

Dicionário de dados

Participar da Aliança pela Mobilidade Sustentável

Ingenico do Brasil

The Ingenico logo features the word 'ingenico' in a bold, lowercase, sans-serif font, with a horizontal line above the 'i'.

Bahia, BA
Versão 1.0
Setembro de 2024

Identificação

Título: Dicionario de dados
Projeto: Participar da Aliança pela Mobilidade Sustentável
Data: Setembro 2024
Local: Bahia, BA
Versão: 1.0

Revisões

Data	Alterações / Comentário	Revisor
20.09.2024	Criação do documento.	João Vitor N. Ramos

Equipe de Desenvolvimento

Tutor

Marcelo Silva

Tutor

Rogério de Jesus

Líder Técnico

João Vitor Nascimento Ramos

Desenvolvedores

Albert Silva de Jesus

Danilo da Conceição Santos

Everlan Santos de Rosário

Leonardo Ribeiro Barbosa Santos

Sumário

Sumário	4	
1	INTRODUÇÃO	7
1.1	Objetivo do Dicionário de Dados	7
1.2	Importância do Dicionário de Dados	7
1.3	Escopo do Documento	7
2	ESTRUTURA DO DICIONÁRIO DE DADOS	8
2.1	Definição de Termos	8
2.2	Formato e Organização dos Dados	8
3	METADADOS	9
3.1	Definição de Metadados	9
3.2	Tipos de Metadados	9
3.2.1	Metadados Descritivos	9
3.2.2	Metadados Estruturais	9
3.2.3	Metadados Administrativos	9
4	DIAGRAMA ENTIDADE-RELACIONAMENTO	11
5	TABELAS E CAMPOS	12
5.1	Listagem das Tabelas	12
5.1.1	Tabela Users	12
5.1.2	Tabela Roles	13
5.1.3	Tabela User_Roles	13
5.1.4	Tabela Tokens	14
5.1.5	Tabela Audit_Log	14
5.1.6	Tabela Category	14
5.1.7	Tabela Mail	15
5.1.8	Tabela Vehicle_Type	15
5.1.9	Tabela Brand	16
5.1.10	Tabela Propulsion	16
5.1.11	Tabela Autonomy	16
5.1.12	Tabela Model	17
5.1.13	Tabela Vehicle	17
5.1.14	Tabela Vehicle_Users	18

5.1.15	Tabela Address	18
5.1.16	Tabela category_avg_autonomy_stats	19
5.1.16.0.1	Restrições e Relacionamentos	19
6	TRIGGERS	20
6.1	Função para Atualização	20
6.2	Triggers Associados	21
6.2.1	Tabela vehicle	21
6.2.2	Tabela autonomy	21

Lista de abreviaturas e siglas

BIGSERIAL	Identificador único que é automaticamente incrementado. Usado para chaves primárias.
VARCHAR(n)	Tipo de dado que armazena strings com comprimento máximo definido por <i>n</i> . Utilizado para armazenar informações como nomes e e-mails.
DATE	Armazena datas no formato YYYY-MM-DD. Usado para campos que requerem a data, como a data de nascimento.
TWTZ	TIMESTAMP WITH TIME ZONE: Armazena data e hora, incluindo informações sobre o fuso horário. É útil para registrar momentos específicos em que eventos ocorrem, garantindo que a informação temporal seja precisa independentemente do local onde é acessada.
BOOLEAN	Tipo de dado que pode armazenar valores verdadeiros ou falsos. Usado para indicar estados, como se um usuário está ativado ou se um token está desativado.
PK	Chave Primária: Assegura que cada valor na coluna é único e não nulo, identificando de forma exclusiva cada registro na tabela.
NOTNULL	Impede que uma coluna aceite valores nulos, garantindo que informações essenciais sejam sempre fornecidas.
UNIQUE	Garante que todos os valores em uma coluna sejam diferentes, evitando duplicatas.
FK	Chave Estrangeira: Estabelece um vínculo entre duas tabelas, assegurando a integridade referencial.
DEFAULT	Define um valor padrão para uma coluna quando nenhum valor é especificado durante a inserção.

1 Introdução

1.1 Objetivo do Dicionário de Dados

O objetivo do dicionário de dados é fornecer uma descrição clara e abrangente dos dados utilizados em um sistema ou projeto. Ele serve como um repositório centralizado de informações sobre as estruturas de dados, seus significados, relacionamentos e regras de negócio associadas. O dicionário de dados é uma ferramenta fundamental para garantir que todos os envolvidos no projeto, incluindo desenvolvedores, analistas e stakeholders, tenham uma compreensão comum dos dados, facilitando a comunicação e a colaboração.

