**Banco de dados - Prof. Dr.-Ing. Leonardo Andrade Ribeiro**

**Estudantes:** Lisandra Cristina de Moura Menezes, Marcos Vinicios Pereira de Morais e Pedro Henrique Souza Martins Souto.

**Introdução**

Com base em nossas observações e estudos, nosso projeto teve como fonte de inspiração a Associação Pastoral da Terra. A menção deste fato se faz necessária para corrigir uma informação apresentada em nossa descrição do mini mundo, que indicava que a Associação não possuía um sistema de digitalização de seus documentos.

Na verdade, a Associação Pastoral da Terra tem, sim, um processo de digitalização e disponibiliza o acesso a esses documentos digitais a pesquisadores que possam justificar a relevância e a intenção de uso desses dados para seus estudos. Contudo, identificamos uma limitação crucial nesse sistema: a falta de uma organização sistemática e intuitiva dos documentos digitais. Com isso, a localização de informações específicas dentro das pastas de digitalização torna-se uma tarefa árdua e praticamente inviável.

Esperamos que essa clarificação transmita de forma mais precisa a situação atual da gestão de documentos da Associação Pastoral da Terra e o contexto que nos motivou a desenvolver nosso mini mundo.

Não obtivemos a devida autorização para utilizar os dados, visto que a sua utilização demandaria a supervisão de um orientador e a declaração explícita de uma intenção de pesquisa. Entretanto, tivemos a oportunidade de compreender a organização dos dados no Google Drive.

