

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
PARAÍBA**
CAMPUS - CAJAZEIRAS

PROJETO DE BANCO DE DADOS

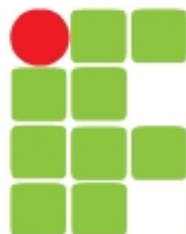
CLÍNICA MÉDICA

Cajazeiras

Abril/2009

Sumário

1. PRIMEIRA ETAPA DO PROJETO	3
1.1. Análise de Requisitos	4
1.2. Diagrama Entidade - Relacionamento	5
1.3. Dicionário de Dados Conceitual.....	6
2. SEGUNDA ETAPA DO PROJETO	11
2.1. Mapeamento Entidade - Relacionamento	12
2.2. Dicionário Lógico de Dados.....	13
2.3. Scripts SQL (Criação das Tabelas).....	27
3. TERCEIRA ETAPA DO PROJETO	34
3.1. Scripts SQL (Inserções)	35
3.2. Scripts SQL (Consultas).....	41
3.3. Scripts SQL (Visões).....	44
3.4. Scripts PL/Pg SQL (Procedimentos Armazenados).....	45



PRIMEIRA ETAPA DO PROJETO

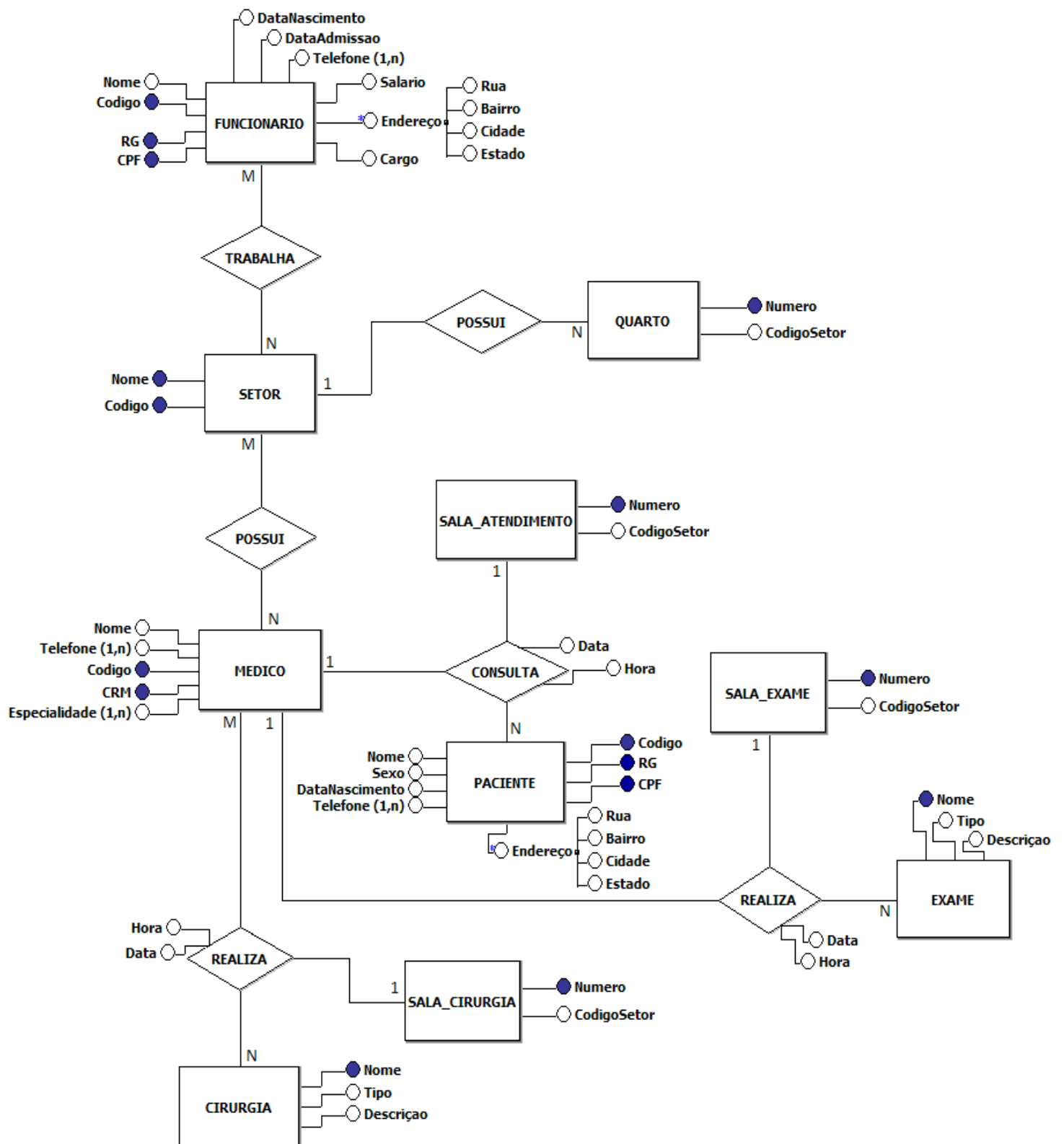
CLÍNICA MÉDICA

1.1. Análise de requisitos

Sistema para uma clínica médica

Uma clínica médica é dividida em setores (ortopédico, traumatológico, ginecológico etc.). Cada setor possui um nome, um código, funcionários e médicos. Um funcionário possui um código, nome, RG, CPF, data de nascimento, data de admissão, telefone, endereço (que é composto por rua, bairro, cidade/estado), salário e cargo. Cada médico possui um código, nome, telefone, CRM, especialidade. Um médico pode ter formação em uma ou diversas especialidades. Um médico pode atender em vários setores. Cada setor também possui salas, e quartos. Cada sala, assim como os quartos possuem um número único, e o código do setor em que estão localizados. Uma sala pode ser de atendimento, exame, ou cirúrgica. Uma consulta envolve um médico de determinado setor e um paciente, sendo informado ao mesmo a sala, data e hora, podendo ou não ser requisitado um exame, e/ou cirurgia. Em caso de exame, o paciente é informado sobre a sala, data, e a hora em que o exame será realizado. Em caso de cirurgia, o paciente será informado da sala, data, hora, e o quarto em que ficará internado. Cada exame, assim como cirurgia, possui código, tipo e descrição. Um paciente possui código, nome, sexo, data de nascimento, identidade, CPF, telefone e endereço, podendo ou não estar filiado a um convênio. Um convênio possui um nome e um código.

1.2. Diagrama Entidade – Relacionamento



1.3. Dicionário de Dados Conceitual

Entidade SETOR:

SETOR: guarda as informações referentes a determinado setor.

