SOFTWARE REQUIREMENT SPECIFICATION DOCUMENT:

STUDENT RESULT MANAGEMENT SYSTEM :

Introduction:

This document provides a detailed description of the requirements and specifications for the Student Result Management System (SRMS). The SRMS is designed to facilitate efficient management and tracking of student academic records and results within educational institutions of the students.

Purpose:

The primary purpose of this Software Requirements Specification (SRS) document is to:

• Define the scope and objectives of the Student Result Management System. • Clearly outline the functional and non-functional requirements of the software. • Establish a shared understanding between the development team, educational institutions, and users regarding the system's functionality and capabilities. • Serve as a foundation for the design, development, testing, and deployment of the SRMS.

Scope:

The Student Result Management System aims to provide the following key features:

- User authentication and role-based access control for administrators, teachers, and students.
- Management of student profiles, including personal information and enrollment details.
- Recording and storage of course information, including subjects, curriculum, and grading criteria.
- Entry and calculation of student examination scores, assignments, and assessments.
- Generation and distribution of individual student result reports.
- Real-time availability of results to authorized users.

Overall description:

The Student Result Management System (SRMS) is a comprehensive web-based application designed to streamline and enhance the management, processing, and dissemination of student academic results within educational institutions.

The system aims to provide administrators, teachers, and students with an efficient and user-friendly platform to manage and access academic records, assessment scores, and performance analytics.

Assumptions and Constraints:

The development of the Student Result Management System operates under the following assumptions and constraints:

- The system will be developed using modern web technologies, ensuring compatibility with standard web browsers.
- The hardware infrastructure and hosting environment will meet the system's performance and security requirements.
- The system's database will store sensitive student data, and appropriate security measures will be implemented to protect this information.
- The SRMS will adhere to relevant data protection regulations and privacy laws, ensuring the confidentiality of user data.

Features and requirements:

Functional requirements:

- **User Authentication and Access Control:**
- Users must be able to log in with unique usernames and passwords. • The system must provide role-based access control, distinguishing between administrators, teachers, and students.
- Administrators should have access to all system functionalities, while teachers and students have access to their respective features.
- **Student Profile Management:**
- Users should be able to create and update student profiles with personal information, contact details, and enrollment data.
- The system should allow administrators to manage student records, including admissions, transfers, and graduations.
- **Course Management:**
- Teachers and administrators should be able to define and manage courses, including course titles, descriptions, prerequisites, and credit hours.
- The system must support the association of students with courses and maintain accurate enrollment records.
- **Grading and Assessment:**
- Teachers must be able to input and update assessment scores, including exams, quizzes, assignments, and projects.
- The system should calculate and store cumulative grades based on predefined grading criteria.
- **Result Generation and Publication:**
- Teachers and administrators should be able to generate individual student result

reports based on assessment scores.

• Students should be able to access their own result reports through a secure login. •

Progress Tracking and Analytics:

• The system should provide tools for teachers and administrators to track student progress over multiple semesters or terms.

• Statistical analysis and visualizations should be available to identify trends and patterns in student performance.

• **Communication and Notifications:**

• The system should support communication between teachers, administrators, and students through notifications, emails, or messages.

• Automated alerts should be sent to students and administrators for important events, such as publishing of results or registration deadlines.

• **Attendance Tracking:**

• Teachers should be able to mark student attendance for classes and other activities. • The system should maintain attendance records and provide reports on attendance percentages.

• **Course Registration:**

• Students should be able to register for courses within specified registration periods. • The system should prevent registration conflicts and enforce prerequisites when assigning courses to students.

• **Data Management and Reporting:**

• Administrators should have the ability to generate various reports, such as enrollment statistics, class rosters, and GPA distributions.

• The system should allow for exporting data in standard formats for external analysis and reporting.

• **System Administration:**

• Administrators must have the capability to manage user accounts, roles, and permissions.

• The system should provide tools for configuring system settings, including academic terms, grading scales, and assessment criteria.

• **Backup and Data Security:**

• The system should regularly back up data to prevent data loss.

• Data should be stored securely, and appropriate access controls should be in place to protect sensitive information.

• These functional requirements provide an overview of the key features and capabilities of a Student Result Management System. Customize and expand upon these requirements based on the specific needs and goals of your project. **Non-Functional requirements:**

• **Performance:**

• The system must provide fast response times for user interactions, with a maximum page load time of 2 seconds.

• It should handle concurrent user interactions efficiently, supporting a minimum of 100

simultaneous users without significant performance degradation.

• **Scalability:**

• The system should be designed to accommodate future growth in terms of student enrollment and user traffic.

• It should be capable of handling a 20% increase in the number of registered students within a single academic year.

• **Reliability:**

• The system must have a 99.9% uptime, with scheduled maintenance windows communicated in advance.

