

PROJETO BANCO DE DADOS

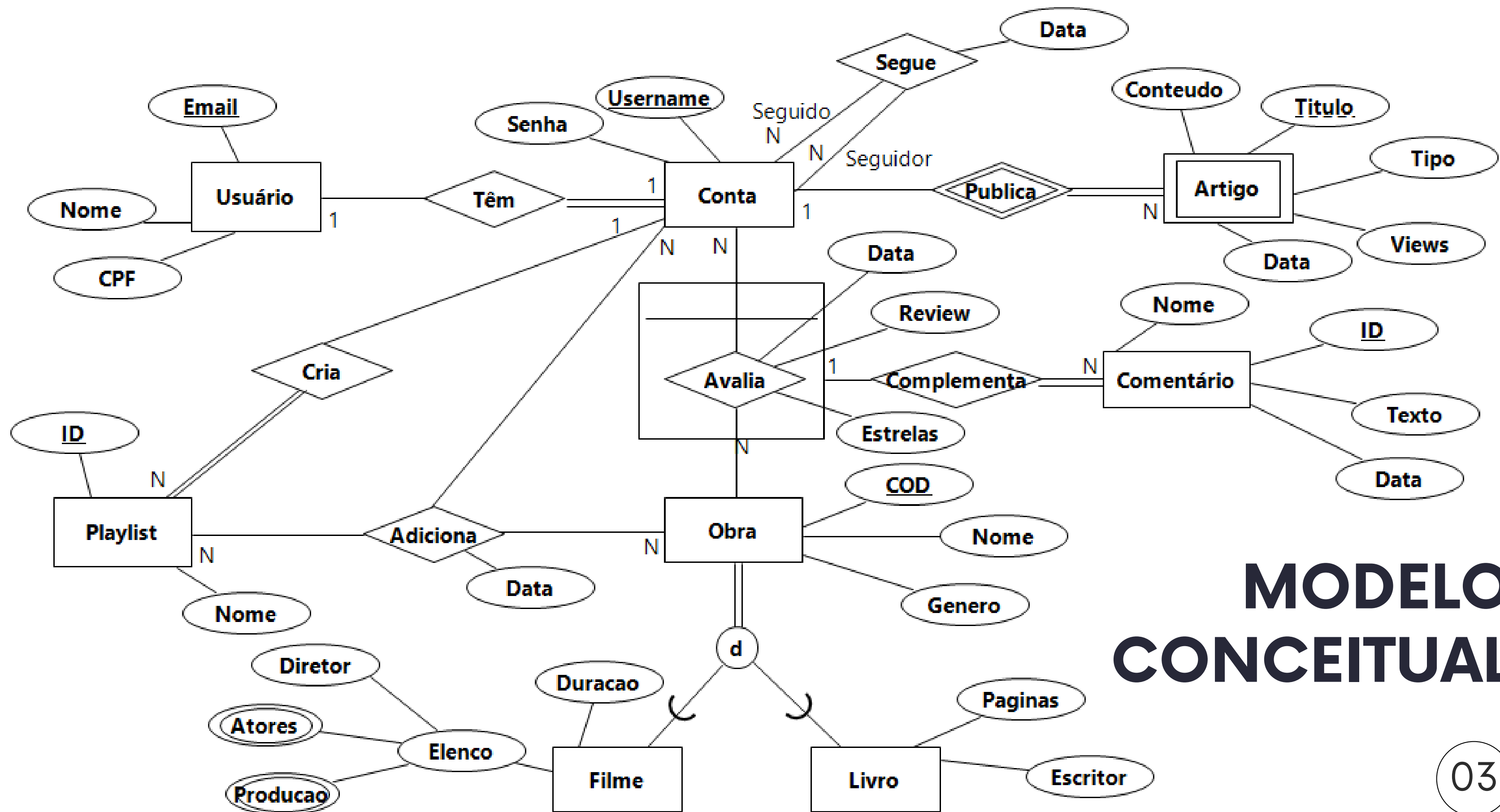
Antônio Henrique
Gabriel Aragão
Geydson Renan

Maria Clara Barretto
Mariana Amorim
Rodrigo Barbosa

LETTERBOXD

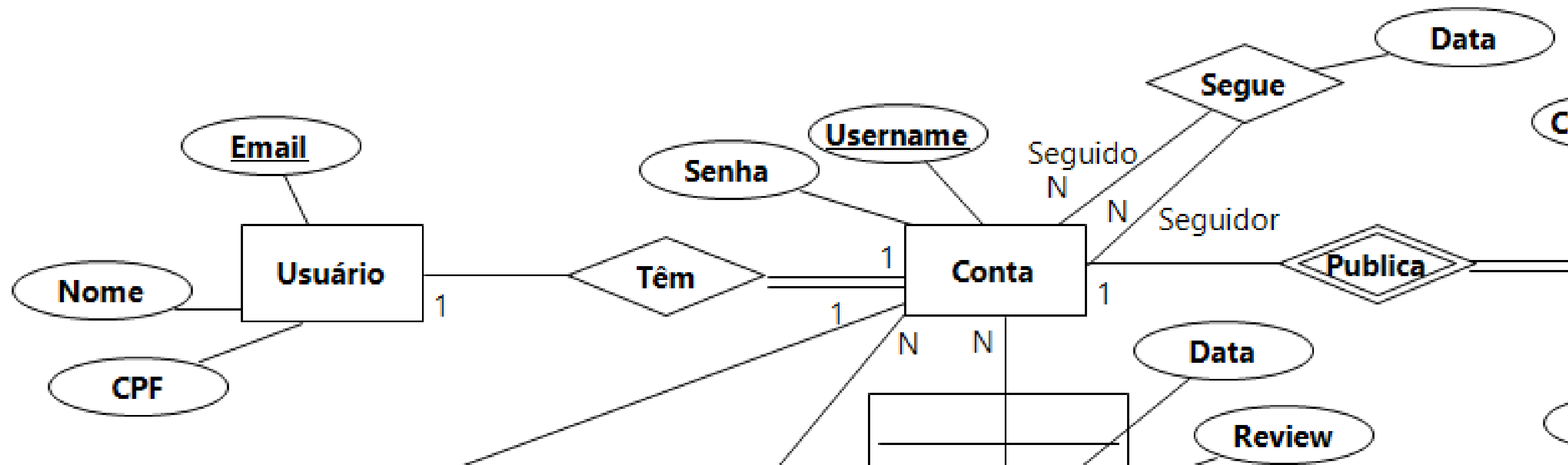
O tema central do nosso Banco de Dados é o aplicativo Letterboxd. Nessa linha, para que possa ser realizada a avaliação de obras (livros e filmes) é necessário que haja o cadastro de uma conta do usuário que irá fazer a avaliação e também o cadastro da obra a ser avaliada. Assim, a conta poderá avaliar, criar playlists, artigos e também realizar comentários de diversas obras.





