

Hospital Database (Parte 1)

Grupo 406

up201806538@fe.up.pt	Henrique Manuel Ruivo Pereira
up201801011@fe.up.pt	Iohan Xavier Sardinha Dutra Soares
up201806554@fe.up.pt	Telmo Alexandre Espirito Santo Baptista

3 de Abril de 2020

Projeto BDAD - 2019/20 - MIEIC

Professora das Aulas Laboratorias: Carla Alexandra Teixeira Lopes



Índice

1	Contexto	3
2	UML - Modelo Conceptual	4
3	Modelo Relacional	5
3.1	Classes	5
3.2	Associações	6
3.3	Generalizações	6
4	Análise de DFs e FNs	7
5	Restrições	15

1 Contexto

É pretendido modelar uma base de dados para um hospital com diversos tipos de serviços disponíveis.

Sobre o próprio hospital, interessa guardar informação genérica, como o seu nome, localização completa (morada, código postal, localidade, etc), contato telefónico, bem como se o hospital é público ou privado.

O hospital é constituído por vários departamentos. Cada um destes tem nome, identificador, especializações e entidade responsável pelo departamento.

Staff, pacientes e médicos de família são pessoas, acerca das quais interessa saber o nome, o seu número de identificação único (cartão de cidadão ou equivalente), a sua morada completa (morada propriamente dita, código postal, localidade, etc), a(s) sua(s) nacionalidade(s), o seu contato telefónico, o seu número de beneficiário, o seu sexo e a sua data de nascimento.

A entidade responsável por um departamento é um membro da staff do hospital, podendo este tratar-se de um médico, enfermeiro ou técnico no hospital. Cada membro do staff tem o seu código identificador no hospital.

Especificamente sobre os médicos, é necessário guardar o número do seu consultório no hospital (caso tenha) e a sua especialização (caso tenha). Sobre os enfermeiros, apenas é necessário guardar a sua especialização, caso a tenha. Sobre os técnicos, apenas interessa guardar que serviço oferecem.

Sobre os médicos de família, interessa guardar o centro de saúde ao qual estão associados.

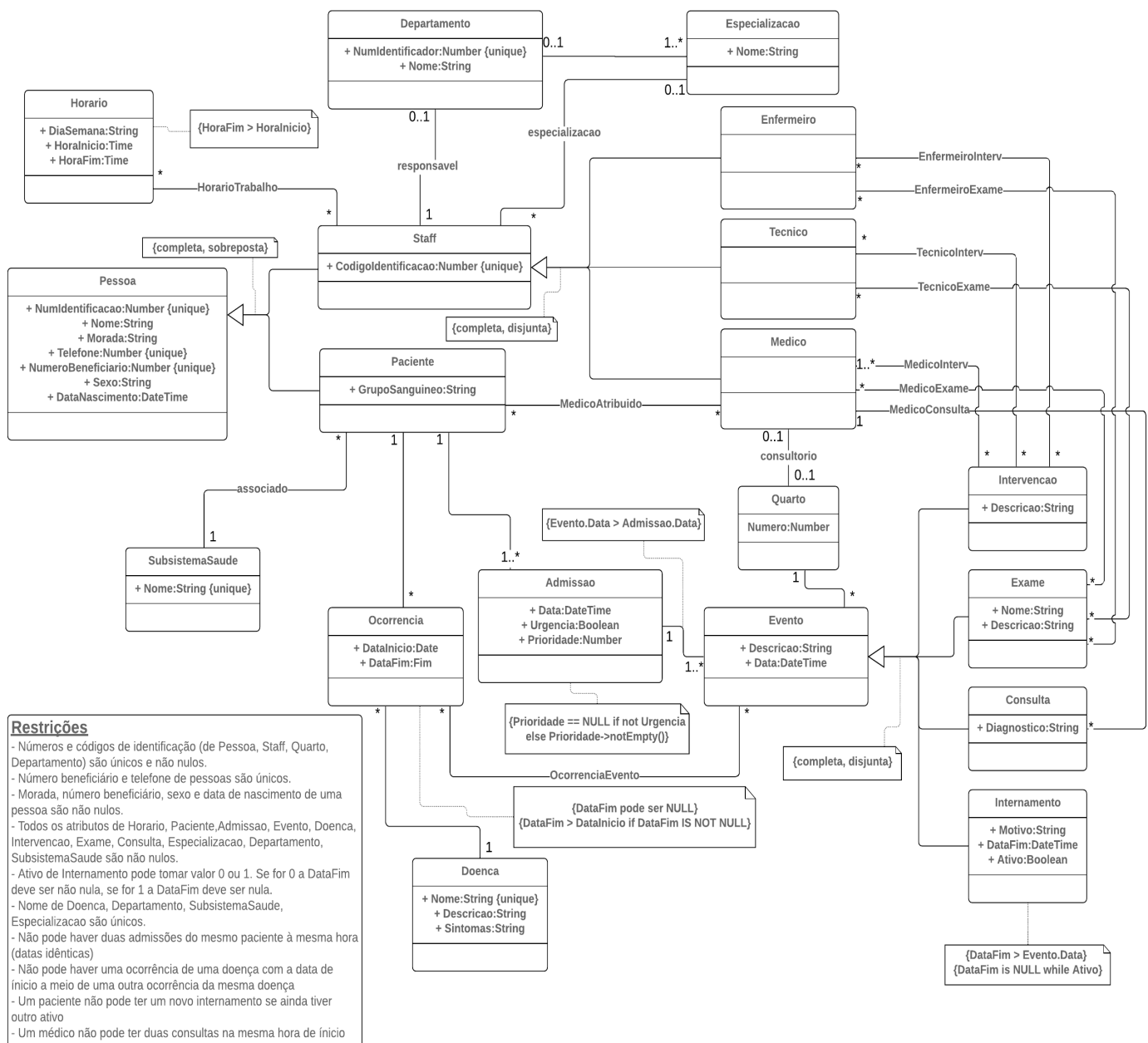
O hospital guarda informação sobre os seus pacientes como o seu grupo sanguíneo, o subsistema de saúde ao qual o paciente está associado, as doenças que o paciente tem, o seu médico de família, os médicos atribuídos naquele hospital, e as suas admissões no hospital.

