

UNIVERSIDADE FEDERAL FLUMINENSE
INSTITUTO DE COMPUTAÇÃO

Projeto de Banco de Dados - 2025/2 Prof.
Marcos Bedo

RELATÓRIO DE IMPLEMENTAÇÃO:
SISTEMA DE GERENCIAMENTO DE
PLATAFORMAS DE STREAMING

Integrantes:

Gabriel Vieira
Arthur Mota Danillo
Cyrilo Pedro Piaes
Matheus Andrade

Niterói - RJ
Novembro de 2024

Conteúdo

1	Introdução	2
2	Definição de Dados (DDL)	2
2.1	Tipos de Dados Personalizados e Domínios.....	2
2.2	Entidades Principais.....	2
2.3	Plataformas e Canais.....	2
2.4	Conteúdo e Interação.....	2
2.5	Sistema Financeiro (Herança).....	3
3	Consultas e Procedimentos Armazenados	3
3.1	Relatórios de Patrocínio e Inscritos.....	3
3.2	Relatórios de Doações.....	3
3.3	Rankings (Top K).....	4
3.4	Faturamento Consolidado.....	4
4	Otimização e Vistas (Views)	4
4.1	Views e Materialized Views.....	4
4.2	Índices de Performance.....	5
5	Triggers e Regras Ativas (Consistência)	6
5.1	Atualização Automática de Usuários na Plataforma.....	6
5.2	Atualização Automática de Visualizações no Canal.....	6
5.3	Atualizar Status de Doação ao Remover Comentário.....	6
5.4	Padronização Monetária Automática.....	6
5.5	Auditoria de Mudança de Nick.....	6
5.6	Gestão de Inscrições em Canais Privados.....	6
6	Conclusão	7

1 Introdução

Este documento detalha o processo de modelagem e implementação de um banco de dados relacional voltado para o gerenciamento de múltiplas plataformas de streaming, criadores de conteúdo (streamers), usuários e transações financeiras (doações e inscrições). O objetivo principal deste projeto é garantir a integridade referencial dos dados e otimizar a estrutura para um ambiente de alto volume de informações, assegurando que regras de negócio, como tipos de canais e status de pagamentos, sejam rigorosamente seguidas.

2 Definição de Dados (DDL)

As constraints NOT NULL e UNIQUE foram implementadas para garantir a integridade e consistência dos dados, seguindo as regras de negócio do sistema. Os campos NOT NULL são aplicados a informações essenciais que não podem ser omitidas, como dados de identificação básica (nomes de empresas, plataformas e usuários), informações obrigatórias de contato (email, telefone, endereço), datas fundamentais (data de fundação, nascimento) e valores monetários necessários para cálculos financeiros. Essas constraints previnem a inserção de dados incompletos que comprometeriam a funcionalidade do sistema.

Já as constraints UNIQUE garantem a não duplicidade de informações críticas para a identificação única de registros. Isso inclui nomes fantasia de empresas, nomes de plataformas, emails de usuários, combinações de passaporte por país e identificadores nacionais de empresas. A unique também é usada em chaves alternativas, como a relação entre plataforma e número de usuário na PlataformaUsuario, assegurando que um usuário tenha apenas um número identificador por plataforma. Essas restrições mantêm a consistência referencial e previnem duplicatas que causariam conflitos nas operações do sistema de streaming e doações.

NOT NULL:

- **Empresa.nome**: Não faz sentido ter uma empresa sem nome
- **Empresa.nome_fantasia**: Toda empresa precisa de um nome fantasia para identificação comercial
- **Conversao.nome** e **Conversao.fator_conver**: Uma moeda deve ter nome e taxa de conversão definidos
- **Plataforma.nome**, **Plataforma.empresa_fund**, **Plataforma.empresa_respo**, **Plataforma.data_fund**: Plataforma precisa de nome, empresas fundadora e responsável, e data de fundação
- **Pais.nome** e **Pais.moeda**: País deve ter nome e moeda oficial definidos
- **Usuario.email**, **Usuario.data_nasc**, **Usuario.telefone**, **Usuario.end_postal**, **Usuario.pais_residencia**: Dados essenciais do usuário para contato e localização
- **PlataformaUsuario.nro_usuario**: Número identificador único do usuário na plataforma
- **StreamerPais.nro_passaporte**: Número do passaporte é obrigatório para controle de nacionalidade
- **EmpresaPais.id_nacional**: ID nacional é necessário para empresa atuar em um país
- **Canal.tipo**, **Canal.data**, **Canal.nick_streamer**: Canal precisa de tipo, data de criação e streamer responsável
- **Patrocino.valor**, **Patrocino.data_inicio**: Patrocínio deve ter valor e data de início
- **NivelCanal.nivel**, **NivelCanal.valor**, **NivelCanal.gif**: Nível precisa de classificação, valor e gif associado
- **Inscricao.nivel**, **Inscricao.data_inscricao**: Inscrição requer nível e data
- **Video.dataH**, **Video.tema**, **Video.duracao**: Vídeo precisa de data/hora, tema e duração

