Projeto Banco de Dados

Mateus Felipe da Silveira Vieira
RA: 10723904

Rayana Pimentel Marques Lopes
RA: -

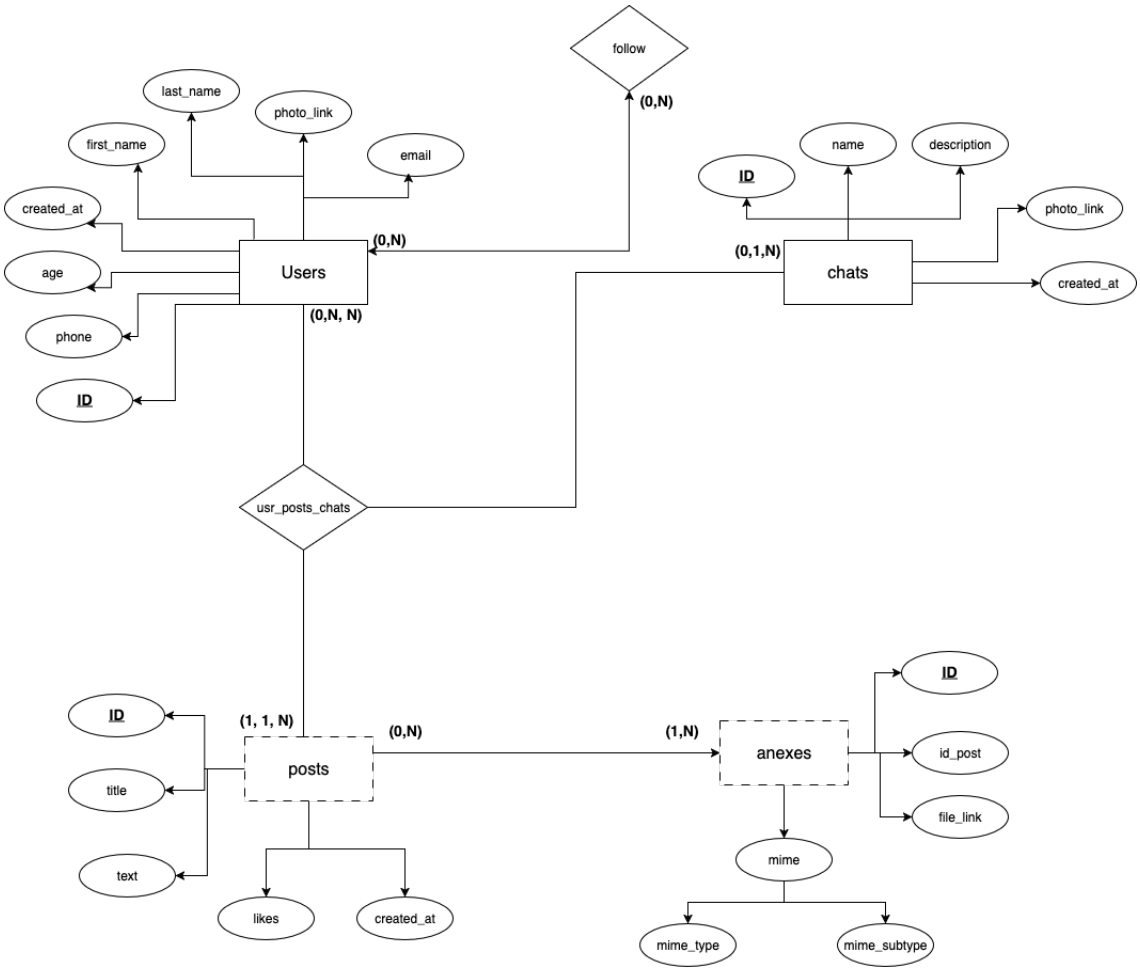
September 27, 2025

Contents

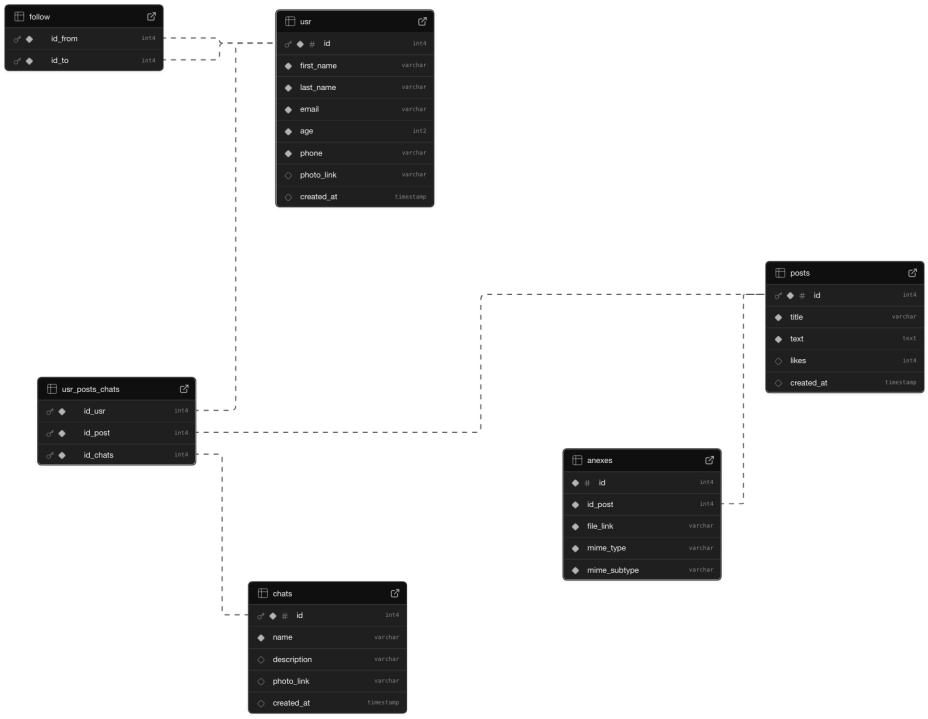
1 Texto de Modelagem

Foi criado uma modelagem na qual há *Usuários*, *Posts*, *Anexos* e *Chats* como entidades, na qual Cada Usuário pode possuir diversos Posts em diversos Chats e cada post deve pertencer a pelo menos um chat e um usuário criador, a entidade Usuário é independente não dependendo de Post nem Chats para ser criada, enquanto Chats depende depende de pelo menos um Usuário para existir. A entidade Anexos está relacionada com Post, com um post podendo possuir muitos anexos, mas cada anexo pertence exclusivamente a um post (por motivos de não ser possível assegurar de uma maneira computacionalmente viável em escala se um anexo é o mesmo entre diversos posts). Além disto nesta rede social há a possibiliadade de um usuário poder seguir e ser seguido por muitos usuários, incluindo uma relação recíproca em que um par de usuários se seguem mutuamente, sendo o sentido de tal relação distinto, distinguindo-se o seguidor do seguido (este processo poderia ser descrito e otimizado em um Banco de Dados exclusivo para gráfos, mas por motivos de aprendizado será utilizado uma modelagem para Banco SQL).

2 Diagrama Modelo Entidade-Relacionamento



3 Diagrama Modelo Relacional



4 Preparação (PostgreSQL)

4.1 Criando Tabelas

4.1.1 Chats

Query SQL:

```
CREATE TABLE chats(
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  name VARCHAR(20) NOT NULL,
  description VARCHAR(50),

  photo_link VARCHAR(20),

  created_at TIMESTAMP DEFAULT NOW()
);
```

4.1.2 Usr

Query SQL:

```
CREATE TABLE usr (
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  first_name VARCHAR(30) NOT NULL,
  last_name VARCHAR(30) NOT NULL,
  email VARCHAR(30) NOT NULL,
  age SMALLINT CHECK(age >= 18) NOT NULL,
  phone VARCHAR(20) NOT NULL,
  photo_link VARCHAR(20),

  created_at TIMESTAMP DEFAULT NOW()
);
```

