy a Domain

Open terminal

create a folder

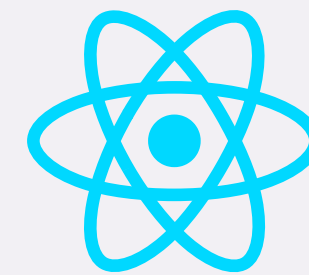
Initialize GIT

CHOOSE A JAVASCRIPT FRAMEWORK

koa



express

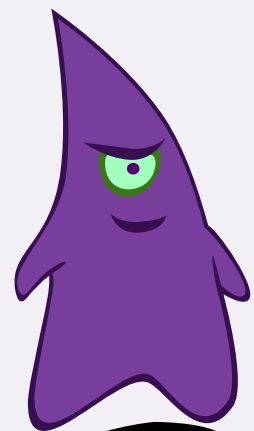


React



ANGULAR

~~NEXT~~.JS



PURPLEJS

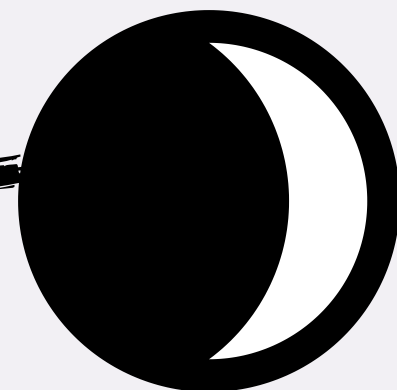


MOCHA



BACKBONE.JS

BABEL



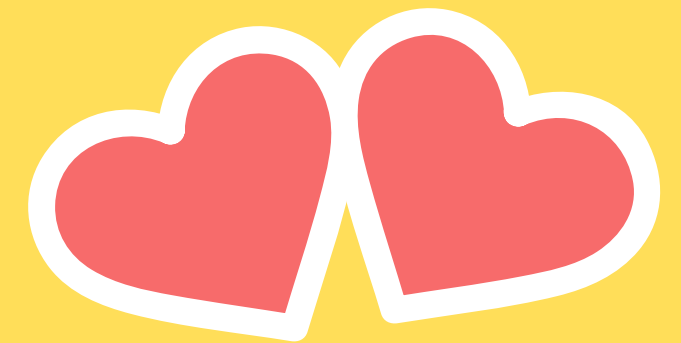
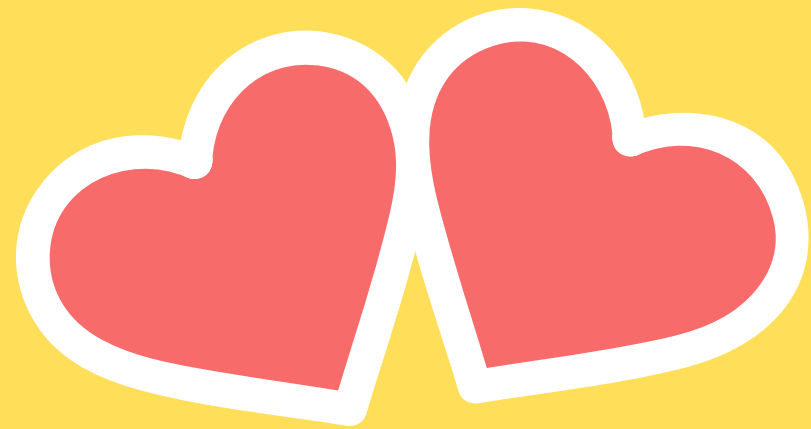
lunr.js



.js



PICK ME



YOU + ME + package.json

=

love

Struct

```
"scripts": {
  "start": "expo start",
  "android": "expo start --android",
  "ios": "expo start --ios",
  "web": "expo start --web",
  "eject": "expo eject",
  "test": "jest"
},
"dependencies": {
  "expo": "^35.0.0",
  "jest": "^24.9.0",
  "react": "16.8.3",
  "react-dom": "16.8.3",
  "react-native": "https://github.com/expo/react-native/archive/s
  "react-native-web": "^0.11.7"
},
"devDependencies": {
  "babel-preset-expo": "^7.0.0"
},
"private": true
```

