

# # STATIC CMS

## Criação de um Site com Conteúdo Estático Com bons recursos para gerenciamento do conteúdo

**Achei que seria interessante mudar o nome deste projeto para Static CMS, pois esta é a ideia, criar uma estrutura que forneça as facilidades de gerenciamento de conteúdo que um bom CMS fornece, além de contar ainda com bons recursos e grande controle por conta da simplicidade dos componentes.**

Após ter criado o site usando o Bootstrap, que tem ótimos recursos, decidi agora criar outro site, com o mesmo template (aparência) e as mesmas funcionalidades mas apenas com recursos mais simples, sem o bootstrap, para que possa ter um controle completo sobre o mesmo.

### Objetivo

O objetivo é criar uma estrutura de site de uso tão prático quanto usando um CMS, mas somente com conteúdo estático e recursos simples. Usando boas práticas e boas ferramentas para poder abrigar este site em servidores gratuitos como o GitHub.

### Controle

Neste projeto preciso de controle completo sobre a estrutura do site, por isso usarei apenas componentes bem simples ou apenas código simples em HTML, CSS e JavaScript mas ainda assim também simples de entender e alterar.

### Solução Adotada

- Criar um arquivo cabeçalho.html, contendo o head e o menu
- Criar um pequeno script com o método load da jQuery que importa o cabeçalho.html
- Adicionar este pequeno script em cada arquivo de conteúdo

### Obs

Importante usar um virtual host ou seu host como único host de todos os arquivos, pois os links são absolutos.

Exige usar o servidor só para si.  
Todos os links são absolutos:

Exemplo: href="/joomla/"  
/basicos  
/cakephp

### Recursos utilizados:

- Usei um template com layout moderno e agradável (O Visualize criado pela Templated - <https://templated.co/visualize>)
- O layout do template é responsivo
- Como grid usei SimpleGrid - <https://thisisdallas.github.io/Simple-Grid/>

- Menu dropdown multinível e fixo mas agora na lateral esquerda

<http://stackoverflow.com/questions/8141513/how-to-create-a-css-only-vertical-drop-down-menu>

Além de funcional tem um código bem simples de entender e fácil de alterar.

- Editor HTML FCKEditor para formatar o conteúdo e criar as páginas (<http://fckeditor.com>)

- Editor Ace, que ajuda a corrigir erros no HTML, JavaScript, CSS e muito mais

(<https://github.com/ajaxorg/ace-builds>)

Evitarei usar a ferramenta acima procurando indentar rigorosamente as tags para que possa identificá-las com facilidade

- O path de todos os includes deve ser o path absoluto, para que seja visto por todos os subdiretórios

- Os includes dos javascript são inseridos ao final de cada arquivo de conteúdo, visto ser pouco conteúdo, mas pode ser criado um outro arquivo e importado com o load. Lembrando que nem sempre isso é possível. Alguns componentes exigem que o JavaScript esteja no início para funcionar. Mas neste projeto não foi necessário nenhum arquivo de javascript, a não ser o da jquery, por conta do método load.

## Domínio Virtual

Começamos criando um domínio virtual para nosso projeto, que chamarei de projetoc.dev que fica na pasta /var/www/html/projetoc

Baixei o template e descompactei na pasta do projetoc. Este template vem apenas com uma pasta assets (css, js e imgs), uma pasta images (imagens do portfolio) e um arquivo index.html e um LICENSE.txt, da licença Creative Commons.

Chamei pelo navegador para verificar o que deixo e o que removo.

## Veja ele original



**This is Visualize, a responsive site template designed by [TEMPLATED](#) and released for free under the Creative Commons License.**

- [Twitter](#)
- [Facebook](#)
- [Instagram](#)
- [500px](#)
- [Email](#)

[Lorem ipsum dolor sit amet](#)

[Lorem ipsum dolor sit amet](#)

[Lorem ipsum dolor sit amet](#)

[Lorem ipsum dolor sit amet](#)

[Lorem ipsum dolor sit amet](#)

