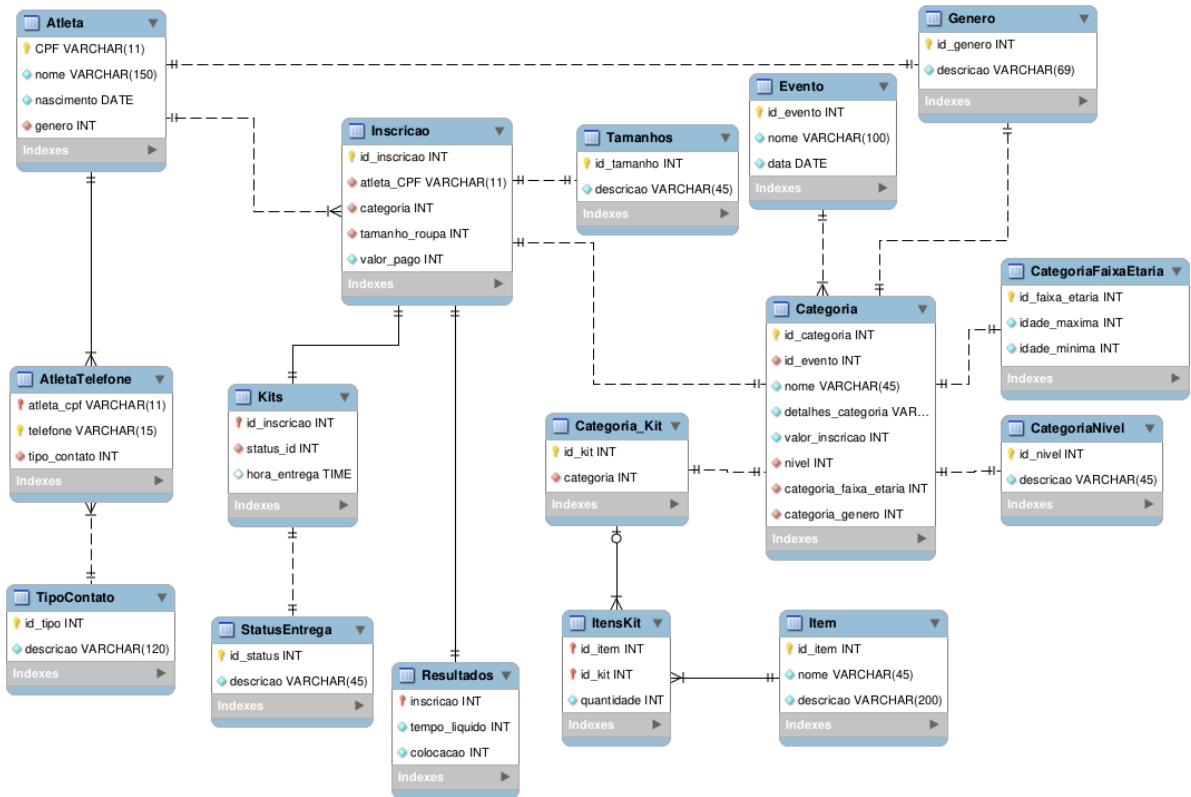


# **Sistema de Gestão para Eventos Esportivos.**

A organização de eventos esportivos como ciclismo e corridas de rua frequentemente apresentam falhas decorrentes da utilização de sistemas fragmentados e pouco relacionados entre si. Inscrições, controle de kits e divulgação dos resultados individuais de cada atleta costumam operar em plataformas distintas, o que causa inconsistência e compromete a atomicidade e consistência dos valores, o que aumenta a probabilidade de erros logísticos, como a perda de kits e inconsistências nos dados dos atletas.

A ausência de um sistema que centralize todo esse processo evidencia a necessidade de uma solução bem estruturada que garanta integridade referencial, consistência dos dados e controle entre transações. A partir disso, propõe a modelagem de um banco de dados relacional voltado à gestão integrada de eventos esportivos, desde o gerenciamento das inscrições, distribuição de kits por categoria até o registro final dos resultados, assegurando a integridade e atomicidade das informações.



