Software Requirements Specification

Version 3.0

04.03.2025

University Department Information System (UDIS)

122CS0096: PRAPTI PRACHITA PRADHAN
122CS0076: ARCHITA DUTTA
122CS0116: JEEVAN TADWAL
122CS0296: DHARAVATH LIKITHA

Table of Contents

Table of Contents i
List of Figures ii
1.0. Introduction 1
1.1. Purpose 1
1.2. Scope of Project 1
1.3. Glossary 2
1.4. References 2
1.5. Overview of Document 2
2.0. Overall Description 4
2.1 System Environment 4
2.2 Functional Requirements Specification 5
2.2.1 Customer Use Case 5
Use case: Search Article 5
2.2.2 Sales Clerk Use Case 6
Use case: Submit Article 6
2.2.3 Manager Use Case 7
Use case: Submit Review 7
2.2.4 Editor Use Cases 8
Use case: Update Author 8
Use case: Update Reviewer 9
Use case: Update Article 9
Use case: Receive Article 10
Use case: Assign Reviewer 11
Use case: Receive Review 11
Use case: Check Status 12
Use case: Send Response 12
Use case: Send Copyright 13
Use case: Remove Article 14
Use case: Publish Article 14
2.3 User Characteristics 15
2.4 Non-Functional Requirements 15
3.0. Requirements Specification 17
3.1 External Interface Requirements 17
3.2 Functional Requirements 17
3.2.1 Item Management 17
3.2.2 Billing and Payments 18
3.2.3 Inventory Management 18
3.2.4 Sales Statistics and Reporting 19
3.2.5 Price Management 19
3.2.6 Update Article Status 20
3.2.7 Enter Communication 20
3.2.8 Assign Reviewer 21
3.2.9 Check Status 21
3.2.10 Send Communication 22
3.2.11 Publish Article 22
3.2.12 Remove Article 23
3.3 Detailed Non-Functional Requirements 23

3.3.1 Logical Structure of the Data..... 23
3.3.2 Security..... 25
Index..... 26

List of Figures

Figure 1 - System Environment 4
Figure 2 - Article Submission Process 6
Figure 3 - Editor Use Cases 8
Figure 4 - Logical Structure of the Article Manager Data 23

1.0. Introduction

The University Department Information System (UDIS) is a web-based application designed to manage and streamline administrative tasks within university departments. It centralizes the management of student records, course registrations, grading, department inventory, financial transactions, and research activities. By automating processes, the system improves efficiency, accuracy, and accessibility for students, faculty, and administrative staff.

We will base this project on our college website, NITRIS, which offers secure login functionalities for both students and staff.

The system administrator will assign role-based credentials to each department member, controlling their access levels and permissions. These roles include the secretary, instructors, students, research assistants, and department heads, each with access to the features and information relevant to their responsibilities.

The system architecture follows a modern three-tier design, leveraging the MERN stack: a responsive front-end built with React.js for an enhanced user experience, an Express.js backend running on Node.js to handle business logic and API endpoints, and a MongoDB database for efficient data management.

1.1. Purpose

This document provides a detailed overview of the University Department Information System (UDIS), covering its core functionalities, technical architecture, user interfaces, operational parameters, and response mechanisms for various use cases. It serves as a reference for developers, system administrators, end-users, and department leadership. The system requires approval from the Dean's office and the IT governance board before deployment.

1.2. Scope of Project

The University Department Information System (UDIS) streamlines academic and administrative processes with the following functionalities:

1.2.1 Student Management

- Stores student details, including name, address, and registered courses.
- 8Tracks completed courses and backlogs.
- Allows the department secretary to register students for courses each semester.
- Displays student records when a roll number is entered.
- Computes semester GPA and cumulative GPA (CGPA).
- Generates and prints student grade sheets.

1.2.2 Inventory Management

- Maintains records of department equipment, furniture, and their locations.

1.2.3 Financial Management

- Tracks annual grants from the university and expenditures on books, equipment, and stationery.
- Records income from consultancy services.
- Manages departmental accounts, including income, expenses, and balance.

1.2.4 Research and Faculty Management

- Documents ongoing research projects within the department.
- Maintains records of faculty publications.

1.2.5 Query and Reporting System

- Supports real-time queries for student details using roll numbers.
- Allows financial queries, displaying department income, expenditures, and balance.

This system ensures accuracy, efficiency, and transparency in managing departmental operations.

1.3. Glossary

Term	Definition
Student Record	A database entry containing a student's personal details, registered courses, completed courses, and backlogs.
Semester GPA (SGPA)	The grade point average calculated for a specific semester based on course grades.
Cumulative GPA (CGPA)	The overall grade point average calculated from all completed semesters.
Backlog	A course that a student has failed and needs to retake.
Grade Sheet	A document displaying a student's grades for a semester.
Department Inventory	A record of equipment, furniture, and other assets within the department.

Equipment Tracking	The process of monitoring the location and status of departmental assets.
Annual Grant	Yearly funding provided by the university for departmental expenses.
Departmental Accounts	Financial records tracking income, expenses, and balances.
Expenditure	Money spent on books, equipment, stationery, and other departmental needs.
Research Project	A study or investigation conducted by faculty members or researchers within the department.
Faculty Publications	Research papers, books, and articles authored by department faculty members.
Financial Query	A function that provides details of departmental income, expenditure, and balance.
Real-time Reporting	Instant generation of reports based on up-to-date data.

1.4. References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.

1.5. Overview of Document

The next chapter, the **Overall Description** section, provides an overview of the functionality of the supermarket automation software. It outlines the informal requirements and serves to establish a context for the detailed technical requirements specified in the following chapter.

The third chapter, the **Requirements Specification** section, is primarily intended for developers and describes the system's functionality in technical terms. It details how various components, such as inventory management, pricing updates, and point-of-sale operations, are implemented within the software.

Both sections describe the same supermarket automation system in its entirety but are tailored for different audiences, using different levels of technical language.

.