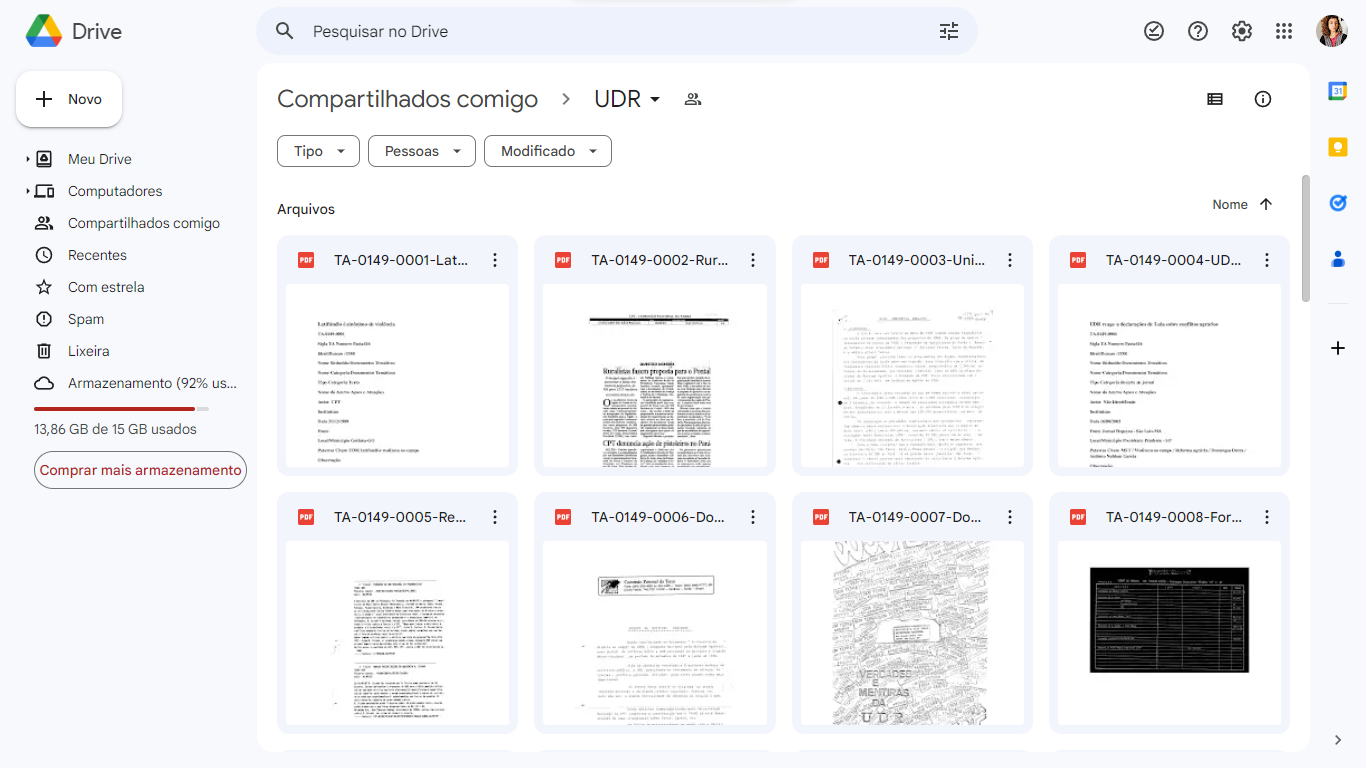


Fig. 1 - Print da pasta UDR da Associação Pastoral da Terra

Observa-se que existe uma evidente tentativa de estruturação dos dados em questão, na qual o título do documento aparenta representar algum tipo de classificação bibliográfica. No entanto, nossa análise sugere que a plataforma Google Drive, apesar de suas diversas funcionalidades, pode não ser o meio mais apropriado para o armazenamento e gerenciamento desses dados.

Imagine que você é um historiador e deseja fazer uma pesquisa historiográfica sobre a criação de Goiânia. Indo a um órgão como a Pastoral da Terra, você pode seguir por duas formas possíveis: ter acesso ao drive ou ter acesso ao acervo físico. Nas duas opções você vai levar dias para descobrir os arquivos que você precisa para sua pesquisa e tem a chance de não conseguir encontrar nada. A solução é a criação de um banco de dados relacional, na qual será possível encontrar arquivos a partir de palavras chaves, nome do artigo, data de publicação entre outros.

1. **Descrição do Mini mundo**

Na associação Pastoral da Terra, são armazenados de forma física documentos oficiais, recortes de jornais, artigos e fotografias de momentos históricos que geralmente envolvem os temas relacionados ao MST, conflitos de terra, terra indígena e etc. Decidiu-se armazenar tal acervo em um banco de dados que traga consigo uma interface gráfica para que os usuários possam interagir com o acervo de forma prática e estruturada. Criaremos também, ambiente para criação de fórum de discussão sobre artigos que relacionam temas pesquisados no acervo, para promover a discussão e produção de novos conteúdos e fomentar uma rede de pesquisadores interessados no tema para que possam organicamente se ajudarem.