4.1.3 Posts

Query SQL:

```
CREATE TABLE posts (
```

```
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  title VARCHAR(40) NOT NULL,
  text TEXT NOT NULL,
  likes INTEGER DEFAULT(0),

  created_at TIMESTAMP DEFAULT NOW()
);
```

4.1.4 usr_posts_chats

Query SQL:

```
CREATE TABLE usr_posts_chats(
  id_usr INT REFERENCES usr(id),
  id_post INT REFERENCES posts(id),
  id_chats INT REFERENCES chats(id),
  PRIMARY KEY(id_usr, id_post, id_chats)
);
```

4.1.5 follow

Query SQL:

```
CREATE TABLE follow(
  id_from INT REFERENCES usr(id),
  id_to INT REFERENCES usr(id),

  PRIMARY KEY(id_from, id_to)
);
```

4.1.6 Anexes

Query SQL:

```
CREATE TABLE anexes (
  id INT GENERATED BY DEFAULT AS IDENTITY,
  id_post INT REFERENCES posts(id) NOT NULL,
  file_link VARCHAR(50) NOT NULL,
  mime_type VARCHAR(20) NOT NULL,
  mime_subtype VARCHAR(20) NOT NULL
```

```
);
```

4.2 Inserindo Dados

4.2.1 Chats

Query SQL:

```
INSERT INTO chats(
  name,
  description
)
VALUES(
  'TECNOLOGIES',
  'A chat about tech'
),
(
  'Music',
  'Discussion about music'
),
(
  'Deutsch Sprechen',
  'A sehr schön über Deutsch Sprachen'
),
(
  'Daily life',
  'A chat about daily life'
),
(
  'Travelling',
  'A chat to share your experiences around the world'
);
```

4.2.2 Usr

Query SQL:

```
INSERT INTO usr (
  first_name,
  last_name,
  email,
  age,
```

```
  phone
)
VALUES (
    'Mateus Felipe',
    'Silveira Vieira Schreiner',
    '10723904@mackenzista.com',
    19,
    '11 979246-6876'
  ),
  (
    'Rayana',
    'Pimentel',
    'ra@mackenzie.com',
    19,
    '11 979246-6876'
  ),
  (
    'Cidinha',
    'Delgado',
    'cidinha@musica.com',
    60,
    '11 979248-6876'
  ),
  (
    'Vitinho',
    'Roveri',
    'vitor@pianista.com',
    37,
    '11 979236-6876'
  ),
  (
    'Karen',
    'Comanduli',
    'pio_x@musica.com',
    50,
    '11 979246-6876'
  );
```

4.2.3 Posts

Query SQL:

```
INSERT INTO posts(
  title,
  text
```

```
)
VALUES(
  'What is free (as in freedom) software',
  'Specifically, free software means users have the four essential freedoms: (0) to run the
),
(
  'Why I like Linux',
  'I like linux because is free, is flexible and costumizable'
),
(
  'Best Classical Musics to Sing in a Chor',
  'There a lot of options. However Bach, Mozart and Beethoven are good choiches'
),
(
  'How theach music for children',
  'Fist is important a lot of patience and perceverance'
),
(
  'My day today',
  'I have a pretty good day today. I awakend and ...'
);
```

4.2.4 usr_posts_chats

Query SQL:

```
INSERT INTO usr_posts_chats(
  id_usr,
  id_post,
  id_chats
)
VALUES(1,1,1),(1,2,1),(5,3,2),(3,4,2),(2,5,4);
```

4.2.5 follow

Query SQL:

```
INSERT INTO follow(
  id_from,
  id_to
)
VALUES(1,5), (2,1), (1,2), (4,5), (4,3);
```

4.2.6 Anexes

Query SQL:

```
INSERT INTO anexes(
  id_post,
  file_link,
  mime_type,
  mime_subtype
)
VALUES(
  1,
  'https://www.gnu.org/philosophy/philosophy.html.en',
  'html',
  'hyperlink'
),
(
  1,
  'https://www.gnu.org/graphics/agplv3-155x51.png',
  'image',
  'png'
),
(
  2,
  'https://www.kernel.org/theme/images/logos/tux.png',
  'image',
  'png'
),
(
  4,
  'https://freesvg.org/img/noun_project_928.png',
  'image',
  'png'
);
```

