

# University Management System

A Secure, Automated,  
and Transaction-Based  
SQL Server Architecture

Aseel Menhem 6651  
Maryline Kraram 6599

Database Design & Implementation Project

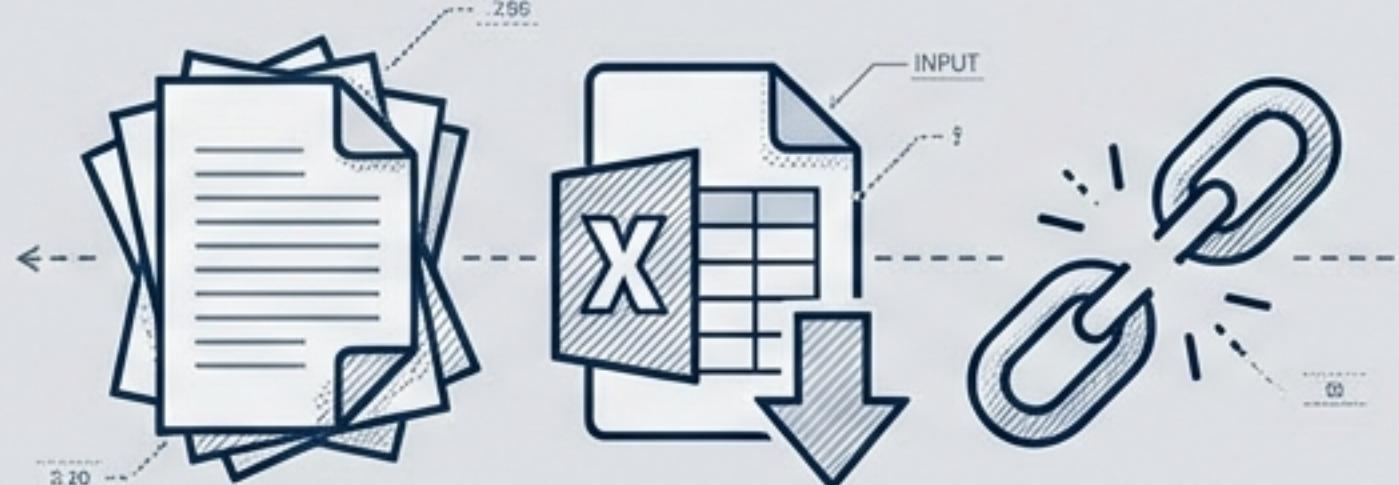


Automation (Triggers & Procedures)  
Security (RBAC & Least Privilege)  
Data Integrity (ACID Transactions)  
Financial Control (Real-time Balance Checks)

# From Fragmentation to Centralized Control



## The Problem: Manual Systems



- Data Redundancy & Inconsistency
- Phantom Enrollments (Overcapacity)
- Disconnected Financials (Debt ignored)
- Security Vulnerabilities

## The Solution: Integrated Database



- ACID-Compliant Data Consistency
- Automated Grade & GPA Processing
- Financial Control (Enrollment blocked on debt)
- Centralized Business Logic



# System Specifications & Requirements



## Functional Requirements



- **User Management:** Registration tracks for Students, Instructors, and Admin.
- **Academic Operations:** Course creation, Section scheduling, Grade locking.
- **Enrollment Logic:** Real-time capacity checks and prerequisite validation.
- **Finance:** Payment processing and Salary generation based on hourly rates.