MODELO CONCEITUAL

MODELO CONCEITUAL + LÓGICO

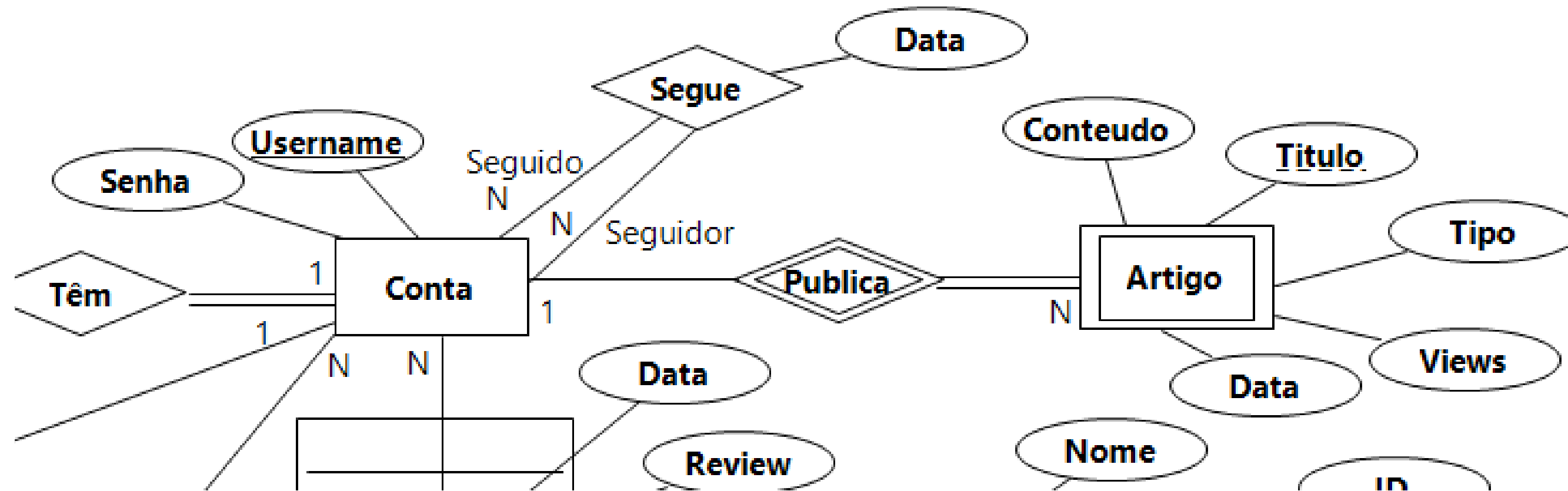


Usuário(email, nome, cpf)

Conta(username, senha, [email]!)
email -> Usuário(email)

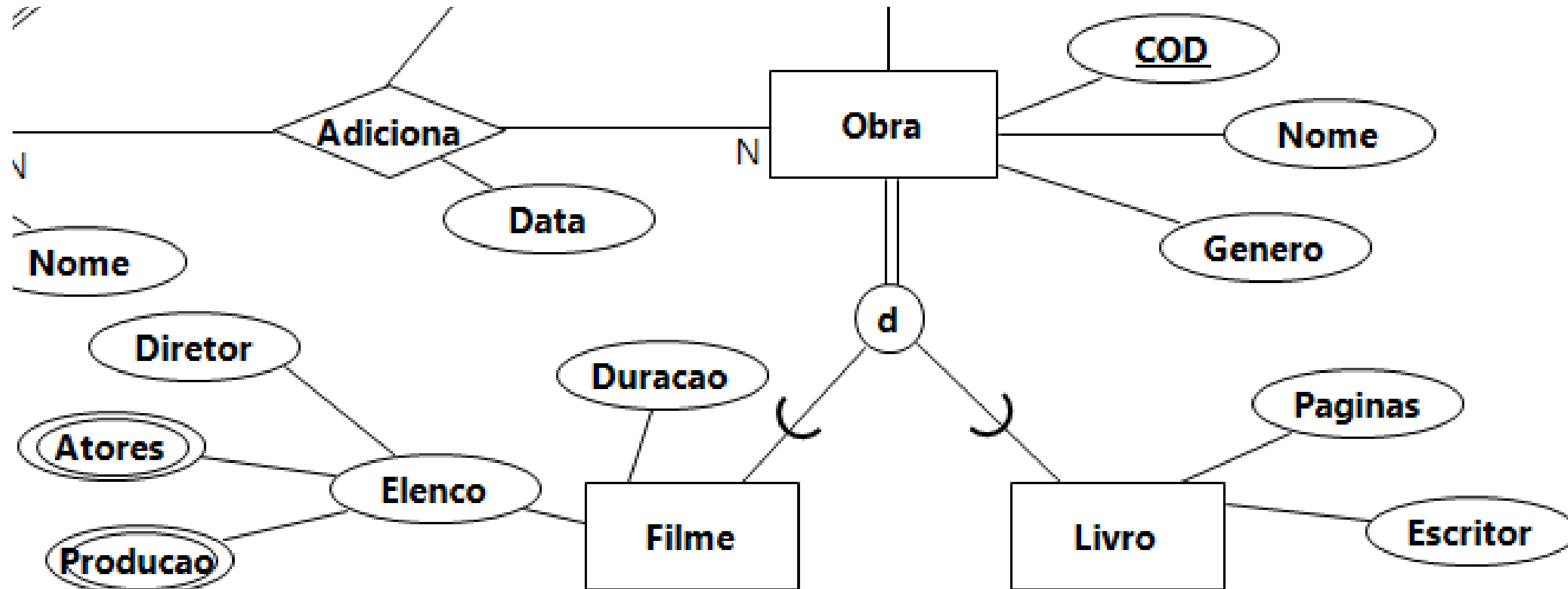
Seguir(segue, seguido, data)
segue -> Conta(username)
seguido -> Conta(username)