• It should automatically recover from system failures or crashes within 10 minutes without data loss.

• **Security:**

• User data, including personal information and assessment scores, must be stored securely using encryption.

• Role-based access control should prevent unauthorized users from accessing sensitive information.

• The system should undergo regular security audits and vulnerability assessments. •

Usability:

• The user interface should be intuitive and user-friendly, requiring minimal training for teachers, administrators, and students.

• The system should adhere to accessibility guidelines, allowing users with disabilities to navigate and interact effectively.

• **Compatibility:**

• The system should be compatible with modern web browsers (e.g., Chrome, Firefox, Safari) and responsive across different device types (desktops, tablets, smartphones). • It should support common operating systems used by administrators, teachers, and students. • **Data Integrity:**

• The system should ensure the accuracy and consistency of assessment scores and student records.

• Data validation mechanisms should prevent incorrect or incomplete data entry. •

Data Backup and Recovery:

• The system must perform regular automated data backups to prevent data loss in case of hardware failure or other emergencies.

• Backups should be stored securely and should be easily recoverable. •

Audit Trails:

• The system should maintain an audit trail of important user actions, such as data modifications or access attempts.

• Administrators should be able to review and analyze audit logs.

• **Regulatory Compliance:**

• The system should adhere to relevant data protection regulations and privacy laws, such as GDPR or HIPAA.

• It should provide mechanisms for users to manage their consent for data processing. •

Support and Maintenance:

• The system should have a well-defined maintenance plan, including regular updates, bug fixes, and feature enhancements.

• User support should be available during business hours, with a guaranteed response time of 24 hours for critical issues.

• **Documentation:**

- Comprehensive user documentation and training materials should be provided for administrators, teachers, and students.
- Technical documentation for system maintenance and troubleshooting should also be available.

External interface requirements:

1. User Interfaces:

- The system shall provide a user-friendly web-based interface accessible via standard web browsers (Chrome, Firefox, Safari).

- The user interface shall be responsive and adaptable to different screen sizes, including desktops, tablets, and smartphones.

2. Authentication and Authorization:

- The system shall integrate with an external authentication provider (e.g., LDAP, OAuth) to enable single sign-on for users.

- Role-based access control shall be implemented to manage user permissions and access to system features.

3. External Systems Integration:

- The system shall integrate with the institution's Student Information System (SIS) to synchronize student enrollment data.

- Integration with email servers shall allow sending notifications and alerts to users.

4. Communication Interfaces:

- The system shall use standard email protocols (SMTP) to send automated notifications to students and administrators.

- It shall support RESTful APIs for integration with external applications, enabling data exchange and reporting.

5. Data Import and Export:

- The system shall support importing student data, including profiles and enrollment information, from CSV or Excel files.

- It shall allow administrators to export result reports and other relevant data in standard

formats (PDF, Excel).

6. Reporting Interfaces:

- The system shall provide a built-in reporting module for generating predefined reports, such as class rosters, GPA distributions, and attendance summaries.

- It shall support the export of custom reports in various formats, including PDF and Excel.

7. Payment Gateways (If Applicable):

- If the system handles fee payments, it shall integrate with a secure payment gateway to facilitate online payment processing.

- Payment interfaces shall comply with industry security standards (e.g., PCI DSS) to ensure the protection of payment data.

8. Third-Party Services:

- The system may integrate with third-party services for analytics and monitoring (e.g., Google Analytics) to track user engagement and system performance.

- Social media integration may be considered for sharing system updates and achievements.

9. SMS Notifications (Optional):

- The system may integrate with SMS gateways to send important notifications and alerts to users via text messages.

10. LMS Integration (Learning Management System):

- If applicable, the system shall integrate with the institution's LMS to synchronize course details and student enrollments.

11. Document Management Systems:

- The system shall support integration with document management systems (e.g., Google Drive, Microsoft OneDrive) for storing and sharing documents related to student performance.

Glossary and references:

• **SRMS:** Student Result Management System, the software application being developed to manage student academic results within educational institutions. • **SRS:**

Software Requirements Specification, the document outlining the functional and non-functional requirements of the SRMS.

- **Use Case:** A description of interactions between users and the SRMS, depicting how the system will be used to achieve specific goals.
- **User Role:** A predefined set of permissions and access rights that define the actions a user can perform within the system.
- **Assessment Score:** The numerical or qualitative value assigned to a student's performance in a specific assessment or activity.
- **Grading Criteria:** The predefined standards used to evaluate student performance and calculate final grades.
- **API:** Application Programming Interface, a set of protocols and tools that allows different software applications to communicate with each other.

1]https://ieeexplore.ieee.org/document/9927597/**Student Information Management Decision System Based on Decision Tree Classification Algorithm**

ABINAYA H

STUDENT RESULT MANAGEMENT SYSTEM