



UFRN - UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

IMD - INSTITUTO METRÓPOLE DIGITAL

CURSO DE GRADUAÇÃO EM TECNOLOGIA DA INFORMAÇÃO

Relatório Final: Plataforma Imobiliar

Componente Curricular: IMD0401 - Banco De Dados

Docente: Prof. Dr. João Carlos Xavier Júnior

Discentes:

Emilly Miller Moreira

Gabrielle de Vasconcelos Borja



Sumário

1 Introdução.....	01
2 Objetivos.....	02
2.1 - Gerais	
2.2 - Específicos	
3 Diagrama Entidade Relacionamento.....	03
3.1 - Imagem do Diagrama Entidade Relacionamento	
3.2 - Entidades	
4 Metadados.....	06
4.1 - Views	
4.2 - Triggers	
5 Conclusão.....	12

1 - Introdução

Uma das maiores dificuldades no desenvolvimento de um sistema de banco de dados é representar de forma precisa as entidades e as regras de negócio que refletem a realidade de um determinado cenário. No caso deste projeto, foi desenvolvida a **Plataforma Imobilar**, um sistema voltado à intermediação entre anunciantes e clientes interessados na venda ou aluguel de imóveis residenciais e comerciais. A plataforma se inspira em modelos reais e de conhecimento mundial como Booking, Airbnb e OLX, que intermediam as transações de aluguel e venda de imóveis em primeira pessoa, sem a necessidade de um corretor ou imobiliária para fechar um contrato.

Assim, a proposta da Imobilar surge da necessidade de organizar e automatizar processos comuns no mercado imobiliário, como o cadastro de imóveis, o gerenciamento eficiente de contratos, o controle preciso de disponibilidade, e a troca de informações entre os usuários da plataforma. Para isso, o banco de dados foi cuidadosamente modelado para representar com fidelidade a dinâmica das interações entre clientes, anunciantes e os imóveis ofertados, garantindo a integridade das relações e transações.

O projeto contempla também funcionalidades adicionais que enriquecem a experiência dos usuários, como sistema de avaliações de anunciantes, controle de acessos, além de triggers e views que garantem a integridade e a eficiência das operações do sistema. Dessa forma, a modelagem da Plataforma Imobilar foi construída com base em práticas consolidadas de banco de dados, resultando em uma estrutura robusta, flexível e alinhada às necessidades reais de um sistema moderno de gestão imobiliária.

2 - Objetivos

2.1 - Gerais

O sistema tem como objetivo centralizar e organizar todas as informações relacionadas à gestão de imóveis para venda e aluguel, conectando anunciantes e clientes por meio de uma estrutura relacional bem definida. A Plataforma Imobiliar busca garantir a integridade, a consistência e a confiabilidade dos dados cadastrados, evitando a descentralização de informações em planilhas ou sistemas informais, e permitindo o gerenciamento eficiente de contratos, avaliações, usuários, imóveis e suas respectivas interações.

2.2 - Específicos

A aplicação permitirá a realização de diversas operações e consultas úteis, tais como:

- Cadastrar e consultar imóveis disponíveis, com filtros por cidade, valor e tipo de imóvel.
- Registrar contratos de venda e aluguel, com controle de disponibilidade e prazos.
- Avaliar anunciantes com base em experiências anteriores, gerando uma média de reputação.
- Gerenciar perfis de usuários (clientes e anunciantes) e suas informações pessoais.
- Impedir inconsistências como contratos duplicados para um mesmo imóvel ou datas inválidas em aluguéis.
- Disponibilizar visões analíticas com os principais dados da plataforma por meio de *views*.

