

Information Systems and Databases



PROJECT ASSIGNMENT

PART I

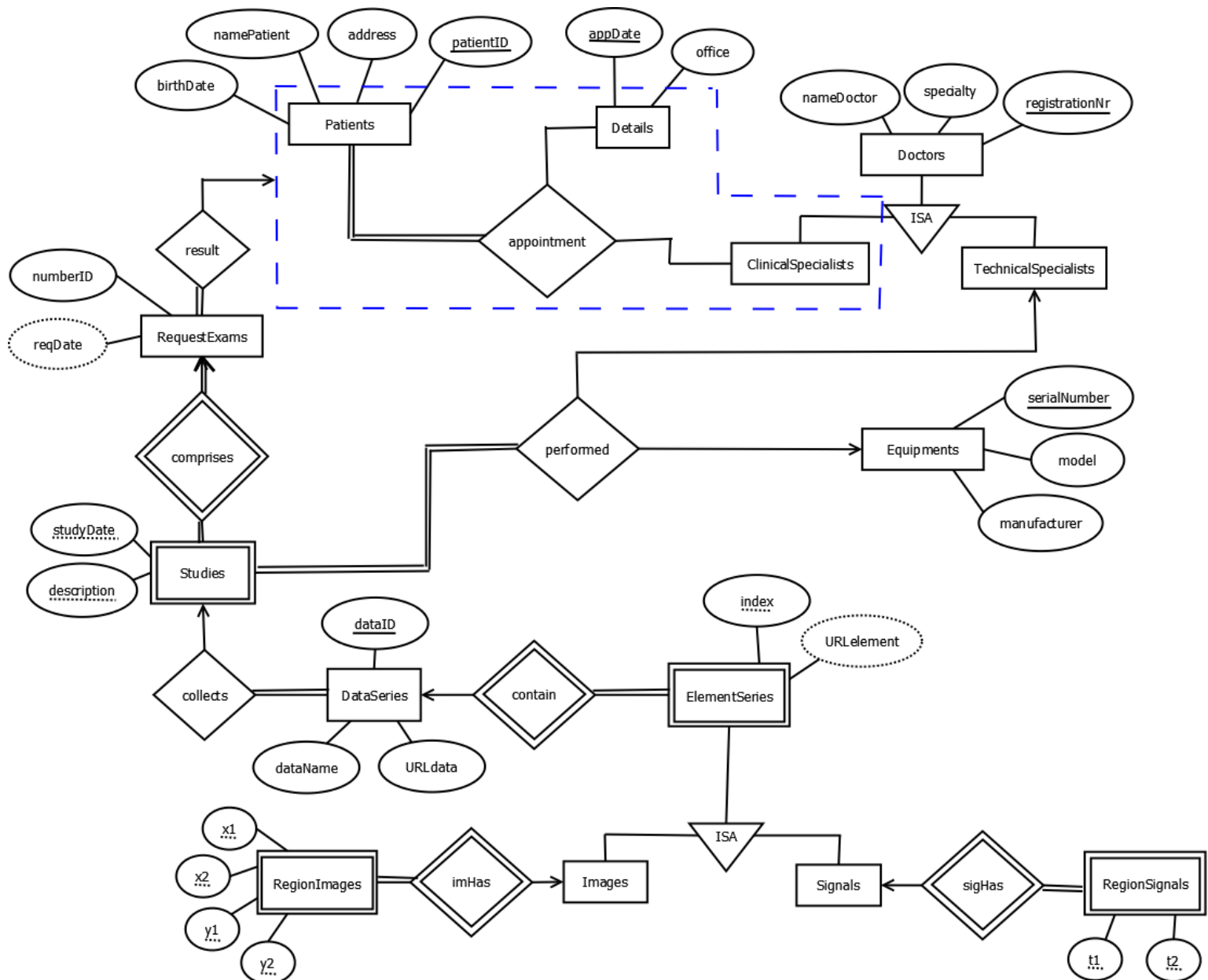
Group 32

75572 - Orlando Bastos Vaz

75637 - Inês de Miranda de Matos Lourenço

75988 - João Pedro Beirão

E-R Diagram



Relational Model

1. Strong Entities

Doctors (registrationNr, nameDoctor, specialty)