2.0. Overall Description

2.1 Product Perspective

The **University Department Information System (UDIS)** is designed as an integrated platform for managing academic and administrative operations within a university department. It serves as a digital transformation of traditional paper-based processes, ensuring efficiency, accuracy, and accessibility. The system will be a **web-based** solution leveraging the **MERN stack (MongoDB, Express.js, React.js, Node.js)**. It will be deployed on a secure cloud infrastructure, ensuring scalability and availability.

2.2 Product Functions

- **Student Management:** Capture and maintain student profiles, including personal information and academic records.
- **Course Registration:** Facilitate course enrollment at the beginning of each semester.
- **Grading System:** Allow faculty to input grades and generate automated grade sheets.
- **Inventory Tracking:** Maintain a catalog of department assets, including equipment and furniture.
- **Financial Management:** Track departmental budgets, expenses, and income sources.
- **Research and Faculty Management:** Store research publications and ongoing projects.
- **Query and Reporting:** Provide real-time analytics and reporting capabilities.

2.3 User Characteristics

- **Students:** Should be familiar with web navigation to access course registration, academic records, and grading information.
- **Instructors:** Should have proficiency in using web-based systems to manage grades and student performance.
- **Department Secretaries:** Should handle administrative tasks, including student registration, inventory management, and financial tracking.
- **Department Heads:** Should have access to high-level reports and approve financial and academic decisions.
- **Administrators:** Should maintain the system, manage user access, and ensure data security.

2.4 Constraints

The system must comply with university data protection policies to ensure the confidentiality and integrity of sensitive information. It should be capable of efficiently handling at least 500 concurrent users, maintaining optimal performance under high loads. To guarantee reliability, the system must achieve 99.9% uptime, minimizing disruptions and ensuring continuous availability. Additionally, it should adhere to all relevant academic and financial governance policies, ensuring regulatory compliance and seamless integration within the university's operational framework.

2.5 Assumptions and Dependencies

The system operates under the assumption that all users have internet access and possess basic familiarity with web applications, ensuring a smooth user experience. It relies on the university's authentication services to implement role-based access control, maintaining secure and restricted access as needed. To optimize cost efficiency, the system will be developed using open-source technologies, reducing licensing expenses. Additionally, the university IT department will be responsible for providing the necessary hardware and network support, ensuring stable infrastructure and seamless system deployment.

3.0. Requirements Specification

3.1 External Interface Requirements

3.1.1 User Interfaces

- The system shall provide a web-based graphical user interface accessible via modern browsers.
- The user interface shall include dashboard views for different user roles.
- The system shall support responsive design, ensuring usability on desktops, tablets, and mobile devices.

3.1.2 Hardware Interfaces

- The system shall be compatible with standard desktop and mobile devices.
- The system shall interface with barcode scanners and printers for record-keeping and report generation.

3.1.3 Software Interfaces

- The system shall integrate with university authentication systems (e.g., LDAP, OAuth).
- The system shall interface with financial management software for budget tracking.

3.1.4 Communication Interfaces

- The system shall support email notifications for course registration, grades, and financial transactions.
- The system shall enable API-based access for third-party applications to retrieve data securely.

3.2 Functional Requirements

3.2.1 Student Management

- The system shall allow department secretaries to **register new students**.
- The system shall track **student records**, including personal details and academic history.
- The system shall support **automatic course enrollment verification**.

3.2.2 Course Registration

- Students shall be able to **register for courses** at the beginning of each semester.
- The system shall ensure that **prerequisite requirements** are met before allowing registration.

3.2.3 Grading System

- Instructors shall be able to **enter grades** for students.
- The system shall **automatically compute GPA and CGPA** based on student performance.

3.2.4 Inventory Management

- The system shall track **departmental assets**, including lab equipment and furniture.
- The system shall generate **inventory reports** for department administration.

3.2.5 Financial Management

- The system shall track **annual department funding** and expenses.
- The system shall allow authorized users to **generate financial reports**.

3.2.6 Research and Faculty Management

- The system shall allow faculty to **record research projects and publications**.
- The system shall provide **search functionality** to query faculty research records.

3.2.7 Reporting and Queries

- The system shall support **querying student details using roll numbers**.
- The system shall provide **real-time financial reporting** for department heads.

3.3 Detailed Non-Functional Requirements

3.3.1 Performance Requirements

- The system shall support at least 500 concurrent users.
- The system shall process user queries within 2 seconds.
- Reports shall be generated within 5 seconds.
- The system uptime shall be maintained at 99.9%.

3.3.2 Safety and Security Requirements

- The system shall implement role-based access control (RBAC) to restrict access to authorized users.
- All data transmissions shall be encrypted using TLS 1.3.
- User authentication shall support multi-factor authentication (MFA) for administrative users.
- Automated backup mechanisms shall allow system recovery within 30 minutes.
- System logs shall be securely stored for audit compliance.

3.3.3 Software Quality Attributes

3.4.3.1 Reliability

- The system shall provide automated error handling and fault recovery mechanisms.
- Database replication shall ensure data redundancy and failover support.
- Maximum allowable downtime shall not exceed 15 minutes per incident.

3.3.3.2 Maintainability

- The system shall be developed using modular architecture for easy updates.
- A logging and debugging system shall be in place for administrators.
- The system shall follow industry-standard version control for code management.

3.3.3.3 Usability

- The user interface shall be intuitive and accessible, requiring minimal training.
- The system shall include search functionality for quick access to records.

- The system shall comply with WCAG 2.1 standards to ensure accessibility for all users.

3.3.3.4 Scalability

- The system shall support scaling up to 5,000 users.
- It shall be optimized for distributed computing and cloud deployment.
- API endpoints shall handle large dataset queries efficiently.