Sobre cada doença interessa guardar o seu nome, uma descrição da doença e os seus sintomas predominantes.

Uma admissão no hospital tem uma data, se se trata de uma admissão de urgência e, caso seja uma urgência, a prioridade desta. Uma admissão pode desencadear vários tipos de eventos. Cada evento tem uma descrição sobre do que se trata, uma data, e outras informações dependendo do tipo de evento, que pode corresponder a: uma consulta, um exame, uma intervenção ou um internamento.

Numa consulta, interessa guardar o médico que a realizou e o diagnóstico da consulta. Num exame, guarda-se o nome do exame feito e uma descrição deste. Numa intervenção, guarda-se uma descrição da intervenção realizada. Num internamento é necessário guardar o quarto onde o paciente se encontra, o motivo do internamento e a data na qual o paciente recebeu alta (saiu do internamento) caso tal já tenha acontecido.

2 UML - Modelo Conceptual



3 Modelo Relacional

3.1 Classes

Classe Pessoa

- Pessoa(PessoaID, NumIdentificacao, Nome, Morada, Telefone, NumeroBeneficiario, Sexo, DataNascimento)

Classe Staff

- Staff(PessoaID→Pessoa, CodigoIdentificacao, Especializacao→Especializacao)
 - Enfermeiro(StaffID→Staff)
 - Tecnico(StaffID→Staff)
 - Medico(StaffID→Staff, Consultorio→Quarto)
- Especializacao(EspecializacaoID, Nome, Departamento→Departamento)
- Horario(HorarioID, DiaSemana, HoraInicio, HoraFim)
- Departamento(NumIdentificador, Nome, Responsavel→Staff)

Classe Paciente

- Paciente(PessoaID→Pessoa, GrupoSanguineo, SubsistemaSaude→SubsistemaSaude)
- SubsistemaSaude(SubsistemaSaudeID, Nome)

