Nome: Gustavo Camalionti Gomes

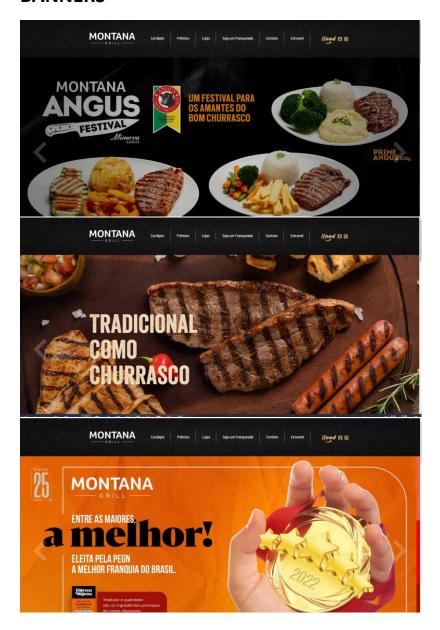
TIA: 21495580

A Halipar é uma holding de franquias, dona das marcas JinJin, Croasonho, Montana e Griletto. Apesar de sua proposta ser destinada na venda, manutenção e tudo que envolve a administração de franquias, ela também necessita de muito desenvolvimento interno de softwares, promoções, landing page e painéis internos. A empresa possui apenas dois desenvolvedores, um deles sou eu e o outro é meu lider, Senior e Gestor. Meu nome é Gustavo Camalionti, sou desenvolvedor FullStack Junior e atualmente trabalho com PHP, Laravel, Jquery, MySQL e JavaScript. Laravel é extremamente focado no padrão Arquitetural MVC, vou mostrar um pouco do projeto que estou fazendo nessas semanas.

Tudo começou com o desafio de refatorar um projeto legado, com todas as tecnologias atuais atualizadas e funcionais, reorganizar a arquitetura e códigos que estavam fugindo do padrão e desenvolver um painel administrativo no mesmo projeto onde o pessoal de marketing teriam acesso, poderiam fazer a gestão de usuários e ter acesso a diversas funcionalidades importantes que afetariam diretamente o site www.montanagrill.com.br

A princípio, teremos duas funcionalidades iniciais, onde aos poucos iremos incluindo novos módulos e feedback do pessoal: Customização de Banners e Lojas. S Apesar de ter iniciado a pouco menos de 3 semanas, já desenvolvi bastante coisa, segue algumas imagens referente o painel e as funcionalidades citadas:

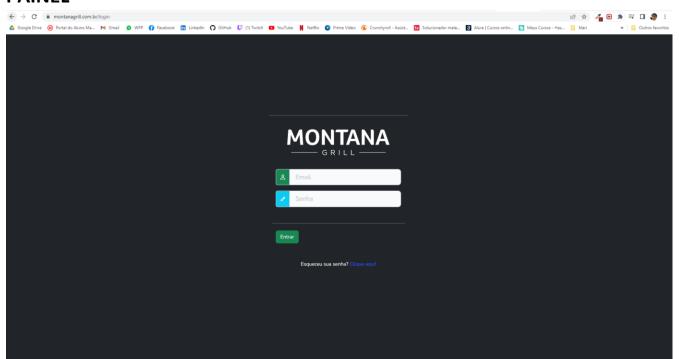
## **BANNERS**

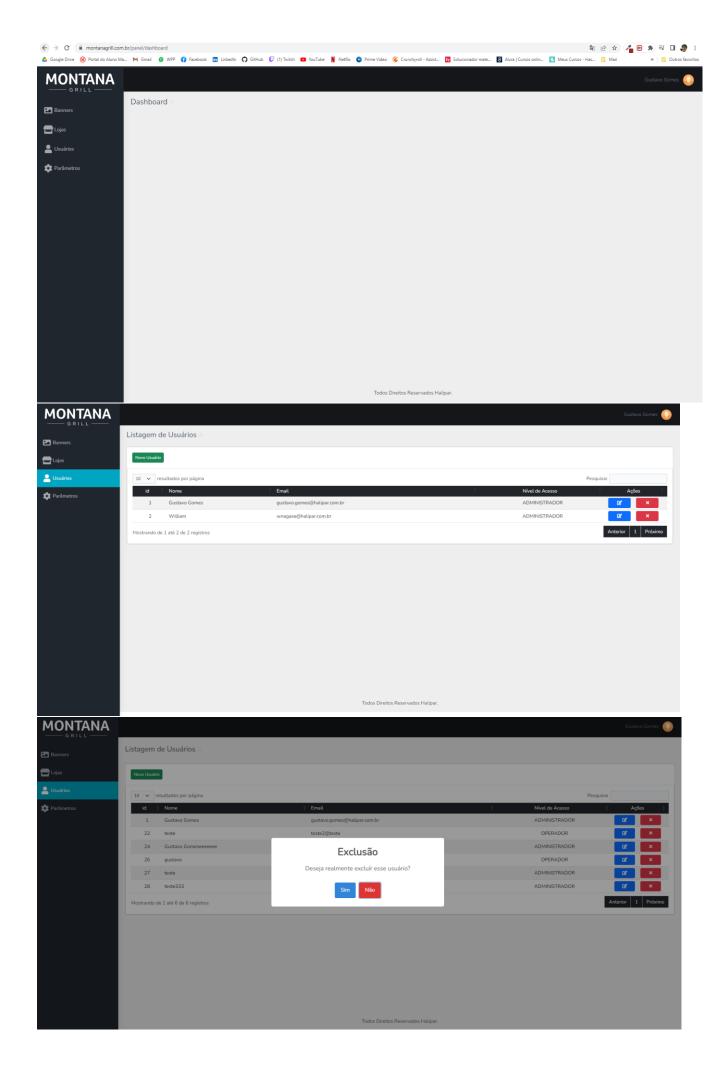


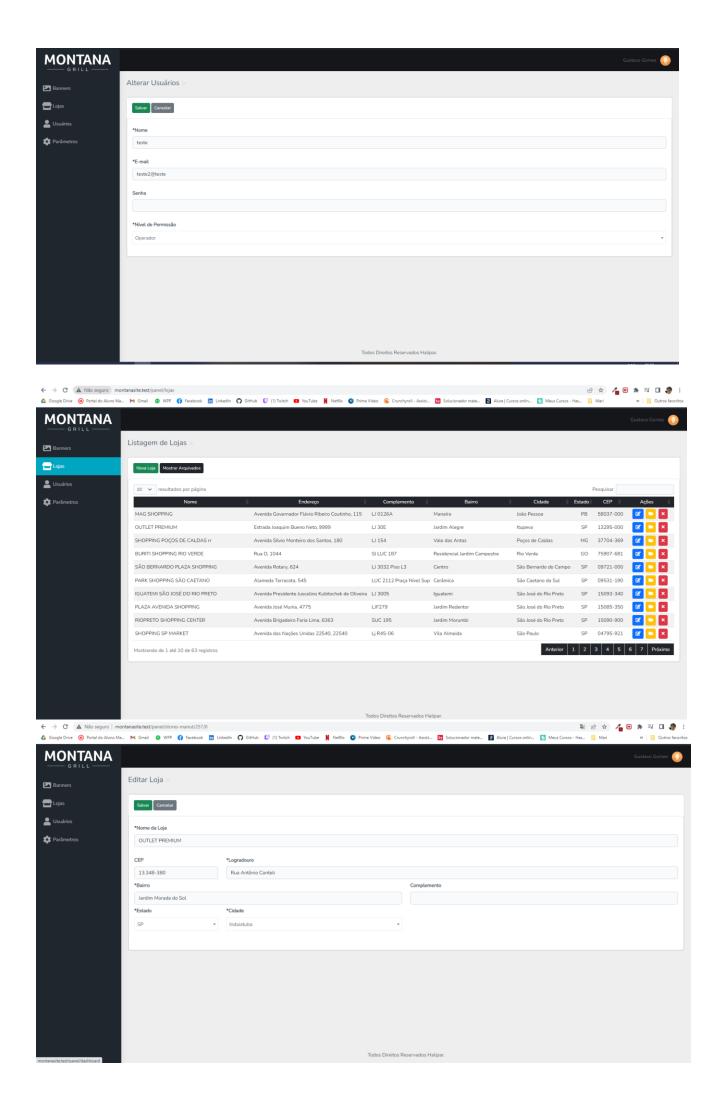
## **LOJAS**

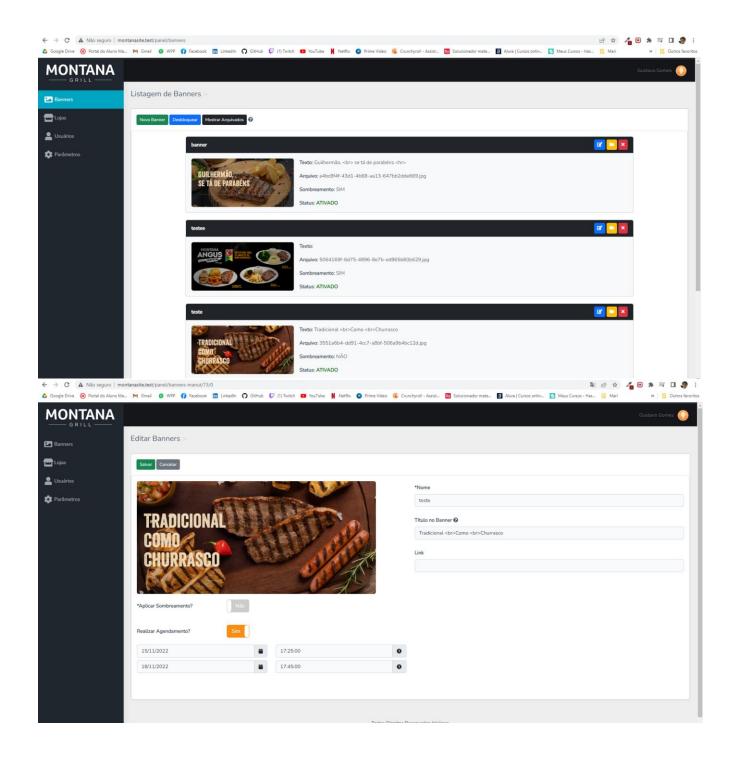


## **PAINEL**





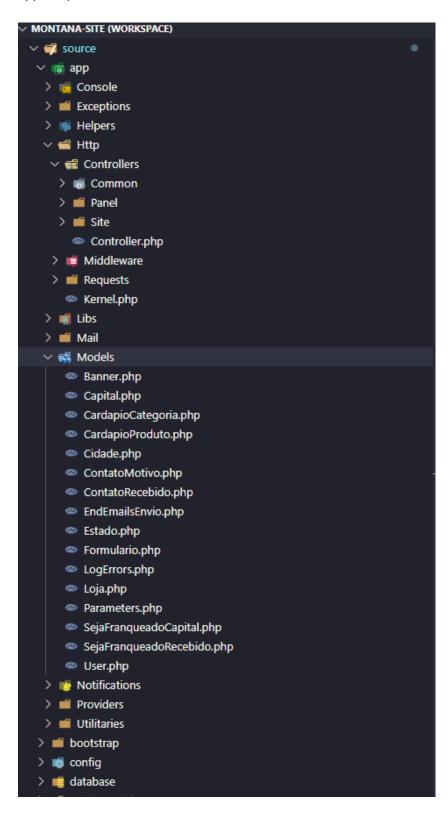


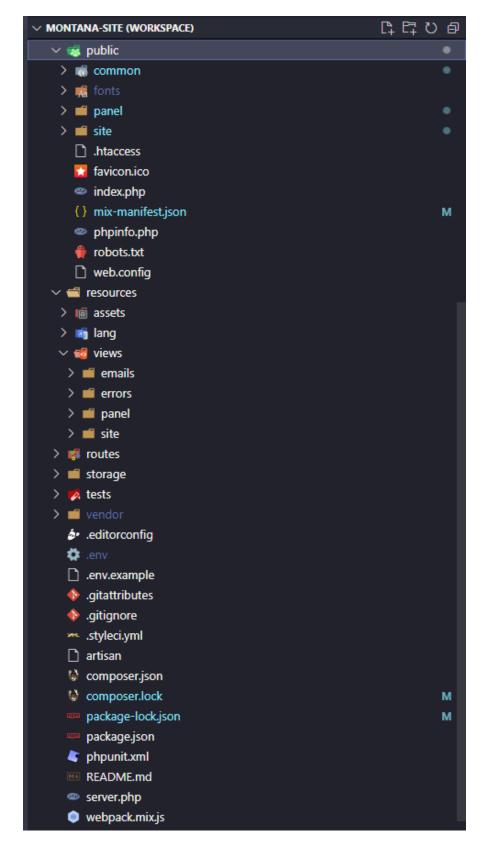


## **Arquitetura do Projeto (Montana-Site)**

Nessas duas primeira imagem, podemos perceber que precisa ser muito bem dividido, já que temos praticamente 2 grandes projetos no mesmo software. Como por exemplo:

- M | Todos os acessos ao banco está na pasta app>models.
- V | Todos as as páginas que o usuário irá interagir está dentro de Resource>views.
- C | Todos os controladores que irá tratar a informação e fazer o contato entre views e models fica em app>Http>Controllers.





Além dessas camadas extremamente importante, também possuimos algumas outras, como por exemplo:

- A pasta assets, dentro de resources. Nela foi colocado tudo relacionado com arquivos(css,js), plugins, icones, img e fontes.
- Rotas em routes. Obviamente todas as definições de rotas e configurações ficam nesse diretório.
- Public. Semelhante a assets, temos uma partição onde todo o conteúdo bruto de resources>assets irá ser compilado, processado e minificado, reduzindo códigos desnecessários, espaços inúteis e o tamanho dos arquivos, além de ser uma medida de segurança.
- Database. Diretório responsável pelas seeds, factories e migrations. Enquanto que a funcionalidade de migrations é relacionado em versionamento de banco de dados, temos a seeds e factories responsáveis em

popular em larga escala o banco de dados afim de realizar testes expressivos e aplicar as regras de como esses dados irão ser incluídos no banco. Como podemos ver, esse diretório basicamente faz o papel de se comunicar diretamente com o banco.

Acredito que eu tenha citado algumas outras camadas super importante usando como base o projeto que estou desenvolvendo. Ele está sendo um grande desafio e estou conseguido extrair muito aprendizado! Muito obrigado Professor.