3 - Diagrama Entidade Relacionamento

3.1 - Imagem do Diagrama de Entidade Relacionamento

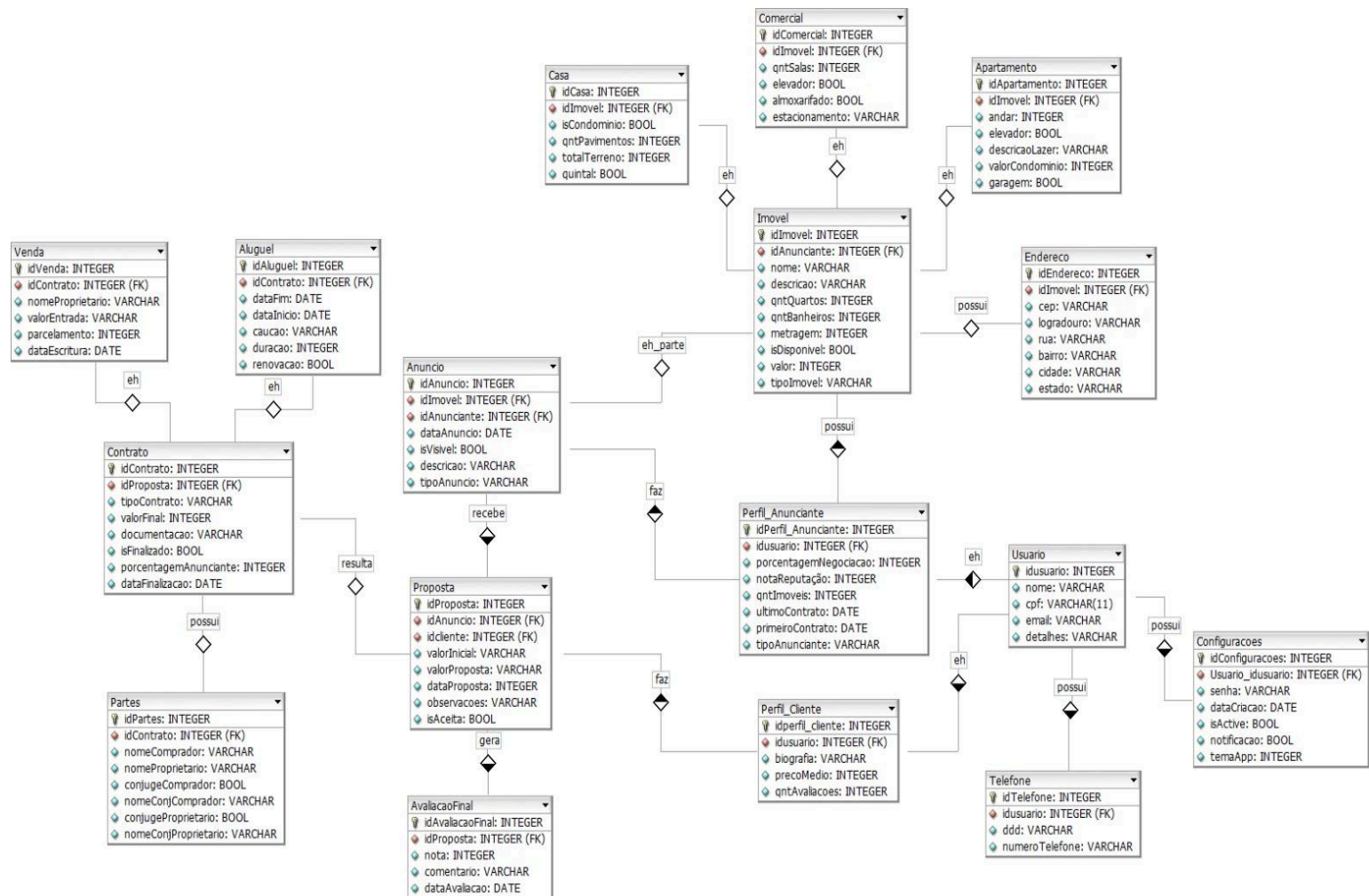


Figura 01 - Diagrama Entidade Relacionamento.

