coreMVC 3.0

MVC PHP

Índice

[composer.json 4](#_Toc13049729)

[database\_tables.sql 4](#_Toc13049730)

[Help.html 4](#_Toc13049731)

[public 4](#_Toc13049732)

[.htaccess 4](#_Toc13049733)

[index.php 4](#_Toc13049734)

[assets 5](#_Toc13049735)

[css 5](#_Toc13049736)

[custom\_web.css 5](#_Toc13049737)

[img 5](#_Toc13049738)

[js 5](#_Toc13049739)

[logs 5](#_Toc13049740)

[uploads 6](#_Toc13049741)

[src 6](#_Toc13049742)

[app 6](#_Toc13049743)

[database.php 6](#_Toc13049744)

[routes.php 6](#_Toc13049745)

[Controllers 6](#_Toc13049746)

[AdminController.php 6](#_Toc13049747)

[ArticlesController.php 6](#_Toc13049748)

[Models 6](#_Toc13049749)

[Article.php 6](#_Toc13049750)

[Author.php 6](#_Toc13049751)

[Category.php 7](#_Toc13049752)

[Newsletter.php 7](#_Toc13049753)

[Visitor.php 7](#_Toc13049754)

[Views 7](#_Toc13049755)

[404.phtml 7](#_Toc13049756)

[layout.phtml 7](#_Toc13049757)

[layout\_admin.phtml 7](#_Toc13049758)

[layout\_login.phtml 7](#_Toc13049759)

[layout\_preview.phtml 7](#_Toc13049760)

[admin 7](#_Toc13049761)

[login.html 7](#_Toc13049762)

[article 7](#_Toc13049763)

[create.phtml 7](#_Toc13049764)

[edit.phtml 7](#_Toc13049765)

[index.phtml 7](#_Toc13049766)

[preview.phtml 7](#_Toc13049767)

[author 8](#_Toc13049768)

[create.phtml 8](#_Toc13049769)

[edit.phtml 8](#_Toc13049770)

[index.phtml 8](#_Toc13049771)

[category 8](#_Toc13049772)

[create.phtml 8](#_Toc13049773)

[edit.phtml 8](#_Toc13049774)

[index.phtml 8](#_Toc13049775)

[newsletter 8](#_Toc13049776)

[index.phtml 8](#_Toc13049777)

[visitor 8](#_Toc13049778)

[index.phtml 8](#_Toc13049779)

[alerts 8](#_Toc13049780)

[\_errors.phtml 8](#_Toc13049781)

[\_sucess.phtml 8](#_Toc13049782)

[article 8](#_Toc13049783)

[about.phtml 9](#_Toc13049784)

[adblock.phtml 9](#_Toc13049785)

[blog.phtml 9](#_Toc13049786)

[category.phtml 9](#_Toc13049787)

[contact.phtml 9](#_Toc13049788)

[index.phtml 9](#_Toc13049789)

[show.phtml 9](#_Toc13049790)

[core 9](#_Toc13049791)

[Auth.php 9](#_Toc13049792)

[Authenticate.php 9](#_Toc13049793)

[BaseController.php 9](#_Toc13049794)

[BaseModel.php 9](#_Toc13049795)

[bootstrap.php 9](#_Toc13049796)

[Container.php 9](#_Toc13049797)

[DataBase.php 9](#_Toc13049798)

[Helper.php 9](#_Toc13049799)

[Redirect.php 10](#_Toc13049800)

[Route.php 10](#_Toc13049801)

[Session.php 10](#_Toc13049802)

[UploadHelper.php 10](#_Toc13049803)

[storage 10](#_Toc13049804)

[marketing 10](#_Toc13049805)

[disqus\_i3r.inc 10](#_Toc13049806)

[vendor 10](#_Toc13049807)

[LICENSE 10](#_Toc13049808)

[11](#_Toc13049809)

# composer.json

Este arquivo é necessário para instalar a aplicação em si. Baixe os arquivos, descompacte e rode o comando abaixo:

*$ php composer.phar install*

Ou se o composer já estiver instalado no seu servidor:

*$ php composer install*

Aqui considerando ambiente de sistemas Linux. Se você usa Windows, não terá problemas em adaptar.

Com o comando acima será gerada a pasta vendor, e ela não será manipulada em nenhum momento da sua aplicação. Mas é essencial para que o carregamento das classes e métodos funcionem de forma correta.

