



TEMAS

Roteiro de Atualização

São José dos Campos

Julho, 2014

Av. Dr. João Batista Soares Queiroz Jr., 1120 - sala 03, Jd. das Indústrias - São José dos Campos - SP - CEP 12.240-000
Telefones (12) 3302-5573 / (12) 9 9604-5004

www.webcartorios.com.br

contato@webcartorios.com.br



SUMÁRIO

1. Sobre	1
2. Ferramentas	1
2.1. UsO ObrigatóriO	1
2.1.1. HeidiSQL.....	1
2.1.2. FileZilla	1
2.1.3. PHP Designer 8	1
2.2. UsO Opcional	2
2.2.1. Google Web Fonts	2
2.2.2. Font Awesome	2
3. Arquivos Disponíveis	3
Diretório Raiz /www	3
Diretório /www/layout_nomeCliente	3
Diretórios Diversos	3
3.1. Para baixar da pasta __inc	3
4. Estrutura do Tema	4
4.1. Estrutura Base	4
4.2. Arquivos Padrões	4
4.3. Arquivos Personalizados	5
4.4. Tabelas do Banco de Dados.....	5
4.4.1. Prefixo das Tabelas	5
4.4.2. Tabelas para Alterar	5
5. Alterando e Adaptando um Tema	6
5.1. Mudanças Gerais.....	6
5.2. css.cores.php.....	6
5.3. FuncoesPersonalizadas.php	7
5.4. index.php	7
5.5. inc.topo.php	9



5.6.	inc.menu_topo.php	9
5.7.	inc.estrutura.php	9
5.8.	inc.home.php.....	10
5.9.	inc.titulo_modulo.php.....	10
5.10.	inc.rodape.php.....	11
5.11.	Instruções Extras	11
6.	Cartórios x Webconteúdos	12
6.1.	Banco de Dados.....	12
6.2.	Arquivos.....	12
6.3.	Substituição em massa.....	12
7.	Testes Preliminares.....	13



1. Sobre

Este documento descreve um roteiro a ser seguido para a **criação e atualização de temas**. Temas, normalmente, são adquiridos de terceiros, ou possuem sua base construída apenas com *HTML* e *CSS* – Sendo então necessário adaptá-los aos padrões do **Webcartórios**.

Esta apostila foi criada com o intuito de funcionar como um guia para os desenvolvedores, definindo ferramentas, estruturas padrões, correções obrigatórias, entre outros.

2. Ferramentas

Este capítulo lista as ferramentas utilizadas para o desenvolvimento de um tema, abordando softwares e serviços diversos.

- Uma ferramenta de **uso obrigatório** deve, por padrão, ser utilizada – apenas um Analista ou Coordenador poderá indicar o uso de outro software para substituição.
- Uma ferramenta de **uso opcional** deve ser utilizada caso se adapte ao tema sendo desenvolvido, sendo assim especificado na análise do serviço.

2.1. Uso Obrigatório

2.1.1. HeidiSQL

Sistema de gerenciamento de banco de dados (SGBD) livre, *open-source*, capaz de manipular databases criadas a partir da linguagem SQL. A ferramenta oferece suporte as principais plataformas, incluindo Linux, Windows e MAC OS, além de apresentar alto desempenho e estabilidade.

2.1.2. FileZilla

FileZilla é um cliente FTP, SFTP e FTPS de código livre para Microsoft Windows e GNU/Linux. É distribuído em licença GNU *General Public License*.

2.1.3. PHP Designer 8

Ambiente de desenvolvimento integrado (IDE) e editor PHP, além de também funcionar como editor HTML (5), CSS (3) e JavaScript. Fornece suporte à programação orientada a objetos e à utilização de frameworks.



2.2. Uso Opcional

2.2.1. Google Web Fonts

Serviço que disponibiliza uma coleção online de fontes para uso em websites, a fim de reduzir as restrições que um desenvolvedor ou designer enfrenta para escolher e manipular a tipografia de um projeto. Com o Google Web Fonts é possível importar fontes gravadas através de *cloud computing*, ou seja, a fonte utilizada não mais tem a necessidade de estar presente na máquina do usuário, pois com esse serviço pode ser compartilhada através de servidores webs.