ClinicalSpecialists (registrationNr)
registrationNr: FK (Doctors)

TechnicalSpecialists (registrationNr)
registrationNr: FK (Doctors)

Details (office, appDate)

Patients (patientID, namePatient, address, birthDate)

RequestExams (patientID, registrationNr, appDate, numberID)
registrationNr: FK (ClinicalSpecialists)
patientID: FK (Patients)
appDate: FK (Details)

Equipments (serialNumber, manufacturer, model)

DataSeries (studyDate, description, numberID, URLdata, dataName, dataID)
studyDate, description, numberID: FK (Studies)

Images (dataID, index)
index, dataID: FK (ElementSeries)

Signals (dataID, index)
index, dataID: FK (ElementSeries)

2. Weak Entities

Studies (studyDate, description, numberID, registrationNr, serialNumber)
numberID: FK (RequestExams)
registrationNr: FK(TechnicalSpecialists)
serialNumber: FK(Equipments)

ElementSeries (dataID, index)
dataID: FK (DataSeries)

RegionImages (dataID, index, x1, x2, y1, y2)
index, dataID: FK (Images)

RegionSignals (dataID, index, t1, t2)
index, dataID: FK (Signals)

3. Relationships

appointment (patientID, registrationNr, appDate)
registrationNr: FK (ClinicalSpecialists)
appDate: FK (Details)
patientID: FK(Patients)

Discussão

Doctors é decomposto em *ClinicalSpecialists* e *TechnicalSpecialists* porque os dois casos têm tarefas distintas na clínica.

Para descrever um *appointment*, criou-se uma agregação de três entidades, uma vez que a consulta apenas pode acontecer com a presença de um *Patient* e de um *ClinicalSpecialist*. A terceira entidade (*Details*) surge pelo facto de poderem existir várias consultas com o mesmo *Patient* e com o mesmo *ClinicalSpecialist*. Esta entidade, permite distinguir a consulta com uma data e um local. Para além disso, o atributo *appDate* precisa de incluir não só a data como também o horário, uma vez que um *Patient* e um *ClinicalSpecialist* podem ter mais do que uma consulta no mesmo dia. A participação da entidade *Patients* em *appointment* é total porque se considera que um determinado doente é apenas inserido na base de dados da clínica aquando da marcação de uma consulta. Deste modo, todos os *Patients* têm pelo menos um *appointment*. Por outro lado, a participação do *ClinicalSpecialist* na relação é parcial, uma vez que se considerou a possibilidade de existirem médicos que não dão consultas (por exemplo um médico que é coordenador de um determinado serviço da clínica). Nesta agregação, a relação é *many-to-many-to-many* porque um *Patient* pode ter várias consultas com vários *ClinicalSpecialists*; um *ClinicalSpecialist* pode dar várias consultas a vários *Patients*; e um *Patient* e um *ClinicalSpecialist* podem ter várias consultas com diferentes *Details* (data e local).

Como resultado de uma consulta (agregação mencionada acima) podem surgir um ou mais *RequestExams*, no entanto, cada um diz respeito apenas a um *appointment* (facto que justifica a relação ser *one-to-many*). A participação do *RequestExam* na relação é total porque todos resultam de um *appointment*. Contudo, o contrário não se verifica, isto é, uma consulta pode não produzir nenhum *RequestExam*. Nesta entidade, a data do requerimento (*reqDate*) corresponde à data do *appointment* sendo, por isso, um *derived attribute*.