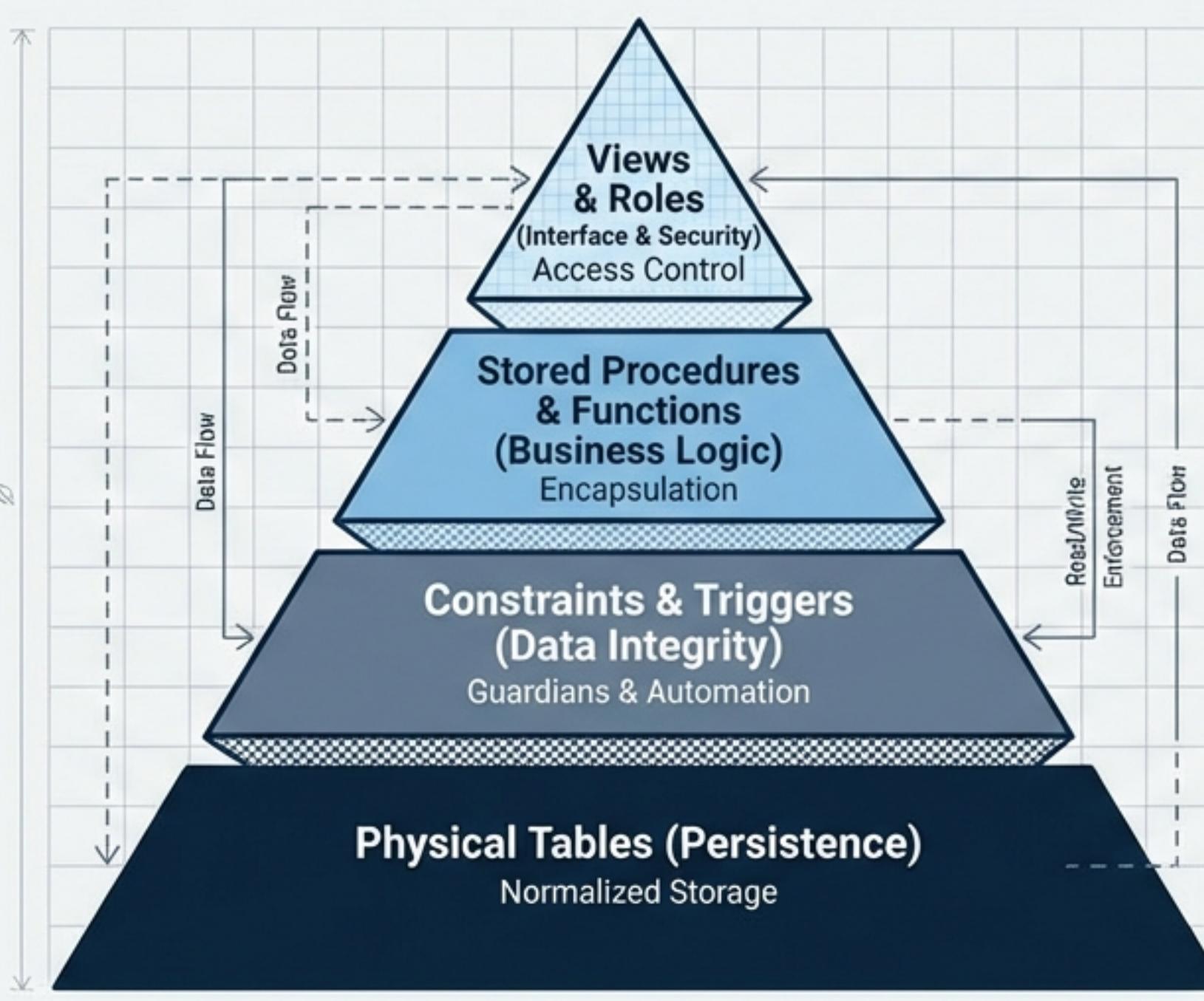
## Non-Functional Requirements



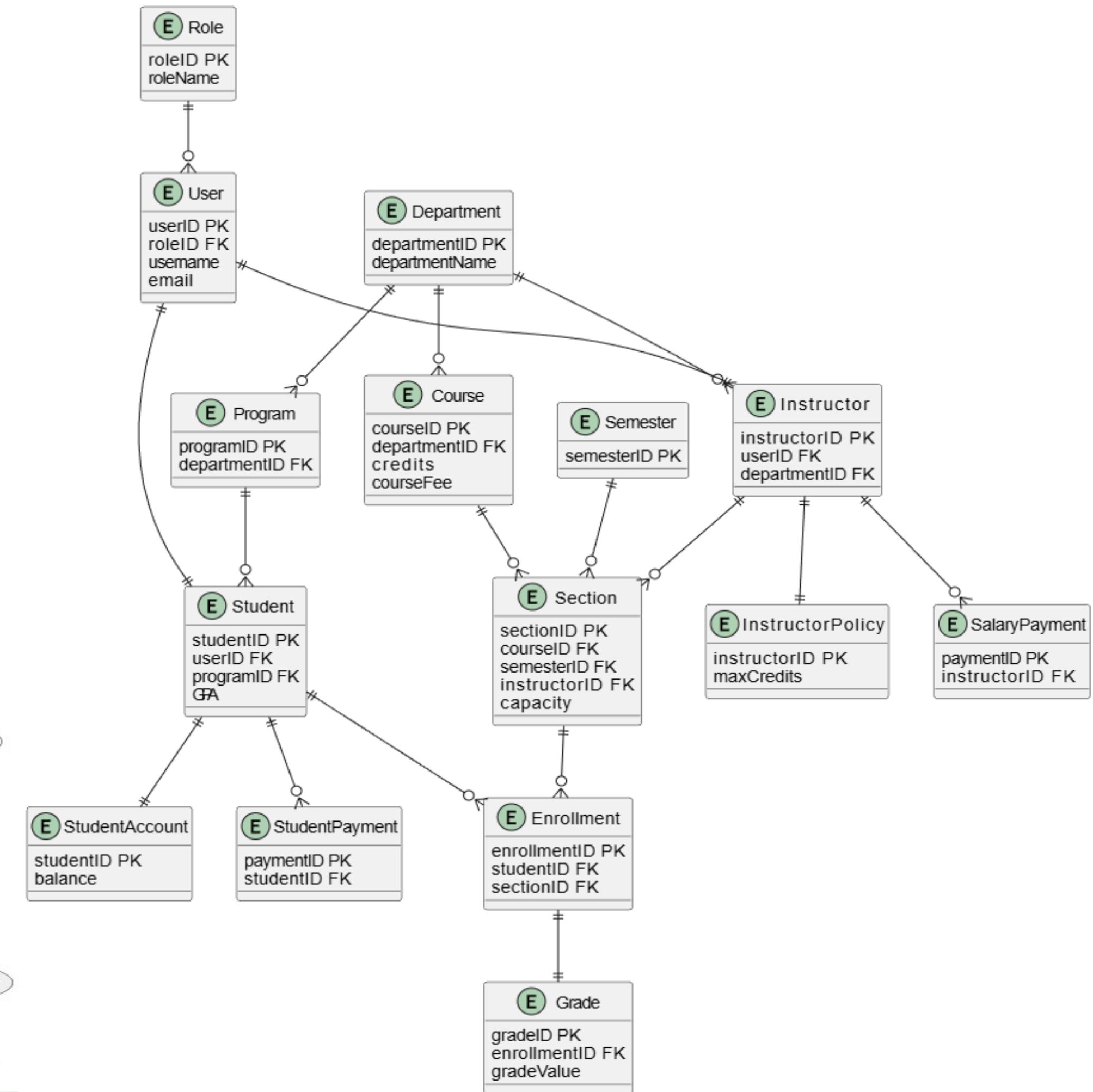
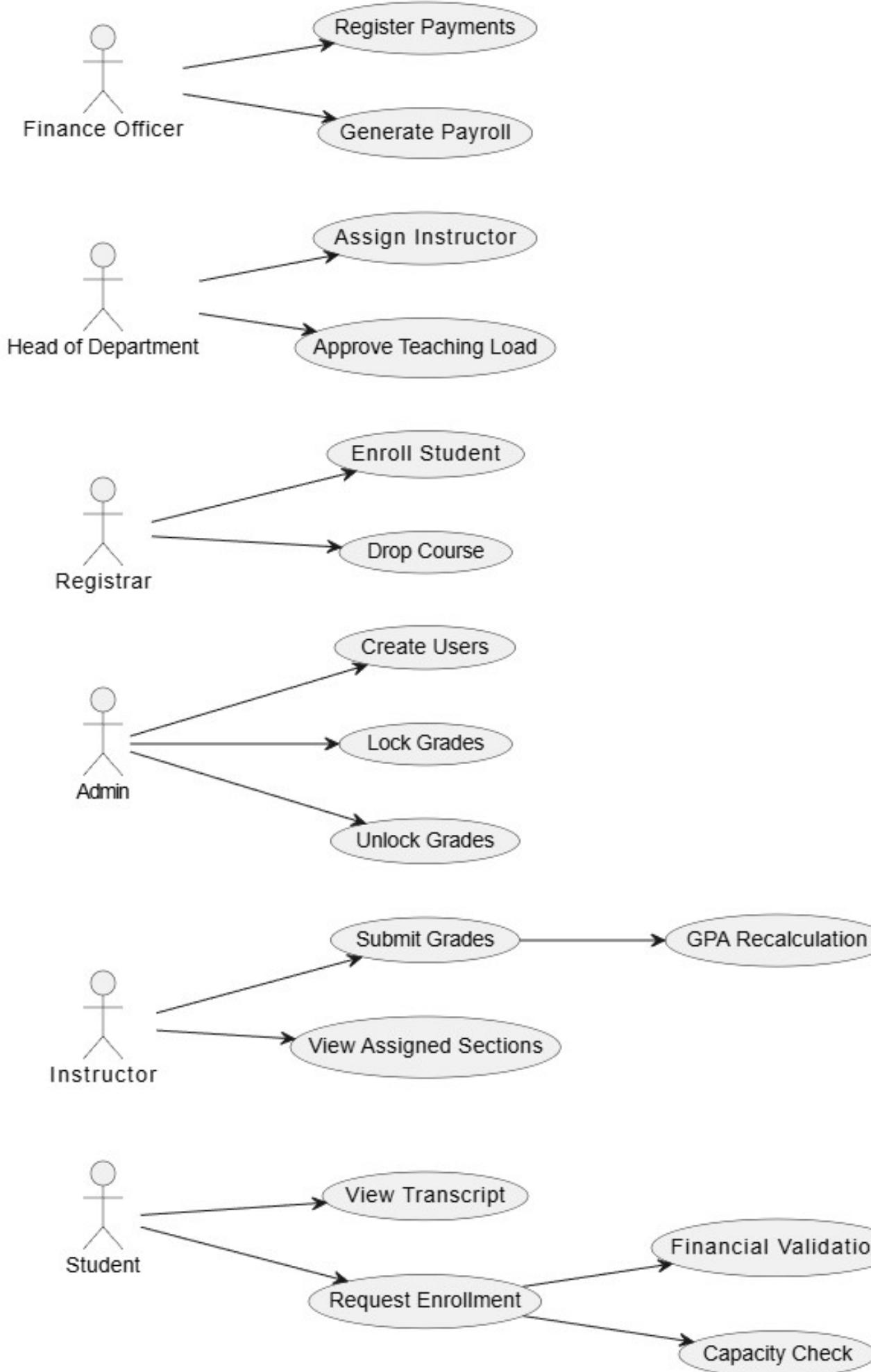
- **Security:** Granular permission handling via SQL Roles.
- **Reliability:** TRY/CATCH error handling with transaction rollbacks.
- **Scalability:** Normalized schema supports department/program expansion.
- **Integrity:** Strict Foreign Key and CHECK constraints (e.g., GPA range, Credits).