Atributos da entidade SETOR:

Nome: atributo criado para guardar o nome do setor.

Codigo: atributo criado para guardar o código de identificação de cada setor.

Relacionamento(s) da entidade SETOR:

Trabalha: relaciona a entidade setor com a entidade funcionário, onde um funcionário pode trabalhar em vários setores, e em um setor podem trabalhar vários funcionários.

Possui: relaciona a entidade setor com a entidade médico, onde um setor pode possuir vários médicos, assim como um médico pode pertencer a vários setores.

Possui: relaciona a entidade setor com a entidade quarto, onde um setor pode possuir vários quartos, enquanto um quarto pertence apenas a um setor.

Entidade FUNCIONARIO:

FUNCIONARIO: guarda as informações referentes aos funcionários que trabalham na clínica.

Atributos da entidade FUNCIONARIO:

Nome: atributo criado para guardar o nome do funcionário.

Codigo: atributo criado para guardar o código de identificação de cada funcionário.

RG: atributo criado para guardar o número da identidade do funcionário.

CPF: atributo criado para guardar o número do CPF do funcionário.

DataNascimento: atributo criado para guardar a data do nascimento de determinado funcionário.

DataAdmissao: atributo criado para guardar a data em que o funcionário foi admitido.

Telefone: atributo criado para guardar o numero do(s) telefone(s) do funcionário.

Salario: atributo criado para guardar o valor do salário que o funcionário recebe da clínica.

Endereço: atributo criado para guardar as informações do endereço do funcionário.

Cargo: atributo criado para guardar o nome do cargo exercido pelo funcionário.

Relacionamento(s) da entidade FUNCIONARIO

Trabalha: relaciona a entidade funcionário com a entidade setor, onde um funcionário pode trabalhar em vários setores, e em um setor podem trabalhar vários funcionários.

Entidade QUARTO:

QUARTO: guarda as informações referentes ao(s) quarto(s) que determinado setor possui.

Atributos da entidade QUARTO:

Numero: atributo criado para guardar o número do quarto.

CodigoSetor: atributo criado para guardar o código do setor em que o quarto está localizado.

Relacionamento(s) da entidade QUARTO:

Possui: relaciona a entidade quarto com a entidade setor, onde um setor pode possuir vários quartos, enquanto um quarto pertence apenas a um setor.

Entidade MEDICO:

MEDICO: guarda as informações dos médicos que prestam serviços à clínica.

Atributos da entidade MEDICO:

Nome: atributo criado para guardar o nome do médico.

Telefone: atributo criado para guardar o número do(s) telefone(s) do médico.

Codigo: atributo criado para guardar o código de identificação do médico.

CRM: atributo criado para guardar o número do CRM do médico.

Especialidade: atributo criado para guardar o nome da(s) especialidade(s) do médico.

Relacionamento(s) da entidade MEDICO:

Possui: relaciona a entidade medico com a entidade setor.

Consulta: relaciona a entidade médico com as entidades paciente e sala de atendimento, onde um médico pode realizar consultas a vários pacientes, enquanto um paciente é consultado por um único médico em determinada sala de atendimento (sendo informada a data e hora da realização da consulta).

Realiza: relaciona a entidade médico com as entidades exame e sala de exame, onde um médico pode realizar vários exames, enquanto um exame é realizado por um único médico em determinada sala de exames (sendo informada a data e hora da realização do exame).

Realiza: relaciona a entidade médico com as entidades cirurgia e sala de cirurgia, onde um médico pode realizar várias cirurgias, assim como uma cirurgia pode ser realizada por vários médicos em uma determinada sala de cirurgia (sendo informada a data e hora da realização da cirurgia).

Entidade SALA_ATENDIMENTO:

SALA_ATENDIMENTO: guarda as informações da(s) sala(s) onde são realizadas as consultas médicas de cada setor da clínica.

Atributos da entidade SALA_ATENDIMENTO:

Numero: atributo criado para guardar o numero da sala de atendimento.

CodigoSetor: atributo criado para guardar o código do setor em que a sala de atendimento está localizada.

Relacionamento(s) da entidade SALA_ATENDIMENTO:

Consulta: relaciona a entidade sala de atendimento com as entidades médico e paciente, onde um médico pode realizar consultas a vários pacientes, enquanto um paciente é consultado por um único médico em determinada sala de atendimento (sendo informada a sala, data e hora da realização da consulta).

Entidade PACIENTE:

PACIENTE: guarda as informações dos pacientes que usufruem dos serviços da clínica.

Atributos da entidade PACIENTE:

Nome: atributo criado para guardar o nome do paciente.

Sexo: atributo criado para guardar o sexo do paciente.

DataNascimento: atributo criado para guardar a data do nascimento do paciente.

Telefone: atributo criado para guardar o(s) telefone(s) do paciente.

Codigo: atributo criado para guardar o código de identificação do paciente.

RG: atributo criado para guardar o número da identidade do paciente.

CPF: atributo criado para guardar o número do CPF do paciente.

Endereço: atributo criado para guardar as informações do endereço do paciente.

Relacionamento(s) da entidade PACIENTE:

Consulta: relaciona a entidade paciente com as entidades médico e sala de atendimento, onde um médico pode realizar consultas a vários pacientes, enquanto um paciente é consultado por um único médico em determinada sala de atendimento (sendo informada a sala, data e hora da realização da consulta).

Entidade SALA_EXAME:

SALA_EXAME: guarda as informações da(s) sala(s) onde são realizados os exames.

Atributos da entidade SALA_EXAME:

Numero: atributo criado para guardar o numero da sala de exame.

CodigoSetor: atributo criado para guardar o código do setor em que a sala de exame está localizada.

Relacionamento(s) da entidade SALA_EXAME:

Realiza: relaciona a entidade sala de exame com as entidades médico e exame, onde um médico pode realizar vários exames, enquanto um exame é realizado por um único médico em determinada sala de exames (sendo informada a sala, data e hora da realização do exame).

Entidade EXAME:

EXAME: guarda as informações do exame realizado por um médico para com um paciente.

Atributos da entidade EXAME:

Nome: atributo criado para guardar o nome do exame.

Tipo: atributo criado para guardar o tipo do exame.

Descrição: atributo criado para guardar a descrição do exame.

Relacionamento(s) da entidade EXAME:

Realiza: relaciona a entidade exame com as entidades médico e sala de exame, onde um médico pode realizar vários exames, enquanto um exame é realizado por um único médico em determinada sala de exames (sendo informada a sala, data e hora da realização do exame).

Entidade SALA_CIRURGIA:

SALA_CIRURGIA: guarda as informações da(s) sala(s) onde são realizadas as cirurgias.

