

Software Requirements Specification (SRS) Template

Project: Academic Performance Analytics System

Version: 1.0

Authors: Diya D Bhat (Team Lead), Dhanya Prabhu, Delisha Riyona Dsouza, Eshwar R A

Date: 30-08-2025

Status: Draft

Revision history

Version	Date	Authors	Change summary	Approval
1.0	30-08-2025	Diya D Bhat, Dhanya Prabhu, Delisha Riyona Dsouza, Eshwar R A	SRS with diagrams embedded	

Approvals

Role	Name	Signature / Email	Date
Course Coordinator			

Table of Contents

1. Introduction
2. Overall description
3. External interfaces
4. System features (detailed)
5. Non-functional requirements (detailed)
6. Quality attributes & Acceptance tests
7. UML Use-Case Diagram
8. Requirements Traceability Matrix (RTM)

1. Introduction

1.1 Purpose

This Software Requirements Specification (SRS) document defines the requirements for the Academic Performance Analytics System (APAS). The purpose of this system is to provide educational institutions with a centralized platform to collect, analyze, and visualize student academic performance data, including grades, attendance, and assignment submissions. By leveraging predictive analytics and data visualization, the system aims to help faculty members and administrators identify at-risk students early, make data-driven decisions, and improve overall academic outcomes. This document serves as a detailed reference for the development team, quality assurance team, and other stakeholders, ensuring that all functional and non-functional requirements are clearly understood and implemented as intended.

1.2 Scope

The Academic Performance Analytics System will be a web-based platform designed for educational institutions on a small to medium scale. It will collect and organize basic academic data, such as grades, attendance, and assignment submissions, and display this information in simple, role-based dashboards for students, instructors, and administrators.

This project focuses on a few practical features:

- Students can view their academic records in one place and track their progress.
- Instructors can see class-level summaries and identify students who may need help.
- Administrators can access an overview of department performance and download anonymized reports.
- A basic predictive model will highlight students who might be at risk based on their data trends.

The system will include **secure login, role-based access, and basic reporting** features. It demonstrates how data integration, analytics, and visualization can be combined into a working prototype.

1.3 Audience

Developers, QA Engineers, Faculty/Instructors, Administrators, Students

1.4 Definitions

List of acronyms

- **APAS**: Academic Performance Analytics System.

- **SSO:** Single Sign-On, a method for logging in with institutional credentials.
- **RBAC:** Role-Based Access Control, restricting features based on user roles.
- **PII:** Personally Identifiable Information, sensitive student data requiring protection.
- **CSV:** A simple file format used for importing and exporting tabular data.
- **Predictive Analytics:** Using basic statistical or machine learning techniques to estimate a student's likelihood of academic difficulties.
- **Dashboard:** A visual summary of key information.

2. Overall description

2.1 Product perspective

The **Academic Performance Analytics System (APAS)** is being developed as a **standalone web-based platform** designed for use within educational institutions. It is not meant to replace existing Learning Management Systems (LMS) but will integrate with them through simple data imports or APIs to gather grades, attendance, and assignment details. The platform will focus on analytics, dashboards, and visual reports, making it easier for users to understand and act on academic performance trends. For development, the system will use a **web application model** with a browser-based interface accessible on laptops, desktops, and mobile devices. A relational database (e.g., PostgreSQL or MySQL) will store all data, while the backend will handle authentication, analytics, and reporting logic.

2.2 Major product functions (detailed)

- **User Authentication:** Login through institutional credentials or local accounts, with different access levels for students, instructors, and administrators.
- **Data Ingestion:** Import grades, attendance, and assignment records through LMS APIs or CSV uploads.
- **Student Dashboard:** Show personal academic progress, including performance summaries and risk alerts.
- **Instructor Dashboard:** Provide an overview of class performance, attendance trends, and alerts for struggling students.
- **Admin Dashboard:** Display department or institution-wide statistics and support anonymized data exports.
- **Basic Predictive Analytics:** Generate a simple risk score to identify students who may need intervention.
- **Reporting and Exports:** Allow users to download performance summaries in CSV or PDF format.

