

21417 - Database Design

Assignment

1 Introduction

The objective of the assignment is to design a database. During the lab sessions you will have to:

1. Propose an Entity-Relationship Model for the proposed task.
2. Translate the ER Model into a Relational Model.
3. Write a report explaining the Entity-Relationship and Relational models, together with the changes they have undergone during the creation process, and explain the different functionalities of the database.

In the last session (10th week, 13-14/03) you will have to give a short presentation (10 minutes) of your database in class. The objective of this presentation is to explain the design decisions you have taken and show some of its functionalities.

1.1 Conditions

1. The assignment should be done in groups of two-three students.
2. The report, together with the code and the presentation, should be delivered **before** the last practicum session (10h30 CET. 13-14/III/2017).
3. The ER Model must follow the guidelines introduced in the theory classes. There are different programs to draw ER Models. You can find a list of them here: [Quora-good-apps-for-ER-Diagrams](#).
4. The Database must be implemented using either MySQL¹ or MariaDB².
5. The report must include (a) a general description of the database, (b) a detailed description of the Entity-Relationship Model, explaining the different design choices, (c) an explanation of the Relational Model implemented and (d) the different *views* on the database, its functionalities and examples of how to access/modify the stored data.
6. The code must work! Otherwise, the assignment will be evaluated as a failing grade.

¹MySQL: <https://www.mysql.com>.

²MariaDB: <https://mariadb.org>

2 Description

The objective of the assignment is to design a database, called *UPFinder*, to manage data related to the UPF. You should propose a database system that will replace the current system.

The database design should allow the creation of different users, spaces and courses and describe how the different elements in the database interact. This section lists a series of *minimum functionalities* of the database, but in your design you are expected to (a) analyse the necessities of the database and (b) propose a design that responds to these requirements.

2.1 Users

Firstly, the database should be able to store the information related to its different users: their name, surname, birth date, e-mail, when they started (studying or working), their class of user (which kind of user it is) and if it is an active user or not. Every user should be identified by a unique id.

There are different kinds of users: students, PDI (temporary and permanent) and PAS. Each kind of user will have access to different parts of the Database. Students are enrolled in one or more degrees, and can access to the set of courses they are following.

Teachers (PDIs) can have assigned one or more courses, but the system must ensure that they do not have two classes at the same time. Each teacher works in a department. They also do research and work on one or more research projects. Some teachers are part of the governing team of their department and/or some are part of the governing team of the university. If a PDI is a member of a governing team, she must have access to the budget information of that department (see 2.5).

Finally, PAS users are workers not directly involved in the courses, such as librarians or cleaning workers. PDIs and PAS are UPF employees, and as such they have assigned a salary, a start date and an end date. It may be that PDIs or PAS workers have been hired multiple times: the database must provide a way to capture the different working periods of the employees.

2.2 Courses

The database has to store all the information related to courses. This include its name, to which degree(s) it corresponds, a list of users (students and teachers), when it starts, when it ends, the hours of the course (which days and at what time), where it takes place (which classroom), and its number of ECTS credits³. Additionally, each student enrolled in a course has a mark. As in the previous subsection, each course should be identified by a unique id. On top of that, students (and teachers) enrolled in a course must have access to the list of material associated with that course.

³See https://en.wikipedia.org/wiki/European_Credit_Transfer_and_Accumulation_System.

The access to the information associated with a course depends on the class of user. For instance, a teacher can add new material to a course but a student can not. Each course has a course coordinator, who is a teacher.

On top of regular courses, there is a bachelor degree thesis (*treball final de grau*), a compulsory project during the last year of a degree in which a student must work individually. It must have a unique id, a name, a supervisor (teacher) and a mark.

2.2.1 Course material

Each course has associated a list of material associated (see 2.2). It consists of a set of documents with different formats (PDFs, textfiles, etc.). Each of them has a unique id, a name, a size (in bytes), a format and must be linked to the course it belongs to. This material can only be added by teachers, but it can be accessed by students.

2.3 Spaces

The database must be able to manage the various spaces of the university.

2.3.1 Classrooms

The database has to store the information of where a course takes place. There are different kinds of classrooms: regular rooms, seminar rooms, computer labs, etc.. They are identified by a unique id, their kind, their capacity (number of seats) and their location (which building, floor and door number). The system must ensure that there are not two courses assigned to the same classroom at the same time.

