Uma imagem com texto

Descrição gerada automaticamenteBase de Dados de uma Faculdade

Entrega 2 – Base de Dados 21/22 | Grupo 408 | 2LEIC04

André Gabriel Correia Vieira - [up202004159@edu.fe.up.pt](mailto:up202004159@edu.fe.up.pt)

Francisca Horta Guimarães - [up202004229@edu.fe.up.pt](mailto:up202004229@edu.fe.up.pt)

Pedro Joaquim Alves Oliveira - [up202004324@edu.fc.up.pt](mailto:up202004324@edu.fc.up.pt)

12/12/2021

Índice

1. **Contexto** ……………………………………………………………………………………………………….. 3

1.1) Introdução …….………….………………………………………………………………………………. 3

1.2) Problema …………….….………………………………………………………………………………… 3

**2. Diagrama UML** ………………………………………………………………………………………………. 4

**3. Diagrama UML Revisto** ………………………………………………………………………………….. 5

**4. Esquema Relacional** …………………………………………………………………………………….... 6

**5. Análise de Dependências Funcionais e Formas Normais** ………………………………. 7

**6. Restrições** ……………………………………………………………………………………………………… 9

1. **Contexto**
   1. Introdução

Pretende-se armazenar informação relativa ao funcionamento de uma faculdade.

Uma faculdade é composta por diferentes órgãos de funcionamento, podendo ser organizada em Departamentos, Docentes, Alunos, Cursos, Disciplinas, etc. Destes, é importante saber informações que os identifiquem para uma melhor gestão dos dados. De seguida, contextualizamos de uma forma mais detalhada o problema.

1.2) Problema

De cada aluno pretendemos armazenar o nome, NIF, morada, número mecanográfico e data de nascimento. Relativamente às unidades curriculares existentes numa faculdade sabemos que pertencem a um curso, têm vários alunos e é relevante armazenar o nome, o código, o número de créditos cujo máximo são 8 créditos e o programa curricular da unidade curricular em questão.

Cada uma destas últimas está encarregue a um docente responsável e tem vários docentes a lecioná-la. Têm também uma ocorrência que as localiza no tempo, estando tal ocorrência definida pelo ano letivo e curricular, o semestre em que ocorre e o número de inscrições cujo máximo são 500 inscrições.

Existem dois tipos de funcionários, os docentes e os não-docentes, caracterizados pelo nome e pelo número. Quanto aos docentes sabemos que pertencem a apenas um departamento e é relevante armazenar a sigla e o seu correio eletrónico. Estes funcionários apresentam diferentes estatutos e o seu salário depende do mesmo, sendo que nunca será menor que o salário mínimo nacional. Quanto aos não-docentes é importante guardar a sua função na faculdade, a duração do seu contrato com um mínimo de 6 meses e o seu salário, também com um mínimo do salário mínimo nacional.

Relativamente aos departamentos, estes são identificados pelo nome, pelo código do departamento, pelos cursos que oferecem e os docentes que lhe pertencem.

Os cursos são identificados por um nome e um código do curso. Apresentam várias disciplinas e estão encarregues a um departamento. Sabemos também que apresentam uma direção constituída obrigatoriamente por um diretor e um subdiretor que pertencem ao departamento responsável pelo curso. Esta direção pode mudar e a data do início da nova direção é registada.

Por fim, existem dois tipos de cursos, as Licenciaturas e os Mestrados. Quanto à Licenciatura guardamos o seu código, o número de vagas cujo máximo são 400 vagas e a sua duração que está compreendida entre 6 a 8 semestres. Já o Mestrado é identificado por um código, um número de vagas cujo máximo são 200 vagas e a sua duração que está compreendida entre 3 a 4 semestres.