Para começar a organizar esse banco de dados, notou-se que precisamos de dois tipos de usuários: um que tenha a possibilidade exclusiva de operar o acervo, excluindo, atualizando ou editando, e o outro usuário pesquisador que tem a exclusiva possibilidade de compartilhar nos fóruns seu artigo.

Sabe-se que para operar um acervo tão valioso é necessário restrições de segurança, por isso faz-se necessário algum tipo de monitoramento que armazena informações básicas sobre o usuário e suas ações no acervo. Aplicamos esse mecanismo de segurança para todos os usuários, para fins de monitoramento dos funcionários que podem acessar e gerenciar os dados históricos do acervo, mas também para os usuários pesquisadores, para que as informações retidas sirvam futuramente para sistemas de recomendação.

Ao iniciar a sessão dos usuários, damos a possibilidade de acessarem tanto o acervo como o fórum de discussão. Dentro do acervo o usuário pode consultar documentos oficiais ou fotos ou recortes de jornais. E no fórum, os usuários acessam artigos ou comentários. Todo usuário pode criar novos fóruns e fazer comentários, assim como todos os usuários visualizam o acervo.

Ao publicar um artigo é necessário a definição de estritamente três palavras chaves, essa palavras serão uma alternativa mais fácil para pesquisar por temas específicos dentro do fórum. Todos os itens no acervo também possuem palavras chaves.

1. **Modelagem Conceitual**

No usuário destacamos atributos como CPF, nome, data de nascimento e senha. Caso o CPF do usuário esteja cadastrado como funcionário, o usuário pertence a especialização de USER-funcionário com o ID-func. Isso implica que, o USER-funcionário poderá editar, atualizar e excluir elementos do acervo. Caso o CPF não esteja cadastrado como funcionário, logo o usuário será necessariamente USER-pesquisador com seu ID-pesq. Logo, o USER-pesquisador poderá exclusivamente publicar artigos no fórum de discussão.

