# LEVANTAMENTO DE REQUISITOS E MODELAGEM DE DADOS - REDE SOCIAL LABORATÓRIO DB

## Introdução

Este documento tem como objetivo detalhar os requisitos e a modelagem de dados necessários para o desenvolvimento de uma rede social para a conclusão da matéria de Laboratório de Banco de Dados. Através deste levantamento, serão especificadas as funcionalidades essenciais da plataforma, as interações entre os usuários e a estrutura de dados que suportará as operações do sistema.

## Requisitos Funcionais

Esta seção descreve as funcionalidades específicas que o sistema deve oferecer para garantir que as necessidades dos usuários sejam atendidas.

## **Cadastro de Usuários**

* O sistema deve permitir que os usuários se registrem na plataforma com nome, e-mail, senha, data de nascimento e foto de perfil.
* O sistema deve validar o e-mail e a senha antes de criar o perfil.
* O cadastro deve gerar um ID único para cada usuário.

## **Postagens**

* O usuário deve ser capaz de criar postagens contendo texto e mídia (imagens ou vídeos).
* O sistema deve permitir que outras pessoas curtam ou comentem nas postagens.
* Cada postagem terá um contador de interações (curtidas e comentários).

## **Interações**

* O sistema deve permitir que um usuário curta ou comente nas postagens de outros usuários.
* O tipo de interação (curtida ou comentário) deve ser registrado e associado à postagem.
* A cada interação, o número de interações de uma postagem deve ser atualizado automaticamente.

## **Grupos e Comunidades**

* O sistema deve permitir a criação de grupos temáticos com nome, descrição e data de criação.
* Cada grupo pode ter múltiplos membros (usuários).
* O sistema deve gerenciar a relação de pertencimento do usuário aos grupos, utilizando uma tabela de associação.

## **Mensage**ns Privadas

* O sistema deve permitir o envio de mensagens privadas entre usuários.
* As mensagens devem ser armazenadas no banco de dados, associadas aos respectivos remetente e destinatário.

## **Notificações**

* O sistema deve gerar notificações para os usuários sempre que ocorrerem interações relevantes (ex: novas curtidas ou comentários em suas postagens).
* O sistema deve registrar o tipo de notificação (curtida, comentário, nova postagem, etc.).

## Modelagem de Dados

Nesta seção, serão descritas as principais entidades, seus atributos e relacionamentos, fundamentais para a implementação da rede social.

## Entidades e Atributos

#### **Usuário**

* **ID\_Usuario** (INT, PK): Identificador único do usuário.
* **Nome\_Usuario** (VARCHAR(100)): Nome completo do usuário.
* **Email** (VARCHAR(50)): Endereço de e-mail do usuário.
* **Senha** (VARCHAR(20)): Senha do usuário.
* **Data\_Nascimento** (DATETIME): Data de nascimento.
* **Foto\_Perfil** (LONGBLOB): Foto do perfil.
* **Num\_Seguidores** (INT): Número de seguidores do usuário.
* **Num\_Seguindo** (INT): Número de pessoas seguidas.

#### **Postagem**

* **ID\_Postagem** (INT, PK): Identificador único da postagem.
* **ID\_Usuario** (INT, FK): Identificador do usuário que fez a postagem.
* **Texto** (TEXT): Texto da postagem.
* **Midia** (LONGBLOB): Arquivo de mídia anexado à postagem.
* **Data\_Hora\_Postagem** (DATETIME): Data e hora da postagem.
* **Num\_Interacoes** (INT): Número total de interações (curtidas e comentários).

#### **Interação**

* **ID\_Interacao** (INT, PK): Identificador único da interação.
* **ID\_Postagem** (INT, FK): Identificador da postagem interagida.
* **ID\_Usuario** (INT, FK): Identificador do usuário que fez a interação.
* **Tipo\_Interacao** (VARCHAR(45)): Tipo de interação (curtida ou comentário).
* **Data\_Hora\_Interacao** (DATETIME): Data e hora da interação.

#### **Comentário**

* **ID\_Comentario** (INT, PK): Identificador único do comentário.
* **ID\_Interacao** (INT, FK): Identificador da interação associada.
* **Texto\_Comentario** (TEXT): Texto do comentário.
* **Data\_Hora\_Comentario** (DATETIME): Data e hora do comentário.