3.2 - Entidades

- **Usuário**
 - Tabela base que armazena os dados de todas as pessoas que utilizam o sistema, sejam elas anunciantes, compradores ou ambos.
 - Cada usuário possui dados como nome, CPF, email e detalhes pessoais.
 - Outras entidades como *Perfil_Anunciante* e *Perfil_Cliente* referenciam o *idusuario*.
- **Perfil_Anunciante**
 - Especialização de *Usuário* que representa quem deseja anunciar um imóvel.
 - Armazena dados específicos como a porcentagem de negociações bem-sucedidas, nota de reputação e o número de imóveis já anunciados.
 - Está ligado à tabela *Imovel*.
- **Perfil_Cliente**
 - Especialização de *Usuário* para representar quem deseja comprar ou alugar um imóvel.
 - Guarda informações como biografia, preço médio que costuma pagar e quantidade de avaliações recebidas.
- **Telefone**
 - Armazena os telefones de contato dos usuários.
 - Cada usuário pode ter um ou mais telefones cadastrados.
- **Configuracoes**
 - Define preferências de uso do sistema para cada usuário, como senha, tema do aplicativo, se o perfil está ativo e se deseja receber notificações.
- **Imovel**
 - Representa os imóveis cadastrados na plataforma, com informações como nome, descrição, número de quartos e valor.
 - Cada imóvel pertence a um anunciante (*Perfil_Anunciante*).
 - A tabela é especializada em tipos como *Casa*, *Apartamento* e *Comercial*.
- **Casa / Apartamento / Comercial**
 - Tabelas que especializam a entidade *Imovel*, armazenando características específicas conforme o tipo. Por exemplo, *Casa* inclui quintal e pavimentos; *Apartamento* inclui andar e condomínio; *Comercial* possui salas, elevadores, etc.
- **Endereco**
 - Relaciona-se com um imóvel e armazena sua localização completa: rua, bairro, cidade, estado, etc.
 - Cada imóvel possui um único endereço.

- **Anuncio**
 - Representa o anúncio ativo ou inativo de um imóvel na plataforma. Está relacionado ao imóvel e ao anunciante.
 - Um imóvel pode ter diversos anúncios ao longo do tempo, mas apenas um ativo por vez.
- **Proposta**
 - Registro de uma oferta feita por um cliente (*Perfil_Cliente*) para um determinado anúncio.
 - Inclui valores negociados, observações, data e se a proposta foi aceita ou não.
- **AvaliacaoFinal**
 - Após a negociação e possível contrato, um cliente pode avaliar a experiência com a negociação.
 - Essa avaliação está ligada a uma proposta e armazena nota, comentários e a data.
- **Contrato**
 - Formaliza a negociação entre cliente e anunciante após a aceitação de uma proposta.
 - Define o tipo do contrato (venda ou aluguel), valor final, documentação envolvida e se foi finalizado.
- **Partes**
 - Registra os dados das pessoas envolvidas no contrato: comprador, proprietário e seus cônjuges.
 - Serve como um documento detalhado das partes acordantes.
- **Venda**
 - Especialização de *Contrato* para casos de compra e venda de imóvel.
 - Armazena informações específicas como valor de entrada, forma de parcelamento e data da escritura.
- **Aluguel**
 - Especialização de *Contrato* para locações.
 - Guarda dados como duração do contrato, caução, se há renovação automática e data de término.
- **Notificacao**
 - Armazena mensagens enviadas ao usuário dentro do sistema, como alertas sobre propostas, contratos ou alterações nos imóveis.
 - Inclui data e se a notificação já foi lida.

4 - Metadados

4.1 - Views

4.1.1. - vw_reputacao_anunciante: Esta view calcula e exibe o campo calculável da reputação média dos anunciantes com base nas avaliações recebidas. Ela agrupa os dados por anunciante, mostrando o nome, o total de avaliações e a nota média arredondada para duas casas decimais. Essa visão é essencial para rankear anunciantes e fornecer transparência aos clientes sobre a qualidade dos serviços oferecidos. Como o cálculo é feito na view, há a garantia que a nota de reputacao_atual estará atualizada sempre que consultada.

