**Website**

**Murano Design**

Guia do Programador

1. **Projeto**

Este projeto foi desenvolvido em HTML5, PHP5, e outras tecnologias que serão mencionadas com melhor detalhamento no decorrer do documento e no tópico nº 2.

O website, tanto HTML quanto estilo SCSS, foi separado em partes, cada parte é um “Row” totalmente independente do layout original do site, semelhante a uma torre.

|  |
| --- |
| Cabeçalho/Menu Mobile |
| Introdução |
| Quem Somos |
| Funcionários |
| O Que Fazemos |
| O Que Fizemos |
| Contato/Fale Conosco |
| Rodapé |

Diferente do Arquivo **Script.js** que está todo estruturado em um arquivo só, separado por comentários.

Para seu desenvolvimento, deve-se ter algumas dependencias instaladas em sua máquina (Vide tópico 2), algumas são opcionais e este documento o auxiliará a utilizá-los no website, Instale-os e passe para o próximo parágrafo.

Caso tenha dificuldades pra instalar o Compass utilize o Gems do Ruby. Neste caso, Abra o CMD (Windows+R > escreva “CMD” > Enter), e escreva “gem install compass”. Caso tudo ocorra bem, em poucos minutos terá Compass instalado em sua máquina.

**Antes da programação**

Após instalar as dependências em sua máquina, deverá inicializar os programas de segundo plano. Neste caso, referindo ao Compass.

Caso se sinta melhor utilizando algum software próprio, Compass tem um em seu site oficial, pode baixa-lo e utilizá-lo do modo que desejar.

Agora vamos iniciar compilador do Compass.

**Iniciando : Compass**

Não irei me aprofundar no Sass como linguagem, mas pode utilizá-lo igual ao CSS, mas com algumas vantagens. A documentação do site explica bem o que deve ser feita, é um ótimo local para quem está iniciando.

Abra o CMD (Windows+R > escreva “CMD” > Enter). Vá até a pasta do projeto MuranoDesign (“cd pasta/para/projeto”), e inicialize o Compass, escrevendo “Compass watch sass/styles.scss”. Após isso ele vai estar verificando o arquivo **styles.scss**.

O comando *watch* serve para que toda vez que uma modificação for feita em qualquer arquivo conectado e o próprio **styles.scss,** compile de *SCSS* para *CSS*.

Caso quiser compilar para modo de produção (CSS Minify), pode-se utilizar o comando “Compass watch sass/styles.scss –output-style=compressed”. Assim, todo o código compilado será minificado, economizando no carregamento da pagina.

Para cancelar o Compass, use o comando Ctrl+C.

**Iniciando : Xampp**

Para testar os arquivos \*.php deve ter um suporte para rodar estes arquivos. E Xampp é uma plataforma que eu recomendo utilizar, ele vem com Tomcat, Apache, MySQL, Filezilla(Recomendado baixar do site oficial) e Mercury.

Neste caso vamos utilizar somente o Apache, o Website não tem Banco de Dados.

Caso tenha problema na instalação do Apache tente :

1. Abrir no modo Administrador, e Tente instalar á esquerda do nome “Apache”;

Ao Instalar, deve-se aparecer um Simbolo de “Correto”.

1. Modificar as portas de conexão nos arquivos de configuração do Apache(httpd.conf), em certos casos, a porta utilizada no Apache é a mesma utilizada no Skype.

A pasta deve ser salva no diretório htdocs (é comum que ela seja instalada no endereço C:\xampp\htdocs). Após copiar a pasta do projeto pra lá, entre no endereço localhost:[porta]/NomeDaPastaDoProjeto. O Projeto deverá abrir o conteúdo sem problemas. Caso haja erro, verifique a versão do PHP.