GETTING STARTED CAN BE UNREASONABLY CONFLICTING

Package Manager

Application Architecture – (SPA, Universal or Isomorphic)

Folder structure

Code Paradigms - DRY - SOLID

Target Devices - Mobile, Web, IOT

Frontend Frameworks

API Framework

Bundlers, Builds & Task Runners

CSS & CSSinJS

CI /Deployment

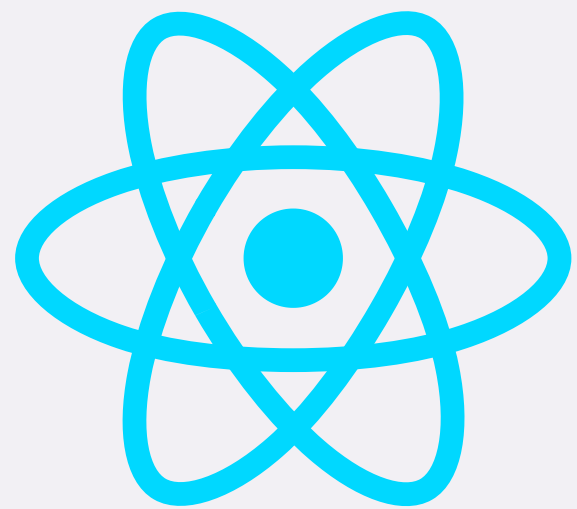
HARD TRUTH



IT DOESNT REALLY MATTER HOW YOU DO YOUR JOB, AS LONG AS YOU GET
SHIT DONE

TARGET DEVICES

Mobile



REACT
NATIVE

<https://facebook.github.io/react-native/>

NATIVE
SCRIPT

<https://www.nativescript.org/>

IONIC/
PHONEGAP

<https://ionicframework.com/>

METEOR/
CORDOVA

<https://meteor.com>

IOT

OBNIZ

<https://obniz.io/>

NODE RED

<https://nodered.org/>

CYCLON.JS

<https://cylonjs.com/>

JERRY SCRIPT

<https://jerryscript.net/>

IOT

Hello World!

Microcontrollers and SoC platforms like to say "Hello World" with a simple blinking LED; the following demonstrates how to do this with the Johnny-Five framework.

1. Install [Node.js](#) (Prefer LTS!).
2. [Setup your board](#).
3. Get Johnny-Five: `npm install johnny-five *`
4. Run your program! `node blink.js`