5 Testando com Dados

5.1 Tabela Chats - (Grupos)

Query SQL:

```
SELECT id, name, description, created_at FROM chats;
```

id	name	description	created_at
1	TECNOLOGIES	A chat about tech	2025-09-07 18:21:20.70906
2	Music	Discussion about music	2025-09-07 18:21:20.70906
3	Deutsch Sprechen	A sehr schön über Deutsch Sprachen	2025-09-07 18:21:20.70906
4	Daily life	A chat about daily life	2025-09-07 18:21:20.70906
5	Travelling	A chat to share your experiences around the world	2025-09-07 18:21:20.70906

5.2 Tabela Usr - (Usuários)

Query SQL:

```
SELECT id, first_name, last_name, email, age, phone FROM usr;
```

id	first_name	last_name	email	age	phone
1	Mateus Felipe	Silveira Vieira Schreiner	10723904@mackenzista.com	19	11 979246-6876
2	Rayana	Pimentel	ra@mackenzie.com	19	11 979246-6876
3	Cidinha	Delgado	cidinha@musica.com	60	11 979248-6876
4	Vitinho	Roveri	vitor@pianista.com	37	11 979236-6876
5	Karen	Comanduli	pio_x@musica.com	50	11 979246-6876

5.3 Tabela Posts - (Mensagens)

Query SQL:

```
SELECT id, title, SUBSTRING(text FROM 1 FOR 20) || '...'
AS text__init, likes, created_at FROM posts;
```

id	title	text__init	likes	created_at
1	What is free (as in freedom) software	Specifically,...	0	2025-09-07 18:42:50.005913
2	Why I like Linux	I like linux ...	0	2025-09-07 18:42:50.005913
4	How theach music for children	Fist is impor...	0	2025-09-07 18:42:50.005913
5	My day today	I have a pret...	0	2025-09-07 18:42:50.005913
3	Best Classical Musics to Sing in a Chor	There a lot o...	1	2025-09-07 18:42:50.005913

5.4 Tabela usr_posts_chats (Relação: Usr <N> - Posts <N> - Chats <N>)

Query SQL:

```
SELECT id_usr, id_post, id_chats FROM usr_posts_chats;
```

id_usr	id_post	id_chats
1	1	1
1	2	1
5	3	2
3	4	2
2	5	4

```
SELECT posts.title, SUBSTRING(text FROM 1 FOR 13) || '...' AS text,
chats.name AS chat__name,
first_name, usr.id AS usr__id  FROM usr

INNER JOIN usr_posts_chats ON usr.id = usr_posts_chats.id_usr
INNER JOIN posts ON posts.id = usr_posts_chats.id_post
INNER JOIN chats ON chats.id = usr_posts_chats.id_chats;
```

title	text	chat__name	first_name	usr__id
What is free (as in freedom) software	Specifically,...	TECNOLOGIES	Mateus Felipe	1
Why I like Linux	I like linux ...	TECNOLOGIES	Mateus Felipe	1
Best Classical Musics to Sing in a Chor	There a lot o...	Music	Karen	5
How theach music for children	Fist is impor...	Music	Cidinha	3
My day today	I have a pret...	Daily life	Rayana	2

5.5 Tabela follow (Relação: Usr <N> - Usr <N> [auto-relação])

Query SQL:

```
SELECT id_from, id_to FROM follow;
```

id_from	id_to
1	5
2	1
1	2
4	5
4	3

```
SELECT usr_from.id AS id_from, follow.id_to,
usr_from.first_name AS name_from, usr_to.first_name AS name_to FROM usr usr_from
INNER JOIN follow ON usr_from.id = follow.id_from
INNER JOIN usr usr_to ON usr_to.id = follow.id_to;
```

[1]

id_from	id_to	name_from	name_to
1	5	Mateus Felipe	Karen
2	1	Rayana	Mateus Felipe
1	2	Mateus Felipe	Rayana
4	5	Vitinho	Karen
4	3	Vitinho	Cidinha