Atributos da entidade SALA_CIRURGIA:

Numero: atributo criado para guardar o numero da sala de cirurgia.

CodigoSetor: atributo criado para guardar o código do setor em que a sala de cirurgia está localizada.

Relacionamento(s) da entidade SALA_CIRURGIA:

Realiza: relaciona a entidade sala de cirurgia com as entidades médico e cirurgia, onde um médico pode realizar várias cirurgias, assim como uma cirurgia pode ser realizada por vários médicos em uma determinada sala de cirurgia (sendo informada a sala, data e hora da realização da cirurgia).

Entidade CIRURGIA:

CIRURGIA: guarda as informações da cirurgia a ser realizada por um médico para com um paciente.

Atributos da entidade CIRURGIA:

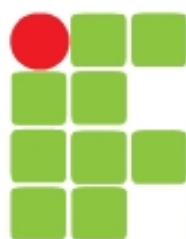
Nome: atributo criado para guardar o nome da cirurgia.

Tipo: atributo criado para guardar o tipo da cirurgia.

Descrição: atributo criado para guardar a descrição da cirurgia.

Relacionamento(s) da entidade SALA_CIRURGIA:

Realiza: relaciona a entidade médico com as entidades cirurgia e sala de cirurgia, onde um médico pode realizar várias cirurgias, assim como uma cirurgia pode ser realizada por vários médicos em uma determinada sala de cirurgia (sendo informada a sala, data e hora da realização da cirurgia).



SEGUNDA ETAPA DO PROJETO

CLÍNICA MÉDICA

2.1. Mapeamento Entidade – Relacionamento

FUNCIONARIO (Codigo, Nome, RG, CPF, DataNascimento, DataAdmissao, Salario, Rua, Bairro, Cidade, Estado, Cargo);

TELEFONE_FUNCIONARIO (CodigoFuncionario, Telefone);

SETOR (Codigo, Nome);

QUARTO (Numero, CodigoSetor);

MEDICO (Codigo, CRM, Nome);

TELEFONE_MEDICO (CodigoMedico, Telefone);

ESPECIALIDADE_MEDICO (CodigoMedico, Especialidade);

SALA_ATENDIMENTO (Numero, CodigoSetor);

PACIENTE (Codigo, Nome, RG, CPF, Rua, Bairro, Cidade, Estado, Sexo, DataNascimento);

TELEFONE_PACIENTE (CodigoPaciente, Telefone);

CIRURGIA (Nome, Tipo, Descrição);

SALA_CIRURGIA (Numero, CodigoSetor);

SALA_EXAME (Numero, CodigoSetor);

EXAME (Nome, Tipo, Descrição);

TRABALHA_SETOR (CodigoFuncionario, CodigoSetor);

POSSUI_MEDICO (CodigoSetor, CodigoMedico);

CONSULTA_PACIENTE (CodigoMedico, NumeroSalaAtendimento, CodigoPaciente, Data, Hora);

REALIZA_CIRURGIA (CodigoMedico, NomeCirurgia, NumeroSalaCirurgia, Data, Hora);

REALIZA_EXAME (CodigoMedico, NomeExame, NumeroSalaExame, Data, Hora).

2.2. Dicionário Lógico de Dados

FUNCIONARIO: Relação que armazena os dados de cada funcionário da clínica			
Atributo	Descrição	Tipo	Restrição
Codigo	Atributo que representa o código de identificação do funcionário	Int	▪ Chave Primária
Nome	Atributo que representa o nome do funcionário	String	▪ Não Nulo
RG	Atributo que representa o RG do funcionário	String	▪ Não Nulo ▪ Único
CPF	Atributo que representa o CPF do funcionário	String	▪ Não Nulo ▪ Único
DataNascimento	Atributo que representa a data de nascimento do funcionário	Date	▪ Não Nulo
DataAdmissao	Atributo que representa a data em que o funcionário foi admitido	Date	▪ Não Nulo
Salario	Atributo que representa o salário do funcionário	Real	▪ Não Nulo
Rua	Atributo que representa a rua onde o funcionário reside	String	▪ Não Nulo
Bairro	Atributo que representa o bairro onde o funcionário reside	String	▪ Não Nulo

Cidade	Atributo que representa a cidade onde o funcionário reside	String	▪ Não Nulo
Estado	Atributo que representa o estado onde o funcionário reside	String	▪ Não Nulo
Cargo	Atributo que representa o cargo que cada funcionário ocupa	String	▪ Não Nulo

TELEFONE_FUNCIONÁRIO: Relação que armazena o(s) telefone(s) de cada funcionário			
Atributo	Descrição	Tipo	Restrição
CodigoFuncionario	Atributo que representa o código de identificação do funcionário	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “FUNCIONARIO” ▪ Chave Primária
Telefone	Atributo que representa o telefone do funcionário	String	<ul style="list-style-type: none"> ▪ Chave Primária

SETOR: Relação que armazena os dados de cada setor da clínica médica

Atributo	Descrição	Tipo	Restrição
Codigo	Atributo que representa o código de identificação do setor	Int	▪ Chave Primária
Nome	Atributo que representa o nome do setor	String	▪ Não Nulo

QUARTO: Relação que armazena os dados de cada quarto da clínica médica

Atributo	Descrição	Tipo	Restrição
Numero	Atributo que representa o número de identificação do quarto	Int	▪ Chave Primária
CodigoSetor	Atributo que representa o código de identificação do setor	Int	▪ Chave Estrangeira que referencia o atributo "Codigo" da relação "SETOR"

MEDICO: Relação que armazena os dados de cada médico da clínica			
Atributo	Descrição	Tipo	Restrição
Codigo	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Primária
CRM	Atributo que representa o CRM do médico	String	<ul style="list-style-type: none"> ▪ Não Nulo ▪ Único
Nome	Atributo que representa o nome do médico	String	<ul style="list-style-type: none"> ▪ Não Nulo

TELEFONE_MEDICO: Relação que armazena o(s) telefone(s) de cada médico			
Atributo	Descrição	Tipo	Restrição
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo "Codigo" da relação "MEDICO" ▪ Chave Primária
Telefone	Atributo que representa o telefone do médico	String	<ul style="list-style-type: none"> ▪ Chave Primária

ESPECIALIDADE_MEDICO: Relação que armazena a(s) especialidade(s) de cada médico			
Atributo	Descrição	Tipo	Restrição
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “MEDICO” ▪ Chave Primária
Especialidade	Atributo que representa a especialidade do médico	String	<ul style="list-style-type: none"> ▪ Chave Primária