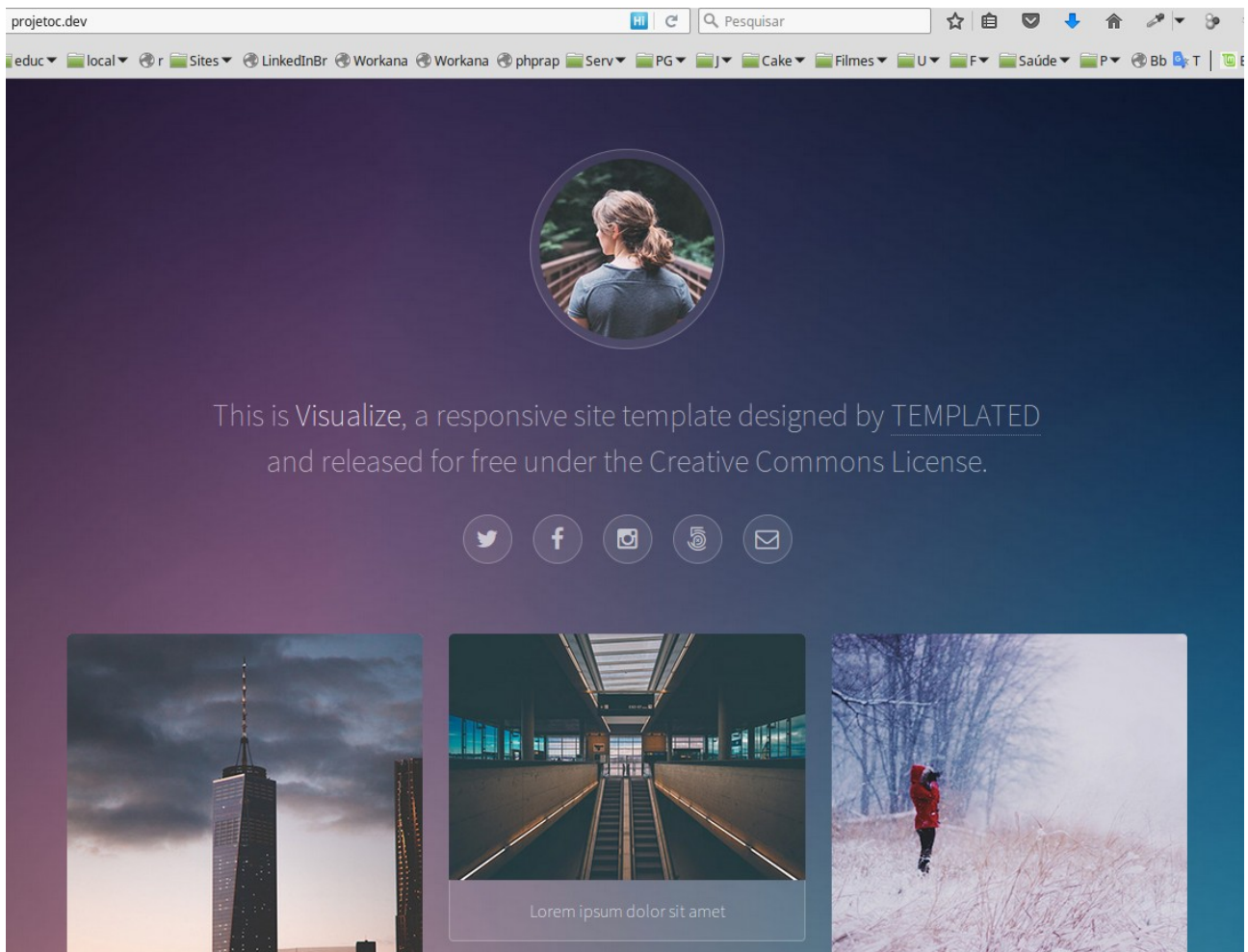
[Lorem ipsum dolor sit amet](#)

[Lorem ipsum dolor sit amet](#)

Por que isso?

Por que estou usando no linux em meu desktop e ainda não encontrei uma forma de setar as permissões de forma automática. Como descompactei o conteúdo para a pasta no /var, agora preciso executar um script que criei para ajustar as permissões.

Veja agora:



Agora vamos começar.

### Qual é a ideia deste template?

Veja que não existem menus. Parece que quer apenas mostrar um portfolio juntamente com uma foto e contatos.

Vou trocar a foto por uma minha, mas também meio de perfil, visto que não estou querendo vender minha imagem. O colega André me ajudou aqui, criando esta foto. Valeu novamente André.

Vou trocar as imagens do postfolio por ícones/fontes tipo os dos contatos (usando as fontes awesome (<http://fontawesome.io/icons/>)). Adicionando links e uma pequena descrição. Ao invés de usar menus tradicionais usarei estas novas estruturas para cada área de conteúdo do meu site. Interessante é que o arquivo font-awesome.min.css é importado no início do main.css usando: `@import url(font-awesome.min.css);`

Aí também importa fontes:

```
@import url("http://fonts.googleapis.com/css?family=Source+Sans+Pro:200");
```

Moverei os ícones de contato para o rodapé.

## O que trocar?

### Foto

A foto por uma minha.

Reduzirei um pouco. Isso será feito reduzindo a foto em 70% e no assets/css/main.css:

```
#header .avatar img {  
    border-radius: 100%;  
    display: block;  
    width: 7em; /* Originalmente era 10 */  
}
```

Posso deixar a foto no tamanho original, que talvez tenha uma melhor qualidade e reduzir apenas no CSS.

### Frase de Boas Vindas

Olá!

<br>

Seja sinceramente muito bem vindo(a)!

<br>

Fique inteiramente à vontade e escolha um dos assuntos abaixo para tem acesso.

Qualquer crítica ou sugestão, por favor, me contacte.

### Scripts

Existia includes de 4 scripts js no rodapé que removi, pois não eram utilizados.

### As fotos do portfolio por ícones/fontes, com links e pequenas descrições.

E troquei o código do portfólio por um para ícones/fontes para cada um dos antigos itens de menu do site antigo.

Para isso criei uma grid usando o SimpleGrid. Apenas copieei o código que usei no site criado anteriormente, mas não ficava como eu queria. Não ficava como antes. Ficava com os ícones brancos e não ficava centralizado. Centralizei nas divs das duas linhas e pesquisando sobre a cor encontrei novamente no “stackoverflow” adicionei estilo em cada linha para mudar a cor.

