



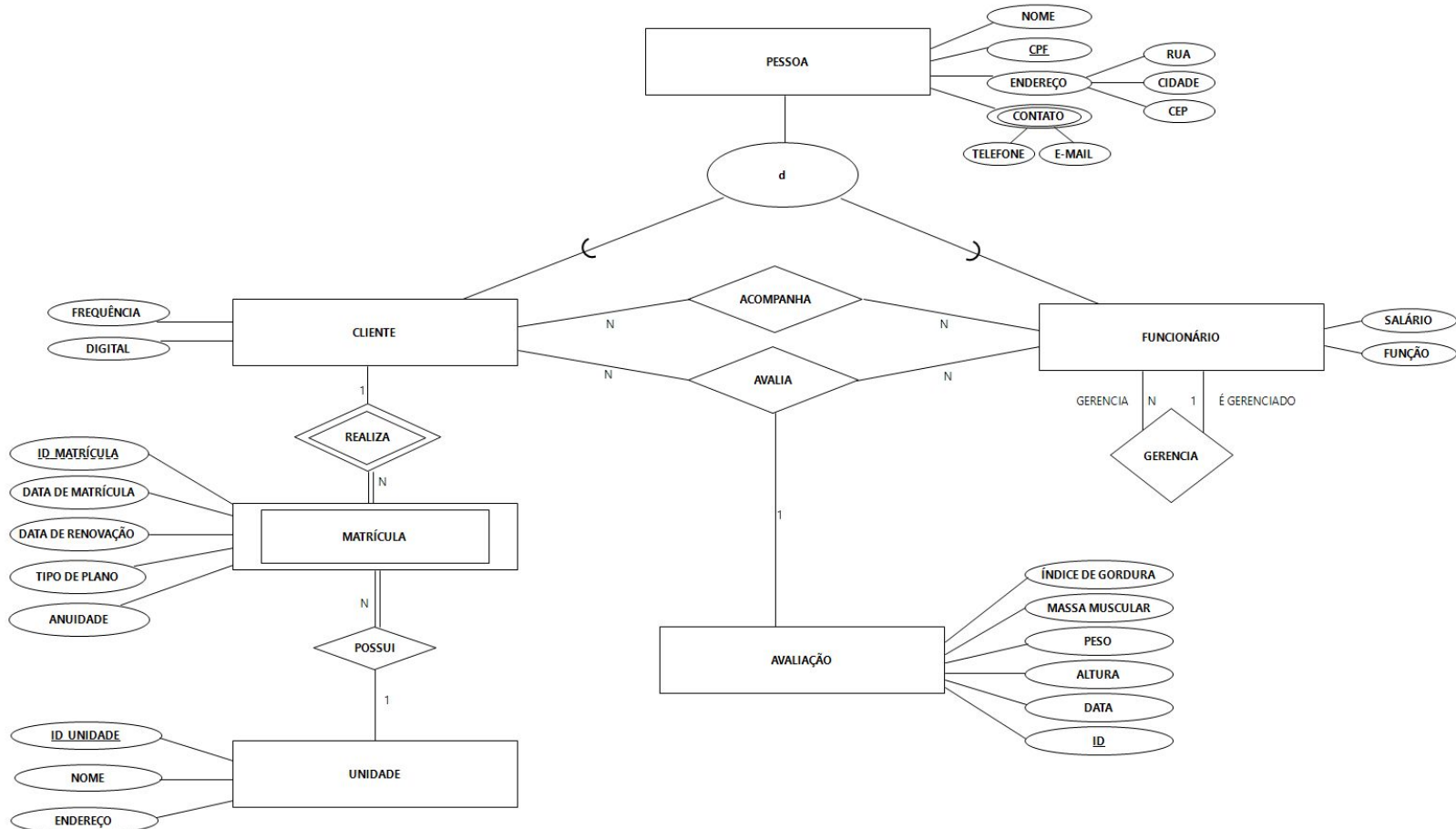
Fit Cln

Dara Caroline de Souza Vasconcelos - dcsv
Edinaldo Barbosa da Costa Filho - ebcf2
João Pedro Souza Pereira de Moura - jpspm
Maria Eduarda Souza Oliveira de Senna -
mesos
Rafael Jesus de Araujo Vasconcelos - rjav

Mini Mundo

Um sistema de uma rede de academias, "Fit CIn", há pessoas que são registradas contendo : CPF (Identificador), Nome, Endereço (Rua, Cidade, CEP) e podem ter contato (multivalorado) contendo email e/ou telefone; elas se dividem em 2 categorias : clientes e funcionários. Os atributos dos clientes são: Frequência, Digital. Os atributos dos funcionários são: Salário e Função na academia. Para cada cliente tem unicamente uma matrícula relacionada, constando : Data de Renovação, Tipo de Plano, Anuidade, Data de Matrícula. Funcionários acompanham clientes durante o treino, mas podem gerenciar outros funcionários. Um único funcionário pode acompanhar diversos clientes e um cliente pode ser acompanhado por diversos funcionários. Um cliente pode ser avaliado por um ou mais funcionários, e um funcionário pode avaliar um ou mais clientes, gerando avaliações contendo índice de gordura, massa muscular, peso, altura e data como atributos.

