



**FUNDAÇÃO UNIVERSIDADE  
FEDERAL DE MATO GROSSO DO SUL**

**ENGENHARIA DE COMPUTAÇÃO  
CIÊNCIAS DA COMPUTAÇÃO**

BANCO DE DADOS  
PROFESSOR(A): VANESSA BORGES

LUIZ GUSTAVO PAREDES DELGADO 2017.1905.067-8  
JOÃO VITOR NANTES DE SILVA MATOS  
2017.1905.025-2 GIOVANNI PRATTO 2016.1904.061-1

**CAMPO GRANDE - MS  
01/10/2018**

LUIZ GUSTAVO PAREDES DELGADO 2017.1905.067-8  
JOÃO VITOR NANTES DE SILVA MATOS  
2017.1905.025-2 GIOVANNI PRATTO 2016.1904.061-1

Trabalho apresentado ao professor(a)  
Vanessa Borges, como forma de  
avaliação da disciplina de  
banco de dados.

Orientador(a): Vanessa Borges

## **RESUMO**

Este trabalho visa empregar os métodos abordados em sala de aula para a implementação de um banco de dados em uma empresa. No qual usamos o exemplo de um consultório odontológico onde realizamos a entrevista com o cliente final para especificar todos os requisitos e, assim, definir um modelo conceitual consistente e detalhado.

## ÍNDICE

1. INTRODUÇÃO	04
2. LEVANTAMENTO DE REQUISITOS	05
3. CARACTERÍSTICAS DA ENTIDADE	06
4. ESQUEMA CONCEITUAL	08
5. ESQUEMA RELACIONAL	09
6. SCRIPT E CONSULTAS SQL	11
7. REFERÊNCIAS	12

## **INTRODUÇÃO**

Neste projeto foram aplicados os conceitos para criação e implementação de um sistema de banco de dados em um determinado consultório de uma clínica odontológica a fim de resolver e otimizar o sistema de cadastro de pacientes, cadastro de funcionários, consultar data e hora de procedimentos agendados, gerar atestados clínicos, laudos e prescrições médicas .

## **DETALHAMENTO**

### **Levantamento de Requisitos**

1. O sistema cadastra o usuário;
2. O sistema pede que o usuário tenha um número de cpf;
3. O sistema cadastra, modifica, exclui e imprime relatórios;
4. O sistema será ágil no procedimento das consultas;
5. O sistema terá informação dos profissionais envolvidos;
6. O sistema poderá informar a data e hora dos procedimentos;
7. O sistema gerará um atestado caso o usuário solicite;
8. O sistema informará ao dentista data e hora dos procedimentos.

## CARACTERÍSTICAS DA ENTIDADE

### 1 - Pessoa:

- CPF (PK)
- Nome
- data de nascimento
- Estado civil
- Telefone
- Endereço(rua, cep, número)

### 2 - Atendimento:

- Nro\_atendimento(PK)
- Nro\_pront
- Cod\_func
- Data\_consulta
- Hora\_consulta
- Data\_atual