1. **Diagrama UML**

**Uma imagem com texto, captura de ecrã, ecrã

Descrição gerada automaticamente**

1. **Diagrama UML Revisto**

A screenshot of a computer

Description automatically generated with low confidence

1. **Esquema Relacional**

Aluno (numMecanografico, nome, nif, morada, dataNasc)

AlunoUc (numMecanografico -> Aluno, codUc -> Uc)

Uc (codUc, nomeUc, numCreditos, programaCurr, codCurso -> Curso)

OcorrenciaUc (idOcorrenciaUc, anoLetivo, anoCurricular, semestre, numInscricoes, codUc -> Uc)

NaoDocente (numFuncionario, nome, salario, durContrato, funcaoFac)

Docente (numFuncionario, nome, sigla, correioEletronico, idEstatuto -> Estatuto, codDep -> Departamento)

Lecionam (codUc -> Uc, numFuncionario -> Docente)

->DocenteResp (codUc -> Uc, numFuncionario -> Docente)

Estatuto (idEstatuto, salario, qualificacao)

Departamento (codDep, nomeDep)

Curso (codCurso, nomeCurso, codDep -> Departamento)

Licenciatura (codCurso -> Curso, codLic, numVagasLic, duracaoLic)

Mestrado (codCurso -> Curso, codMest, numVagasMest, duracaoMest)

DirecaoCurso (idDirecaoCurso, dataMudanca, codCurso -> Curso)

Diretor (numFuncionario -> Docente, idDirecaoCurso -> DirecaoCurso)

SubDiretor (numFuncionario -> Docente, idDirecaoCurso -> DirecaoCurso)

1. **Análise de Dependências Funcionais e Formas Normais**

* Aluno (numMecanografico, nome, nif, morada, dataNasc)

- FDs: numMecanografico -> nome, nif, morada, dataNasc

nif -> numMecanografico, nome, morada, dataNasc

- Formas: 3NF: sim BCNF: sim

* AlunoUc (numMecanografico -> Aluno, codUc -> Uc)

- FDs: ----

- Formas: 3NF: sim BCNF: sim

* Uc (codUc, nomeUc, numCreditos, programaCurr, codCurso -> Curso)

- FDs: codUc -> nomeUc, numCreditos, programaCurr, codCurso

- Formas: 3NF: sim BCNF: sim

* OcorrenciaUc (idOcorrenciaUc, anoLetivo, anoCurricular, semestre, numInscricoes, codUc -> Uc)

- FDs: idOcorrenciaUc -> anoLetivo, anoCurricular, semestre, numIscricoes, codUc

- Formas: 3NF: sim BCNF: sim

* NaoDocente (numFuncionario, nome, salario, durContrato, funcaoFac)

- FDs: numFuncionario -> nome, salario, durContrato, funcaoFac

- Formas: 3NF: sim BCNF: sim

* Docente (numFuncionario, nome, sigla, correioEletronico, idEstatuto -> Estatuto, codDep -> Departamento)

- FDs: numFuncionario -> nome, sigla, correioEletronico, idEstatuto, codDep

correioEletronico -> numFuncionario, nome, sigla, idEstatuto, codDep

sigla -> numFuncionario, nome, correioEletronico, idEstatuto, codDep

- Formas: 3NF: sim BCNF: sim

* Lecionam (codUc -> Uc, numFuncionario -> Docente)

- FDs: ----

- Formas: 3NF: sim BCNF: sim

* DocenteResp (codUc -> Uc, numFuncionario -> Docente)

- FDs: codUc -> numFuncionario

- Formas: 3NF: sim BCNF: sim

* Estatuto (idEstatuto, salario, qualificacao)

- FDs: idEstatuto -> salario, qualificacao

- Formas: 3NF: sim BCNF: sim

* Departamento (codDep, nomeDep)

- FDs: codDep -> nomeDep

- Formas: 3NF: sim BCNF: sim

* Curso (codCurso, nomeCurso, codDep -> Departamento)

- FDs: codCurso -> nomeCurso, codDep

- Formas: 3NF: sim BCNF: sim

* DirecaoCurso (idDirecaoCurso, dataMudanca, codCurso -> Curso)

- FDs: idDirecaoCurso -> dataMudanca, codCurso

- Formas: 3NF: sim BCNF: sim

* Licenciatura (codCurso -> Curso, codLic, numVagasLic, duracaoLic)

- FDs: codCurso -> codLic, numVagasLic, duracaoLic

codLic -> codCurso, numVagasLic, duracaoLic

- Formas: 3NF: sim BCNF: sim

* Mestrado (codCurso -> Curso, codMest, numVagasMest, duracaoMest)