# database\_tables.sql

Este arquivo mostra as tabelas usadas nesse framework com um usuário já cadastrado, não funcionando sem isso. O usuário inicial será:

**Usuário:** [admin@admin.com.br](mailto:admin@admin.com.br), **senha:** 12345

Ao entrar na sessão administrativa do site ([http://seusite.alguma.coisa**/admin**](http://seusite.alguma.coisa/admin)) use essas credenciais para primeiro acesso e depois exclua ou mude o login para sua segurança. A senha é criptografada para envio ao banco de dados e, se esquecer, o acesso ao banco de dados não ajudará muito a recuperá-la. Para gravar manualmente uma nova senha consulte <https://www.php.net/manual/pt_BR/function.password-hash.php> e <https://www.php.net/manual/pt_BR/function.password-verify.php>. Veremos mais a frente como a validação da senha é realizada.

# Help.html

Este arquivo que esta lendo!

# public

Pasta onde ficam os arquivos “públicos” do site, como imagens, css, javascript entre outros dentro da subpasta **assets**.

## .htaccess

Este arquivo sem nome, com a extensão .htaccess é um arquivo de configuração do Servidor Web Apache. O htaccess é utilizado para o controle de diversos parâmetros, entre eles restringir acessos, redirecionamentos, páginas de erro (403, 404, entre outras), URLs amigáveis, etc.

As quatro primeiras linhas são essenciais para o bom funcionamento da aplicação. As outras podem ser omitidas sem causar nenhum problema.

## index.php

Arquivo chave de qualquer site, é o primeiro a ser procurado no servidor. Este arquivo carrega os arquivos /vendor/autoload.php e /src/core/bootstrap.php que dão o primeiro “arranque” ao site/aplicação.

## assets

Pasta pública que armazena os arquivos de folha de estilo, imagens, scripts javascript, etc

### css

Subpasta de assets, armazena as folhas de estilo do site.

#### custom\_web.css

Neste arquivo, usei @import para importar de servidores próprios, folhas de estilos como bootstrap, font-awesome etc, bastando mudar os links para atualizar todo o estilo do site. Também algumas caracteristicas próprias são aplicadas neste arquivo.

### img

Subpasta de assets que armazena as imagens usadas na aplicação/site. Os arquivos de uso geral aparecem na raiz, tais como:

* *1one\_favicon.ico → imagem de favicon*
* *banner\_admin\_index.jpg → banner da área administrativa do site*
* *banner\_admin\_login.jpg → banner da área de login*
* *error\_404.jpg → imagem da página de erro*

Este arquivo é

### js

Subpasta de assets, que armazena arquivos javascript usados no site, tais como:

* myJSfunctions.js → armazena, atualmente, script para mover a página para o topo de forma rápida;
* sorttable.js → script aplicado na página administrativa para ordenar as colunas de todas as tabelas da página. Basta clicar no seu cabeçalho. Pode ser usada em outras áreas do site;
* tinymce\_config.js → a plataforma de edição de texto usada na área admistrativa, para descrição e texto de artigos, e em formulário de contato.
* tinymce\_pt\_BR.js → arquivo de configuração do TinyMCE para o português.

### logs

Subpasta de assets, que armazena os logs de erros gerados na aplicação. Sua configuração encontra-se no arquivo /src/core/bootstrap.php (*ini\_set('error\_log', "assets/logs/log\_$data.log"*). Infelizmente nem todos os logs vão para essa pasta, (bug não corrigido ainda), podendo aparecer **logs** de erro em outras pastas do sistema.

### uploads

Subpasta de assets, onde está sendo feito upload das imagens dos artigos dos sites. Mas nela deve ser feito upload de todo e qualquer arquivo, tais como pdf, .exe, etc.

O upload das imagens dos artigos é realizado para a pasta **articles**, sendo gerada no momento do upload, **subpastas** **com o nome do artigo**, sem acentos, espaços ou qualquer caractere especial.

Para a exibição das imagens, o script **thumbs.php,** presente dentro da pasta articles, é usado para redimensionar a imagem para qualquer valor colocado no “src” da imagem. Será melhor exemplificado no arquivo /src/core/Helper.php pois é usado em conjunto com o método *setPathImg($art, $img, $w, $h).*

# src

Esse diretório armazena dois outros: o **app** e o **core**. App armazena os *controllers, models* e *views* e os arquivos *database.php* e *routes.php*. Já em core, temos os arquivos que raramente serão alterados, sendo o núcleo do framework.

## app

Subdiretório de src/. Nele tem-se as subpastas *controllers, models* e *views* e os arquivos *database.php* e *routes.php*

### database.php

Este arquivo é responsável pela conexão ao banco de dados. Basta inserir o host, o nome do banco de dados (database), usuário, entre outros. Também é possível usar o banco de dados sqlite, fazendo a mudança em *'driver' => 'mysql',* se desejar.

