

Dupla: Thiago Sant' Helena e Adriel Kirch

Modelo Lógico Textual

PlanoSaude(idPlano, nome)
Fabricante(CNPJFabricante, nome)
CID10(codCID, descricao)
TipoDeInternacao(idInternacao, descricao)
RegimeInternacao(idRegimeInternacao, descricao)
Estado(idEstado, nome, sigla)
CBOS(idCBOS, descricao)
TipoDeAcomodacao(codAcomodacao, tabela, descricao)
Procedimento(codProcedimento, descricao, tabela)
UnidadeDeSaude(CNPJUnidade, nome, CNES)
Responsavel(CPFResponsavel, nome)
Beneficiario(CPFBeneficiario, nome, numeroCarteiraNacSaude, #idPlano, validadeCarteira, nroCarteira)
Conselho(idConselho, nome, #idEstado)
Medico(CPFMedico, nome, CRM, #idConselho, numeroMedicoConselho)
CBOSMedico(#idCBOS, #CPFMedico)
OPM(codOPM, descricao, tabela, #CNPJFabricante)
GuiaIntenacao(nroGuia, dataEmissaoGuia, dataAutorizacao, caraterInternacao, qtdDiariasSolicitadas, indicacaoClinica, observacaoAutorizacao, dataProvavelAdmissao, qtdDiariasAutorizadas, dataAssMedResponsavel, dataAssBeneficiario, dataAssRespGuia, assMedResponsavel, assBeneficiario, assRespGuia, assinanteBeneficiario, senha, validadeSenha, tipoDoenca, tempo, unidadeTempo, indicacaoAcidente, #CPFBeneficiario, #CNPJUnidadeSolicitada, #codAcomodacao, #idRegimeInternacao, #CPFMedico, #CNPJUnidadeAutorizada)
AcomodacaoAutorizadaGuia(#nroGuia, #codAcomodacao)
ProcedimentoSolicitado(#nroGuia, #codProcedimento, qtdSolicitada, qtdAutorizada)
OPMSolicitada(#nroGuia, #codOPM, quantidade, valorUnidade)
HipoteseGuiaCID(#nroGuia, #codCID)
UnidadeSolicita(#CNPJUnidade, #nroGuia)
Prorrogacao(idProrrogacao, #nroGuia, senha, data, qtdAutorizada, #codAcomodacao, #CPFResponsavel)
ProcedimentoSolicitadoProrrogacao(#idProrrogacao, #codProcedimento, qtdSolicitada, qtdAutorizada)
OPMSolicitadaProrrogacao(#idProrrogacao, #codOPM, quantidade, valorUnidade)

Modelo Físico

```
CREATE TABLE PlanoSaude(  
    idPlano SERIAL PRIMARY KEY NOT NULL,  
    nome VARCHAR(50) NOT NULL  
);  
  
CREATE TABLE Fabricante(  
    CNPJFabricante INT PRIMARY KEY NOT NULL,  
    nome VARCHAR(50) NOT NULL  
);
```

```

CREATE TABLE CID10(
    codCID      INT          PRIMARY KEY NOT NULL,
    descricao  VARCHAR(20) NOT NULL
);

CREATE TABLE TipoDeInternacao(
    idInternacao SERIAL      PRIMARY KEY NOT NULL,
    descricao   VARCHAR(20) NOT NULL
);

CREATE TABLE RegimeInternacao(
    idRegimeInternacao SERIAL PRIMARY KEY NOT NULL,
    descricao      VARCHAR(20) NOT NULL
);

CREATE TABLE Estado(
    idEstado SERIAL      PRIMARY KEY NOT NULL,
    nome      VARCHAR(19) NOT NULL, -- RN tem 19 letras
    sigla     CHAR(2)    NOT NULL
);

CREATE TABLE CBOS(
    idCBOS      SERIAL      PRIMARY KEY NOT NULL,
    descricao  VARCHAR(20) NOT NULL
);

CREATE TABLE TipoDeAcomodacao(
    codAcomodacao INT          PRIMARY KEY NOT NULL,
    descricao     VARCHAR(20) NOT NULL,
    tabela        CHAR(2)      NOT NULL
);

CREATE TABLE Procedimento(
    codProcedimento INT          PRIMARY KEY NOT NULL,
    descricao       VARCHAR(20) NOT NULL,
    tabela          CHAR(2)
);

CREATE TABLE UnidadeDeSaude(
    CNPJUnidade INT          PRIMARY KEY NOT NULL,
    nome        VARCHAR(50) NOT NULL,
    CNES        INT
);

CREATE TABLE Responsavel(
    CPFResponsavel INT          PRIMARY KEY NOT NULL,
    nome           VARCHAR(50) NOT NULL
);

CREATE TABLE Beneficiario(
    CPFBeneficiario INT          PRIMARY KEY NOT NULL,
    nome            VARCHAR(50) NOT NULL,
    numeroCarteiraNacSaude INT,
    idPlano         INT          NOT NULL,
    validadeCarteira DATE        NOT NULL,
    nroCarteira     INT          NOT NULL,
    FOREIGN KEY (idPlano) REFERENCES PlanoSaude (idPlano) ON DELETE RESTRICT
);

CREATE TABLE Conselho(

