

Esboço da arquitetura do projeto

Lavínia Barbosa Dantas de Souza – 32221BSI014

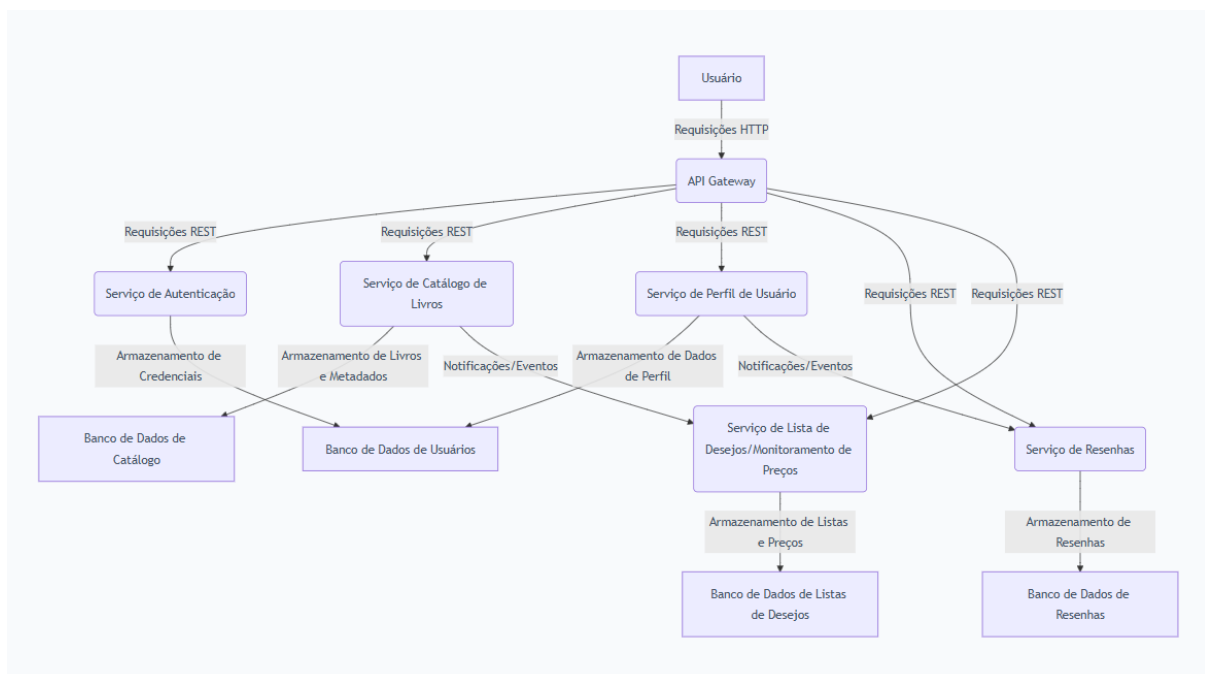
Letterbooks

1. Ideia: Um site semelhante ao Letterboxd, mas para livros. Haverá um catálogo vasto de livros, e os usuários poderão montar suas listas de desejos, realizar resenha dos livros lidos e ver as publicações e listas dos demais usuários.

2. Metas:

- Escalabilidade:** O site deve lidar as diversas requisições dos vários usuários para acessar o catálogo, fazer alterações em suas listas e realizar publicações das resenhas.
- Compartilhamento de recursos:** Diferentes usuários deverão ter acesso ao mesmo catálogo.
- Disponibilidade:** Como a ideia é fazer diversos **microserviços**, espera-se que se um componente falhar, outras partes do sistema continuem funcionando.

3. Diagrama da Arquitetura Inicial:



Usuário

- Responsabilidade:** Representa o cliente que interage com o sistema.

- **Comunicação:** Realiza requisições HTTP (GET, POST, PUT, DELETE) para o **API Gateway**.
- **Implementação:** Não será um componente do sistema distribuído em si, mas o ponto de interação externo.