### 3 - Funcionário:

- Cod\_func(PK)
- Nome\_Funcionário

### 4 - Secretária:

- Cod\_func(PK)
- idiomas
- tipo

### 5 - Dentista:

- Cod\_func(PK)
- CRM
- especialização

### 6 - Paciente:

- Nro\_pront(PK)
- Pcpf

### 7 - Procedimento:

- Nro\_procedimento(PK)
- Cod\_func
- Nro\_pront
- data\_consulta

**8 - Cirurgia:**

- Nro\_procedimento(PK)
- procedimentos

**9 - Consulta:**

- Nro\_procedimento(PK)
- medicamentos

**11- Atestado:**

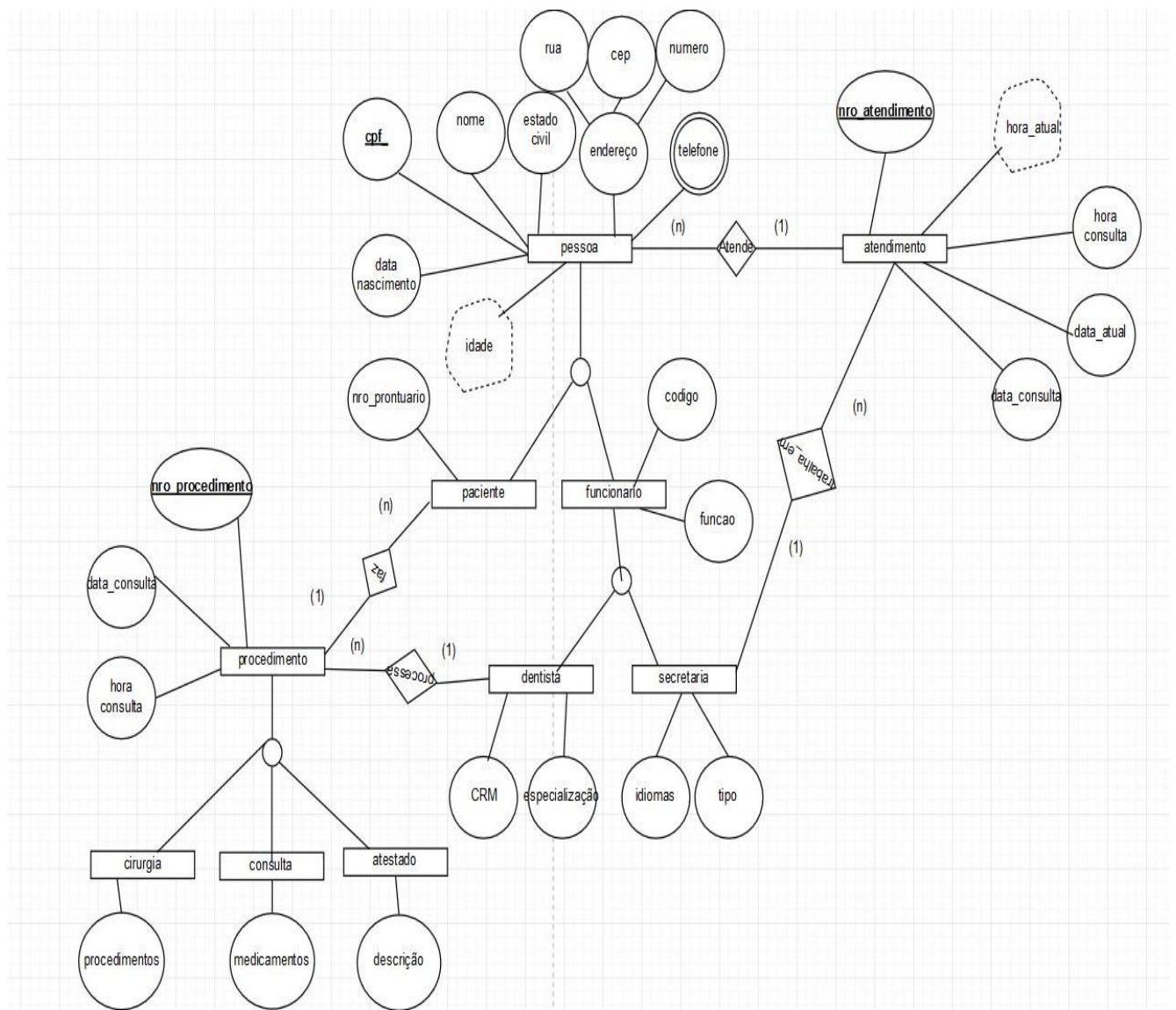
- Nro\_procedimento(PK)
- descrição



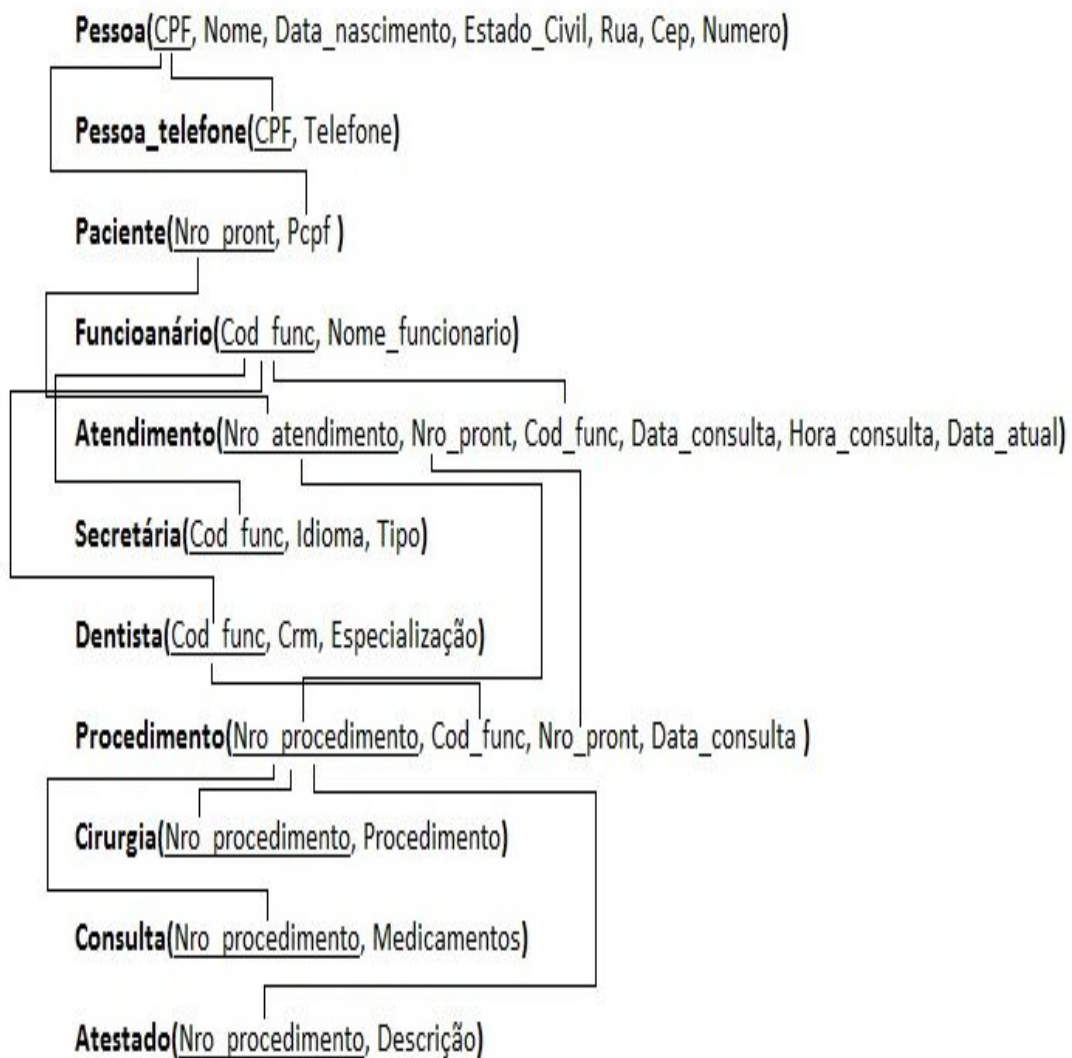
## ESQUEMA CONCEITUAL

### Modelo Entidade-Relacionamento

No modelo conceitual usamos o conceito de entidades para poder empregar os conceitos comuns às quais desejamos armazenar os dados, as entidades são representadas através de um retângulo, no caso do relacionamento entre as entidades com um significado específico dentro do mundo real, objetos do mundo real não ocorrem de forma isolada, eles se associam ou se vinculam, os relacionamentos são representados em forma de um triângulo.



## MODELO RELACIONAL



## SCRIPT E CONSULTAS SQL

#####

```
#criando um database clinica
create database if not exists Clinica
default character set utf8
default collate utf8_general_ci;
```

#####

```
#Tabela pessoa
create table if not exists pessoa(
cpf int(11) not null primary key,
nome varchar(30) not null,
estado_civil enum ('S','C') not null,
nascimento date not null,
endereco varchar(30) not null,
telefone int(10) not null
)default charset = utf8;
```

insert into pessoa values

```
('1234567890','Giovanni Pratto','C','1984-05-26','Rua:Paradise n.12','99123'),
('1234567891','Carol Pratto','C','1984-05-26','Rua:cecilio alves correa n.07','991234'),
('1234567892','Pedro Lima','S','1984-05-26','Rua:Pampinas n.123','991235'),
('1234567893','Giovanna ferrari','S','1984-05-26','Rua:Brilhante n.329','991236'),
('1234567894','Lautreamont F.','C','1984-05-26','Rua:Poneia n.05','991237');
```