SALA_ATENDIMENTO: Relação que armazena os dados de cada sala de atendimento da clínica médica			
Atributo	Descrição	Tipo	Restrição
Numero	Atributo que representa o número de identificação da sala de atendimento	Int	<ul style="list-style-type: none"> ▪ Chave Primária
CodigoSetor	Atributo que representa o código de identificação do setor	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “SETOR”

PACIENTE: Relação que armazena os dados de cada paciente da clínica médica			
Atributo	Descrição	Tipo	Restrição
Codigo	Atributo que representa o código de identificação do paciente	Int	▪ Chave Primária
Nome	Atributo que representa o nome do paciente	String	▪ Não Nulo
RG	Atributo que representa o RG do paciente	String	▪ Não Nulo ▪ Único
CPF	Atributo que representa o CPF do paciente	String	▪ Não Nulo ▪ Único
Rua	Atributo que representa a rua onde o paciente reside	String	▪ Não Nulo
Bairro	Atributo que representa o bairro onde o paciente reside	String	▪ Não Nulo
Cidade	Atributo que representa a cidade onde o paciente reside	String	▪ Não Nulo
Estado	Atributo que representa o estado onde o paciente reside	String	▪ Não Nulo
Sexo	Atributo que representa o sexo do paciente	String	▪ Não Nulo

DataNascimento	Atributo que representa a data de nascimento do paciente	Date	<ul style="list-style-type: none"> ▪ Não Nulo
TELEFONE_PACIENTE: Relação que armazena o(s) telefone(s) de cada paciente			
Atributo	Descrição	Tipo	Restrição
CodigoPaciente	Atributo que representa o código de identificação do paciente	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo "Codigo" da relação "PACIENTE" ▪ Chave Primária
Telefone	Atributo que representa o telefone do paciente	String	<ul style="list-style-type: none"> ▪ Chave Primária

CIRURGIA: Relação que armazena os dados de cada cirurgia da clínica médica			
Atributo	Descrição	Tipo	Restrição
Nome	Atributo que representa o nome da cirurgia	String	<ul style="list-style-type: none"> ▪ Chave Primária
Tipo	Atributo que representa o tipo da cirurgia	String	<ul style="list-style-type: none"> ▪ Não Nulo
Descricao	Atributo que representa a descrição da cirurgia	String	<ul style="list-style-type: none"> ▪ Não Nulo

SALA_CIRURGIA: Relação que armazena os dados de cada sala de cirurgia da clínica médica			
Atributo	Descrição	Tipo	Restrição
Numero	Atributo que representa o número de identificação da sala de cirurgia	Int	▪ Chave Primária
CodigoSetor	Atributo que representa o código de identificação do setor	Int	▪ Chave Estrangeira que referencia o atributo "Codigo" da relação "SETOR"

SALA_EXAME: Relação que armazena os dados de cada sala de exame da clínica médica			
Atributo	Descrição	Tipo	Restrição
Numero	Atributo que representa o número de identificação da sala de exame	Int	▪ Chave Primária
CodigoSetor	Atributo que representa o código de identificação do setor	Int	▪ Chave Estrangeira que referencia o atributo "Codigo" da relação "SETOR"

EXAME: Relação que armazena os dados de cada exame da clínica médica

Atributo	Descrição	Tipo	Restrição
Nome	Atributo que representa o nome do exame	String	▪ Chave Primária
Tipo	Atributo que representa o tipo do exame	String	▪ Não Nulo
Descrição	Atributo que representa a descrição do exame	String	▪ Não Nulo

TRABALHA_SETOR: Relação que armazena os dados do relacionamento entre “FUNCIONARIO” e “SETOR”

Atributo	Descrição	Tipo	Restrição
CodigoFuncionario	Atributo que representa o código de identificação do funcionário	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “FUNCIONARIO” ▪ Chave Primária
CodigoSetor	Atributo que representa o código de identificação do setor	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “SETOR” ▪ Chave Primária

POSSUI_MEDICO: Relação que armazena os dados do relacionamento entre “SETOR” e “MEDICO”			
Atributo	Descrição	Tipo	Restrição
CodigoSetor	Atributo que representa o código de identificação do setor	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “SETOR” ▪ Chave Primária
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “MEDICO” ▪ Chave Primária

CONSULTA_PACIENTE: Relação que armazena os dados do relacionamento entre “MEDICO”, “SALA_ATENDIMENTO” e “PACIENTE”			
Atributo	Descrição	Tipo	Restrição
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> Chave Estrangeira que referencia o atributo “Codigo” da relação “MEDICO” Chave Primária
NumeroSalaAtendimento	Atributo que representa o número de identificação da sala de atendimento	Int	<ul style="list-style-type: none"> Chave Estrangeira que referencia o atributo “Numero” da relação “SALA_ATENDIMENTO” Chave Primária
CodigoPaciente	Atributo que representa o código de identificação do paciente	Int	<ul style="list-style-type: none"> Chave Estrangeira que referencia o atributo “Codigo” da relação “PACIENTE” Chave Primária
Data	Atributo que representa a data de realização da consulta	Date	<ul style="list-style-type: none"> Não Nulo

Hora	Atributo que representa o horário de realização da consulta	Time	<ul style="list-style-type: none"> ▪ Não Nulo
------	---	------	--

REALIZA_CIRURGIA: Relação que armazena os dados do relacionamento entre “MEDICO”, “CIRURGIA” e “SALA_CIRURGIA”			
Atributo	Descrição	Tipo	Restrição
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “MEDICO” ▪ Chave Primária
NomeCirurgia	Atributo que representa o nome da cirurgia	String	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Nome” da relação “CIRURGIA” ▪ Chave Primária
NumeroSalaCirurgia	Atributo que representa o número de identificação da sala de cirurgia	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Numero” da relação “SALA_CIRURGIA” ▪ Chave Primária

Data	Atributo que representa a data de realização da cirurgia	Date	<ul style="list-style-type: none"> ▪ Não Nulo
Hora	Atributo que representa o horário de realização da cirurgia	Time	<ul style="list-style-type: none"> ▪ Não Nulo

REALIZA_EXAME: Relação que armazena os dados do relacionamento entre “MEDICO”, “EXAME” e “SALA_EXAME”			
Atributo	Descrição	Tipo	Restrição
CodigoMedico	Atributo que representa o código de identificação do médico	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Codigo” da relação “MEDICO” ▪ Chave Primária
NomeExame	Atributo que representa o nome do exame	String	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Nome” da relação “EXAME” ▪ Chave Primária