Software utilizado : MySQL Workbench

## -Dicionário de Dados:

### Cadastros Principais -

Atleta

AtletaTelefone

**Gestão de Evento -**

Evento

Categoria

Inscricao

**Logística e Kits**

Kits

ItensKit

Item

StatusEntrega

## **Pós-Evento -**

Resultados

## **Tabelas Auxiliares -**

TipoContato

Genero

Tamanhos

CategoriaFaixaEtaria

CategoriaNivel

CategoriaKit

## Cadastrros Principais :

### Atleta

A entidade atleta representa o indivíduo que será cadastrado no sistema, responsável pela participação nos eventos e pelo vínculo ao registro de desempenho.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>CPF</b>	varchar(11)	PK	Identificador do Atleta	-
nome	varchar(150)	-	Nome do Atleta	-
nascimento	date	-	Data de nascimento do Atleta.	-
genero	int	FK	Gênero do Atleta	Ref entidade <a href="#">Genero</a> 1:1

O atributo genero se relaciona com a entidade [Genero](#) por meio de uma chave estrangeira, garantindo a integridade referencial.

A chave primária CPF é armazenada como varchar(11) para preservar zeros à esquerda e evitar inconsistências de formatação, garantindo a integridade do dado ao longo do sistema

Código SQL

```
CREATE TABLE Atleta(
    CPF varchar(11) PRIMARY KEY,
    nome varchar(150) NOT NULL,
    nascimento DATE NOT NULL,
    genero INT NOT NULL,

    CONSTRAINT fk_atleta_genero
        FOREIGN KEY(genero)
        REFERENCES Genero(id_genero)
        ON UPDATE CASCADE
);
```

## AtletaTelefone

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>atleta_cpf</u>	varchar(11)	PK, FK	Identificador do Atleta	Ref entidade <a href="#">Atleta</a> 1:N
<b>telefone</b>	varchar(15)	PK	Número de Telefone	-
<u>tipo_contato</u>	int	FK	Distingue o tipo de telefone cadastrado, ex : emergência, pessoal...	Ref entidade <a href="#">TipoContato</a> 1:N

O atributo telefone é tratado como `varchar(15)` pois telefones não são utilizados em operações aritméticas além de poder conter zeros à esquerda e códigos internacionais.

O atributo tipo\_contato se relaciona com a entidade [TipoContato](#) por meio de uma chave estrangeira, caracterizando uma relação de 1:N, onde um tipo de contato pode estar associado a vários registros de telefone.

### Código SQL

```
CREATE TABLE AtletaTelefone(
    atleta_CPF varchar(11),
    telefone varchar(15),
    tipo_contato int NOT NULL,
    PRIMARY KEY (atleta_CPF, telefone),
    CONSTRAINT fk_atletaTelefone_atleta
        FOREIGN KEY (atleta_CPF)
        REFERENCES Atleta(CPF),
    CONSTRAINT fk_atletaTelefone_tipoContato
        FOREIGN KEY (tipo_contato)
        REFERENCES TipoContato(id_tipo)
);
```

## Gestão de Evento:

### Evento

A entidade **Evento** representa os eventos esportivos gerenciados pelo sistema, sua principal responsabilidade é servir como entidade-pai para as categorias associadas e definir a ocorrência do evento, identificando e estabelecendo uma data.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_evento</b>	int	PK	Identificador do Evento	-
nome	varchar(150)	-	Nome do Evento	-
date	date	-	Data que o evento ocorrerá	-

Código SQL

```
CREATE TABLE Evento(
    id_evento INT PRIMARY KEY AUTO_INCREMENT,
    nome varchar(150) NOT NULL,
    data DATE NOT NULL
);
```

## Categoria

A entidade **Categoria** atua como uma subdivisão organizacional do Evento, definindo características específicas de cada categoria, determinando o valor da inscrição, classificando os participantes dentro de um evento e se vinculando obrigatoriamente a um evento.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>id_categoria</u>	int	PK	Identificador da Categoria	-
<u>id_evento</u>	int	FK	Identifica o evento que a categoria está vinculada.	Ref entidade <a href="#">Evento</a> N:1
nome	varchar(45 )	-	Nome da categoria	-
detalhes_categoria	varchar(200)	-	Descrição detalhada da categoria	-
valor_inscricao	int	-	Valor em Centavos da Inscrição	-
categoria_faixa_etaria	int	FK	Define a faixa etária da categoria	Ref entidade <a href="#">CategoriaFaixaEtaria</a> 1:N
categoria_genero	int	FK	Define o gênero da categoria	Ref entidade <a href="#">Genero</a> 1:N
nivel	int	FK	Define o nível da categoria	Ref entidade <a href="#">CategoriaNivel</a> 1:N

O atributo valor\_inscrição representados em centavos (int) para evitar imprecisões decorrentes de tipos de dados de ponto flutuante.

