Software Engineering Lab

­

***University Department Information System***

( Software Requirement Specification Document)

## **Prepared by:**

**Purvi Baranwal**

**121CS0142**

**B.Tech., Pre-Final Year**

National Institute of Technology, Rourkela

Department of Computer Science and Engineering

CONTENTS

# Introduction

*1.1 Purpose*

*1.2 Scope*

* 1. *Environmental Characteristics*
  2. *Definitions*

*1.5 References*

*1.6 Overview of Developer’s Responsibilities*

# General Description

*2.1Product Perspective*

*2.2 Product Function Overview*

*2.3 User Characteristics*

*2.4 Operating Environment*

*2.5 General Constraints*

# Functional Requirement

*3.1 Introduction*

*3.2 Inputs*

*3.3 Processing*

*3.4 Outputs*

# External Interface Requirements

# Performance Requirements

# Design Constraints

# Other Requirements

# Introduction

1.1 Purpose:

The purpose of this document is to present a detailed description of the University Department Information System. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate and how the system will react to external stimuli. This document is intended for both the stakeholders (students and department secretary) and the developers of the system and will be used to evaluate my assignment for Software Engineering Lab for sixth semester.

1.2 Scope

This software system will be a University Department Information System. Department offices in different universities do a lot of book-keeping activities and it is necessary to develop software to automate these activities. This system will be designed to maximize the department secretary’s productivity by providing tools to assist in automating student records, course registrations, grade tracking, inventory management, financial accounting, research projects, and faculty publications which would otherwise have to be performed manually. By maximizing the secretary’s work efficiency and production the system will meet the needs while remaining easy to understand and use.

More specifically, this system is designed to allow students to register for their respective courses and check their grade cards. The software will facilitate up to date information exchange between students, faculties, and the department through the system to provide a uniform review process. The system also contains a relational database containing a list of student records, course registrations, inventory, financial transactions, research projects, and publications.

1.3 Environmental Characteristics

**Hardware Environment**

1. ***Office Computers:*** The software will interact with standard office computers, including desktops and laptops, used by department secretaries, instructors, and administrative staff.
2. ***Printers and Scanners:*** Interaction with peripherals for generating hard copies of grade sheets and reports.

**Software Environment**

1. ***Operating System:*** Compatible with prevalent operating systems like Windows, macOS, and Linux.
2. ***Web Browsers:*** Accessible through standard web browsers (Chrome, Firefox, Safari, etc.) for user interfaces and online functionalities.
3. ***Database Systems:*** Interacts with university databases for storing and retrieving student information securely.
4. ***Network Infrastructure:*** Connectivity with university networks for data synchronization and communication with external systems.

These environmental characteristics outline the hardware and software components that the University Department Information System will interact with, ensuring compatibility and seamless operation.

1.4 Definitions

1. UDIS : University Department Information System
2. RDBMS: Relational Database Management System
3. SRS : Software Requirement Specification

1.5 References

1. IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
2. IEEE software requirements specifications template.

1.6 Overview of Developer Responsibilities

Developers are responsible for designing, implementing, and maintaining the Information System, ensuring its functionality aligns with the specified requirements and addressing any technical issues that may arise.

# General Description

2.1 Product Perspective

***2.1.1 Hardware Interfaces***

* 1. Hard Disk: The database connectivity requires a hardware configuration that is online. This makes it necessary to have a fast database system (such as RDBMS) running on high rpm hard disk permitting complete data redundancy and backup system to support the primary goal of reliability.
  2. The system must interface with the standard output device, keyboard, and mouse to interact with the software.

***2.1.2 Software Interfaces***

1. Any database management software: MongoDB
2. For GUI and user interaction: Chrome Browser.

2.2 Product Functions Overview

**(i) Student Management:**

- Enrolling new students into the system.

- Updating and maintaining student information.

**(ii) Course Registration:**

- Facilitating online course registration for students.

- Managing course availability and prerequisites.

- Handling courses add/drop requests.

- Providing course-related information to students.

**(iii) Grade Tracking:**

- Recording and calculating student grades.

- Managing grade submission by faculty.

- Generating grade card for students.

**(iv) Inventory Management:**

- Keeping track of departmental assets and equipment.

- Managing inventory of educational materials.

- Updating and maintaining stock levels.

- Handling procurement requests for added items.

**(v) Financial Accounting, Research Project Tracking and Faculty Publications:**

- Managing departmental budget and expenses.

- Recording financial transactions.

- Recording details of ongoing research projects.

- Recording and updating faculty publications.

- Managing citation and publication details.

- Providing a platform for faculty to upload their work.

These functions contribute to the effective operation of a University Department Information System, covering key aspects such as student management, academic processes, financial management, and research tracking.

2.3 User Characteristics

1. Users include department secretaries, instructors, and administrative staff. They should have basic computer literacy and familiarity with data entry processes.
2. The intended user of this software need not have specific knowledge as to what is the internal operation of the system. Thus, the end user is at a prominent level of abstraction that allows easier, faster operation and reduces the knowledge requirement of end user.
3. The product is absolutely user friendly, so the intended users can be the naïve users.