```

```

    idConselho SERIAL          PRIMARY KEY NOT NULL,
    nome          VARCHAR(50) NOT NULL,
    idEstado      INT           NOT NULL,
    FOREIGN KEY (idEstado) REFERENCES Estado (idEstado) ON DELETE RESTRICT
);

CREATE TABLE Medico(
    CPFMedico      INT           PRIMARY KEY NOT NULL,
    nome           VARCHAR(50) NOT NULL,
    CRM            INT           NOT NULL,
    idConselho     INT           NOT NULL,
    numeroMedicoConselho INT      NOT NULL,
    FOREIGN KEY (idConselho) REFERENCES Conselho (idConselho) ON DELETE RESTRICT
);

CREATE TABLE CBOSMedico(
    idCBOS         INT NOT NULL,
    CPFMedico      INT NOT NULL,
    PRIMARY KEY (idCBOS, CPFMedico),
    FOREIGN KEY (idCBOS) REFERENCES CBOS (idCBOS) ON DELETE RESTRICT,
    FOREIGN KEY (CPFMedico) REFERENCES Medico (CPFMedico) ON DELETE RESTRICT
);

CREATE TABLE OPM(
    codOPM         INT           PRIMARY KEY NOT NULL,
    descricao      VARCHAR(20) NOT NULL,
    tabela         CHAR(2)      NOT NULL,
    CNPJFabricante INT          NOT NULL,
    FOREIGN KEY (CNPJFabricante) REFERENCES Fabricante (CNPJFabricante) ON DELETE RESTRICT
);

CREATE TABLE GuiaInternacao(
    nroGuia        INT           PRIMARY KEY NOT NULL,
    dataEmissaoGuia DATE        NOT NULL,
    dataAutorizacao DATE,
    caraterInternacao CHAR(1) NOT NULL CHECK(caraterInternacao IN ('E', 'U')),
    qtdDiariasSolicitadas INT     NOT NULL,
    indicacaoClinica TEXT        NOT NULL,
    observacaoAutorizacao TEXT,
    dataProvavelAdmissao DATE     NOT NULL,
    qtdDiariasAutorizadas INT     NOT NULL,
    dataAssMedResponsavel DATE    NOT NULL,
    dataAssBeneficiario DATE,
    dataAssRespGuia DATE         NOT NULL,
    assMedResponsavel TEXT        NOT NULL,
    assBeneficiario TEXT,
    assRespGuia    TEXT         NOT NULL,
    assinanteBeneficiario BOOLEAN, -- True se o beneficiário estiver assinando
    senha          INT,
    validadeSenha  DATE,
    tipoDoenca     CHAR(1) NOT NULL CHECK(tipoDoenca IN ('A', 'C')),
    tempo          INT     NOT NULL,
    unidadeTempo   CHAR(1) NOT NULL CHECK(tipoDoenca IN ('A', 'M', 'D')),
    indicacaoAcidente INT     NOT NULL CHECK(indicacaoAcidente IN (0, 1, 2)),
    CPFBeneficiario INT       NOT NULL,
    CNPJUnidadeSolicitada INT  NOT NULL,
    codAcomodacao  INT       NOT NULL,
    idRegimeInternacao INT     NOT NULL,
    CPFMedico      INT       NOT NULL,
    CNPJUnidadeAutorizada INT  NOT NULL,