O atributo id\_evento se relaciona com a entidade [Evento](#) por meio de uma chave estrangeira, caracterizando uma relação N:1 na qual um Evento pode possuir várias Categorias, enquanto uma Categoria depende obrigatoriamente de um único evento.

O atributo categoria\_faixa\_etaria, categoria\_genero e nivel se relacionam com as entidades [CategoriaFaixaEtaria](#), [Genero](#) e

[\*\*CategoriaNivel\*\*](#) para garantir a integridade e padronização dos dados armazenados.

Código SQL

```
CREATE TABLE Categoria(
    id_categoria int PRIMARY KEY AUTO_INCREMENT,
    id_evento int NOT NULL,
    nome varchar(45) NOT NULL,
    detalhes_categoria varchar(200),
    valor_inscricao int NOT NULL,
    categoria_faixa_etaria int NOT NULL,
    categoria_genero int NOT NULL,
    nivel int NOT NULL,

    CONSTRAINT fk_Categoria_faixaEtaria
        FOREIGN KEY (categoria_faixa_etaria)
        REFERENCES CategoriaFaixaEtaria(id_faixa_etaria),

    CONSTRAINT fk_Categoria_genero
        FOREIGN KEY (categoria_genero)
        REFERENCES Genero(id_genero),

    CONSTRAINT fk_Categoria_nivel
        FOREIGN KEY (nivel)
        REFERENCES CategoriaNivel(id_nivel),

    CONSTRAINT fk_Categoria_evento
        FOREIGN KEY (id_evento)
        REFERENCES Evento(id_evento)
);
```

## Inscricao

A entidade **Inscricao** é responsável por associar as entidades [Atleta](#) e [Categoria](#), armazenando informações relacionadas à formalização e registro da participação do atleta, incluindo o valor pago e tamanho de roupa selecionado para o kit.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>id_inscricao</u>	int	PK	Identificador da inscrição	-
<u>atleta_cpf</u>	varchar(11)	FK	Identificador do atleta	Ref entidade <a href="#">Atleta</a> N:1
<u>categoria</u>	int	FK	Identificador da categoria	Ref entidade <a href="#">Categoria</a> N:1
<u>tamanho_roupa</u>	int	FK	Tamanho de Roupa do atleta	Ref entidade <a href="#">Tamanhos</a> N:1
<u>valor_pago</u>	int	-	Registra o valor que o atleta pagou na inscrição.	-

Além de registrar o valor de inscrição na tabela [Categoria](#), é registrado no atributo valor\_pago o valor efetivamente pago pelo atleta, podendo variar pelo uso de cupons ou eventos promocionais.

O atributo atleta\_cpf se relaciona a partir de uma chave estrangeira à entidade [Atleta](#) responsável pela inscrição, caracterizando uma relação N:1, pois um atleta pode efetuar várias inscrições, no entanto a inscrição obrigatoriamente necessita de apenas um único atleta.

O atributo categoria se relaciona com a entidade [Categoria](#) associada à inscrição, uma categoria pode estar associada a várias inscrições, enquanto uma inscrição depende obrigatoriamente de uma única categoria, caracterizando uma relação N:1.

O atributo tamanho\_roupa se relaciona com a entidade [Tamanhos](#) a fim de garantir a integridade referencial e padronização dos valores fornecidos.

## Código SQL

```
CREATE TABLE Inscricao(
    id_inscricao int PRIMARY KEY AUTO_INCREMENT,
    atleta_cpf varchar(11) NOT NULL,
    categoria int NOT NULL,
    tamanho_roupa int NOT NULL,
    valor_pago int NOT NULL,

    CONSTRAINT fk_Inscricao_atleta
        FOREIGN KEY (atleta_cpf)
        REFERENCES Atleta(CPF),

    CONSTRAINT fk_Inscricao_categoria
        FOREIGN KEY (categoria)
        REFERENCES Categoria(id_categoria),

    CONSTRAINT fk_Inscricao_tamanhoRoupa
        FOREIGN KEY (tamanho_roupa)
        REFERENCES Tamanhos(id_tamanho)

);
```