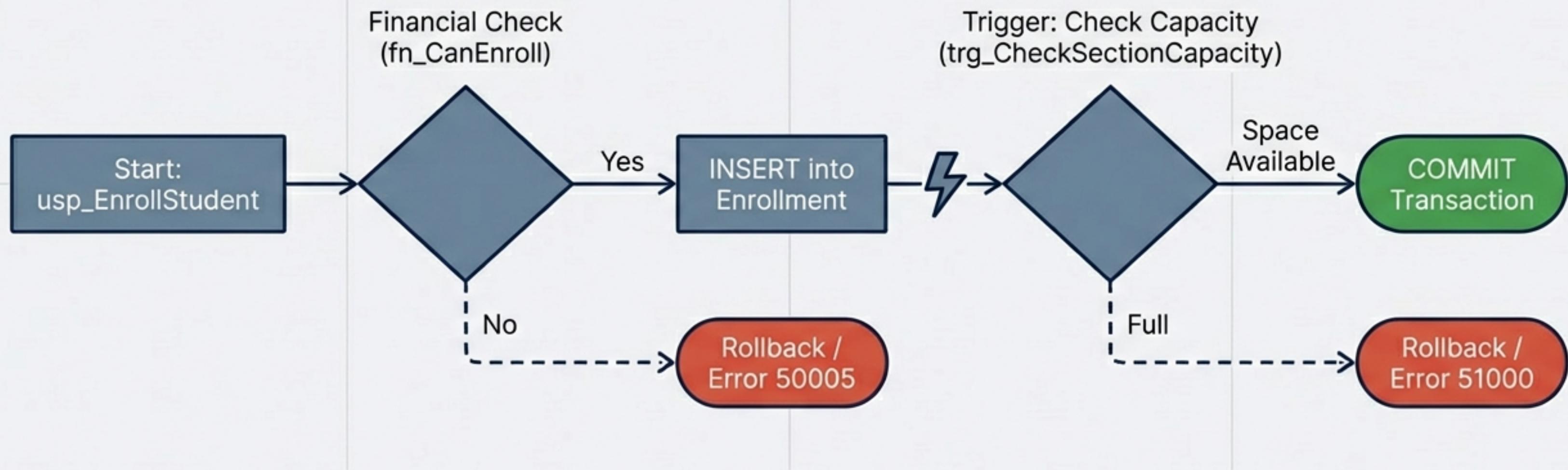
# Architectural Design Pattern



- **No Direct Table Access:** Users interact strictly through Procedures and Views.
- **Encapsulation:** Complex logic (e.g., GPA Calc) is hidden in functions.
- **Reactive Data:** Triggers prevent invalid states before commit.