MODELO CONCEITUAL + LÓGICO



Artigo(username, título, data, tipo, views, conteúdo)
username -> Conta(username)

MODELO CONCEITUAL + LÓGICO



Obras(cod, nome, gênero)

Filmes(cod, elen_diretor, duracao)
cod -> Obras(cod)

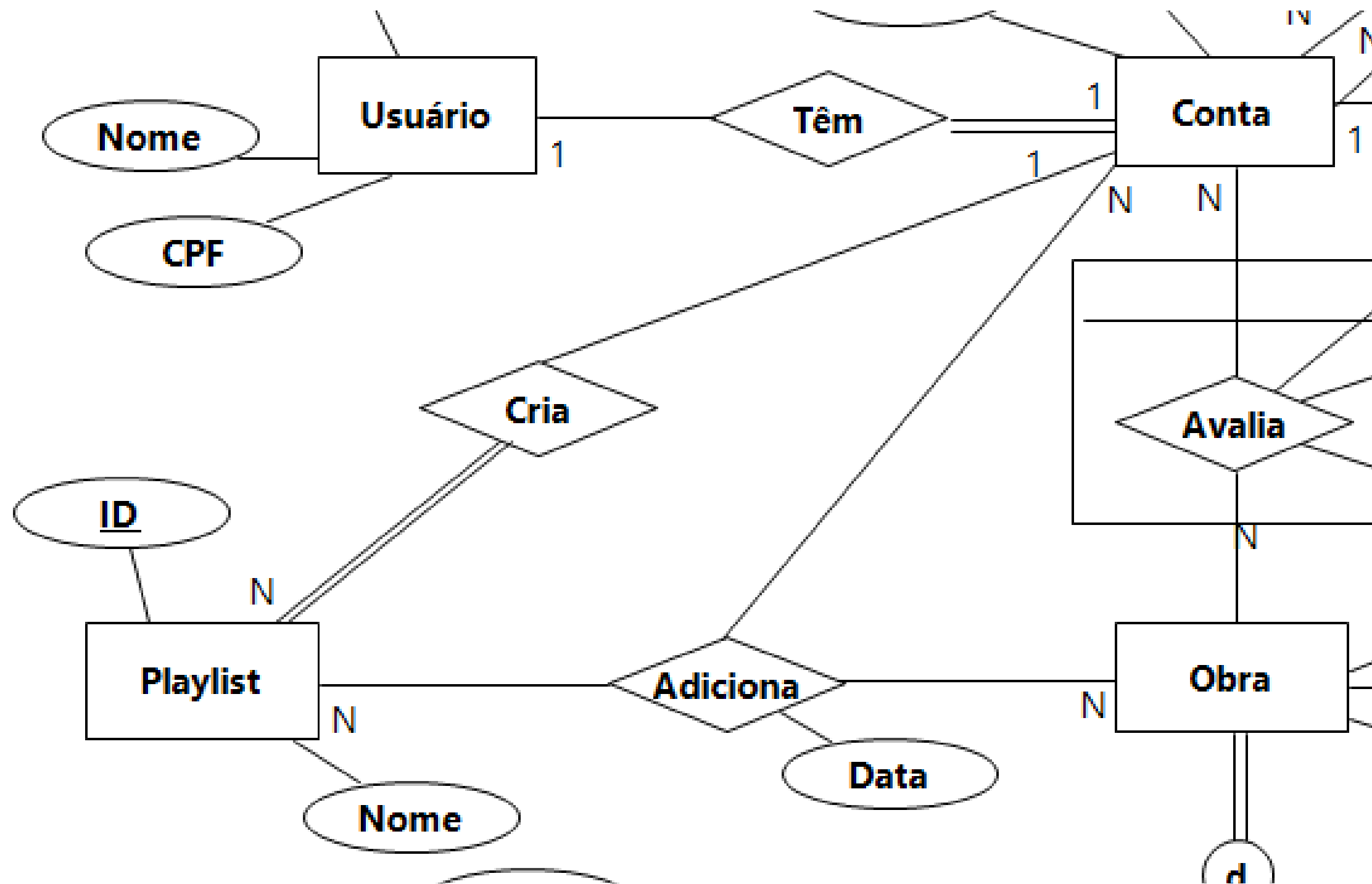
Atores(cod, atores)

cod -> Filmes(cod)