3.4 Data Dictionary

3.4.1 Student Table

Field Name	Description	Data Type
Student_ID	Unique identifier for the student.	Integer/String
Student_Name	Name of the student.	String
Student_Address	Residential address of the student.	String
Student_Contact	Contact phone number or email address.	String
Courses_Completed	List of completed courses.	Array of Strings
Courses_Backlog	List of backlogged courses.	Array of Strings
Current_GPA	GPA for the current semester.	Float
Cumulative_GPA	Cumulative GPA across all semesters.	Float

☐

3.4.2 Course Table we

Field Name	Description	Data Type
Course_ID	Unique identifier for the course.	Integer/String
Course_Name	Name of the course.	String
Instructor_Name	Name of the instructor for the course.	String
Credits	Number of credits awarded for completing the course.	Integer
Semester_Offered	The semester when the course is offered.	String
Course_Description	Description of the course content.	String

3.4.3 Inventory Table

Field Name	Description	Data Type
Inventory_ID	Unique identifier for the inventory item.	Integer/String
Item_Name	Name of the inventory item (e.g., equipment).	String
Item_Description	Description of the inventory item.	String
Quantity_Available	Number of units available.	Integer
Item_Price	Price of a single unit of the inventory item.	Float
Location	Physical location of the item.	String

3.4.4 Research & Publications Table

Field Name	Description	Data Type
Research_Project_ID	Unique identifier for the research project.	Integer/String
Project_Name	Name of the research project.	String
Project_Description	A brief description of the research project.	String
Faculty_Name	Name of the faculty member(s) involved.	String
Publication_Title	Title of the publication.	String
Publication_Details	Additional publication details.	String

3.4.5 Financial Records Table

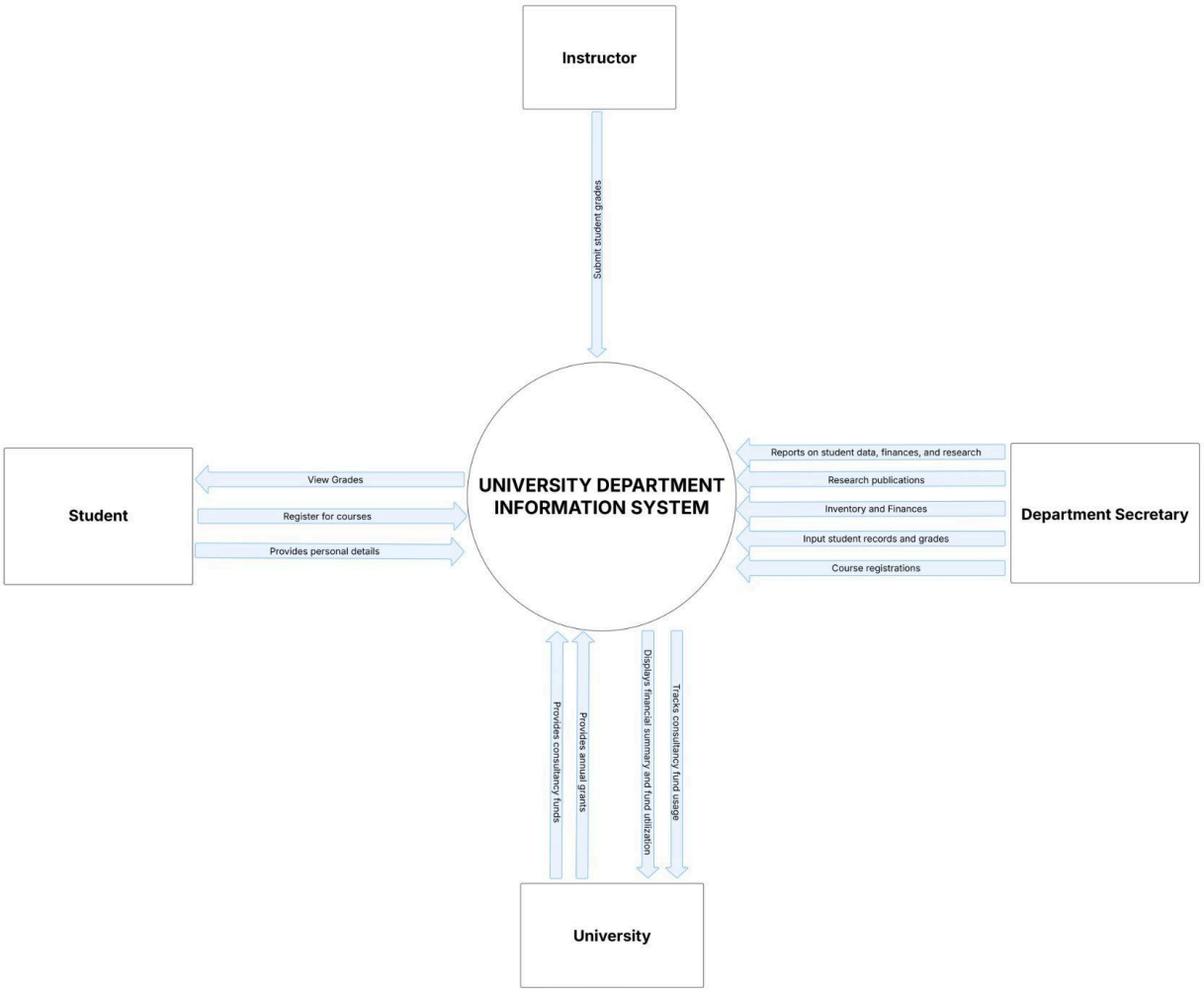
Field Name	Description	Data Type
Income_Type	Type of income received (e.g., grant, consultancy).	String
Income_Amount	Amount of income received.	Float
Expenditure_Type	Type of expenditure (e.g., salaries, equipment).	String
Expenditure_Amount	Amount of money spent.	Float
Balance	Remaining balance.	Float
Cash_Book_Report	Record of income, expenditure, and balance summary.	String

