

# **System for Managing Academic Information**

The process of managing information required in an academic environment is a time-consuming activity. In order to produce and manage this information, using a dedicated software application is from far the best solution. Mainly it is about classroom activities engaging students and teachers. The System has to keep tracking information for each student starting with the first undergraduate year and finishing with the last one or with the last master year.

Each faculty may have different master specializations both for the undergraduate and master degrees; a student is enrolled in one of these.

Students are currently inscribed in a year of study (or at most two years). Before the beginning of an academic year, each student has to sign with the faculty a contract of studies stating the academic curriculum agreed. This curriculum contains a mandatory list of disciplines and an optional one.

Establishing the optional list for each student is among the functionalities that the software system has to support. The algorithm for doing this is explained in the next:

- Each teacher (having at least a lecturer degree) proposes a list of maximum 2 optional courses intending to give students in the next academic year.
- The chief of each department approves the final list of optional courses; includes each course in a category (group) for each semester. Usually there are between 2 or 3 groups. The decision is uploaded on the appropriate page of the faculty site.
- Students consult the list of optional courses and specify their preferences in decreasing order for each group.
- Then, the process of assigning optional courses to students may start. In order to be given to students, a course must have at least 20 potential followers. The courses with less potential followers will not be organized. By consequence, for the students choosing these courses the next preferred course will be assigned. After, each assignment need to be validated by the criteria above mentioned.

Once the contracts of studies signed, the catalogs for optional courses can be created.

The next functionality is related to the management of results obtained by each student at disciplines included in his contract of studies. Only the teacher giving a course may upload the results of students at that course.

At the end of each semester students are classified in decrease order of obtained results. This classification together with the funding level is used to decide studying grants.

Each user of the system is identified by its username and password that have to be unique. The username and password are proposed by each user. Previously an automatic username and password is sent to each user.

Members of the administrative staff of the faculty are able to ask and print different documents like:

- The list of students from each group ordered by their professional results
- The list of students from each year ordered by their professional results or complying with some criteria like the average mark included in an interval, a.s.o.

The chief of department may ask other different info like:

- The teacher (discipline) with best or worse results obtained
- The disciplines given by a teacher in a semester or in an academic year.

## The approach

As the SE course proposes an object-oriented approach, it is required that both the domain model and the solution model to be object oriented. There are mandatory requirements regarding the artifacts produced during the software development life cycle like:

- Use Case diagrams and documents, describing the functional model
- Class diagrams for the analysis and design model
- The application will be multilayer structured: User Interface, Business Logic and Database. Object orientation concerns mainly the Business Logic
- By its nature, this is a web application. So, it is recommended that UI provides browser forms specific for each user. The front end functionality will be responsible to gathering data, transmission, and appropriate kind of validations. The database and the business logic will be resident on the server. For the BL layer it is recommended the usage of Java or C# languages.
- A particular attention will be given to the validation process by using assertions (pre & post-conditions, invariants) also to the testing phase, including the techniques of selecting (generating) tests data.
- Also the usage of design patterns in the late design model will be appreciated. The same attitude related to capturing and accessing the rationale of the system.
- Software production is a team activity. So, specifying the different tasks to each team member, analyzing and managing the risks of the development will be considered.