#####

```
#Tabela paciente
create table if not exists paciente(
nra_pront int(5) not null primary key,
pcpf int(11) not null unique,
foreign key (pcpf) REFERENCES pessoa(cpf) ON UPDATE NO ACTION ON DELETE NO ACTION
)default charset = utf8;
```

insert into paciente values

```
('12341','1234567890'),
('12342','1234567891'),
('12343','1234567892'),
('12344','1234567893'),
('12345','1234567894');
```

#####

```
#tabela funcionario
create table if not exists funcionario(
cod_func int(5) not null primary key,
nome_funcionario varchar(30) not null
)default charset = utf8;
```

```
insert into funcionario values
('33333','Paulo Roberto'),
('44444','Vania Gabinete'),
('55555','Lario Larica'),
('66666','Betania Furacao');
```

```
#####
```

```
#tabela atendimento
```

```
create table if not exists atendimento(
nro_atendimento int(5) not null primary key ,
nro_pront int(5) not null unique,
cod_func int(5) not null ,
data_consulta date not null,
hora varchar(10) not null,
data_atual date not null,
foreign key (nro_pront) REFERENCES paciente(nra_pront) ON UPDATE NO ACTION ON DELETE NO
ACTION,
foreign key (cod_func) REFERENCES secretaria(cod_func) ON UPDATE NO ACTION ON DELETE NO
ACTION
)default charset = utf8;
```

```
insert into atendimento values
```

```
('01','12341' , '33333','2018-05-26','18:50','2018-05-20'),
('02','12342' , '44444','2018-04-16','12:30','2018-04-10'),
('03','12343' , '33333','2018-07-06','11:30','2018-06-04'),
('04','12344' , '44444','2018-06-29','14:25','2018-06-20'),
('05','12345' , '33333','2018-03-13','07:30','2018-03-01');
```

```
#####
```

```
#tabela secretaria
```

```
create table if not exists secretaria(
cod_func int(5) not null primary key,
idioma enum ('ingles','portugues','espanhol') not null,
tipo enum ('chefe','auxiliar') not null,
foreign key (cod_func) REFERENCES funcionario(cod_func) ON UPDATE NO ACTION ON DELETE
NO ACTION
)default charset = utf8;
```

```
insert into secretaria values
```

```
('33333','ingles','auxiliar'),
('44444','espanhol','chefe');
```

```
#####
```

```
#tabela dentista
```

```
create table if not exists dentista(
cod_func int(5) not null primary key,
crm int(5) not null,
especializacao enum ('ortodontia','implantodontia') not null,
```

```
foreign key (cod_func) REFERENCES funcionario(cod_func) ON UPDATE NO ACTION ON DELETE NO ACTION
)default charset = utf8;
```

```
insert into dentista values
('55555','567','ortodontia'),
('66666','568','implantodontia');
```

```
#####
```

```
#tabela procedimento
```

```
create table if not exists procedimento(
nro_procedimento int(5) not null primary key,
cod_func int(5) not null ,
nro_pront int(5) not null ,
data_consulta date not null,
foreign key (nro_procedimento) REFERENCES atendimento(nro_atendimento) ON UPDATE NO ACTION ON DELETE NO ACTION,
foreign key (cod_func) REFERENCES dentista(cod_func) ON UPDATE NO ACTION ON DELETE NO ACTION,
foreign key (nro_pront) REFERENCES atendimento(nro_pront) ON UPDATE NO ACTION ON DELETE NO ACTION
)default charset = utf8;
```

```
insert into procedimento values
('01' ,'55555','12341','2018-05-26'),
('02' ,'66666','12342','2018-04-16'),
('03' ,'55555','12343','2018-07-06'),
('04' ,'66666','12344','2018-06-29'),
('05' ,'55555','12345','2018-03-13');
```

```
#####
```

```
#tabela cirurgia
```

```
create table if not exists cirurgia(
nro_procedimento int(5) not null primary key,
procedimento enum ('implante','aparelho') not null,
foreign key (nro_procedimento) REFERENCES procedimento(nro_procedimento) ON UPDATE NO ACTION ON DELETE NO ACTION
)default charset = utf8;
```

```
insert into cirurgia values
```

```
('01','aparelho'),
('02','implante'),
('03','aparelho'),
('04','implante'),
('05','aparelho');
```

```
#####
```

```
#tabela consulta
```

```
create table if not exists consulta(
nro_procedimento int(5) not null primary key,
```

```
medicamento varchar(20) not null,  
foreign key (nro_procedimento) REFERENCES procedimento(nro_procedimento) ON UPDATE NO  
ACTION ON DELETE NO ACTION  
)default charset = utf8;
```

```
insert into consulta values  
( '01','dipirona'),  
( '02','antibiotico'),  
( '03','antialergico'),  
( '04','Pasta de dente'),  
( '05','fio de dental ');
```

```
#####
```

```
#tabela atestado
```

```
create table if not exists atestado(  
nro_procedimento int(5) not null primary key,  
descricao varchar(50) not null,  
foreign key (nro_procedimento) REFERENCES procedimento(nro_procedimento) ON UPDATE NO  
ACTION ON DELETE NO ACTION  
)default charset = utf8;
```

```
insert into atestado values  
( '01','dor de dente'),  
( '02','cirurgia implante'),  
( '03','manutencao'),  
( '04','cirurgia implante'),  
( '05','troca de borracha');
```