#### **Grupo**

* **ID\_Grupo** (INT, PK): Identificador único do grupo.
* **Nome\_Grupo** (VARCHAR(100)): Nome do grupo.
* **Descricao** (TEXT): Descrição do grupo.
* **Data\_Criacao** (DATETIME): Data de criação do grupo.

#### **Usuário\_has\_Grupo (Tabela de Associação)**

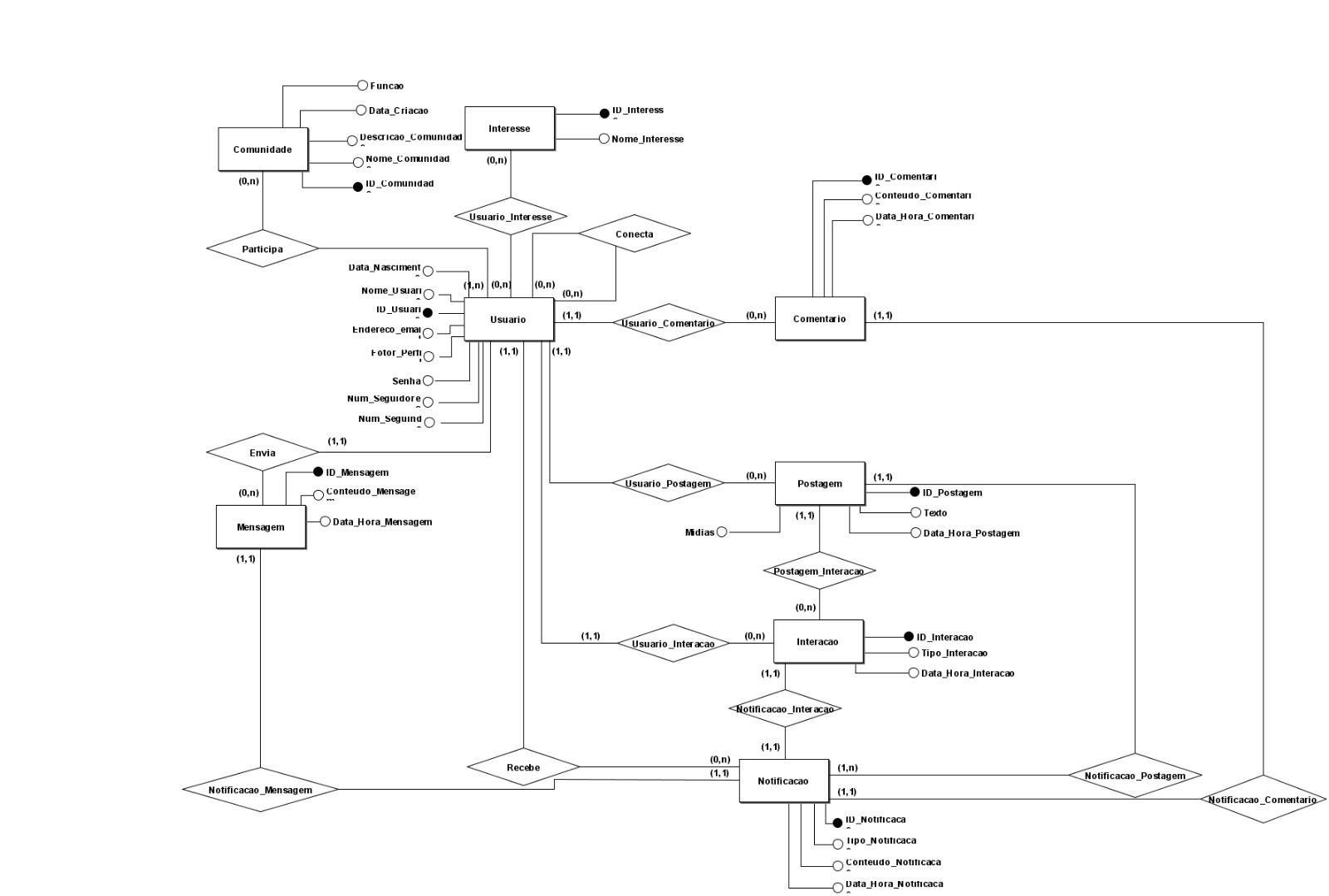
* **ID\_Usuario** (INT, FK): Identificador do usuário.
* **ID\_Grupo** (INT, FK): Identificador do grupo.