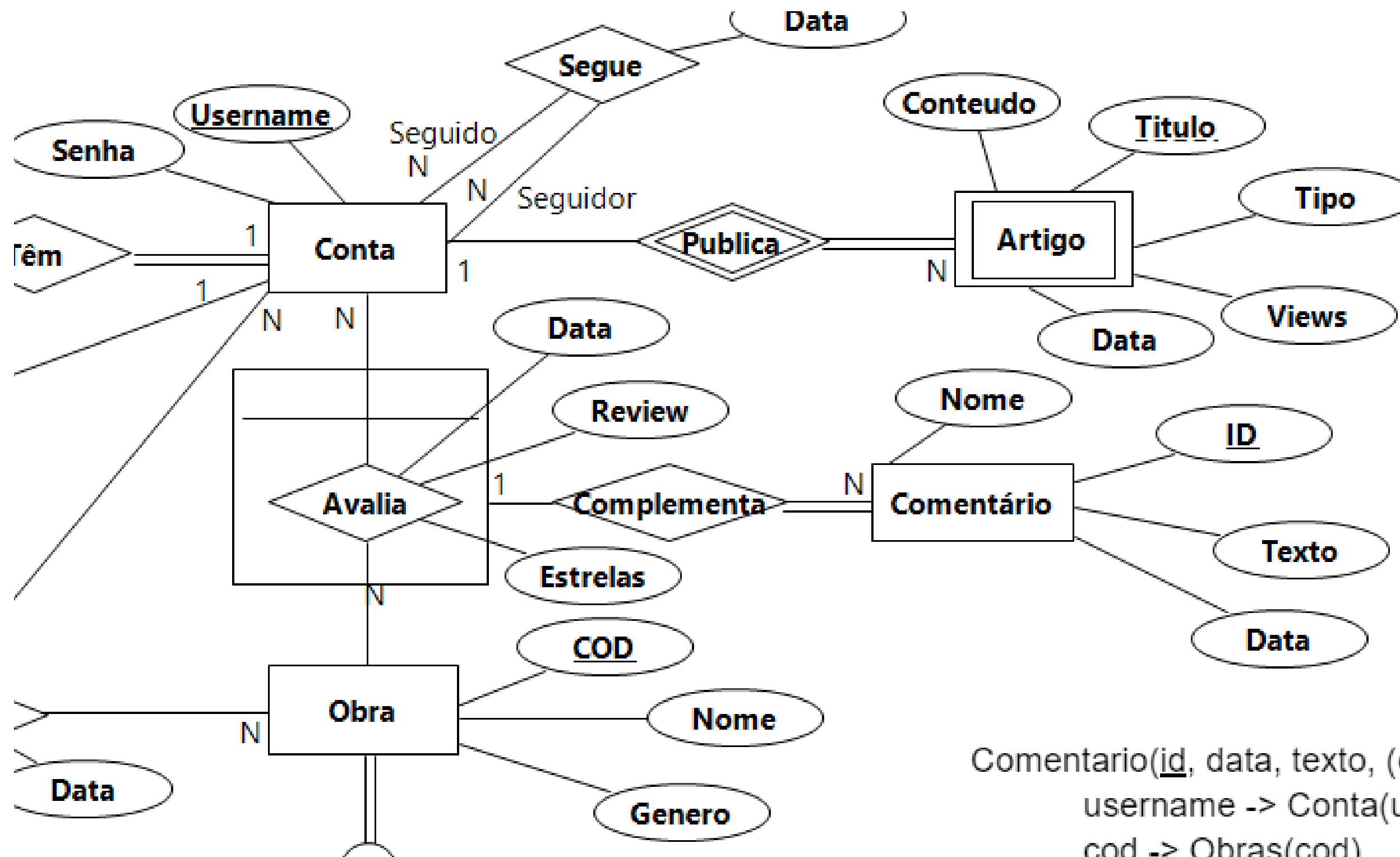
Produção(cod, produção)
cod -> Filmes(cod)

Livros(cod, paginas, escritor)
cod -> Obras(cod)

MODELO CONCEITUAL + LÓGICO



Playlist(id, nome, username!)
username -> Conta(username)



MODELO CONCEITUAL + LÓGICO

Comentario(id, data, texto, (cod, username)!, nome)
 username -> Conta(username)
 cod -> Obras(cod)

Avalia(cod, username, review, data, estrelas)
 username -> Conta(username)
 cod -> Obras(cod)

MODELO RELACIONAL

USUÁRIO(EMAIL, NOME, CPF)

CONTA(USERNAME, SENHA, [EMAIL]!)
EMAIL -> USUÁRIO(EMAIL)

SEGUIR(SEGUE, SEGUIDO, DATA)
SEGUE -> CONTA(USERNAME)
SEGUIDO -> CONTA(USERNAME)

ARTIGO(USERNAME, TITULO, DATA, TIPO, VIEWS, CONTEUDO)
USERNAME -> CONTA(USERNAME)

OBRAS(COD, NOME, GÊNERO)

FILMES(COD, ELEN_DIRETOR, DURACAO)
COD -> OBRAS(COD)
ATORES(COD, ATORES)
COD -> FILMES(COD)

PRODUÇÃO(COD, PRODUÇÃO)
COD -> FILMES(COD)

LIVROS(COD, PAGINAS, ESCRITOR)
COD -> OBRAS(COD)