1.2 Importância do Dicionário de Dados

A importância do dicionário de dados reside em sua capacidade de promover a consistência e a qualidade dos dados dentro de uma organização. Ele ajuda a evitar ambiguidades e mal-entendidos sobre o significado dos dados, reduzindo o risco de erros na implementação e no uso dos sistemas. Além disso, um dicionário de dados bem estruturado é essencial para a manutenção e evolução dos sistemas ao longo do tempo, pois fornece uma referência confiável para desenvolvedores e analistas que precisam entender as mudanças nos dados.

1.3 Escopo do Documento

Este documento abrange todos os aspectos relevantes do dicionário de dados para o sistema em questão. Ele inclui definições detalhadas das tabelas e campos, metadados associados, relacionamentos entre tabelas, regras de negócio pertinentes e diretrizes para a escrita e organização do código. O escopo também contempla a documentação necessária para garantir que todos os usuários possam acessar e compreender as informações contidas no dicionário de dados, promovendo assim um uso eficiente e eficaz dos dados no sistema.

2 Estrutura do Dicionário de Dados

2.1 Definição de Termos

A definição de termos é uma parte fundamental do dicionário de dados, pois estabelece um vocabulário comum que todos os usuários devem entender. Essa seção inclui definições claras e concisas para termos técnicos e específicos utilizados no contexto do sistema. A padronização da terminologia ajuda a evitar confusões e garante que todos os envolvidos no projeto tenham uma compreensão uniforme dos conceitos abordados.

2.2 Formato e Organização dos Dados

O formato e a organização dos dados referem-se à maneira como as informações são estruturadas e apresentadas dentro do dicionário. Esta seção descreve a estrutura das tabelas, incluindo a disposição das colunas, tipos de dados, tamanhos e restrições. Além disso, aborda como as informações são organizadas para facilitar a navegação e a consulta, garantindo que os usuários possam localizar rapidamente os dados necessários. Um formato bem definido não apenas melhora a legibilidade, mas também contribui para a eficiência no uso dos dados.

3 Metadados

3.1 Definição de Metadados

Metadados são dados que fornecem informações sobre outros dados. Eles descrevem as características, propriedades e contextos dos dados, permitindo que os usuários entendam melhor o conteúdo e a estrutura das informações. Os metadados desempenham um papel crucial na organização, recuperação e gerenciamento de dados, facilitando a busca e a interpretação das informações em sistemas complexos.

3.2 Tipos de Metadados

Os metadados podem ser classificados em diferentes tipos, cada um com sua função específica. Abaixo estão os principais tipos de metadados utilizados em sistemas de informação:

3.2.1 Metadados Descritivos

Os metadados descritivos fornecem informações sobre o conteúdo dos dados, como título, autor, data de criação e resumo. Eles ajudam os usuários a identificar e localizar dados relevantes em um repositório. Esses metadados são especialmente importantes em bibliotecas digitais e bancos de dados, onde a busca por informações específicas é comum.

3.2.2 Metadados Estruturais

Metadados Estruturais Os metadados estruturais descrevem a organização interna dos dados e como eles estão relacionados entre si. Eles incluem informações sobre tabelas, campos, tipos de dados e relacionamentos entre entidades. Esses metadados são essenciais para entender como os dados estão organizados em um banco de dados e para garantir a integridade referencial.

3.2.3 Metadados Administrativos

Os metadados administrativos contêm informações sobre a gestão dos dados, incluindo detalhes sobre direitos autorais, restrições de acesso, histórico

de alterações e informações sobre o ciclo de vida dos dados. Esses metadados são importantes para garantir a conformidade com regulamentações e políticas organizacionais, além de facilitar a manutenção e o gerenciamento dos dados ao longo do tempo.

4 Diagrama Entidade-relacionamento

O diagrama entidade-relacionamento (DER) é uma representação visual das entidades presentes no sistema e de seus relacionamentos. Ele ajuda a ilustrar a estrutura do banco de dados, definindo as tabelas, os campos e as chaves que conectam as entidades. Este diagrama serve como base para o desenvolvimento do modelo de dados e garante a integridade e consistência das informações.

