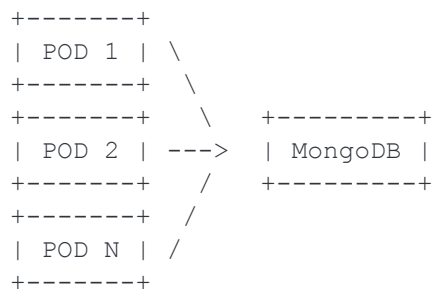


\* Back-end

### \* Pergunta 1

Suponha que vc tem uma API que roda 24/7 e que possui uma enorme quantidade de requisições. Para suportar o tráfego, existem várias instâncias desta aplicação rodando num cluster Kubernetes.

Atualmente, esta aplicação utiliza MongoDB como forma de persistir seus dados:



No entanto, o time técnico deseja migrar de banco para o PostgreSQL. Como você faria para migrar a aplicação sem downtime e sem perda de dados?

- Resposta: Primeiramente criaria uma estrutura semelhante no postgres e em paralelo os dev's ajustariam para serem gravados nas duas fontes, no caso do mongoDB e Postgres, o legado seria migrado em background e no final com tudo ok e funcionando faríamos um backup de todo mongoDB, efetuaríamos todo o restante da migração que faltasse e por fim tiraríamos o apontamento para o mongoDB, deixando somente o do Postgres.

Uma outra maneira também seria termos é claro a estrutura semelhante no postgres, um script de dump que migraria os dados para o postgres e criarmos uma fila de sincronismo de dados, ou seja, toda vez que um dado fosse gravado no banco de dados mongoDB esse dado era sincronizado com o postgres, em paralelo teríamos os ajustes necessários no fonte e por fim com o dump do banco faríamos um backup. Existem algumas ferramentas e bibliotecas que podem auxiliar como: mosql

### \* Pergunta 2

Suponha que vc esteja trabalhando numa API, melhorando um endpoint que grava em que ponto de um vídeo um usuário está. Atualmente, estes dados são gravados no mesmo banco de dados onde as demais informações dos vídeos

são armazenadas (nome, descrição, matéria, professor, etc). Com o aumento do número de usuários, o grande volume de escritas causadas por este endpoint tem impactado os demais endpoints desta API. Ainda assim, o time de produto deseja aumentar a granularidade do tracking de 15s para 5s - ou seja, guardaremos as posições dos usuários nos vídeos a cada 5s. Como vc faria para resolver esta situação sem impactar a experiência do usuário e atendendo as requisições de produto?

- Resposta: Podemos utilizar um banco de dados chave-valor como o Redis por exemplo para montarmos uma espécie de “cache”, onde colocaríamos o id do usuário como chave, e, no valor teríamos um hashmap, ou um objeto contendo o id do vídeo e o segundo. Acredito que assim diminuiríamos o custo de ficar gravando em um banco de dados relacional.

### \* Pergunta 3

Suponha que vc está colocando uma nova API no ar. Ela é composta de endpoints de escrita e leitura REST que se comunicam com um banco de dados SQL que não possui redundância. A API está de aberta para a Internet, sem um servidor de front / borda. Como vc acompanharia este serviço para garantir que tudo está saudável?

- Resposta: Primeiramente teríamos os testes rodando nessa api, como exemplo poderíamos utilizar o Jest. Teríamos log's também como o Sentry por exemplo que nos enviaria email caso algo acontecesse com a api. Podemos usar também a ferramenta pm2 que fornece praticamente tudo para monitorarmos a api.