2.3 User roles and characteristics (expanded)

The system is designed for three main user groups:

- **Students:** Primary users who log in to view their own grades, attendance records, and academic progress. They are expected to be comfortable with basic web applications.
- **Instructors:** Users responsible for managing and monitoring class performance. They need clear summaries, easy navigation, and alerts for at-risk students.
- **Administrators:** Users with access to institution-level reports, system settings, and anonymized data exports. They require secure access and oversight functionality.

2.4 Operating environment

The APAS will run as a web application hosted on a server or cloud platform. Users will access the system via modern web browsers (e.g., Chrome, Firefox, Edge) on desktops, laptops, or mobile devices. The backend will be developed in a widely used framework (e.g., Flask/Django for Python or Node.js), and data will be stored in a secure relational database. The system will require a stable internet connection for live updates.

2.5 Constraints

- Integration limited to basic LMS APIs or CSV data uploads (not full ERP integration).
- Security features implemented at a basic level (authentication, encryption, access control), not full enterprise-level compliance.
- Scaled for small to medium datasets (e.g., a few thousand students).

3. External interface requirements

3.1 User interfaces

Primary UI: Web-based responsive dashboards accessible via desktop, tablet, and mobile.

Role-based dashboards:

- Student dashboard: personal grades, attendance, assignment submissions, risk prediction.
- Instructor dashboard: course/class averages, student drill-downs, alerts.

- Admin dashboard: department/institution-level reports, anonymized exports, system settings.

Accessibility:

- WCAG 2.1 AA compliance (screen-reader support, high-contrast theme, keyboard navigation).
- Scalable text and charts for visually impaired users.

3.2 Hardware interfaces

- Server-side: Runs on institutional servers/cloud (Linux/Windows).
- Client-side: Accessed via standard devices (PC, laptop, tablet, smartphone) with a modern browser.
- Optional peripherals: Printers for generating PDF reports; classroom projectors for dashboard display.

3.3 Software interfaces

- LMS API Integration (Moodle/Canvas/Blackboard): Import student grades, assignments, attendance (JSON over TLS).
- CSV Import/Export: Upload bulk data and export reports in CSV.
- Database: PostgreSQL (for secure academic data storage).
- Authentication: SSO via OAuth2/SAML; fallback local login.
- Reporting Service: Generate PDF reports.
- Analytics Engine: Python ML service for predictive risk scores.

3.4 Communications

- Protocols: All communication over HTTPS (TLS 1.2+).
- API Security: OAuth2 bearer tokens for authentication.
- Notifications: SMTP server for email alerts.
- Offline handling: Retry/backoff logic for failed uploads.

4. System features (detailed)

Each requirement below includes acceptance criteria and a reference test case. IDs follow APAS-F-###.

4.1 Authentication

Description: Authenticate users via institutional SSO/local login and authorize based on role.

Req ID	Requirement	Type	Priority	Source/Shareholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-001	The system shall authenticate	Functional	High	Security / Admin	AC-APAS-F-001: Valid	Requires SSO/OAuth integration

	users via institutional SSO or local login.				credentials - login success; If invalid then denied. Test: TC-Auth-01	
APAS-F-002	The system shall enforce role-based access (student, instructor, admin).	Functional	High	Security/Admin	AC-APAS-F-002: student cannot view other student's data. Test: TC-Auth-02	RBAC Config
APAS-F-003	System shall log all login attempts. (success/failure).	Functional	Medium	Security	AC-APAS-F-003: Audit log contains username, timestamp, result. Test: TC-Auth-03 attempts are blocked and message displayed. Test: TC-Auth-03	Stored in secure logs

4.2 Data Ingestion

Description: Track and manage student academic performance including grades, attendance, and assignment submissions. Ensure data accuracy and timely updates.

