



UNIVERSIDADE FEDERAL DE MATO GROSSO
ARAGUAIA
CIÊNCIA DA COMPUTAÇÃO

WILLYAN JOSUÉ BASTOS SILVA
WESLEY ANTONIO JUNIOR DOS SANTOS

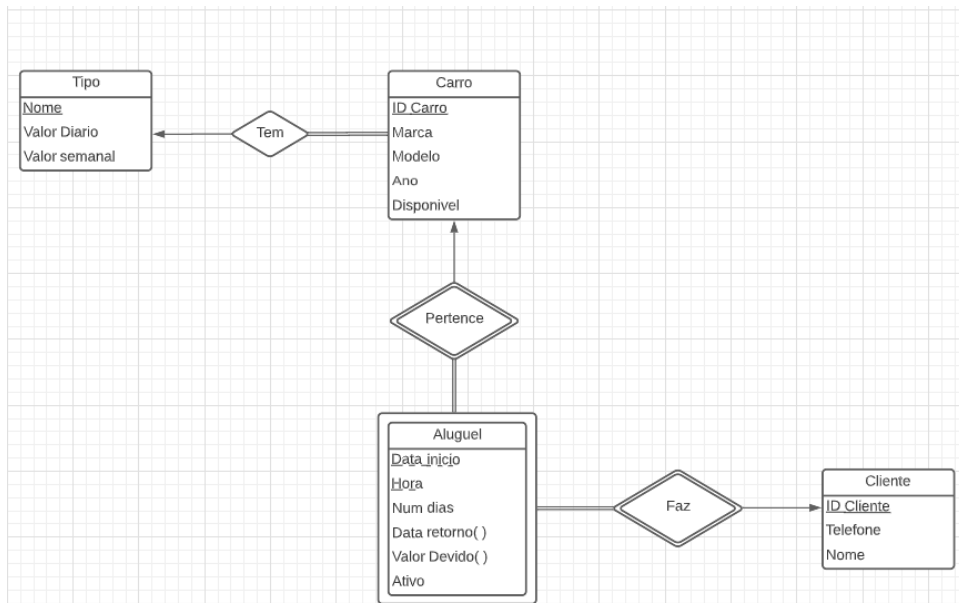
Universidade Federal De Mato Grosso – UFMT

Barra Do Garças

2022

RELATÓRIO DO TRABALHO DE LAORATÓRIO DE BANCO DE DADOS

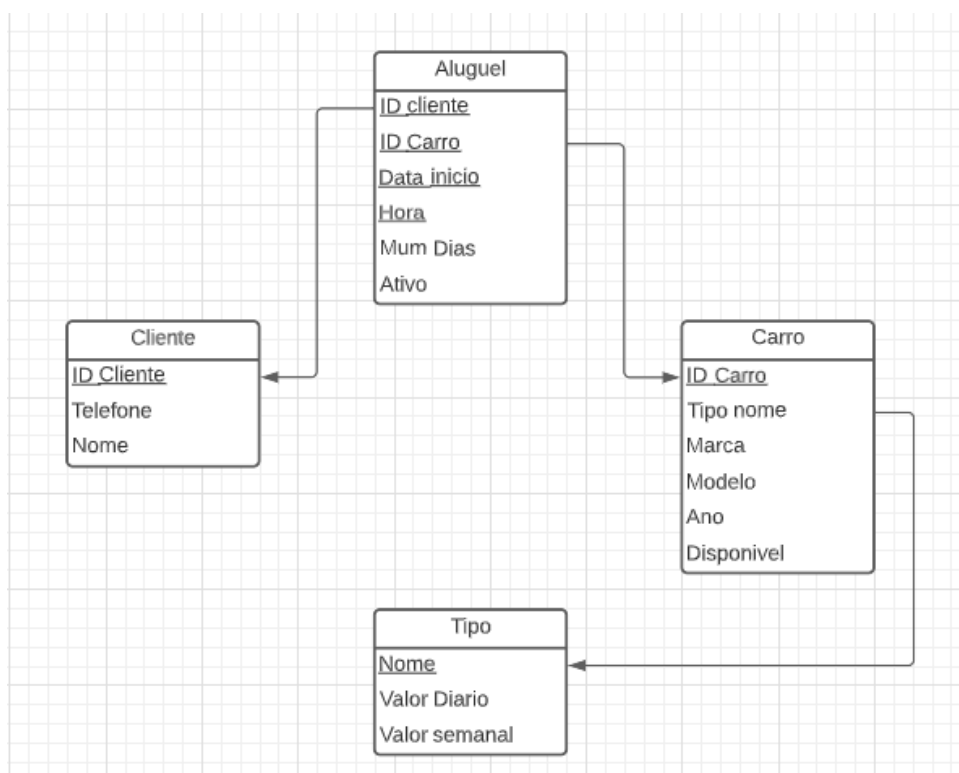
A hierarquia das tabelas é: Cliente e Carro, Aluguel e Tipo