Classe Evento

- Admissao(AdmissaoID, Data, Urgencia, Prioridade, Paciente→Paciente)
- Doenca(DoencaID, Nome, Descricao, Sintomas)
- Ocorrencia(OcorrenciaID, DataInicio, DataFim, Paciente→Paciente, Doenca→Doenca)
- Quarto(Numero)
- Evento(EventoID, Descricao, Data, Admissao→Admissao, Quarto→Quarto)
 - Intervencao(EventoID→Evento, Descricao)
 - Exame(EventoID→Evento, Nome, Descricao)
 - Consulta(EventoID→Evento, Diagnostico, Medico→Medico)
 - Internamento(EventoID→Evento, Motivo, DataFim, Ativo)

3.2 Associações

Associações muitos-para-muitos

Associações relativas à classe Staff:

- HorarioTrabalho(StaffID→Staff, HorarioID→Horario)

Associações relativas à classe Paciente:

- MedicoAtribuido(PacienteID→Paciente, MedicoID→Medico)

Associações relativas à classe Evento:

- EnfermeiroInterv(EnfermeiroID→Enfermeiro, IntervID→Intervencao)
- EnfermeiroExame(EnfermeiroID→Enfermeiro, ExameID→Exame)
- TecnicoInterv(TecnicoID→Tecnico, IntervID→Intervencao)
- TecnicoExame(TecnicoID→Tecnico, ExameID→Exame)
- MedicoInterv(MedicoID→Medico, IntervID→Intervencao)
- MedicoExame(MedicoID→Medico, ExameID→Exame)
- OcorrenciaEvento(OcorrenciaID→Ocorrencia, EventoID→Evento)

Associações muitos-para-um

Para este tipo de associações foi adotado o método de adicionar uma chave estrangeira para a relação "um" na relação "muitos".

Associações um-para-um

Para este tipo de associações foi adicionado uma chave estrangeira à relação que possuirá o menor número de tuplos, exceto nos casos em que essa relação poder ter esse elemento como nulo, nos casos em que ambos podem ser nulos,

3.3 Generalizações

Quanto as generalizações, optou-se por usar o método E/R, criando uma relação para cada classe, e adicionando a chave da super-classe às subclasses, devido às restrições que necessitam ser implementadas nestas classes.

4 Análise de Dependências Funcionais e Formas Normais

Para cada relação será estudado se esta se encontra na forma normal de Boyce-Codd e na 3ª forma normal.

Forma normal de Boyce-Codd (BCNF)

Uma relação R com dependências funcionais FD está na forma normal de Boyce-Codd se:

- Para cada dependência funcional não trivial $\overline{A} \rightarrow \overline{B}$, \overline{A} é uma (super)chave

Terceira forma normal (3NF)

Uma relação R com dependências funcionais FD está na terceira forma normal se:

- Para cada dependência funcional não trivial $\overline{A} \rightarrow \overline{B}$, se se verificar pelo menos uma das seguintes propriedades:
 - \overline{A} é uma (super)chave
 - \overline{B} consiste apenas de atributos primos (atributos que pertençam a uma chave)

Vale notar que se uma relação estiver na forma normal de Boyce-Codd (BCNF) esta também se encontra na terceira forma normal (3NF), já que as 3NF são um superset das BCNF. Se uma relação apenas consistir de dependências funcionais triviais, então a relação também está na forma normal de Boyce-Codd e por consequência na terceira forma normal.

Pessoa

- $PessoaID \rightarrow NumIdentificacao, Nome, Morada, Telefone, NumeroBeneficiario, Sexo, DataNascimento$
- $NumIdentificacao \rightarrow PessoaID, Nome, Morada, Telefone, NumeroBeneficiario, Sexo, DataNascimento$
- $Telefone \rightarrow PessoaID, NumIdentificacao, Nome, Morada, NumeroBeneficiario, Sexo, DataNascimento$
- $NumeroBeneficiario \rightarrow PessoaID, NumIdentificacao, Nome, Morada, Telefone, Sexo, DataNascimento$

Chaves candidatas:

$$\{PessoaID\}, \{NumIdentificacao\}, \{Telefone\}, \{NumeroBeneficiario\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

Staff

- $PessoaID \rightarrow CodigoIdentificacao, Especializacao$
- $CodigoIdentificacao \rightarrow PessoaID, Especializacao$

Chaves candidatas:

$$\{PessoaID\}, \{CodigoIdentificacao\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