PLAYLIST(ID, NOME, USERNAME!)
USERNAME -> CONTA(USERNAME)

AVALIA(COD, USERNAME, REVIEW, DATA, ESTRELAS)
USERNAME -> CONTA(USERNAME)
COD -> OBRAS(COD)

COMENTARIO(ID, DATA, TEXTO, (COD, USERNAME)!, NOME)
USERNAME -> CONTA(USERNAME)
COD -> OBRAS(COD)

ADICIONA(ID, COD, USERNAME, DATA)
ID -> PLAYLIST(ID)
COD -> OBRAS(COD)
USERNAME -> CONTA(USERNAME)

SQL

```

-- Tabela Usuário
CREATE TABLE Usuario (
    email VARCHAR(255) PRIMARY KEY,
    nome VARCHAR(255) NOT NULL,
    cpf VARCHAR(11) UNIQUE NOT NULL
);

-- Tabela Conta
CREATE TABLE Conta (
    username VARCHAR(255) PRIMARY KEY,
    senha VARCHAR(255) NOT NULL,
    email VARCHAR(255) UNIQUE NOT NULL,
    FOREIGN KEY (email) REFERENCES Usuario(email)
);

-- Tabela Seguir
CREATE TABLE Seguir (
    segue VARCHAR(255),
    seguido VARCHAR(255),
    data DATE NOT NULL,
    PRIMARY KEY (segue, seguido),
    FOREIGN KEY (segue) REFERENCES Conta(username),
    FOREIGN KEY (seguido) REFERENCES Conta(username)
);

```

```

-- Tabela Artigo
CREATE TABLE Artigo (
    username VARCHAR(255),
    data DATE NOT NULL,
    tipo VARCHAR(255) NOT NULL,
    titulo VARCHAR(255),
    views INT NOT NULL DEFAULT 0,
    conteudo TEXT NOT NULL,
    PRIMARY KEY (username, titulo),
    FOREIGN KEY (username) REFERENCES Conta(username)
);

-- Tabela Obras
CREATE TABLE Obras (
    cod INT PRIMARY KEY AUTO_INCREMENT,
    nome VARCHAR(255),
    genero VARCHAR(255)
);

-- Tabela Filmes
CREATE TABLE Filmes (
    cod INT PRIMARY KEY,
    diretor VARCHAR(255),
    duracao INT,
    FOREIGN KEY (cod) REFERENCES Obras(cod)
);

```

criação

criação

```
-- Tabela Atores
CREATE TABLE Atores (
    cod INT,
    ator VARCHAR(255),
    PRIMARY KEY (cod, ator),
    FOREIGN KEY (cod) REFERENCES Filmes(cod)
);

-- Tabela Producao
CREATE TABLE Producao (
    cod INT,
    producao VARCHAR(255),
    PRIMARY KEY (cod, producao),
    FOREIGN KEY (cod) REFERENCES Filmes(cod)
);

-- Tabela Livros
CREATE TABLE Livros (
    cod INT PRIMARY KEY,
    paginas INT,
    escritor VARCHAR(255),
    FOREIGN KEY (cod) REFERENCES Obras(cod)
);
```

```
-- Tabela Playlist
CREATE TABLE Playlist (
    id INT PRIMARY KEY AUTO_INCREMENT,
    username VARCHAR(255) NOT NULL,
    nome VARCHAR(255) NOT NULL,
    FOREIGN KEY (username) REFERENCES Conta(username)
);

-- Tabela Avalia
CREATE TABLE Avalia (
    cod INT,
    username VARCHAR(255),
    review TEXT,
    data DATE NOT NULL,
    estrelas DECIMAL(2,1) NOT NULL,
    PRIMARY KEY (cod, username),
    FOREIGN KEY (cod) REFERENCES Obras(cod) ON DELETE CASCADE,
    FOREIGN KEY (username) REFERENCES Conta(username) ON DELETE CASCADE
);
```


CRIAÇÃO

```
-- Tabela Comentario
CREATE TABLE Comentario (
    id INT AUTO_INCREMENT,
    data DATE NOT NULL,
    texto TEXT NOT NULL,
    nome VARCHAR(255) NOT NULL,
    cod INT NOT NULL,
    username VARCHAR(255) NOT NULL,
    PRIMARY KEY (id),
    FOREIGN KEY (cod) REFERENCES Obras(cod) ON DELETE CASCADE,
    FOREIGN KEY (username) REFERENCES Conta(username) ON DELETE CASCADE
);
```

```
-- Tabela Adiciona
CREATE TABLE Adiciona (
    ID INT,
    COD INT,
    USERNAME VARCHAR(255),
    DATA DATE NOT NULL,
    PRIMARY KEY (ID, COD, USERNAME),
    FOREIGN KEY (COD) REFERENCES Obras(cod),
    FOREIGN KEY (USERNAME) REFERENCES Conta(username),
    FOREIGN KEY (ID) REFERENCES Playlist(ID)
);
```