NumeroSalaExame	Atributo que representa o número de identificação da sala de exame	Int	<ul style="list-style-type: none"> ▪ Chave Estrangeira que referencia o atributo “Numero” da relação “SALA_EXAME” ▪ Chave Primária
Data	Atributo que representa a data de realização do exame	Date	<ul style="list-style-type: none"> ▪ Não Nulo
Hora	Atributo que representa o horário de realização do exame	Time	<ul style="list-style-type: none"> ▪ Não Nulo

2.3. Scripts SQL (Criação das Tabelas)

```
CREATE TABLE FUNCIONARIO(
```

```
    Codigo INT,  
    Nome VARCHAR(30) NOT NULL,  
    RG VARCHAR(7) NOT NULL UNIQUE,  
    CPF VARCHAR(14) NOT NULL UNIQUE,  
    DataNascimento DATE NOT NULL,  
    DataAdmissao DATE NOT NULL,  
    Salario REAL NOT NULL,  
    Rua VARCHAR(40) NOT NULL,  
    Bairro VARCHAR(40) NOT NULL,  
    Cidade VARCHAR(40) NOT NULL,  
    Estado VARCHAR(20) NOT NULL,  
    Cargo VARCHAR(20) NOT NULL,  
    PRIMARY KEY(Codigo)
```

```
)
```

```
CREATE TABLE TELEFONE_FUNCIONARIO(
```

```
    CodigoFuncionario INT,  
    Telefone VARCHAR(15),  
    PRIMARY KEY(CodigoFuncionario,Telefone),  
    FOREIGN KEY(CodigoFuncionario) REFERENCES FUNCIONARIO(Codigo) ON DELETE  
    RESTRICT ON UPDATE CASCADE
```

```
)
```

```
CREATE TABLE SETOR(
```

```
    Codigo INT,  
    Nome VARCHAR(15) NOT NULL,  
    PRIMARY KEY(Codigo)
```

```
)
```

```

CREATE TABLE QUARTO(
    Numero INT,
    CodigoSetor INT,
    PRIMARY KEY(Numero),
    FOREIGN KEY(CodigoSetor) REFERENCES SETOR(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

CREATE TABLE MEDICO(
    Codigo INT,
    CRM VARCHAR(10) NOT NULL UNIQUE,
    Nome VARCHAR(30) NOT NULL,
    PRIMARY KEY(Codigo)
)

CREATE TABLE TELEFONE_MEDICO(
    CodigoMedico INT,
    TelefoneMedico VARCHAR(15),
    PRIMARY KEY(CodigoMedico,TelefoneMedico),
    FOREIGN KEY(CodigoMedico) REFERENCES MEDICO(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

CREATE TABLE ESPECIALIDADE_MEDICO(
    CodigoMedico INT,
    Especialidade VARCHAR(20),
    PRIMARY KEY(CodigoMedico,Especialidade),
    FOREIGN KEY (CodigoMedico) REFERENCES MEDICO(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

```

```
CREATE TABLE SALA_ATENDIMENTO(  
    Numero INT,  
    CodigoSetor INT,  
    PRIMARY KEY(Numero),  
    FOREIGN KEY (CodigoSetor) REFERENCES SETOR(Codigo) ON DELETE RESTRICT  
    ON UPDATE CASCADE  
)
```

```
CREATE TABLE PACIENTE(  
    Codigo INT,  
    Nome VARCHAR(30) NOT NULL,  
    RG VARCHAR(7) NOT NULL UNIQUE,  
    CPF VARCHAR(14) NOT NULL UNIQUE,  
    Rua VARCHAR(40) NOT NULL,  
    Bairro VARCHAR(40) NOT NULL,  
    Cidade VARCHAR(40) NOT NULL,  
    Estado VARCHAR(20) NOT NULL,  
    Sexo VARCHAR(9) NOT NULL,  
    DataNascimento DATE NOT NULL,  
    PRIMARY KEY(Codigo)  
)
```

```
CREATE TABLE TELEFONE_PACIENTE(  
    CodigoPaciente INT,  
    Telefone VARCHAR(15),  
    PRIMARY KEY(CodigoPaciente,Telefone),  
    FOREIGN KEY(CodigoPaciente) REFERENCES PACIENTE(Codigo) ON DELETE  
    RESTRICT ON UPDATE CASCADE  
)
```

```

CREATE TABLE CIRURGIA(
    Nome VARCHAR(30),
    Tipo VARCHAR(30) NOT NULL,
    Descrição VARCHAR(300) NOT NULL,
    PRIMARY KEY(Nome)
)

CREATE TABLE SALA_CIRURGIA(
    Numero INT,
   CodigoSetor INT,
    PRIMARY KEY(Numero),
    FOREIGN KEY (CodigoSetor) REFERENCES SETOR(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

CREATE TABLE SALA_EXAME(
    Numero INT,
   CodigoSetor INT,
    PRIMARY KEY(Numero),
    FOREIGN KEY(CodigoSetor) REFERENCES SETOR(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

CREATE TABLE EXAME(
    Nome VARCHAR(30),
    Tipo VARCHAR(30) NOT NULL,
    Descrição VARCHAR(300) NOT NULL,
    PRIMARY KEY(Nome)
)

```

```

CREATE TABLE TRABALHA_SETOR(
    CodigoFuncionario INT,
    CodigoSetor INT,
    PRIMARY KEY(CodigoFuncionario,CodigoSetor),
    FOREIGN KEY (CodigoFuncionario) REFERENCES FUNCIONARIO(Codigo) ON
    DELETE RESTRICT ON UPDATE CASCADE
    FOREIGN KEY (CodigoSetor) REFERENCES SETOR(Codigo) ON DELETE RESTRICT
    ON UPDATE CASCADE
)

CREATE TABLE POSSUI_MEDICO(
    CodigoSetor INT,
    CodigoMedico INT,
    PRIMARY KEY(CodigoSetor,CodigoMedico),
    FOREIGN KEY (CodigoSetor) REFERENCES SETOR(Codigo) ON
    DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (CodigoMedico) REFERENCES MEDICO(Codigo) ON
    DELETE RESTRICT ON UPDATE CASCADE
)

CREATE TABLE CONSULTA_PACIENTE(
    CodigoMedico INT,
    NumeroSalaAtendimento INT,
    CodigoPaciente INT,
    Data DATE NOT NULL,
    Hora TIME NOT NULL,
    PRIMARY KEY(CodigoMedico,NumeroSalaAtendimento,CodigoPaciente),
    FOREIGN KEY (CodigoMedico) REFERENCES MEDICO(Codigo) ON
    DELETE RESTRICT ON UPDATE CASCADE,
    FOREIGN KEY (NumeroSalaAtendimento)

