

# **UNIVERSIDADE FEDERAL DE SÃO CARLOS**

## **GRUPO 2**

FELIPE BONADIA BRAVO - 813908

LAURA RIEKO MARÇAL IMAI - 813451

RYAN DE MELO ANDRADE - 812899

## **FASE INTERMEDIÁRIA 01 - PROJETO PROCESSAMENTO MASSIVO DE DADOS**

Definição da proposta - Hotéis AirBnb

Profa. Dra. Sahudy Montenegro González

SOROCABA

2025

# ÍNDICE

<b>Definição do tema.....</b>	<b>3</b>
<b>Introdução ao tema.....</b>	<b>3</b>
<b>Objetivos.....</b>	<b>4</b>
<b>Modelagem.....</b>	<b>4</b>
Tecnologias escolhidas.....	4
Decisões de projeto.....	5
Arquitetura e fluxo de dados.....	6
Sobre o uso do programa.....	7
<b>Bibliografia.....</b>	<b>8</b>

## **Definição do tema**

O tema que iremos explorar no desenvolvimento do trabalho é o “Sistema de Recomendação para Hospedagens Airbnb”, feito com base em um banco de dados híbrido - de Documentos e Grafos. Nessa senda, visa-se modelar e analisar dados sobre hotéis, suas comodidades, localizações, preços, dentre outros atributos, com objetivo de fornecer recomendações personalizadas baseado nas preferências demonstradas pelo histórico do usuário.

## **Introdução ao tema**

Com a popularização das plataformas de hospedagem como o Airbnb, a possibilidade de oferecer recomendações “personalizadas” aos usuários é um diferencial, que melhora a experiência deste. Nesse contexto, um sistema de recomendação desempenha um papel fundamental para auxiliar na seleção de acomodações que atendem aos requisitos e necessidades específicas com base em um histórico.

Para o projeto em específico, propomos um modelo híbrido de banco de dados, que integra dois paradigmas: Documentos e Grafos. As informações estruturadas e descritivas acerca dos hotéis, como nome, descrição, número de quartos e preços serão armazenadas em Documentos, à medida que relacionamentos entre as hospedagens, suas comodidades oferecidas, localizações e avaliações de usuário serão representadas em Grafo, facilitando, assim, uma análise mais eficiente dessas conexões complexas. Desse modo, os dados armazenados em documentos possibilitam consultas detalhadas sobre atributos específicos dos hotéis, ao passo que os dados em grafos permitem identificar, de forma mais rápida, hospedagens que, por exemplo, possuem características semelhantes de comodidade ou localização, visto que, no banco de grafos, as relações entre os dados são parte central da estrutura, e podem ser consultadas e navegadas eficientemente.

Com isso em mente, a proposta do sistema de recomendação está pautada em dois principais requisitos de negócio: sugerir hotéis com comodidades similares àquelas dos hotéis que o usuário já alugou anteriormente; e sugerir hotéis com quantidades de quartos e faixa de preço semelhantes às adotadas pelo histórico de usuário.

Para isso, utilizaremos o conjunto de dados obtido pelo próprio site do MongoDB, sendo esses a amostra do conjunto de dados de listagens AirBnB, que possui detalhes sobre cerca de 5.500 anúncios da rede de hotéis do AirBnB.

Desse modo, o projeto propõe uma aplicação prática de tecnologias NoSQL, ao mesmo tempo em que evidencia como a análise de dados pode contribuir para a personalização e aprimoramento de serviços de hospedagem.

## Objetivos

O programa tem como objetivo oferecer um sistema de recomendação de hotéis Airbnb baseado nas preferências demonstradas pelo histórico do usuário. Para ser mais específico, baseado nas avaliações anteriores do usuário, será identificado os hotéis que ele alugou no passado. A partir disso, serão analisadas as características desses hotéis, como seu número de quartos e comodidades oferecidas, o que oferecerá um padrão de preferências desse usuário, que servirá como base para a recomendação de novos hotéis após designada uma localização. Desse modo, será possível implementar um sistema baseado nas preferências de usuários, com apoio de tecnologias NoSQL para representar, de modo eficiente tanto os dados descritivos, quanto os seus relacionamentos.

Em primeira análise, destaca-se como eixo a ser explorado pela aplicação a análise de preferências do usuário, sendo esta composta pela identificação, por meio do histórico de reservas do usuários, dos atributos mais comuns das hospedagens escolhidas - como número de quartos e faixa de preço -, além da detecção de comodidades mais recorrentes nessas hospedagens anteriores.

Em segunda análise, deve-se compreender o eixo de recomendações baseadas em similaridade, ou seja, sugerir hospedagens com perfil similar, no que diz respeito a atributos e comodidades, às preferências passadas, possibilitando, em consonância a isso, uma filtragem por localização para refinar os resultados.