2.3.2 Offices

Each PDI works in an office. Offices may be shared or not among PDIs. Each office has a unique id, a capacity (the maximum number of PDIs it may contain) and a location (which building, floor and door number).

2.3.3 Library

The University has several libraries. Each library has several working areas. Each working area has a unique id, a capacity, a location (which library, floor and door number) and a timetable of when it is available or not. Students can ask librarians to book one working area for a period no longer than four hours.

2.4 Degrees

A degree stores all the courses that are part of it. A course may belong to one or more degrees. Each degree has a unique id and a degree coordinator (teacher). In order to complete a degree, a student must pass all the courses of that degree and the bachelor degree thesis.

2.5 Budget

Each department has assigned a budget, used to pay the salary of each of its PDIs. The amount assigned to the budget of a department must be higher than the sum of the salaries of each PDI of that department. Additionally, the university has another budget to pay the salary of the PAS employees. The amount assigned to the university budget must be higher than the salaries of all PAS employees in the system. The salaries each UPF worker must perceive are public, and can be accessed here: https://www.upf.edu/personal/_pdf/rh/taules_retrib/tret17gen17.pdf.

2.5.1 Research grants

On top of that, permanent PDIs have assigned a budget, as they can apply and get grants from national or international organisations for research projects. These grants are used to recruit graduate students (master, PhD or post-docs, all as temporary PDIs) for their research projects. If a permanent PDI gets a grant, its amount must be added to (a) her personal budget and to (b) her department of affiliation. Therefore, the database must contain the information of (a) the amount of the salaries of the permanent PDIs for each department and (b) additionally the money permanent PDIs have obtained through competitive grants.

Each grant has a unique id, a name, a research project(s) associated to it (see 2.5.2), an amount of money, a start and end date and a PI (Principal Investigator), the permanent PDI that obtained the grant.

2.5.2 Research projects

Research projects are research enterprises that one or more PDI work on. Each research project has a unique id, a name, a PI (Principal Investigator), a number of researchers (permanent or temporary PDIs) associated to the research project(s) and a list of publications⁴.

2.6 Material

The university gives access to different materials: books, computers, etc.. Each material has a unique id, a name, an indicator if it is available or not, a loan time (start and end) and a waiting list (if the material is taken a student can ask to have access to it once the loan expires).

3 Minimum Requirements

The system should allow:

1. The creation of different users (e.g.: students, teachers, other employees, etc.):

⁴Only if you decide to design the additional functionalities specified in part 4.2.

- (a) The database design should allow different searches on users, such as the list of courses taken by a student, the grades obtained during her studies or the list of courses assigned a teacher.
- 2. Create and manage degrees and courses. This part should at least provide the following functionalities:
 - (a) Add and remove users.
 - (b) Upload materials (e.g.: presentations, documents, assignments, etc.) and administrate the permissions different users have on those documents (who can upload/read what, deadlines, etc.).
 - (c) Assign grades to students (within a course).
- 3. Management of physical spaces and material:
 - (a) Classrooms (regular rooms, seminar rooms, computer labs, etc.) and a way to reserve/assign them.
 - (b) Offices and other spaces.
 - (c) Material loan: books, computers, films, etc..
- 4. Budget management:
 - (a) The database must represent the amount of money associated with (a) each department and (b) the university.
 - (b) Each PDI must have access to its personal budget, which corresponds to the amount of money obtained through grants.
 - (c) The database must store the information related to research grants and the different research projects.

4 Additional Functionalities

1. Extend the database to store information about graduate students (master, PhD candidates and post-docs). Take into account that some of these students may be in charge of a part or the totality of an undergraduate course. Master students are enrolled in a master: you should modify the Database design accordingly in order to model this type of education. Master and PhD candidates may be hired on one research project. Post-docs are hired as temporary PDIs.
2. Teachers and graduate students have an academic track of the different publications their research has lead to. Propose a design that allows the system to keep track of the publications and perform searches according to authors, research fields, research projects, the nature of publications (books, PhD theses, journal articles, conference articles), the name of the publication, their impact factor⁵, the date in which they were published and their number of reads.

⁵See https://en.wikipedia.org/wiki/Impact_factor.