POVOAMENTO

```
INSERT ALL
INTO Usuario (email, nome, cpf) VALUES ('joao.s@example.com', 'João', '12345678900')
INTO Usuario (email, nome, cpf) VALUES ('maria.o@example.com', 'Maria', '98765432100')
INTO Usuario (email, nome, cpf) VALUES ('pedro.s@example.com', 'Pedro', '45678912300')
INTO Usuario (email, nome, cpf) VALUES ('ana.c@example.com', 'Ana', '32165498700')
INTO Usuario (email, nome, cpf) VALUES ('carlos.p@example.com', 'Carlos', '78912345600')
INTO Usuario (email, nome, cpf) VALUES ('juliana.f@example.com', 'Juliana', '65498732100')
INTO Usuario (email, nome, cpf) VALUES ('lucas.a@example.com', 'Lucas', '23456789000')
INTO Usuario (email, nome, cpf) VALUES ('luiza.s@example.com', 'Luiza', '87654321000')
INTO Usuario (email, nome, cpf) VALUES ('marcelo.l@example.com', 'Marcelo', '34567890100')
INTO Usuario (email, nome, cpf) VALUES ('camila.r@example.com', 'Camila', '67890123400')
SELECT * FROM dual;
```

```
INSERT ALL
INTO Conta (username, senha, email) VALUES ('joao.s', 'senha123', 'joao.s@example.com')
INTO Conta (username, senha, email) VALUES ('maria.o', 'senha456', 'maria.o@example.com')
INTO Conta (username, senha, email) VALUES ('pedro.s', 'senha789', 'pedro.s@example.com')
INTO Conta (username, senha, email) VALUES ('ana.c', 'senhaabc', 'ana.c@example.com')
INTO Conta (username, senha, email) VALUES ('carlos.p', 'senha1234', 'carlos.p@example.com')
INTO Conta (username, senha, email) VALUES ('juliana.f', 'senha7890', 'juliana.f@example.com')
INTO Conta (username, senha, email) VALUES ('lucas.a', 'senhaabc1', 'lucas.a@example.com')
INTO Conta (username, senha, email) VALUES ('luiza.s', 'senha12345', 'luiza.s@example.com')
INTO Conta (username, senha, email) VALUES ('marcelo.l', 'senha4567', 'marcelo.l@example.com')
INTO Conta (username, senha, email) VALUES ('camila.r', 'senha78901', 'camila.r@example.com')
SELECT * FROM dual;
```

POVOAMENTO

15

```
INSERT ALL
INTO Obras (cod, nome, genero) VALUES (1, 'Dom Quixote', 'Romance')
INTO Obras (cod, nome, genero) VALUES (2, '1984', 'Ficção Científica')
INTO Obras (cod, nome, genero) VALUES (3, 'Duna 2', 'Ficção Científica')
INTO Obras (cod, nome, genero) VALUES (4, 'Clube da Luta', 'Suspense')
INTO Obras (cod, nome, genero) VALUES (5, 'O Senhor dos Anéis', 'Fantasia')
INTO Obras (cod, nome, genero) VALUES (6, 'Interstellar', 'Ficção Científica')
INTO Obras (cod, nome, genero) VALUES (7, 'Harry Potter e a Pedra Filosofal', 'Fantasia')
INTO Obras (cod, nome, genero) VALUES (8, 'Psicopata Americano', 'Comédia')
INTO Obras (cod, nome, genero) VALUES (9, 'Amor e Outras Drogas', 'Comédia Romântica')
INTO Obras (cod, nome, genero) VALUES (10, 'A Metamorfose', 'Ficção Absurda')
INTO Obras (cod, nome, genero) VALUES (11, 'Batman O Cavaleiro Das Trevas', 'Ação')
SELECT * FROM dual;
```

```
INSERT ALL
INTO Avalia (cod, username, data, review, estrelas) VALUES (1, 'marcelo.l', TO_DATE('2023-12-08', 'YYYY-MM-DD'), 'AMEI ESTE LIVRO, ROMANCE DE PRIMEIRA CLASSE. DOM QUIXOTE É MUITO BRABO NA PORRADA.', 2.0)
INTO Avalia (cod, username, data, review, estrelas) VALUES (3, 'juliana.f', TO_DATE('2024-03-10', 'YYYY-MM-DD'), 'FILME DE PRIMEIRA CLASSE, COBRONA BEM DETALHADA E GRANDE.', 4.5)
INTO Avalia (cod, username, data, review, estrelas) VALUES (10, 'ana.c', TO_DATE('2024-01-23', 'YYYY-MM-DD'), 'QUE BARATONA SAPECA! LIVRO IMPECÁVEL E MUDOU MINHA PERCEPÇÃO DE MUNDO.', 5.0)
INTO Avalia (cod, username, data, review, estrelas) VALUES (6, 'maria.o', TO_DATE('2023-12-29', 'YYYY-MM-DD'), 'BURACO NEGROOOOOOOOOOOOOOOOOOOOOOOOOOOOO.', 5.0)
INTO Avalia (cod, username, data, review, estrelas) VALUES (7, 'joao.s', TO_DATE('2024-01-12', 'YYYY-MM-DD'), 'ARRASTA A TABACA NA VARA VAI SENTANDO NA VASSOURA EITA BRUXINHA RABUDA.', 4.0)
INTO Avalia (cod, username, data, review, estrelas) VALUES (1, 'joao.s', TO_DATE('2024-01-23', 'YYYY-MM-DD'), 'BOM DIA PRINCESAAA PUFavo SENTI NA GLOCK', 5.0);
SELECT * FROM dual;
```

