

Bing

Para criar um arquivo de configuração em PHP que chame a página `index.php` localizada na pasta `public` do seu projeto, você pode seguir os seguintes passos:

1. **Estrutura do Projeto:** Primeiro, crie a estrutura básica do seu projeto. Vamos chamá-lo de `simple-mvc`. Aqui está a organização de diretórios que você pode adotar:

```

simple-mvc/
├── Application/
│   ├── controllers/
│   │   ├── Home.php
│   │   └── User.php
│   ├── core/
│   │   ├── App.php
│   │   ├── Controller.php
│   │   └── Database.php
│   ├── models/
│   │   └── Users.php
│   ├── views/
│   │   ├── home/
│   │   │   └── index.php
│   │   └── user/
│   │       ├── index.php
│   │       └── show.php
│   ├── .htaccess
│   └── autoload.php
├── public/
│   ├── assets/
│   │   ├── css/
│   │   │   └── bootstrap.min.css
│   │   ├── img/
│   │   │   └── erro.png
│   │   └── js/
│   │       ├── bootstrap.min.js
│   │       └── jquery.slim.min.js
│   ├── .htaccess
│   └── index.php

```

2. **Configuração do `.htaccess`:** Crie um arquivo `.htaccess` na raiz do seu projeto (fora da pasta `Application` e `public`). O conteúdo desse arquivo deve redirecionar as solicitações para a pasta `public`. Aqui está um exemplo:

```

RewriteEngine On
RewriteRule ^(.*)$ public/$1 [L]

```

3. **Conteúdo do `index.php`:** O arquivo `index.php` dentro da pasta `public` será o ponto de entrada para sua aplicação. Ele deve incluir o bootstrap e carregar os arquivos de configuração. Aqui está um exemplo simples:

```

<?php
// public/index.php

// Carrega o arquivo de configuração
require_once '../Application/autoload.php';

// Inicializa a aplicação
$app = new App();
$app->run();

```

4. **Configuração do Autoload:** O arquivo `autoload.php` dentro da pasta `Application` é responsável por carregar automaticamente as classes do seu projeto. Você pode usar o Composer para gerenciar o autoload ou implementar seu próprio autoloader.
5. **Implementação do Framework MVC:** Agora você pode começar a implementar os controladores, modelos e visualizações dentro da pasta `Application`. O foco está na separação de responsabilidades e na construção

do padrão MVC.

6. **Banco de Dados:** Certifique-se de criar o banco de dados (por exemplo, `mvc_db`) e a tabela (por exemplo, `users`) conforme mencionado no artigo.

Lembre-se de adaptar essas instruções ao seu projeto específico e às suas necessidades. Boa sorte com o desenvolvimento do seu framework MVC! 🚀