3.4.6 Query System Table

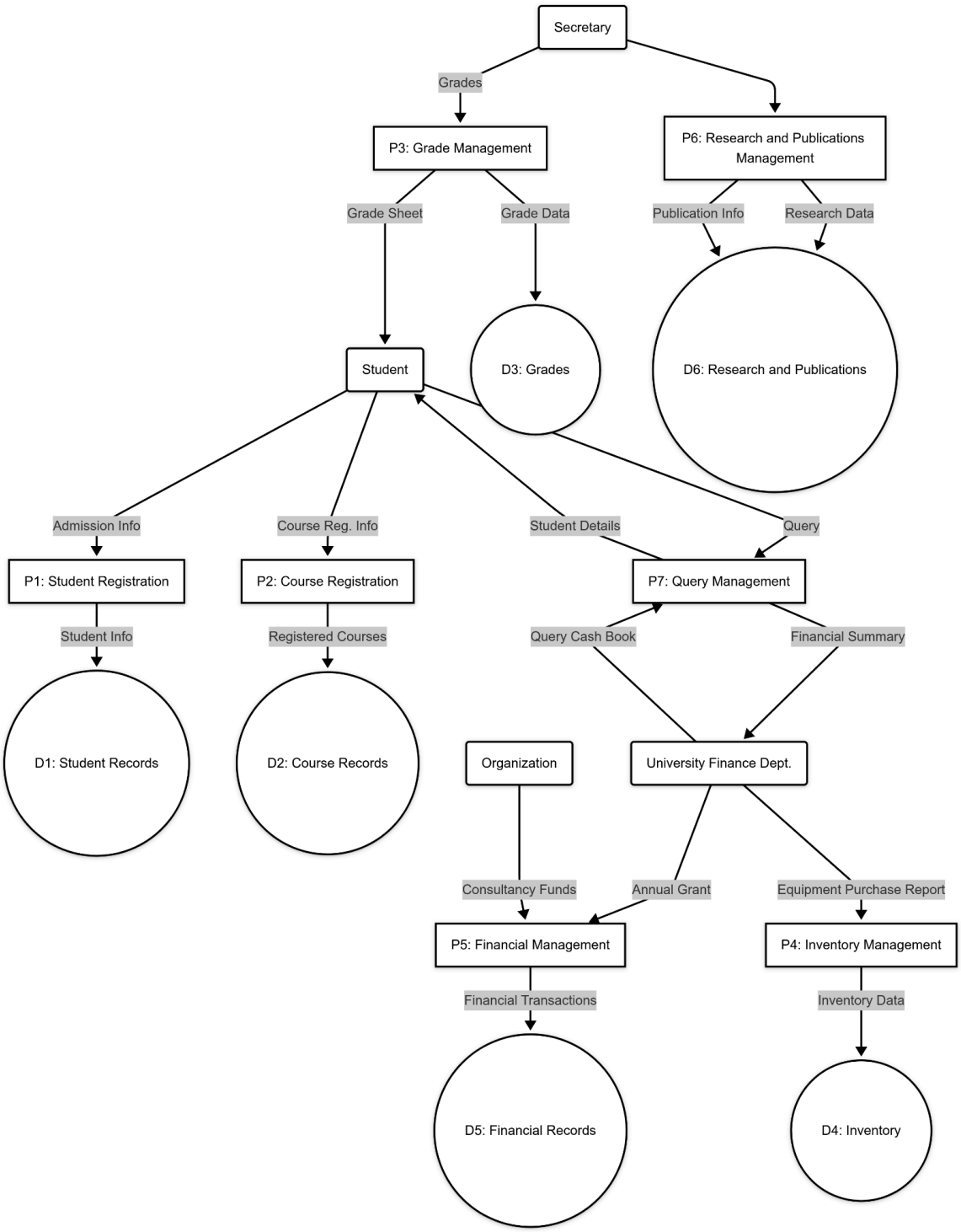
Field Name	Description	Data Type
Query_Type	Type of query (e.g., "Student", "Cash Book").	String
Roll_Number	Roll number used to query student details.	Integer/String
Query_Result	Output generated from the query.	String

3.5 Data Flow Diagram(DFD)

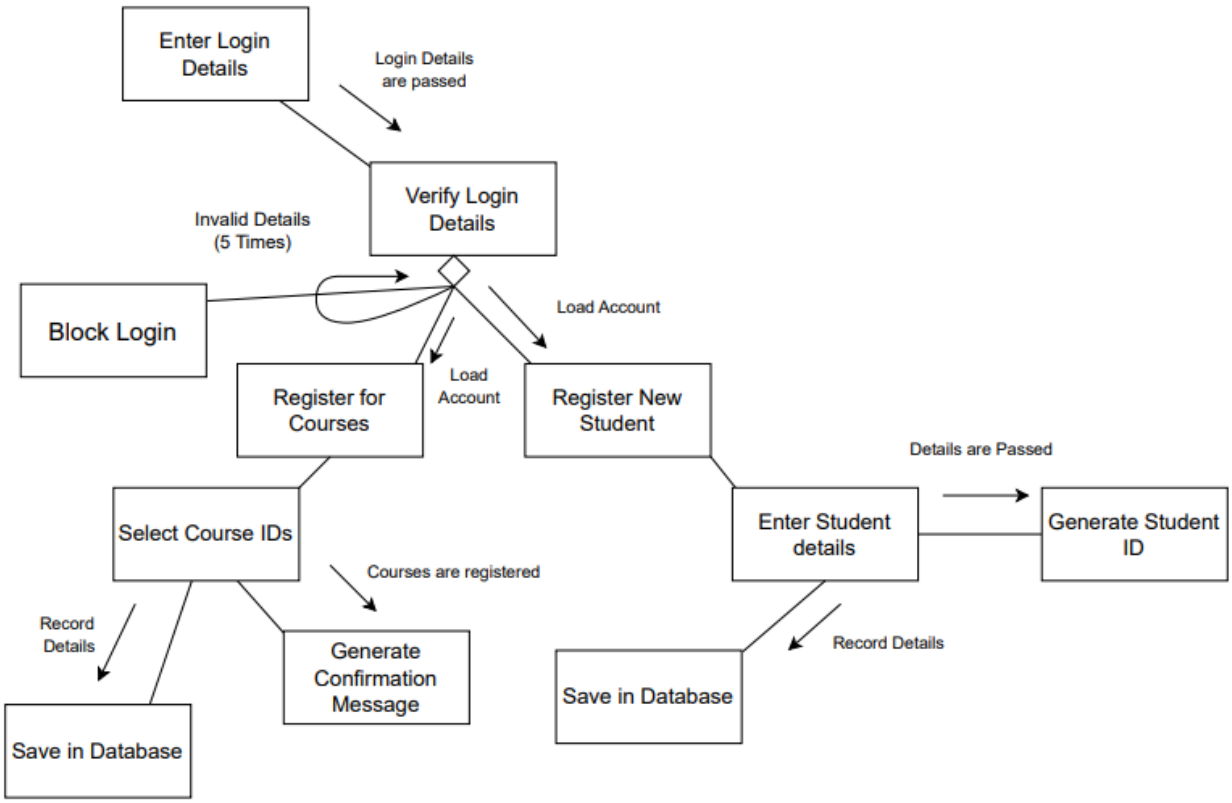
Level 0:



Level 1:


Secretary
Grades
P3: Grade Management
P6: Research and Publications Management
Grade Sheet
Grade Data
Publication Info
Research Data
Student
D3: Grades
D6: Research and Publications
Admission Info
Course Reg. Info
Student Details
Query
P1: Student Registration
P2: Course Registration
P7: Query Management
Student Info
Registered Courses
Query Cash Book
Financial Summary
D1: Student Records
D2: Course Records
Organization
University Finance Dept.
Consultancy Funds
Annual Grant
Equipment Purchase Report
P5: Financial Management
P4: Inventory Management
Financial Transactions
Inventory Data
D5: Financial Records
D4: Inventory

3.6 Structure Chart



3.7 Use Case

3.7.1 Login

Actors: Student, Instructor, Department Secretary
Description: Allows users to authenticate into the system.
Includes:

- Student Login (For students to access their details)
- Instructor Login (For instructors to manage courses and grades)
- Department Secretary Login (For administrative tasks)

3.7.2 Display Student Details

Actors: Student, Department Secretary
Description: Fetches and displays student information, including personal details, courses, and grades.

3.7.3 Register New Student

Actors: Department Secretary
Description: Allows the department secretary to register a new student in the system, entering details like name, address, roll number, and course information.

3.7.4 Semester Registration

Actors: Student, Department Secretary
Description: Enables students to register for a semester. The department secretary can also assist in course registration.
Includes:

- Student Registration (Where students register for their semester)
- Course Registration (Selection of courses for the semester)

3.7.5 Course Registration

Actors: Student, Department Secretary
Description: Allows students to enroll in specific courses for the semester.

3.7.6. Display Student Course Information

Actors: Department Secretary
Description: Shows details of courses registered by a student, including completed courses and backlogs.

3.7.7 Grades Calculation

Actors: Instructor
Description: Instructors enter grades for students, and the system calculates the semester GPA and CGPA.
Extends:

- Print Grade Sheet (After grade calculation, the grade sheet can be generated)

3.7.8 Print Grade Sheet

Actors: Department Secretary
Description: Generates a grade sheet for a student, displaying semester GPA and CGPA.

3.7.9 Manage Inventory

Actors: Department Secretary
Description: Keeps track of department assets like equipment, furniture, and their locations.

3.7.10 Manage Department Account

Actors: Department Secretary
Description: Maintains financial records, including income (grants, consultancy funds) and expenses (equipment, books, stationery).

3.7.11 Update Research Information

Actors: Department Secretary
Description: Maintains research project details, including faculty publications and ongoing research projects.