```

```

FOREIGN KEY (CPFBeneficiario) REFERENCES Beneficiario (CPFBeneficiario) ON DELETE RESTRICT,
FOREIGN KEY (CNPJUnidadeSolicitada) REFERENCES UnidadeDeSaude (CNPJUnidade) ON DELETE RESTRICT,
FOREIGN KEY (codAcomodacao) REFERENCES TipoDeAcomodacao (codAcomodacao) ON DELETE RESTRICT,
FOREIGN KEY (idRegimeInternacao) REFERENCES RegimeInternacao (idRegimeInternacao) ON DELETE RESTRICT,
FOREIGN KEY (CPFMedico) REFERENCES Medico (CPFMedico) ON DELETE RESTRICT,
FOREIGN KEY (CNPJUnidadeAutorizada) REFERENCES UnidadeDeSaude (CNPJUnidade) ON DELETE RESTRICT
);

CREATE TABLE AcomodacaoAutorizadaGuia(
    nroGuia          INT NOT NULL,
    codAcomodacao    INT NOT NULL,
    PRIMARY KEY (nroGuia, codAcomodacao),
    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (codAcomodacao) REFERENCES TipoDeAcomodacao (codAcomodacao) ON DELETE RESTRICT
);

CREATE TABLE ProcedimentoSolicitado(
    nroGuia          INT NOT NULL,
    codProcedimento  INT NOT NULL,
    qtdSolicitada    INT NOT NULL,
    qtdAutorizada    INT NOT NULL,
    PRIMARY KEY (nroGuia, codProcedimento),
    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (codProcedimento) REFERENCES Procedimento (codProcedimento) ON DELETE RESTRICT
);

CREATE TABLE OPMSolicitada(
    nroGuia          INT          NOT NULL,
    codOPM           INT          NOT NULL,
    quantidade       INT          NOT NULL,
    valorUnidade     NUMERIC(2) NOT NULL,
    PRIMARY KEY (nroGuia, codOPM),
    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (codOPM) REFERENCES OPM (codOPM) ON DELETE RESTRICT
);

CREATE TABLE HipoteseGuiaCID(
    nroGuia INT NOT NULL,
    codCID  INT NOT NULL,
    PRIMARY KEY (nroGuia, codCID),
    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (codCID) REFERENCES CID10 (codCID) ON DELETE RESTRICT
);

CREATE TABLE UnidadeSolicita(
    CNPJUnidade INT NOT NULL,
    nroGuia     INT NOT NULL,
    PRIMARY KEY (CNPJUnidade, nroGuia),
    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (CNPJUnidade) REFERENCES UnidadeDeSaude (CNPJUnidade) ON DELETE RESTRICT
);

CREATE TABLE Prorrogaçao(
    idProrrogaçao INT PRIMARY KEY NOT NULL,
    nroGuia       INT NOT NULL,
    senha         INT NOT NULL,
    data          DATE NOT NULL,
    qtdAutorizada INT NOT NULL,
    codAcomodacao INT NOT NULL,
    CPFResponsavel INT NOT NULL,

```

```

    FOREIGN KEY (nroGuia) REFERENCES GuiaInternacao (nroGuia) ON DELETE RESTRICT,
    FOREIGN KEY (CPFResponsavel) REFERENCES Responsavel (CPFResponsavel) ON DELETE RESTRICT,
    FOREIGN KEY (codAcomodacao) REFERENCES TipoDeAcomodacao (codAcomodacao) ON DELETE RESTRICT
);

CREATE TABLE ProcedimentoSolicitadoProrrogacao(
    idProrrogacao    INT NOT NULL,
    codProcedimento  INT NOT NULL,
    qtdSolicitada    INT NOT NULL,
    qtdAutorizada    INT NOT NULL,
    PRIMARY KEY (idProrrogacao, codProcedimento),
    FOREIGN KEY (idProrrogacao) REFERENCES Prorrogacao (idProrrogacao) ON DELETE RESTRICT,
    FOREIGN KEY (codProcedimento) REFERENCES Procedimento (codProcedimento) ON DELETE RESTRICT
);

CREATE TABLE OPMSolicitadaProrrogacao(
    idProrrogacao    INT          NOT NULL,
    codOPM           INT          NOT NULL,
    quantidade       INT          NOT NULL,
    valorUnidade     NUMERIC(2) NOT NULL,
    PRIMARY KEY (idProrrogacao, codOPM),
    FOREIGN KEY (idProrrogacao) REFERENCES Prorrogacao (idProrrogacao) ON DELETE RESTRICT,
    FOREIGN KEY (codOPM) REFERENCES OPM (codOPM) ON DELETE RESTRICT
);

```

Consultas exemplo

-- Selecionar todos os médicos com CBO-S "Médico imunologista"

```

SELECT m.* FROM Medico m, CBOSMedico r, CBOS c
WHERE r.CPFMedico = m.CPFMedico and
      r.idCBOS = c.idCBOS and
      c.descricao = 'Médico imunologista';

```

-- Selecionar nome e informações do plano de saúde dos beneficiarios que já tiveram alguma Prorrogação

```

SELECT b.nome, p.*
FROM Beneficiario b, PlanoSaude p
WHERE b.nome in(SELECT b.nome
                FROM Beneficiario b, Prorrogacao pr, GuiaInternacao g
                WHERE g.CPFBeneficiario = b.CPFBeneficiario and
                      pr.nroGuia = g.nroGuia
                GROUP BY b.nome
                HAVING count(*) > 0) and
      p.idPlano = b.idPlano;

```

-- Selecionar informações de todos os procedimentos solicitados pela guia de numero 42, incluindo por

```

(SELECT p.*, r.qtdSolicitada, r.qtdAutorizada
FROM Procedimento p, ProcedimentoSolicitado r, GuiaInternacao g
WHERE g.nroGuia = 42 and
      r.nroGuia = g.nroGuia and
      r.codProcedimento = p.codProcedimento)

```

UNION

```

(SELECT p.*, r.qtdSolicitada, r.qtdAutorizada
FROM Procedimento p, ProcedimentoSolicitadoProrrogacao r, GuiaInternacao g, Prorrogacao pr
WHERE g.nroGuia = 42 and
      pr.nroGuia = g.nroGuia and
      r.idProrrogacao = pr.idProrrogacao and
      r.codProcedimento = p.codProcedimento)

```

);