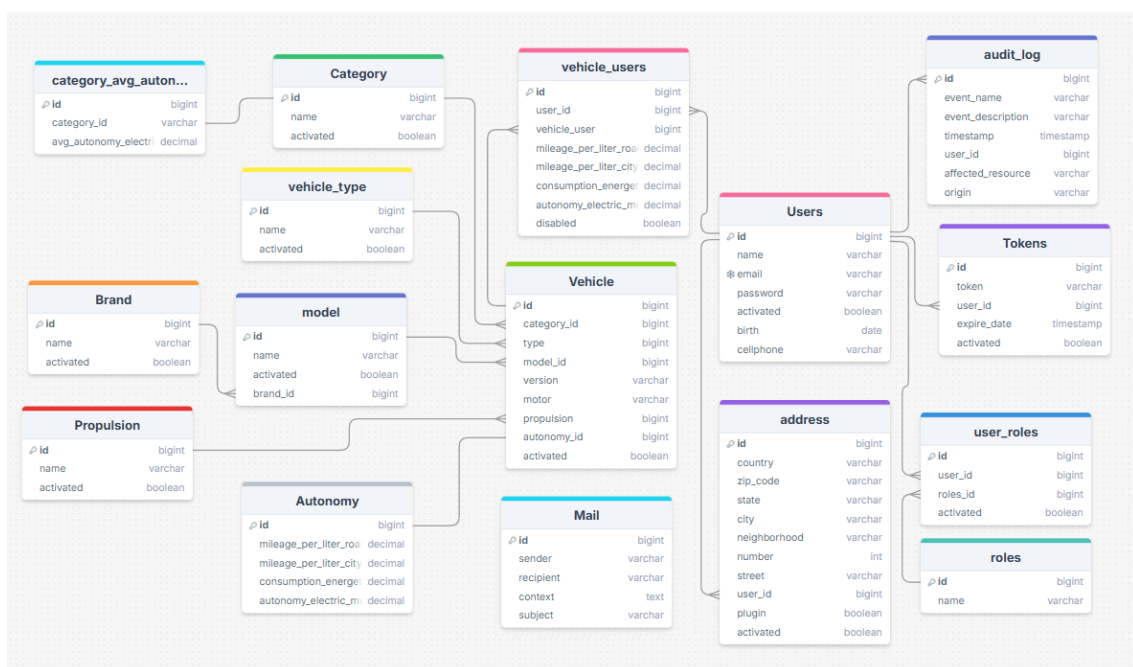


Figura 1 – Diagrama Entidade-Relacionamento (DER) do Sistema

O diagrama apresentado na Figura 1 mostra as principais tabelas do sistema, seus atributos e os relacionamentos entre elas, como a ligação entre os usuários e veículos por meio da tabela **Vehicle_Users**. As chaves primárias e estrangeiras também estão indicadas para garantir a integridade referencial.

5 Tabelas e Campos

5.1 Listagem das Tabelas

Este documento descreve as tabelas principais do banco de dados, incluindo suas definições, campos e relacionamentos. As tabelas a seguir são essenciais para o funcionamento do sistema.

5.1.1 Tabela *Users*

A tabela *users* armazena informações sobre os usuários do sistema. Abaixo estão os campos que compõem esta tabela:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do usuário	<i>PK</i>	-
name	VARCHAR(100)	Nome completo do usuário	<i>NOTNULL</i>	-
email	VARCHAR(100)	Endereço de e-mail do usuário	<i>UNIQUE, NOTNULL</i>	-
password	VARCHAR(255)	Senha do usuário	<i>NOTNULL</i>	-
birth	DATE	Data de nascimento do usuário	<i>NOTNULL</i>	-
cellphone	VARCHAR(255)	Número de celular do usuário	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se o usuário está ativado	<i>NOTNULL</i>	false

Tabela 1 – Tabela Users

A senha do usuário deve ser armazenada de forma criptografada utilizando um tipo de algoritmo. É importante garantir que as senhas sejam sempre comparadas utilizando o mesmo algoritmo de codificação ao realizar operações de autenticação, assegurando que a segurança dos dados dos usuários seja mantida.

5.1.2 Tabela Roles

A tabela *roles* contém informações sobre os diferentes papéis que os usuários podem ter no sistema. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do papel	<i>PK</i>	-
name	VARCHAR(255)	Nome do papel (ex: admin, user)	<i>NOTNULL</i>	-

Tabela 2 – Tabela Roles

Exemplos de valores para o campo *name* incluem:

- **admin:** Usuário com permissões totais, capaz de gerenciar outros usuários e acessar todas as funcionalidades do sistema.
- **user:** Usuário comum com permissões limitadas, restrito ao uso básico do sistema.