## Logística e Kits:

### Kits

A entidade **Kits** é responsável por gerenciar a entrega dos kits, registrar o horário em que o kit foi entregue e se a entrega foi realizada.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>id_inscricao</u>	int	PK,FK	Identificador da inscrição referente ao kit	Ref entidade <a href="#">Inscricao</a> 1:1
<u>status_id</u>	int	FK	Define o status do Kit (Entregue, Aguardando, Abandonado).	Ref. entidade <a href="#">StatusEntrega</a> N:1
hora_entrega	time	-	Horário em que o kit foi entregue.	-

id\_inscricao é uma chave estrangeira referente à entidade [Inscricao](#) utilizada como chave primária para garantir que cada inscrição tenha apenas um único kit, caracterizando uma relação 1:1.

status\_id se relaciona com a entidade [StatusEntrega](#) a fim de garantir a integridade referencial.

O atributo hora\_entrega é responsável por registrar o horário em que o kit foi entregue. Caso não seja entregue, o valor pode ser nulo.

Código SQL

```
CREATE TABLE Kits(
    id_inscricao int PRIMARY KEY,
    status_id int NOT NULL,
    hora_entrega TIME,

    CONSTRAINT fk_Kits_inscricao
        FOREIGN KEY (id_inscricao)
        REFERENCES Inscricao(id_inscricao),

    CONSTRAINT fk_Kits_status
        FOREIGN KEY (status_id)
        REFERENCES StatusEntrega(id_status)
```

```
);
```

## ItensKit

A entidade **ItensKit** atua como uma entidade associativa entre as entidades [Item](#) e [CategoriaKit](#), além de armazenar quais itens irão compor cada kit e quantidade de cada item.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>id_item</u>	int	PK, FK	Identifica o item	Ref. entidade <a href="#">Item</a> N:1
<u>id_kit</u>	int	PK, FK	Identifica o Kit que o item será adicionado.	Ref. entidade <a href="#">CategoriaKit</a> N:1
quantidade	int	-	Define a quantidade do item em unidades	-

Os atributos id\_item e id\_kit compõem juntamente a chave primária da entidade garantindo a unicidade e impedindo duplicidade de associação.

O atributo id\_item atua como chave estrangeira se relacionando com a entidade [Item](#) identificando qual item será adicionado a um kit específico.

O atributo id\_kit atua como chave estrangeira se relacionando com a entidade [CategoriaKit](#) referenciando o kit qual o item será adicionado.

Código SQL

```
CREATE TABLE ItensKit(
    id_item int PRIMARY KEY AUTO_INCREMENT,
    id_kit int NOT NULL,
    quantidade int NOT NULL,
    CONSTRAINT fk_ItensKit_kit
        FOREIGN KEY (id_kit)
        REFERENCES CategoriaKit(id_kit)
);
```



## Item

A entidade **Item** representa individualmente os itens que podem compor um kit.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_item</b>	int	PK	Identificador do item	-
nome	varchar(45)	-	Nome do item	-
descricao	varchar(200)	-	Descrição do item	-

Código SQL

```
CREATE TABLE Item(
    id_item INT PRIMARY KEY AUTO_INCREMENT,
    nome varchar(45) NOT NULL UNIQUE,
    descricao varchar(200) UNIQUE
);
```

## StatusEntrega

A entidade **StatusEntrega** é responsável por armazenar os possíveis status associados à entrega de um kit, como **Entregue**, **Aguardando** ou **Abandonado**.

Garantindo a padronização e integridade dos valores registrados no sistema.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_status</b>	int	PK	Identificador do status	-
descricao	varchar(45)	-	Descrição do status	-

Código SQL

```
CREATE TABLE StatusEntrega(
    id_status INT PRIMARY KEY AUTO_INCREMENT,
    descricao varchar(45) NOT NULL UNIQUE
);
```

## Pós-Evento :

### Resultados

A entidade Resultados é responsável por registrar o desempenho associado a uma inscrição específica, armazenando o tempo líquido e a colocação do atleta dentro de sua categoria.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<u>inscricao</u>	int	PK,FK	Identificador da Inscrição.	Ref entidade <a href="#">Inscricao</a> 1:1
tempo_liquido	int	-	Registra o tempo líquido associado ao atleta, registrado em segundos.	-
colocacao	int	-	Registra a colocação final do atleta dentro de sua categoria.	-

O atributo inscricao se relaciona com a entidade [Inscricao](#) por meio de uma chave estrangeira, que também é utilizada como chave primária da entidade garantindo que cada inscrição possua no máximo um único resultado. Caracterizando uma relação 1:1.