## SimpleGrid

Uma explicação rápida sobre a SimpleGrid.

Ela divide a tela em 12 colunas.

Se quero criar duas células com tamanho da metade da tela cada, então uso:

```
<div class="grid grid-pad">
  <div class="col-6-12">
    <div class="content">
      <h3>Conteúdo</h3>
    </div>
  </div>
  <div class="col-6-12">
    <div class="content">
      <h3>Conteúdo</h3>
    </div>
  </div>
</div>
```

Se quero 3 células, sendo uma com 2, outra com 7 e a última com 3:

```
<div class="grid grid-pad">
  <div class="col-3-12">
    <div class="content">
      <h3>Conteúdo</h3>
    </div>
  </div>
  <div class="col-10-12">
    <div class="content">
      <h3>Conteúdo</h3>
    </div>
  </div>
  <div class="col-3-12">
    <div class="content">
      <h3>Conteúdo</h3>
    </div>
  </div>
</div>
```

Então a sintaxe é:

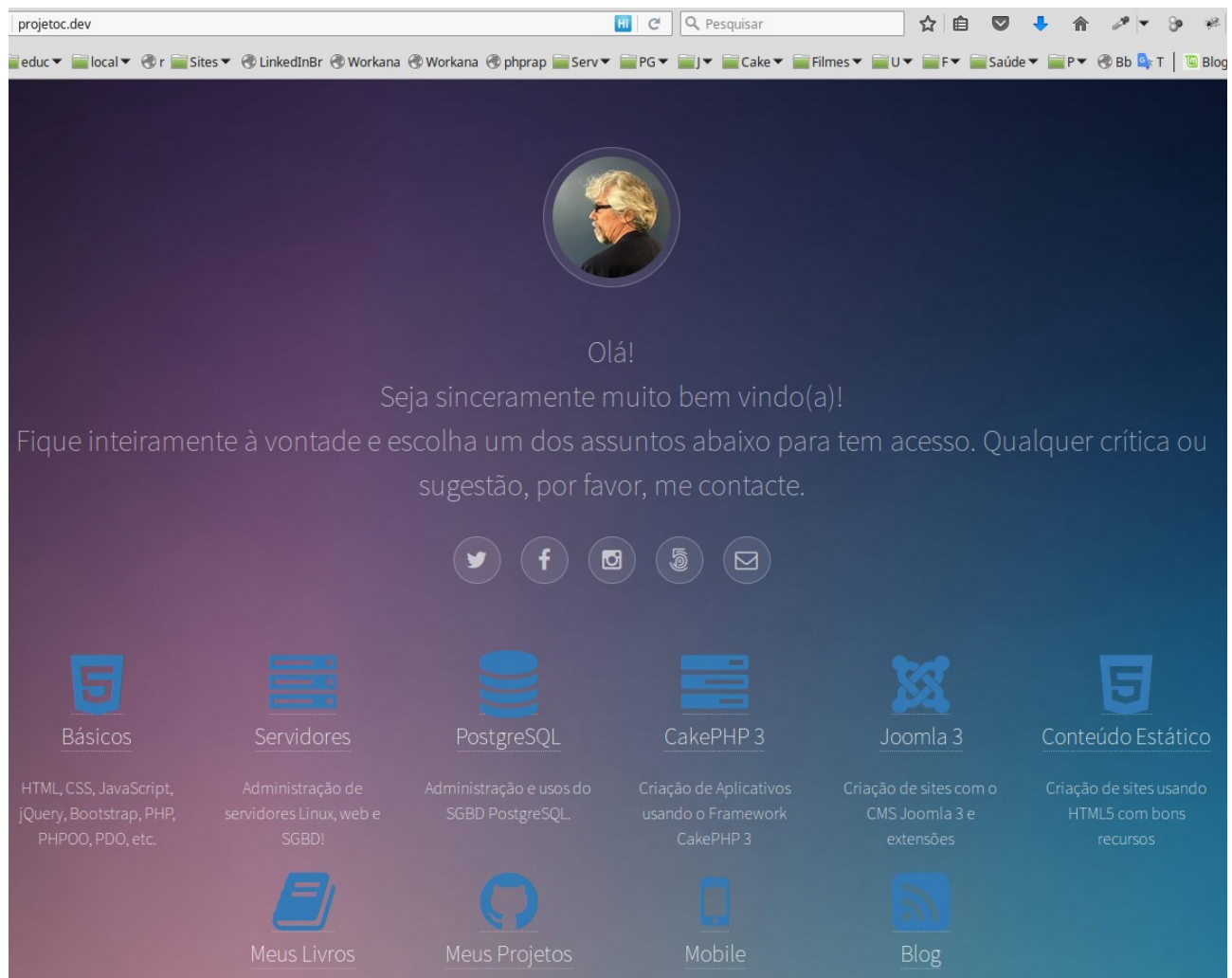
class="col-x-12"

x de 12, sendo x variando de 1 a 12.

Veja como centralizei a index com esta grid. Na segunda linha inseri uma coluna com apenas 1 no início.

Usar a SimpleGrid ao invés do Bootstrap, trouxe uma vantagem extra, que foi procurar mudar a cor dos ícones. Acabe por mudar para uma cor que combine melhor com o template.