Esses papéis ajudam a definir a estrutura de segurança e acesso dentro da aplicação, garantindo que cada usuário tenha apenas as permissões necessárias para suas funções.

5.1.3 Tabela User_Roles

A tabela *user_roles* estabelece um relacionamento entre usuários e seus papéis no sistema. Os campos desta tabela são:

Campo	Tipo	Descrição	Restrições	DEFAULT
user_id	BIGINT	Identificador do usuário	<i>FK, NOT NULL</i>	-
role_id	BIGINT	Identificador do papel	<i>FK, NOT NULL</i>	-
activated	BOOLEAN	Indica se está ativo	<i>NOT NULL</i>	true

Tabela 3 – Tabela User_Roles

As chaves primárias e estrangeiras são definidas da seguinte forma:

- **Chave Primária:** A combinação dos campos *user_id* e *role_id* forma a chave primária da tabela *user_roles*.
- **Chaves Estrangeiras:**
 - *user_id* referencia o campo *id* na tabela *users*, com a opção de exclusão em cascata.
 - *role_id* referencia o campo *id* na tabela *roles*, também com a opção de exclusão em cascata.

5.1.4 Tabela *Tokens*

A tabela *tokens* armazena informações sobre os *tokens* de autenticação gerados para os usuários. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do token.	<i>PK</i>	-
token	VARCHAR(255)	O valor do token gerado.	<i>NOTNULL</i>	-
user_id	BIGINT	Identificador do usuário associado ao token.	<i>FK, NOTNULL</i>	-
expire_date	TWTZ	Data e hora de expiração do token.	-	-
disabled	BOOLEAN	Indica se o token está desativado.	<i>NOTNULL</i>	-

Tabela 4 – Tabela Tokens

A tabela *tokens* é essencial para a gestão de autenticação e autorização no sistema, permitindo que os usuários mantenham sessões seguras e temporárias. Além disso, é possível desativar *tokens* a depender da situação, garantindo que acessos não autorizados possam ser rapidamente revogados quando necessário.

5.1.5 Tabela *Audit_Log*

A tabela *audit_log* é utilizada para gerenciar auditorias no sistema, registrando eventos significativos que ocorrem. Isso permite rastrear ações dos usuários e mudanças importantes, contribuindo para a segurança e a conformidade. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do registro de auditoria.	<i>PK</i>	-
event_name	VARCHAR(150)	Nome do evento que está sendo registrado.	<i>NOTNULL</i>	-
event_description	VARCHAR(255)	Descrição detalhada do evento.	-	-
timestamp	TIMESTAMP	Data e hora em que o evento ocorreu.	-	-
user_id	BIGINT	Identificador do usuário que acionou o evento.	<i>FK</i>	-
affected_resource	VARCHAR(255)	Recurso afetado pelo evento (ex: ID de um objeto).	-	-
origin	VARCHAR(255)	Origem do evento (ex: IP, sistema, etc.).	-	-

Tabela 5 – Tabela Audit_Log

A tabela *audit_log* é fundamental para a gestão de segurança e conformidade no sistema, permitindo a rastreabilidade das ações dos usuários e a análise de eventos críticos. Ela possibilita também a identificação de comportamentos suspeitos e a manutenção da integridade dos dados.

5.1.6 Tabela *Category*

A tabela *category* armazena informações sobre as categorias dos veículos. Os campos da tabela são os seguintes:

A tabela *category* é fundamental para a classificação dos veículos no sistema, permitindo a organização e filtragem de informações relacionadas a dife-

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único da categoria.	<i>PK</i>	-
name	VARCHAR(255)	Nome da categoria.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se a categoria está ativada.	<i>NOTNULL</i>	TRUE

Tabela 6 – Tabela Category

rentes tipos de veículos. Exemplos de categorias incluem "Sub Compact", "Compact", "Medium", "Large", "Extra Large", "Compact SUV", "Large SUV", "Large 4x4 SUV", "Compact Off-Road", "Large Off-Road", "Minivan", "Commercial", "Compact Pickup", "Pickup" e "Sport". A coluna 'activated' permite que o sistema gerencie quais categorias estão ativas e disponíveis para uso, garantindo que apenas as categorias relevantes sejam exibidas aos usuários.