	idperfil_anunciante integer	nomeanunciante character varying	totalavaliacoes bigint	reputacao_atual numeric
1	3	Flawbert Lorrان	1	5.00
2	4	Luiz Augusto Lopes	2	4.50
3	1	Ana Maria Lima	4	4.00
4	2	José Oliveira	3	3.00

Figura 02 - View Complexa: Reputação média de cada anunciante

4.1.2. - vw_casas_disponiveis: Apresenta todas as casas disponíveis para compra ou aluguel, incluindo detalhes do imóvel (valor, quartos, banheiros, metragem) e características específicas de casas (se é em condomínio, número de pavimentos, tamanho do terreno e se possui quintal). Os resultados são ordenados por valor em ordem decrescente, facilitando a busca por imóveis de maior valor.

	idimovel integer	nome character varying	valor integer	descricao character varying	qntquartos integer	qntbanheiros integer	metragem integer	cidade character varying	estado character varying	iscondominio boolean	qntpavimentos integer	totalterreno integer	quintal boolean
1	1	Casa Terrea	350000	Casa ampla e ventilada	3	2	140	Natal	RN	true	1	200	true

Figura 03 - View Complexa: Todas as casas disponíveis

4.1.3. - vw_apartamentos_disponiveis: Lista apartamentos disponíveis com informações completas, incluindo detalhes do imóvel e atributos específicos de apartamentos (andar, presença de elevador, áreas de lazer, valor do condomínio e

garagem). Assim como nas outras views de imóveis, os resultados são ordenados por valor para facilitar comparações.

	idimovel integer	nome character varying	valor integer	descricao character varying	qntquartos integer	qntbanheiros integer	metragem integer	cidade character varying	estado character varying	andar integer	elevador boolean	descricaoalazer character varying	valorcondominio integer	garagem boolean
1	8	Apartamento recém-entregue	200000	Apto vazio primeiro dono	2	2	75	Parnamirim	RN	5	true	Playground, portaria 24h	750	true

Figura 04 - View Complexa: Todos os apartamentos disponíveis

4.1.4. - vw_comerciais_disponiveis: Mostra imóveis comerciais disponíveis, exibindo características gerais do imóvel e detalhes comerciais como número de salas, existência de elevador, almoxarifado e tipo de estacionamento. Essa view é valiosa para investidores e empresários buscando espaços comerciais adequados às suas necessidades.

	idimovel integer	nome character varying	valor integer	descricao character varying	qntquartos integer	qntbanheiros integer	metragem integer	cidade character varying	estado character varying	qntsalas integer	elevador boolean	almoxarifado boolean	estacionamento character varying
1	3	Salão Comercial	5000	Salão em rua	1	1	32	Natal	RN	2	false	true	2 vagas
2	5	Meia Sala Comercial	2000	Sala com coshare	1	1	15	Parnamirim	RN	1	true	false	Compartilhado com coworking

Figura 05 - View Complexa: Lista de todos os materiais disponíveis

4.1.5. - vw_contratos_ativos: Fornece uma visão consolidada de todos os contratos ativos no sistema, incluindo tipo de contrato, valor final, datas importantes e nomes das partes envolvidas (cliente e imóvel). Essa view é crucial para a gestão operacional, permitindo o monitoramento em tempo real dos negócios em andamento.

	idcontrato integer	tipocontrato character varying	valorfinal integer	datafinalizacao date	dataproposta integer	nome_imovel character varying	nome_cliente character varying
1	1	Venda	350000	[null]	20240620	Casa Terrea	Gabrielle Borja
2	2	Aluguel	5000	[null]	20240622	Salão Comercial	Emily Miller
3	5	Aluguel	1800	2025-05-09	20240610	Meia Sala Comercial	Mariana Timbó
4	6	Venda	580000	[null]	20240530	Casa anos 90	Mariana Timbó
5	8	Venda	200000	[null]	20240601	Apartamento recém-entregue	Anna Clarice Batista
6	10	Aluguel	9800	[null]	20240605	Apartamento Luxo	Alice Barbosa