Req ID	Requirement (shall...)	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-010	The system shall ingest grades, attendance	Functional	High	Faculty	AC-APAS-F-010: Valid API - data stored. Test:	Requires API token

	ce, and assignments via API.				TC-Ingest-01	
APAS-F-011	The system shall allow CSV upload as fallback.	Functional	Medium	Faculty	AC-APAS-F-011: Valid CSV processed, invalid rejected. Test: TC-Ingest-02	Validation required

4.3 Dashboards

Description: Visualize performance for different roles.

Req ID	Requirement (shall...)	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-020	Instructor dashboard shall show course KPIs (avg grade, pass %, attendance).	Functional	High	Faculty	AC-APAS-F-020: Dashboard matches computed dataset. Test: TC-Dash-01	Requires processed data
APAS-F-021	Student dashboard shall show personal performance timeline and risk score.	Functional	High	Students	AC-APAS-F-021: Data matches student record. Test: TC-Dash-02	Privacy enforced

APAS-F-022	Admin dashboard shall show aggregated departmental reports.	Functional	Medium	Admin	AC-APAS-F-022: Data aggregated across courses. Test: TC-Dash-03	Requires admin role
------------	---	------------	--------	-------	---	---------------------

4.4 Predictive Analysis

Description: ML-driven predictions for student risk.

Req ID	Requirement (shall...)	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-030	The system shall calculate risk-of-failure score (0–1) per student.	Functional	High	Faculty	AC-APAS-F-030: Risk computed on test dataset. Test: TC-ML-01	Logistic regression baseline
APAS-F-031	The system shall allow configurable thresholds for risk alerts.	Functional	Medium	Admin/Faculty	AC-APAS-F-031: Alerts trigger above threshold. Test: TC-Alert-01	Configurable in UI
APAS-F-032	The system shall support retraining ML models	Functional	Medium	Analyst	AC-APAS-F-032: Retrain job produces updated model.	Offline training pipeline

	with new data.				Test: TC-ML-02	
--	----------------	--	--	--	-------------------	--

4.5 Reporting & Exports

Description: Generate reports for stakeholders.

Req ID	Requirement (shall...)	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-040	The system shall export dashboards to PDF and CSV.	Functional	Medium	Faculty/Admin	AC-APAS-F-040: Export matches UI data. Test: TC-Report-01	Requires export service
APAS-F-041	The system shall provide anonymized exports for research.	Functional	Medium	Admin/Analyst	AC-APAS-F-041: Export removes PII. Test: TC-Export-01	Hash student IDs

4.6 Notifications

Description: Provide a technician menu for diagnostics, remote monitoring API integration, secure admin authentication, and encrypted log shipping.

Req ID	Requirement (shall...)	Type	Priority	Source/Stakeholder	Acceptance criteria / Test case ref	Comments / Dependencies
APAS-F-050	The system shall	Functional	High	Faculty	AC-APAS-F-050: Alert sent	SMTP config required

	send email alerts for at-risk students.				when risk > threshold. Test: TC-Alert-01	
APAS-F-051	The system shall support in-app notifications in the dashboard.	Functional	Medium	Students /Faculty	AC-APAS-F-051: Notification visible in dashboard. Test: TC-Alert-02	UI update required

5. Non-functional requirements (detailed)

NFRs below are measurable and tied to test plans. IDs APAS-NF-####.

Req ID	Requirement	Category	Priority	Acceptance criteria / Measurement
APAS-NF-001	Dashboard & UI must respond within 2 seconds for 90% of requests (≤ 100 users).	Performance	High	Load test with 100 users shows 90th percentile $\leq 2s$.
APAS-NF-002	Nightly analytics batch jobs must be completed within 2 hours for 50k students.	Performance / Throughput	High	Batch run observed $\leq 2h$ with a sample dataset.
APAS-NF-003	System availability shall be 99.5% monthly (excluding scheduled maintenance).	Reliability	High	Monitoring reports uptime $\geq 99.5\%$.

APAS-N F-004	System shall scale up to 500 concurrent users with graceful degradation.	Scalability	Medium	Load test confirms stable performance with 500 users.
APAS-N F-005	All student data must be encrypted at rest and in transit (TLS 1.2+).	Security	High	Verified DB/storage encryption + TLS enabled.