## Relacionamentos

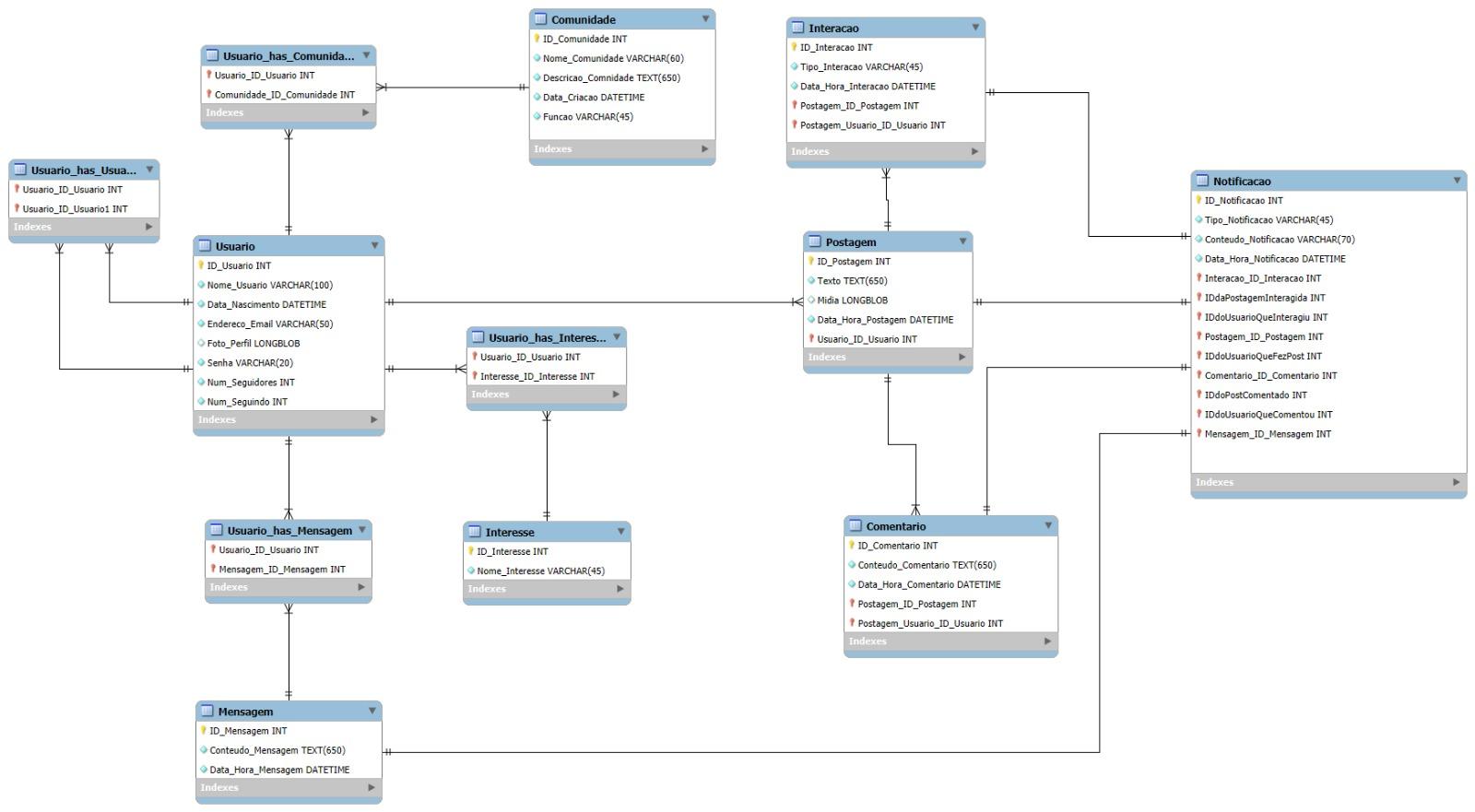
* **Usuário - Postagem**: Relação 1:1 (um para um). ada postagem é associada a apenas um usuário.
* **Postagem - Interação**: Relação 1:1 (um para um). Cada postagem pode ter uma interação. .
* **Interação - Comentário**: Relação 1:1 (um para um). Cada interação pode ter um comentário associado..
* **Usuário - Grupo**: Relação N:M (muitos para muitos). Um usuário pode pertencer a vários grupos e cada grupo pode ter múltiplos usuários.

