

# Relatório sobre o Projeto Rede Social (Banco de Dados)

## Funcionalidades Principais

### Usuário:

Está praticamente no projeto todo, usado para fazer cadastro e login do site, qualquer dado (grupo, post é perfil) lançado no sistema vai estar de algum jeito conectado ao ID do usuário.

### Grupo:

Suas funcionalidades principais é fazer o cadastro do grupo (nome e descrição), ao usuário que fizer esse grupo automaticamente vai ser o Administrador é ao outras pessoas que entrarem vão ser somente membro.

### Post:

Função de receber dados diferentes (**Texto, Texto é imagem, Imagem, Vídeo**) é jogar essas informações na página home (menu).

## Funcionalidades Secundarias

### Mensagens, comments (comentário), replies (reposta):

São tabelas diferentes, mas com a mesma lógica de comunicação entre usuários. As **mensagens** são voltadas para conversas dentro dos **grupos**, enquanto os **comentários** são feitos nas postagens da home e as **respostas** (replies) são utilizadas para interagir diretamente com os comentários. Todas essas interações são vinculadas a um usuário e armazenam a data e o conteúdo do texto.

### Avaliação (like, dislikes):

Funcionalidade que permite aos usuários avaliarem tanto postagens quanto comentários, por meio de **like e dislikes**. Isso gera maior engajamento na rede e permite identificar conteúdos mais relevantes. As reações são armazenadas separadamente em tabelas específicas (**post\_reactions e comment\_reactions**).

## Relacionamentos

### 1. usuarios → post

→ (1 pra N): Um usuário pode criar muitos posts.

 post.user\_id → usuarios.id

### 2. usuarios → comments

→ (1 pra N): Um usuário pode fazer muitos comentários.

 comments.user\_id → usuarios.id

3. usuarios → replies

→ (1 pra N): Um usuário pode fazer muitas respostas.

 replies.user\_id → usuarios.id

4. usuarios → post\_reactions

→ (1 pra N): Um usuário pode reagir a muitos posts.

 post\_reactions.user\_id → usuarios.id

5. usuarios → comment\_reactions

→ (1 pra N): Um usuário pode reagir a muitos comentários.

 comment\_reactions.user\_id → usuarios.id

6. usuarios → usuarios\_grupos

→ (1 pra N): Um usuário pode participar de muitos grupos.

 usuarios\_grupos.usuario\_id → usuarios.id

7. usuarios → grupos (como administrador)

→ (1 pra 1): Um usuário pode ser administrador de apenas um grupo.

 grupos.administrador\_id → usuarios.id

---

■ grupos:

8. grupos → mensagens

→ (1 pra N): Um grupo pode ter muitas mensagens.

 mensagens.grupo\_id → grupos.id

9. grupos → usuarios\_grupos

→ (1 pra N): Um grupo pode ter muitos usuários.

 usuarios\_grupos.grupo\_id → grupos.id

---

■ usuarios\_grupos:

10. usuarios\_grupos é a tabela de associação (N pra N) entre:

- usuarios e grupos

---

■ post:

11. post → comments

→ (N pra N): Um post pode ter vários comentários.

 comments.post\_id → post.id

## 12. post → post\_reactions

→ (1 pra N): Um post pode ter várias reações.

 post\_reactions.post\_id → post.id

---

## comments:

### 13. comments → replies

→ (N pra N): Muitos comentários podem ter várias respostas.