- FDs: codCurso -> codMest, numVagasMest, duracaoMest

codMest -> codCurso, numVagasMest, duracaoMest

- Formas: 3NF: sim BCNF: sim

* Diretor (numFuncionario -> Docente, idDirecaoCurso -> DirecaoCurso)

- FDs: ----

- Formas: 3NF: sim BCNF: sim

* SubDiretor (numFuncionario -> Docente, idDirecaoCurso -> DirecaoCurso)

- FDs: ----

- Formas: 3NF: sim BCNF: sim

Todas as relações da base de dados encontram-se tanto na 3ª Forma Normal (3NF) como na Forma Normal de Boyce-Cod (BCNF).

Sabemos que uma relação está em BCNF se, para todo o A -> B não trivial, A é uma superkey/key.

Sabemos, também, que uma relação está na 3NF se, para todo o A -> B não trivial, A é uma superkey/key ou B é constituído apenas por atributos primos, isto é, atributos que são membros de pelo menos uma chave da relação.

Para todas as relações e todas as FDs da base de dados, a partir da parte esquerda de cada FD (conjunto de atributos A) conseguimos conhecer todos os atributos da relação, o que implica que A é uma (super)key. Assim, justifica-se o facto de se afirmar que todas as relações estão em BCNF e 3NF.

1. **Restrições**

* **NaoDocente**

Todos os Docentes são identificados pelo seu número de funcionário

- numFuncionario PRIMARY KEY

Têm um nome válido

- nome NOT NULL

O seu salário é superior ou igual ao salário mínimo

nacional

- CHECK (salario >= 665)

A duração do seu contrato é igual ou superior a 6 meses

- CHECK (durContrato >= 6)

Todos os Não-Docentes desempenham uma função na faculdade

- funcaoFac NOT NULL

* **Alunos**

Não podem existir dois alunos com o mesmo

número mecanográfico

- numMecanografico PRIMARY KEY

Todos os alunos devem ter uma data de nascimento,

um nome e uma morada que lhes são atribuídos.

- nome NOT NULL

- morada NOT NULL

- dataNasc NOT NULL

- nif NOT NULL

Não existem alunos com o mesmo NIF

- nif UNIQUE

* **Uc**

Cada unidade curricular tem um código que a identifica

- codUc PRIMARY KEY

Todas as unidades curriculares têm um nome e um

curso que lhes é atribuído

* **Docente**

Todos os Docentes são identificados pelo seu ID, este ID corresponde a um número de um funcionário na tabela Funcionários

- numFuncionario PRIMARY KEY

Todos os docentes têm uma sigla, um correioEletronico e um nome

- NOT NULL (nome, sigla, correioEletronico)

Não existem dois docentes com o mesmo correioEletronico e a mesma sigla

- correioEletronico UNIQUE

- sigla UNIQUE

Um Estatuto corresponde ao id de um Docente

- idEstatuto REFERENCES

Estatuto (idEstatuto)

Um departamento é constituído por Docentes

- codDep REFERENCES

Departamento (codDep)

- nomeUc NOT NULL

- codCurso NOT NULL

O seu número de créditos é maior que 0 e menor que 8

- CHECK (numCreditos > 0 AND numCreditos < 8)

O código de um curso corresponde a um curso

- codCurso REFERENCES Curso (codCurso)

* **AlunoUc**

Não existem duas instâncias com o mesmo par

numMecanografico/codUc

- PRIMARY KEY (numMecanografico, codUc)

O número mecanográfico do aluno deve corresponder

a um número mecanográfico da tabela Aluno e a mesma

coisa para o código da unidade curricular na tabela Uc

- numMecanografico REFERENCES

Aluno (numMecanografico)

- codUc REFERENCES Uc (codUc)

* **Lecionam**

- PRIMARY KEY (numFuncionario, codUc)

O número do funcionário deve corresponder

a um numero de funcionário da tabela Docente e a mesma coisa para o código da unidade curricular na tabela Uc

- numFuncionario REFERENCES

Docente (numFuncionario)

- codUc REFERENCES Uc (codUc)