## Modelagem do Banco

## Modelo Conceitual



## **Modelo** Lógico



## **Modelo** Físico

| -- MySQL Workbench Forward Engineering  SET @OLD\_UNIQUE\_CHECKS=@@UNIQUE\_CHECKS, UNIQUE\_CHECKS=0;  SET @OLD\_FOREIGN\_KEY\_CHECKS=@@FOREIGN\_KEY\_CHECKS, FOREIGN\_KEY\_CHECKS=0;  SET @OLD\_SQL\_MODE=@@SQL\_MODE, SQL\_MODE='ONLY\_FULL\_GROUP\_BY,STRICT\_TRANS\_TABLES,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,NO\_ENGINE\_SUBSTITUTION';  -- -----------------------------------------------------  -- Schema mydb  -- -----------------------------------------------------  -- -----------------------------------------------------  -- Schema mydb  -- -----------------------------------------------------  CREATE SCHEMA IF NOT EXISTS mydb DEFAULT CHARACTER SET utf8 ;  USE mydb ;  -- -----------------------------------------------------  -- Table mydb.Usuario  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Usuario (  ID\_Usuario INT NOT NULL,  Nome\_Usuario VARCHAR(100) NOT NULL,  Data\_Nascimento DATETIME NOT NULL,  Endereco\_Email VARCHAR(50) NOT NULL,  Foto\_Perfil LONGBLOB NULL,  Senha VARCHAR(20) NOT NULL,  Num\_Seguidores INT NOT NULL,  Num\_Seguindo INT NOT NULL,  PRIMARY KEY (ID\_Usuario),  UNIQUE INDEX ID\_Usuario\_UNIQUE (ID\_Usuario ASC) VISIBLE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Comunidade  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Comunidade (  ID\_Comunidade INT NOT NULL,  Nome\_Comunidade VARCHAR(60) NOT NULL,  Descricao\_Comnidade TEXT(650) NOT NULL,  Data\_Criacao DATETIME NOT NULL,  Funcao VARCHAR(45) NOT NULL,  PRIMARY KEY (ID\_Comunidade),  UNIQUE INDEX ID\_Comunidade\_UNIQUE (ID\_Comunidade ASC) VISIBLE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Mensagem  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Mensagem (  ID\_Mensagem INT NOT NULL,  Conteudo\_Mensagem TEXT(650) NOT NULL,  Data\_Hora\_Mensagem DATETIME NOT NULL,  PRIMARY KEY (ID\_Mensagem),  UNIQUE INDEX ID\_Mensagem\_UNIQUE (ID\_Mensagem ASC) VISIBLE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Interesse  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Interesse (  ID\_Interesse INT NOT NULL,  Nome\_Interesse VARCHAR(45) NOT NULL,  PRIMARY KEY (ID\_Interesse),  UNIQUE INDEX ID\_Interesse\_UNIQUE (ID\_Interesse ASC) VISIBLE)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Postagem  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Postagem (  ID\_Postagem INT NOT NULL,  Texto TEXT(650) NOT NULL,  Midia LONGBLOB NULL,  Data\_Hora\_Postagem DATETIME NOT NULL,  Usuario\_ID\_Usuario INT NOT NULL,  PRIMARY KEY (ID\_Postagem, Usuario\_ID\_Usuario),  UNIQUE INDEX ID\_Postagem\_UNIQUE (ID\_Postagem ASC) VISIBLE,  INDEX fk\_Postagem\_Ususario1\_idx (Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Postagem\_Ususario1  FOREIGN KEY (Usuario\_ID\_Usuario)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Interacao  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Interacao (  ID\_Interacao INT NOT NULL,  Tipo\_Interacao VARCHAR(45) NOT NULL,  Data\_Hora\_Interacao DATETIME NOT NULL,  Postagem\_ID\_Postagem INT NOT NULL,  Postagem\_Usuario\_ID\_Usuario INT NOT NULL,  PRIMARY KEY (ID\_Interacao, Postagem\_ID\_Postagem, Postagem\_Usuario\_ID\_Usuario),  UNIQUE INDEX