2.2.2. Font Awesome

Fonte que agrega uma coleção de 439 ícones, incluindo ícones para redes sociais, player de vídeo, formulários, editores de texto, entre outro. Funciona como vetores (infinitamente escaláveis), dispensando o uso de JavaScript e permitindo total manipulação, através de CSS.

3. Arquivos Disponíveis

Esta apostila é acompanhada por um arquivo zipado, já com a hierarquia de pastas e arquivos básicos que compõem um tema. Estes são definidos no item **4. Estrutura do Tema.**

Diretório Raiz /www

- index.php ou index_nomeCliente.php

Diretório /www/layout_nomeCliente

- Funcoes_Personalizadas.php
- css.cores.php
- inc.estrutura.php
- inc.topo.php
- inc.menu_topo.php (Exemplos: inc.menu_topo_notas.php e inc.menu_topo_ri.php)
- inc.titulo_modulo.php
- inc.home.php
- inc.rodape.php

Diretórios Diversos

- layout_nomeCliente/css , layout_nomeCliente/js , layout_nomeCliente/img e layout_nomeCliente/fonts

Observações:

- **nomeCliente** deve ser alterado para o nome do cliente para qual o tema está sendo desenvolvido. **Exemplo:** layout_webcartorios
- O nome do **arquivo Index** varia de acordo com o local onde o tema está sendo inicialmente desenvolvido: Se é no ambiente de desenvolvimento do cliente, recebe o nome index.php; Se é em ftp compartilhado, recebe o nome index_nomeCliente.php.

3.1. Para baixar da pasta __inc

Obter a versão atualizada (baixar do FTP Webtemas, por exemplo) dos seguintes arquivos:

- cartorio.php **OU** webc.php
Utilizar apenas um dos dois. O arquivo **cartorio.php** é utilizado para o desenvolvimento de temas para cartório; O arquivo **config.php** é utilizado nos demais websites, como Webconteúdos (Exemplos: Eleven, ARPEN-SP, CNB-SP, entre outros).
- Funcoes.php
Antes de realizar o upload, verificar se estes arquivos já não existem no ambiente de desenvolvimento e se estão em sua versão mais recente.
- **Outros:** css.css , ui.all.css , colorbox.css , funcoes.js, jquery/ui/ui.core.js , jquery/jquery.meiomask.js, jquery/ui/ui.datepicker.js e jquery/jquery.colorbox.js .

4. Estrutura do Tema

Dado um tema, supondo que este, inicialmente, foi construído basicamente com HTML, CSS e demais scripts, deve-se adaptá-lo a estrutura dinâmica de um *website* desenvolvido pelo **Webcartórios**.

Para isso, é necessário incluir arquivos básicos de configuração e funções, além de separar, dividir o tema, de forma que diferentes blocos da página sejam controlados de forma autônoma. Assim, o código é separado em arquivos distintos para criação e gerenciamento do topo, menu, home, conteúdo, rodapé, entre outras estruturas.

4.1. Estrutura Base

ARQUIVO: __inc/cartorio.php OU __inc/config.php

Sobre: Arquivo principal com os dados de conexão ao banco de dados. Alterar apenas se solicitado por um Analista ou Coordenador.

ARQUIVO: __inc/Funcoes.php

Sobre: Contém funções, classes e outros objetos genéricos do projeto.

4.2. Arquivos Padrões

ARQUIVO: index.php OU index.nomeCliente.php

Sobre: Responsável pela exibição de todas as páginas do site

DIRETÓRIO: layout_nomeCliente

Sobre: Diretório de arquivos do tema. Inclusão de todos os javascripts, css, fontes e imagens.

ARQUIVO: layout_nomeCliente/Funcoes_Personalizadas.php

Sobre: Funções e outros itens personalizados do tema (caso necessário).

ARQUIVO: layout_nomeCliente/inc.estrutura.php

Sobre: Responsável pela verificação de módulos ativos e includes de arquivos. Caso o *website* esteja no index, carrega o arquivo inc.home.php.

ARQUIVO: layout_nomeCliente/css.cores.php

Sobre: Arquivo com CSS personalizados e cores especificadas na tabela mysql "cartorio_cores" ou "webc_cores". Verificar diferenças no item 4.3. Tabelas do Banco de Dados.