Veja agora:



Removi as fotos existentes da pasta images, deixando apenas bg.jpg, que é usada no background e trocando avatar pela minha foto.

A bg.jpg é usada no main.css, como background-image do body.

Sem o bootstrap os ícones/fontes não ficaram azuis, mas brancos. Então fiz uma rápida pesquisa e adicionei ao final da tag `<i>`:  
`style="color:#337AB7"`

E ficou igual, pois capturei a cor usando a excelente extensão ColorZilla para o Firefox.

## Avatar

Subi mais a imagem avatar.  
Para isso alterei o main.css

```
#header {  
    padding: 1em 0 2em 0; /* Era 4 onde tem 1.  
}
```

Como adiviquei? Fui procurando top, padding e parecidos. Depois voltei para 4, pois achei melhor.

## Seção de Ícones do Conteúdo

O layout desta seção foi criado pesquisando pela grid do bootgrap. Agora adaptado para o SimpleGrid.

Criei duas linhas, sendo a primeira com 6 colunas de largura 2 e a segunda com apenas 5.

Observe que adicionei links também nas imagens, para ser mais amigável.

## Contatos

Movi os contatos para a região do rodapé, pois tudo indica que as pessoas chegam no meu site, mais pelo conteúdo que para me contatar. Pensando assim deveria mover a foto para abaixo dos contatos, mas vou deixar acima por questão de estética.

Adicionei alguns `<br>` para espaçar mais as seções.

## Enxugar CSS

O arquivo main.css tem muita coisa não utilizada no site e irei remover agora.

Apenas citarei as seções (estão comentadas):

- grid (enorme)
- form
- box
- image
- list (boa parte)
- table
- buttom
- thumbnails
- Poptrox

E mais algumas pequenas remoções, sempre tendo o cuidado de atualizar o site e verificar se não fez falta.

Deixar isso gera dois problemas: gordura desnecessária (que degrada o desempenho) e pode entrar em conflito com novos CSS futuramente adicionados.

O arquivo original tinha 43,9 kb e ficou com apenas 7kb. Isso significa problemas a menos, mesmo que para isso eu tenha gastado algum tempo, mas que não foi grande.

Os arquivos de CSS do bootstrap e o de fontes não irei otimizar. Poderia também, para isso usando o fonte e não o mínimo. Ao final compacta-se e usa-se a versão mínima no site.

## Excluir JavaScripts

Nenhum dos scripts em javascript que acompanham o template Visualize são usados, então eu os removi inclusive seus includes da index.html.

Somente a jquery foimantida mas não na index, somente nas outras páginas por conta do método load.

## Adicionar Conteúdo

Agora é a hora de adicionar todo o meu conteúdo e customizar para que fique com o mesmo template.

## Import do HTML

Criarei um arquivo cabecalho.html para cada área de conteúdo contendo o head, juntamente com os includes de CSS e também com o menu da respectiva seção.

Também usarei um pequeno script em JavaScript que usa o método load da jQuery que será adicionado ao início de cada página com conteúdo para importar o cabecalho.html da respectiva seção.

Assim:

- script com o load para o cabecalho.html
- arquivo html com o conteúdo

```
<script src="/assets/js/jquery.min.js"></script>
<script>
  $(function(){
    $('#cabecalho').load("./cabecalho_basicos.html");
  });
</script>
<div id="cabecalho"></div>
```

## Adicionar a Primeira seção

Adicionar Conhecimentos básicos.

Para isso crio uma sub pasta chamada **basicos**.

Dentro dela crio um arquivo chamado cabecalho.html e outro chamado index.html.

## Cada nova página de conteúdo é composta de

cabecalho.html

conteudo.html (este arquivo em o script com load no início)

No cabecalho.html entro com o head, os includes css e o menu.

No index.html entro no início o script que importará o cabecalho.html e abaixo entro o contrúdo propriamente dito.

Quando fui inserir um ícone/fonte do HTML5 na index de Conhecimentos Básicos, ele não apareceu. Virei, mexi e suspeitei que o menu.css o estava escondendo. Então mudei o include do menu.css para cima do bootstrap e apareceu. Precisamos ficar atentos com isto.



## Menu DropDown Multi Nível e Fixo na Margem Lateral Esquerda

Foi ótimo, pois as páginas de conteúdo, para facilitar a leitura eu reduzi sua largura, ficando um espaço livre à esquerda e à direita, pois deixei centralizado (usei 'margin 0 auto;' no body).

Gostei deste menu pois é muito simples. O CSS altera o comportamento das tags e não preciso ficar inserindo classes nem ids, apenas uso as tags ul e li. A estrutura do menu fica bem mais simples que outras soluções, mesmo que a do Bootstrap.

## Adicionando Conteúdo

Para adicionar o conteúdo faço assim: pegando como exemplo a seção **Conhecimentos Básicos** do meu site:

- Criei o sub diretório **basicos**;
- Criei os arquivos **cabecalho\_basicos.html** e o **index.html**. No cabecalho coloco o head e o menu. No index.html coloco um texto introdutório sobre o assunto.
- Esta seção é sobre HTML, CSS, JavaScript, jQuery, etc.
- Então se quiser falar pouco sobre cada um destes assuntos eu apenas crio um arquivo .html sobre cada um e o chamo no menu. Se quiser falar muito eu crio um sub diretório para cada um e dentro dele coloco arquivos para cada assunto.