API Gateway

- **Responsabilidade:** Ponto de entrada unificado para todas as **requisições (threads)** dos usuários. Ele atua como um roteador de requisições, direcionando-as para os microserviços apropriados, e pode lidar com autenticação e autorização preliminares.
- **Comunicação:** Recebe requisições HTTP do **Usuário** e as repassa para os serviços internos (Autenticação, Catálogo, Perfil, Resenhas, Lista de Desejos) usando comunicação **cliente-servidor** (provavelmente REST).
- **Implementação:** Será implementado como um **processo** separado, rodando em um servidor.

Serviço de Autenticação

- **Responsabilidade:** Gerenciar o registro, login, logout e validação de credenciais dos usuários. Ele é responsável por emitir e validar tokens de autenticação (JWT).
- **Comunicação:** Recebe requisições do **API Gateway** e se comunica com o **Banco de Dados de Usuários** para armazenar e consultar credenciais. A comunicação é **cliente-servidor**.
- **Implementação:** Será implementado como um **processo** separado.

Serviço de Catálogo de Livros

- **Responsabilidade:** Armazenar e fornecer informações detalhadas sobre os livros (título, autor, ISBN, capa, sinopse, etc.). Lida com a consulta e busca de livros.
- **Comunicação:** Recebe requisições do **API Gateway** e se comunica com o **Banco de Dados de Catálogo** para acessar os dados dos livros. A comunicação é **cliente-servidor**.
- **Implementação:** Será implementado como um **processo** separado.

Serviço de Perfil de Usuário

- **Responsabilidade:** Gerenciar os dados do perfil de cada usuário, incluindo suas listas de interesses (que não são as resenhas em si, mas as "listas" criadas pelos usuários, estilo Letterboxd) e a visualização das listas de outros usuários.

- **Comunicação:** Recebe requisições do **API Gateway** e se comunica com o **Banco de Dados de Usuários** para armazenar e consultar dados de perfil. Pode, eventualmente, notificar outros serviços sobre atualizações de perfil (via eventos). A comunicação é **cliente-servidor**.
- **Implementação:** Será implementado como um **processo** separado.

Serviço de Resenhas

- **Responsabilidade:** Gerenciar a criação, edição, exclusão e visualização de resenhas de livros postadas pelos usuários.
- **Comunicação:** Recebe requisições do **API Gateway** e se comunica com o **Banco de Dados de Resenhas** para armazenar e consultar as resenhas. Pode receber notificações do Serviço de Perfil de Usuário sobre novos usuários ou livros lidos. A comunicação é **cliente-servidor**.
- **Implementação:** Será implementado como um **processo** separado.

Bancos de Dados (DB1, DB2, DB3, DB4)

- **Responsabilidade:** Cada banco de dados é responsável por persistir os dados de um ou mais serviços, garantindo a integridade e disponibilidade das informações.
- **Comunicação:** Os serviços se comunicam com seus respectivos bancos de dados usando **cliente-servidor** (através de drivers de banco de dados).
- **Implementação:** Serão instâncias de banco de dados (ex: PostgreSQL), que podem rodar como **processos** dedicados em servidores separados.

Extra:

Serviço de Lista de Desejos/Monitoramento de Preços

- **Responsabilidade:** Permitir que os usuários criem listas de desejos de livros e monitore os preços desses livros em diferentes varejistas, notificando os usuários sobre quedas de preço.
- **Comunicação:** Recebe requisições do **API Gateway** e se comunica com o **Banco de Dados de Listas de Desejos** para armazenar e consultar as listas. Pode interagir com o **Serviço de Catálogo de Livros** para obter metadados de livros e, para o monitoramento de preços, pode realizar chamadas externas para APIs de varejistas. A comunicação é **cliente-servidor**. Pode também receber notificações do Serviço de Catálogo sobre novos livros para monitoramento.
- **Implementação:** Será implementado como um **processo** separado.