```

```

REFERENCES    SALA_ATENDIMENTO(Numero)

ON DELETE RESTRICT ON UPDATE CASCADE,

FOREIGN KEY (CodigoPaciente) REFERENCES PACIENTE(Codigo)

ON DELETE RESTRICT ON UPDATE CASCADE

)

CREATE TABLE REALIZA_CIRURGIA(

    CodigoMedico INT,

    NomeCirurgia VARCHAR(30),

    NumeroSalaCirurgia INT,

    Data DATE NOT NULL,

    Hora TIME NOT NULL,

    PRIMARY KEY(CodigoMedico, NomeCirurgia, NumeroSalaCirurgia),

    FOREIGN KEY (CodigoMedico) REFERENCES MEDICO(Codigo) ON

    DELETE RESTRICT ON UPDATE CASCADE,

    FOREIGN KEY (NomeCirurgia)

    REFERENCES CIRURGIA(Nome)

    ON DELETE RESTRICT ON UPDATE CASCADE,

    FOREIGN KEY (NumeroSalaCirurgia) REFERENCES SALA_CIRURGIA(Numero)

    ON DELETE RESTRICT ON UPDATE CASCADE

)

CREATE TABLE REALIZA_EXAME(

    CodigoMedico INT,

    NomeExame VARCHAR(30),

    NumeroSalaExame INT,

    Data DATE NOT NULL,

    Hora TIME NOT NULL,

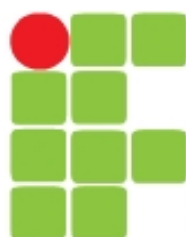
    PRIMARY KEY(CodigoMedico, NomeExame, NumeroSalaExame),

    FOREIGN KEY (CodigoMedico) REFERENCES MEDICO(Codigo) ON

```



```
DELETE RESTRICT ON UPDATE CASCADE,  
FOREIGN KEY (NomeExame)  
REFERENCES EXAME(Nome)  
ON DELETE RESTRICT ON UPDATE CASCADE,  
FOREIGN KEY (NumeroSalaExame) REFERENCES SALA_EXAME(Numero)  
ON DELETE RESTRICT ON UPDATE CASCADE  
)
```



TERCEIRA ETAPA DO PROJETO

CLÍNICA MÉDICA

3.1. Scripts SQL (Inserções)

Tabela FUNCIONARIO:

```
INSERT INTO FUNCIONARIO VALUES  
(1111,'Pedro','1111111','11111111111111','01/01/1983','02/01/2000',1800.00,'Antônio Amorim','Centro','Cajazeiras','PB','Atendente');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1112,'Maria','1111112','11111111111112','04/03/1985','02/01/2000',1800.00,'Pedro Firmino','Centro','Cajazeiras','PB','Atendente');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1113,'José','1111113','11111111111113','01/05/1982','02/03/2000',800.00,'Altemar Dutra','Cangote do Urubú','Cajazeiras','PB','Fachineiro');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1114,'João','1111114','11111111111114','07/12/1987','04/01/2001',800.00,'Lacerda Gomes','Centro','Cajazeiras','PB','Fachineiro');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1115,'Josefa','1111115','11111111111115','12/04/1984','21/09/2005',1400.00,'Amado Batista','Centro','Cajazeiras','PB','Vigilante');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1116,'Armando','1111116','11111111111116','30/07/1983','20/01/2003',1400.00,'Coronel Andrade','Asa I','Cajazeiras','PB','Vigilante');
```

```
INSERT INTO FUNCIONARIO VALUES  
(1117,'Cláudia','1111117','11111111111117','21/03/1980','02/02/2003',2000.00,'Michael Jackson','Rabo da Gata','Cajazeiras','PB','Secretária');
```

Tabela TELEFONE_FUNCIONARIO:

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1111,'083 99762015');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1112,'083 99863614');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1113,'083 88762015');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1114,'083 87762015');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1115,'083 99672015');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1116,'083 99734015');
```

```
INSERT INTO TELEFONE_FUNCIONARIO VALUES (1117,'083 99798715');
```

Tabela SETOR:

```
INSERT INTO SETOR VALUES (0001,'Ortopedia');
```

```
INSERT INTO SETOR VALUES (0002,'Dermatologia');
```

INSERT INTO SETOR VALUES (0003,'Fonologia');
INSERT INTO SETOR VALUES (0004,'Cirurgia');
INSERT INTO SETOR VALUES (0005,'Cardiologia');
INSERT INTO SETOR VALUES (0006,'Tomografia')

Tabela QUARTO:

INSERT INTO QUARTO VALUES (1001,'1');
INSERT INTO QUARTO VALUES (1002,'2');
INSERT INTO QUARTO VALUES (1003,'3');
INSERT INTO QUARTO VALUES (1004,'4');
INSERT INTO QUARTO VALUES (1005,'5');
INSERT INTO QUARTO VALUES (1006,'6')

Tabela MEDICO:

INSERT INTO MEDICO VALUES (2221,'1235PB','Dr. Washington');
INSERT INTO MEDICO VALUES (2222,'1236CE','Dr. Hernanes');
INSERT INTO MEDICO VALUES (2223,'1237PE','Dr. Zé Luiz');
INSERT INTO MEDICO VALUES (2225,'1238RN','Dr. Jean');
INSERT INTO MEDICO VALUES (2226,'1239MG','Dr. Bosco');
INSERT INTO MEDICO VALUES (2227,'1232CE','Dr. André Dias')

Tabela TELEFONE_MEDICO:

INSERT INTO TELEFONE_MEDICO VALUES (2221,'083 88922010');
INSERT INTO TELEFONE_MEDICO VALUES (2222,'083 88722070');
INSERT INTO TELEFONE_MEDICO VALUES (2223,'083 88652032');
INSERT INTO TELEFONE_MEDICO VALUES (2225,'083 88628215');
INSERT INTO TELEFONE_MEDICO VALUES (2226,'083 88622920');
INSERT INTO TELEFONE_MEDICO VALUES (2227,'083 88624040')

Tabela ESPECIALIDADE_MEDICO:

INSERT INTO ESPECIALIDADE_MEDICO VALUES (2221,'Clínico Geral');
INSERT INTO ESPECIALIDADE_MEDICO VALUES (2222,'Ortopedista');

```

INSERT INTO ESPECIALIDADE_MEDICO VALUES (2223,'Pneumologista');
INSERT INTO ESPECIALIDADE_MEDICO VALUES (2225,'Dentista');
INSERT INTO ESPECIALIDADE_MEDICO VALUES (2226,'Radiologista');
INSERT INTO ESPECIALIDADE_MEDICO VALUES (2227,'Neurologista')

```

Tabela SALA_ATENDIMENTO:

```

INSERT INTO SALA_ATENDIMENTO VALUES (1122,1);
INSERT INTO SALA_ATENDIMENTO VALUES (1133,2);
INSERT INTO SALA_ATENDIMENTO VALUES (1144,3);
INSERT INTO SALA_ATENDIMENTO VALUES (1155,4);
INSERT INTO SALA_ATENDIMENTO VALUES (1166,5);
INSERT INTO SALA_ATENDIMENTO VALUES (1177,6)

```

Tabela PACIENTE:

```

INSERT INTO PACIENTE VALUES (123,'José Alfredo
Meneguel','1287345','12398754345','Carlos Pereira','Centro','São José de
Piranhas','PB','Masculino','20/08/1979');

INSERT INTO PACIENTE VALUES (133,'Maria do
Socorro','1212095','12323454865','Maria Perante','ASA
II','Cajazeiras','PB','Feminino','02/06/1979');

INSERT INTO PACIENTE VALUES (143,'Nando Peres
Lira','1212145','96823454345','Miguel
Filho','Centro','Uiraúna','PB','Masculino','25/05/1983');

INSERT INTO PACIENTE VALUES (153,'Antônio
Albuquerque','3712345','37223454345','Merles
Neto','Centro','Horebe','PB','Masculino','28/10/1989');

INSERT INTO PACIENTE VALUES (163,'Carolina Ferreira
Lima','9812345','05423454345','Almir Sáter','Alto','Bonito de Santa
Fé','PB','Feminino','22/07/1976');

INSERT INTO PACIENTE VALUES (173,'Carlos Miller
Filho','3912345','19523454345','Alfredo
Juncão','Centro','Conceição','PB','Masculino','22/07/1995');

INSERT INTO PACIENTE VALUES (183,'Marcia
Gonçalves','5612345','05523454345','Lacerda
Contras','Centro','Marizópolis','PB','Feminino','22/07/1964')

```

Tabela TELEFONE_PACIENTE:

INSERT INTO TELEFONE_PACIENTE VALUES (123,'083 88323674');

INSERT INTO TELEFONE_PACIENTE VALUES (133,'083 88435674');

INSERT INTO TELEFONE_PACIENTE VALUES (143,'083 88342344');

INSERT INTO TELEFONE_PACIENTE VALUES (153,'083 88323144');

INSERT INTO TELEFONE_PACIENTE VALUES (163,'083 88398074');

INSERT INTO TELEFONE_PACIENTE VALUES (173,'083 88327474');

INSERT INTO TELEFONE_PACIENTE VALUES (183,'083 88309684')

Tabela CIRURGIA:

INSERT INTO CIRURGIA VALUES ('Traqueostomia','Traquéia','Cirurgia realizada na traquéia');

INSERT INTO CIRURGIA VALUES ('Tireoidectomia subtotal','Tireóide','Cirurgia realizada na tireóide');

INSERT INTO CIRURGIA VALUES ('Gastrostomia','Estômago','Cirurgia realizada no estômago');

INSERT INTO CIRURGIA VALUES ('Prostatectomia Suprapúbica','Próstata','Cirurgia realizada na próstata');

INSERT INTO CIRURGIA VALUES ('Mamoplastia','Mamas','Cirurgia realizada nas mamas')

Tabela SALA_CIRURGIA:

INSERT INTO SALA_CIRURGIA VALUES (2233,1);

INSERT INTO SALA_CIRURGIA VALUES (2244,2);

INSERT INTO SALA_CIRURGIA VALUES (2255,3);

INSERT INTO SALA_CIRURGIA VALUES (2266,4);

INSERT INTO SALA_CIRURGIA VALUES (2277,5);

INSERT INTO SALA_CIRURGIA VALUES (2288,6)

Tabela SALA_EXAME:

INSERT INTO SALA_EXAME VALUES (3344,1);

INSERT INTO SALA_EXAME VALUES (3355,2);

INSERT INTO SALA_EXAME VALUES (3366,3);

INSERT INTO SALA_EXAME VALUES (3377,4);

INSERT INTO SALA_EXAME VALUES (3388,5);

INSERT INTO SALA_EXAME VALUES (3399,6)

Tabela EXAME:

INSERT INTO EXAME VALUES ('Colposcopia','Ginecológico','Exame do cólo do útero para prevenção do câncer ginecológico');

INSERT INTO EXAME VALUES ('Densitometria óssea','Ortopédico','Exame que avalia a quantidade de cálcio nos ossos');

INSERT INTO EXAME VALUES ('Ecocardiografia','Cardiológico','Exame que mostra as estruturas do coração, através de ondas sonoras');

INSERT INTO EXAME VALUES ('Eletromiografia','Muscular','Exame que avalia o funcionamento dos músculos');

INSERT INTO EXAME VALUES ('Laparoscopia','Abdominal','Exame que visualiza o interior da cavidade abdominal')

Tabela TRABALHA_SETOR:

INSERT INTO TRABALHA_SETOR VALUES (1111,1);

INSERT INTO TRABALHA_SETOR VALUES (1112,2);

INSERT INTO TRABALHA_SETOR VALUES (1113,3);

INSERT INTO TRABALHA_SETOR VALUES (1114,4);

INSERT INTO TRABALHA_SETOR VALUES (1115,5);

INSERT INTO TRABALHA_SETOR VALUES (1116,6)

Tabela POSSUI_MEDICO:

INSERT INTO POSSUI_MEDICO VALUES (1,2221);

INSERT INTO POSSUI_MEDICO VALUES (2,2222);

INSERT INTO POSSUI_MEDICO VALUES (3,2223);

INSERT INTO POSSUI_MEDICO VALUES (4,2225);

INSERT INTO POSSUI_MEDICO VALUES (5,2226);

INSERT INTO POSSUI_MEDICO VALUES (6,2227)

Tabela CONSULTA_PACIENTE:

```
INSERT INTO CONSULTA_PACIENTE VALUES (2221,1122,123,'05/04/2009','11:30');
INSERT INTO CONSULTA_PACIENTE VALUES (2222,1133,133,'06/04/2009','10:30');
INSERT INTO CONSULTA_PACIENTE VALUES (2223,1144,143,'07/04/2009','10:30');
INSERT INTO CONSULTA_PACIENTE VALUES (2225,1155,153,'08/04/2009','08:30');
INSERT INTO CONSULTA_PACIENTE VALUES (2226,1166,163,'09/04/2009','10:30');
INSERT INTO CONSULTA_PACIENTE VALUES (2227,1177,173,'10/04/2009','07:30')
```

Tabela REALIZA_CIRURGIA:

```
INSERT INTO REALIZA_CIRURGIA VALUES
(2221,'Traqueostomia',2233,'12/03/2009','09:30');