## Ferramentas

### Editor HTML FCKEditor

Supondo que queira criar um arquivo chamado xdk.html.

Abro o FCKEditor para criar o conteúdo. É um editor muito bom, onde formato fontes, com tamanho, negrito, itálico, alinho texto, insiro imagens, etc.

Quando o conteúdo está pronto clico no botão Código-Fonte, seleciono tudo, copio, então abro o gedit e colo todo o conteúdo e salvo como xdk.html.

Depois disso abro o xdk.html no gedit e adiciono em seu início o código do script com o método load, citado acima.

Salvo e está pronto meu arquivo xdk.html.

### Criando um Cabeçalho

Para criar um arquivo de cabeçalho uso apenas um editor simples como o gedit, pois ele contém apenas código HTML, CSS e JavaScript.

## Conflito de CSS

Quando usamos vários arquivos de CSS precisamos ficar atentos pois pode ser que exista código conflitante entre eles.

Um exemplo é o menu.css e o main.css que têm bastante código comum. Vários trechos sobrescrevem o comportamento de tags do HTML.

Especificamente a tag <ul> estava com comportamento conflitante nos dois. No menu eu queria reduzir a margem esquerda entre o submenu e o menu, mas não encontrava onde.

Então usando o Firebug percebi que o main.css também sobrescrevia a tag <ul>. Então fui ao main.css e comentei apenas as duas linhas comentadas abaixo:

```
ul {  
    list-style: disc;  
    /*margin: 0 0 2em 0;  
    padding-left: 1em;*/  
}
```

E reduziu a margem de forma que me agradou.

Também posso deixar sem nenhuma margem na esquerda do sub menu, mas preferi deixar assim.

Se eu removesse todo o trecho \* List \*/ atingia a index mostrando os ícones de contato na vertical e não na horizontal, como está. Estão saindo rastreando até chegar nas duas linhas.

Isso num projeto grande deve dar bastante trabalho, por isso muitas vezes deixa-se como está, mas por outro lado mostra que realmente precisamos conhecer tudo que estamos usando e que seja o mais simples possível.

## Ajustando o CSS

Quiz remover uma linha que havia acima dos ícones de contato na index, pois após adicionar o download achei que não ficava bem.

Procurei na seção icons do main.css mas não encontrei. Então usando o Firebug, maravilhoso Firebug, ele me mostra que fica na tag <footer>. Aí ficou fácil, bastou comentar a linha:

```
/*border-top: solid 1px rgba(255, 255, 255, 0.25);*/
```

E a linha desapareceu.

Gosto de comentar e deixar lá, ao invés de remover, pois futuramente posso querer de volta.

## Solução de ToolTip

Encontrei uma solução de tooltip simples e sem conflito. Já havia testado várias outras sem sucesso. Geralmente escondiam o elemento onde estava aplicando. Esta tem uma desvantagem, não esconde o atributo title do HTML.

Usa somente CSS e encontrei em

[http://www.w3schools.com/css/css\\_tooltip.asp](http://www.w3schools.com/css/css_tooltip.asp)

Apliquei no ícone de download da index.

Um problema: se aplicar sobre um H2 ele fica grande. Corrigi fixando o tamanho da fonte somente no evento hover. Aumentei a margin-left.

## **CSS Reset**

Alguns elementos do HTML já tem um valor de margin, padding, borda e outros tipos de formatação definidos como padrão. O que acontece é que esses valores pré-definidos são necessários para que quando o site seja visto sem CSS algum, o usuário conseguirá ter um mínimo de legibilidade na visita.

Quando você vai implementar o CSS, esses valores atrapalham um bocado. Por isso, usamos essa técnica para zerar todas esses valores pré-definidos e inserir os valores que realmente usaremos para reproduzir o layout.

Quando não especificamos estilo para alguns elementos do HTML, cada navegador que visualize nosso site usará para estes elementos seus valores default e isso pode levar a aparência e comportamento inesperados. O reset cuida disso, de definir estilo para alguns elementos complicados do HTML, para evitar isso.

Estou usando o Reset do Eric Meyer - <http://meyerweb.com/eric/>

## **Problema com o método Load da jQuery**

Sempre que abria uma das páginas diferente da index e que efetuava refresh, acontecia uma grande piscada e dava até para ver a página sem CSS. Procurei muito para resolver isso até que comecei a ler o livro sobre jQuery da Succinctly e logo no início me apareceu a solução, que nem tem a ver especificamente com a jQuery, mas com o JavaScript em geral. Todo o CSS deve ser incluído antes de incluir a jQuery.

Isso resolveu o problema e agora a estrutura do site está semelhante a qualquer site.

## **Cuidado**

Sei que se começar a implementar novos recursos corro o risco de criar uma estrutura complexa tanto quando um dos atuais CMS, mas como trabalho sozinho e meus conhecimentos (tenho consciência) não são tantos quanto os de elementos das equipes dos atuais CMS, então não criarei estruturas complexas nem que queira. Outro detalhe importante é que mantere o objetivo de manter tudo isso simples e com grande controle.