Figura 06 - View Complexa: Lista de todos os contratos ativos

4.1.6. - vw_detalhes_contrato: Oferece uma visão detalhada de todos os contratos, mostrando informações completas sobre o tipo de contrato, valores, porcentagem do anunciante, status, e dados das partes envolvidas (anunciante e cliente). Essa view serve como um relatório completo para análise de contratos.

	idcontrato integer	tipocontrato character varying	valorfinal integer	porcentagemanunciante integer	isfinalizado boolean	dataproposta integer	nome_imovel character varying	tipoanunciante character varying	nome_anunciante character varying	nome_cliente character varying
1	1	Venda	350000	12	false	20240620	Casa Terrea	Profissional	Ana Maria Lima	Gabrielle Borja
2	2	Aluguel	5000	10	false	20240622	Salão Comercial	Profissional	José Oliveira	Emilly Miller
3	3	Aluguel	2000	10	true	20240615	Meia Sala Comercial	Profissional	José Oliveira	Mariana Timbó
4	4	Aluguel	10000	10	true	20240618	Apartamento Luxo	Profissional	Ana Maria Lima	Gabrielle Borja
5	5	Aluguel	1800	10	false	20240610	Meia Sala Comercial	Profissional	José Oliveira	Mariana Timbó
6	6	Venda	580000	12	false	20240530	Casa anos 90	Iniciante	Flawbert Lorrán	Mariana Timbó
7	7	Aluguel	1200	10	true	20240612	Apartamento Kitnet	Intermediário	Luiz Augusto Lopes	Andriel Fernandes
8	8	Venda	200000	8	false	20240601	Apartamento recém-entregue	Intermediário	Luiz Augusto Lopes	Anna Clarice Batista
9	9	Venda	330000	10	true	20240525	Casa Terrea	Profissional	Ana Maria Lima	Anna Clarice Batista
10	10	Aluguel	9800	10	false	20240605	Apartamento Luxo	Profissional	Ana Maria Lima	Alice Barbosa

Figura 07 - View Complexa: Lista de todos os detalhes de contrato

4.1.7. - vw_imoveis_por_anunciante: Organiza todos os imóveis cadastrados no sistema por anunciante, mostrando nome do anunciante, detalhes do imóvel (nome, valor, metragem) e disponibilidade. Essa visão é útil para anunciantes gerenciarem seu portfólio e para administradores monitorarem a distribuição de imóveis.

	idperfil_anunciante integer	nome_anunciante character varying	idimovel integer	nome_imovel character varying	valor integer	metragem integer	isdisponivel boolean
1	1	Ana Maria Lima	1	Casa Terrea	350000	140	true
2	1	Ana Maria Lima	2	Apartamento Luxo	10000	210	false
3	2	José Oliveira	3	Salão Comercial	5000	32	true
4	2	José Oliveira	4	Salão Comercial Luxo	15000	60	false
5	2	José Oliveira	5	Meia Sala Comercial	2000	15	true
6	3	Flawbert Lorrán	6	Casa anos 90	580000	430	false
7	4	Luiz Augusto Lopes	7	Apartamento Kitnet	1200	24	false
8	4	Luiz Augusto Lopes	8	Apartamento recém-entregue	200000	75	true

Figura 08 - View Complexa: Lista de todos os imóveis por anunciante

4.1.8. - vw_partes_contrato_finalizados: Exibe informações detalhadas sobre contratos finalizados, incluindo dados legais das partes envolvidas (comprador, proprietário e respectivos cônjuges quando aplicável). Essa view é essencial para fins jurídicos e de documentação histórica dos negócios concluídos.

