

Esse é um exercício de modelagem. Vocês não vão programar nada, o entregável é o planejamento, que pode ser feito através de desenhos ou escrito. Tente explicar o fluxo de mudanças.

Exercício:

Vocês fazem parte de um time de desenvolvimento que atende a um PO responsável por melhorar os fluxos de cadastro. Esse fluxo é realizado por um sistema implementado com características de monólito.

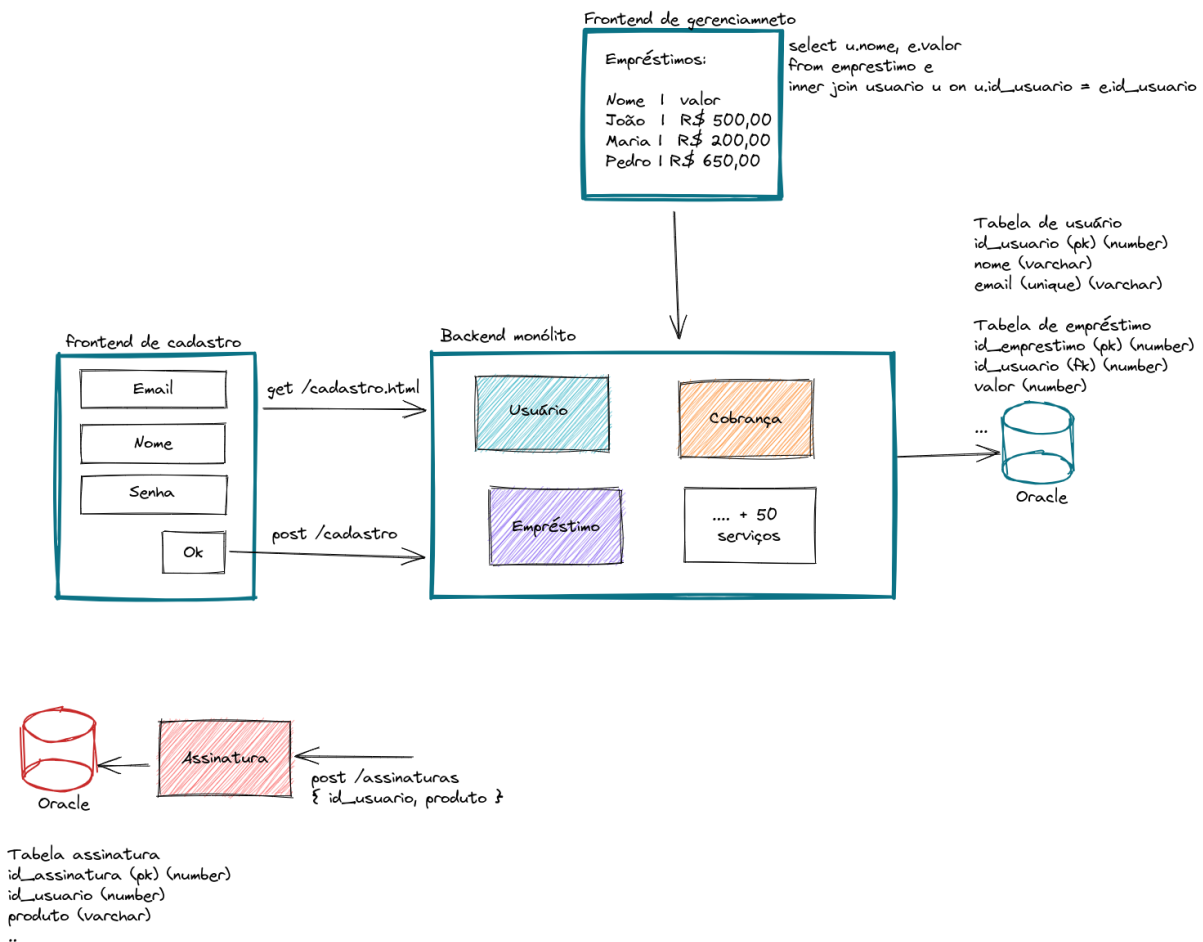
Esse monólito conecta-se a um banco de dados Oracle, implementa mais de 50 serviços diferentes e além de fornecer os serviços, é responsável por servir também as views do front-end (server side templating). Porém, o tamanho desse sistema está atrapalhando a produtividade da empresa, e além disso, a concorrência no uso de recursos (pool de threads, memória, CPU, pool de conexões, etc) ocasionalmente faz com que algum serviço atrapalhe o funcionamento dos demais, o que causa indisponibilidades.