POVOAMENTO

```
INSERT ALL
INTO Livros (cod, paginas, escritor) VALUES (1, 1033, 'Miguel de Cervantes')
INTO Livros (cod, paginas, escritor) VALUES (2, 416, 'George Orwell')
INTO Livros (cod, paginas, escritor) VALUES (5, 1568, 'J.R.R. Tolkien')
INTO Livros (cod, paginas, escritor) VALUES (7, 208, 'J. K. Rowling')
INTO Livros (cod, paginas, escritor) VALUES (10, 112, 'Franz Kafka')
SELECT * FROM dual;
```

```
INSERT ALL
INTO Filmes (cod, diretor, duracao) VALUES (3, 'Denis Villeneuve', 160)
INTO Filmes (cod, diretor, duracao) VALUES (4, 'David Fincher', 140)
INTO Filmes (cod, diretor, duracao) VALUES (6, 'Christopher Nolan', 160)
INTO Filmes (cod, diretor, duracao) VALUES (8, 'Mary Harron', 100)
INTO Filmes (cod, diretor, duracao) VALUES (9, 'Pericles Sardinha', 110)
INTO Filmes (cod, diretor, duracao) VALUES (11, 'Christopher Nolan', 150)
SELECT * FROM dual;
```

```
INSERT ALL
INTO atores (cod, ator) VALUES (3, 'Zendaya')
INTO atores (cod, ator) VALUES (4, 'Brad Pitt')
INTO atores (cod, ator) VALUES (6, 'Jessica Chastian')
INTO atores (cod, ator) VALUES (8, 'Jared Leto')
INTO atores (cod, ator) VALUES (9, 'Clara')
SELECT * FROM dual;
```

```
INSERT ALL
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('joao.s', 'O livro mais vendido de todos os tempos, depois da bíblia', 'velho porém muito atual em seus temas. Don quixote é um personagem muito interessante',
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('maria.o', 'Amor e outras drogas', 'FILME FOFÍSSIMO E EMOCIONANTE ANNE HATHAWAY E JAKE GYLLENHAALL MEUS FAVS &lt3', 'Resenha', 18, TO_DATE('2022-12-01', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('ana.c', 'Harry potter', 'dos meus livros do harry é o meu favorito!', 'Resenha', 0, TO_DATE('2016-07-29', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('ana.c', 'A metamorfose', 'kafkaesco', 'Resenha', 2, TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('carlos.p', 'A obra prima de kafka', 'o titulo já diz tudo', 'Resenha', 2893, TO_DATE('2021-03-21', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('marcelo.l', 'uma fantasia masculina sem pé nem cabeça', 'nossa q horrível esse filme', 'Artigo', 342, TO_DATE('2020-05-18', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('lucas.a', '5 estrelas', 'amei gente recomendo para todo mundo que tem pai', 'Artigo', 97, TO_DATE('2022-04-19', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('lucas.a', 'deplorável', 'livro chato gostei não', 'Artigo', 8, TO_DATE('2018-09-03', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('luiza.s', 'OSCAR-WORTHY', 'esse filme vai ganhar todas as suas indicações ao oscar viu se preparem pra lavada que vai ser', 'Artigo', 972, TO_DATE('2011-02-11', 'YYYY-MM-DD'))
INTO Artigo (username, titulo, conteudo, tipo, views, data) VALUES ('juliana.f', 'meh', ':/', 'Resenha', 0, TO_DATE('2019-10-23', 'YYYY-MM-DD'))
SELECT * FROM dual;
```

```
INSERT ALL
INTO comentario (id, cod, username, nome, texto, data) VALUES (1, 3, 'juliana.f', 'marcelo.l', 'concordo que a cobrona é grande mas a minha é maior.', TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO comentario (id, cod, username, nome, texto, data) VALUES (2, 3, 'juliana.f', 'lucas.a', 'IH ALA QUE PAPO É ESSE KSKSKSKSKSKSK', TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO comentario (id, cod, username, nome, texto, data) VALUES (3, 7, 'joao.s', 'maria.o', 'KKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKKK CHAPOU LEGAL', TO_DATE('2024-01-29', 'YYYY-MM-DD'))
INTO comentario (id, cod, username, nome, texto, data) VALUES (4, 7, 'joao.s', 'maria.o', 'ME CHAMA DE HARRY PORRAAAAAAAA!', TO_DATE('2024-01-12', 'YYYY-MM-DD'))
INTO comentario (id, cod, username, nome, texto, data) VALUES (5, 7, 'joao.s', 'carlos.p', 'SE ACHA O ENGRAÇADINHO NE OTARIO', TO_DATE('2024-01-13', 'YYYY-MM-DD'))
INTO comentario (id, cod, username, nome, texto, data) VALUES (6, 10, 'ana.c', 'juliana.f', 'AMO BARATAS TAMBEM', TO_DATE('2024-01-23', 'YYYY-MM-DD'))
SELECT * FROM dual;
```

```
INSERT ALL
  INTO Playlist (id, username, nome) VALUES (1, 'joao.s', 'Arafilmes')
  INTO Playlist (id, username, nome) VALUES (2, 'maria.o', 'Melhores filmes de terror de 1945')
  INTO Playlist (id, username, nome) VALUES (3, 'juliana.f', 'Filmes de Ficção')
  INTO Playlist (id, username, nome) VALUES (4, 'juliana.f', 'Playlist Lua de Mel')
  INTO Playlist (id, username, nome) VALUES (5, 'ana.c', 'Filmes com conteúdo +18')
SELECT * FROM dual;
```