## Relatórios Das Consultas,dos funcionários e dos pacientes

#consulta os procedimento e funcionario

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf and p.nome = 'Giovanni Pratto' )
inner join dentista g on (a.cod_func = g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento =
t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join
atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf and p.nome = 'Carol Pratto' ) inner
join dentista g on (a.cod_func = g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento =
t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join
atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf and p.nome = 'Pedro Lima' ) inner
join dentista g on (a.cod_func = g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento =
t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join
atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf and p.nome = 'Giovanna ferrari' )
inner join dentista g on (a.cod_func = g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento =
t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join
atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf and p.nome = 'Lautreamont F.' )
inner join dentista g on (a.cod_func = g.cod_func)
```

```
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento = t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf ) inner join dentista g on (a.cod_func
= g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento = t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

**Consulta número 1:** O motivo desta consulta foi selecionar o número do atendimento, nome do paciente, data do atendimento, hora do atendimento, funcionário (Secretaria(o)) que realizou o agendamento, idioma (inglês ou espanhol) que ele fala e tipo (auxiliar ou chefe). Necessidade desta consulta foi ter o controle dos atendimentos realizados.

```
select b.nro_atendimento, c.nome,b.data_consulta, b.hora, f.nome_funcionario, d.idioma, d.tipo
from paciente a inner join atendimento b on (b.nro_pront = a.nra_pront ) inner join pessoa c on(a.pcpf = c.cpf ) inner join secretaria d on (b.cod_func = d.cod_func)
inner join funcionario f on(d.cod_func = f.cod_func)
group by b.nro_pront ;
```

**Consulta número 2:** O motivo desta consulta foi selecionar o número do procedimento, nome do paciente, código do funcionário, nome do funcionário (dentista), descrição do problema, procedimento que irá ser realizado e medicamento que irá ser usado após o termino da consulta. Necessidade desta consulta foi ter controle dos procedimentos realizados.

```
select a.nro_procedimento, p.nome, a.cod_func, r.nome_funcionario,t.procedimento,y.medicamento,
u.descricao
from procedimento a inner join atendimento b on (a.nro_pront = b.nro_pront) inner join paciente c
on(b.nro_pront = c.nra_pront) inner join pessoa p on (p.cpf = c.pcpf ) inner join dentista g on (a.cod_func
= g.cod_func)
inner join funcionario r on (r.cod_func = g.cod_func) inner join cirurgia t on (a.nro_procedimento = t.nro_procedimento) inner join consulta y on (a.nro_procedimento = y.nro_procedimento) inner join atestado u on (a.nro_procedimento = u.nro_procedimento)
group by p.nome order by a.nro_procedimento ;
```

**Consulta número 3:** O motivo desta consulta foi selecionar o cpf, nome do paciente, estado civil, nascimento, endereço e número do prontuário. Necessidade desta consulta foi ter o controle dos prontuários dos pacientes.

```
select distinct *from pessoa p,paciente f
where p.cpf = f.pcpf;
```



## **REFERÊNCIAS BIBLIOGRÁFICAS**

ELMASRI, R.; NAVATHE, S. B. Sistemas de Banco de Dados. 6. ed. Pearson Education Brasil, 2011.