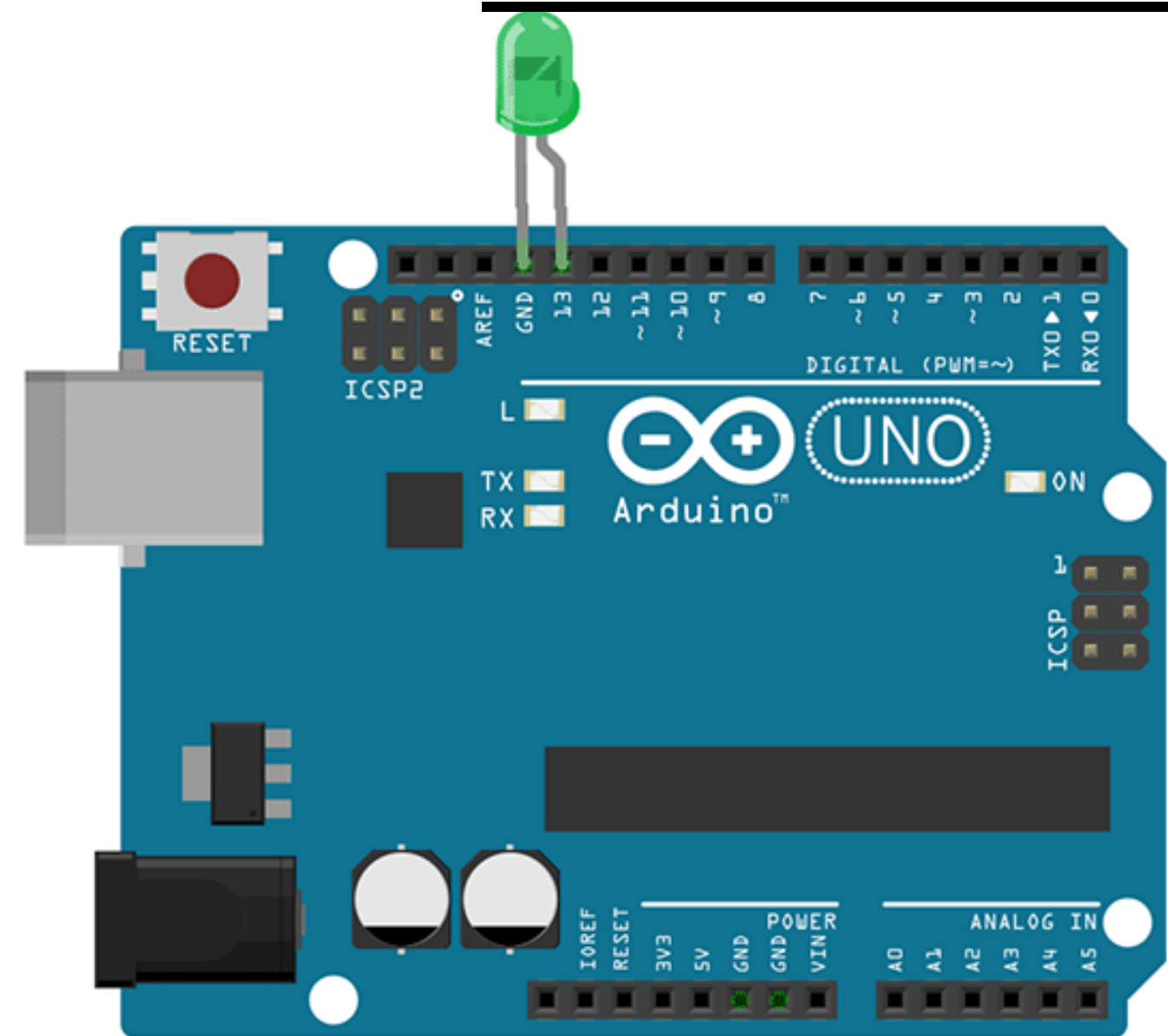
[Arduino](#) | [Tessel 2](#) | [Raspberry Pi](#) | [Intel Edison](#) | [Particle Photon](#) | [See All...](#)

```
var five = require("johnny-five");
var board = new five.Board();

board.on("ready", function() {
  var led = new five.Led(13);
  led.blink(500);
});
```

[Browse more examples](#)

* There are likely to be other steps involved in preparing your target platform, the best place to start is by looking at [Platform Support](#).