ARQUIVO: layout_nomeCliente/inc.topo.php

Sobre: Arquivo com a estrutura do topo do site.



ARQUIVO: layout_nomeCliente/inc.menu_topo.php

Sobre: Arquivo com a estrutura do menu principal do site. inc.menu_topo_notas.php apresenta um exemplo de menu para um cartório com a especialidade **Notas**; inc.menu_topo_ri.php apresenta um exemplo de menu para um cartório com a especialidade **Registro de Imóveis**.

ARQUIVO: layout_nomeCliente/inc.home.php

Sobre: Página inicial do site.

ARQUIVO: layout_nomeCliente/inc.titulo_modulo.php

Sobre: Página que inclui cabeçalho com nome do módulo em páginas internas.

ARQUIVO: layout_nomeCliente/inc.rodape.php

Sobre: Rodapé do tema.

4.3. Arquivos Personalizados

- Caso seja necessário desmembrar outros itens do tema, basta criá-los dentro do diretório layout_nomeCliente, no formato **inc.nome_elemento.php**.
- Arquivos **CSS** do tema devem ser incluídos no diretório **/css**.
- Arquivos **JavaScript** do tema devem ser incluídos no diretório **/js**.
- **Imagens** do tema devem ser incluídas no diretório **/img**.
- **Fontes** do tema devem ser incluídas no diretório **/fonts**.

4.4. Tabelas do Banco de Dados

4.4.1. Prefixo das Tabelas

O prefixo do nome das tabelas depende do tipo de projeto:

- **Cartórios:** Utilizar prefixo **cartorio_**.
- **Webconteúdos e Outros:** Utilizar prefixo **webc_**.

4.4.2. Tabelas para Alterar

- **cartorio_dados ou webc_dados:** Configurações gerais do projeto/informações do cliente e do projeto. A coluna Layout_Dir deve ser alterada para layout_nomeCliente (o mesmo nome que está sendo utilizado no diretório de arquivos do tema)
- **cartorio_cores ou webc_cores:** Cores do projeto, utilizadas pelas variáveis do arquivo css.cores.php.

5. Alterando e Adaptando um Tema

5.1. Mudanças Gerais

Em todos os arquivos, alterar no bloco de comentários inicial:

1. Data do campo "Início".
2. Incluir mais um campo "Programadores" e completar com seu nome.
3. Seguir o modelo para incluir ajustes, anotações, alterações feitas no arquivo.

```
/* index.php - Responsável pela exibição de todas as páginas do site */
/* Início: XX/XX/XXXX */
/* Versão 1.0 */
/* Projeto: DYNAMIC SYSTEM E VISH! Internet e Sistemas Ltda - ME */
/* Programadores: Cleiser Rian Lima */
/* Programadores: José Mário Pácio dos Santos */
/* Programadores: Alex Arantes Pedrosa de Lima */
/* Programadores: Daniel Jun Shimizu */
/* Documentação: Eduarda Oliveira Shimizu */
//
// Ajustes, alterações e versões
// XX/XX/XXXX - Nome_do_Developador - Anotar alterações
//
```

5.2. css.cores.php

Alguns elementos do tema devem ser editáveis através do banco de dados (tabela cartorio_cores para Cartórios ou webc_cores para Weconteúdo), a fim de tornar mais práticas e gerenciáveis as mudanças de cores em títulos, links, fundos, entre outros.

Para este fim, algumas classe e identificadores são definidos no arquivo css.cores.php, com seu valor obtido de variáveis que acessam os dados do banco. O arquivo base apresenta uma referência de como acessar o valor dessas variáveis:

```
body {
    background: <?= $cor->bg ?> !important;
    background-color: <?= $cor->cor_bg ?> !important;
    color: <?= $cor->cor_texto ?> !important;
}
```

Assim, como indicado no exemplo, um atributo CSS recebe o valor cadastrado no Banco de Dados, recebendo através do objeto \$cor->nome_var.