Enfermeiro

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

Tecnico

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

Medico

- $StaffID \rightarrow Consultorio$
- $Consultorio \rightarrow StaffID$

Chaves candidatas:

$$\{StaffID\}, \{Consultorio\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

Especializacao

- EspecializacaoID \rightarrow Nome, Departamento
- Nome \rightarrow EspecializacaoID, Departamento

Chaves candidatas:

$$\{EspecializacaoID\}, \{Nome\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Horario

- HorarioID \rightarrow DiaSemana, HoraInicio, HoraFim

Chaves candidatas:

$$\{HorarioID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Departamento

- NumIdentificador \rightarrow Nome, Responsavel
- Responsavel \rightarrow NumIdentificador, Nome

Chaves candidatas:

$$\{NumIdentificador\}, \{Responsavel\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Paciente

- $PessoaID \rightarrow GrupoSanguineo, SistemaSaude$

Chaves candidatas:

$$\{PessoaID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

SistemaSaude

- $SistemaSaudeID \rightarrow Nome$
- $Nome \rightarrow SistemaSaudeID$

Chaves candidatas:

$$\{SistemaSaudeID\}, \{Nome\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

Admissao

- $AdmissaoID \rightarrow Data, Urgencia, Prioridade, Paciente$
- $Data, Paciente \rightarrow AdmissaoID, Urgencia, Prioridade$

Chaves candidatas:

$$\{AdmissaoID\}, \{Data, Paciente\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

Doenca

- DoencaID → Nome, Descricao, Sintomas
- Nome → DoencaID, Descricao, Sintomas

Chaves candidatas:

$$\{DoencaID\}, \{Nome\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Ocorrencia

- OcorrenciaID → DataInicio, DataFim, Paciente, Doenca
- DataInicio, Paciente, Doenca → OcorrenciaID, DataFim

Chaves candidatas:

$$\{OcorrenciaID\}, \{DataInicio, Paciente, Doenca\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Quarto

Não tem dependências funcionais.

Evento

- EventoID → Descricao, Data, Admissao, Quarto

Chaves candidatas:

$$\{EventoID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Intervencao

- EventoID → Descricao

Chaves candidatas:

$$\{EventoID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Exame

- EventoID → Nome, Descricao

Chaves candidatas:

$$\{EventoID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Consulta

- EventoID → Diagnostico, Medico

Chaves candidatas:

$$\{EventoID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira form normal (3NF)

Internamento

- EventoID → Motivo, DataFim, Ativo

Chaves candidatas:

$$\{EventoID\}$$

Como para cada dependência não trivial o conjunto de atributos no lado esquerdo (A) é composto por uma key, então esta relação está na forma normal de Boyce-Codd (BCNF) e por consequência também se encontra na terceira forma normal (3NF)

HorarioTrabalho

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

MedicoAtribuido

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

EnfermeiroInterv

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

EnfermeiroExame

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

TecnicoInterv

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

TecnicoExame

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

MedicoInterv

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

MedicoExame

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

OcorrenciaEvento

Não tem dependências funcionais não triviais, logo, esta relação encontra-se na forma normal de Boyce-Codd (BCNF) e na terceira forma normal (3NF)

5 Restrições

Pessoa

- cada pessoa tem um ID único (**PRIMARY KEY**)
- cada pessoa deve ter um número de identificação, nome, morada, número beneficiário, sexo e data nascimento (**NOT NULL**)
- o número de identificação, número beneficiário e telefone devem ser únicos (**UNIQUE**)

Staff

- cada membro da staff tem um ID único (**PRIMARY KEY**)
- além disso, PessoaID é chave estrangeira para Pessoa (**REFERENCES**)
- não é possível eliminar uma pessoa enquanto houver staff mapeada para essa pessoa, mas pode-se alterar a pessoa que alterará na staff que a mapeia (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- Especializacao é chave estrangeira para Especializacao, e tal será posta a **NULL** caso a Especializacao seja eliminada, em caso de alterações, as alterações serão feitas em toda a staff que mapeia essa especialização (**ON DELETE SET NULL** e **ON UPDATE CASCADE**)