5.1.7 Tabela Mail

A tabela *mail* armazena informações sobre as mensagens de *e-mail* enviadas dentro do sistema. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do e-mail.	<i>PK</i>	-
sender	VARCHAR(255)	Endereço de e-mail do remetente.	-	-
recipient	VARCHAR(255)	Endereço de e-mail do destinatário.	-	-
content	TEXT	Conteúdo da mensagem de e-mail.	-	-
subject	VARCHAR(255)	Assunto do e-mail.	-	-

Tabela 7 – Tabela Mail

5.1.8 Tabela Vehicle_Type

A tabela *vehicle_type* armazena informações sobre os tipos de veículos. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do tipo de veículo.	<i>PK</i>	-
name	VARCHAR(255)	Nome do tipo de veículo.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se o tipo de veículo está ativado.	<i>NOTNULL</i>	TRUE

Tabela 8 – Tabela Vehicle_Type

A tabela *vehicle_type* é fundamental para a classificação dos veículos no sistema, permitindo a organização e filtragem de informações relacionadas a diferentes tipos de veículos. Um exemplo de tipo de veículo que pode ser inserido na tabela é "CAR", que está ativado por padrão.

5.1.9 Tabela Brand

A tabela *brand* armazena informações sobre as marcas de veículos. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único da marca.	<i>PK</i>	-
name	VARCHAR(255)	Nome da marca.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se a marca está ativada.	<i>NOTNULL</i>	TRUE

Tabela 9 – Tabela Brand

A tabela *brand* é fundamental para a classificação das marcas de veículos no sistema, permitindo a organização e filtragem de informações relacionadas a diferentes fabricantes. O sistema contém diversas marcas, incluindo "AUDI", "BMW", "BYD", "CAOA CHERY", "CHEVROLET", "CITROEN", "DONFENG", "FERRARI", "FIAT", "FORD" e "GWM".

5.1.10 Tabela Propulsion

A tabela *propulsion* armazena informações sobre os tipos de propulsão dos veículos. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do tipo de propulsão.	<i>PK</i>	-
name	VARCHAR(255)	Nome do tipo de propulsão.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se o tipo de propulsão está ativado.	<i>NOTNULL</i>	TRUE

Tabela 10 – Tabela Propulsion

A tabela *propulsion* é fundamental para a classificação dos tipos de propulsão dos veículos no sistema, permitindo a organização e filtragem de informações relacionadas a diferentes tecnologias de motorização. O sistema contém os seguintes tipos de propulsão: "Elétrico", "Plug-in" e "Híbrido".

5.1.11 Tabela Autonomy

A tabela *autonomy* armazena informações sobre as autonomias dos veículos elétricos. Os campos da tabela são os seguintes:

A tabela *autonomy* é fundamental para a avaliação da eficiência dos veículos elétricos, permitindo a análise do consumo energético e da autonomia em diferentes condições. Os dados armazenados incluem o consumo em estrada e na cidade, bem como a autonomia em modo elétrico.

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único da autonomia.	<i>PK</i>	-
mileage_per_liter_road	DECIMAL(8, 2)	Consumo de energia em estrada (km/kWh).	-	-
mileage_per_liter_city	DECIMAL(8, 2)	Consumo de energia na cidade (km/kWh).	-	-
consumption_energetic	DECIMAL(8, 2)	Consumo energético (kWh/100km).	-	-
autonomy_electric_mode	DECIMAL(8, 2)	Autonomia em modo elétrico (km).	-	-

Tabela 11 – Tabela Autonomy

5.1.12 Tabela Model

A tabela *model* armazena informações sobre os modelos de veículos. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do modelo.	<i>PK</i>	-
name	VARCHAR(255)	Nome do modelo do veículo.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se o modelo está ativado.	<i>NOTNULL</i>	TRUE
brand_id	BIGINT	Identificador da marca associada ao modelo.	<i>NOTNULL</i>	-

Tabela 12 – Tabela Model

A tabela *model* é fundamental para a classificação dos modelos de veículos no sistema, permitindo a organização e filtragem de informações relacionadas a diferentes marcas e seus respectivos modelos.

Exemplos de modelos contidos no sistema:

- DOLPHIN MINI: Modelo da marca BYD.
- ICAR EQ1: Modelo da marca CAO A CHERY.
- 500E: Modelo da marca FIAT.
- E-JS1: Modelo da marca JAC.
- E-KWID: Modelo da marca RENAULT.
- COOPER: Modelo da marca MINI.
- E-208: Modelo da marca PEUGEOT.
- A3 SPORTBACK: Modelo da marca AUDI.