3.7.12 Register New Instructor

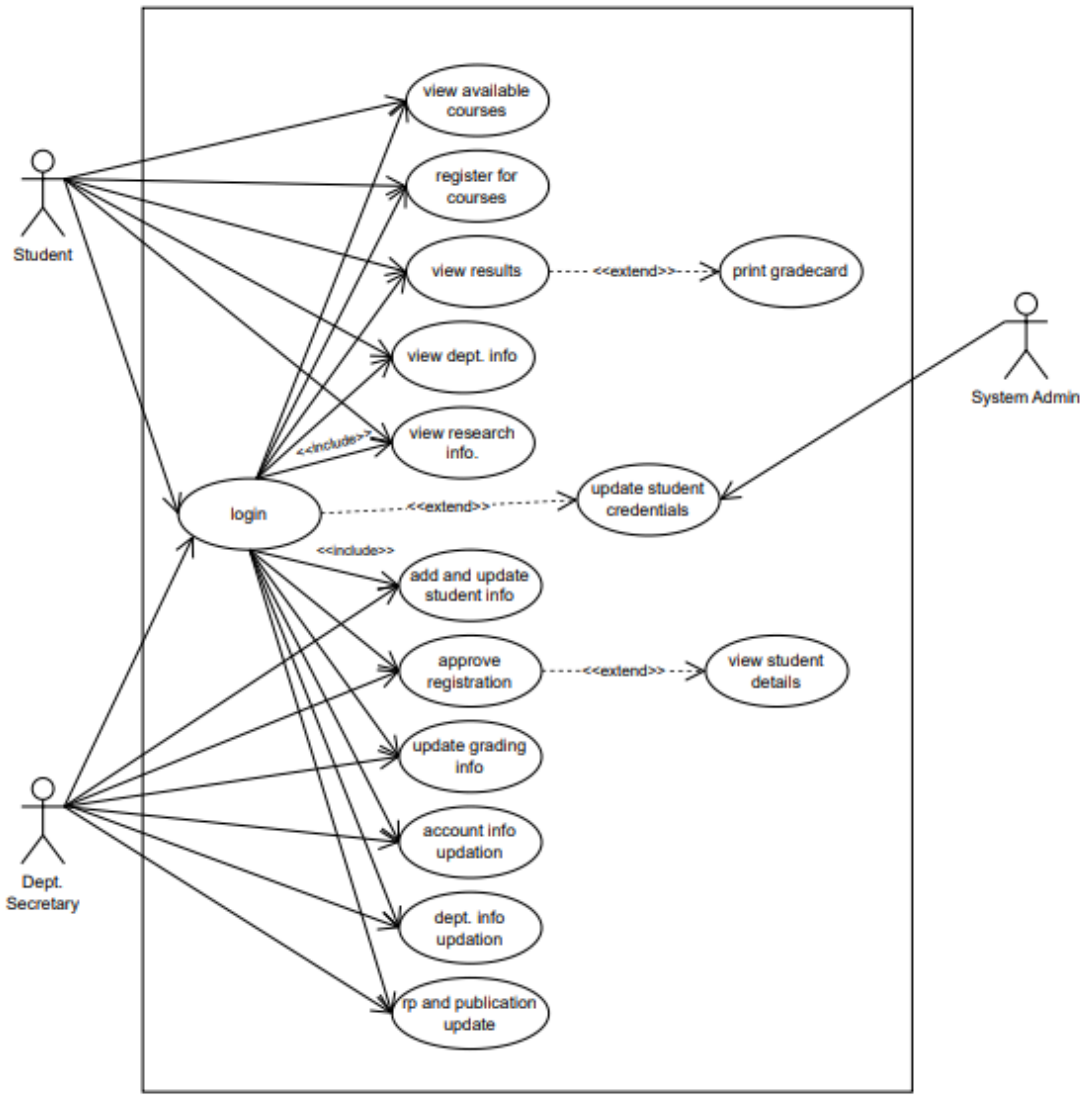
Actors: Department Secretary

Description: Allows the department secretary to add new instructors to the system.

