Academic Portal (Project Phase A)
Database Management Systems: CS301

Dr. Viswanath Gunturi

Team Members:

Nitish Goyal 2019CSB1103

Bharat Sharma 2019CSB1081

Rajesh Runiwal 2019CSB1110

The aim of this project is to create a database based implementation of an Academic Portal that provides functionality for students to enroll for courses and view their data; for faculty to offer courses for enrolling and uploading grades; ticket generation and propagation from Student to faculty to batch advisor to Dean of Academics in case Prerequisites of a course are not met. Different users/roles have been created which have different levels of access to the database tables to ensure data security.

Note: Dean has read and write access to all tables corresponding to his year.

Below is the detailed schema:

1. Students:

Contains basic data of students.
Read: All

Students	
Column	Data Type
Student_ID	VARCHAR(30) NOT NULL PRIMARY KEY
Name	VARCHAR(30) NOT NULL
Year	INTEGER NOT NULL
Dept	VARCHAR(6) NOT NULL

2. Student_Registeration_$Student_ID

Contains registration details of a student for every semester. The Dean inserts a new record for every student every semester.
Read: Student corresponding to student_id in name
Write: Student corresponding to student_id in name

Student_Registration_$Student_ID	
Column	Data Type
Semester	INTEGER NOT NULL
YEAR	INTEGER NOT NULL
Cred_limit	FLOAT(3) NOT NULL
Current_credits	FLOAT(3) NOT NULL
Program_core_credit	FLOAT(3) NOT NULL
Program_Elective_Credit	FLOAT(3) NOT NULL

Science_Core_Credit	FLOAT(3) NOT NULL
Open_Elective_Credit	FLOAT(3) NOT NULL
CGPA	FLOAT(3) NOT NULL

3. Student_Courses_$Student_ID

Contains the offering_ID of all the courses successfully enrolled in by the student corresponding to student_id
Read: Student corresponding to student_id in name
Write: Student corresponding to student_id in name

Student_Courses_$Student_ID	
Column	Data Type
Offering ID	VARCHAR(30) NOT NULL

4. Student_Tickets_$Student_ID

Contains details of all tickets generated by the student. A ticket is generated automatically if the student registers for a course whose prerequisites he or she does not meet.
Read: Student corresponding to student_id in name
Write: Student corresponding to student_id in name

Student_Tickets_$Student	
Column	Data Type
Ticket_ID	VARCHAR(10) NOT NULL PRIMARY KEY
Offering_ID	VARCHAR(10) NOT NULL
Status	VARCHAR(10) NOT NULL
Request	VARCHAR(1000) NOT NULL

5. Student_grades_$student_id

Contains grade obtained in all courses enrolled till now
Read: Student corresponding to student_id in name

Student_grades_$student_id	
Column	Data Type
Offering_id	VARCHAR (100) NOT NULL

Grade	INTEGER

6. Course

This is the course catalog and contains fundamental information about the course\
Read: All

Course	
Column	Data Type
Course_ID	VARCHAR(10) NOT NULL PRIMARY KEY
Title	VARCHAR(30) NOT NULL
Dept	VARCHAR(6) NOT NULL
L	INTEGER
T	INTEGER
P	INTEGER
S	INTEGER
Credit	FLOAT(2)

7. Course_offering:

Contains details about a specific course offering including semester, year, instructor etc
Read: All
Write: Faculty

Course_Offering	
Column	Data Type
Offering_ID	VARCHAR(100) NOT NULL PRIMARY KEY
Course_ID	VARCHAR(10) NOT NULL
INS_ID	VARCHAR(30) NOT NULL
Slot_ID	VARCHAR(10) NOT NULL
Semester	INTEGER NOT NULL
YEAR	INTEGER NOT NULL
Status	VARCHAR(50) NOT NULL

8. Offering_enrollment_$Offering_id

Student_id and grade obtained by students enrolled in the course offering
Read: Faculty teaching
Write: Faculty teaching

Offering_Enrollment_$Offering_Id	
Column	Data Type
Student_ID	VARCHAR(30) NOT NULL
Grade	INTEGER