[1] Para essa query SQL foi nessesário utilizar um nome intermediário, pois como há uma relação N:N entre a mesma entidade Usr, através da tabela de ligação follow, é presiso diferenciar as colunas em questão do usuário que segue (from) e do que é seguido (to), caso o contrário ocorreia que somente uma coluna resultante seria mostrada pois teriam o mesmo nome, apesar de possuirem valores diferentes

5.6 Tabela Anexes (Relação: Anexes <N> - Posts <1>)

Query SQL:

```
SELECT id, id_post, file_link, mime_type, mime_subtype FROM anexes;
```

id	id_post	file_link	mime_type	mime_subtype
1	1	https://www.gnu.org/philosophy/philosophy.html.en	html	hyperlink
2	1	https://www.gnu.org/graphics/agplv3-155x51.png	image	png
3	2	https://www.kernel.org/theme/images/logos/tux.png	image	png
4	4	https://www.kernel.org/theme/images/logos/tux.png	image	png

```
SELECT anexes.id AS anexes__id, posts.id AS post__id, posts.title,
SUBSTRING(file_link FROM 1 FOR 15) || '...' AS file_link,
mime_type, mime_subtype FROM anexes
INNER JOIN posts ON anexes.id_post = posts.id;
```

anexes__id	post__id	title	file_link	mime_type	mime_subtype
1	1	What is ...	https://www.gnu...	html	hyperlink
2	1	What is ...	https://www.gnu...	image	png
3	2	Why I li...	https://www.ker...	image	png
4	4	How thea...	https://freesvg...	image	png