- **Comentario.texto, Comentario.dataH:** Comentário deve ter texto e timestamp
- **Doacao.valor, Doacao.status, Doacao.data_doacao:** Doação requer valor, status e data
- **BitCoin.TxID:** Transação Bitcoin precisa do ID da transação
- **PayPal.IdPayPal:** Pagamento PayPal requer ID do PayPal
- **CartaoCredito.nro, CartaoCredito.bandeira:** Cartão precisa número e bandeira
- **MecanismoPlat.seq_plataforma:** Sequência única da plataforma para doações

UNIQUE:

- **Empresa.nome_fantasia:** Nome fantasia não pode repetir entre empresas
- **Conversao.nome:** Nomes de moedas devem ser únicos
- **Plataforma.nome:** Nomes de plataformas não podem se repetir
- **Pais.nome:** Nomes de países devem ser únicos
- **Usuario.email:** Email deve ser único por usuário
- **PlataformaUsuario.(nro_plataforma, nro_usuario):** Número do usuário deve ser único por plataforma
- **StreamerPais.(ddi_pais, nro_passaporte):** Passaporte deve ser único por país
- **EmpresaPais.(ddi_pais, id_nacional):** ID nacional deve ser único por país
- **Canal.(nro_plataforma, nick_streamer):** Um streamer pode ter apenas um canal por plataforma
- **Comentario.(nome_canal, nro_plataforma, titulo_video, dataH_video, nick_usuario, seq):** Sequência de comentário deve ser única por vídeo e usuário
- **BitCoin.TxID:** ID de transação Bitcoin deve ser único
- **PayPal.IdPayPal:** ID do PayPal deve ser único
- **MecanismoPlat.(nro_plataforma, seq_plataforma):** Sequência da plataforma deve ser única por plataforma

Otimização e Vistas (Views)

Esta seção aborda as estratégias de otimização de performance e simplificação de consultas complexas através do uso de Views, Views Materializadas e Índices específicos para cargas de trabalho de leitura intensiva.

2.1 Views e Materialized Views

View de Patrocínios em Canais (vw_patrocinos_canais)

Objetivo: Listar canais patrocinados, valor do patrocínio e empresa patrocinadora de forma desnormalizada para relatórios rápidos.

View de Membros Inscritos (vw_membros_inscritos)

Objetivo: Calcular, para cada usuário, de quantos canais ele é membro e a soma mensal desembolsada. Devido à alta cardinalidade (milhões de usuários) e ao padrão de acesso pontual (consulta por ID), optou-se por uma **View Virtual** em

vez de materializada, evitando custos proibitivos de armazenamento e refresh. A performance é garantida pelo índice criado na tabela Inscricao.

View de Doações por Canal (Materializada - vw_doacoes_canais)

Objetivo: Pré-calcular a soma total de doações por canal, evitando varreduras completas na tabela de doações para dashboards. Esta view filtra dados inconsistentes (status recusado) na origem.

View de Doações Lidas por Vídeo (vw_doacoes_lidas_por_video)

Objetivo: Filtrar e somar apenas as doações que foram efetivamente lidas (status 'lido') por vídeo, facilitando a análise de engajamento qualificado.

View de Receita Total por Canal (Materializada - vw_receita_total_canal)

Objetivo: Consolidar todas as fontes de receita (patrocínio, membros e doações) em uma única visão unificada. Devido ao alto custo computacional de agregar três fontes distintas e realizar junções completas (FULL OUTER JOIN), esta visão é candidata ideal para materialização, otimizando drasticamente o relatório de ranking financeiro (Consulta 8).

2.2 Índices de Performance

O índice **nome_canal** na tabela Canal é um dos mais importantes porque o nome do canal é usado constantemente em buscas e relacionamentos. Ele ajuda diretamente em cinco consultas diferentes: quando precisamos buscar doações por canal, calcular os tops de patrocínio, membros, doações e o faturamento total. Sem este índice, o banco teria que varrer a tabela inteira de canais toda vez que alguém fizesse uma busca por nome de canal, o que ficaria cada vez mais lento conforme o sistema fosse crescendo.

O índice **nome_nro_patrocincio** na tabela Patrocincio é um índice composto que acelera as consultas que envolvem tanto o canal quanto a empresa patrocinadora. Ele é especialmente útil para a função que lista todos os patrocínios e para os relatórios de top canais, onde precisamos somar valores de patrocínio por canal. Esse índice evita que o banco de dados tenha que fazer ordenações demoradas em disco quando estamos agrupando dados de patrocínio.

