CS 353-2 Project Proposal

GROUP 20

Members:

Abdullah Mahir Özer

Mert Sezer

Gökberk Aktulay

Fuad Ahmad

Proposal for Course Information and Enrollment System Project

A stars-like course information and enrollment system is going to be implemented. The system is going to have students, teacher assistants (TA), instructors. This system will play crucial role for the students to get information about their courses, they will also be able learn their grades and attendances. Students will be able to enroll for a course and withdraw from a course, they can also upload assignments. Instructors will be able to offer courses, determine course requirements, and grading, and can also grade exams. TAs will be able to grade labs, projects, homeworks. All these operations will be implemented on a database system and students will be able view the related information from the database tables such as grades, attendances and possibly other data that we will decide in the implementation.

We are going to have several requirements, namely functional and non-functional requirements. Functional requirements are as such:

- Displaying grades and attendances - students should be able to see their grades and attendances.
- Sending grades via e-mail system.
- Offer and manage courses for instructors - Instructors should be able to offer courses, upload a syllabus, specify course requirements such as attendance percentage etc.
- Enroll/Withdraw from courses - students should be able to take and drop course they want.
- Uploading assignments - uploading homeworks should be able in system.
- Grading - Instructors should be able to grade exams, and TAs should be able to grade assignments. These grades are SMS'ed to the students(an upgrade over the Bilkent's STARS system which only e-mails some of the grades.)

- Learning grades by SMS service - students should be able to learn their grades by SMS notification system.
- Getting notifications by the SMS service - students should get SMS when there are 3 days left before the deadline of an assignment. This helps with remembering the assignments so there is no last day rush ups, and can also improve the grades of the students since such a countdown-like notification can make the students more interested and adds more importance to the assignment psychologically.

In the following lines non-functional requirements is demonstrated.
- Availability - Users of the system should be able to access system at any time.
- Response Time- Response time should be small enough to reduce time delays that users waste using the system.
- It should have a simple UI.

Limitations (constraints) of the project is demonstrated in the following lines.
- Time- Since we have hard deadlines dates must be met.

Homeworks, projects, and labs are in the assignment table.

Planning of the System Design:

-The student entity keeps the taken courses via the *enroll relation* with the course entity.
-Grades are kept as an attribute of the submit relation which has the participants student and assignment.
-Each course has a section key which has the usual max_quota, available_quota.
-Instructors store the offered courses via the *teaches relation*. The sections that these instructors teach are kept under the course entity which has a relation with the section entity.
-Sections have a relationship with classrooms and time slots. The classroom entity has a room number, the room where the lectures of that course is taught. The time slot defines the time of the lecture.
-Sections store the student entities that are in that section via the *attending relation*.

ENTITIES

INSTRUCTOR
PK: PersonnelID
Name
Surname
Department
Offered_Courses: The courses that this instructor offers at the current semester. For example instructor A offers courses CS202 and CS353.

SMS
PK:SMSKey
PK:SMSCode
Telephone Number: Takes telephone number from the student via the SMS relation.

Grade:

STUDENT
PK: StudentID
Name
Surname
Starting Year
Expected Graduation
Department
Scholarship Status: Can be full scholarship, %50 scholarship, or no scholarship.
Payments: Students can pay for different services and mainly for education purposes.
Taken_Courses_Array: Courses that student took will be held in a array.
Curriculum: All of the department courses that is taken or will be taken.
Current_Courses: Courses that students takes now.
GPA: Average of the grades according to courses credits.
Telephone number: Required to make contact with student in necessary cases.

COURSE
PK: CourseID
Name: Title of the course
Passing_grade: minimal grade to pass the course.
passing_attendance_percent: minimal attendance ratio to not fail the course.

ASSIGNMENT
is_late_assignment: checking if assignment is lately uploaded or not.
deadline: last day and time for uploading assignment.
Assignment_type: type of the assignment which are labs, homeworks, and quizzes.
Title:

EXAM (have is a relationship with assignment)
Title
Time: date and time that exam has been held.

SECTION
max_quota: Maximum quota for the sections that students can enroll.
classroom: place that lessons of the section are going to be held.
time_slot: weekly scheduled time slots of the sections.

TIME_SLOT
PK: time_slot_id { day, start_time, end_time }
Time slot of the section determined by it is day and start, end time.

TA

PK: ID
Name

CLASSROOM
PK: RoomNumber
PK: Building
capacity: capacity of the specific room.

RELATIONS

Submit Relation
Binary Relation, Participants: Student-Assignment
Description: A student can submit an assignment through the system.
Relation attributes: Grade

Enroll Relation
Binary Relation, Participants: Student, Course
A student can enroll for a course or drop a course within the first week of the semester. A student can withdraw from a course within the first eight weeks of the semester.

SMS Relation
Ternary Relation, Participants: Student, SMS, Assignment
Takes the telephone number from the student entity.
Takes the grade from the Assignment entity.
Sends an SMS to the student when the assignment deadline draws close(An upgrade over the current STARS system.) Also sends an SMS when the grade is given for an assignment.
Check_Exam Relation
Binary Relation, Participants: Instructor, Exam
An instructor can grade the exams of a student.

Grading Relation
Ternary Relation, Participants: Student, TA, Assignment
TA can grade the assignments of a student.

Advisor Relation
Binary Relation, Participants: Instructor, Student

Instructors can give advice and even recommendations to students for helping them to regulate their academic decisions.