## **Comentário para o Site**

Como temos um site estático, fica difícil criar um sistema de comentários para o mesmo. Então adicionei um ótimo sistema gratuito, que é o Disqus:

<https://disqus.com/>

Removi os comentários do Discus do site (ficava lento e não combinava com o template) e adicionei outro link para as redes sociais - <https://disqus.com/home/channel/simplecmsstatic/>

## **WebMail**

Também precisamos usar um bom serviço de webmail.

Com esta onda de invasão de privacidade optei pelo Tutanota, que criptografa todos os e-mails e me permite adicionar senha aos e-mails importantes, de forma que quem recebe somente abrirá se souber uma senha combinada com antecedência ou depois.

Ele oferece 1GB de espaço para a conta free.

<https://tutanota.com/>

## **Botões de Redes Sociais**

Adicionei uma barra de botões das redes sociais e o e-mail. Usei o mais simples que encontrei.

Praticamente os ícones com os links. Este foi o escolhido:

<https://simplesharingbuttons.com/#code>

## **Adição de menu de topo**

Adicionei um menu no topo e esperei as regiões para que ao clicar em um item de menu apareça na tela somente a respectiva região.

## **Centralização do Menu**

Deu muito trabalho centralizar o menu de topo. Tive que fazer uma pesquisa e achei aqui:

<http://stackoverflow.com/questions/2005954/center-a-positionfixed-element>

## **Melhora dos ícones das Redes Sociais**

Procurei ícones que combinem mais com o tema. E ajustei melhor o CSS que mostra somente uma seção de cada vez.

## **Adição de um Novo Template**

Criei uma sub pasta chamada templates e adicionei o template Delicious da Themeshouter.com.

Deu algum trabalho, mas depois da experiência com o primeiro este foi mais simples.

## **Alteração do nome do Projeto**

O projeto agora chama-se **static-cms-simple**, pois cms-static é uma categoria, seu nome é simple.

## Adicionar um Forum para o site no Slack

Criei um canal no Slack:

<https://ribafs.slack.com/messages/simple-static-cms/>

Vou adicionar um link no Site, nas Redes Sociais.

## Busca via DuckduckGo

Cheguei a adicionar busca no site para o DuckDuck mas removi, pois achei que não ajudaria, pois somente procura na index.

```
<iframe src="https://duckduckgo.com/search.html?
duck=yes&site=ribafs.org&prefill=Busca na web com privacidade"
style="overflow:hidden;margin:0;padding:0;width:483px;height:60px;"
frameborder="0"></iframe>
```

## Boas Referências Gratuitas

- <https://www.caelum.com.br/apostila-html-css-javascript/> (Muito boa e em português)
- <https://www.syncfusion.com/resources/techportal/ebooks> Muita coisa boa e free (inglês):  
[https://www.syncfusion.com/resources/techportal/details/ebooks/GitHub\\_Succinctly](https://www.syncfusion.com/resources/techportal/details/ebooks/GitHub_Succinctly)  
<https://www.syncfusion.com/resources/techportal/details/ebooks/css3>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/twitterbootstrap3>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/linux>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/regexexpressions>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/postgres>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/javascript>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/http>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/git>  
<https://www.syncfusion.com/resources/techportal/details/ebooks/jquery>

## Forums

<http://forum.imasters.com.br/>

## Troca do Menu

Como após mexer por por algum tempo eu não tive facilidade de alterar o menu tipo iOS, então o o troquei por um mais fácil. Estimulado por um aviso de aumento de hospedagem.

Troquei o menu tipo iOS pelo superior central da w3c, especificamente por este:

[https://www.w3schools.com/css/tryit.asp?filename=trycss\\_navbar\\_horizontal\\_dividers](https://www.w3schools.com/css/tryit.asp?filename=trycss_navbar_horizontal_dividers)

É fácil perceber como fiz as customizações, olhando no CSS atual.

## Redes Sociais

Não gostei do design dos ícones das redes sociais, então substitui por ícones de fontes Font Awesome, que são ótimos. Apenas copiei do conteúdo e sobrescrevi e adaptei.

## Correção de link nos arquivos de Menu.html

Correção de todos os menu.html, removendo #conteudo do href="/index.html#conteudo" para href="/index.html"

## Permissão para user default no /var/www/html

Para quem usa o apache2 em todo o /var/www e para facilitar e o usuário default possa ter permissão de escrita ao /var/www:

```
sudo nano /usr/local/bin/perms
```

```
#!/bin/sh
clear;
echo "Aguarde enquanto configuro as permissões do /var/www/$1";
echo "";
chown -R www-data:webdevel /var/www/$1;
chmod -R g+s /var/www/$1;
find /var/www/$1 -type d -exec chmod 2775 {} \;
find /var/www/$1 -type f -exec chmod 2664 {} \;
echo "";
echo "Concluído!";
```

```
Executei
chmod 700 /usr/local/bin/perms
```

```
chmod +x /usr/local/bin/perms
```

```
Para executar Executar
sudo perms
```

## Melhor Editor

A meu ver o melhor editor para trabalhar aqui é o FKEditor online  
<https://ribafs.github.io/html-editor/samples/index.html>

Idealmente online, que funciona melhor. Faça o download e use em qualquer diretório.