9. Faculty

Basic information about the faculty
Read: All

Faculty	
Column	Data Type
INS_ID	VARCHAR(30) NOT NULL PRIMARY KEY
Name	VARCHAR(30) NOT NULL
Dept	VARCHAR(6) NOT NULL

10. Batch_adv

List of batch_adv by department and year
Read: All

Batch_Adv	
Column	Data Type
INS_ID	VARCHAR(30) NOT NULL
Dept	VARCHAR(6) NOT NULL
YEAR	INTEGER NOT NULL

11. Dean_acad

List of dean_acad by department and year
Read: All

Dean_Acad	
Column	Data Type
INS_ID	VARCHAR(30) NOT NULL
Dept	VARCHAR(6) NOT NULL
YEAR	INTEGER NOT NULL

12. Program_courses

Information about curriculum courses (core courses and electives) by department and year
Read: All

Program_Courses	
Column	Data Type
Course_ID	VARCHAR(10) NOT NULL
Dept	VARCHAR(10) NOT NULL
YEAR	INTEGER NOT NULL
Course_Type	VARCHAR(20)

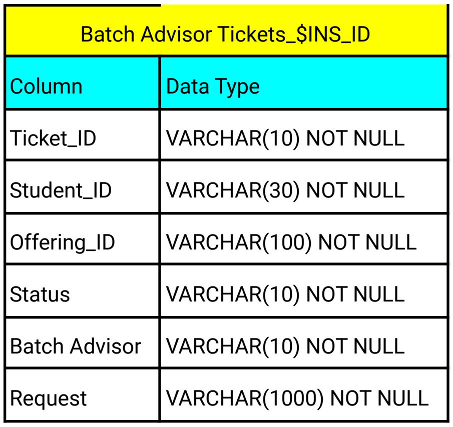
13. Faculty_tickets_$ins_id

Contains details of all tickets received by the instructor.
Read: Faculty corresponding to ins_id in name

Faculty_Tickets_$INS_ID	
Column	Data Type
Ticket_ID	VARCHAR(10) NOT NULL
Student_ID	VARCHAR(30) NOT NULL
Offering_ID	VARCHAR(100) NOT NULL
Status	VARCHAR(10) NOT NULL
Faculty Approval	VARCHAR(10) NOT NULL
Request	VARCHAR(1000) NOT NULL

14. Adv_tickets_$ins_id

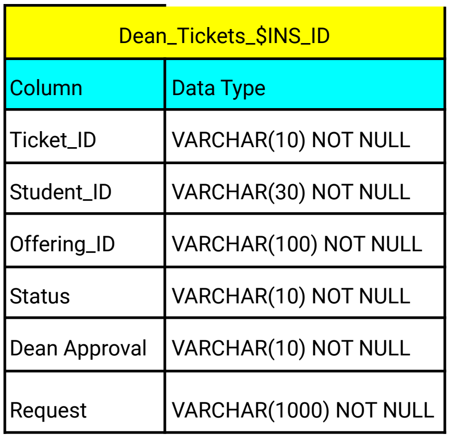
Contains details of all tickets received by the batch advisor
Read: Faculty corresponding to ins_id in name



Batch Advisor Tickets_$INS_ID	
Column	Data Type
Ticket_ID	VARCHAR(10) NOT NULL
Student_ID	VARCHAR(30) NOT NULL
Offering_ID	VARCHAR(100) NOT NULL
Status	VARCHAR(10) NOT NULL
Batch Advisor	VARCHAR(10) NOT NULL
Request	VARCHAR(1000) NOT NULL

15. Dean_tickets_$ins_id

Contains details of all tickets received by the dean.
Read: Faculty corresponding to ins_id in name



Dean_Tickets_$INS_ID	
Column	Data Type
Ticket_ID	VARCHAR(10) NOT NULL
Student_ID	VARCHAR(30) NOT NULL
Offering_ID	VARCHAR(100) NOT NULL
Status	VARCHAR(10) NOT NULL
Dean Approval	VARCHAR(10) NOT NULL
Request	VARCHAR(1000) NOT NULL