6 Full script

```sql
-- drop all tables from the last test
DROP TABLE IF EXISTS posts CASCADE;
DROP TABLE IF EXISTS usr CASCADE;
DROP TABLE IF EXISTS chats CASCADE;

DROP TABLE IF EXISTS usr_posts_chats;
DROP TABLE IF EXISTS follow;
DROP TABLE IF EXISTS anexes;

-- Re create tables

CREATE TABLE chats(
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  name VARCHAR(20) NOT NULL,
  description VARCHAR(50),

  photo_link VARCHAR(20),

  created_at TIMESTAMP DEFAULT NOW()
);

CREATE TABLE usr (
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  first_name VARCHAR(30) NOT NULL,
  last_name VARCHAR(30) NOT NULL,
  email VARCHAR(30) NOT NULL,
  age SMALLINT CHECK(age >= 18) NOT NULL,
  phone VARCHAR(20) NOT NULL,
  photo_link VARCHAR(20),

  created_at TIMESTAMP DEFAULT NOW()
);

CREATE TABLE posts (
  id INTEGER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  title VARCHAR(40) NOT NULL,
  text TEXT NOT NULL,
  likes INTEGER DEFAULT(0),

  created_at TIMESTAMP DEFAULT NOW()
```

```
);

CREATE TABLE follow(
  id_from INT REFERENCES usr(id),
  id_to INT REFERENCES usr(id),

  PRIMARY KEY(id_from, id_to)
);

CREATE TABLE usr_posts_chats(
  id_usr INT REFERENCES usr(id),
  id_post INT REFERENCES posts(id),
  id_chats INT REFERENCES chats(id),
  PRIMARY KEY(id_usr, id_post, id_chats)
);


CREATE TABLE anexes (
  id INT GENERATED BY DEFAULT AS IDENTITY,
  id_post INT REFERENCES posts(id) NOT NULL,
  file_link VARCHAR(50) NOT NULL,
  mime_type VARCHAR(20) NOT NULL,
  mime_subtype VARCHAR(20) NOT NULL
);



-- insert dummy data

INSERT INTO chats(
  name,
  description
)
VALUES(
  'TECNOLOGIES',
  'A chat about tech'
),
(
  'Music',
  'Discussion about music'
),
(
  'Deutsch Sprechen',
  'A sehr schön über Deutsch Sprachen'
),
(
```

```
  'Daily life',
  'A chat about daily life'
),
(
  'Travelling',
  'A chat to share your experiences around the world'
);


INSERT INTO usr (
  first_name,
  last_name,
  email,
  age,
  phone
)
VALUES (
    'Mateus Felipe',
    'Silveira Vieira Schreiner',
    '10723904@mackenzista.com',
    19,
    '11 979246-6876'
  ),
  (
    'Rayana',
    'Pimentel',
    'ra@mackenzie.com',
    19,
    '11 979246-6876'
  ),
  (
    'Cidinha',
    'Delgado',
    'cidinha@musica.com',
    60,
    '11 979248-6876'
  ),
  (
    'Vitinho',
    'Roveri',
    'vitor@pianista.com',
    37,
    '11 979236-6876'
  ),
  (
    'Karen',
```

```
    'Comanduli',
    'pio_x@musica.com',
    50,
    '11 979246-6876'
  );

INSERT INTO follow(
  id_from,
  id_to
)
VALUES(1,5), (2,1), (1,2), (4,5), (4,3);


INSERT INTO posts(
  title,
  text
)
VALUES(
  'What is free (as in freedom) software',
  'Specifically, free software means users have the four essential freedoms: (0) to run the
),
(
  'Why I like Linux',
  'I like linux because is free, is flexible and costumizable'
),
(
  'Best Classical Musics to Sing in a Chor',
  'There a lot of options. However Bach, Mozart and Beethoven are good choiches'
),
(
  'How theach music for children',
  'Fist is important a lot of patience and perceverance'
),
(
  'My day today',
  'I have a pretty good day today. I awakend and ...'
);

INSERT INTO usr_posts_chats(
  id_usr,
  id_post,
  id_chats
)
VALUES(1,1,1),(1,2,1),(5,3,2),(3,4,2),(2,5,4);

INSERT INTO anexes(
```

```
  id_post,
  file_link,
  mime_type,
  mime_subtype
)
VALUES(
  1,
  'https://www.gnu.org/philosophy/philosophy.html.en',
  'html',
  'hyperlink'
),
(
  1,
  'https://www.gnu.org/graphics/agplv3-155x51.png',
  'image',
  'png'
),
(
  2,
  'https://www.kernel.org/theme/images/logos/tux.png',
  'image',
  'png'
),
(
  4,
  'https://freesvg.org/img/noun_project_928.png',
  'image',
  'png'
);

-- add like to see if works
UPDATE posts SET likes = likes + 1 WHERE id = 3;

SELECT id, name, description, created_at FROM chats;  -- chats

SELECT id, first_name, last_name, email, age, phone FROM usr; --usr

SELECT id, title, SUBSTRING(text FROM 1 FOR 13) || '...'
AS text__init, likes, created_at FROM posts; -- posts

-- usr_posts_chats
SELECT id_usr, id_post, id_chats FROM usr_posts_chats;

SELECT posts.title, SUBSTRING(text FROM 1 FOR 13) || '...' AS text,
chats.name AS chat__name,
first_name, usr.id AS usr__id  FROM usr
```

```sql
INNER JOIN usr_posts_chats ON usr.id = usr_posts_chats.id_usr
INNER JOIN posts ON posts.id = usr_posts_chats.id_post
INNER JOIN chats ON chats.id = usr_posts_chats.id_chats;

-- followers
SELECT id_from, id_to FROM follow;

SELECT usr_from.id AS id_from, follow.id_to,
usr_from.first_name AS name_from, usr_to.first_name AS name_to FROM usr usr_from
INNER JOIN follow ON usr_from.id = follow.id_from
INNER JOIN usr usr_to ON usr_to.id = follow.id_to;

-- anexes
SELECT id, id_post, file_link, mime_type, mime_subtype FROM anexes;

SELECT anexes.id AS anexes__id, posts.id AS post__id, SUBSTRING(posts.title FROM 1 FOR 8) ||
SUBSTRING(file_link FROM 1 FOR 15) || '...' AS file_link,
mime_type, mime_subtype FROM anexes
INNER JOIN posts ON anexes.id_post = posts.id;
```
