**FIAP**

**2018**

**Web Moderna**

**Trabalho de pesquisa**

**Integrantes**

Aline Chaves – RM331494

Bruna Bertollini – RM331749

Danilo Maderal – RM330618

**BOWER – GERENCIADOR DE DEPENDÊNCIAS** 

Bower é um gerenciador de dependências de componentes de front-end, que permite automatizar o processo de instalação dos arquivos necessários para o projeto, através de um arquivo de manifesto.

O Bower depende do Node.js para funcionar e sua instalação é realizada através da utilização do NPM (Node Package Manager).

Após instalado, é criado um novo arquivo de configuração chamado “bower.json”, onde poderão ser adicionadas dependências através do comando “bower install *dependencia*”. Ao configurar as dependências do projeto, as mesmas serão baixadas automaticamente em qualquer ambiente que o projeto for instalado.

Por padrão, o bower utiliza o diretório “bower\_components/”, que pode ser modificado através da criação de um novo arquivo “.bowerrc”, indicando o novo diretório que deseja utilizar, como por exemplo:

{

"directory" : "assets/components"

}

Como exemplo de pacotes a serem instalados pelo Bower, podemos citar Bootstrap e jQuery.

Para utilizar um pacote instalado pelo Bower, é necessário somente incluir o comando <script src="/assets/components/jquery/jquery.min.js"></script> na página.

**GRUNT – AUTOMATIZADOR DE TAREFAS COMUNS **

O Grunt é uma aplicação de linha de comando que tem como objetivo automatizar tarefas, principalmente tarefas em aplicações JavaScript, utilizando a própria linguagem JavaScript para escrever as tarefas e o Node.JS para executá-las.

Para utilizar o Grunt em um projeto, é necessário que exista dois arquivos: o Gruntfile.js e o package.json.

O arquivo Gruntfile.js é um arquivo JavaScript que define a configuração das tarefas a serem executadas pelo Grunt e como irão se comportar, podendo ser criado través de um editor de texto ou através do comando grunt init. Ele é composto da função principal que engloba tudo, das configurações e tarefas da aplicação, do carregamento de grunt plugins e de tarefas personalizadas.

O arquivos package.json armazena as informações da aplicação, como dependências e os plugins do grunt que serão utilizados pelo projeto. Através do comando "npm install" ele procura as dependências descritas neste arquivo, e as instala na pasta do projeto com suas respectivas versões.

O Grunt também dispões diversas opções de plugins que facilitam tarefas, como min, concat e grunt-rsync, plugin utilizado para deploy, mantendo o diretório remoto sempre em sincronia com o diretório base, enviando somente os arquivos que foram alterados.

**SASS – LINGUAGEM DE FOLHAS DE ESTILOS **

Sass é uma linguagem de script que, depois de compilada, gera um css. Nele existem duas sintaxes: o SCSS que é semelhante com CSS normal e o SaSS, que tem que ser identada para funcionar.

Para instalar o Sass, primeiro é obrigado a ter o Ruby instalado em sua máquina e deixá-lo no seu path. Depois é só abrir o prompt de comando e dar o comando “gem install sass”.

Ele é tipado, tem vários tipos de variáveis dentro dele:

• String: usando a anotação “@charset”;

• Números: ele não distingue números inteiro e flutuantes e os números a direito não servem para nada, então devem ser tirados;

• Cores: como o nome sugere são cores, só pode suar e devem ser escritas com HSL.

Possui comentários igual do Java, Loops usando “@for”, “@each” e “@while”, condicionais usando o “@if” e o “@else”, herança usando o “@extend” e avisos de erros.

O Mixins usa o um compomente DRY, que usada a não repetir o código.

Alguns frameworks do Sass:

• Sierra: o menor e mais leve framework do Scss;

• Compass: o mais usado, ele é muito pesado;

• Kickoff: ele roda em Sass, mas tem sua própria nomenclatura.

**Exemplos**:

1. **Sass:**

$cor-primaria: #3bbfce

$margem: 16px

.content-navigation

border-color: $cor-primaria

color: darken($cor-primaria, 10%)

.border

padding: $margem/2

margin: $margem/2

border-color: $cor-primaria

1. **Scss:**

$cor-primaria: #3bbfce;

$margem: 16px;

.content-navigation {

border-color: $cor-primaria;

color: darken($cor-primaria, 10%);

}

.border {

padding: $margem / 2;

margin: $margem / 2;

border-color: $cor-primaria;

}

**Referencias**

<http://www.carvalhoweb.com/articles/bower-guia-definitivo/> - Bower: O guia/tutorial definitivo

<https://bower.io/> - Bower - A package manager for the web

<https://tableless.com.br/bower-na-pratica/> - Bower na prática

<https://nandovieira.com.br/gerenciando-dependencias-client-side-com-bower> - Gerenciando dependências client-side com Bower

<https://tableless.com.br/grunt-voce-deveria-estar-usando/> - Grunt: você deveria estar usando! <https://pt.wikiversity.org/wiki/Grunt> - Grunt

https://pt.wikipedia.org/wiki/Sass\_(linguagem\_de\_folhas\_de\_estilos)

https://sass-guidelin.es/pt/

http://blog.caelum.com.br/css-menos-sofrido-com-sass/

https://www.webdesignerdepot.com/2017/06/9-exciting-open-source-sass-frameworks/