16. Prerequisites

Current prerequisites for all courses
Read: All

Prerequisites	
Column	Data Type
Applied_Course_ID	VARCHAR(10) NOT NULL
Pre_Course_ID	VARCHAR(10) NOT NULL

17. Slots

Time slots of time table
Read: All

Slots	
Column	Data Type
Slot_ID	VARCHAR(10) NOT NULL PRIMARY KEY
Start_time	time
End_time	time

Triggers and Functions

1. Function update_grades(offering_id varchar (40), file varchar(1000)):
   Executed by dean/faculty as a query to update grades for given offering_id. Grades are imported from a given file(complete path to be entered). Once the grades are imported, the CG and Credits of all the enrolled students is updated and the Status of the course in course offering is changed. Final grades are also inserted into the student_grades table

2. Function is_the_student_ready_to_graduate(student_id1 varchar (40)):
   Executed by the dean as a query to check if the student is ready to graduate. Returns 'YES' OR 'NO'

3. Function create_st_tables():
   Executed by trigger on create_student_tables table students when a new record is inserted into the students table. Creates registration, courses and tickets table for the new student. User is Created using student_id as username and default password 'iitropar'.

4. Function check_eligibility():
   Executed by trigger create_ticket_trigger_$student_id on table Student_Courses for a

student and runs when the student tries to enroll into a new course offering. Checks if the Student meets the Course enrollment criteria like Prerequisites or Cg Criteria Set by the Instructor. It also checks if there are any slot clashes. If the Student is eligible then he/she is enrolled in the course, and then the credit limit of the respective student is decreased, otherwise a ticket is generated for the Enrollment request to be approved by the Faculty, Batch Advisor and Dean.

5. Function create_offtable():
   Executed by trigger create_offering_enrollment_table On table Course_offering when a new entry is inserted in the table. Creates a separate Offering_Enrollment table for that Course Offering.

6. Function create_advtable():
   Executed by trigger create_adv_ticket_table On table batch_adv when a new entry is inserted in the table. Creates the Ticket Table for that Batch Advisor. User is Created using adv_ins_id as username and default password ‘iitropar’.

7. Function create_dtable():
   Executed by trigger create_dean_ticket_table On table dean_acad when a new entry is inserted in the table. Creates the Ticket Table for that Dean Academic Officer. User is Created using dean_ ins_id as username and default password ‘iitropar’.

8. Function create_table():
   Executed by trigger create_faculty_ticket_table On table faculty when a new entry is inserted in the table. Creates the Ticket Table for that Instructor. User is Created using ins_id as username and default password ‘iitropar’.

9. Function update_faculty_ticket():
   Executed by trigger update_fticket_trigger_$ins_id On table faculty_ticket when the concerned faculty updates the status of a ticket in his Ticket Table and propagates the ticket to concerned advisor

10. Function update_adv_ticket():
    Executed by trigger update_aticket_trigger_$ins_id On table adv_ticket when the concerned adv updates the status of a ticket in his Ticket Table and propagates the ticket to concerned dean

11. Function update_dean_ticket():
    Executed by trigger update_dticket_trigger_$ins_id On table dean_tickets when the concerned dean updates the status of a ticket in his Ticket Table. The Final decision by the dean is reflected in all ticket tables corresponding to that ticket. If the dean approves the Request then the student is enrolled in the concerned course and the credit limit of the concerned student is decreased.

12. Function gradesheet(student_id1 varchar (40)):
    Executed by dean/concerned student as a query to get gradesheet for given student_id.

Roles

1. Dean_acad: Inserts new students, faculties, batch advisors into respective tables. Inserting and updating courses into the course catalog, program courses, prerequisites, slots Has write access to all the tables which correspond to his year. Also handles tickets.

2. Faculty: Updates and inserts course offerings and prerequisites. Also has access to offering_enrollments for his offerings and can add grades to it. Also handles tickets.

3. Batch_adv: Handles tickets.

4. Students: Has read access to respective student_registration, student_tickets, student_grades table. Can write to the respective student_courses table to enroll/generate tickets. Can also generate his/her own grade sheet/transcript.