4.1.9. - vw_propostas_completa: Apresenta uma visão abrangente de todas as propostas feitas no sistema, incluindo valores oferecidos, status, informações do cliente, anunciante, imóvel relacionado e detalhes do anúncio. Essa view é fundamental para acompanhar o pipeline de negociações e analisar o comportamento de compra dos clientes.

	idproposta integer	valorinicial character varying	valorproposta character varying	dataproposta integer	observacoes character varying	isaceita boolean	nomecliente character varying	nomeanunciante character varying
1	1	330000	350000	20240620	Negociando valor final	true	Gabrielle Borja	Ana Maria Lima
2	2	4500	5000	20240622	Fechado aluguel por 12 mes...	true	Emilly Miller	José Oliveira
3	3	1800	2000	20240615	Aceita com desconto	true	Mariana Timbó	José Oliveira
4	4	9500	10000	20240618	Tentando renegociar	true	Gabrielle Borja	Ana Maria Lima
5	5	1700	1800	20240610	Proposta de aluguel finalizada	true	Mariana Timbó	José Oliveira
6	6	550000	580000	20240530	Compra direta aceita	true	Mariana Timbó	Flawbert Lorrان
7	7	1000	1200	20240612	Kitnet no centro	true	Andriel Fernandes	Luiz Augusto Lopes
8	8	190000	200000	20240601	Apto novo em condomínio	true	Anna Clarice Batista	Luiz Augusto Lopes
9	9	300000	330000	20240525	Revisando proposta	true	Anna Clarice Batista	Ana Maria Lima
10	10	9500	9800	20240605	Discutindo valores ainda	true	Alice Barbosa	Ana Maria Lima

Continuação:

idimovel integer	valorimovel integer	metragem integer	tipoimovel character varying	cidade character varying	idanuncio integer	tipoanuncio character varying	dataanuncio date
1	350000	140	Casa	Natal	1	VENDA	2025-06-09
3	5000	32	Comercial	Natal	3	ALUGUEL	2025-07-02
5	2000	15	Comercial	Parnamirim	5	ALUGUEL	2024-07-09
2	10000	210	Apartamento	Mossoró	2	ALUGUEL	2025-04-09
5	2000	15	Comercial	Parnamirim	5	ALUGUEL	2024-07-09
6	580000	430	Casa	Natal	6	VENDA	2025-05-09
7	1200	24	Apartamento	Mossoró	7	ALUGUEL	2025-06-18
8	200000	75	Apartamento	Parnamirim	8	VENDA	2025-05-09
1	350000	140	Casa	Natal	1	VENDA	2025-06-09
2	10000	210	Apartamento	Mossoró	2	ALUGUEL	2025-04-09

Figura 10 - View Complexa: Propostas completas

4.2 - Triggers e functions

Para garantir a integridade dos dados e automatizar processos críticos, implementamos um conjunto de triggers e funções que replicam as regras de negócio do sistema. Estas funcionalidades asseguram a consistência das operações e previnem inconsistências nos dados.

4.2.1. Função verificar_similaridade_proposta: Esta função é acionada após a inserção de uma nova proposta e tem como objetivo calcular o grau de similaridade entre a descrição do anúncio e o perfil do cliente. Ela combina a biografia do cliente com seu preço médio e utiliza a função `similarity()` para comparar com o texto do anúncio. O resultado é exibido como NOTICE, permitindo identificar se a proposta está alinhada com as preferências do cliente sem bloquear a operação. Essa funcionalidade é útil para análise de compatibilidade entre ofertas e interesses dos clientes.

4.2.2. Função mostrar_similaridade_contrato: Quando um contrato é marcado como finalizado (`isFinalizado = TRUE`), esta função entra em ação para verificar a coerência entre os termos do contrato e o anúncio original. Ela compara a descrição do anúncio com o

tipo de contrato e valor final, calculando um índice de similaridade. O resultado é exibido como NOTICE, servindo como ferramenta de auditoria para identificar possíveis discrepâncias entre o que foi anunciado e o que foi efetivamente contratado.

4.2.3. Função criar_avaliacao_vazia: Esta função tem uma finalidade operacional importante: ela automaticamente cria um registro vazio na tabela AvaliacaoFinal quando um novo contrato é gerado. O registro contém apenas a data atual e mantém os campos de nota e comentário como NULL, criando um placeholder que será preenchido posteriormente quando o cliente avaliar o serviço. Isso garante que todo contrato tenha um canal aberto para feedback, melhorando o controle de qualidade do sistema.