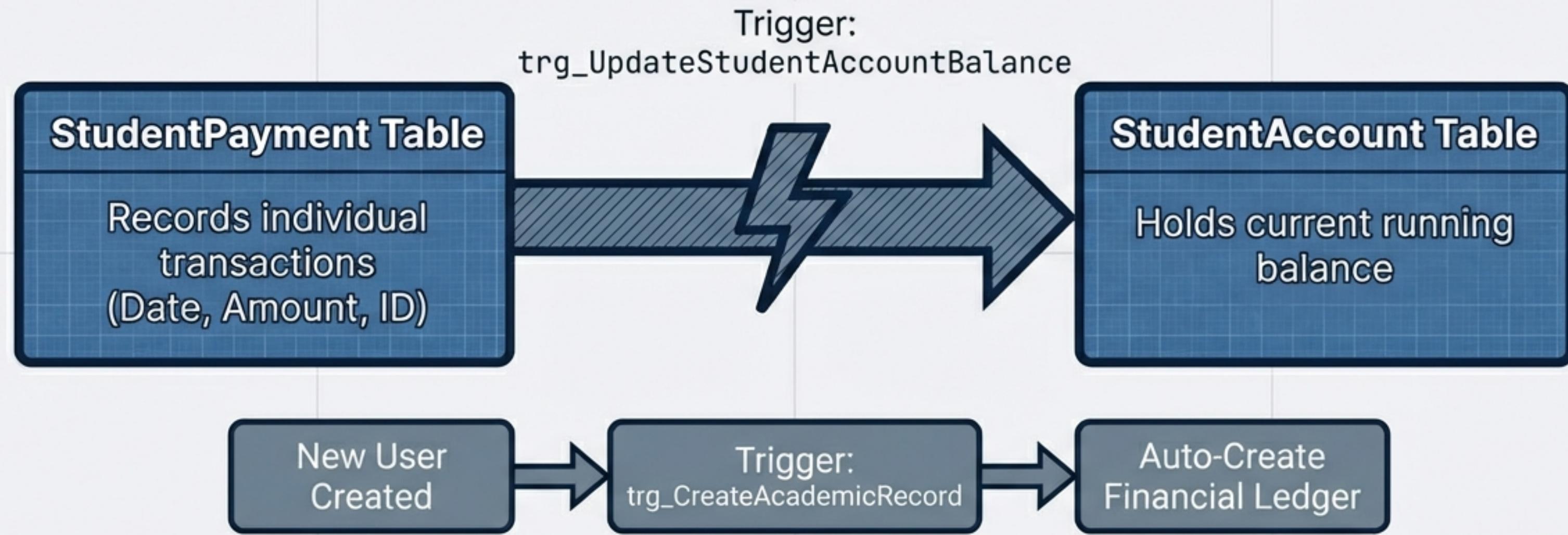


# Workflow: Intelligent Enrollment



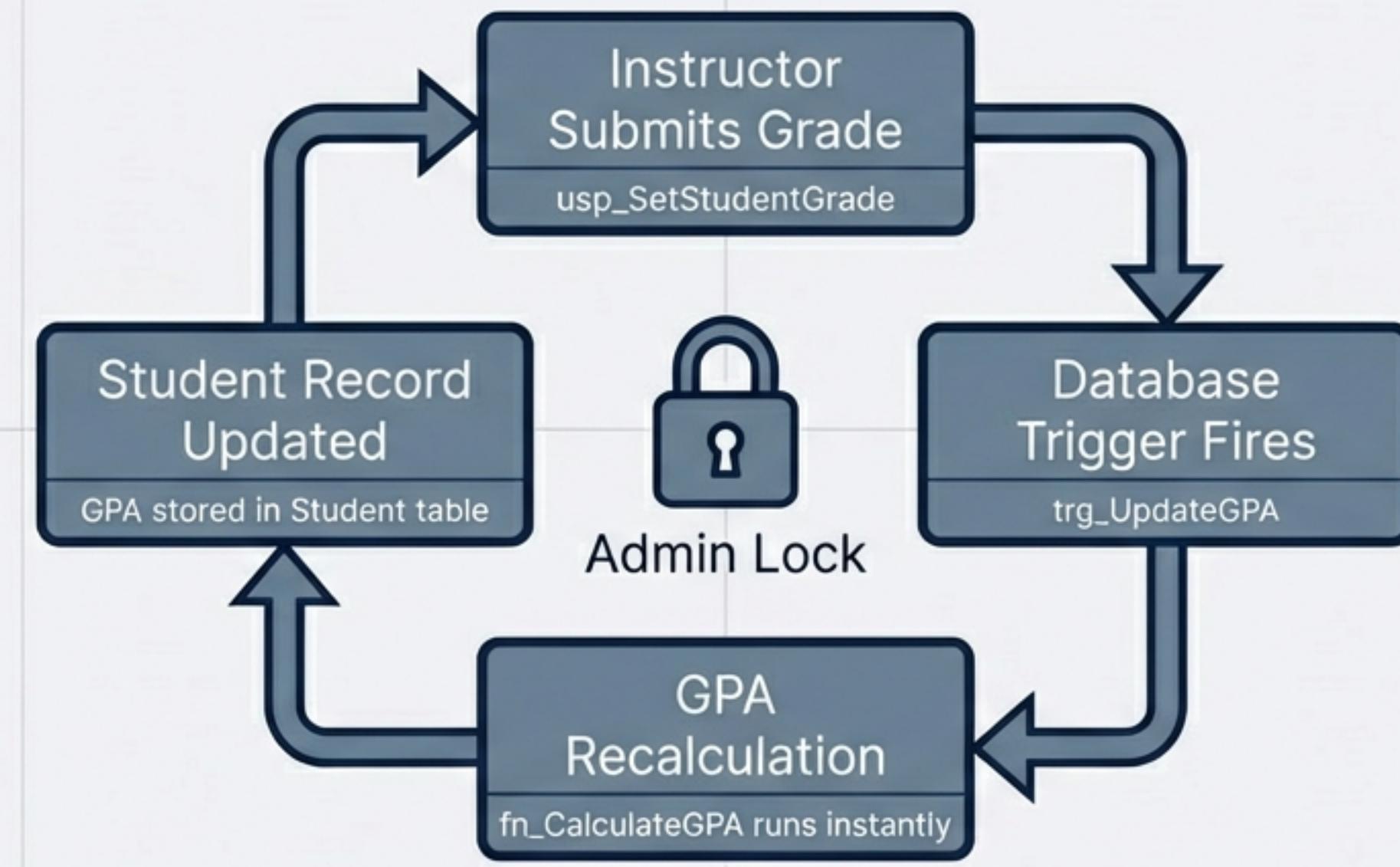
Enrollment is a complex transaction requiring simultaneous validation of academic availability and financial standing.