Por sua vez os artigos possuem nome do artigo, título, palavra-chave, autor, orientador, instituição, PDF e uma chave primária ID-art como atributos. É comum a todos os usuários criar quantos fóruns quiser e fazer quantos comentários quiser, sendo que para cada fórum ou comentário é associado a apenas 1 usuário. O fórum não pode ser operado pelo usuário funcionário, e nos fóruns podemos ter tanto artigos como comentários, seus atributos são ID-forum, nomeForum, NomeUser, Datapublicação. E por fim, nos comentários temos dois atributos ID\_com e o texto publicado.

No acervo, temos 3 categorias, as fotos, os documentos oficiais com e os recortes de jornais. E todos herdam os atributos ID-acervo, palavra-chave, título, data de publicação, autor, ID-Atividade e data-Acesso.

Os usuários iniciam sessões todas as vezes que entram no sistema e geram data de acesso e ID da sessão como chave primária, além disso as sessões são monitoradas por uma CE fraca Log-Acesso que armazena informações sobre as sessões.

Usamos para a modelagem conceitual a ferramenta Lucid, por ser uma ferramenta acessível e que ocorre o compartilhamento instantâneo de telas e alterações. Com o Lucid conseguimos fazer reuniões online para construir o MER (Estendido).

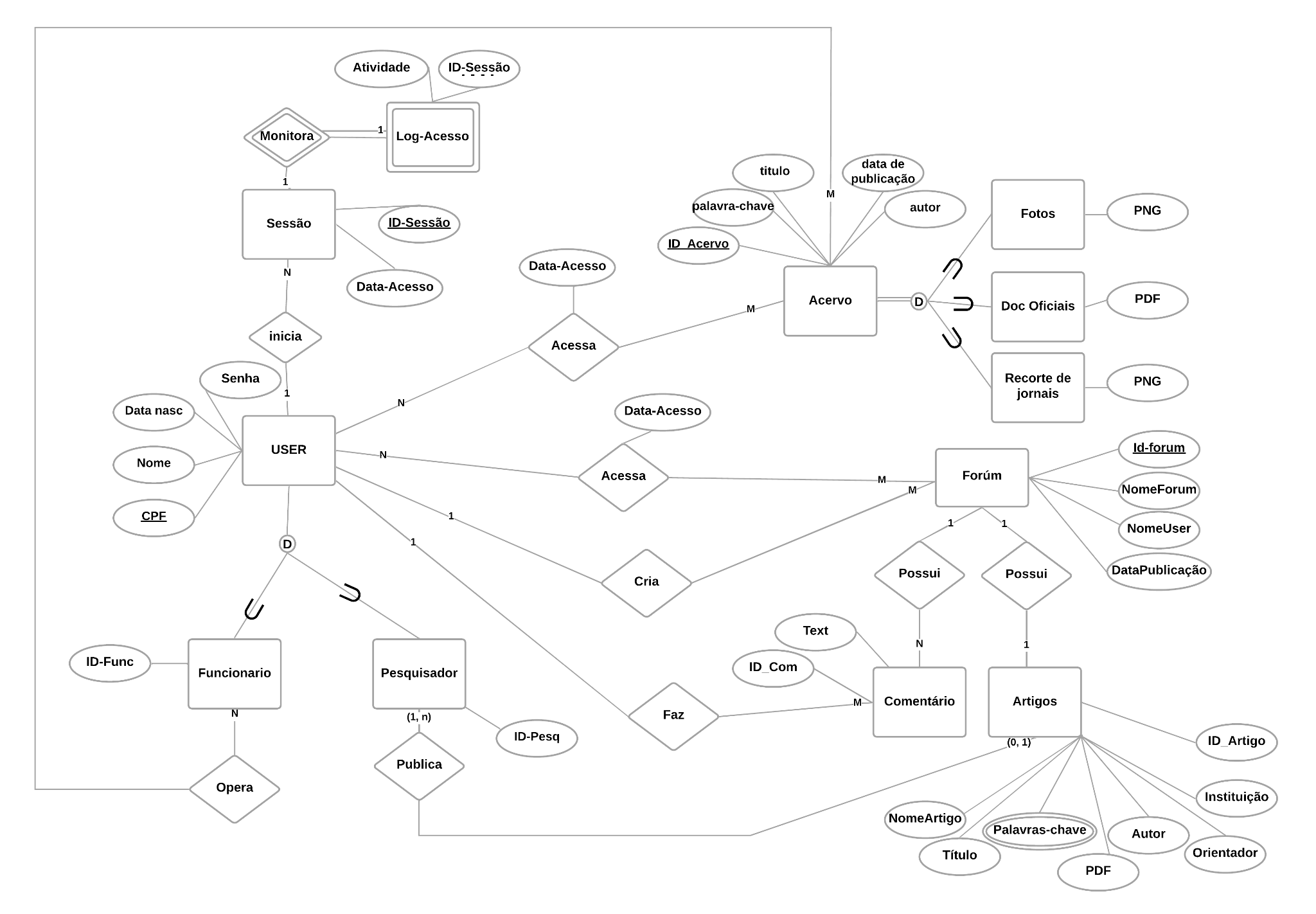


Fig. 2 - Modelo de Entidade-Relacionamento

Note na imagem acima que temos uma entidade USER que se especializa em dois tipos de usuário, cada um deles faz modificações distintas no nosso banco de dados. Ressaltamos novamente o artifício para monitorar a ação dos usuários que servirá para documentação das alterações feitas e sistemas de recomendação de conteúdo.

1. **Modelagem Lógica**

O modelo relacional é um tipo de estrutura de banco de dados baseada na lógica de conjuntos e na teoria das relações, que é intuitiva e eficaz para organizar dados. No nosso modelo, usamos exclusivamente chaves estrangeiras por não ter nenhuma cardinalidade que nos impossibilita de usar essa estratégia.

As CE’s User e acervo possuem especialização, portanto, suas chaves primárias são também a chave estrangeira nas suas respectivas especializações. Assim, evitamos usar chaves primárias excessivamente, facilitando a normalização que vem a seguir.