- Cola ou didiga
- Clica no botão Source
- Seleciona todo o texto digitado ou colado
- Tecla Ctrl+C
- Vai até o editor com o artigo
- Clica onde será inserido

- Tecla Ctrl+V

Ele cola com todo o código fonte HTML entrado no editor.  
Efetuar as correções desejadas.

Na criação ou edição pode usar as taclas de atalho:

Ctrl+B

Ctrl+I

Ctrl+U

Ctrl+L

Ctrl+R

Ctrl+J

Ctrl+C

Ctrl+V

Para negrito, itálico, etc.

## **Colando de processadores como LibreOffice**

Cole no FCKEditor, edite e cole alterado, especialmente os espaçamentos verticais.

## **Customizando o Reset**

O main.css contém um trecho de reset, onde ele altera e anula o comportamento normal de algumas tags. Tive dificuldade foi a strong e outras. Gosto de usar para dar ênfase à algumas palavras e tive que remover do reset. Bastou remover strong, h1, h2 e h3 do início do main.css. Caso tenha dificuldade com outras pode fazer parecido.

## **Páginas Básicas do Simple CMS**

O Simple CMS trabalha com duas páginas básicas:

- a index.html da entrada. Esta contém basicamente o css embutido e o menu também.
- a index.html de cada seção. Esta usa o menu.css e o menu.css.

O primeiro é a index do CMS e o segundo é usado em todas as seções do site.

Exemplo:

- Index.html (menu central, conteúdo com submenus)
- index.html, menu.html incluído via jquery (conteúdo com menu incluído)

## **Menu Central da index geral**

Como apresentou problema eu troquei por outro.

Estamos agora usando este:

<http://cssdeck.com/labs/centering-horizontal-menu-navigation-or-tabs-lists-with-css>

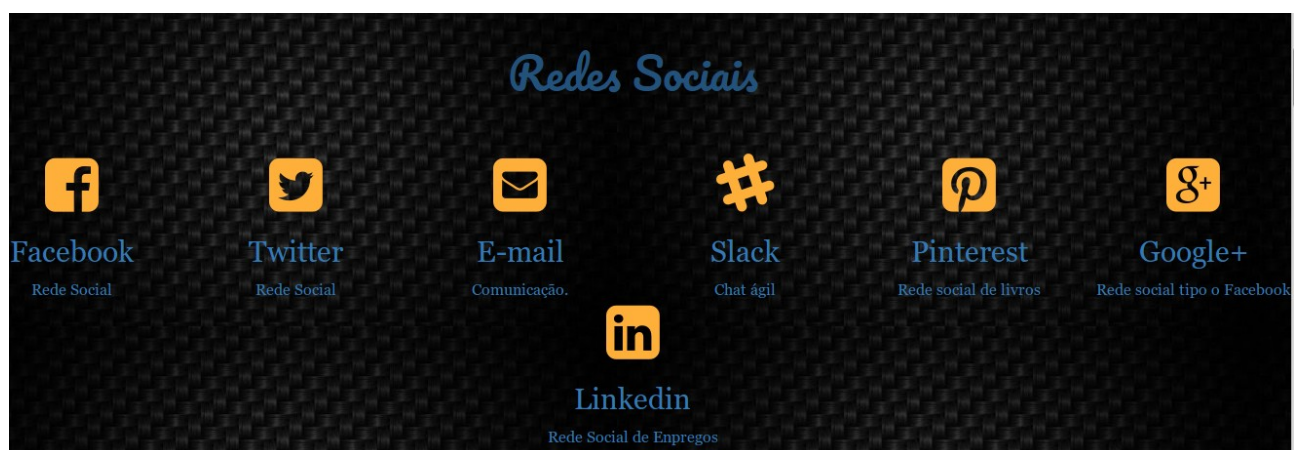
Aproveitei e fiz uma adaptações.

## Capturas

Veja como está a index agora (menu central):



## Redes Sociais





## Conteúdo



## Sobre



## Rodapé



## Estrutura Lógica do Simple CMS

Vejamos agora, depois das capturas, a estrutura lógica do nosso arrogante CMS.

### Seções da index principal

1 - <head>

Nesta seção include-se alguns css:

```
<link rel="stylesheet" type="text/css" href="assets/css/bootstrap.min.css" rel="stylesheet">
<link rel="stylesheet" type="text/css" href="assets/css/main.css">
<link rel="stylesheet" type="text/css" href="assets/css/sociais.css" />
```

2 – Primeira seção de CSS

```
<style>
```

```
/* MENU Central */
```

3 – Segunda seção de CSS

```
<style>
```

```
/* Seções do Site */
```

4 -

```
<body>
```

```
<!-- Menu html →
```

5 – Seção da imagem

```
<!--banner AQUI →
```

```
<section id="banner">
```

6 – Seção de ícones sociais

```
<section id="sociais">
```

7 – Seção de menu conteúdos

```
<section id="conteudo">
```

8 – Seção do ícone de download

```
<section id="download">
```

9 – Seção do ícone sobre mim

```
<section id="sobre">
```

10 – Seção do rodapé

```
<footer id="rodape">
```

## Na segunda seção, Menu Seção

```
background: #191717;  
/* background: url(./images/bg-menu.png);*/  
}
```

Deixei um fundo extra para o menu, basta descomentar e coentar a linha anterior.  
Já deixei a imagem respectiva.