As variáveis (substituir em nome_var) disponíveis para uso são:

- **Geral:** bg, cor_bg, cor_texto.
- **Links:** cor_link, cor_link_hover.
- **Menu Principal:** cor_menu, cor_menu_hover, cor_menu_texto.
- **Menu Secundário:** cor_submenu, cor_submenu_hover, cor_submenu_texto_hover, cor_submenu_texto.
- **Módulos:** cor_titulo_modulos, cor_titulo_modulos_texto.
- **Itens:** cor_titulo_item, cor_subtitulo_item.
- **Notícias:** cor_titulo_noticias, cor_subtitulo_noticias, cor_corpo_noticias.
- **Outros:** cor_destaque.

5.3. FuncoesPersonalizadas.php

O arquivo base FuncoesPersonalizadas.php apresenta um exemplo de como documentar uma função. Ao incluir funções personalizadas para o tema sendo desenvolvido, este padrão deve ser seguido para todas as funções incluídas.

```
/**
 * @Funcao soma
 * @Autor Daniel J Shimizu
 * @CriadoEm 10/04/2014
 * @Parametro int ; num1
 * @Parametro int ; num2
 * @Retorno null
 * @Retorno int ; resultado
 * @Descricao Esta função realiza a soma entre dois elementos.
 */
function soma($num1 , $num2){
    die;
}
```

5.4. index.php

1. Caso o site desenvolvido não seja um cartório, no bloco **Config**, alterar o include do arquivo cartorio.php para config.php.

```
-----
// Bloco Config
// Inclui arquivo principal com os dados de conexão ao banco de dados
// ATENÇÃO! Caso o projeto seja WEBCONTEÚDO, substituir a inclusão do arquivo cartorio.php por:
// include(' _inc/config.php');
-----
include(' _inc/cartorio.php');
```

2. No bloco **Shortcut Icon** é possível definir o ícone (Favoritos) do *website*.

```
<?php
-----
// Bloco Shortcut ICON
// Definição do ícone de Favoritos / barra de endereço
-----
?>
<link rel="shortcut icon" href="" />
```

3. No bloco **CSS do Tema**, a partir dos exemplos, alterar o caminho (nome_do_arquivo.css) para os arquivos CSS do tema sendo desenvolvido.

```
<?php
//-----
// Bloco CSS do TEMA
//-----
?>
<link href="<?=$cliente->Layout_Dir?>/css/nome_do_arquivo1.css?<?php echo time(); ?>" rel="stylesheet" />
<link href="<?=$cliente->Layout_Dir?>/css/nome_do_arquivo2.css?<?php echo time(); ?>" rel="stylesheet" />
<link href="<?=$cliente->Layout_Dir?>/css/nome_do_arquivo3.css?<?php echo time(); ?>" rel="stylesheet" />
<?php // Fim CSS do tema ?>
```

4. No bloco **Javascript do Tema**, a partir dos exemplos, alterar o caminho (nome_do_arquivo.js) para os arquivos Javascript ou funções do tema sendo desenvolvido.

```
<?php
//-----
// Bloco Javascript do Tema
//-----
?>
<script type="text/javascript" src="<?=$cliente->Layout_Dir?>/js/nome_do_arquivo1.js"></script>
<script type="text/javascript" src="<?=$cliente->Layout_Dir?>/js/nome_do_arquivo2.js"></script>
<script type="text/javascript" src="<?=$cliente->Layout_Dir?>/js/nome_do_arquivo3.js"></script>
<?php // FIM Javascript do Tema ?>
```

5. No bloco **Javascript Webcartórios** a versão do Jquery deve ser alterada APENAS no caso de conflito com o tema/plugins personalizados utilizados no tema. *Antes de alterar, informar o Analista ou Coordenador do Projeto.*

```
<?php
//-----
// Bloco Javascript Webcartórios
// A versão do Jquery deve ser alterada APENAS no caso de conflito com o tema/plugins
// personalizados utilizados
//-----
?>
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.min.js"></script>
```

6. A codificação do template deve ser inserida após o bloco **Inclusão do Tema**. O tema deve ser recortado e dividido em seções, respeitando a inclusão dos quatro arquivos básicos:

- inc.topo.php
- inc.menu_topo.php
- inc.estrutura.php*
- inc.rodape.php

* O arquivo **inc.estrutura.php** agrega o corpo principal do tema e varia, caso a visualização seja da home, ou da página interna. Ou seja, inc.topo.php, inc.menu_topo.php e inc.rodape.php incluem estruturas visíveis em **todas** as páginas do website. Caso seja necessário, *é possível incluir arquivos extras de recorte.*

5.5. inc.topo.php

O arquivo inc.topo.php deve incluir os blocos de conteúdo a serem exibidos no topo do *website*. No arquivo base há apenas uma referência para inclusão do código, ou seja, o mesmo deve ser adaptado ao tema sendo desenvolvido. Poderá conter um link para Login de usuário.

```
<?php //Blocos TOPO ?>
<div class="topo"></div>
```

5.6. inc.menu_topo.php

O arquivo inc.menu_topo.php deve incluir os blocos de conteúdo a serem exibidos no menu superior do *website*. No arquivo base há apenas uma referência para inclusão do código, ou seja, o mesmo deve ser adaptado ao tema sendo desenvolvido e a especialidade (notas, protesto, RI, ou personalizado para Webconteúdos).

```
<?php //Bloco Menu Superior ?>
<div class="menu_topo">
  <ul class="menu_sup">
    <li>Item</li>
    <li>Item</li>
    <li>Item</li>
    <li>Item</li>
    <li>Item</li>
  </ul>
</div>
```

Para incluir um **Menu Reorganizável**, a documentação referente a este deve ser consultada.

5.7. inc.estrutura.php

1. O bloco Homepage controla seções do tema que só serão visíveis na home (Index), como um sldideshow de destaques e notícias. Por padrão, o arquivo base já contém a inclusão do arquivo **inc.home.php**. Dependendo do tema, poderá ser necessário incluir mais arquivos. A "regra" é incluir todos os blocos que devem ser exibido *apenas na homepage*.

```
//-----
// Bloco Homepage
// Inclui os arquivos que devem ser exibidos na homepage (index) do website
//-----
if ($pagina == ''){

  // Caso não tenha nenhuma página sendo chamada ele entra na home.
  // Dependendo do tema, poderá ser necessário incluir arquivos além do inc.home.php
  // A "regra" é incluir todos os com blocos que devem ser exibidos APENAS na homepage
  include("inc.home.php");

}

// Fim Homepage
```

2. O bloco **Título do Módulo** é utilizado para incluir um bloco de cabeçalho no início de todas as páginas de conteúdo, com codificação definida no arquivo **inc.titulo_modulos.php**. Seu uso é opcional, isto é, utilize caso se adapte ao tema sendo desenvolvido. Caso contrário, basta remover o bloco.

```
//-----  
// Bloco Título do Módulo  
// Inclui um cabeçalho das páginas  
// Remover se não se adaptar ao tema  
//-----  
include("inc.titulos_modulos.php");
```

3. O bloco **Conteúdo das Páginas** determina o corpo das páginas internas. Sendo assim, deve-se adaptar o bloco ao tema sendo utilizado. No arquivo de exemplo há apenas uma referência, onde o bloco é delimitado pela div `<div class="bloco-conteudo">...</div>`.

```
//-----  
// Bloco de Conteúdo das Páginas (módulos)  
// Adaptar de acordo com o tema sendo utilizado  
//-----  
echo '<div class="bloco-conteudo">';  
// Título do módulo  
echo '<div class="titulo-conteudo">'.$titulo_modulo.'</div>';
```

5.8. inc.home.php

O arquivo inc.home.php deve incluir os blocos de conteúdo a serem exibidos apenas na homepage. No arquivo base há apenas uma referência para inclusão do código, ou seja, o mesmo deve ser adaptado ao tema sendo desenvolvido.

```
<?php //Blocos da HOME ?>  
<div class="conteudo-home">  
    <div class="slide_destaque"></div>  
</div>
```