Como visto acima no Diagrama EER, não existem muitas diferenças em comparação ao que se é pedido na especificação do trabalho. Foi inserido o atributo 'disponível', do tipo booleano na tabela Carro para que possamos ter um maior controle sobre o nosso veículo estar ou não alugado. Também foi inserido o atributo 'hora', na tabela Aluguel, para que possamos ter uma informação extra quanto ao aluguel do veículo.

A tabela Aluguel foi criada como identidade fraca, pois ela precisava tanto do id_cliente quanto do id_carro para que possa ser identificada.

Modelo Relacional:



Tendo em vista o Diagrama EER mostrado anteriormente, a partir de tal, foi-se montado o modelo relacional do mesmo.

Criação das Tabelas no MySQL:

```
create table Cliente (  
    cpf int not null,  
    telefone varchar(12) not null,  
    nome varchar(50) not null,  
    primary key (cpf)  
);  
create table Tipo (  
    nome varchar(20) not null,  
    valor_diario float not null,  
    valor_semanal float not null,  
    primary key (nome)  
);  
create table Carro (  
    id_carro int auto_increment not null,  
    tipo_nome varchar(20) not null,  
    marca varchar(20) not null,  
    modelo varchar(20) not null,  
    ano varchar(20) not null,  
    disponivel boolean not null,  
    primary key (id_carro),  
    foreign key (tipo_nome) references Tipo(nome)  
);  
create table Aluguel (  
    cpf int not null,  
    id_carro int not null,  
    data_inicio date not null,  
    hora TIME,  
    aluguel_ativo BOOLEAN not null,  
    num_dias int not null,  
    valor float not null,  
    primary key (cpf, id_carro, data_inicio, hora),  
    foreign key (id_carro) references Carro(id_carro),  
    foreign key (cpf) references Cliente(cpf)  
);
```

A única alteração feita na criação das tabelas em relação ao último diagrama apresentado, é uma mudança de nomes de atributo das tabelas Cliente e Aluguel, ambas têm um atributo que anteriormente era denominado de ID_Cliente e atualmente está sendo chamado de CPF.

Cientes inseridos:

cpf	telefone	nome
18444	66999785567	Wesley
23145	66999786688	Juliana
55555	66999887723	linder
56687	66999785580	Paulo
66666	66999785580	Pedro
77777	66999785599	Carlos
84652	66999785569	John
84875	66999785527	Cleudinet
88888	66999785590	Maria
99999	66999785525	Beth

10 rows in set (0.00 sec)

Carros inseridos:

id_carro	tipo_nome	marca	modelo	ano	disponivel
1	COMPACTO	Fiat	Mobi	2020	1
2	COMPACTO	Renault	Kwid	2018	0
3	COMPACTO	Volkswagen	Up	2019	1
4	COMPACTO	Fiat	Uno	2015	1
5	COMPACTO	Nissan	March	2018	1
6	COMPACTO	Volkswagen	Fox	2016	1
7	COMPACTO	Toyota	Etios	2019	1
8	COMPACTO	Volkswagen	Gol	2010	1
9	COMPACTO	Chevrolet	Celta	2013	1
10	COMPACTO	Chevrolet	Onix	2020	1
11	MEDIO	Volkswagen	Polo	2015	1
12	MEDIO	Fiat	Argo	2020	1
13	MEDIO	Hyundai	HB20	2017	1
14	MEDIO	Toyota	Yaris Hatch	2019	1
15	GRANDE	Hyundai	Tucson	2019	1
16	GRANDE	Chery	Tiggor	2020	1
17	GRANDE	Chevrolet	Spin	2016	1
18	SUV	Fiat	Pulse	2020	1
19	SUV	Jac	T40 Plus	2019	1
20	SUV	Peugeot	2008	2021	1
21	SUV	Renault	Duster	2019	1
22	VAN	Fiat	Ducato	2018	1
23	VAN	Mercedes-Benz	Sprinter 313	2020	1
24	VAN	Peugeot	Boxer	2021	1
25	CAMINHÃO	Volvo	FH 540	2015	1
26	CAMINHÃO	Scania	R450	2017	1
27	CAMINHÃO	Volkswagen	Delivery 11.180	2016	1
28	CAMINHÃO	Volvo	FH 460	2017	1
29	CAMINHÃO	Volkswagen	Constellation 24.280	2018	1
30	CAMINHÃO	Mercedes-Benz	Actros 2651	2019	1
31	COMPACTO	Fiat	Mobi	2019	1