## Index das sub seções

### 1 – Seção <head>

Aqui são incluídos alguns css:

```
<link rel="stylesheet" href="/assets/css/menu.css" />  
<link rel="stylesheet" href="/assets/css/main.css" />
```

### 2 – Seção de include do menu pelo jquery:

```
<body>
```

```
<br>
```

```
<!-- Include do menu.html com jquery →
```

### 3 - <!-- Início do conteúdo-->

```
<!-- Início do conteudo -->
```

```
<div id='conteudo'>
```

### 4 - <!-- Final de conteudo -->

```
<!-- Rodapé-->
```

```
<footer id="rodape">
```

## Menu das sub seções

### 1 – Início do menu

```
<nav class="vertical">
```

### 2 – Todo o conteúdo do menu e finaliza com:

```
</nav>
```

```
<!-- Final do Cabeçalho →
```

## **Aplicando o Static CMC ao meu site pessoal RibaFS Portal, como caso de uso**

Pelo menos em parte.

### **Alerta**

Ao usar uma estrutura como o Simple CMS para criar sites precisamos ter em conta de que muitos recursos dos grandes CMS não estarão à nossa disposição. Então temos que ter em conta que não teremos o melhor dos dois mundos. Eu tenho em conta que perderei a agilidade do editor do CMS.

Então usarei muito os PDF, mas em contrapartida usarei e abusarei dos menus múltiplos das seções internas para abrigar programas, como o Layout de CSS.

Precisamos reconhecer que esta estrutura, apesar de simples, exige mais que o usual.

Acho muito importante, especialmente nós programadores, mantermos nossos cérebros bem ativos e não somente para nossa profissão, para a vida e para todos em geral.

E mais um fator importante, estamos a dada momento lidando com conhecimentos da nossa área.

### **Material de Criação do Site**

- Podemos usar HTML nos arquivos básicos
- Arquivos linkando:
  - PDF
  - HTML
- Outras formas

Aliados fortes são o LibreOffice writer e o FCKEditor, osados offline e também o Document HTML do LibreOffice, que é um criador de páginas HTML.

### **User Friendly**

Pensando no usuário sempre. Ao voltar de um menu como CSS ele volta para o nível imediatamente anterior, no caso Básicos.

### **Abrigar um domínio em um repositório do GitHub**

- Crie uma conta no GitHub
- Crie um novo repositório
- Configurar domínio
- Requer um CNAME em seu DNS
- Crie um arquivo no raiz do seu repositório:  
CNAME

Veja detalhes no GitHub

**GitHub Pages**

Your site is ready to be published at <https://ribafs.github.io/ribafs-portal/>.

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

**Source**

Your GitHub Pages site is currently being built from the `master` branch. [Learn more.](#)

master branch ▾

Save

**Theme chooser**

Select a theme to build your site with a Jekyll theme. [Learn more.](#)

Choose a theme

**Custom domain**

Custom domains allow you to serve your site from a domain other than `ribafs.github.io`. [Learn more.](#)

Save

☒ **Enforce HTTPS** — Required for your site because you are using the default domain (`ribafs.github.io`)  
HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site.  
When HTTPS is enforced, your site will only be served over HTTPS. [Learn more.](#)

Em Source se selecionar None desaparece o link para o site e o Custom domain. Mudei de None para master branch. Reiniciou a página e ficou assim.

Agora basta criar um CNAME vazio no raiz do repositório e atualizar.

## Firebug

O Firebug é uma ferramenta muito útil.

É bom frisar isso, especialmente a ferramenta Inspetor, a seta azul da esquerda, vizinho ao bug.

## Menos Capricho

Observo agora que os arquivos de conteúdo em HTML não capricho tanto, não uso o CSS. Para ficar melhor é melhor também importar o início do index.html. Vou ver.

Realmente, dá outra vida. Bastou introduzir o começo da index.html.

Além de estética é prática, pois mantém o arquivo com o menu.

**A cada arquivo novo HTML que não faça parte da estrutura, inciar com :**

```
<!DOCTYPE HTML>
<!-- Visualize by TEMPLATED   templated.co @templatedco Released for free under the Creative Commons Attribution 3.0 license (templated.co/license) -->
<html>
  <head>
    <title>RibaFS - Uma Grande Li&ccedil;&atilde;o de Vida</title>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1" />

    <link rel="stylesheet" href="/assets/css/menu.css" />
    <link rel="stylesheet" href="/assets/css/main.css" />
  </head>
  <body>
    <br>
    <script src="/assets/js/jquery.min.js"></script>
    <script>
      $(function(){
        $('#menu').load("./menu.html");
      });
    </script>
    <div id="menu"></div>
    <!-- Final do Cabeçalho -->

    <!-- Início do conteúdo-->
    <meta charset="UTF-8" >
    <style>
    #conteudo{
      padding: 10px;
      font-family: 'Arial', Helvetica, sans-serif;
      margin: 0 auto;
      width: 800px;
      padding-bottom: 50px;
      background-color: #fff;
      color: #000;
    }
    #rodape{
      margin-top: 400px;
    }
    </style>

    <!-- Início de conteudo -->
    <div id='conteudo'>
    <h2 align="center" class="titulo">Título</h2>
```