EduSphere - Software Requirements Specification (SRS)

1. Introduction

1.1 Purpose

This document provides a detailed description of the functional and non-functional requirements for EduSphere, a Unified University Management and Student Support Platform.

1.2 Scope

EduSphere is a web-based system integrating academic management, hostel management, attendance monitoring, parent engagement, grievance redressal, and role-based dashboards.

1.3 Intended Audience

- Project Guide / Faculty Evaluators
- Development Team
- Academic Review Committee
- System Administrators

2. Overall Description

2.1 Product Perspective

EduSphere follows a client-server architecture using RESTful APIs and Role-Based Access Control (RBAC).

2.2 User Roles

- Student
- Faculty
- Administrator
- Parent
- Warden

2.3 Operating Environment

- Web Browsers (Chrome, Edge, Firefox)
- Backend Server (Node.js/Django)
- Database (MongoDB/PostgreSQL)
- Docker (Optional)

3. Functional Requirements

3.1 Authentication & Authorization

- JWT-based authentication
- Role-Based Access Control
- Encrypted password storage
- 24-hour faculty login validity

3.2 Student Module

- View attendance and results
- Submit gatepass requests
- Track gatepass status
- View timetable and exam schedule
- Submit grievances via chatbot
- Receive attendance shortage alerts

3.3 Faculty Module

- Mark attendance for multiple lectures at once
- Upload marks and grades
- View performance analytics
- Respond to grievances

3.4 Admin Module

- Manage users and roles
- Create timetable and exam schedules
- Manage notices and events
- Generate attendance reports

3.5 Parent Module

- View ward attendance
- Approve gatepass requests
- View in/out timings
- Schedule mentor meetings

3.6 Warden Module

- Approve final gatepass
- Maintain in/out logs
- Monitor hostel grievances

3.7 Automated Attendance Monitoring

Monthly cron-based attendance calculation triggers notifications for students, faculty, and parents if attendance falls below threshold.

3.8 Notice & Event Management

Admin can create notices and events. All users can view them on dashboard.

4. Non-Functional Requirements

- Security: JWT, RBAC, password hashing
- Performance: Optimized queries and bulk attendance handling
- Usability: Responsive and intuitive UI
- Reliability: Data consistency and workflow validation

5. System Requirements

- Frontend: React.js (Vite)
- Backend: Node.js with Express or Django
- Database: MongoDB or PostgreSQL
- Authentication: JWT

6. Future Scope

- AI-based performance prediction
- NLP-based grievance chatbot
- Biometric hostel entry integration
- Mobile application support

7. Conclusion

EduSphere provides a comprehensive academic ecosystem integrating university management, hostel administration, parent engagement, and automated monitoring into a unified platform.