ID\_Interacao\_UNIQUE (ID\_Interacao ASC) VISIBLE,  INDEX fk\_Interacao\_Postagem1\_idx (Postagem\_ID\_Postagem ASC, Postagem\_Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Interacao\_Postagem1  FOREIGN KEY (Postagem\_ID\_Postagem , Postagem\_Usuario\_ID\_Usuario)  REFERENCES mydb.Postagem (ID\_Postagem , Usuario\_ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Comentario  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Comentario (  ID\_Comentario INT NOT NULL,  Conteudo\_Comentario TEXT(650) NOT NULL,  Data\_Hora\_Comentario DATETIME NOT NULL,  Postagem\_ID\_Postagem INT NOT NULL,  Postagem\_Usuario\_ID\_Usuario INT NOT NULL,  PRIMARY KEY (ID\_Comentario, Postagem\_ID\_Postagem, Postagem\_Usuario\_ID\_Usuario),  UNIQUE INDEX ID\_Comentario\_UNIQUE (ID\_Comentario ASC) VISIBLE,  INDEX fk\_Comentario\_Postagem1\_idx (Postagem\_ID\_Postagem ASC, Postagem\_Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Comentario\_Postagem1  FOREIGN KEY (Postagem\_ID\_Postagem , Postagem\_Usuario\_ID\_Usuario)  REFERENCES mydb.Postagem (ID\_Postagem , Usuario\_ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Notificacao  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Notificacao (  ID\_Notificacao INT NOT NULL,  Tipo\_Notificacao VARCHAR(45) NOT NULL,  Conteudo\_Notificacao VARCHAR(70) NOT NULL,  Data\_Hora\_Notificacao DATETIME NOT NULL,  Interacao\_ID\_Interacao INT NOT NULL,  IDdaPostagemInteragida INT NOT NULL,  IDdoUsuarioQueInteragiu INT NOT NULL,  Postagem\_ID\_Postagem INT NOT NULL,  IDdoUsuarioQueFezPost INT NOT NULL,  Comentario\_ID\_Comentario INT NOT NULL,  IDdoPostComentado INT NOT NULL,  IDdoUsuarioQueComentou INT NOT NULL,  Mensagem\_ID\_Mensagem INT NOT NULL,  PRIMARY KEY (ID\_Notificacao, Interacao\_ID\_Interacao, IDdaPostagemInteragida, IDdoUsuarioQueInteragiu, Postagem\_ID\_Postagem, IDdoUsuarioQueFezPost, Comentario\_ID\_Comentario, IDdoPostComentado, IDdoUsuarioQueComentou, Mensagem\_ID\_Mensagem),  UNIQUE INDEX ID\_Notificacao\_UNIQUE (ID\_Notificacao ASC) VISIBLE,  INDEX fk\_Notificacao\_Interacao1\_idx (Interacao\_ID\_Interacao ASC, IDdaPostagemInteragida ASC, IDdoUsuarioQueInteragiu ASC) VISIBLE,  INDEX fk\_Notificacao\_Postagem1\_idx (Postagem\_ID\_Postagem ASC, IDdoUsuarioQueFezPost ASC) VISIBLE,  INDEX fk\_Notificacao\_Comentario1\_idx (Comentario\_ID\_Comentario ASC, IDdoPostComentado ASC, IDdoUsuarioQueComentou ASC) VISIBLE,  INDEX fk\_Notificacao\_Mensagem1\_idx (Mensagem\_ID\_Mensagem ASC) VISIBLE,  CONSTRAINT fk\_Notificacao\_Interacao1  FOREIGN KEY (Interacao\_ID\_Interacao , IDdaPostagemInteragida , IDdoUsuarioQueInteragiu)  REFERENCES mydb.Interacao (ID\_Interacao , Postagem\_ID\_Postagem , Postagem\_Usuario\_ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Notificacao\_Postagem1  FOREIGN KEY (Postagem\_ID\_Postagem , IDdoUsuarioQueFezPost)  REFERENCES mydb.Postagem (ID\_Postagem , Usuario\_ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Notificacao\_Comentario1  FOREIGN KEY (Comentario\_ID\_Comentario , IDdoPostComentado , IDdoUsuarioQueComentou)  REFERENCES mydb.Comentario (ID\_Comentario , Postagem\_ID\_Postagem , Postagem\_Usuario\_ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Notificacao\_Mensagem1  FOREIGN KEY (Mensagem\_ID\_Mensagem)  REFERENCES