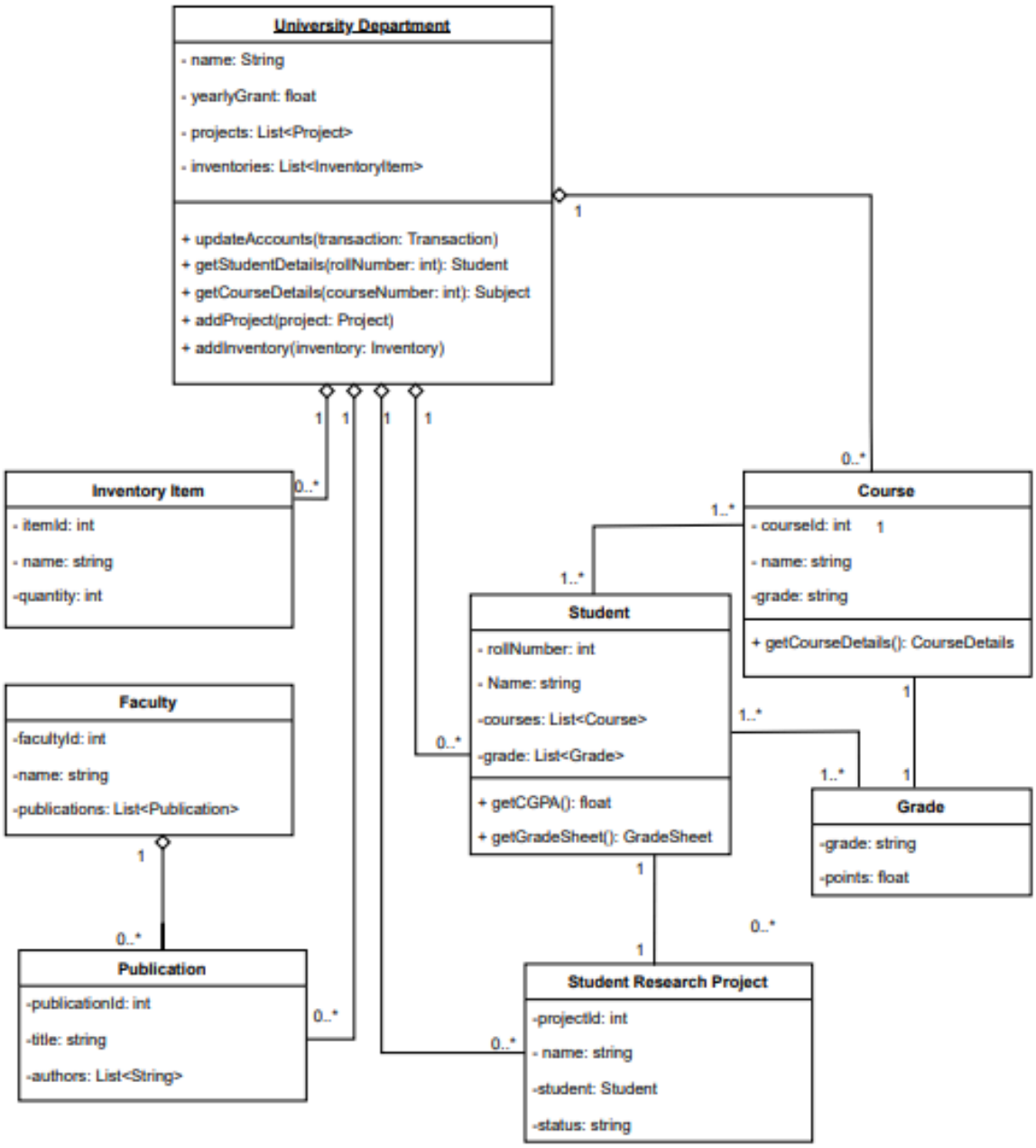
3.7.13 Display Instructor Details

Actors: Instructor

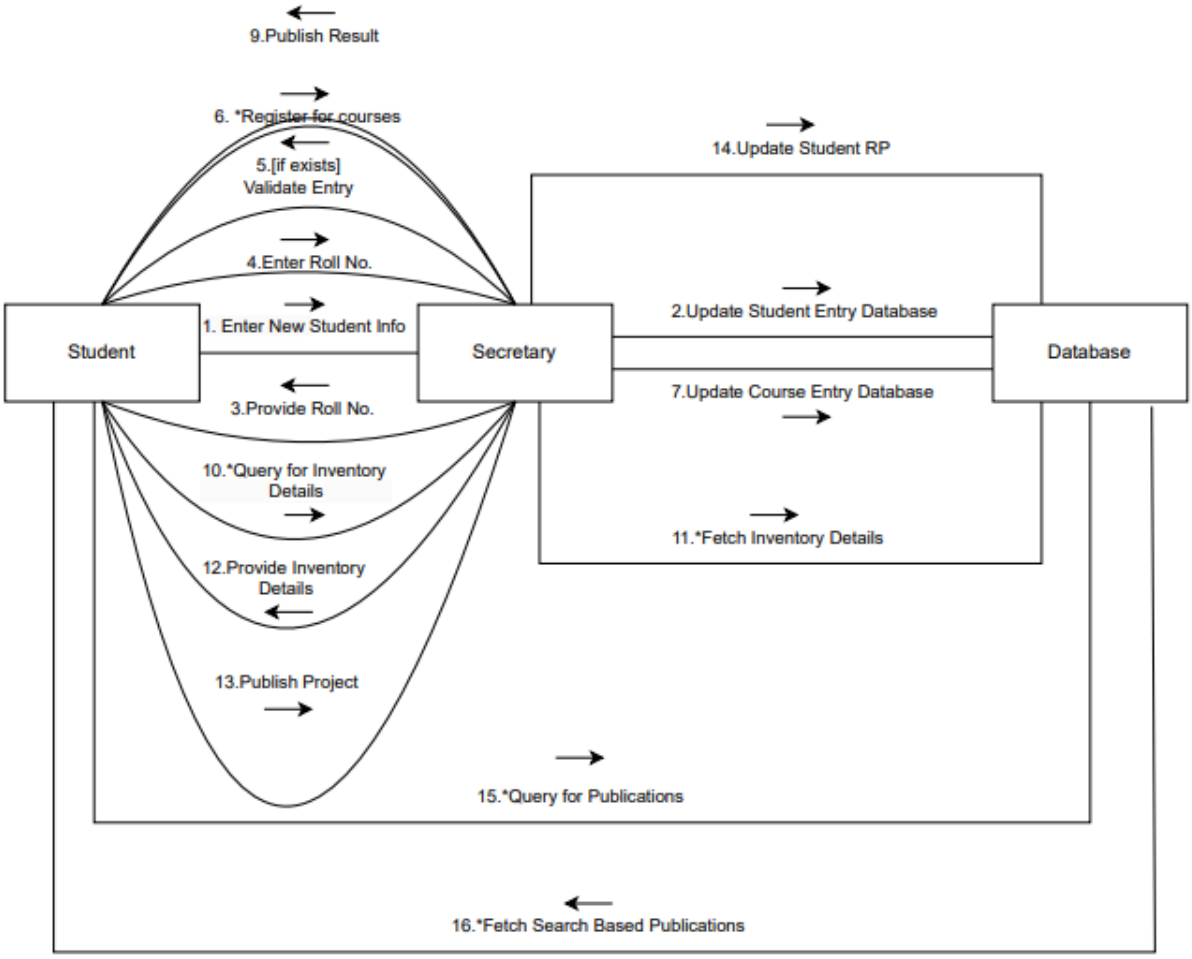
Description: Enables instructors to view their details, including assigned courses and research contributions.