Código SQL

```
CREATE TABLE Resultados(
    inscricao int PRIMARY KEY,
    tempo_liquido int,
    colocacao int,

    CONSTRAINT fk_Resultados_Inscricao
        FOREIGN KEY (inscricao)
        REFERENCES Inscricao(id_inscricao)
);
```

## Tabelas Auxiliares :

### TipoContato

A entidade **TipoContato** é responsável por armazenar o tipo de contato associado a um atleta a partir da entidade associativa [AtletaTelefone](#), como : **Contato de Emergência, Contato Pessoal, Contato Familiar**, etc. Garantindo a padronização dos dados fornecidos.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_tipo</b>	int	PK	Identificador do tipo de contato	-
descricao	varchar(120)	-	Descreve o tipo de contato	-

Código SQL

```
CREATE TABLE TipoContato(
    id_tipo INT PRIMARY KEY AUTO_INCREMENT,
    descricao varchar(120) NOT NULL UNIQUE
);
```

### Genero

A entidade **Genero** é responsável por armazenar os gêneros que podem ser associados às entidades [Atleta](#) e [Categoria](#), garantindo a padronização dos valores fornecidos.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_genero</b>	int	PK	Identificador do gênero	-
descricao	varchar(120)	-	Descreve o gênero	-

Código SQL

```
CREATE TABLE Genero(
    id_genero INT PRIMARY KEY AUTO_INCREMENT,
    descricao varchar(120) NOT NULL UNIQUE
);
```

## Tamanhos

A entidade **Tamanhos** é responsável por armazenar informações sobre os tamanhos de roupas que o atleta pode solicitar no kit, garantindo padronização dos valores fornecidos.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_tamanho</b>	int	PK	Identificador do tamanho de roupa	-
descricao	varchar(120)	-	Descreve o tamanho	-

Código SQL

```
CREATE TABLE Tamanhos(
    id_tamanho INT PRIMARY KEY AUTO_INCREMENT,
    descricao varchar(120) NOT NULL UNIQUE
);
```

## CategoriaFaixaEtaria

A entidade **CategoriaFaixaEtaria** responsável por definir as faixa-etárias que podem ser associadas a uma [Categoria](#), definindo valores de idade máxima e mínima.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_faixa_etaria</b>	int	PK	Identificador da faixa-etária	-
idade_maxima	int	-	Define a idade máxima	-
idade_minima	int	-	Define a idade mínima	-

Código SQL

```
CREATE TABLE CategoriaFaixaEtaria(
    id_faixa_etaria INT PRIMARY KEY AUTO_INCREMENT,
    idade_maxima INT NOT NULL,
    idade_minima INT NOT NULL
    CHECK (idade_minima < idade_maxima)
);
```

## CategoriaNivel

Entidade responsável por definir os níveis que uma categoria pode assumir, como **Profissional**, **Sport** ou **Amadora**, garantindo a padronização dos dados fornecidos.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_nivel</b>	int	PK	Identificador do nível da categoria	-
descricao	varchar(200)	-	Descreve o nível da categoria	-

Código SQL

```
CREATE TABLE CategoriaNivel(
    id_nivel INT PRIMARY KEY AUTO_INCREMENT,
    descricao varchar(200) NOT NULL UNIQUE
);
```

## CategoriaKit

A entidade **CategoriaKit** representa o kit padrão definido para uma categoria específica, estabelecendo um vínculo entre a categoria e os itens que compõem o kit específico dela.

Atributo	Tipo	Chave	Descrição	Origem / Relacionamento
<b>id_kit</b>	int	PK	Identificador do kit de cada categoria	-
<u>categoria</u>	int	FK	Identifica a categoria que o kit pertence	Rel entidade <a href="#">Categoria</a> 1:1

O atributo categoria é uma chave estrangeira que se relaciona com a entidade [Categoria](#), onde uma categoria pode ter apenas um único kit, enquanto um kit depende obrigatoriamente de uma única categoria, caracterizando uma relação 1:1.

Código SQL

```
CREATE TABLE CategoriaKit(
    id_kit int PRIMARY KEY AUTO_INCREMENT,
    categoria int NOT NULL,

    CONSTRAINT fk_CategoriaKit_Categoria
        FOREIGN KEY (categoria)
        REFERENCES Categoria(id_categoria)
);
```