2.4 Operating Environment

1. **Hardware Requirements:**
   * Standard computing hardware such as servers, workstations, and personal computers.
   * Sufficient processing power and memory to handle the database management system and computation tasks.
   * Peripheral devices such as printers for generating grade sheets and reports.
2. **Software Requirements:**
   * Database Management System (DBMS) to efficiently store and manage student information, course details, grading data, inventory, and financial records.
   * Application software tailored for student management, course registration, grade tracking, inventory management, and financial accounting.
   * Operating System to provide a stable platform for the execution of the software applications.
3. **Network Infrastructure:**
   * Local Area Network (LAN) to enable seamless communication between different components of the system within the department.
   * Internet connectivity for accessing external databases, services, and for communication with other university systems if necessary.
4. **Security Measures:**
   * User authentication and authorization protocols to ensure that only authorized personnel can access sensitive data.
   * Encryption mechanisms to secure data during transmission over the network.
5. **Training and Support:**
   * Training programs for departmental staff to ensure effective utilization of the information system.
   * Helpdesk or support services for addressing issues, answering queries, and aiding with users.

2.5 General Constraints

***- Schedule:*** The system development should adhere to a specified timeline i.e. it should be completed within the span of one semester.

***- Cost:*** The development should stay within the allocated budget, taking into consideration the financial constraints of the university department.

***- Resources:***

* The project team should include developers, database administrators, system analysts, and other relevant experts to contribute to different facets of system development.
* Adequate infrastructure, including computing hardware, software licenses, and network resources, should be in place to support the development process.

1. Functional Requirements

- Introduction: System should provide a user-friendly interface.

- Input: Capture and validate student details, course registrations, grading information, financial transactions, research projects, and faculty publications.

- Processing: Perform calculations for GPA, CGPA, financial balances, and handle inventory updates.

- Output: Generate grade sheets, financial reports, and provide on-demand student details.

\*\*System Characteristics for University Department Information System:\*\*

3.1 Introduction - User-Friendly Interface

- The University Department Information System should feature an intuitive and user-friendly interface to enhance accessibility for departmental staff.

- Design elements such as filtering, well-organized menus, and a consistent layout contribute to a positive user experience.

- User training programs should be incorporated to familiarize departmental personnel with the system's interface and functionalities.

3.2 Input - Capture and Validate

**3.2.1 Student Details**

- Capture and validate comprehensive student information, including name, address, contact details, and other relevant particulars at the time of admission.

- Implement data validation checks to ensure accuracy and completeness of entered student details.

**3.2.2 Course Registrations**

- Facilitate the input of course registration data by the department secretary, allowing efficient management of enrolled courses for each student.

- Validate course registration inputs to prevent errors and discrepancies.

**3.2.3 Grading Information**

- Enable instructors to input grading information at the end of each semester.

- Implement validation checks for grading data to ensure accuracy in the calculation of Grade Point Averages (GPA) and Cumulative Grade Point Averages (CGPA).

**3.2.4 Financial Transactions**

- Capture financial transactions, including income from grants, consultancy funds, and departmental expenditures on equipment, books, and stationery.

- Validate financial data inputs to maintain accuracy in budget management.

**3.2.5 Research Projects and Faculty Publications**

- Allow the secretary to input information about ongoing research projects and faculty publications.

- Implement validation checks to ensure the completeness and accuracy of research and publication details.

3.3 Processing

**3.3.1 Calculation of GPA and CGPA**

- Implement algorithms to calculate the Grade Point Average (GPA) for each student at the end of the semester.

- Compute the Cumulative Grade Point Average (CGPA) based on the grades earned in all completed courses.

**3.3.2 Financial Balances**

- Process financial transactions to maintain an up-to-date record of income, expenditure, and overall balance.

- Generate financial reports that provide a clear overview of the department's financial status.

**3.3.3 Inventory Updates**

- Process inventory updates to keep track of departmental assets, equipment, and other inventory items.

- Implement mechanisms to manage procurement and maintain accurate inventory levels.

3.4 Output

**3.4.1 Grade Sheets**

- Generate and print grade sheets for each student, summarizing their performance in the respective semester.

- Ensure that grade sheets are accurate and reflective of the calculated GPA and CGPA.

**3.4.2 Financial Reports**

- Generate comprehensive financial reports, including income statements, expenditure summaries, and overall financial balances.

- Provide detailed insights into the department's financial health.

