



Projeto - GDI

Arthur Thierre dos Santos Silva

Luan Gabriel Araujo Jardim

Lucas Inojosa

Luís Felipe Rodrigues de Oliveira

Marcela Arruda Asfora

Thiago José Alves de Souza

Minimundo

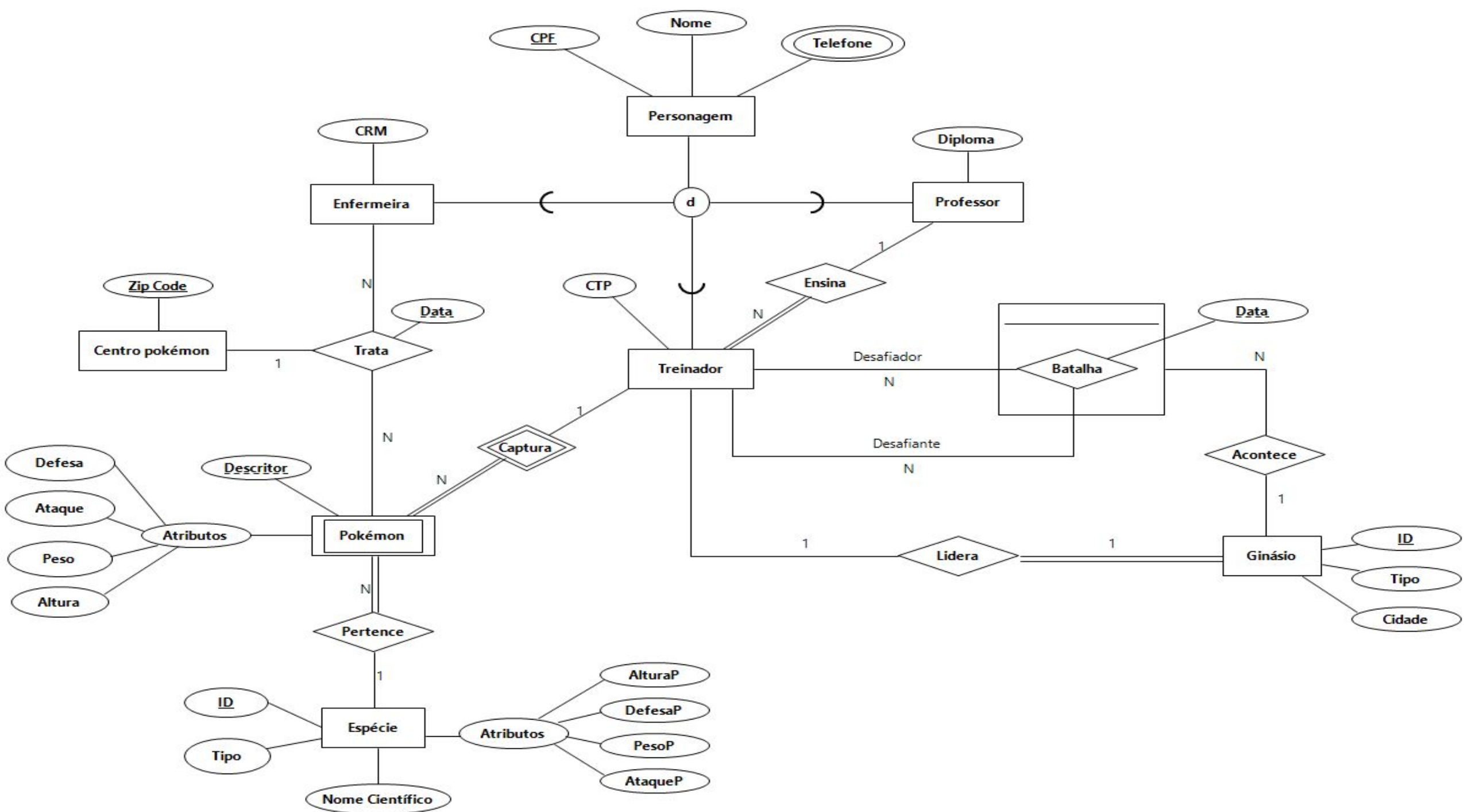
Este é um universo habitado por criaturas misteriosas conhecidas como Pokémon. Neste universo há humanos que se dedicam ao estudo de pokémon, pessoas que se dedicam ao cuidado da saúde desses seres e pessoas que capturam e treinam pokémon para batalhas em ginásios.



Modelo Conceitual

Construído utilizando a ferramenta EER Case





Modelo Lógico



Personagem(CPF, Nome)

Telefone(CPF, numero)

CPF -> Personagem(CPF)

Enfermeira(CPF, [CRM]!)

CPF -> Personagem (CPF)

Professor(CPF, Diploma!)

CPF -> Personagem(CPF)

Treinador(CPF, [CTP]!, CPF-prof!)

CPF -> Personagem(CPF)

CPF_prof -> Personagem(CPF)

Pokemon(CPF, Descritor, ID!, Ataque, Defesa, Altura, Peso)

CPF -> Treinador(CPF)

ID -> Espécie(ID)

Centro Pokémon(Zip Code)

Espécie(ID, Nome científico, Tipo, AtaqueP, DefesaP, AlturaP, PesoP)

Ginásio(ID, Cidade, Tipo, [CPF-líder]!)