5.9. inc.titulo_modulo.php

1. O arquivo inc.titulo_modulo.php define um cabeçalho no início de todas as páginas de conteúdo. O código do arquivo base é apenas uma referência, apresentando algumas variáveis que podem ser úteis neste bloco. Sendo assim, o layout deve ser adaptado de acordo com o tema sendo desenvolvido.

```
<?php // Adaptar de acordo com o tema ?>
<div>
  <?php
    echo $titulo_modulo;
  ?><br />
  <?php
    $texto_help = $_SESSION['HELP_MODULO'];
    Help('', $texto_help);
  ?><br />
  <?php
    echo $texto_introducao;
  ?><br />
</div>
```

5.10. inc.rodape.php

O arquivo inc.rodape.php deve incluir os blocos de conteúdo a serem exibidos no rodapé do *website*. No arquivo base há apenas uma referência para inclusão do código, ou seja, o mesmo deve ser adaptado ao tema sendo desenvolvido – entretanto, um link para a **Intranet** (Painel Administrativo) deve ser incluído, caso o usuário logado seja nível 8 ou 9.

Se não informado o contrário, também incluir logomarca do Webcartórios.

5.11. Instruções Extras

1. Comentários devem ser incluídos apenas entre tags PHP, *mesmo em código HTML*.
Exemplo: <?php //Meu comentário ?>
2. Arquivos css, arquivos Javascripts, imagens, entre outros, localizados na pasta tema do cliente, devem ser incluídos utilizando o objeto **\$cliente->Layout_Dir**.
Exemplo:
3. Consultas SQL devem ser liberadas, após uso/armazenamento dos resultados da busca.
Exemplo:

```
$sqlPaginas = mysql_query("SELECT Id, Titulo FROM cartorio_paginas
WHERE Ativo = 1 order by Titulo");
$Paginas = mysql_fetch_object($sqlPaginas);
@mysql_free_result($sqlFormularios);
```
4. Arquivos .php devem ser incluídos utilizando a variável \$INCLUDE_SITE.
Exemplo: include (\$INCLUDE_SITE.'__inc/inc.estrutura.php');



6. Cartórios x Webconteúdos

6.1. Banco de Dados

Para sites de **Cartório**, o prefixo utilizado para tabelas no Banco de Dados é `cartorio_`. Para sites de **Webconteúdos** deve-se utilizar o prefixo `webc_`.

Exemplo:

- **Cartório:** `cartorio_dados`
- **Webconteúdos:** `webc_dados`

O correto é que seja fornecido pelo Coordenador ou Analista arquivos base já pré-definidos ao tipo de *website*. Entretanto, caso o desenvolvimento do tema apresente erros em consultas SQL, verifique se o prefixo está correto.

6.2. Arquivos

A inclusão de alguns arquivos também varia de acordo com o tipo de site.

Arquivo de configurações:

- **Cartório:** `__inc/cartorio.php`
- **Webconteúdos:** `__inc/config.php`

Arquivo de funções js:

- **Cartório:** `__inc/funcoes_cartorios.js`
- **Webconteúdos:** `__inc/funcoes__webconteudos.js`

Caso o desenvolvimento do tema apresente erros, verifique se foi feita a inclusão correta do arquivo correspondente.

6.3. Substituição em massa

Caso seja necessário realizar uma substituição em massa de nomes de arquivos, prefixos de tabela, etc, converse com o Analista ou Coordenador a respeito. A partir de um **projeto local**, criado pelo software *phpDesigner*, é possível realizar buscas e substituições em todos os arquivos vinculados a ele.

7. Testes Preliminares

1. **Referências:** Comparar tema atualizado com tema exemplo da empresa.
2. **Links:** verificar se direcionam para o local correto.
3. **Opção Buscar:** Sempre deve ser incluída, se não houver especificações contrárias por parte do Analista ou Coordenador. Verificar se funciona.
4. **Mapa de localização:** Verificar se funciona.
5. **Tema Responsivo:** Verificar se o tema está correto em todas as resoluções.
6. **Tema agradável:** Combinar e harmonizar cores de fontes e fundos, imagens, design, de acordo com a paleta determina pelo Analista ou Coordenador de Projeto.
7. **Painel Administrativo:** Verificar funcionamento do link e do gerenciamento dos módulos.
8. **Conteúdo fonte:** Eliminar referências do tema original (visíveis no browser e as de código fonte).
9. **Dúvidas:** A função HELP gera um botão de dúvidas, configurável nas Funções Personalizadas.