POVOAMENTO

POVOAMENTO

```
INSERT ALL
INTO Adiciona (id, cod, username, data) VALUES (1, 11, 'joao.s', TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (1, 8, 'joao.s', TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (1, 6, 'joao.s', TO_DATE('2024-03-09', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (1, 1, 'joao.s', TO_DATE('2024-03-01', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (1, 5, 'joao.s', TO_DATE('2024-03-10', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (3, 3, 'maria.o', TO_DATE('2024-03-02', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (3, 3, 'juliana.f', TO_DATE('2024-02-29', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (3, 6, 'ana.c', TO_DATE('2024-03-01', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (4, 9, 'juliana.f', TO_DATE('2024-03-02', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (4, 2, 'lucas.a', TO_DATE('2024-03-02', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (4, 3, 'lucas.a', TO_DATE('2024-03-02', 'YYYY-MM-DD'))
INTO Adiciona (id, cod, username, data) VALUES (5, 7, 'ana.c', TO_DATE('2024-03-08', 'YYYY-MM-DD'))
SELECT * FROM dual;
```

```
INSERT ALL
INTO Seguir (segue, seguido, data) VALUES ('joao.s', 'maria.o', TO_DATE('2012-05-22', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('joao.s', 'juliana.f', TO_DATE('2012-05-22', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('joao.s', 'ana.c', TO_DATE('2012-05-27', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('lucas.a', 'maria.o', TO_DATE('2023-11-22', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('maria.o', 'juliana.f', TO_DATE('2022-09-03', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('maria.o', 'joao.s', TO_DATE('2012-05-22', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('maria.o', 'lucas.a', TO_DATE('2023-09-01', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('ana.c', 'maria.o', TO_DATE('2015-03-12', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('marcelo.l', 'juliana.f', TO_DATE('2017-12-09', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('juliana.f', 'marcelo.l', TO_DATE('2017-12-10', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('luiza.s', 'camila.r', TO_DATE('2021-04-13', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('luiza.s', 'maria.o', TO_DATE('2021-04-13', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('carlos.p', 'camila.r', TO_DATE('2017-03-21', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('camila.r', 'carlos.p', TO_DATE('2019-10-01', 'YYYY-MM-DD'))
INTO Seguir (segue, seguido, data) VALUES ('juliana.f', 'carlos.p', TO_DATE('2018-01-22', 'YYYY-MM-DD'))
SELECT * FROM dual;
```

Consultas

CONSULTAS

```
-- Username das contas que possuem mais de uma playlist criada (GROUP BY OU HAVING) --  
select username, count(*) AS qtd_playlists  
from playlist p  
group by username  
having count(*) > 1;
```

USERNAME	QTD_PLAYLISTS
juliana.f	2

CONSULTAS

```
-- Código da obra de uma avaliação que recebeu comentários na data 10/03/2024 (JUNÇÃO INTERNA) --  
select distinct a.cod  
from avalia a inner join comentario c on a.cod = c.cod and c.username = a.username  
where c.data = '2024/03/10';
```

COD

3

CONSULTAS

```
-- 0 username das contas que avaliaram obras --  
SELECT c.username  
from conta c  
where exists (select username  
              from avalia a  
              where c.username = a.username);
```

USERNAME
ana.c
joao.s
juliana.f
marcelo.l
maria.o

CONSULTAS

```
-- 0 nome das obras que não foram avaliadas --
```

```
SELECT o.nome
```

```
from obras o
```

```
where not exists (select a.cod
```

```
from avalia a
```

```
where o.cod = a.cod);
```

NOME

Batman O Cavaleiro Das Trevas

1984

Psicopata Americano

Clube da Luta

O Senhor dos Anéis

Amor e Outras Drogas

CONSULTAS

```
-- Quantidade de obras avaliadas por uma conta (SUBCONSULTA ESCALAR) --  
SELECT username, (select count(*) from avalia a where c.username = a.username)  
from conta c;
```

USERNAME	(SELECT COUNT(*) FROM AVALIA A WHERE C.USERNAME=A.USERNAME)
ana.c	1
camila.r	0
carlos.p	0
joao.s	2
juliana.f	1
lucas.a	0
luiza.s	0
marcelo.l	1
maria.o	1
pedro.s	0

CONSULTAS

```
-- Código da obra de uma avaliação que recebeu comentário de id = 6 (SUBCONSULTA TIPO LINHA) --  
select cod  
from avalia a  
where (username, cod) = (select username, cod  
                        from comentario  
                        where id = 6);
```

COD

10

CONSULTAS

```
-- 0 nome das obras que foram avaliadas (SUBCONSULTA TABELA)
SELECT o.nome
from obras o
where o.cod in (select a.cod
                from avalia a
                where o.cod = a.cod);
```

NOME
Dom Quixote
Duna 2
Interstellar
Harry Potter e a Pedra Filosofal
A Metamorfose

CONSULTAS

```
-- MOSTRAR O USERNAME DAS PESSOAS QUE POSSUEM E NÃO POSSUEM AVALIAÇÃO --  
--| E SUAS RESPECTIVAS AVALIAÇÕES (JUNÇÃO EXTERNA) --  
select c.username, a.review  
from conta c left outer join avalia a on c.username = a.username;
```

USERNAME	REVIEW
marcelo.l	AMEI ESTE LIVRO, ROMANCE DE PRIMEIRA CLASSE. DOM QUIXOTE É MUITO BRABO NA PORRADA.
juliana.f	FILME DE PRIMEIRA CLASSE, COBRONA BEM DETALHADA E GRANDE.
ana.c	QUE BARATONA SAPECA! LIVRO IMPECÁVEL E MUDOU MINHA PERCEPÇÃO DE MUNDO.
maria.o	BURACO NEGRO000000000000000000000000000000.
joao.s	BOM DIA PRINCESAAA PUFAVO SENTI NA GLOCK
carlos.p	-
lucas.a	-
camila.r	-
camila.r	-
luiza.s	-
pedro.s	-

CONSULTAS

```
-- Mostrar todos os artigos das contas de username ana.c e lucas.a --  
--(OPERADOR DE CONJUNTO UNION (UNIÃO)) --  
(select * from artigo a where a.username = 'ana.c')  
UNION  
(select * from artigo a1 where a1.username = 'lucas.a');
```

OBRIGADO!

LINK DO PROJETO:

https://github.com/RodrigoBarbosaa/Projeto_BD