31 rows in set (0.00 sec)

Tipos inseridos:

nome	valor_diario	valor_semanal
CAMINHAO	250	1400
COMPACTO	40	224
GRANDE	73	408.8
MEDIO	52	291.2
SUV	80	448
VAN	101	565.6

6 rows in set (0.00 sec)

Aluguéis feitos:

cpf	id_carro	data_inicio	hora	aluguel_ativo	num_dias	valor
18444	2	2022-10-29	18:37:15	0	3	120
18444	2	2022-10-29	18:41:18	0	10	344
18444	2	2022-10-29	18:41:27	1	10	344

3 rows in set (0.00 sec)

Procedimentos e Triggers no Banco de Dados:

Procedimento para a inserção de Tipos de carro

```
create procedure insere_tipo(in nome varchar(20), in valor float)
```

```
begin
```

```
    declare semanal float;
```

```
    set semanal = 7 * (valor - (valor * 0.20));
```

```
    insert into Tipo values(nome, valor, semanal);
```

```
end
```

Procedimento para a inserção de carro

```
create procedure insere_carro(in tipo varchar(20), in marca varchar(20),  
                             in modelo varchar(20), in ano varchar(5))
```

```
begin
```

```
    insert into Carro(tipo_nome, marca, modelo, ano, disponivel)
```

```
    values(tipo, marca, modelo, ano, TRUE);
```

```
end
```

Procedimento para a inserção de Clientes

```
create procedure insere_cliente(in cpf int, in telefone varchar(12), in nome varchar(50))
```

```
begin
```

```
    insert into Cliente(cpf, telefone, nome) values(cpf, telefone, nome);
```

```
end
```

```

# Procedimento para a inserção do Aluguel
create procedure insere_aluguel(in cpf int, in id_carro int,
                               in data_inicio date, in num_dias int)
begin
    declare tipo_car varchar(20);
    declare val_diario float default 0.0;
    declare val_semanal float default 0.0;
    declare valor float default 0.0;
    declare disp boolean;
    declare dias int;

    set dias = num_dias;
    set tipo_car = (select c.tipo_nome from Carro as c where c.id_carro = id_carro);
    set val_diario = (select a.valor_diario from tipo as a where a.nome = tipo_car);
    set val_semanal = (select a.valor_semanal from tipo as a where a.nome = tipo_car);
    set disp = (select c.disponivel from Carro as c where c.id_carro = id_carro);

    if disp = TRUE then
        update Carro as c
        set c.disponivel = FALSE
        where c.id_carro = id_carro;

        while dias >= 7 do
            set dias = dias - 7;
            set valor = valor + val_semanal;
        end while;

        set valor = valor + (dias * val_diario);

        insert into Aluguel(cpf, id_carro, data_inicio, hora, num_dias, aluguel_ativo, valor)
        values(cpf, id_carro, data_inicio, TIME(CURRENT_TIMESTAMP()), num_dias, TRUE,
valor);
        end if;
    end

end

# trigger que atualiza os carros quando um aluguel e finalizado
CREATE TRIGGER atualiza_carro_e_aluguel AFTER UPDATE ON aluguel
FOR EACH ROW
BEGIN
    UPDATE carro SET disponivel = True where id_carro = new.id_carro;
END

# Procedimento que atualiza o valor dos carros atrasados e a cada dia que
# passar da devolucao adiciona mais uma diaria ao valor
CREATE PROCEDURE atrasados(IN id_c int, in quant_dias int)
begin
    declare data_atual date;
    declare valor_diario float;
    set data_atual = (select curdate());
    set valor_diario = (select distinct tip.valor_diario from Carro as Car, tipo as tip
where Car.tipo_nome = tip.nome and Car.id_carro = id_c);

    UPDATE aluguel
    set valor = valor + ( valor_diario * quant_dias), data_inicio = data_atual, num_dias = 1
    where id_carro = id_c;
end

```