HELLO WORLD IOT WITH J5

<http://johnny-five.io/>

Frontend

REACT

<https://reactjs.org/>

ANGULAR

<https://angular.org>

VUEJS

<https://vuejs.com/>

EMBER

<https://emberjs.org>

API

ex



APOLLO

METEOR

LOOPBACK

EXPRESS

SAILS

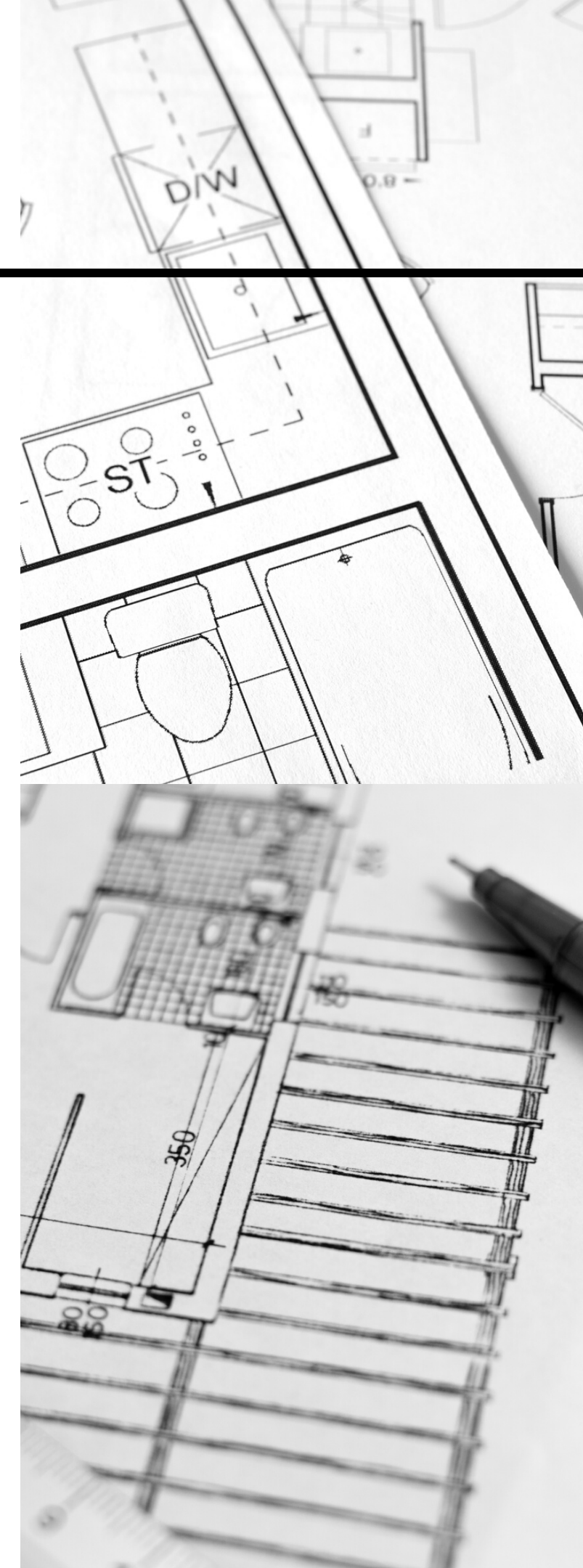
Package Managers



APPLICATION ARCHITECTURE

SPA - SINGLE PAGE APPLICATIONS

The most common, high-level architecture for web applications is called SPA, which stands for Single Page Application. SPAs are big blobs of JavaScript that contain everything the application needs to work properly. The UI is rendered entirely client-side, so no reloading is required.

