**Informações sobre as tecnologias utilizados no projeto**

* HMVC (Hierarchical Model View Controller)

É uma evolução do MVC (Model View Controller), a ideia principal dessa estrutura é fazer com que seja possível criar vários módulos dentro de um único projeto. Ou seja, é possível dividir o projeto em domínios.

* DDD (Domain-Driven Design)

Designer de projeto, com a finalidade de fazer com que o projeto fique organizado de forma que as classes de domínio(MODEL), possam ser organizadas de tal forma, que elas só tratam de um assunto, e que o acesso aos dados possam ser divididos de forma, que a manutenção do projeto seja mínima possível.

* Doctrine

É um framework de abstração a banco de dados, ou um ORM (Object-relational mapping). A principal função dos ORMs, é fazer com que o projeto não tenha acesso a forma de pesistencia de dados, acessando diretamente os dados, ou até mesmo que se utilize uma linguagem comum de banco de dados (SQL).

* Symfony

Framework de php. Criado com a proposta de ser um framework full-stack (completo), e ao mesmo tempo, modular, altamente desacoplado (mínimo de dependências de outras bibliotecas) e rápido.

* GIT

Versionador de arquivos. A principal proposta é fazer com que você podessa versionar seus arquivos, de forma a ter total controle sobre essas versões.

* PHP

Linguagem de programação executada no servidor. Hoje é a linguagem mais popular entre os web desenvelopers.

* AngularJS

Framework MVC de front-end (utilizada para desenvolver a interface com o usuário). Hoje é um dos frameworks mais populares do mercado, já que oferece uma estrutura robusta é rápida para o desenvolvimento de interfaces.

* Mysql

Um dos bancos de dados mais populares do mundo.

* Gulp

Automatizador de funções. Ou seja, sabe aquela tarefa que você precisa fazer todas as vezes que vai publicar o seu projeto para internet, tipo: minimizar o código fonte do projeto, compactar as imagens, realizar um monte de configurações para o servidor de produção e entre muitas outras, que brincando leva umas 2 horas de trabalho. Pois é esse cara você cria um script, e ele faz tudo isso em 30 segundos.

* Docker

É uma plataforma de contenização. Ou seja sabe aquele trabalho todo para criar uma máquina virtual, e toda vez que for necessário inicializar leva aquele tempão. Então com esse cara você leva geralmente 5% do tempo para inicializar a mesma máquina.

As principais vantagens é que não é necessário fazer a virtualização(Tornar os hadwares físicos em bytes na sua mémoria) do sua máquina física(O PC que você está olhando para ele), já que o Docker, só sobe o Kernel(Core do sistema operacional(Linux, Windows, Mac)) do SO(Sistema operacional)