5.1.13 Tabela Vehicle

A tabela *vehicle* armazena informações sobre os veículos. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do veículo.	<i>PK</i>	-
category_id	BIGINT	Identificador da categoria do veículo.	<i>NOTNULL</i>	-
type_id	BIGINT	Identificador do tipo de veículo.	<i>NOTNULL</i>	-
model_id	BIGINT	Identificador do modelo do veículo.	<i>NOTNULL</i>	-
version	VARCHAR(255)	Versão do veículo.	<i>NOTNULL</i>	-
motor	VARCHAR(255)	Tipo de motor do veículo.	<i>NOTNULL</i>	-
propulsion_id	BIGINT	Identificador do tipo de propulsão.	<i>NOTNULL</i>	-
autonomy_id	BIGINT	Identificador da autonomia do veículo.	<i>NOTNULL</i>	-
activated	BOOLEAN	Indica se o veículo está ativado.	<i>NOTNULL</i>	TRUE
year	BIGINT	Ano do modelo do veículo.	<i>NOTNULL</i>	-

Tabela 13 – Tabela Vehicle

A tabela *vehicle* é fundamental para a classificação e gestão dos veículos no sistema, permitindo a organização e filtragem de informações relacionadas a diferentes categorias, tipos, modelos e suas características.

Exemplos de veículos contidos no sistema:

- ICAR EQ1: Um modelo elétrico da CAO A CHERY, versão TEC, com motor elétrico.
- 500E: Um modelo elétrico da FIAT, versão ICON, com motor elétrico.
- DOLPHIN MINI: Um modelo elétrico da BYD, versão padrão, com motor elétrico.

5.1.14 Tabela Vehicle_Users

A tabela *vehicle_users* armazena informações sobre a relação entre usuários e veículos, permitindo que os usuários registrem seus próprios dados de autonomia. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único da relação usuário-veículo.	<i>PK</i>	-
user_id	BIGINT	Identificador do usuário.	<i>NOTNULL</i>	-
vehicle_id	BIGINT	Identificador do veículo.	<i>NOTNULL</i>	-
mileage_per_liter_road	DECIMAL(8, 2)	Consumo de energia em estrada (km/kWh).	-	NULL
mileage_per_liter_city	DECIMAL(8, 2)	Consumo de energia na cidade (km/kWh).	-	NULL
consumption_energetic	DECIMAL(8, 2)	Consumo energético (kWh/100km).	-	NULL
autonomy_electric_mode	DECIMAL(8, 2)	Autonomia em modo elétrico (km).	-	NULL
activated	BOOLEAN	Indica se a relação está ativada.	<i>NOTNULL</i>	TRUE

Tabela 14 – Tabela Vehicle_Users

5.1.15 Tabela Address

A tabela *address* armazena informações sobre os endereços dos usuários. Os campos da tabela são os seguintes:

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único do endereço.	<i>PK</i>	-
country	VARCHAR(255)	Nome do país.	<i>NOTNULL</i>	-
zip_code	VARCHAR(255)	Código postal do endereço.	<i>NOTNULL</i>	-
state	VARCHAR(255)	Estado do endereço.	<i>NOTNULL</i>	-
city	VARCHAR(255)	Cidade do endereço.	<i>NOTNULL</i>	-
neighborhood	VARCHAR(255)	Bairro do endereço.	<i>NOTNULL</i>	-
number	INTEGER	Número da residência.	<i>NOTNULL</i>	-
street	VARCHAR(255)	Nome da rua.	<i>NOTNULL</i>	-
user_id	BIGINT	Identificador do usuário associado ao endereço.	<i>NOTNULL</i>	-
complement	VARCHAR(255)	Complemento do endereço (opcional).	-	NULL
plugin	BOOLEAN	Indica se o endereço é um plugin.	<i>NOTNULL</i>	FALSE
activated	BOOLEAN	Indica se o endereço está ativado.	<i>NOTNULL</i>	TRUE

Tabela 15 – Tabela Address

A tabela *address* é fundamental para armazenar e gerenciar os endereços dos usuários, facilitando a associação de informações de localização aos perfis dos usuários.