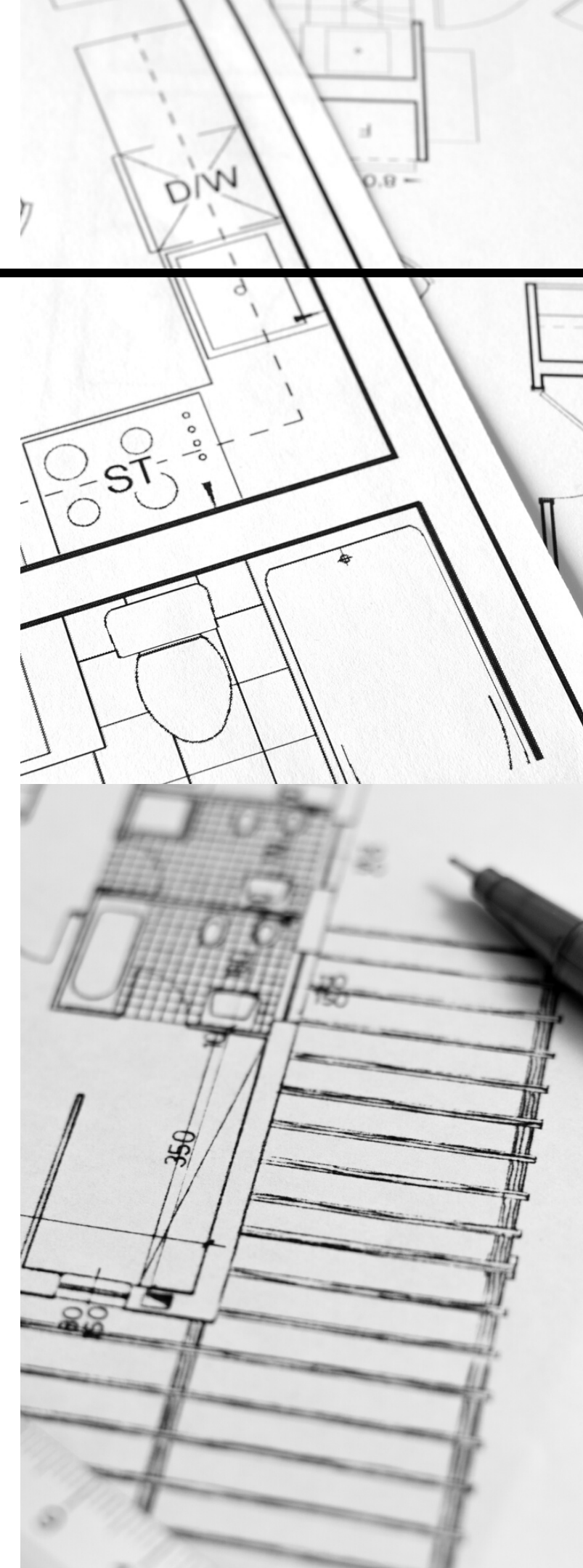


APPLICATION ARCHITECTURE

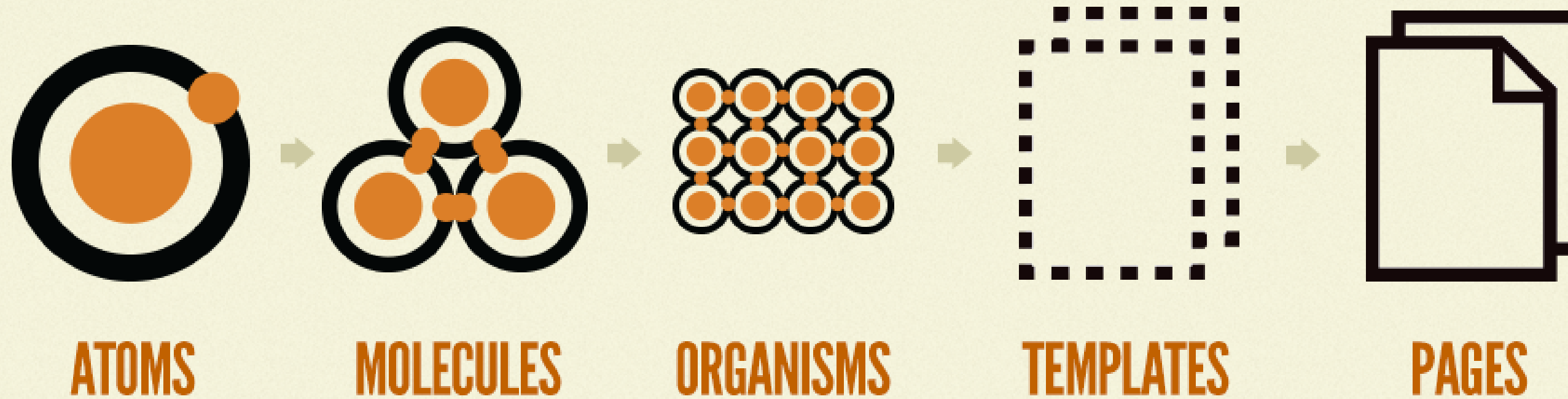
ISOMORPHIC/ UNIVERSAL APPS

An isomorphic app is a web app that blends a server-rendered web app with a single-page application. On the one hand, we want to take advantage of fast perceived performance and SEO-friendly rendering from the server.

In this type of architecture, most of the code can be executed both on the server and the client. You can choose what you want to render on the server for a faster initial page load, and after that, the client takes over the rendering while the user is interacting with the app.



FOLDER STRUCTURE



ATOMIC DESIGN PRINCIPLE -
BEST APPLIED IN FE DEV.