CPF-líder -> Treinador(CPF)

Trata(CPF-Enfermeira, CPF-Pokemon, Descritor, Data, Zip Code!)

CPF-Enfermeira -> Enfermeira (CPF)

CPF-Pokemon, Descritor -> Pokemon (CPF-Pokemon, Descritor)

Batalha(CPF-Desafiador, CPF-Desafiante, Data, ID)

CPF-Desafiador -> Treinador(CPF)

CPF-Desafiante -> Treinador(CPF)

ID -> Ginásio(ID)



Consultas



```
1  -- Subconsulta do tipo tabela -----
2  -- RETORNE OS DE POKEMONS QUE POSSUEM ATAQUE IGUAL A MÉDIA DOS POKEMONS DE CADA TREINADOR
3  v SELECT A.CPF, A.ID, A.DESCRITOR
4  FROM POKEMON A
5  WHERE (CPF, ATK) IN (
6      SELECT CPF, AVG(ATK)
7      FROM POKEMON
8      GROUP BY CPF
9  );
```




```
1  -- Subconsulta do tipo escalar -----  
2  -- informa a quantidade máxima de pokemons capturados por um único treinador  
3  
4  SELECT MAX(QTD)  
5  FROM (  
6      SELECT COUNT(*) AS QTD  
7      FROM POKEMON P  
8      GROUP BY P.CPF);
```



```
1  -- Subconsulta do tipo linha -----  
2  -- Selecionando o pokemon com maior defesa e ataque  
3  
4  SELECT P.ID  
5  FROM POKEMON P  
6  WHERE (P.ATK, P.DEF) = (SELECT MAX(P2.ATK), MAX(P2.DEF)  
7                          FROM POKEMON P2);  
8
```



```
1  -- Junção interna -----  
2  -- Mostra o nome e cpf das enfermeiras que atenderam pokemons a partir do dia 09/apr/23  
3  
4_v SELECT DISTINCT P.CPF, P.NOME  
5  FROM TRATA T INNER JOIN PERSONAGEM P ON (T.CPF_ENFERMEIRA = P.CPF)  
6  WHERE T.DATA >= '09-APR-2023';
```



```
1  -- Junção externa -----  
2  -- coloque o CPF e o  NOME dos personagens que não possuem telefone  
3  
4  SELECT P.CPF, P.NOME  
5  FROM PERSONAGEM P LEFT JOIN telefone T ON P.CPF = T.CPF  
6  WHERE NUMERO IS NULL;
```



```
1  -- Semi junção -----
2  -- O cpf das enfermeiras que trataram mais de três vezes pokémons de um treinador de ginásio
3
4  select distinct tr.cpf_enfermeira
5  from trata tr
6  where exists (
7      select *
8      from treinador t
9      where t.cpf = tr.cpf_pokemon and exists (
10         select *
11         from ginasio g
12         where t.cpf = g.cpf_lider)
13 );
```




```
1  -- Anti-junção -----  
2  -- Seleciona o cpf dos professores sem alunos  
3  ✓ SELECT P.CPF  
4  FROM PROFESSOR P  
5  WHERE CPF NOT IN (  
6      SELECT T.CPF_PROF  
7      FROM TREINADOR T  
8  );
```



```
1  -- Group by/Having -----
2  -- QUANTIDADE DE POKEMONS QUE CADA TREINADOR TEM EM ORDEM DECRESCENTE
3  v SELECT A.CPF,A.NOME, COUNT(PO.DESCRITOR) AS QTD_DE_POKEMONS
4  FROM (
5      SELECT *
6      FROM PERSONAGEM P
7      WHERE P.CPF = ANY(
8          SELECT T.CPF
9          FROM TREINADOR T
10     )
11 ) A LEFT JOIN POKEMON PO
12 ON PO.CPF = A.CPF
13 GROUP BY (A.CPF,A.NOME)
14 ORDER BY QTD_DE_POKEMONS DESC;
```



```
1  -- Operação de conjunto -----
2  -- Seleciona treinador que tem pokemon do tipo fogo, mas não batalhou no ginasio de grama
3
4  v SELECT T.CPF, P.NOME
5  FROM TREINADOR T INNER JOIN PERSONAGEM P ON T.CPF = P.CPF
6  WHERE EXISTS (SELECT P.CPF
7                 FROM POKEMON P
8                 WHERE P.ID IN (SELECT E.ID
9                                FROM ESPECIE E
10                               WHERE E.TIPO = 'FOGO') AND T.CPF = P.CPF);
11
12  v MINUS
13
14  SELECT T.CPF, P.NOME
15  FROM TREINADOR T INNER JOIN PERSONAGEM P ON T.CPF = P.CPF
16  WHERE EXISTS (SELECT B.DESAFIANTE
17                 FROM BATALHA B
18                 WHERE T.CPF = B.DESAFIANTE AND B.ID =(SELECT G.ID
19                                                         FROM GINASIO G
20                                                         WHERE G.TIPO = 'GRAMA'));
```