**Programação:PHP**

Exite o arquivo principal, **index.php**, que importa cada “*row*” para esta pagina, que é montada e mostrada para o usuário. Os arquivos em partes estão na pasta *assets/includes*. Lá você pode mexer em cada arquivo separadamente sem influenciar nas paginas adjacente. Tem total liberdade em mexer em uma certa tela do website e, talvez o caso, adicionar uma “*row*” nova.

No começo do arquivo há uma verificação do Localhost, se for numa maquina local e não em um servidor online, ele irá trazer os arquivos \*.php da pasta assets\includes.

Caso esteja em um servidor online, ele irá trazer os arquivos \*.php da pasta assets\includes\minify, que estão minificados para melhor desempenho no carregamento do website.

Cada tela, ou *Row* do site é uma tag separada. Um #ID diferente para cada camada, assim facilitando a indentificação da div para com o documento, tanto HTML quando SCSS.

Foi utilizado *Bootstrap* para facilitar o desenvolvimento de uma página única, sempre visando o *Mobile-First*. Para fazer testes de performance, há o *YSlow* e *PageSpeed by Google.*

A class “.*ContentMax*” foi feita para criar uma margem nas laterais das divs, para que não colidam com a borda. Isso foi feito para retirar a div “.container”, que estava interferindo no layout do site.

Usuário não pode redimensionar o website no mobile. Caso deseje modificar isso, verifique na tag “*head*” a meta-tag “*viewport*” no arquivo **index.php**.

Estilos devem ser chamados no início do arquivo (mais precisamente na tag *Head*)

Scripts devem ser chamados no fim do arquivo (mais precisamento no fim do *body*)

**Programação:SCSS**

A linguagem SCSS foi escolhida pois, diferentemente do SASS, tem mais semelhança com o código original do CSS, e a mudança para SASS poderia influenciar e dificultar o desenvolvimento e manutenção futura sendo que as sintaxes são diferentes.

Outro motivo para sua escolha foi sua estrutura. O desenvolvedor consegue importar arquivos diferentes, utilizar variáveis, funções, dentre outros. Mas o importar foi essencial para que o site fosse dividido em vários arquivos, um para cada tela, para que a manutenção fosse facilitada. E que pudesse ser compilada em um arquivo único e utilizado no site inteiro.

Aqui está alguns detalhamentos dos arquivos SCSS :

**Styles.scss**

Este arquivo serve para importar todos os estilos SCSS’s, juntar em um só e no final compilar em um unico arquivo para diminuir as requisições de HTTP.

**Pages/\*.scss**

Os arquivos estão separados em telas. Cada arquivo é referente a uma tela do Website, e isso foi feito dessa forma para facilitar o desenvolvimento e manutenção do estilo.

**\_custom.scss**

Este arquivo contém todas as telas importadas, caso precise adicionar ou retirar um estilo/arquivo .scss, é neste local que estão as importações.

**\_customVariables.scss**

Variáveis do SCSS são setadas neste arquivo. Variáveis como Cor de fundo, Cores dos Títulos, Largura da Imagem, etc. Caso precise, pode-se adicionar uma nova Variável, porém deve ter cuidado ao retirar alguma, deixei comentários sobre o que cada variável se refere ao site.

**\_customClasses.scss**

Em certos casos, não foi possível ser específico ou um estilo serve para o site todo, incluindo todos os arquivos pages/\*.SCSS, então foi criado um arquivo de Classes customizadas para utilizar um estilo global. Nesta parte, a modificação de uma classe influencia em todas de mesmo nome no website.

**Responsive**

O Website é responsivo, e os media queries utilizados são originais aos do Bootstrap:

Extra small devices Phones (<768px)

Small devices Tablets (≥768px)

Medium devices Desktops (≥992px)

Large devices Desktops (≥1200px)

**Programação:Javascript/JQuery**

Há somente um arquivo editável de javascript, o **script.js** tem as funções de todas as páginas, incluindo sobre as Fotos dos Funcionários. Ele foi dividido em seções de comentários, caso esteja utilizando Sublime Text, Dreamweaver, Netbeans e afins, terá a possibilidade de resumir o código por ele estar indentado. Assim facilitando o desenvolvedor a encontrar parte referente a cada tela.

