

Projects Manual for Developing Web-Enabled Decision Support Systems

Using Access, VB .NET and ASP .NET

Sandra D. Ekşioğlu

**Industrial and Systems Engineering
Mississippi State University, Mississippi State**

Abhijit A. Pol

**Computer and Information Science and Engineering
University of Florida, Gainesville**

Ravindra K. Ahuja

**Industrial and Systems Engineering
University of Florida, Gainesville &
Innovative Scheduling, Inc. Gainesville**

4 Course Timetabling

Problem Description

Preparing course timetables is an important and difficult task for academic institutions. Given (a) a list of courses offered that semester, (b) a list of faculty members, (c) a list of students registered for the courses, (d) a list of classrooms available, and (e) a list of periods in a week, the task is to prepare a timetable for the current semester. To make this task easier, the timetable from the previous year/semester will be taken as an input. If there are no changes, the same timetable can be used. Possible changes should be identified and it should be ensured that they would not cause any conflicts. For example, if the number of students taking a course changes, this may force a change in the classroom assigned due to room capacity; if the instructor of a course changes, this may also cause a conflict in that instructor's weekly schedule.

The courses are grouped by college and department, and within a department they are grouped into four different levels (i.e., courses offered to freshmen, ..., seniors). In a semester, a group of students follows a set of courses and hence these courses must be assigned to different time periods. The business rules to be followed in building a course timetable are identified as follows:

1. All courses must be assigned to the required number of periods.
2. An instructor cannot be assigned to more than one course in a period.
3. Courses belonging to the same group cannot be assigned to the same time period.
4. A course can only be assigned to an available room in a period if the number of students taking that course is at least 75% and at most 105% of room capacity.
5. The number of courses assigned in a particular period cannot be greater than the number of rooms available.

Database Design

The following are the main entity types of the academic institution database. For each entity type, we provide some of the corresponding attributes. Use this information in order to: (a) Build an Enhanced E-R diagram; (b) Transform the Enhanced E-R diagram to a relational database. Identify the primary key(s) and the foreign key(s) for each relation. Draw the relational integrality constraints; (c) For each of the relations created, indicate its normal form. If the relation is not in the 3NF, decompose it into 3NF relations.

1. *Instructor*: The main attributes are identification number, name, address, name of the department and college s/he is working for, area of specialty, etc.
2. *Course*: The main attributes are course number, name, prerequisites, number of credit hours, group of courses it belongs to, etc.
3. *Student*: The main attributes are identification number, name, address, birthday, gender, department and college s/he is in, etc.
4. *Classroom*: The main attributes are room number, name of the building, capacity, and description of any special feature (for example, media equipment, computers, video, etc.).
5. *Period*: The main attributes are period number, start time, and end time.

6. *Assignment:* This is an associative entity. Its attributes are number of students taking a particular course taught by a particular professor during a particular time period.

Note that the relation between students and courses is a many-to-many relationship. Students get a grade for each course they take.

Access Application Development

The following are some of the queries, forms, and reports that students can create in order to increase the functionality of the database:

Queries:

We already mentioned that the timetable from the previous year/semester is taken as an input in building the current semester's timetable. The schedule would change in the following cases: (a) A professor prefers to teach in a different classroom in a different time period; (b) A professor is offering a new class; (c) The number of students registered for the class is bigger than classroom capacity; etc. The following queries facilitate the process of updating the course timetable.

1. Professors are interested to know their weekly schedule. Create a query that prompts for the professors' name and returns their weekly schedule.
2. Create a query that prompts for students' identification numbers and returns their timetables for the current semester.
3. Create a query that prompts for the name of a course and returns its current schedule.
4. For each course offered in the current semester, present the total number of students registered, the classroom capacity, and the classroom description. This query will help to identify possible changes to the timetable due to the class capacity or equipment (computers, video, etc.) availability.
5. Every semester there are a number of new courses offered that have to be assigned to a period/room.
 - a. List all the courses offered for the first time.
 - b. List the eligible periods/classrooms for an unassigned course. This list should adhere to all the business rules given above.
6. For each time period, list the addresses of the classrooms that are available. This list will be given to the professors. In the case that professors do not like their current schedules, they can re-allocate the class in one of the available period/classroom combinations.
7. Create a query that prompts for the name of a department and course level (freshmen, ..., senior) and returns the timetable of the courses for this particular group. The results from this query are used to check whether the courses belonging to this particular group are assigned to the same class period.