Pre_req Relation
Some courses have pre-requisite courses.

sec_time_slot Relation
Binary Relation, Participants: Section, time_slot
Each section has its own time slot.

sec_class Relation
Binary Relation, Participants: Section, Classroom
As it was in sec_time_slot, each section has its own place or in other words classroom, where classes has been held.

attending Relation
Binary Relation, Participants: Student, Section
Attribute of the relation: attendance
attendance: for keeping track of the students class attendance.

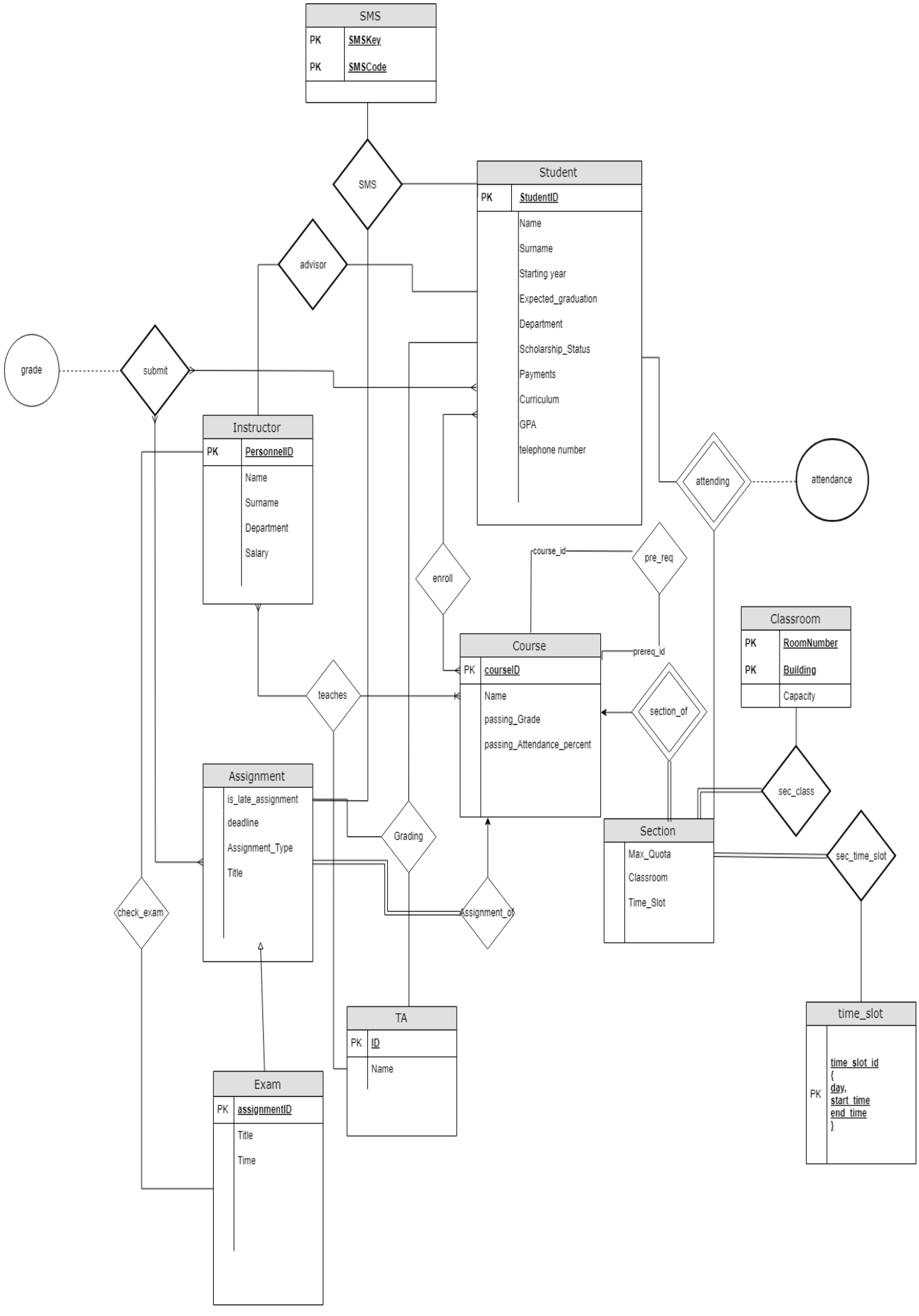
section_of Relation
Binary Relation, Participants: Course, Course
To determine each section of the course.

check_exam Relation
Binary Relation, Participants: Exam, Instructor
Different case from assignments for exams which should be checked by instructors.

teaches Relation
Ternary Relation, Participants: TA, Instructor, Course
Relationship that illustrates TAs and Instructors teaches the course.

assignment_of Relation
Binary RElationship, Participants: Assignment, Course
For identifying assignments of the course.

Link of website which includes report:
https://enrollsystem.blogspot.com/2018/10/cs-353-2-project-proposal-group-20.html


SMS
PK SMSKey
PK SMSCode
SMS
Student
PK StudentID
Name
Surname
Starting year
Expected_graduation
Department
Scholarship_Status
Payments
Curriculum
GPA
telephone number
advisor
grade
submit
Instructor
PK PersonnelID
Name
Surname
Department
Salary
attending
attendance
enroll
course_id
pre_req
prereq_id
Course
PK courseID
Name
passing_Grade
passing_Attendance_percent
Classroom
PK RoomNumber
PK Building
Capacity
teaches
section_of
Assignment
is_late_assignment
deadline
Assignment_Type
Title
Grading
sec_class
Section
Max_Quota
Classroom
Time_Slot
sec_time_slot
check_exam
Assignment_of
TA
PK ID
Name
time_slot
PK time_slot_id
{
day,
start_time
end_time
}
Exam
PK assignmentID
Title
Time