Enfermeiro

- cada enfermeiro tem o seu ID (**StaffID**) que é chave estrangeira para Staff (**REFERENCES**)
- não é possível eliminar um membro da staff enquanto houver um enfermeiro mapeado para esse membro da staff, mas pode-se alterar a staff que alterará também no enfermeiro que a mapeia (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- além disso, este ID deve ser único (**PRIMARY KEY**)

Tecnico

- cada técnico tem o seu ID (**StaffID**) que é chave estrangeira para Staff (**REFERENCES**)
- não é possível eliminar um membro da staff enquanto houver um técnico mapeado para esse membro da staff, mas pode-se alterar a staff que alterará também no técnico que a mapeia (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- além disso, este ID deve ser único (**PRIMARY KEY**)

Medico

- cada médico tem o seu ID (**StaffID**) que é chave estrangeira para Staff (**REFERENCES**)
- não é possível eliminar um membro da staff enquanto houver um médico mapeado para esse membro da staff, mas pode-se alterar a staff que alterará também no médico que a mapeia (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- além disso, este ID deve ser único (**PRIMARY KEY**)
- Consultorio é chave estrangeira para Quarto (**REFERENCES**)
- o consultório deve ser único (**UNIQUE**)
- caso se elimine um quarto que é mapeado como consultório, o valor do consultório é colocado a nulo, em caso de alteração, o consultório também é alterado (**ON DELETE SET NULL** e **ON UPDATE CASCADE**)

Especializacao

- cada especialização deve ter um ID único (**PRIMARY KEY**)
- nome da especialização deve ser único e não nulo (**NOT NULL** e **UNIQUE**)
- Departamento é chave estrangeira para Departamento
- caso se elimine um departamento, será colocado a nulo o valor do departamento em todas as especializações que o mapeiem, em caso de alteração, será alterado também (**ON UPDATE CASCADE** e **ON DELETE SET NULL**)

Horario

- cada horário tem um ID único (**PRIMARY KEY**)
- cada horário deve ter um dia da semana em qual está incidido, uma hora de início e uma hora de fim (**NOT NULL**)
- A hora de início deve ser antes da hora de fim (**CHECK**)

Departamento

- cada departamento tem o seu número identificador único (**PRIMARY KEY**)
- cada departamento tem um nome e um responsável (**NOT NULL**)

- Responsavel é chave estrangeira para staff (**REFERENCES**)
- não é possível remover um membro da staff caso esse esteja mapeado como um Responsavel de um departamento, mas é possível alterá-lo, tais alterações serão aplicadas no departamento que mapeia o membro da staff (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- não pode haver o mesmo membro da staff responsável por dois departamentos (**UNIQUE**)

Paciente

- cada paciente tem um número de identificação único (**PRIMARY KEY**)
- além disso, número de identificação (**PessoaID**) é chave estrangeira para Pessoa (**REFERENCES**)
- não é possível eliminar uma pessoa enquanto houver um paciente mapeada para essa pessoa, mas pode-se alterar a pessoa que alterará no paciente que a mapeia (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- cada paciente tem um grupo sanguineo (**NOT NULL**)
- SistemaSaude é chave estrangeira para SistemaSaude (**REFERENCES**)
- ao eliminar um sistema de saúde colocará em todos os pacientes que o mapeavam o valor **NULL**, em caso de alteração, alterará também no paciente que o mapeia (**ON DELETE SET NULL** e **ON UPDATE CASCADE**)

SubsistemaSaude

- cada sistema de saúde tem um ID único (**PRIMARY KEY**)
- nome do sistema de saúde deve ser único e não nulo (**NOT NULL** e **UNIQUE**)

Admissao

- cada admissão tem um ID único (**PRIMARY KEY**)
- cada admissão tem uma data e um paciente (**NOT NULL**)
- uma admissão pode ser urgência ou não (isto é, tem o valor 1 ou 0, respetivamente), por omissão trata-se de uma admissão não urgente (**NOT NULL** e **CHECK** e **DEFAULT**)
- caso seja uma admissão de urgência a prioridade deve ter um valor maior que 0, em caso não urgente, a prioridade por omissão é 0 (**NOT NULL**, **CHECK** e **DEFAULT**)
- Paciente é chave estrangeira para Paciente (**REFERENCES**)