* **OcorrênciaUc**

Não existem duas ocorrências iguais

- idOcorrenciaUc PRIMARY KEY

O ano letivo, o ano curricular e o semestre existem e são

válidos

- anoLetivo NOT NULL

- anoCurricular NOT NULL

- semestre NOT NULL

O semestre deve ser um semestre válido

- semestre CHECK (semestre = 1 OR semestre = 2)

o número de inscrições é maior que 0

- CHECK (numInscricoes > 0 AND numInscricoes < 500)

O código da Uc à qual a corresponde a ocorrência,

corresponde a um código de uma instancia da tabela Uc

* **Departamento**

Cada departamento está identificado pelo seu código

- codDep PRIMARY KEY

Todos os Departamentos têm um nome

- nomeDep NOT NULL

- codUc REFERENCES Uc (codUc)

* **Mestrado**

Todas os Mestrados são identificados pelo seu ID,

este ID corresponde a um código de um curso na

tabela Curso

- codCurso PRIMARY KEY

REFERENCES Curso (codCurso)

Apresentam um número de vagas superior a 0

- CHECK (0<=numVagasMest < 200)

Têm uma duração compreendida entre 3 a 4 semestres

- duracaoMest CHECK (duracaoMest>=3 AND duracaoMest <=4)

Têm um código válido

- codMest UNIQUE

* **DocenteResp**

Não deve haver duas instâncias com o mesmo código

da unidade curricular

- codUc PRIMARY KEY

O número de funcionário deve corresponder a um

número de funcionário na tabela Docente e a

mesma coisa para o código da unidade curricular

na tabela Uc

numFuncionario REFERENCES

Docente (numFuncionario)

- codUc REFERENCES Uc (codUc)

O número do funcionário não pode ser nulo

- numFuncionario NOT NULL

* **Curso**

Cada Curso está identificado pelo seu código

- codCurso PRIMARY KEY

Todos os Cursos têm um nome

- nomeCurso NOT NULL

O código do departamento responsável pelo curso

corresponde a um código de uma instância da tabela Departamento

- codDep REFERENCES Departamento (codDep)

* **Estatuto**

Não há dois estatutos com o mesmo ID

- idEstatuto PRIMARY KEY

O salário é superior ou igual ao salário mínimo nacional

- CHECK (salario >= 665)

Todos os estatutos apresentam uma qualificação

- qualificacao NOT NULL

* **Diretor**

O número do Funcionário deve corresponder

a um número de funcionário da tabela Docente e a mesma coisa para o ID da Direção de Curso na tabela DirecaoCurso

- numFuncionario REFERENCES

Docente (numFuncionario)

- idDirecaoCurso REFERENCES

DirecaoCurso (idDirecaoCurso)

Tanto o número do funcionário como o ID da direção são válidos

- numFuncionario NOT NULL

- idDirecaoCurso NOT NULL

Um nome numa classe Diretor não pode estar numa classe de SubDiretor e vice-versa

* **SubDiretor**

O numero do Funcionário deve corresponder

a um numero de funcionário da tabela Docente e a mesma coisa para o ID da Direção de Curso na tabela DirecaoCurso

- numFuncionario REFERENCES

Docente (numFuncionario)

- idDirecaoCurso REFERENCES

DirecaoCurso (idDirecaoCurso)

Tanto o número do funcionário como o ID da direção são válidos

- numFuncionario NOT NULL

- idDirecaoCurso NOT NULL

Um nome numa classe de SubDiretor não pode estar numa classe Diretor e vice-versa

* **Licenciatura**

Todas as Licenciaturas são identificadas pelo seu ID,

este ID corresponde a um código de um curso na

tabela Curso

- codCurso PRIMARY KEY

REFERENCES Curso (codCurso)

Apresentam um número de vagas superior a 0

- CHECK (0<=numVagasLic <400)

Têm uma duração compreendida entre 6 a 8 semestres

- duracaoLic CHECK (duracaoLic>=6 AND duracaoLic <=8)

Têm um código válido

- codLic UNIQUE

* **DirecaoCurso**

A nova direção de curso é identificada por um ID

- idDirecaoCurso PRIMARY KEY

É registada a data de mudança para a nova direção

- dataMudanca NOT NULL

O código do curso ao qual a direção pertence

corresponde a um código de uma instância da tabela Curso

- codCurso REFERENCES Curso (codCurso)