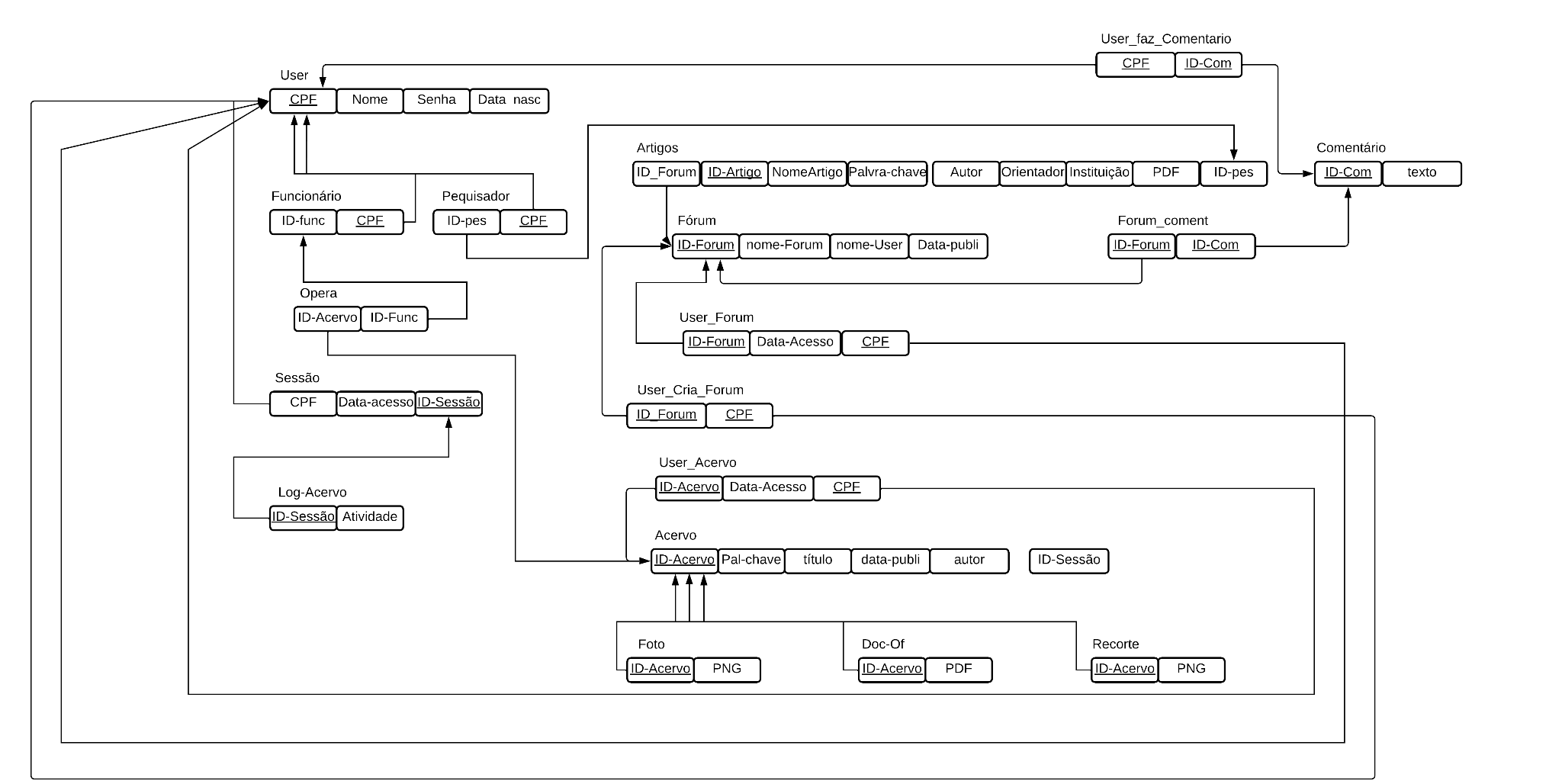
****

Fig. 3 - Mapeamento ER-Relacional

1. **Normalização**

Após análise das relações e de suas dependências funcionais, chegou-se à conclusão de que seria necessário fazer algumas alterações, em específico, nas relações Artigos e Acervo. Observa-se, pela imagem abaixo, que a relação Artigos não está normalizada, visto que o atributo Autor depende funcionalmente de um atributo não chave, ID-pes, o que resultaria em uma transitividade e, por consequência, na violação da Terceira Forma Normal (3FN).

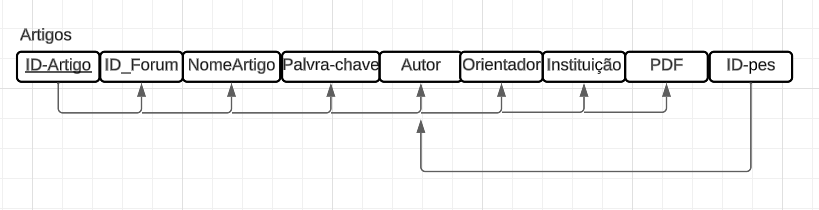


Fig. 4 - Dependências Funcionais da relação Artigos.

Portanto, foi optado pela decomposição dessa relação em duas outras: a tabela Artigos com os atributos ID-Artigo(PK), ID-Forum, NomeArtigo, Palavra-Chave, Orientador, Instituição e PDF; e a tabela autores, com atributos ID-Artigo(PK), ID-pes(PK) e Autor. Note que agora as duas relações se encontram devidamente de acordo com as regras de normalização da 3FN.



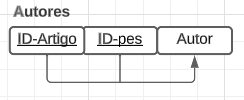


Fig. 5 e 6 - Normalização da antiga relação “Artigos”.

Além disso, note que a relação Acervo perdeu um atributo, ID-Sessão. Isso ocorreu pois foi entendido que tal atributo não exercia papel algum na relação, portanto, não faria sentido mantê-lo.

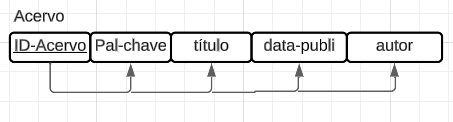


Fig. 6 - Normalização da relação Acervo.

Para os demais casos, não foram encontradas violações das regras de normalização, segue abaixo as dependências funcionais de todas as relações:

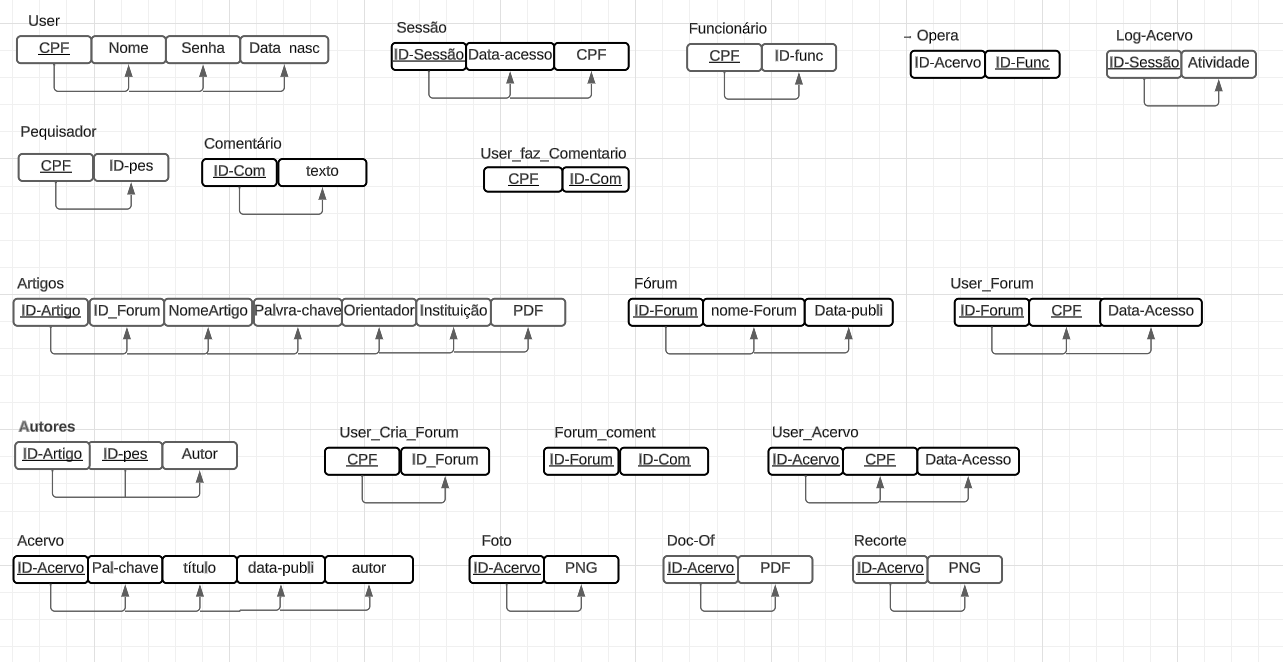


Fig. 7 - Dependências Funcionais das relações.

Portanto, após a normalização, o esquema do banco de dados ficaria da seguinte forma:

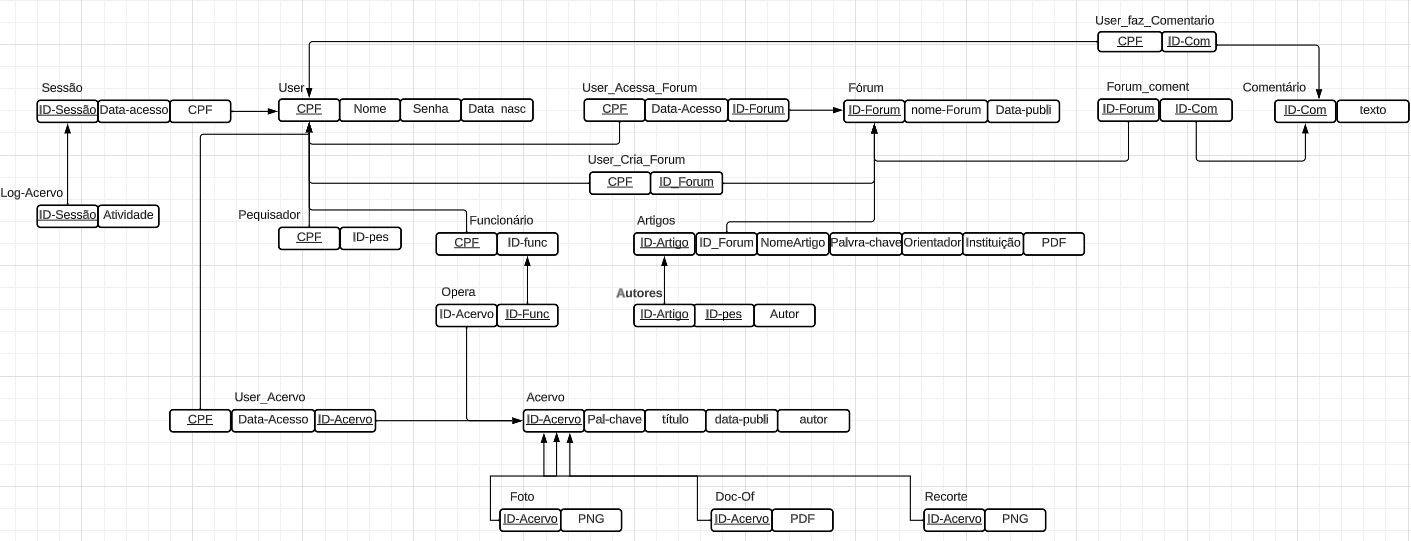


Fig. 8 - Esquema final do banco de dados.

1. **Construção e População do Banco de Dados**

Utilizamos os princípios fundamentais de SQL, adquiridos durante a disciplina de Banco de Dados, para a elaboração do banco de dados proposto neste documento. Por meio do link a seguir, é possível acessar e baixar todos os arquivos SQL necessários para a implementação deste banco de dados em seu ambiente. [[Clique aqui](https://drive.google.com/drive/folders/1hqEXFpjra9D8FOwkCWaTqtXq_rxUxw-j?usp=sharing)].

As tabelas em SQL são a espinha dorsal de qualquer banco de dados relacional. Elas são utilizadas para armazenar dados que são inter-relacionados de maneira estruturada. Para essa etapa de construção e população fizemos 19 tabelas, respectivas ao mapeamento da entidade relacional após a normalização.

Na população do banco de dados, chegamos a considerar opções com o Faker, uma biblioteca python que gera dados sintéticos a partir de um script simples. Porém, devido a interconectividade das tabelas preferimos usar o Chat Gpt para gerar os dados. Usamos o seguinte prompt:

Criei essas tabelas no workbench usando SQL, contendo restrições, relações entre tabelas e suas chaves primárias e estrangeiras, segue abaixo:  
“INSIRA A TABELA”

Agora quero fazer a população do meu banco de dados. Gere dados fictícios, preservando a consistência das chaves. A inserção deve ser feita em uma ordem que satisfaça as dependências entre as tabelas. [[1]](#footnote-0)

No link anteriormente disponibilizado, localizará o código de inserção referente a resposta deste prompt.

1. **Especificação de Operações sobre o Banco de Dados**

A Linguagem de Consulta Estruturada, ou SQL, é uma linguagem padrão para armazenar, manipular e recuperar dados em bancos de dados. Ela é utilizada em sistemas de gerenciamento de banco de dados relacional e é essencial para diversas soluções.

Uma das principais funções do SQL é realizar consultas aos bancos de dados. As consultas permitem que os usuários busquem um subconjunto específico de dados de uma ou mais tabelas. O comando mais básico para isso é a SELECT, que pode ser combinada com uma variedade de cláusulas (como WHERE, GROUP BY, ORDER BY, entre outras) para refinar e ordenar os resultados da consulta.

A eficácia das consultas SQL depende não apenas da correta formatação e sintaxe, mas também da estrutura e design do banco de dados subjacente. Por exemplo, um banco de dados bem normalizado permitirá consultas mais eficientes e resultados mais precisos.

É importante destacar que as operações executadas nesse banco de dados foram norteadas pela orientação do projeto de banco de dados. Todo o material está disponibilizado no drive, basta [[Clicar aqui](https://drive.google.com/drive/folders/1hqEXFpjra9D8FOwkCWaTqtXq_rxUxw-j?usp=sharing)], e uma amostra das consultas se encontra no apêndice deste trabalho.

Foram mais de 25 consultas feitas sobre esse conjunto de dados e para não deixar esse relatório demasiadamente extenso, disponibilizamos uma pasta no Drive, no qual pode-se fazer o download das consultas em SQL, existem também comentários que discriminam cada consulta e seus propósitos.

1. **Metodologia de Trabalho**

Através de uma abordagem diversificada, utilizamos uma combinação de ferramentas, incluindo vídeos, leituras, reuniões e discussões, para aprofundar a compreensão do tópico e incentivar a troca de ideias.

Com acesso a vídeos no youtube conseguimos esclarecer algumas dúvidas pontuais sobre o conteúdo do trabalho, e buscamos encontrar artigos que poderiam nos ajudar em nosso desafio, pois, enfrentamos dificuldade em entender nosso mini-mundo.