Dado esse contexto, a empresa resolveu investir na adoção de uma arquitetura mais distribuída, adotando micro serviços. A infraestrutura necessária para executar esses serviços já está montada, e inclusive já existem alguns serviços implementados fora desse monólito.

Já o cadastro é uma funcionalidade relativamente simples. Basicamente ele tem uma tela em html simples, onde o usuário preenche um email, nome e senha e realiza o seu cadastro. No backend, além da validação de preenchimento de todos os campos, a única outra validação de negócio que é feita é da unicidade do email, inclusive o email é unique na tabela de usuário no banco de dados, e se um usuário preencher um email que já existe, ele recebe na tela uma mensagem de erro pedindo para escolher outro email.

Dado isso, o PO de cadastro, insatisfeito com o tempo que o cadastro fica indisponível devido aos problemas na aplicação e no banco desse monólito, resolve priorizar com o time de desenvolvimento a remoção do cadastro de usuários de dentro do monólito. Então o que ele quer é um planejamento de como remover a dependência síncrona que o cadastro tem com o sistema monólito (tanto da aplicação quanto do banco de dados). Além disso, sabendo que a maioria dos usuários se cadastram e logo depois criam uma assinatura, ele quer incluir na tela de cadastro uma opção do usuário escolher o produto que deseja assinar junto com o cadastro, mas com a condição de que, quando o cadastro incluir a seleção de um produto, o fluxo deve ser feito obrigatoriamente cadastrando o usuário e a assinatura, caso de um problema em qualquer um desses serviços, tudo deve ser desfeito de maneira que o usuário possa realizar uma nova tentativa de cadastro.

Por cima, o desenho da estrutura é esse, note que assinatura já está implementada fora da aplicação monólito:



Vocês podem usar, para desenhar as etapas de alterações:

- 1) <https://excalidraw.com/> importando o arquivo:
https://drive.google.com/file/d/1fMsiPJ90j3WiuE3Qf2_pqrzbTFpu6fND/view?usp=sharing
- 2) <https://draw.io> importando o arquivo:
<https://drive.google.com/file/d/1Ng7ZFW2Et7-MWdHhgNVxWZiu6Sf-NVSV/view?usp=sharing>

Pontos de atenção:

- 1) De nomes para os sistemas que forem criar.
- 2) Não basta apenas tirar a aplicação que realiza o cadastro, queremos que o cadastro não dependa de forma síncrona do próprio banco de dados do monólito, então precisamos de um banco novo. Não esqueça de pensar em como será o processo de migração dos dados de usuários do banco de dados do monólito.
- 3) Existem várias tabelas do monólito modeladas com uma chave estrangeira (foreign key) para o ID da tabela de usuário do monólito.

- 4) Existem queries dentro do monólito que fazem join na tabela de usuários e cujos resultados são exibidos em telas servidas pelo monólito.
- 5) Os recursos do monólito são escassos, qualquer perda de performance em alguma funcionalidade pode tornar mais constantes as indisponibilidades do sistema. Seria ruim termos uma solução que tornou mais disponível e escalável o cadastro de usuários, porém contribuiu para aumentar a indisponibilidade do sistema monólito.
- 6) Quando for incluída a opção de se realizar o cadastro de usuário com assinatura (que já está implementada fora do monólito), pense na situação onde ocorre um timeout ou erro (esperado ou não) na criação da assinatura, nesses casos, o usuário que está fazendo o cadastro deve tomar um erro na tela (pois não foi possível incluir se ter certeza da inclusão assinatura), porém se ele retentar a chamada com mesmo email, ele não pode tomar um erro de "email já cadastrado".
- 7) Não assuma nenhuma mudança de usabilidade ou processo que gere indisponibilidade, sem aprovação do PO (mas converse com o PO sobre alguma proposta, ele poderá dizer se aceita ou não).
- 8) Planos de desenvolvimento que possam levar muito tempo para ter alguma entrega, devem ser validado com o PO.