INSERT INTO REALIZA_CIRURGIA VALUES (2222,'Tireoidectomia
subtotal',2244,'5/03/2009','10:30');

INSERT INTO REALIZA_CIRURGIA VALUES
(2223,'Gastrostomia',2255,'16/03/2009','10:30');

INSERT INTO REALIZA_CIRURGIA VALUES (2226,'Prostatectomia
Suprapúbica',2266,'18/03/2009','09:45');

INSERT INTO REALIZA_CIRURGIA VALUES
(2227,'Mamoplastia',2277,'20/03/2009','08:00')
```

Tabela REALIZA_EXAME:

```
INSERT INTO REALIZA_EXAME VALUES (2221,'Colposcopia',3344,'02/06/2009','10:30');

INSERT INTO REALIZA_EXAME VALUES (2222,'Densitometria
óssea',3355,'03/06/2009','11:40');

INSERT INTO REALIZA_EXAME VALUES
(2223,'Ecocardiografia',3366,'03/06/2009','09:40');

INSERT INTO REALIZA_EXAME VALUES
(2225,'Eletromiografia',3377,'03/06/2009','08:00');

INSERT INTO REALIZA_EXAME VALUES (2227,'Laparoscopia',3388,'03/06/2009','07:30')
```


3.2. Scripts SQL (Consultas)

1 - Selecionar o código, nome E RG de todos os funcionários que ganham acima de 1000 reais:

```
SELECT CODIGO, NOME, RG  
FROM FUNCIONARIO  
WHERE SALARIO<1000
```

2 - Selecionar o nome, data de nascimento, data de admissão e salário dos funcionários que moram em Cajazeiras e trabalham como atendentes:

```
SELECT nome AS NomeDoEmpregado, datanascimento,dataadmissao,salario  
FROM FUNCIONARIO  
WHERE cidade='Cajazeiras' AND cargo='Atendente'
```

3 – Calcule caso o funcionário que trabalha como vigilante recebesse um aumento de 20% no salário:

```
SELECT codigo, nome, 1.20 * salario AS novosalario  
FROM FUNCIONARIO  
WHERE cargo='Vigilante'
```

4 – Selecione o nome do médico onde a sua especialidade seja dentista:

```
SELECT M.nome, M.crm, EM.especialidade  
FROM MEDICO M, ESPECIALIDADE_MEDICO EM  
WHERE M.codigo=EM.codigomedico AND EM.especialidade = 'Dentista'
```

5 – Selecione o nome do paciente de nome 'Maria do Socorro', o dia a hora, o médico que a atendeu e a sala de atendimento:

```
SELECT P.nome AS Nomepaciente, M.nome AS nomemedico, SA.numero AS  
numerodasala, CP.data, CP.hora  
  
FROM PACIENTE P, CONSULTA_PACIENTE CP, MEDICO M, SALA_ATENDIMENTO SA  
  
WHERE P.NOME LIKE '%Maria do Socorro%' AND P.codigo=CP.codigopaciente AND  
M.codigo=CP.codigomedico AND SA.numero=CP.numerosalaatendimento
```

6 – Selecione o nome e a cidade dos pacientes que moram no centro de sua cidade em ordem alfabética:

```
SELECT nome, cidade  
  
FROM PACIENTE  
  
WHERE bairro='Centro'  
  
ORDER BY nome ASC
```

7 – Selecione o nome, RG e os salários dos funcionários que ganham menos que os atendentes:

```
SELECT Nome, rg, salario FROM FUNCIONARIO  
  
WHERE Salario < ALL  
  
(SELECT Salario FROM FUNCIONARIO WHERE cargo='Atendente')
```

8 – Selecione o nome, CRM e a especialidade de cada médico:

```
SELECT M.Nome, M.crm, ED.especialidade  
  
FROM (MEDICO M RIGHT OUTER JOIN ESPECIALIDADE_MEDICO ED ON  
M.Codigo=ED.Codigomedico)
```

9 – Conte a quantidade de salários distintos que é pago na clínica, o menor salário, o maior e a média salarial:

```
SELECT COUNT (DISTINCT Salario) AS TiposDeSalario, MIN (salario)
        AS MenorSalario, MAX (salario) AS MaiorSlario,
        AVG (salario) AS MediaSalarial
FROM funcionario
```

10 – Verifique as condições de salários dos funcionários:

```
SELECT nome,rg, salario,
CASE
    WHEN salario<=800 THEN 'Ganha Pouco'
    WHEN salario>=1400 AND salario<=1800 THEN 'Ganha Bem'
    WHEN salario>1800 THEN 'Ganha Muito Bem'
END AS remunecao FROM funcionário
```

3.3. Scripts SQL (Visões)

1 - Setor com seus respectivos funcionários:

```
CREATE VIEW SETOR_COM_FUNCIONARIO AS  
  
SELECT S.Nome AS Setor, F.Nome AS Funcionario, F.Cargo AS Cargo  
  
FROM SETOR S, FUNCIONARIO F, TRABALHA_SETOR TS  
  
WHERE S.Codigo = TS.CodigoSetor AND F.Codigo = TS.CodigoFuncionario
```

2 – Setor com seus respectivos médicos:

```
CREATE VIEW SETOR_COM_MEDICO AS  
  
SELECT S.Nome AS Setor, M.Nome AS Medico, M.CRM AS CRM  
  
FROM SETOR S, MEDICO M, POSSUI_MEDICO PM  
  
WHERE S.Codigo = PM.CodigoSetor AND M.Codigo = PM.CodigoMedico
```

3.4. Scripts PL/PG SQL (Procedimentos Armazenados)

1 – Recupera a quantidade de funcionários que trabalham em determinado setor:

```
CREATE OR REPLACE FUNCTION QuantidadeFuncionariosSetor(VARCHAR)
RETURNS INTEGER
AS '
DECLARE
    codigo ALIAS FOR $1;
    Quantidade INTEGER;
BEGIN
    SELECT INTO Quantidade COUNT(*) FROM TRABALHA_SETOR WHERE
CodigoFuncionario = Codigo;
    RETURN Quantidade;
END
' LANGUAGE plpgsql;
```

2 – Recupera o próximo código de funcionário a ser inserido:

```
CREATE OR REPLACE FUNCTION ProximoCodigoPaciente() RETURNS INTEGER AS'
DECLARE
    MaiorCodigoPaciente INTEGER;
BEGIN
    SELECT INTO MaiorCodigoPaciente MAX(Codigo) FROM Paciente;
    RETURN MaiorCodigoPaciente + 10;
END
' LANGUAGE plpgsql;
```