Projeto Conceitual



Projeto Lógico

Pessoa (CPF, nome, end_rua, end_cidade, end_CEP);

Contato(CPF, telefone, email)

CPF → Pessoa(CPF)

Avaliacao(ID, data, altura, peso, massa_muscular, indice_de_gordura)

Unidade(ID_unidade, nome, endereco)

Funcionario(CPF, funcao, salario)

CPF → Pessoa(CPF)

Gerencia(gerente!, gerenciado)

gerente → Funcionario(CPF)

gerenciado → Funcionario(CPF)

Cliente(CPF, frequencia, digital)

CPF → Pessoa(CPF)

Matricula(CPF, ID_matricula, data_de_matricula, tipo_de_plano, anuidade, data_de_renovacao, ID_unidade!)

CPF → Cliente(CPF)

ID_unidade → Unidade(ID_unidade)

Avalia(CPF_funcionario, CPF_aluno, ID!)

CPF_Funcionario → Funcionario(CPF)

CPF_Aluno → Cliente(CPF)

Acompanha(acompanhante, acompanhado)

acompanhante → Funcionario(CPF)

acompanhado → Cliente(CPF)

--GROUP BY / HAVING

```
SELECT ACAD.end_cidade, ACAD.nome,  
COUNT(*) AS Quantidade  
FROM matricula M, unidade ACAD  
WHERE M.id_unidade = ACAD.id_unidade  
GROUP BY ACAD.end_cidade, ACAD.nome  
HAVING COUNT(*) > 1;
```


Procedimentos

```
CREATE OR REPLACE PROCEDURE print_nome(auxCPF  
VARCHAR) IS auxNome VARCHAR(50);
```

```
BEGIN
```

```
    SELECT P.nome INTO auxNome
```

```
    FROM pessoa P
```

```
    WHERE p.cpf = auxCPF;
```

```
    dbms_output.put_line(auxNome);
```

```
EXCEPTION
```

```
    WHEN NO_DATA_FOUND THEN
```

```
        dbms_output.put_line('CPF NOT FOUND');
```

```
END;
```

```
BEGIN
```

```
print_nome('74055189079');
```

```
END;
```

Procedimentos

```
CREATE OR REPLACE PROCEDURE city(cidade VARCHAR) IS
```

```
CURSOR nomes IS
```

```
    SELECT nome
```

```
    FROM PESSOA
```

```
    WHERE end_cidade = cidade;
```

```
aux VARCHAR(50);
```

```
BEGIN
```

```
    OPEN nomes;
```

```
    LOOP
```

```
        FETCH nomes INTO aux;
```

```
        EXIT WHEN nomes%NOTFOUND;
```

```
        dbms_output.put_line(aux);
```

```
    END LOOP;
```

```
    CLOSE nomes;
```

```
END;
```

```
BEGIN
```

```
city('Recife');
```

```
END;
```

Função

```
CREATE OR REPLACE FUNCTION get_funcao(auxCPF  
VARCHAR) RETURN VARCHAR IS auxFuncao  
VARCHAR(20);
```

```
BEGIN  
    SELECT F.funcao INTO auxFuncao  
    FROM funcionario F  
    WHERE F.cpf = auxCPF;  
  
    RETURN auxFuncao;
```

```
EXCEPTION  
    WHEN NO_DATA_FOUND THEN  
        RETURN 'Funcionario não cadastrado';  
  
END;
```

```
BEGIN  
    dbms_output.put_line(get_funcao( '74055189079' ));  
    ;  
END;
```

```
BEGIN  
    dbms_output.put_line(get_funcao( '74055189079' ));  
    ;  
END;
```


Gatilho

```
CREATE OR REPLACE TRIGGER indicador_membro
```

```
AFTER INSERT ON pessoa
```

```
REFERENCING NEW AS P
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    dbms_output.put_line('O membro ' || :P.nome || ' FOI CADASTRADA  
COM SUCESSO');
```

```
END;
```

```
INSERT INTO pessoa
```

```
VALUES(74055184579, 'Aroldo', 'Avenida Chaves', 'Teresina', 64071250);
```

Gatilho

```
CREATE OR REPLACE TRIGGER indicador_academia
```

```
AFTER INSERT ON unidade
```

```
REFERENCING NEW AS ACAD
```

```
FOR EACH ROW
```

```
BEGIN
```

```
    dbms_output.put_line('A ACADEMIA ' || :ACAD.nome || ' FOI  
REGISTRADA COM SUCESSO');
```

```
END;
```

```
INSERT INTO unidade
```

```
VALUES(00100, 'DO FINO PRO GRANDE', 'Rua das Flores', 'Recife',  
52071250);
```

