

Plataforma Social de Artistas, Shows e Fãs

Introdução

O documento foi elaborado para a disciplina de Banco de Dados Não Relacional do curso de Desenvolvimento de Software Multiplataforma (3º semestre) da Fatec Dr. Thomaz Novelino de Franca–SP, por João Marcos Landi Sousa, Ana Laura Lis Oliveira Zenith, Hérciles Robert Mendes, Eduardo Fernandes Grespi, consolida o descritivo do projeto detalhando o tema, justificativa e objetivos, descrição das entidades presentes e cada um de seus campos além da justificativa para o tipo escolhido para cada relação entre entidades e detalhamento das cardinalidades.

1) Tema justificativa e objetivos:

O presente projeto tem como finalidade o desenvolvimento de uma solução Web (back-end e front-end) com integração com Banco de Dados Não Relacional, voltada para artistas musicais em ascensão. A iniciativa busca impulsionar a visibilidade de artistas nacionais e fortalecer a cultura da música brasileira, valorizando talentos locais e ao mesmo tempo criando um espaço digital que conecta artistas e fãs.

Entre os recursos do projeto, destaca-se uma página que contém a agenda dos shows, para que os fãs consigam acompanhar os eventos de forma atualizada, uma página que contém as informações dos artistas como biografia e redes sociais.

O site também oferece uma página de produtos oficiais dos artistas, funcionando como uma vitrine digital e possibilidade de comercialização de itens exclusivos.

2) Descrição das entidades e seu campos:

Artistas:

- objectID: (string, obrigatório) identificador único para cada artista no banco evitando duplicidade.
- nome: (string, obrigatório) informação que identifica o artista.
- gênero musical: (string, obrigatório) categorizar e oferecer filtros para o usuário.
- bio: (string, opcional) apresenta um resumo da carreira do artista.
- redes sociais: (array<string>, opcional) canal de interação com o artista.
- email: (*string, obrigatório*) meio de contato formal com artista.
- telefone: (string, opcional) meio de contato com artista.

Produtos:

- objectID: (string, obrigatório) identificador único para produto no banco evitando duplicidade.
- artistaID: (string, obrigatório) relaciona produto e artista
- nome: (string, obrigatório) identifica produto

- descricao: (string, opcional) detalha o produto e suas características (Ex: camiseta branca, caneca artista X, poster artista Y).
- preço: (double, obrigatório) informação para transações de venda.
- estoque: (int, obrigatório) controla quantidade disponível dos itens

Eventos:

- objectID: (string, obrigatório) identificador único do evento/show.
- artistaID: (string, obrigatório) relaciona o show com o artista.
- data: (dateTime, obrigatório) define quando o show/evento acontece.
- local: (string, obrigatório) define o local onde o show/evento acontece.
- descrição: (string, opcional) detalha as informações adicionais sobre o evento.
- preçoIngresso: (decimal, obrigatório) informação dos valores de entrada no ingresso, permitindo que o usuário planeje sua participação.

Pedidos:

- objectID: (string, obrigatório) identificador único para o pedido.
- nome: (string obrigatório) identifica o nome do produto.
- data_hora: DateTime define quando o pedido foi realizado.
- cliente_id: identificador para vincular cliente e pedido

ItemPedido:

- objectID: (string, obrigatório) identificador único para o itempedido.
- num_item: Integer, para identificar qual o item.
- quantidade: Double, quantidade de itens no para compra.
- produto_id e pedido_id: identificadores para vincular produtos, itens do pedido e o pedido.

Clientes:

- objectID: (string, obrigatório) identificador único para cada cliente.
- nome: (string, obrigatório) necessário para identificação
- cpf:(string, obrigatório) necessário para validação de usuário
- dataNascimento:(datetime, obrigatório) informações adicionais para verificação de usuário.
- email:(string, obrigatório) usado para cadastrar/logar no sistema
- telefone: (string, opcional) informação para contato com cliente
- senha: (string, obrigatório) credencial de acesso
- endereço: (string, opcional) para entregas de produtos

Fornecedor:

- objectID: (string, obrigatório) identificador único para o fornecedor.
- razao_social: String obrigatório para controle de cadastro de fornecedores

- nome_fantasia: String opcional, informações adicionais para controle de cadastro.
- CNPJ: String obrigatória, informação para validação de usuário
- email: String obrigatório, para gerenciamento de informação do sistema.
- endereco: String obrigatório, para gerenciamento de informações do sistema.

3) Descrição das Relações entre as entidades:

Artistas → Produtos

- Tipo: Composição
- Cardinalidade: 0...* (um artista pode ter zero ou vários produtos, mas cada produto pertence a um único artista).
- Justificativa: Os produtos existem apenas porque estão associados a um artista. Se o artista for removido do sistema, os produtos perdem o sentido de existência.

Artistas → Eventos

- Tipo: Composição
- Cardinalidade: 0...* (um artista pode ter zero ou vários eventos, mas cada evento está vinculado a um único artista).
- Justificativa: Os eventos estão ligados ao artista. Um show sem o artista não tem motivos para existir.

Clientes → Pedidos

- Tipo: Composição
- Cardinalidade: 0...* (um cliente pode ter zero ou vários pedidos, mas cada pedido é único vinculado ao cliente).
- Justificativa: O pedido só existe se o cliente realizar a compra

Pedidos → ItemPedido

- Tipo: Composição
- Cardinalidade: 1 (pelo menos um item para que existe o pedido).
- Justificativa: um pedido só existe se um item for comprado

Produtos → ItemPedido

- Tipo: Composição
- Cardinalidade: 0...* (um produto estar presente em zero itempedido ou vários itens pedidos).
- Justificativa:

Produtos→ Fornecedor

- Tipo: Associação
- Cardinalidade: 0...*(um produto pode ter zero ou vários fornecedores e cada fornecedor por ter zero ou vários produtos).
- Justificativa: Produtos e fornecedores são entidades que não dependem uma da outra para existirem.