## Modelagem

### Tecnologias escolhidas

Para o desenvolvimento do trabalho, serão usados os modelos noSQL MongoDB e Neo4j, com a integração entre eles sendo realizada por meio dos procedimentos da biblioteca apoc.

Adentrando mais na escolha de tecnologias, considerou-se os requisitos de desempenho, flexibilidade, estrutura de dados e facilidade de integração entre estas. Nesse cenário, tem-se que o MongoDB foi escolhido, tendo em vista a facilidade em utilizá-lo para consultas rápidas, sobre hospedagens específicas, além da possibilidade de encapsulamento das informações em um único registro - um documento JSON. Com relação à decisão de uso do Neo4j, como modelo orientado a grafos, tem-se a modelagem natural de conexões, no caso relacionamentos, que é interessante para o projeto e será melhor abordada no tópico de decisões de projeto.

Em consonância com as tecnologias escolhidas, optou-se pela integração feita por meio da biblioteca APOC (Awesome Procedures On Cypher), que será utilizada para

importar os dados do MongoDB diretamente para o Neo4j, fato que facilitará a migração e transformação “automática” dos dados em formato de documento, para grafos.

### **Decisões de projeto**

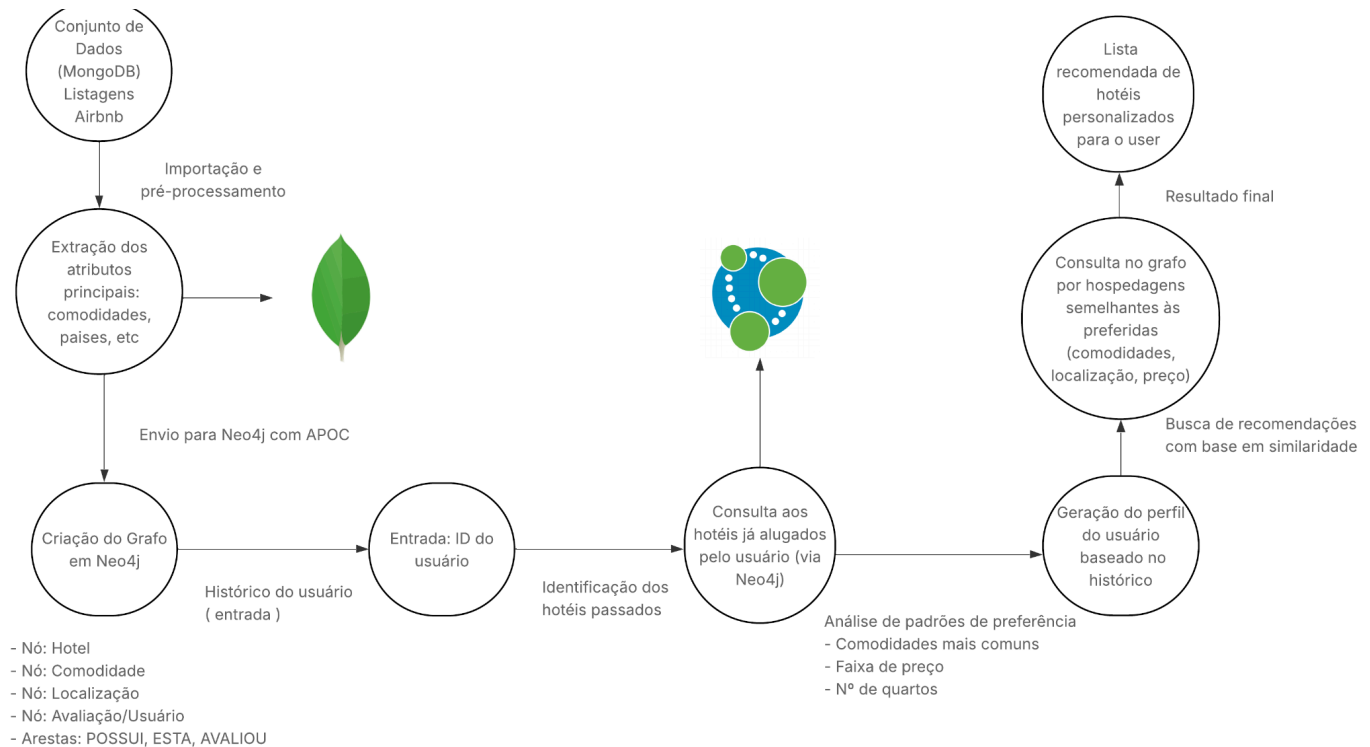
A escolha do uso de um banco de dados orientado a grafos, para armazenar as informações relevantes para a recomendação de hotéis, deve-se ao fato dele ser substancialmente mais eficiente para essa funcionalidade, quando comparado a outros modelos de banco de dados, pois:

- As relações são o foco, o que é o caso para o sistema de recomendação proposto, que considera as relações que os hotéis possuem com as suas comodidades, localizações e avaliações;
- Possui menos custos com junções, uma vez que - no SQL, por exemplo - seriam necessários múltiplos *joins* de diferentes tabelas (no caso, usuário, avaliação, hotel, comodidades), o que degradaria a performance, enquanto em grafos, a mesma lógica se resume a percorrer arestas, tendo, desse modo, custo constante;
- Lida naturalmente com dados conectados, sem duplicação, nem agregações complexas, já que no cenário abordado pelo projeto, caso o conjunto de dados fosse mantido integralmente no MongoDB, onde se origina a priori, seriam necessárias diversas agregações para realizar um sistema de recomendação, e cada comodidade, que é padronizada, poderia estar duplicada em diferentes arquivos de hotéis.

Acerca de como os dados presentes no grafo vão ser organizados, que são as comodidades dos hotéis, as reviews dos usuários, os próprios hotéis e os países que estão disponíveis, todos eles se tornarão nós, dos quais haverão ligações (arestas) que indicam:

- Que um hotel possui uma comodidade, quando um hotel é ligado a uma comodidade;
- Que um hotel está em um determinado país, quando um hotel é ligado a um país;
- Que um usuário avaliou e se hospedou em um hotel, quando um hotel é ligado a uma avaliação.

## Arquitetura e fluxo de dados



A arquitetura do sistema de recomendação segue uma estrutura modular, com etapas bem definidas para ingestão, transformação, armazenamento e análise dos dados.

Com relação ao fluxo de dados, pode-se inferir que o processo se inicia com a coleta de dados de hospedagem, que estão armazenados no MongoDB, contendo várias informações, como nome, quartos, preços, comodidades e avaliações. Posteriormente, será realizada a importação e o pré-processamento dos dados, garantindo que os atributos desejados sejam considerados na análise. Com essa etapa concluída, extraímos os atributos principais, como as comodidades, países, dentre outros.

Em seguida, deve-se realizar o envio dos dados processados para o Neo4j, aproveitando-se da biblioteca APOC, permitindo que a criação dos grafos seja concluída com sucesso, sendo que nele estarão presentes nós dos tipos: Hotel, Comodidade, Localização, Avaliações/usuários - e arestas dos tipos: POSSUI, ESTA, AVALIOU. Após isso, o usuário entra com o ID para receber as recomendações personalizadas, o sistema realiza uma consulta para identificar as hospedagens que já foram alugadas pelo usuário, analisando, desse modo, os padrões de preferência do usuário, no que diz respeito à prioridade de comodidades, preços e quartos por exemplo.

Esses padrões são usados para gerar um perfil do usuário baseado em seu histórico, com o perfil gerado, o sistema consulta o grafo por hospedagens que possuem similaridade com as preferências. Por fim, é apresentada ao usuário uma lista personalizada de hotéis recomendados, com base nas similaridades encontradas.

### **Sobre o uso do programa**

Para ilustrar o funcionamento do sistema de recomendações proposto, considere o caso de um usuário que já alugou três acomodações no Airbnb. Todas elas possuíam dois quartos. As comodidades oferecidas por cada uma foram as seguintes:

- Acomodação 1: Wi-Fi, ar-condicionado, cozinha, extintor de incêndio, geladeira e micro-ondas.
- Acomodação 2: Wi-Fi, cozinha, cafeteira, máquina de lavar, TV e geladeira.
- Acomodação 3: Wi-Fi, garagem, cozinha, aquecedor, geladeira e forno.

Com base nesse histórico, o sistema identifica padrões de preferência do usuário. Neste exemplo, destaca-se a recorrência de acomodações com dois quartos, além das comodidades Wi-Fi, cozinha e geladeira, presentes em todas as estadias anteriores. Assim, ao recomendar novas opções, o programa prioriza aquelas que compartilham essas características, oferecendo sugestões mais alinhadas ao perfil e às preferências do usuário.

## Bibliografia

- SANTURBANO, Andrea. *Transform MongoDB collections automagically into Graphs*. Medium, Neo4j Developer Blog, 29 nov. 2019. Disponível em: <https://medium.com/neo4j/transform-mongodb-collections-automagically-into-graphs-9ea085d6e3ef>. Acesso em: 11 jun. 2025
- MONGODB. *Atlas Sample Data – Sample Airbnb*. MongoDB, [s.d.]. Disponível em: <https://www.mongodb.com/pt-br/docs/atlas/sample-data/sample-airbnb/>. Acesso em: 17 jun. 2025.
- NEO4J. *APOC Library – Awesome Procedures On Cypher*. Neo4j Labs, [s.d.]. Disponível em: <https://neo4j.com/labs/apoc/>. Acesso em: 17 jun. 2025.