- não é possível eliminar um paciente caso este esteja mapeado em Admissao, e alterar um paciente altera também em Admissao (**ON UPDATE CASCADE** e **ON DELETE RESTRICT**)
- um paciente não pode ter duas admissões na mesma hora do mesmo dia (datas idênticas) (**UNIQUE**)

Doenca

- cada doença tem um ID único (**PRIMARY KEY**)
- cada doença tem um nome, uma descrição e os seus sintomas (**NOT NULL**)
- o nome da doença deve ser único (**UNIQUE**)

Ocorrencia

- cada ocorrência tem um ID único (**PRIMARY KEY**)
- cada ocorrência tem uma data início, um paciente e uma doença associada à ocorrência (**NOT NULL**)
- Paciente é chave estrangeira para Paciente, Doenca é chave estrangeira para Doenca (**REFERENCES**)
- não é possível eliminar uma doença ou um paciente se houver ocorrências a mapeá-los, mas ao alterar os mesmos, a informação também será alterada nas ocorrências que os mapeiam (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- se a data de fim for não nula, então deve ser depois da data de início (**CHECK**)
- não pode haver início de uma ocorrência de uma doença a meio de uma ocorrência dessa mesma doença (**TRIGGER**) (esta restrição foi implementada, nesta entrega, apenas considerando datas de início iguais)

Quarto

- cada quarto tem um número único (**PRIMARY KEY**)

Evento

- cada evento tem um ID único (**PRIMARY KEY**)

- cada admissão tem uma descrição e uma data, a admissão ao qual está relacionada e o quarto onde foi procedido esse evento (**NOT NULL**)
- Admissao é chave estrangeira para Admissao, Quarto é chave estrangeira para Quarto (**REFERENCES**)
- não é possível eliminar uma admissão nem um quarto enquanto houver ocorrências a mapeá-los, em caso de alteração, estas também são aplicadas nas ocorrências que os mapeiam (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- a data do evento deve ser maior que a data de admissão (**TRIGGER**)

Inervenciaao

- cada intervenção tem um ID (do evento) único (**PRIMARY KEY**)
- cada intervenção tem uma descrição (**NOT NULL**)
- o ID da intervenção (**EventoID**) é chave estrangeira para Evento (**REFERENCES**)
- não é possível eliminar um evento se houver uma intervenção que o mapeia, em caso de alterações, estas também são aplicadas nas intervenções que os mapeiam (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)

Exame

- cada exame deve ter um ID (do evento) único (**PRIMARY KEY**)
- cada exame tem um nome e uma descrição (**NOT NULL**)
- o ID do exame (**EventoID**) é chave estrangeira para Evento (**REFERENCES**)
- não é possível eliminar um evento se houver um exame que o mapeia, em caso de alterações, estas também são aplicadas nos exames que os mapeiam (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)

Consulta

- cada consulta deve ter um ID (do evento) único (**PRIMARY KEY**)
- cada consulta tem um médico associado e um diagnóstico (**NOT NULL**)
- o ID da consulta (**EventoID**) é chave estrangeira para Evento, Medico é chave estrangeira para Medico (**REFERENCES**)

- não é possível eliminar um evento se houver uma consulta que o mapeia, em caso de alterações, estas também são aplicadas nas consulta que os mapeiam. O mesmo se aplica para a chave estrangeira Medico (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- um médico não pode ter duas consultas na mesma hora de início (**TRIGGER**)

Internamento

- cada internamento deve ter um ID (do evento) único (**PRIMARY KEY**)
- cada internamento tem um motivo e se o internamento está ativo ou não (**NOT NULL**)
- o ID do internamento (**EventoID**) é chave estrangeira para Evento (**REFERENCES**)
- não é possível eliminar um evento se houver um internamento que o mapeia, em caso de alterações, estas também são aplicadas nos internamento que os mapeiam (**ON DELETE RESTRICT** e **ON UPDATE CASCADE**)
- Ativo é um valor booleano, tomando o valor de 0 ou 1 (**CHECK**)
- se Ativo estiver a 1 então DataFim deve ser não nula, se tiver a 0 então DataFim deve ser nula (**CHECK**)
- DataFim deve ser maior que a Data do evento mapeado na chave estrangeira EventoID (**TRIGGER**)
- um paciente não pode ter um novo internamento se tiver um ativo (**TRIGGER**)