5.1. Security

5.1.1 Security Objectives

- Protect confidentiality of student grades and academic records.
- Ensure integrity of analytics and reports.
- Provide secure authentication and authorization for users (students, faculty, admins)
- Ensure availability of the system to authorized users with minimal downtime.
- Provide accountability through secure audit trails of all critical actions.

5.1.2 Security Requirements

Req ID	Requirement (shall...)	Type	Priority	Acceptance criteria / Test case ref
PRJ-SR-001	TLS 1.2+ mandatory for all network connections.	Security	High	Verified via penetration test; no plain HTTP endpoints. (TC-Sec-01)
PRJ-SR-002	All student data shall be encrypted at rest (AES-256 or equivalent).	Data Security	High	Database/storage encryption enabled; files unreadable without keys. (TC-Sec-02)

PRJ-SR-003	The system shall enforce secure authentication (SSO/OAuth2 or username+MFA) with RBAC	Access Control	High	Unauthorized access attempts blocked; role-based restrictions tested. (TC-Auth-01)
PRJ-SR-004	User sessions shall automatically expire after 15 minutes of inactivity.	Access Control	Medium	Verified by test case – inactive user is logged out automatically.
PRJ-SR-005	Data exports (CSV/PDF) shall be restricted to authorized roles and logged.	Data Protection	Medium	Unauthorized export attempts denied; authorized exports logged. (TC-Export-01)

6. Quality attributes & Acceptance tests

Exit criteria for acceptance:

- All high-priority functional requirements implemented and verified
- No critical NFR failures, and RTM shows all test cases passed.
- No open severity-1 defects at delivery.
- Requirements Traceability Matrix (RTM) must show complete coverage: every FR and NFR has at least one acceptance test.

Acceptance test suites:

- Authentication - Verify login with valid and invalid credentials.
- Dashboard rendering and Usability Tests - Measure dashboard load time with 100 users, Ensure <2 sec response for 95% requests.
- Data Ingestion & Processing Tests - Upload student grades/attendance via CSV and API.
- Predictive model correctness - Compare predictions against known outcomes.
- Load tests - Verify response times degrade gracefully without system crash.
- Security and Penetration tests
- Unit test coverage.