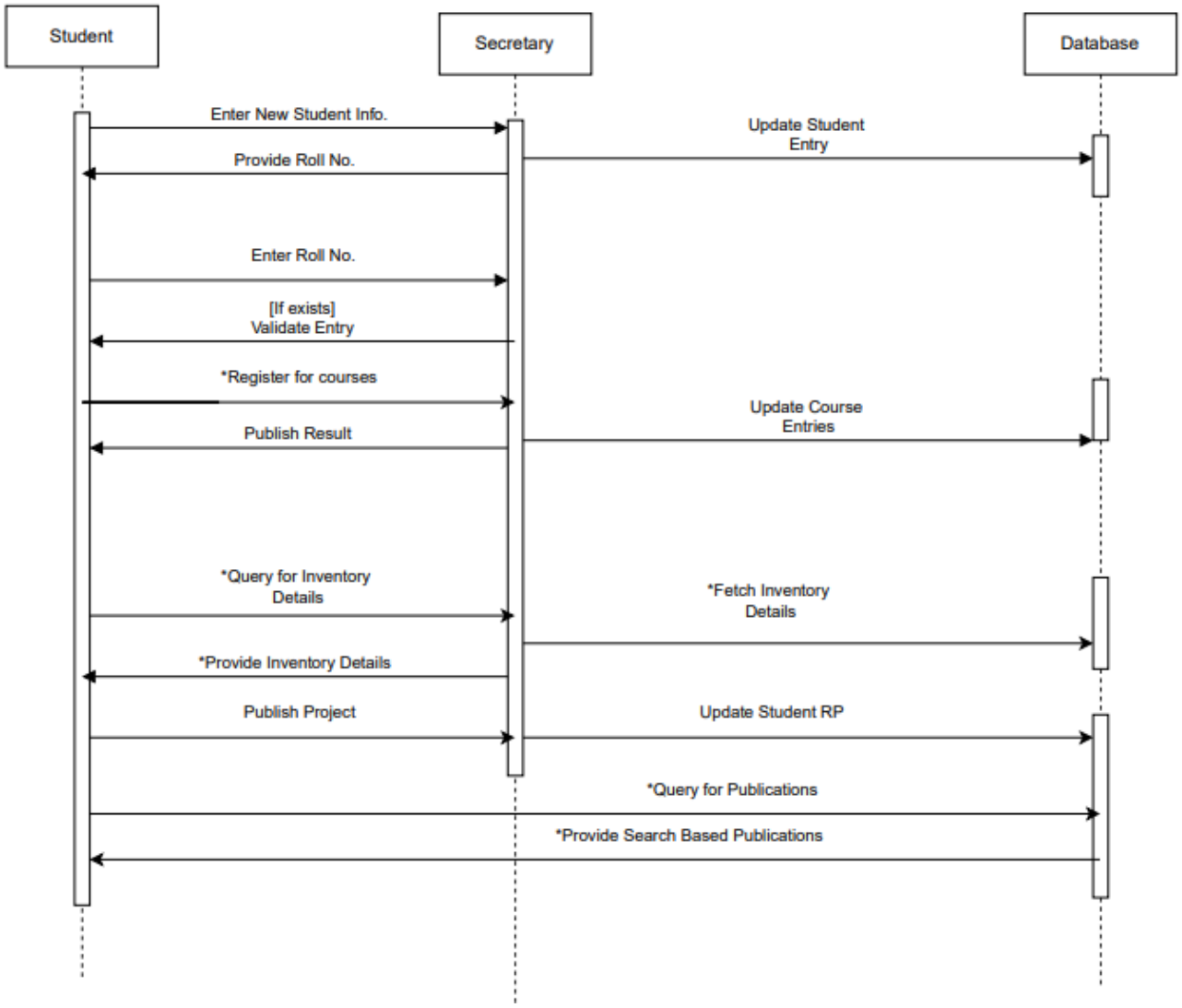
3.8 Class Diagram



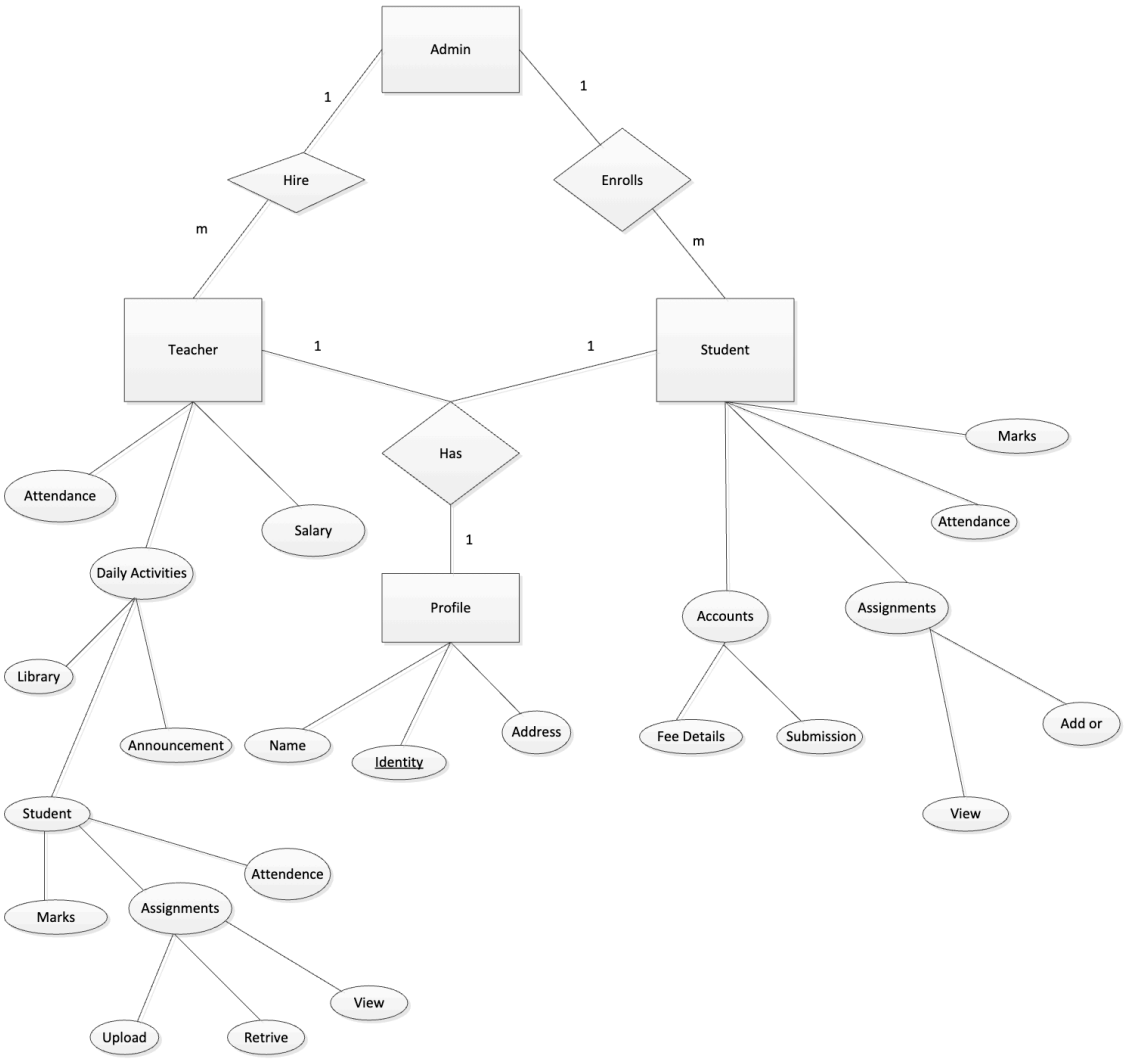
3.9 Collaboration Diagram



3.10 Sequence Diagram



3.11 ER Diagram



Index

Abstract, 6, 17, 27
add, 9, 11, 19, 20, 21
Add, 8, 9, 19
Article, 1, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28
Article Manager, 5, 8, 9, 10, 11, 12, 13, 14, 15, 16, 19, 20, 21, 22, 23, 24, 25, 28
Author, 1, 4, 5, 6, 7, 8, 9, 13, 14, 16, 17, 19, 20, 22, 23, 25, 26, 27
Category, 5, 14, 17, 18, 20, 21, 23, 26, 27

Database, 2, 9, 11, 14, 15, 16, 17, 18, 19, 20, 21, 22, 24, 25, 26, 27
Editor, 1, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 19, 20, 21, 22, 23, 24, 25, 28
Field, 17, 19, 20
Form, 1, 6, 9, 10, 11, 12, 14, 19, 20, 21, 23, 24, 27
Grid, 9, 11, 12, 19, 20, 21
Historical Society, 1, 5, 9, 11, 16, 17, 19, 20, 26
Online Journal, 4, 5, 6, 7, 15, 16, 17, 18, 24, 27, 28
Reader, 4, 5, 6, 16, 17, 18
Review, 1, 7, 11, 12, 18, 21, 23, 26, 27
Reviewer, 1, 4, 5, 6, 7, 9, 11, 16, 17, 19, 20, 21, 22, 23, 26, 27
Security, 27, 28
Status, 11, 12, 13, 14, 17, 21, 22, 23, 27
update, 9, 11, 20, 21
Update, 8, 9, 10, 11, 12, 13, 14, 15, 17, 19, 20, 21, 22
User, 7, 16, 18
Web Publishing System, 1, 4, 5, 17