

MC536 - Projeto 2

Gerenciamento de dados de pacientes de um hospital

Erick Seiji Furukawa - 170600

Jonathas Sardinha - 176522

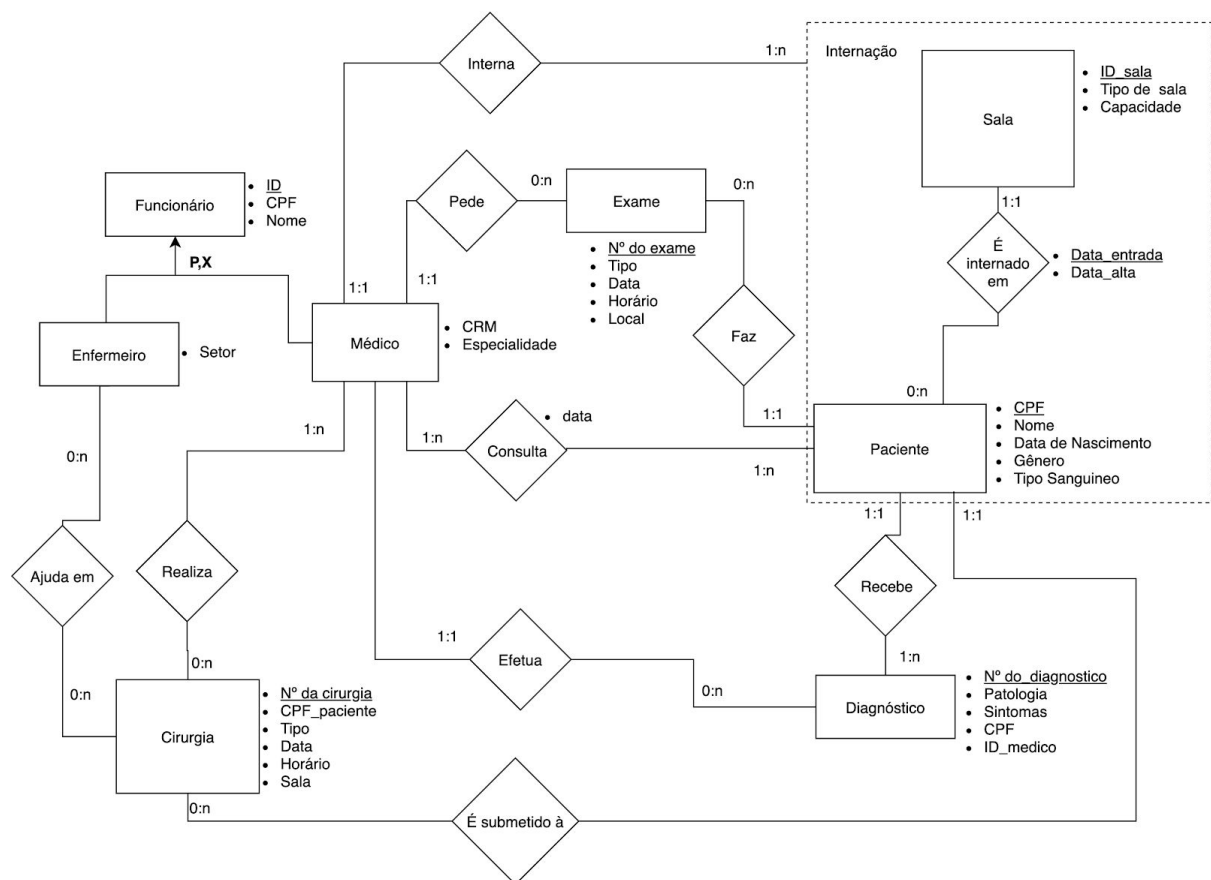
Luciano Zago - 182835

Luis Fernando - 182913

Marcelo Martins - 183303

Tiago Petená - 187700

Modelagem ER



Banco de Dados

- **Entidades**

Funcionário(ID, CPF, Nome)

Enfermeiro(ID, Setor)

Médico(ID, CRM, Especialidade)