4.2.4. Função verificar_anunciante_imovel: Uma função crítica para manter a integridade do sistema, ela valida se o anunciante associado a um novo anúncio é de fato o proprietário do imóvel. A consulta verifica a correspondência entre idAnunciante na tabela Anuncio e idAnunciante na tabela Imovel. Caso seja detectada uma inconsistência, a função retorna um erro que impede a conclusão da operação. Essa verificação previne situações onde um usuário não autorizado tentaria anunciar um imóvel que não lhe pertence.

4.2.5. Função atualizar_disponibilidade_imovel: Esta função automatiza a atualização do status de disponibilidade dos imóveis. Quando um contrato é marcado como finalizado, ela automaticamente define isDisponivel como FALSE para o imóvel relacionado, impedindo que ele apareça em novas buscas ou seja incluído em outros contratos. A função inclui mensagens de NOTICE que registram quando a atualização ocorre, proporcionando transparência no processo de mudança de status dos imóveis.

4.2.6. Função impedir_contratos_duplicados: Uma função essencial do sistema, impedir_contratos_duplicados verifica duas condições críticas antes de permitir a criação de um novo contrato: primeiro, confirma se o imóvel ainda está marcado como disponível; segundo, verifica se não existem outros contratos ativos para o mesmo imóvel. Caso alguma dessas condições não seja atendida, a função retorna um erro que bloqueia a criação do contrato. Essa lógica previne situações onde um mesmo imóvel poderia ser alugado ou vendido múltiplas vezes simultaneamente.

4.2.7. Função validar_data_aluguel: Especializada na validação de prazos, esta função compara a data de fim do aluguel com a data de finalização do contrato. Se a data de fim do aluguel for anterior à data do contrato, a função retorna um erro impedindo a operação. Essa verificação garante a consistência temporal dos dados, assegurando que os períodos de aluguel estejam sempre dentro da vigência do contrato correspondente.

4.2.8. Função validar_tipo_contrato: Esta função genérica é utilizada por dois triggers diferentes para garantir a correta associação entre contratos e seus tipos específicos (Aluguel ou Venda). Ela verifica se o tipo registrado na tabela Contrato corresponde à tabela onde está sendo inserido (Aluguel ou Venda). Em caso de incompatibilidade, a função retorna um erro que bloqueia a operação. Essa validação cruzada mantém a integridade referencial entre as tabelas especializadas e a tabela principal de contratos.

5 - Conclusão

Durante a criação do banco de dados, alguns desafios importantes foram encontrados. Um dos principais foi a modelagem correta da relação entre imóveis, contratos e anunciantes, especialmente no que diz respeito à validação das regras de negócio, como impedir que um mesmo imóvel seja vinculado a múltiplos contratos ativos ou garantir que ele não permaneça disponível após ser alugado ou vendido.

Outro ponto de atenção foi a definição das entidades que representam os perfis dos usuários, diferenciando clientes e anunciantes de forma clara, mas ainda permitindo que compartilhassem uma estrutura comum através da entidade **Usuário**. Esse equilíbrio entre especialização e reutilização de dados foi essencial para manter o modelo normalizado e escalável.

Também foram discutidas e implementadas formas de garantir a integridade dos dados temporais, como a verificação da ordem correta das datas nos contratos de aluguel e a atualização automática da disponibilidade dos imóveis. Para isso, o uso de **triggers** se mostrou indispensável.

Por fim, foi necessário refletir cuidadosamente sobre como representar avaliações e reputação dos anunciantes, considerando que esses dados impactam diretamente a confiabilidade da plataforma. A solução adotada incluiu o cálculo automático da média de avaliações e o armazenamento do histórico de feedbacks dos clientes.

Sobretudo, esses desafios, embora complexos, contribuíram para o amadurecimento do projeto, resultando em uma base de dados sólida, coerente e preparada para atender às demandas reais de um sistema de gerenciamento imobiliário.