System Requirements Document

Project Overview

This system is designed to manage the academic life cycle of students, courses, instructors, and departments within an educational institution. The system aims to handle student registration, course enrollments, grading, and the generation of student transcripts. It will also provide detailed reporting on enrollment statistics and track exam results.

1. Functional Requirements

1.1 Student Management

The system must allow administrators and authorized users to manage student records efficiently. The main functions are:

* Add, Update, and Delete Student Records:
* Users must be able to add new students, update their details, and delete students when required.
* View Student Details (including GPA):
* Each student’s academic profile, including GPA, grades, and course history, should be available.
* Assign Students to Advisors:
* Students should be automatically assigned to advisors based on their department. This can be done manually or through an automated system, such as a trigger or stored procedure.

1.2 Course Management

The system must manage courses offered by the institution, including course schedules, instructors, and classrooms:

* Add New Courses and Assign Instructors:
* New courses must be added, with the option to assign instructors.
* Update Course Schedules and Assign Classrooms:
* The system should allow course schedules to be updated, along with the assignment of classrooms.
* View Course Enrollment Numbers and Cancel Under-enrolled Courses:
* The system should track the number of students enrolled in each course. If a course is under-enrolled, it may be canceled or flagged for review.

1.3 Enrollment Management

This functionality will enable students to enroll in courses and provide a history of enrollments:

* Online Course Enrollment:
* Students can enroll in available courses through the online portal.
* Maintain History of Course Enrollments:
* The system will keep track of all enrollments by students, including details like the semester, course, and date of enrollment.
* Generate Reports on Enrollment Statistics:
* The system should allow administrators to generate reports that show enrollment statistics by course and department.

1.4 Grade Management

Grade management will allow instructors to record grades and track student performance:

* Record Grades for Students in Each Course:
* Instructors can record grades for students based on assessments (e.g., exams, assignments).
* Calculate GPAs for Students:
* The system will automatically calculate GPAs for students based on their grades for each course.
* Generate Student Transcripts:
* The system should generate official student transcripts, showing course grades and GPA.

1.5 Examination Management

The system should allow exams to be scheduled and results tracked:

* Schedule Exams for Courses:
* Instructors or administrators can schedule exams for each course.
* Track Exam Results and Maintain History:
* Exam results for students will be recorded and tracked for future reference.

2. Non-Functional Requirements

2.1 Data Integrity

* Ensure that students cannot enroll in a course without meeting prerequisites. If a student tries to enroll in a course without having completed the necessary prerequisites, the system should block the action.

2.2 Performance

* The system should be able to calculate GPAs and generate transcripts in less than **2 seconds**. The system must be optimized for quick data retrieval and processing.

2.3 Security

* **Authorized Users Only**: Only authorized users (e.g., instructors, registrars) will have the ability to update grades. User roles will be assigned based on their responsibilities.
* **Role-based Access Control (RBAC):** The system will define specific roles (e.g., student, instructor, registrar, admin) and assign permissions accordingly.

2.4 Auditability

* The system should maintain an audit log that tracks grade changes. This log will store information about who made the change, when, and the details of the change.

2.5 Concurrency

* The system must handle simultaneous course registrations by multiple students without any data inconsistencies or performance degradation. Proper locking mechanisms will be used to ensure data integrity during concurrent updates.

3. System Functionalities

3.1 Student Functionality

* Student Registration:
* A student can register into the system using an online form that collects essential details such as name, address, date of birth, department, etc.
* Enrollment in Courses:
* Students can view the available courses for their department, prerequisites, and instructor details, and then register online.

3.2 Instructor Functionality

* Grade Entry:
  + Instructors can record grades for students in each course, either manually or via automated methods.
* View Student Performance:
* Instructors can view student grades, performance history, and generate reports of class performance.

3.3 Administrator Functionality

* Manage Student and Course Records:
* Admins have full access to the system and can add, update, or delete students and courses.
* Course and Department Reports:
* Admins can generate reports for enrollment statistics and course performance.

3.4 Advisor Functionality

* Assign Students to Courses:
* Advisors can review students' academic progress and recommend courses that students should enroll in based on their department.
* View Advisees' Academic Performance:
* Advisors can see the academic performance of students assigned to them.

3.5 System Automation

* Triggers and Stored Procedures:
* **Triggers** will be used to automatically assign students to their respective advisors when they enroll in a department.
* **Stored procedures** will automate batch enrollments and GPA calculations.

3.6 Reporting

* Enrollment Reports:
* Reports that show the enrollment statistics for each course and department. These reports will display the number of enrolled students, available seats, and under-enrolled courses.
* Grade Reports and Transcripts:
* Reports that detail students' performance in courses, including individual grades and final GPAs. Transcripts will be generated for students, showing their entire academic history.

4. Advanced Features

4.1 GPA Calculation Simplification

* Views will be created to simplify complex GPA calculations based on different grading systems used by the institution.

4.2 Batch Enrollment and Automated GPA Calculation

* Stored procedures will handle the bulk enrollment of students into courses and automate the calculation of GPAs based on grading criteria.

4.3 Advisor Assignment via Triggers

* A trigger will automatically assign students to an advisor when they enroll in a department. This process will ensure that students are always paired with an appropriate advisor based on their department.

5. Data Models

5.1 Entity Relationship Model (ERD)

* The system includes multiple entities such as **Student**, **Course**, **Enrollment**, **Grade**, **Instructor**, and **Department**. Each entity will have attributes specific to the task it manages, and relationships between entities (such as enrollment and grade tracking) will be clearly defined.

5.2 Tables & Attributes

Each table will have attributes that store necessary data for each entity, such as:

6. Conclusion

This system will streamline the processes of student enrollment, grade management, course scheduling, and the generation of transcripts. It ensures data integrity, improves security with role-based access, and allows administrators to easily track enrollment and grade statistics. By automating key tasks like GPA calculations and advisor assignments, this system will significantly reduce manual work and enhance efficiency.