PLANEJAMENTO

Gênesis Peças e Serviços



1º - Estudar sobre microserviços, dashboard e gráficos em React.

20/05 – API escolhida para fazer os mapas.

- [React Google Charts](https://www.react-google-charts.com/)

Outras opções.

- [Recharts](https://recharts.org/en-US/)

- [React chartjs-2](https://react-chartjs-2.js.org/)

- [Victory](https://formidable.com/open-source/victory/)

2º - Desenhar o desgner da tela

23/05 – Primeiro layout da dashboard.



3º - Desenhar o fluxograma do banck-end

23/05 – A arquitetura escolhida foi a MSC(Model, Service, Controller).

- Controller para receber as solicitações e mandar para o service apenas o necessário.

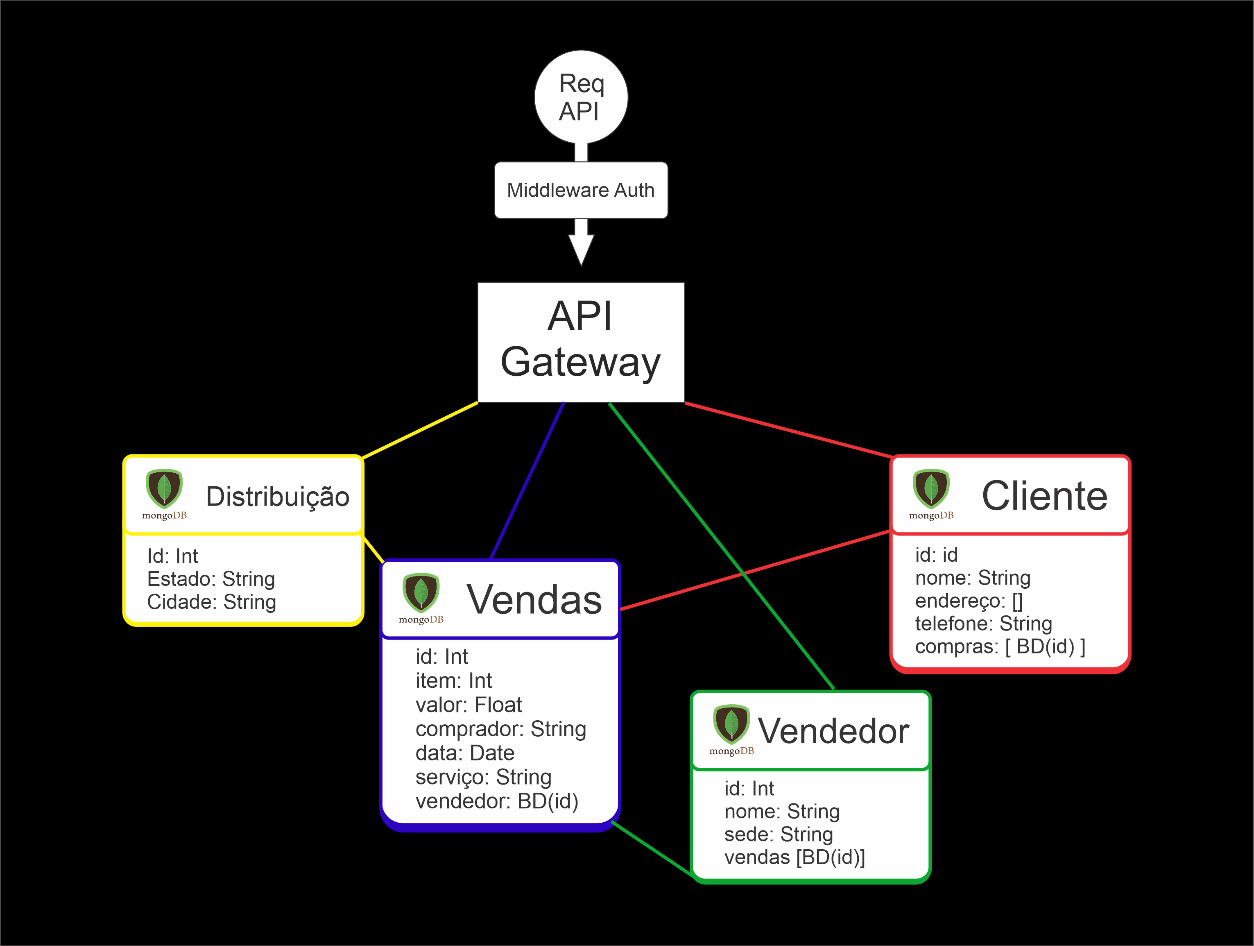
- Service recebe os dados e aplica as validações, e se estiver tudo certo, envia para o model os dados.

- Model faz a conexão com o banco, realiza o método de requisição e devolve um valor para o service.

– Por motivos de segurança, será implementado um middleware de autêntificação no inicio da requisição com o backEnd.

– A estrutura do backEnd ficou decidida da seguinte forma.

Quando um cliente faz uma solicitação, o gateway de API a divide em várias solicitações, as direciona para os locais adequados, produz uma resposta e faz o monitoramento.



4º - Preparar o front-end com dados mockados

24/05 – Header do site feito.

Foi usado a biblioteca [React select](https://react-select.com/home) para fazer os multi seletor de mês e vendedor do header.

Foi usado a biblioteca [MUI](https://mui.com/) para fazer o load da meta no campo meta anual.

26/05 – 6 primeiros gráficos prontos.

27/05 – Restante dos gráficos prontos e readme feito.

Foi usado a biblioteca [recharts](https://recharts.org/en-US/examples/SimpleRadarChart) para fazer o gráfico de radar, visto que a React-Google-Chart não disponibiliza esse tipo de gráfico.

- Resultado final –

Uma imagem com texto, monitor, parede, vários

Descrição gerada automaticamente

5º - Preparar o back-end e subistituir os dados mockados do front-end

6º - Deployer

PONTOS FORTES

1º - Foi desenvolvido uma interface com gráficos onde se é possível visualizar os dados da empresa.

2º - Foi utilizado recursos openSource para contruir os gráficos.

PONTOS FRACOS

1º - Como a API não foi desenvolvida, a integração não foi possível.

Atualmente estou cursando análise e desenvolvimento de sistemas na Estácio, e estou no último período… por esse e outros motivos, o tempo foi meu maior desafio nesse projeto.

2º - Microserviços

Outro ponto é que nunca tive contato direto com um projeto com microserviços, então tive que tirar um tempo para entender melhor esse conceito.

3º - Segurança

A segurança seria feita por meio de um middleware de auth no inicio da requisição.

4º Documentação

Como eu desenvolvi uma dashboard com tema escuro, era preciso alterar muita coisa dos modelos padrões da documentação, porém, foi algo que levou bastante tempo, a documentação era um pouco difícil de entender e com poucos exemplos… consegui mais ajuda no stackOverflow e youtube do que na documentação.

PONTOS DE MELHORIA

Organizar melhor meu tempo.

Estudar e práticar mais sobre microserviços.

Refatorar o desgner da página para algo mais dinâmico.

Refatorar partes repetidas no código.