Além disso, as reuniões regulares foram estabelecidas como um meio para a comunicação constante e colaboração entre os membros da equipe. Estas sessões permitiram o esclarecimento de dúvidas, a revisão dos materiais estudados e a coordenação de esforços para o avanço contínuo do projeto.

Em suma, nossa metodologia foi baseada na ideia de que a aprendizagem eficaz ocorre quando há uma combinação de diversos métodos de ensino. Acreditamos que a combinação de vídeos, leituras, reuniões e discussões proporcionou uma experiência de ensino/aprendizagem melhor para os membros.

1. **Referências**

CPT NACIONAL. Conflitos no campo Brasil. Disponível em: <https://www.cptnacional.org.br/index.php/publicacoes-2/conflitos-no-campo-brasil>. Acesso em: 22/08/2023.

FERREIRA, Mário Roberto. Projeto biblios. Disponível em: [http://www.ufla.br/título-do-artigo.pdf](http://www.ufla.br/t%C3%ADtulo-do-artigo.pdf). Acesso em: 22/08/2023.

HASHTAG PROGRAMAÇÃO. 7 Dicas para Criar Bancos de Dados. [ano da postagem]. Disponível em: <https://www.youtube.com/watch?v=v4ycK11pt_M>. Acesso em: 22/08/2023.

OPENAI. ChatGPT, versão 3.5. Disponível em: <https://chat.openai.com>. Acesso em: 22/08/2023.

1. **Apêndice**

Disponibilizamos um drive com todas as figuras e código que utilizamos para a confecção desse trabalho a partir de um link para facilitar o acesso. [[Clicar aqui](https://drive.google.com/drive/folders/1hqEXFpjra9D8FOwkCWaTqtXq_rxUxw-j?usp=sharing)].

Nessa seção, como o recomendado, destacamos alguns códigos de criação, inserção e consultas em SQL.

CREATE TABLE User (

CPF VARCHAR(11) PRIMARY KEY,

Nome VARCHAR(255),

Senha VARCHAR(255),

data\_nasc DATE

);

CREATE TABLE Sessao (

ID\_sessao INT PRIMARY KEY,

data\_Acesso DATE,

CPF VARCHAR(11),

FOREIGN KEY (CPF) REFERENCES User(CPF)

);

CREATE TABLE Acervo (

ID\_acervo INT PRIMARY KEY,

pal\_chave VARCHAR(255),

titulo VARCHAR(255),

data\_publi DATE,

autor VARCHAR(255)

);

Fig. 9 - Criação das tabelas em SQL.

Neste primeiro exemplo, estamos criando duas tabelas primárias para o nosso banco de dados, i.e, acervo, user e sessão. Perceba que estamos criando as tabelas já com suas relações e devidas referências para chaves estrangeiras.

select u.CPF, u.Nome

from user u

where not exists(

select 1

from user\_acervo ua

where ua.CPF = u.CPF);

Fig. 10 - Consulta em SQL.

Na consulta que vimos acima, usando a cláusula “*where not exists*” queremos destacar entidades que não pertencem a um determinado grupo. No exemplo, estamos destacando os usuários que não acessaram o acervo.

select a.ID\_acervo, a.titulo,

(select count(\*)

from user\_acervo ua

where ua.ID\_Acervo = a.ID\_acervo) as acessos\_individuais,

(select avg(acessos)

from (

select count(\*) as acessos

from user\_acervo

group by ID\_acervo) sub) as media\_acessos

from acervo a;

Fig. 11 - Consulta em SQL.

Nesse exemplo, queremos saber o número de vezes que um acervo foi acessado em comparação com a média geral de acessos.

select f.nome\_forum, f.data\_publi, u.Nome as criador\_nome

from forum f

join user\_cria\_forum ucf

on f.ID\_forum = ucf.ID\_forum

join user u

on ucf.CPF = u.CPF;

Fig. 12 - Consulta em SQL.

Nesse simples exemplo, descobrimos quais usuários criaram os fóruns existentes e encerramos por aqui nossos exemplos. Vale ressaltar novamente que os códigos completos podem ser encontrados no drive ou no arquivo .zip disponibilizado anteriormente.

1. Prompt usado no Chat Gpt 3.5. [↑](#footnote-ref-0)