5.1.16 Tabela category_avg_autonomy_stats

A tabela *category_avg_autonomy_stats* armazena informações relacionadas à autonomia média em modo elétrico de veículos por categoria. É utilizada para realizar análises e estatísticas sobre o desempenho médio dos veículos agrupados por categoria.

Campo	Tipo	Descrição	Restrições	DEFAULT
id	BIGSERIAL	Identificador único da tabela.	<i>PK</i>	-
category_id	BIGINT	Identificador da categoria associada.	<i>NOTNULL, FK</i>	-
avg_autonomy_electric_mode	DECIMAL(8, 2)	Autonomia média em modo elétrico (km).	-	NULL

Tabela 16 – Tabela category_avg_autonomy_stats

5.1.16.0.1 Restrições e Relacionamentos

- A chave primária é o campo *id*.
- O campo *category_id* é uma chave estrangeira que referencia o campo *id* da tabela *category*.

A tabela *category_avg_autonomy_stats* é essencial para calcular métricas relacionadas às categorias de veículos, contribuindo para a análise de desempenho e autonomia elétrica.

6 Triggers

Nesta seção, são apresentados os *triggers* e a função associada à tabela *category_avg_autonomy_stats*. Os *triggers* garantem que a autonomia média de veículos por categoria seja atualizada automaticamente após operações de inserção, atualização ou exclusão nas tabelas *vehicle* e *autonomy*.

6.1 Função para Atualização

A função `update_category_avg_autonomy_stats` é responsável por recalcular a autonomia média em modo elétrico para cada categoria. Ela realiza uma atualização na tabela *category_avg_autonomy_stats* com base nos dados das tabelas *vehicle* e *autonomy*.

Listing 6.1 – Função para Atualizar a Tabela de Médias

```
1 CREATE OR REPLACE FUNCTION update_category_avg_autonomy_stats()
2 RETURNS TRIGGER AS $$
3 BEGIN
4     UPDATE category_avg_autonomy_stats
5     SET avg_autonomy_electric_mode = subquery.
        avg_autonomy_electric_mode
6     FROM (
7         SELECT
8             v.category_id,
9             AVG(a.autonomy_electric_mode) AS
                avg_autonomy_electric_mode
10        FROM
11            vehicle v
12        JOIN
13            autonomy a ON v.autonomy_id = a.id
14        WHERE
15            a.autonomy_electric_mode IS NOT NULL
16            AND v.activated = true
17        GROUP BY
18            v.category_id
19    ) AS subquery
20    WHERE category_avg_autonomy_stats.category_id = subquery.
        category_id;
21    RETURN NEW;
22 END;
23 $$ LANGUAGE plpgsql;
```

6.2 Triggers Associados

Os *triggers* abaixo são configurados para garantir que a função de atualização seja executada após alterações nas tabelas *vehicle* e *autonomy*. Eles são aplicados a eventos de inserção, atualização e exclusão.

6.2.1 Tabela *vehicle*

Listing 6.2 – Triggers para a Tabela *vehicle*

```
1 CREATE TRIGGER update_avg_autonomy_stats
2 AFTER INSERT ON vehicle
3 FOR EACH ROW
4 EXECUTE FUNCTION update_category_avg_autonomy_stats();
5
6 CREATE TRIGGER update_avg_autonomy_stats_update
7 AFTER UPDATE ON vehicle
8 FOR EACH ROW
9 EXECUTE FUNCTION update_category_avg_autonomy_stats();
10
11 CREATE TRIGGER update_avg_autonomy_stats_delete
12 AFTER DELETE ON vehicle
13 FOR EACH ROW
14 EXECUTE FUNCTION update_category_avg_autonomy_stats();
```

6.2.2 Tabela *autonomy*

Listing 6.3 – Triggers para a Tabela *autonomy*

```
1 CREATE TRIGGER update_avg_autonomy_stats_autonomy
2 AFTER INSERT ON autonomy
3 FOR EACH ROW
4 EXECUTE FUNCTION update_category_avg_autonomy_stats();
5
6 CREATE TRIGGER update_avg_autonomy_stats_autonomy_update
7 AFTER UPDATE ON autonomy
8 FOR EACH ROW
9 EXECUTE FUNCTION update_category_avg_autonomy_stats();
10
11 CREATE TRIGGER update_avg_autonomy_stats_autonomy_delete
12 AFTER DELETE ON autonomy
13 FOR EACH ROW
14 EXECUTE FUNCTION update_category_avg_autonomy_stats();
```