7. System models and diagrams

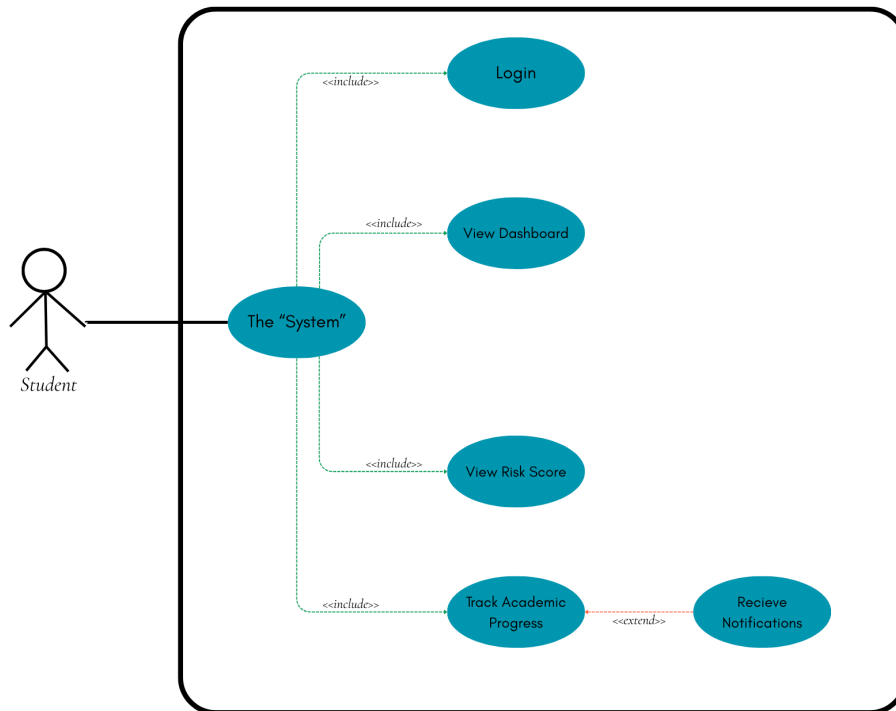
7.1 Student Record System

- Students can view their academic records in one place and track their progress. The SRS specifies that the student dashboard shows a personal performance timeline and risk-of-failure score. Key use cases for the Student actor include:
- **Login:** authenticate via institutional SSO or local credentials (APAS-F-001).
- **View Dashboard:** see consolidated grades, attendance, and assignment data (APAS-F-021).
- **Track Academic Progress:** monitor cumulative performance trends (APAS-F-021).

- **View Risk Score:** check the predictive risk-of-failure score (APAS-F-021).
- **Receive Notifications:** get email and in-app alerts if flagged at-risk (APAS-F-050, APAS-F-051).

These correspond to requirements in the SRS (e.g. APAS-F-021 for the student dashboard with risk score, and APAS-F-050/051 for notifications).

Student Record System (SRS)



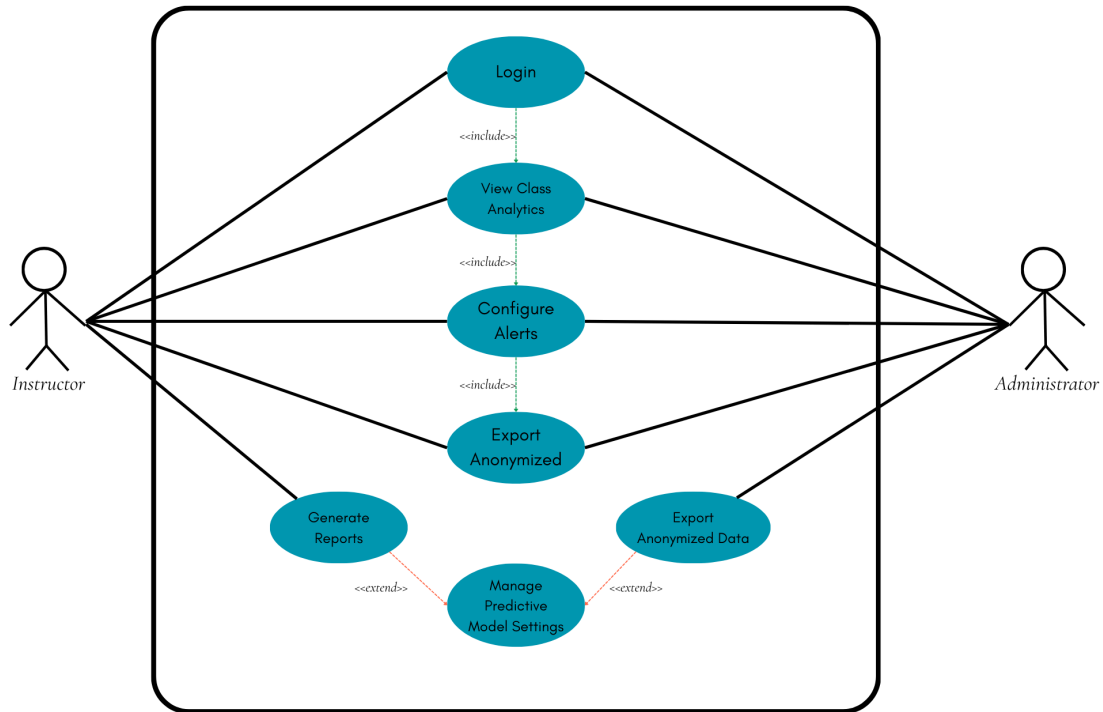
7.2 Instructor and Administrator Use-Case Diagram

Instructors and administrators have overlapping but distinct use cases. The SRS notes that instructors see class-level summaries and at-risk alerts, while administrators view department-wide statistics and can export data. Typical use cases include:

- Login: access the system with role-based credentials (APAS-F-001).
- View Class Analytics: for instructors, view a dashboard of course KPIs (APAS-F-020).
- View Institution Analytics: for admins, view aggregate department/institution dashboards (APAS-F-022).
- Configure Alerts: set risk-alert thresholds (APAS-F-031).
- Generate Reports: produce performance reports (APAS-F-040).
- Export Anonymised Data: allow admins to download anonymised student data (APAS-F-041).
- Manage Predictive Model Settings – adjust retraining or model parameters (APAS-F-032).