FOLDER STRUCTURE

```
▸ app
▸ docs
▸ lib
▸ migrations
▸ node_modules
▸ tests
6 .babelrc
🔗 .eslintrc
🔗 .gitignore
JS config.js
JS config.sample.js
JS migrate.js
{} package.json
```

FRACTAL APP STRUCTURE -
BEST APPLIED IN BE DEV.

CSS LIBRARIES

CSS IN JS

CSS LIBRARIES



Bootstrap

Zurb Foundation

<https://foundation.zurb.com/>

Tachyons CSS

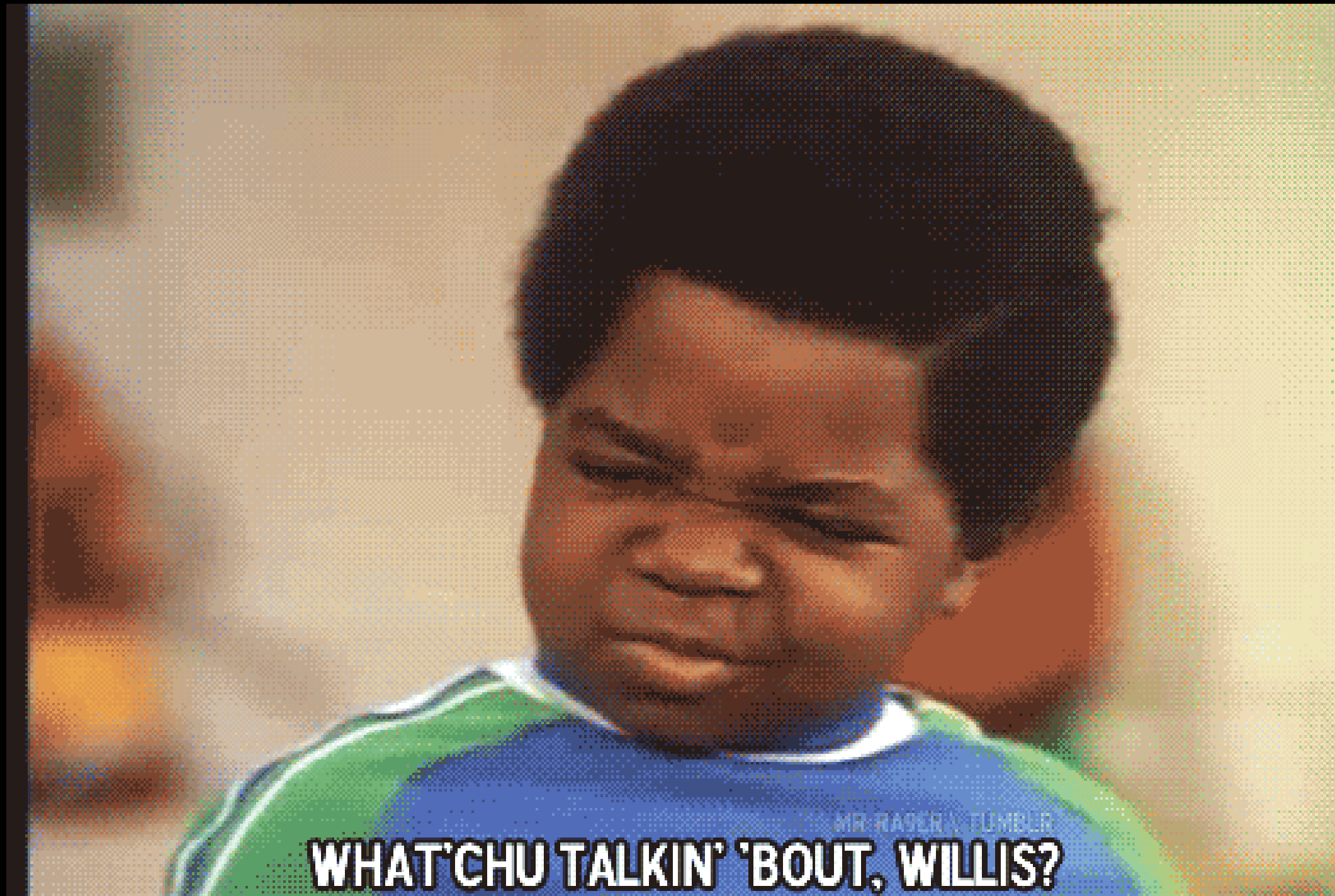
<https://tachyons.io/>

Bulma CSS

<https://bulma.io/>

CSS-IN-JS

THERE SO MANY CHOICES



CSS IN JS

- aphrodite
- babel-plugin-css-in-js
- babel-plugin-pre-style
- bloody-react-styled
- classy
- csjs
- css-constructor
- css-light
- css-loader
- css-ns
- cssobj
- cssx-loader
- emotion
- es-css-modules
- glamor
- glamorous
- hyperstyles
- i-css
- j2c
- jsxstyle
- linaria
- pre-style
- radium
- react-css-builder
- react-css-components
- react-css-modules
- react-cssom
- react-cxs
- react-fela
- react-free-style
- react-inline-css
- react-inline-style
- react-inline
- react-jss
- react-look
- react-native-web
- react-statics-styles
- react-styl
- react-style
- react-styleable
- react-stylematic
- react-theme
- react-vstyle
- reactcss
- restyles
- scope-styles
- smart-css
- stile-react-media-queries
- stilr
- style-it
- styled-components
- styled-jsx
- styletron-react
- styling
- superstyle
- typestyle
- uranium

CSS IN JS == JS

Objects

```
const style = {  
  fontSize: 12,  
  color: 'red',  
  
  ':hover': {  
    color: 'blue'  
  },  
  
  '@media (min-width:320px)' {  
    fontSize: 12  
  }  
}
```

Template strings

```
const style = `  
  font-size: 12px;  
  color: red;  
  
  &:hover {  
    color: blue;  
  }  
  
  @media (min-width:320px) {  
    font-size: 14px;  
  }  
`
```

BUNDLERS AND TASK RUNNERS

WEBPACK, PARCEL, ROLLUPJS

WEBPACK

Webpack is an open-source JavaScript module bundler. It is a module bundler primarily for JavaScript, but it can transform front-end assets like HTML, CSS, and images if the corresponding plugins are included. Webpack takes modules with dependencies and generates static assets representing those modules.

