

[Nome do Sistema]

[Descrição do projeto]

📄 Sobre o Projeto

Este é um projeto desenvolvido em PHP que segue a arquitetura MVC (Model-View-Controller) para organizar o código de forma clara, escalável e de fácil manutenção. O objetivo do sistema é [objetivo].

📁 Estrutura de Pastas

A estrutura de pastas foi projetada para separar as responsabilidades da aplicação, facilitando o desenvolvimento e a manutenção do código. Abaixo está a descrição de cada diretório principal, partindo da raiz `sistema/`.

```
sistema/
├── app/
│   ├── Controlador/
│   │   └── Paginas/
│   ├── Database/
│   ├── Modelo/
│   │   └── Entidade/
│   └── Utilidades/
└── views/
    └── paginas/
```

sistema/app

Este é o coração da sua aplicação. Todo o código relacionado à lógica de negócio, regras e funcionamento principal do sistema está localizado aqui.

- `app/Controlador/Paginas`

Os **Controladores** são responsáveis por intermediar as requisições do usuário, processar os dados de entrada e enviar uma resposta. Cada controlador recebe a requisição, interage com os `Modelos` (se necessário) e envia os dados para uma `Vizualização` ser renderizada.

- `app/Database`

Esta pasta contém todas as classes e configurações relacionadas ao banco de dados.

- `app/Modelo/Entidade`

Os **Modelos** representam a estrutura de dados da sua aplicação e interagem com o banco de dados. A subpasta `Entidade` é designada para conter as "Entidades", que são classes que mapeiam diretamente as tabelas do seu banco de dados.

- `app/Utilidades`

Aqui ficam as classes e funções utilitárias que podem ser usadas em várias partes do sistema. São códigos de ajuda (helpers) que não se encaixam diretamente como um Controller ou Modelo, como por exemplo, classes para formatação de datas, manipulação de arquivos, envio de e-mails, validações genéricas, entre outros.

sistema/views/paginas

As **Vizualizações** são a camada de apresentação da sua aplicação. Elas são responsáveis por exibir os dados ao usuário. Nesta pasta, `paginas` conterá os arquivos que renderizam as diferentes páginas do sistema. As views recebem os dados enviados pelos `Controllers` e os exibem de forma estruturada para o usuário final.