### routes.php

As rotas que o site irá navegar são colocadas neste arquivo. Para melhor organização, estão separadas as partes que requerem login das que não necessitam. Para uma página exigir senha, ser redirecionada para a página “login” basta inserir o augumento ‘auth’ como em *$route[] = ['/admin', 'AdminController@index', 'auth'];*

### Controllers

Essa aplicação usa dois controllers apenas: o *AdminController.php* que controla a parte administrativa do site, onde pode-se inserir novos artigos, deletá-los ou editá-los, inserir, bloquear ou deletar usuários, categorias, bem como ver quem se inscreveu no site (sem função ainda para envio de newsletter aos inscritos), quem acessou, com seu ip, e o que viu (página que acessou, por “ip” único) e o *ArticlesController.php* que gerencia como os artigos são exibidos na tela e algumas outras funções.

#### AdminController.php

dasakskalkfalakf

#### ArticlesController.php

dasakskalkfalakf

### Models

dasakskalkfalakf

#### Article.php

dasakskalkfalakf

#### Author.php

dasakskalkfalakf

#### Category.php

dasakskalkfalakf

dasakskalkfalakf

#### Newsletter.php

dasakskalkfalakf

#### Visitor.php

dasakskalkfalakf

### Views

dasakskalkfalakf

#### 404.phtml

dasakskalkfalakf

#### layout.phtml

dasakskalkfalakf

#### layout\_admin.phtml

dasakskalkfalakf

#### layout\_login.phtml

dasakskalkfalakf

#### layout\_preview.phtml

dasakskalkfalakf

#### admin

dasakskalkfalakf

##### login.html

dasakskalkfalakf

#### article

dasakskalkfalakf

##### create.phtml

dasakskalkfalakf

##### edit.phtml

dasakskalkfalakf

##### index.phtml

dasakskalkfalakf

##### preview.phtml

dasakskalkfalakf

#### author

dasakskalkfalakf

##### create.phtml

dasakskalkfalakf

##### edit.phtml

dasakskalkfalakf

##### index.phtml

dasakskalkfalakf

#### category

dasakskalkfalakf

##### create.phtml

dasakskalkfalakf

##### edit.phtml

dasakskalkfalakf

##### index.phtml

dasakskalkfalakf

#### newsletter

dasakskalkfalakf

##### index.phtml

dasakskalkfalakf

#### visitor

dasakskalkfalakf

##### index.phtml

dasakskalkfalakf

#### alerts

dasakskalkfalakf

##### \_errors.phtml

dasakskalkfalakf

##### \_sucess.phtml

dasakskalkfalakf

#### article

dasakskalkfalakf

##### about.phtml

dasakskalkfalakf

##### adblock.phtml

dasakskalkfalakf

##### blog.phtml

dasakskalkfalakf

##### category.phtml

vdasakskalkfalakf

##### contact.phtml

dasakskalkfalakf

##### index.phtml

vdasakskalkfalakf

##### show.phtml

dasakskalkfalakf

## core

dasakskalkfalakf

### Auth.php

dasakskalkfalakf

### Authenticate.php

dasakskalkfalakf

### BaseController.php

dasakskalkfalakf

### BaseModel.php

dasakskalkfalakf

### bootstrap.php

dasakskalkfalakf

### Container.php

dasakskalkfalakf

### DataBase.php

dasakskalkfalakf

### Helper.php

dasakskalkfalakf

### Redirect.php

dasakskalkfalakf

### Route.php

dasakskalkfalakf

### Session.php

dasakskalkfalakf

### UploadHelper.php

dasakskalkfalakf

# storage

dasakskalkfalakf

## marketing

dasakskalkfalakf

### disqus\_i3r.inc

dasakskalkfalakf

# vendor

Esta pasta não está disponível para download pois é gerada com o comando composer, citado no inicio desse documento. Dentro dessa pasta será gerado os seguintes arquivos: autoload.php e a pasta composer com os arquivos:

* autoload\_classmap.php
* autoload\_namespaces.php
* autoload\_psr4.php
* autoload\_real.php
* autoload\_static.php
* ClassLoader.php
* installed.json

Não é necessário mexer nesses arquivos para a aplicação funcionar.

# LICENSE

Estes códigos são disponibilizados pela Licença: **MIT License**. Para mais detalhes, leia o arquivo LICENSE na sua íntegra.

# 