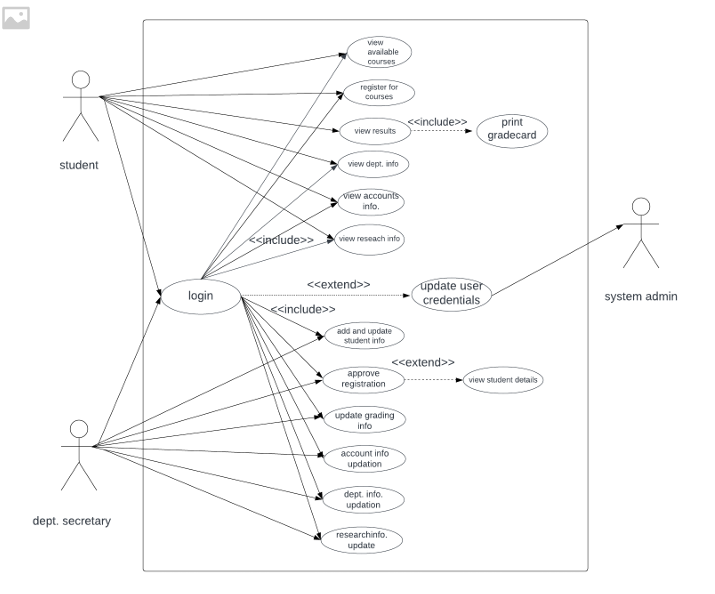
**3.4.3 On-Demand Student Details**

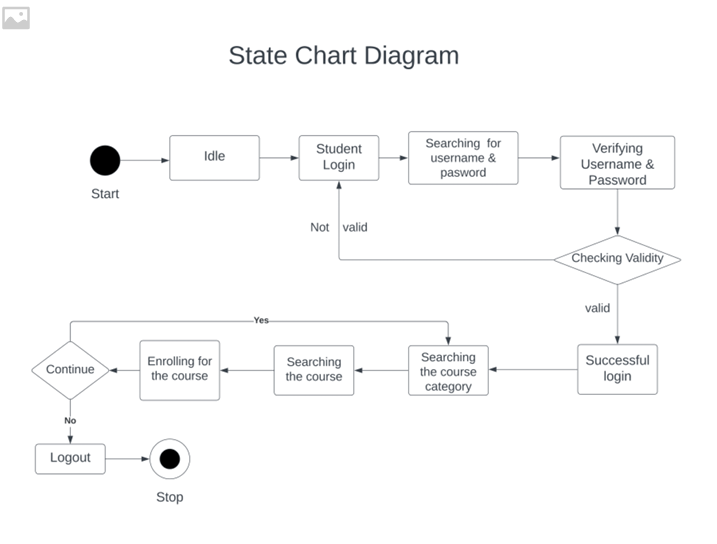
- Support on-demand querying of student details by enabling input of the student's roll number.

- Generate and display up-to-date information, including personal details, course registrations, and academic performance.

For further clarity, the diagrams below can be referenced:

Use Case Diagram





1. External Interface Requirements

4.1 User Interface

The user interface should be designed to be intuitive, ensuring that users can easily input and retrieve data. Clear navigation, well-organized menus, and user-friendly forms are essential to facilitate efficient data management by departmental staff.

4.2 Hardware Interface

The system should be compatible with standard office hardware, such as personal computers and workstations commonly used within the university department. This ensures accessibility and reduces the need for specialized equipment, making the system widely deployable.

4.3 Software Interface

The system should seamlessly integrate with university databases and financial systems, allowing for the exchange of information. Compatibility with standard software interfaces ensures smooth communication, fostering a cohesive information environment within the university. Additionally, support for network interfaces enables secure data transfer between the system and external databases.

4.4 Communication Interface

The system must support secure communication protocols to ensure the confidentiality and integrity of data during transfer between the University Department Information System and external databases. This is essential for protecting sensitive information and maintaining data security standards.

1. Performance Requirements

5.1 Response Time for Queries: < 3 Seconds

The system should respond to user queries within a maximum time of three seconds, ensuring that users can access information promptly. This requirement aims to provide a responsive and efficient user experience, especially during data retrieval processes.

5.2 System Uptime: 99% Availability

The system should be available for use 99% of the time, ensuring minimal downtime. This high availability ensures that departmental staff can rely on the system for their daily tasks without significant disruptions, contributing to operational efficiency.

1. Design Constraints

6.1 Standard Compliance

The system design must adhere to university data privacy and security standards, ensuring the protection of sensitive information. Compliance with established standards is crucial for maintaining the integrity of student and financial data.

6.2 Hardware Limitations

The system should be designed to operate efficiently on standard office computers, considering the limitations and capabilities of typical hardware found within the university department. This constraint ensures that the system remains accessible without requiring extensive hardware upgrades.

6.3 Specific Programming Language

The system development should utilize programming languages that align with university standards. This constraint ensures consistency in software development practices and facilitates future maintenance and updates within the university's technological ecosystem.

1. Other Requirements

7.1 Usability

The software should have a user-friendly interface that is easy to navigate and understand, with clear instructions and help resources.

7.2 Reliability

The software should be reliable, with a low failure rate and minimal downtime, to ensure that shop owner can access inventory data when needed.

7.3 Security

The software should be secure, with appropriate user authentication, access control, and encryption measures to protect sensitive data from unauthorized access.

7.4 Scalability

The software should be designed to handle large volumes of data and support multiple users, with the ability to scale up or down as needed.

7.5 Performance

The software should have high performance, with fast response times, efficient data retrieval and processing, and minimal delays.

7.6 Flexibility

The software should be flexible, with the ability to adapt to changing business requirements, such as new inventory items or regulatory changes.

7.7 Maintainability

The software should be maintainable, with the ability to quickly diagnose and fix bugs, upgrade features, and apply security patches..