# Financial Module: Student Accounts



Separation of transaction history from current standing for auditability. Balances update in real-time to support Enrollment checks.

# Grading Lifecycle & Automation



Grades move from 'Draft' to 'Locked' status. All changes trigger an immediate, automated GPA recalculation ensuring the transcript is always up to date.

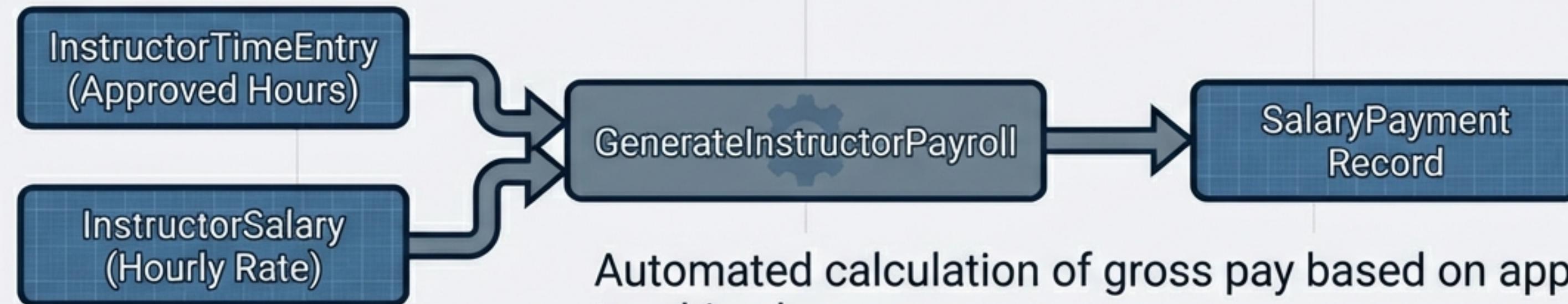
# Instructor Management & Payroll

## Workload Policy



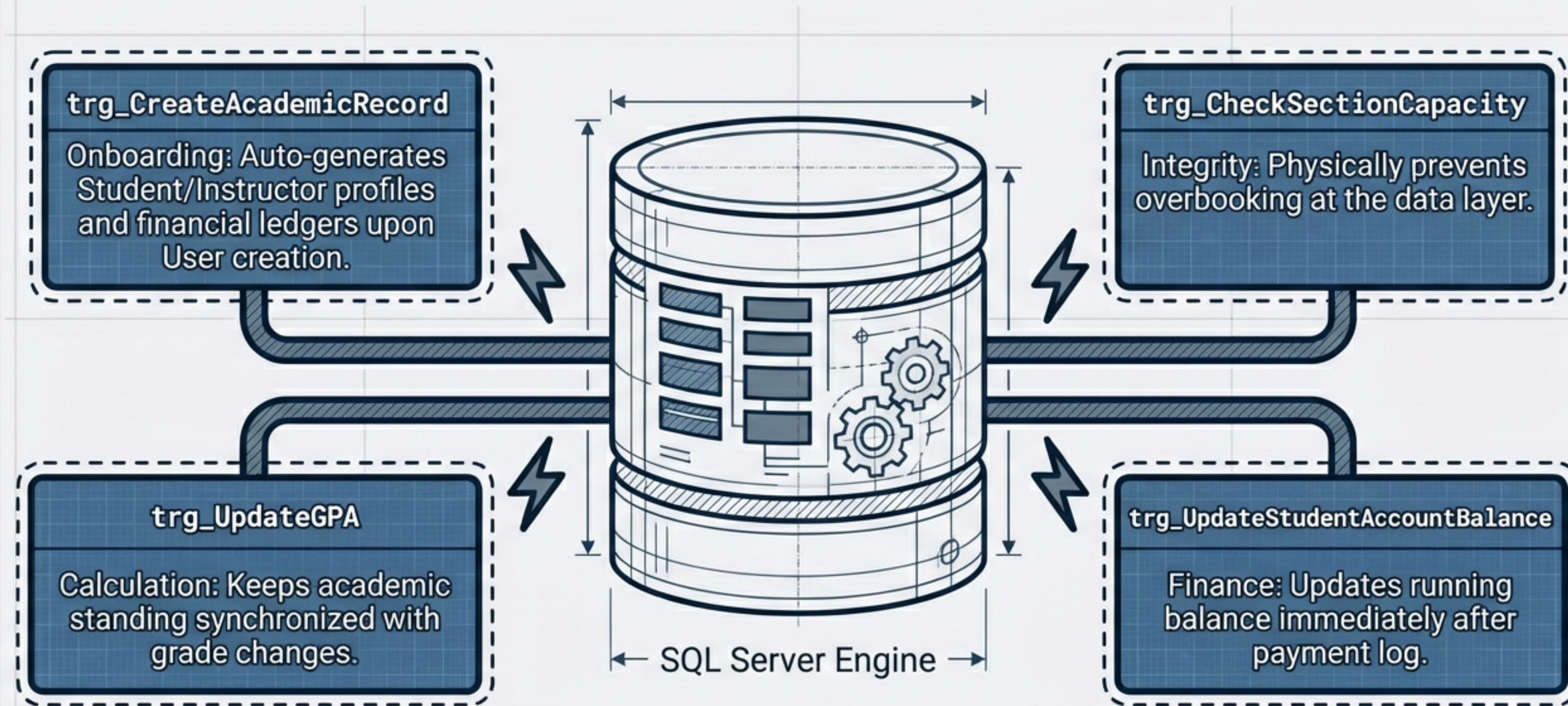
Procedure `AssignInstructorToSection` validates that `CurrentLoad + NewCredits` does not exceed the limit defined in `InstructorPolicy`.

## Payroll Generation



Automated calculation of gross pay based on approved teaching hours.

# The “Active Database” Concept



# Reporting & Views

## Official Transcript - University Management System

Student Name: Aseel Menhem

Program: Computer Engineering

Course Code	Title	Semester	Credits	Grade
CE301	Data Structures	Fall 2024	3	A
EE410	Power Systems	Fall 2024	4	B

Cumulative GPA: 3.55

## Secure Views used:

- vw\_StudentTranscript: Joins 5 tables for a unified history.
- vw\_SectionCapacityStatus: Real-time dashboard for Registrars.
- vw\_InstructorTeachingLoad: Faculty workload monitoring.