HorarioTrabalho

- cada intermante deve ter um ID de staff e um ID de horário único (**PRIMARY KEY**)
- StaffID é chave estrangeira para Staff e HorarioID chave estrangeira para Horario (**REFERENCES**)
- ao eliminar uma staff que esteja mapeada num HorarioTrabalho, eliminará também o HorarioTrabalho, em caso de alteração, também alterará em HorarioTrabalho. O mesmo aplica-se à chave estrangeira Horario (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

MedicoAtribuido

- cada intermante deve ter um ID de um paciente e um ID de um médico único (**PRIMARY KEY**)

- PacienteID é chave estrangeira para Paciente e MedicoID chave estrangeira para Medico (**REFERENCES**)
- ao eliminar um paciente que esteja mapeada num MedicoAtribuido, eliminará também o MedicoAtribuido, em caso de alteração, também alterará em MedicoAtribuido. O mesmo aplica-se à chave estrangeira MedicoID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

EnfermeiroInterv

- deve ter uma intervenção ao qual foi atribuído um enfermeiro (**PRIMARY KEY**)
- EnfermeiroID é uma chave estrangeira para Enfermeiro e IntervID é chave estrangeira para Intervencao
- ao eliminar um enfermeiro que esteja mapeado num EnfermeiroInterv, eliminará também o EnfermeiroInterv, em caso de alteração, também alterará em EnfermeiroInterv. O mesmo aplica-se à chave estrangeira IntervID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

EnfermeiroExame

- deve ter um exame ao qual foi atribuído um enfermeiro (**PRIMARY KEY**)
- EnfermeiroID é uma chave estrangeira para Enfermeiro e ExameID é chave estrangeira para Exame
- ao eliminar um enfermeiro que esteja mapeado num EnfermeiroExame, eliminará também o EnfermeiroExame, em caso de alteração, também alterará em EnfermeiroExame. O mesmo aplica-se à chave estrangeira ExameID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

TecnicoInterv

- deve ter uma intervenção ao qual foi atribuído um técnico (**PRIMARY KEY**)
- TecnicoID é uma chave estrangeira para Tecnico e IntervID é chave estrangeira para Intervencao
- ao eliminar um técnico que esteja mapeado num TecnicoInterv, eliminará também o TecnicoInterv, em caso de alteração, também alterará em TecnicoInterv. O mesmo aplica-se à chave estrangeira IntervID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

TecnicoExame

- deve ter um exame ao qual foi atribuído um técnico (**PRIMARY KEY**)
- TecnicoID é uma chave estrangeira para Tecnico e ExameID é chave estrangeira para Exame
- ao eliminar um técnico que esteja mapeado num TecnicoExame, eliminará também o TecnicoExame, em caso de alteração, também alterará em TecnicoExame. O mesmo aplica-se à chave estrangeira ExameID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

MedicoInterv

- deve ter uma intervenção ao qual foi atribuído um médico (**PRIMARY KEY**)
- MedicoID é uma chave estrangeira para Medico e IntervID é chave estrangeira para Intervencao
- ao eliminar um médico que esteja mapeado num MedicoInterv, eliminará também o MedicoInterv, em caso de alteração, também alterará em MedicoInterv. O mesmo aplica-se à chave estrangeira IntervID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

MedicoExame

- deve ter um exame ao qual foi atribuído um médico (**PRIMARY KEY**)
- MedicoID é uma chave estrangeira para Medico e ExameID é chave estrangeira para Exame
- ao eliminar um médico que esteja mapeado num MedicoExame, eliminará também o MedicoExame, em caso de alteração, também alterará em MedicoExame. O mesmo aplica-se à chave estrangeira ExameID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)

OcorrenciaEvento

- deve ter um evento que esteja ligada a uma ocorrência (**PRIMARY KEY**)
- OcorrenciaID é uma chave estrangeira para Ocorrencia e EventoID é chave estrangeira para Evento

- ao eliminar uma ocorrência que esteja mapeado num OcorrenciaEvento, eliminará também o OcorrenciaEvento, em caso de alteração, também alterará em OcorrenciaEvento. O mesmo aplica-se à chave estrangeira EventoID (**ON DELETE CASCADE** e **ON UPDATE CASCADE**)