These use cases map to SRS features (e.g. APAS-F-020 for the instructor dashboard, APAS-F-041 for anonymised exports).

Student Record System (SRS)



8. Requirements Traceability Matrix (RTM)

Req ID	Requirement short	Section ref / Design Spec	Module	Test case(s)	Status (N/P/A)	Comments
APAS-F-001	User login authentication	4.1 / DS-Auth-01	AuthModule	TC-Auth-01	N	–
APAS-F-002	Role-based access control	4.1 / DS-Auth-02	AuthModule	TC-Auth-02	N	–
APAS-F-003	Login audit logging	4.1 / DS-Auth-03	AuthModule	TC-Auth-03	N	–
APAS-F-010	Ingest data via API	4.2 / DS-Ingest-01	DataIngest Module	TC-Ingest-01	N	Requires API token
APAS-F-011	Ingest data via CSV	4.2 / DS-Ingest-02	DataIngest Module	TC-Ingest-02	N	CSV upload fallback

Req ID	Requirement short	Section ref / Design Spec	Module	Test case(s)	Status (N/P/A)	Comments
APAS-F-020	Instructor dashboard	4.3 / DS-Dashboard-01	InstructorDashboard	TC-Dash-01	N	Course KPI computations
APAS-F-021	Student dashboard	4.3 / DS-Dashboard-02	StudentDashboard	TC-Dash-02	N	Performance timeline
APAS-F-022	Admin dashboard	4.3 / DS-Dashboard-03	AdminDashboard	TC-Dash-03	N	Aggregated reports
APAS-F-030	Risk score computation	4.4 / DS-Predictive-01	PredictiveModule	TC-ML-01	N	Logistic regression model
APAS-F-031	Configurable alert thresholds	4.4 / DS-Predictive-02	PredictiveModule	TC-Alert-01	N	UI-configurable
APAS-F-032	Model retraining support	4.4 / DS-Predictive-03	PredictiveModule	TC-ML-02	N	Offline retraining
APAS-F-040	Export dashboard (CSV/PDF)	4.5 / DS-Reporting-01	ReportingModule	TC-Report-01	N	Export service
APAS-F-041	Anonymized data export	4.5 / DS-Reporting-02	ReportingModule	TC-Export-01	N	Hash student IDs
APAS-F-050	Email alerts for at-risk	4.6 / DS-Notification-01	NotificationModule	TC-Alert-01	N	SMTP configured
APAS-F-051	In-app risk notifications	4.6 / DS-Notification-02	NotificationModule	TC-Alert-02	N	Dashboard inbox update
APAS-N F-001	UI response time <2s (90%)	5.1 / DS-UI-Perf-01	WebUI	TC-Perf-01	N	Performance target
APAS-N F-002	Batch processing ≤2h (50k)	5.1 / DS-Batch-Perf-01	BatchProcessor	TC-Perf-02	N	Overnight analytics job
APAS-N F-003	99.5% availability	5.1 / DS-Reliab-01	System	TC-Reliab-01	N	Uptime monitoring
APAS-N F-004	Scale to 500 users gracefully	5.1 / DS-Scalability-01	System	TC-Scal-01	N	Load testing

Req ID	Requirement short	Section ref / Design Spec	Module	Test case(s)	Status (N/P/A)	Comments
APAS-N F-005	Encrypt data (rest & TLS)	5.1 / DS-Security-01	SecurityModule	TC-Sec-01	N	AES-256, TLS1.2+