



HACETTEPE UNIVERSITY
DEPARTMENT OF COMPUTER
ENGINEERING

BBM473: DATABASE LABORATORY, SPRING 2023

PHASE 1 - ER DIAGRAM AND RELATIONAL SCHEME

Group ID = 20

Student names:

Mert DOĞRAMACI

Özgün AKYÜZ

Student numbers:

21946055

21827005

Course Management System

Project Definition

Course management systems have become increasingly popular in recent years due to their ability to streamline and simplify the course management process for educational institutions. With the rise of online learning, these systems have become even more essential, as they allow for the management of course registration in a virtual environment. One of the key features of course management systems is their centralized access to course descriptions and grades. This allows for easy tracking of student progress with the homeworks. So, these systems provide a secure and organized system for submitting assignments. This is particularly important in the virtual environment, where there is a greater risk of data breaches and unauthorized access to information. By enabling students to access homework delivery systems exclusively, course management systems ensure that assignments are submitted securely and efficiently. The system administrator can create course descriptions, and responsible persons of each branch can define the information for their branches. This enables a customized and personalized learning experience for students.

To implement a course management system, we have chosen to use React and Spring Boot technologies for the front-end and back-end sides, respectively. React is a popular JavaScript library for building user interfaces, while Spring Boot is a popular framework for building web applications in Java. We have also chosen to use PostgreSQL to handle database operations, allowing us to store and manage courses, student and branch manager accounts, and homeworks. Together, these technologies provide a powerful and scalable platform for managing courses in a virtual environment. Finally, we considered that the designs of all the diagrams, schemes, and relations have evolvable characteristics, which provide adaptability to the occurrence of conditional changes.

Main Functions

- Both students and instructors can see their own information, such as their name, surname, phone numbers, emails, and addresses. They can also update or delete them from the system. Additionally, students can see their student IDs, school enrollment dates, and ECTS that they have in the current semester. You can think of this as a profile page.
- Students can see their transcript that has all their courses, and for each course they can see their grades.
- Students may see their course that they have in the current semester and they can also see their non attendance count for each course from the same panel.
- Both students and instructors may see information about the course, such as section information (related information with the section), title, ECTS information, and type of the course. Also, you may see the sections of the course and its attributes, such as

the semester and year of the section, location and time information for the section (TimeSlot), and information about the instructor who will teach the section.

- Students can enroll in courses or sections during the course registration periods. At the same time, before they finalize their choices, they can delete courses from their list, change their section, or add more courses.
- Some courses have prerequisites. So, students must have taken and passed some courses before they take the other course. When students try to add a lecture or course to their list, the student has to satisfy all the pre-required courses. If there are some courses that don't satisfy the prerequisites, then the student cannot add those courses to his or her list. The system must check those conditions.
- When they try to finalize their choices and if quotas for one or more of their selected courses are already full, students cannot add this section to their list and cannot finalize their choice, so they should drop the course from the list.
- A student cannot take a course if that course's ECTS exceeds the remaining ECTS of the student's.
- After students finalize their choices, their advisor instructor can see their course lists, and after checking if there is an inappropriate situation or not (such as conflict in section time slots, taking an unavailable lecture from any department, etc.), the advisor instructor can approve the course list of the student that they advise. Also, the advisor instructor may see all of the grades of a student from the student's transcript. So, the advisor instructor can decide whether the student is able to graduate or not.
- System admin can create new faculties, departments, courses, sections and update their descriptions.
- Students can see their weekly schedule.

Role of entity sets and relations

- An User has an ID (unique identifier), Name, Surname as attributes.
- Each User has exactly one LoginCredentials and one ContactInfo. These entities are weak entities because they can't exist without the existence of User entity.
- LoginCredentials has an ID (unique identifier), username, and password as attributes.
- ContactInfo has an ID (unique identifier), phone, email, address as attributes.
- There are two subclasses of User: Student and Instructor.
- A Student has StudentID (school number), SchoolEnrollmentDate and SemesterECTS as attributes (in addition to User's attributes). SemesterECTS indicates the student's total ECTS in the current semester. A student must enroll in at least one section (they have their grades and non-attendance count for each enrolled section) and belongs to a department. Each student has only one instructor. Also, students have zero or many homework submissions.

- An Instructor has an ID (unique identifier) in addition to User's attributes. An instructor belongs to a department and can be advisor of zero or many students. An instructor can teach zero or many sections.
- A Faculty has an ID (unique identifier) and name. Each faculty has many departments.
- A Department has an ID (unique identifier) and name. Each department belongs to a department and has many students, courses, and instructors.
- A Course has an ID (unique identifier), title, ECTS, type, and description. Each course belongs to a department. A course may require one or many prereq courses and also a course may be a of one or many courses. Each course has one or many sections, opened for a semester.
- A Section has an ID (unique identifier), semester, year, and classroomInfo. It is a weak entity, because it can't exist without the existence of Course entity. Each section belongs to a course and has one or more TimeSlot, which indicates the schedule of sections. A section has zero or many homeworks as well.
- A Homework has an ID (unique identifier), title, content, issueDate, and dueDate. It is a weak entity, because it can't exist without the existence of Section entity. Each homework belongs to a section and has zero or many submissions.
- A Submission has an ID (unique identifier), marks, and content. It is a weak entity, because it can't exist without the existence of Homework entity. A submission belongs to a student and belongs to a homework.