Exame(Nº do exame, Tipo, Data, Horário, Local, CPF_paciente)
 Cirurgia(Nº da cirurgia, Tipo, Data, Horário, Sala)
 Paciente(CPF, Nome, Data de Nascimento, Gênero, Tipo Sanguíneo)
 Diagnóstico(Nº do diagnostico, Patologia, Sintomas, CPF, ID_medico)
 Sala(ID_sala, Tipo de sala, Capacidade)

● Agregações

Internacao(CPF, Data_entrada, Data_alta, ID_sala)

● Relacionamentos

Pede(Nº do exame, ID)
 Interna(CPF, Data_entrada, ID)
 Consulta(CPF, ID, data)
 Realiza(ID, Nº da cirurgia)
 Ajuda em(ID, Nº da cirurgia)
 E_submetido_a(Nº da cirurgia, CPF)

Normalização em 3NF

O ESQUEMA INDICADO JÁ ESTÁ EM 3NF.

Obs: O atributo “sintomas” da entidade diagnóstico é uma única caixa de texto, não apresentando diversos valores e, portanto, não infringe às exigências da primeira forma normal.

Consultas programadas

- Dados da ficha de cadastro de um paciente
- Lista de consultas de um paciente
- Lista de exames de um paciente
- Lista de diagnósticos de um paciente
- Lista de cirurgias de um paciente
- Lista de exames pedidos por um médico
- Lista de consultas feitas por um médico

$\rho_{ID=IDDoMedico}(Medico \bowtie Consulta)$

- Lista de diagnósticos feitos por um médico
- Lista de cirurgias
- Lista de pacientes internados
- Sala em que uma dada cirurgia acontecerá/aconteceu
- Sala em que um dado paciente está internado

$\rho_{cpf=cpfDoPaciente}(Sala \bowtie Internado \bowtie Paciente)$

- Lista de enfermeiros
- Lista de médicos

Início da programação do sistema

1 - A demonstração será feita via execução isolada de comandos SQL.

O usuário terá uma interface com uma tela de login que determinará as permissões de acesso ao sistema. Ele poderá selecionar as seguintes opções no menu:

- Caso seja um médico:
 - Cirurgias (mostrará as cirurgias do dia)
 - Pacientes (pesquisar o paciente por CPF ou nome/data de consulta, caso não saiba CPF. Mostrará também os diagnósticos e exames do paciente)
 - Exames (pedirá o CPF de um paciente e ele poderá checar ou adicionar exames a um paciente)
 - Diagnósticos (funcionamento igual à Exames)
- Caso seja um enfermeiro:
 - Cirurgias (mostrará as cirurgias em que ajuda no dia)
 - Internações (pedirá o CPF de um paciente e ele poderá checar as salas de internação de um paciente)
- Caso seja paciente:
 - Cirurgias (mostrará as cirurgias que o paciente tem para fazer, mesmo futuramente)
 - Internações (mostrará a sala em que o paciente está internado)
 - Exames (mostrará os exames que o paciente tem para fazer)
 - Diagnósticos (mostrará os diagnósticos já recebidos pelo paciente em ordem de data)

2 - MySQL

```
CREATE TABLE Funcionario(
    ID      int      NOT NULL,
    CPF     varchar(11) NOT NULL,
    Nome    varchar(50) NOT NULL,
    PRIMARY KEY(ID)
);
```

```
CREATE TABLE Medico(
    ID      int      NOT NULL,
    CRM     varchar(10) NOT NULL,
    Especialidade varchar(50),
    PRIMARY KEY(ID),
```

```

        FOREIGN KEY (ID) REFERENCES Funcionario(ID)
    );

CREATE TABLE Enfermeiro(
    ID      int      NOT NULL,
    Setor varchar(50),
    PRIMARY KEY(ID),
    FOREIGN KEY (ID) REFERENCES Funcionario(ID)
);

CREATE TABLE Paciente(
    CPF      varchar(11) NOT NULL,
    Nome     varchar(50) NOT NULL,
    DataNascimento DATE NOT NULL,
    Genero   char(1)     NOT NULL,
    TipoSang varchar(2) NOT NULL,
    PRIMARY KEY(CPF)
);

CREATE TABLE Exame(
    NExame      int      NOT NULL,
    Tipo        varchar(50) NOT NULL,
    Data        Date,
    Horario     Time(0),
    Local       varchar(50),
    PRIMARY KEY(NExame)
);

CREATE TABLE Diagnostico(
    NDiag int      NOT NULL,
    CPF   varchar(11) NOT NULL,
    IDMedico int     NOT NULL,
    Patologia varchar(100),
    Sintomas  varchar(500),
    PRIMARY KEY(NDiag)
    FOREIGN KEY (CPF) REFERENCES Paciente(CPF),
    FOREIGN KEY (IDMedico) REFERENCES Medico(ID)
);

CREATE TABLE Sala(
    IDSala varchar(5) NOT NULL,
    Tipo varchar(50),
    Capacidade int,
    PRIMARY KEY(IDSala)
);