[HTTPS://WEBPACK.JS.ORG/](https://webpack.js.org/)

PARCEL

Parcel is a web application bundler, differentiated by its developer experience. It offers blazing fast performance utilizing multicore processing, and requires zero configuration.

[HTTPS://PARCELJS.ORG/](https://parceljs.org/)

DEPLOYING YOUR APP

IAAS

Infrastructure As A Service

PAAS

Platform As A Service

SAAS

Software As A Service



INFRASTRUCTURE AS A SERVICE

IaaS or Infrastructure as a Service is basically a virtual provision of computing resources over the cloud. An IaaS cloud provider can give you the entire range of computing infrastructures such as storage, servers, networking hardware alongside maintenance and support.



PLATFORM AS A SERVICE

Platform as a Service or PaaS is essentially a cloud base where you can develop, test and organize the different applications for your business. Implementing PaaS simplifies the process of enterprise software development. The virtual runtime environment provided by PaaS gives a favorable space for developing and testing applications.



App Engine

SOFTWARE AS A SERVICE

SaaS or Software as a Service is a model that gives quick access to cloud-based web applications. The vendor controls the entire computing stack, which you can access using a web browser. These applications run on the cloud and you can use them by a paid licensed subscription or for free with limited access.



DRY

DONT REPEAT YOURSELF

SOLID

Single Responsibility Principle

Open/Closed Principle

Liskov Substitution Principle

Interface Segregation Principle

Dependency Inversion

SECURITY

Always validate input

Not every feature requires a new npm package

Try to maintain a log

When unsure, use a PAAS

LET'S TALK!

MAILING ADDRESS

me@andrewmiracle.com

WEBSITE

andrewmiracle.com

@HANDLE

@koolamusic

