INTRODUCTION

This document outlines the scope, objectives, requirements, and structural overview of the proposed University Management System (UMS). The UMS is an integrated software application designed to automate and streamline the administrative, academic, and operational processes of the university, providing efficient data management and access for all stakeholders, including students, faculty, and administrators.

OBJECTIVES

The primary objectives of the University Management System are to:

1. **Improve Operational Efficiency:** Automate manual processes such as registration, grading, and scheduling to reduce administrative workload and errors.
2. **Enhance Data Accuracy and Integrity:** Centralize all academic and administrative data into a secure and single source of truth.
3. **Provide Timely Information Access:** Enable students and administrators to access relevant information (e.g., grades, course status, reports) in real-time.
4. **Support Decision Making:** Generate comprehensive reports and analytical data for administrators to facilitate strategic planning and resource allocation.
5. **Increase Stakeholder Satisfaction:** Offer user-friendly interfaces (UIs) tailored for both Student and Admin roles to improve their experience with university services.

FUNCTIONAL REQUIREMENTS

Functional requirements define what the system must do to satisfy user needs. They are categorized by the primary user roles.

3.1 Student Module

Table 1: Student Module Functional Requirements

ID	Requirement	Description
FR-S01	Course Registration	Students must be able to view available courses and enroll/drop courses during specified registration periods.
FR-S02	Grade Viewing	Students must be able to securely view their final grades, GPA, and academic history.
FR-S03	Profile Management	Students must be able to update personal contact information and change their password.
FR-S04	Fee/Payment Status	Students must be able to view outstanding fees and payment history.

3.2 Admin Module

3.3 Core System Requirements

NON-FUNCTIONAL REQUIREMENTS

Non-functional requirements specify criteria that can be used to judge the operation of a system, rather than specific behaviors.

Table 2: Admin Module Functional Requirements

ID	Requirement	Description
FR-A01	User Management	Admins must be able to create, modify, and deactivate Student and Faculty user accounts.
FR-A02	Course & Catalog Management	Admins must be able to add, edit, and remove courses, departments, and academic programs.
FR-A03	Scheduling	Admins must be able to assign faculty to courses and schedule classrooms.
FR-A04	Transcript Generation	Admins must be able to generate certified academic transcripts and student status reports.

Table 3: Core System Functional Requirements

ID	Requirement	Description
FR-C01	Authentication	The system must require secure login with unique credentials for all users (Student, Admin, Faculty).
FR-C02	Data Backup	The system must perform automated daily backups of all critical student and academic data.
FR-C03	Reporting	The system must generate custom reports on enrollment statistics, course load, and resource utilization.

DATA FLOW DIAGRAMS (DFD)

This section visualizes the information flow within the system at two levels of abstraction: Level 0 (Context Diagram) and Level 1 (Process Decomposition).

5.1 DFD Level 0: Context Diagram

The Level 0 DFD represents the entire University Management System as a single ”black box” process that interacts with external entities. It defines the system’s boundaries.

- **System:** University Management System (Process 0.0)
- **External Entities:**
  - **Student:** Provides login credentials and registration requests; Receives schedules and grades.
  - **Admin:** Provides course data, faculty assignments, and user updates; Receives system reports and operational logs.

5.2 DFD Level 1: Process Decomposition

The Level 1 DFD breaks down the single system node from Level 0 into three main subprocesses and introduces the internal Data Stores.

Table 4: Non-Functional Requirements

ID	Type	Requirement	Description
NFR-P01	Performance	Load Time	The system must load all major pages (e.g., Dashboard, Registration) within 3 seconds under peak load (500 simultaneous users).
NFR-S01	Security	Encryption	All transmitted user data and credentials must be encrypted using HTTPS/TLS.
NFR-S02	Security	Compliance	The system must comply with data privacy regulations (e.g., GDPR/FERPA) regarding student records.
NFR-U01	Usability	Responsiveness	The user interface must be intuitive, fully responsive, and accessible on desktop and mobile devices.
NFR-R01	Reliability	Uptime	The system must achieve 99.9% uptime (excluding scheduled maintenance).
NFR-M01	Maintainability	Documentation	The code must be well-documented and follow established software development standards to facilitate future updates.

Processes

1. **Registration & Enrollment (1.0):** Validates student requests against course availability and limits.
2. **Course & Grade Management (2.0):** Handles the recording and retrieval of academic performance.
3. **System Administration & Reporting (3.0):** Manages user access and generates analytical data.

Data Stores

- **D1: Student Database** (Personal info, credentials, financial status)

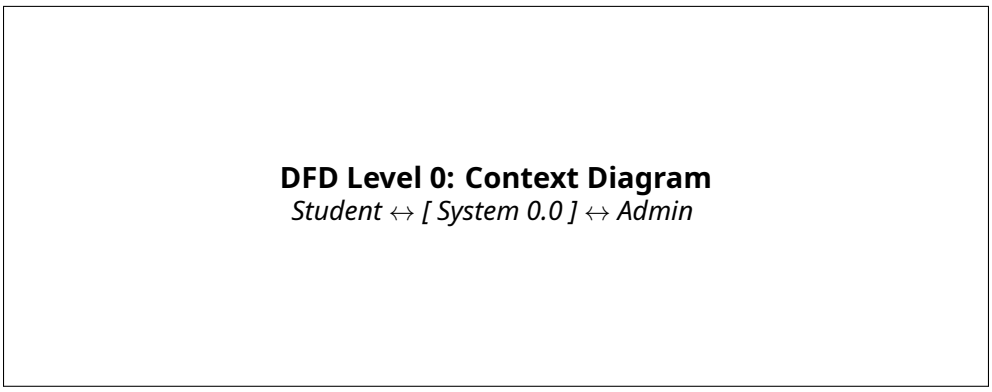

Figure 1: Level 0 Context Diagram showing high-level system boundaries.

- **D2: Course Catalog** (Course details, prerequisites, capacity)
- **D3: Academic Records** (Grades, transcripts, history)

Figure 2: Level 1 DFD illustrating detailed data flows between sub-processes and data stores.