mydb.Mensagem (ID\_Mensagem)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Usuario\_has\_Comunidade  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Usuario\_has\_Comunidade (  Usuario\_ID\_Usuario INT NOT NULL,  Comunidade\_ID\_Comunidade INT NOT NULL,  PRIMARY KEY (Usuario\_ID\_Usuario, Comunidade\_ID\_Comunidade),  INDEX fk\_Ususario\_has\_Comunidade\_Comunidade1\_idx (Comunidade\_ID\_Comunidade ASC) VISIBLE,  INDEX fk\_Ususario\_has\_Comunidade\_Ususario\_idx (Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Ususario\_has\_Comunidade\_Ususario  FOREIGN KEY (Usuario\_ID\_Usuario)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Ususario\_has\_Comunidade\_Comunidade1  FOREIGN KEY (Comunidade\_ID\_Comunidade)  REFERENCES mydb.Comunidade (ID\_Comunidade)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Usuario\_has\_Mensagem  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Usuario\_has\_Mensagem (  Usuario\_ID\_Usuario INT NOT NULL,  Mensagem\_ID\_Mensagem INT NOT NULL,  PRIMARY KEY (Usuario\_ID\_Usuario, Mensagem\_ID\_Mensagem),  INDEX fk\_Ususario\_has\_Mensagem\_Mensagem1\_idx (Mensagem\_ID\_Mensagem ASC) VISIBLE,  INDEX fk\_Ususario\_has\_Mensagem\_Ususario1\_idx (Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Ususario\_has\_Mensagem\_Ususario1  FOREIGN KEY (Usuario\_ID\_Usuario)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Ususario\_has\_Mensagem\_Mensagem1  FOREIGN KEY (Mensagem\_ID\_Mensagem)  REFERENCES mydb.Mensagem (ID\_Mensagem)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Usuario\_has\_Interesse  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Usuario\_has\_Interesse (  Usuario\_ID\_Usuario INT NOT NULL,  Interesse\_ID\_Interesse INT NOT NULL,  PRIMARY KEY (Usuario\_ID\_Usuario, Interesse\_ID\_Interesse),  INDEX fk\_Ususario\_has\_Interesse\_Interesse1\_idx (Interesse\_ID\_Interesse ASC) VISIBLE,  INDEX fk\_Ususario\_has\_Interesse\_Ususario1\_idx (Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Ususario\_has\_Interesse\_Ususario1  FOREIGN KEY (Usuario\_ID\_Usuario)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Ususario\_has\_Interesse\_Interesse1  FOREIGN KEY (Interesse\_ID\_Interesse)  REFERENCES mydb.Interesse (ID\_Interesse)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  -- -----------------------------------------------------  -- Table mydb.Usuario\_has\_Usuario  -- -----------------------------------------------------  CREATE TABLE IF NOT EXISTS mydb.Usuario\_has\_Usuario (  Usuario\_ID\_Usuario INT NOT NULL,  Usuario\_ID\_Usuario1 INT NOT NULL,  PRIMARY KEY (Usuario\_ID\_Usuario, Usuario\_ID\_Usuario1),  INDEX fk\_Usuario\_has\_Usuario\_Usuario2\_idx (Usuario\_ID\_Usuario1 ASC) VISIBLE,  INDEX fk\_Usuario\_has\_Usuario\_Usuario1\_idx (Usuario\_ID\_Usuario ASC) VISIBLE,  CONSTRAINT fk\_Usuario\_has\_Usuario\_Usuario1  FOREIGN KEY (Usuario\_ID\_Usuario)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION,  CONSTRAINT fk\_Usuario\_has\_Usuario\_Usuario2  FOREIGN KEY (Usuario\_ID\_Usuario1)  REFERENCES mydb.Usuario (ID\_Usuario)  ON DELETE NO ACTION  ON UPDATE NO ACTION)  ENGINE = InnoDB;  SET SQL\_MODE=@OLD\_SQL\_MODE;  SET FOREIGN\_KEY\_CHECKS=@OLD\_FOREIGN\_KEY\_CHECKS;  SET UNIQUE\_CHECKS=@OLD\_UNIQUE\_CHECKS; |
| --- |