 replies.comment\_id → comments.id

### 14. comments → comment\_reactions

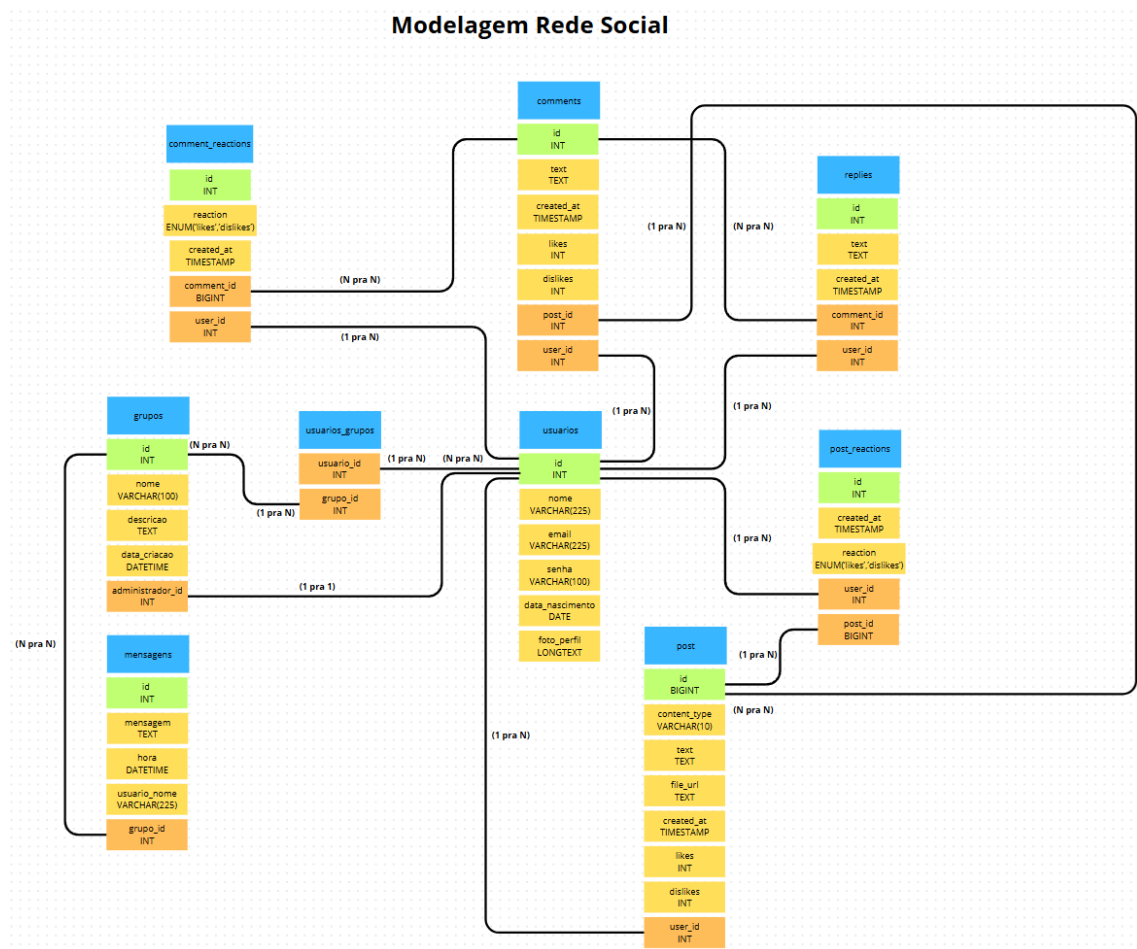
→ (N pra N): Muitos comentários podem ter várias reações.

 comment\_reactions.comment\_id → comments.id

## Resumo dos Relacionamentos

usuários → post	1 pra N
usuários → comments	1 pra N
usuários → replies	1 pra N
usuários → post_reactions	1 pra N
usuários → comment_reactions	1 pra N
usuários → usuarios_grupos	1 pra N
usuários → grupos (como administrador)	1 pra 1
grupos → mensagens	1 pra N
grupos → usuarios_grupos	1 pra N
usuarios_grupos → usuários + grupos	N pra N
post → comments	1 pra N
post → post_reactions	1 pra N
comments → replies	N pra N
comments → comment_reactions	N pra N

## Modelagem Logica



Caso a imagem tenha ficado ruim temos 2 modelos, o [link do canva](#) é o PDF do projeto para baixar.



Modelagem Rede Social.pdf

## Modelo Físico

[Link do Youtube](#) mostrando o Projeto e falando é mostrando o modelo físico do banco.