**Requisitos Funcionais:**

* Criar um endpoint POST /imoveis que receba informações sobre o imóvel e a negociação e as salve no banco de dados (Postgres).
* Criar um endpoint GET /imoveis/{id} que retorne os detalhes da negociação e uma avaliação de risco.
* A avaliação de risco deve considerar os seguintes parâmetros:
  + **Valor do imóvel:** Se o valor for muito alto ou muito baixo em relação à média do mercado, a negociação será reprovada.
  + **Histórico de crédito do comprador:** Se o comprador tem um bom histórico de crédito (baseado em um valor de pontuação de crédito fornecido na requisição), a negociação é aprovada. Caso contrário, a negociação pode ser reprovada.
  + **Renda estimada do comprador:** Se o comprador ganha mensalmente o suficiente para comprar com folga o imóvel, ele deveria ser aprovado.

**Tecnologias:**

* Backend em Python, utilizando o framework **FastAPI**.
* Banco de dados **PostgreSQL**.
* ORM **SQLAlchemy** para interação com o banco de dados.

**Critérios de Avaliação do Risco:** A decisão de aprovação ou reprovação da negociação pode ser feita de maneira simplificada, com as seguintes regras:

* Se o valor do imóvel for maior que 10 milhões ou menor que 100 mil reais, a negociação será **reprovada**.
* Se o comprador tiver uma pontuação de crédito inferior a 500 (num sistema de 0-1000), a negociação será **reprovada**.
* Se o valor do imóvel for maior que 30% da renda do comprador, a negociação será **reprovada**.

**Instruções Técnicas:**

* O banco de dados deve ser configurado para persistir as negociações e as informações dos imóveis.
* A aplicação deve estar estruturada de forma a permitir fácil manutenção e escalabilidade.
* A avaliação de risco deve ser realizada no momento da criação da negociação e retornada na consulta via id, ou seja, ao salvar já avalia o crédito e devolve o id, em seguida ao consultar o id, recebo a avaliação de crédito.
* Você deve documentar a API de forma clara.