```

```

CREATE TABLE Cirurgia(
    NCirurgia    int    NOT NULL,
    IDSala int    NOT NULL,
    CPF  varchar(11)  NOT NULL,
    Tipo  varchar(50)  NOT NULL,
    Data  Date  NOT NULL,
    Horario      Time(0)      NOT NULL,
    PRIMARY KEY(NCirurgia),
    FOREIGN KEY (IDSala) REFERENCES Sala(IDSala),
    FOREIGN KEY (CPF) REFERENCES Paciente(CPF)
);

```

```

CREATE TABLE Internacao(
    IDSala int    NOT NULL,
    CPF  varchar(11)  NOT NULL,
    Data_entrada Date  NOT NULL,
    Data_alta      Date,
    PRIMARY KEY (CPF, Data_entrada)
    FOREIGN KEY (IDSala) REFERENCES Sala(IDSala),
    FOREIGN KEY (CPF) REFERENCES Paciente(CPF)
);

```

```

CREATE TABLE Consulta(
    ID int  NOT NULL,
    CPF  varchar(11)  NOT NULL,
    Data Date  NOT NULL,
    PRIMARY KEY (ID,CPF,Data)
    FOREIGN KEY (ID) REFERENCES Medico(ID),
    FOREIGN KEY (CPF) REFERENCES Paciente(CPF)
);

```

3 - Informações sobre o conteúdo do bd

(a) Descrever como o banco de dados estará sendo alimentado:

O Banco de dados será alimentado através de inserções manuais dos dados, realizados por alguém capaz de, ao receber os dados dos exames, cirurgias, pacientes, etc, inseri-los corretamente no sistema.

(b) Número de tuplas previstas para cada relação criada em (a)

Utilizando os dados do hospital, temos que, dentro da tabela funcionários, haverá uma média de 3300 tuplas, enquanto que, para a tabela paciente, serão inseridas mais de 10000 tuplas diariamente, no qual esses pacientes estarão realizando consultas, exames, recebendo diagnósticos (por volta de 2.5 milhões por ano) que também serão adicionados ao banco de dados.

(c) Dificuldades imaginadas na implementação

Como o sistema não está sendo projetado para ser acessado apenas por pessoas tecnicamente capacitadas a administrá-lo, deve haver um empenho em tornar a utilização do banco de dados o mais simples possível, de forma a fazer com que a utilização do sistema facilite de fato as operações cotidianas do hospital, organizando os dados e agilizando o acesso às informações necessárias para as atividades hospitalares.

Mudanças de rumo

Algumas mudanças foram feitas para facilitar a implementação do sistema, entre elas:

- A partir das sugestões da PED, uma nova agregação “internação” foi criada, eliminando o relacionamento triplo que existia no trabalho anterior entre “sala”, “médico” e “paciente”. Além disso, enfermeiro agora possui um atributo: “setor” e a relação “verifica” foi retirada.
- A chave primária de sala foi alterada, de “Numero. andar. bloco” para apenas “ID_sala”, que será uma única caixa de texto com duas letras e três número, da forma XX111.
- Diagnóstico agora tem os parâmetros de identificação do médico e do paciente, para facilitação na criação de tabelas.
- A relação “acontece em” foi retirada, pois o atributo “sala” foi adicionado à entidade “cirurgia”.
- O campo “cadastro” de funcionário foi alterado para “ID” para ficar com um nome mais intuitivo.
- Como operações de consulta não são relacionamentos, a agregação Histórico deixou de existir, visto que esse era o único relacionamento do qual participava.