

Tayná Crisllen José De Santana
Matrícula: 201905556
Arquitetura de Software - 2021/1 (ES)
Prof. Guilherme Silva Marques

COMÉRCIO BRASIL

1-Visão Lógica

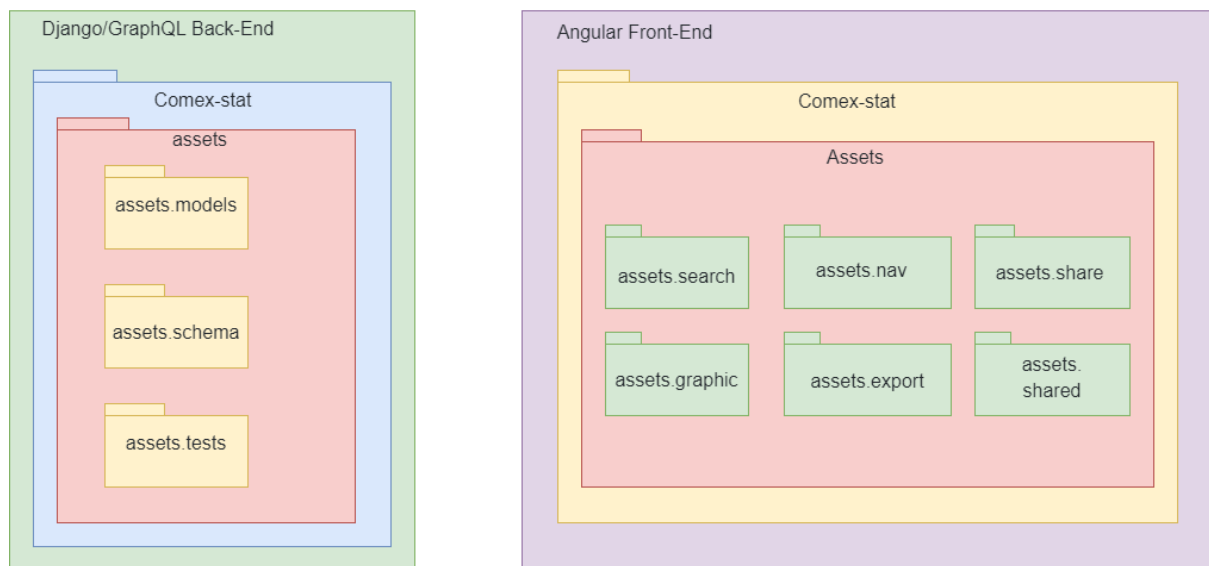
1.1-Visão geral

Usando o framework web Django o sistema será desenvolvido adaptando-se ao seu padrão MVT, tendo o Angular para o front-end e como linguagem e ferramenta de consulta ao banco de dados o GraphQL. Ficando da seguinte forma:

- O API será em Django.
- Como classes de domínios e interface com o banco de dados os Models do Django .
- Funcionando como cliente do back-end e fazendo requisições a ele pela interação do usuário ao acessar a aplicação, de tal forma a processar e mostrar de forma apropriada os dados de saída ao mesmo temos o Front-end em Angular.
- A função de fazer a consulta ao banco, cujos resultados serão posteriormente retornados ao front-end se necessário, será atribuída ao GraphQL como interface entre front-end e back-end.

O usuário acessa diretamente a API pelo navegador, através do GraphQL faz diretamente as suas pesquisas e filtragem nos dados do banco, ou por meio do front-end em Angular, acessando a aplicação web. Para fazer a identificação no padrão da URL que irá direcionar o usuário à View apropriada, através da qual o mesmo poderá fazer suas requisições, que serão então passadas para API da forma já descrita, será utilizado um chamado "router". Logo , os resultados retornados da busca feita são renderizados na View de forma apropriada para a visualização.

1.2- Pacotes de Design Significativos do Ponto de Vista da Arquitetura



Os pacotes de design significativos como sendo: models, schema e tests ficam no Back-End . Sendo cada um com seu conteúdo:

- models: Contém as models do Django, onde são definidos os modelos de dados. Representam a interface com o banco de dados.
- schema: Contém a definição dos métodos de pesquisa e filtragem de dados, unindo Django e GraphQL através da biblioteca Graphene.
- tests: Contém os testes unitários feitos no sistema.

Na representação dos pacotes de design significativos como sendo os Components do módulo "assets", que pertence a aplicação "comex_stat", sendo estes (todos contendo suas devidas Views e Templates relacionadas) estará no Front-End, em Angular:

- search: Component que define a parte da aplicação dedicada às pesquisas de dados
- graphic: Component que define a parte da aplicação dedicada a visualização gráfica dos dados
- nav: Component que define a barra de navegação da aplicação
- export: Component que define a parte da aplicação dedicada a exportação dos dados
- share: Component que define a parte da aplicação dedicada ao compartilhamento de páginas

No módulo "assets" a seção "shared", serão definidas coisas como os Services da aplicação.

2- Referências

[Documento de Arquitetura · Comex Stat \(fga-eps-mds.github.io\)](https://fga-eps-mds.github.io/Documentos/Arquitetura/Comex/Stat/)
[Microsoft SQL Server – Wikipédia, a enciclopédia livre \(wikipedia.org\)](https://pt.wikipedia.org/wiki/Microsoft_SQL_Server)