O índice **nome_usuario_inscricao** na tabela Inscricao foca em melhorar as consultas relacionadas aos usuários. Quando um usuário quer ver em quantos canais está inscrito ou quanto está gastando por mês com assinaturas, esse índice torna a busca muito mais rápida. Ele também ajuda no cálculo do faturamento total dos canais através das assinaturas de membros.

Os índices de ordenação por valor (**ordena_valor_doaacao** e **ordena_valor_patrocincio**) são especializados em consultas financeiras. Eles usam a estrutura BTREE, que é perfeita para ordenações e filtros por faixa de valores. Esses índices são usados em praticamente todos os relatórios financeiros - quando precisamos saber quais canais recebem mais doações, quais patrocínios são mais valiosos, ou quando calculamos totais e médias. Sem eles, o banco teria que ordenar manualmente todos os registros toda vez que alguém pedisse um relatório ordenado por valor.

A consulta mais complexa do sistema top_canais_faturamento é um ótimo exemplo de como todos esses índices trabalham juntos. Ela usa o índice de **nome_canal** para fazer várias junções entre tabelas, aproveita o índice composto de patrocínio para agrupar valores, utiliza os índices de ordenação para calcular somas totais e se beneficia do índice de usuários na parte de agregação de membros. Essa consulta seria extremamente lenta sem a combinação adequada de índices.

A seguir segue uma comparação dos tempos médios das consultas rodando com e sem índice além de alterando os parâmetros de suas respectivas funções visualizadas através do Explain Analyzer:

CONSULTA	USOU ÍNDICE?	AVG. PLANNING TIME (ms)	AVG. EXECUTION TIME (ms)
1	Não	1.426	1.934
1	Sim	1.493	1.306
2	Não	1.428	4.606
2	Sim	1.768	2.245
3	Não	1.432	2.857
3	Sim	1.706	1.370
4	Não	1.320	0.324
4	Sim	2.109	0.430
5	Não	2.064	1.747
5	Sim	2.700	1.306
6	Não	2.457	1.836
6	Sim	2.814	1.660
7	Não	1.294	2.041
7	Sim	1.051	1.822
8	Não	3.284	4.793
8	Sim	3.385	3.837

3 Triggers e Regras Ativas (Consistência)

Esta seção apresenta os gatilhos (triggers) desenvolvidos com foco na manutenção ativa da consistência dos dados. Diferente de validações passivas (que apenas rejeitam dados), estes triggers agem automaticamente para atualizar atributos derivados, realizar conversões ou manter logs de auditoria, garantindo que o banco de dados permaneça sincronizado e confiável.

3.1 Atualização Automática de Usuários na Plataforma

Este trigger mantém o contador de usuários na tabela Plataforma sempre atualizado. Ao inserir ou remover um usuário da tabela de relação PlataformaUsuario, ele incrementa ou decrementa automaticamente o campo qtd_users, otimizando consultas futuras que precisem desse total sem necessidade de COUNT(*) onerosos.

3.2 Atualização Automática de Visualizações no Canal

Mantém a soma total de visualizações de um canal atualizada em tempo real. Quando um vídeo sofre alterações em suas visualizações, o total do canal é ajustado automaticamente, garantindo que a métrica agregada esteja sempre correta.

3.3 Atualizar Status de Doação ao Remover Comentário

Define um comportamento de "fail-safe": se um comentário com doação for excluído, a doação não fica órfã nem é excluída, mas sim marcada como 'recusada' para manter o histórico contábil e evitar "sumiço" de dinheiro nos relatórios.

3.4 Gestão de Arrecadação por Vídeo

Mantém o atributo *valor_arrecadado* na tabela Video. Este trigger soma valores quando uma doação entra, mas subtrai automaticamente se o status da doação for alterado para 'recusado' posteriormente.

3.5 Padronização Monetária Automática

Ao inserir uma doação, este trigger consulta a tabela de conversão e a moeda do país do usuário para calcular e armazenar automaticamente o valor padronizado em Dólar, facilitando relatórios financeiros globais.

3.6 Gestão de Inscrições em Canais Privados

Implementa uma regra de negócio restritiva: se um canal alterar seu status para 'privado' (ou for suspenso), todas as inscrições ativas são automaticamente can-

celadas ou suspensas, mantendo a coerência de acesso ao conteúdo e evitando cobranças indevidas.

4 Conclusão

O desenvolvimento deste projeto permitiu a aplicação prática dos conceitos de modelagem de dados, implementação de consultas complexas e regras ativas. O esquema de banco de dados desenvolvido atende robustamente às necessidades de gerenciamento de plataformas de streaming, garantindo a integridade dos dados através de chaves estrangeiras, restrições e triggers. As consultas armazenadas fornecem uma interface flexível para extração de relatórios gerenciais, cobrindo aspectos de patrocínio, engajamento de usuários e análise financeira consolidada.