Um *RequestExam* pode incluir indicações para a realização de um ou mais *Studies*, mas cada estudo diz respeito apenas a um *RequestExam* (facto que justifica a relação ser *one-to-many*). A participação do *Studies* na relação é total uma vez que todos são abrangidos por um *RequestExam*. Pelo contrário, um *RequestExam* pode não resultar num *Study* (por exemplo nos casos em que o paciente não dá seguimento ao *RequestExam* sugerido pelo médico). A entidade *Studies* é fraca uma vez que os seus atributos não chegam para identificar totalmente um estudo. Isto acontece porque podem existir vários estudos a ocorrer na mesma data (*studyDate*) e com a mesma *description* e, nestes casos, é necessário utilizar a *primary key* da entidade *RequestExam* (*numberID*) para descrever a entidade *Studies*. De notar que a relação *comprises*, entre *RequestExam* e *Studies*, representa uma relação fraca.

Por outro lado, os *TechnicalSpecialists* são os *Doctors* que participam na realização dos *Studies* indicados pelos *RequestExams*. Estes *Studies* têm que ser realizados por um *TechnicalSpecialist* usando um determinado *Equipment*. Por esta razão, a relação entre os três é uma relação tripla, pois considerou-se que todas as partes são necessárias na realização da tarefa em causa. Por isso, escolheu não se considerar uma agregação. A participação dos *Studies* na relação é total pois não pode existir um estudo sem haver um médico e um determinado equipamento a ele designados. Para além disso, entendeu-se pela

interpretação do enunciado que a um estudo corresponde apenas um médico e apenas um equipamento, pelo que a relação será *many-to-one-to-one*.

Cada *Study* produz uma ou mais *DataSeries* e nenhuma delas pode existir sem ser derivada de um estudo (facto que justifica a participação total de ambas as entidades na relação). Dado que uma *DataSeries* diz respeito apenas a um *Study*, a relação é *one-to-many*. Cada *DataSeries* tem um identificador único (*dataID*), um nome (*dataName*), e um URL (*URLdata*), que identifica a *DataSeries* fora da base de dados.

A entidade *DataSeries* relaciona-se com a *ElementSeries* através da relação *contain*. Esta relação, tal como a *comprises*, conecta uma entidade fraca (*ElementSeries*) a uma entidade forte (*DataSeries*). A *ElementSeries* foi considerada uma entidade fraca porque para ser descrita é sempre necessário invocar a *DataSeries* à qual está associada. Isto significa que cada *ElementSeries* é caracterizado por um *index*, mas diferentes *DataSeries* podem ter *ElementSeries* com os mesmos *indices*. Por outro lado, cada *ElementSeries* é também descrito através de um *URLelement* que é construído com base no *URLdata* (atributo da *DataSeries*) e no *index* (atributo da *ElementSeries*). Por ser um *derived attribute*, optou por não se colocar nas tabelas, visto que deriva da combinação de outros dois atributos.

A *ElementSeries* pode ser do tipo *Image* ou do tipo *Signal*, e em cada elemento podem ser representadas uma ou mais regiões de interesse (*RegionImages* ou *RegionSignals*). As *RegionImages* são descritas por dois pontos e, por este motivo, são necessários dois valores de *x* e dois valores de *y*. As *RegionSignals* são caracterizadas por intervalos de tempo (entre *t1* e *t2*). Em ambas as situações, podem existir uma ou mais regiões de interesse o que faz com que cada elemento possa ter associado a si uma lista de regiões. Desta forma, é necessário considerar que *RegionImage* e *RegionSignal* são entidades fracas, visto ser necessário saber qual o elemento (*Image* ou *Signal*) ao qual pertencem. Assim, cada região necessita dos seus atributos e dos atributos do elemento (*Image* ou *Signal*) ao qual estão associadas.

Nota: No *Relational Model*, as relações fracas foram omitidas assim como os casos em que a relação era *one-to-many* ou *many-to-one*. Neste último caso, a *primary key* do “lado one” é acrescentada à tabela do “lado many”.