No inicio há a lista de variáveis globais, também identificada/nomeada com cada tela.

Todo o código tem comandos básicos, e é autoexplicativo com exceção de uma parte.

Nomeado como “Pessoas no Grupo”, é a parte do script que tem as funcionalidades das fotos dos funcionários. Foi deixado um tanto quanto automático, mas alguns números e nomes devem ser modificado a cada alteração do layout atual.

**Variavel Pessoas :**

Esta variável serve para setar, de acordo com a imagem (images/sprite-group\_15.png), quantas e quais pessoas aparecerão na tela. A variável tem um estrutura semelhante ao JSON, tornando fácil a manutenção e utilização.

A estrutura se segue com :

|  |  |
| --- | --- |
| Nome | Nome do Funcionário |
| NomeCompleto | Nome a ser Mostrado na Tela |
| X | (Desktop/Tablet) Posição X |
| Y | (Desktop/Tablet) Posição Y |
| Width | (Desktop/Tablet) Largura da Caixa de Texto |
| Texto | Texto a ser mostrado quando selecionado |

Os números e palavras podem ser facilmente modificados diretamente pelo script.

**Função LinesHD()/LinesTab():**

Duas funções bem semelhantes, elas tem a mesma função, mas em dispositivos diferentes, uma para Desktop(HD()) e outra para Tablet(Tab());

Essa função serve para conectar uma pessoa com outra por uma linha de pontos roxos, mas simplesmente fazer uma função que conecta todos não faz parte da proposta de design do website, então o desenvolvedor terá que fazer essa parte *na unha*.

Dentro da função, ela chama outra, chamada *createLine(Args1, Args2, Args3)*.

Args1 = De onde começa a linha de conexão (Nome Pessoa 1);

Args2 = Aonde deve terminar a linha de conexão (Nome Pessoa 2);

Args3( opcional ) = “Essa linha deve sumir quando eu selecionar alguma pessoa”. Esse valor pode ou não estar na função, e serve para que, caso uma linha fique na frente do texto, simplesmente retire-a ao selecionar uma pessoa específica.

**Lembrete** : Toda a função das Fotos dos Funcionários tem relação com a imagem original. Caso modifique a imagem, possivelmente terá que modificar o código.

Caso queira usar o **Gulp** (Automatizador de Tarefas), instale **NodeJS**, escreva todas suas necessidades no arquivo **Gulpfile.js** e inicialize-o de acordo com sua preferencia.

Se desejar utilizar o BrowserSync sem o NodeJS, existem extensões de Browser que faz um *Reload* automático ao modificar o CSS ou HTML do website.

1. **Dependencias**

* Ruby & Gems(necessário)
* Sass(necessário)
* Compass(necessário)
* Xampp || Wamp (Apache/PHP 5) (necessário)
* Nodejs(opcional)
* Gulp(opcional)
* BrowserSync(opcional)

1. **Referências**

Ruby Website : <https://www.ruby-lang.org/pt/>

Ruby Installer : <http://rubyinstaller.org/>

Sass Website : <http://sass-lang.com/>

Compass Website : <http://compass-style.org/>

Xampp Website : <https://www.apachefriends.org/pt_br/index.html>

Wamp Website : <http://www.wampserver.com/en/>

Apache Website : <http://www.apache.org/>

Node.js Website : <https://nodejs.org/en/>

Gulp Website : <http://gulpjs.com/>

BrowserSync website : <http://www.browsersync.io/>

Bootstrap Website : <http://getbootstrap.com/>

YSlow Website : <http://yslow.org/>

PageSpeed Insights : <https://developers.google.com/speed/pagespeed/insights/>

Site do Projeto (Murano Design) : <http://muranodesign.com.br/>