

Software Architecture and Design (SAD) Document

NAME: ROHAN SURESH, ROHANN RAHUL SHAH, RISHABH JAWAGAL, NIKHILESH VENKAT

SRN: PES1UG23AM240, PES1UG23AM241, PES1UG23AM235, PES1UG23AM192

AIML 5th SEM D SECTION

DATE: 20-09-2025

1. Introduction

1.1 Purpose

This document details the architecture and design of the Peer Rating and Feedback System. It is intended to guide the development, quality assurance, and maintenance of the system by outlining its core components, structure, and design decisions.

1.2 Scope

The system's scope includes user account management, project and team setup, survey creation and distribution, peer rating submission, and the generation of aggregated score and feedback reports in PDF format.

1.3 Audience

This document is for the

Developers, QA Engineers, Project Managers, Instructors, System Administrators, and end-users (students/team members) of the Peer Rating and Feedback System.

1.4 Definitions

- **PRS:** Peer Rating System
- **UI:** User Interface
- **API:** Application Programming Interface
- **PDF:** Portable Document Format
- **PII:** Personally Identifiable Information
- **RBAC:** Role-Based Access Control
- **HTTPS:** Hypertext Transfer Protocol Secure

2. Document Overview

This document provides architectural deliverables, including a component diagram, threat models, and API designs for the Peer Rating and Feedback System.

3. Architecture

3.1 Goals & Constraints

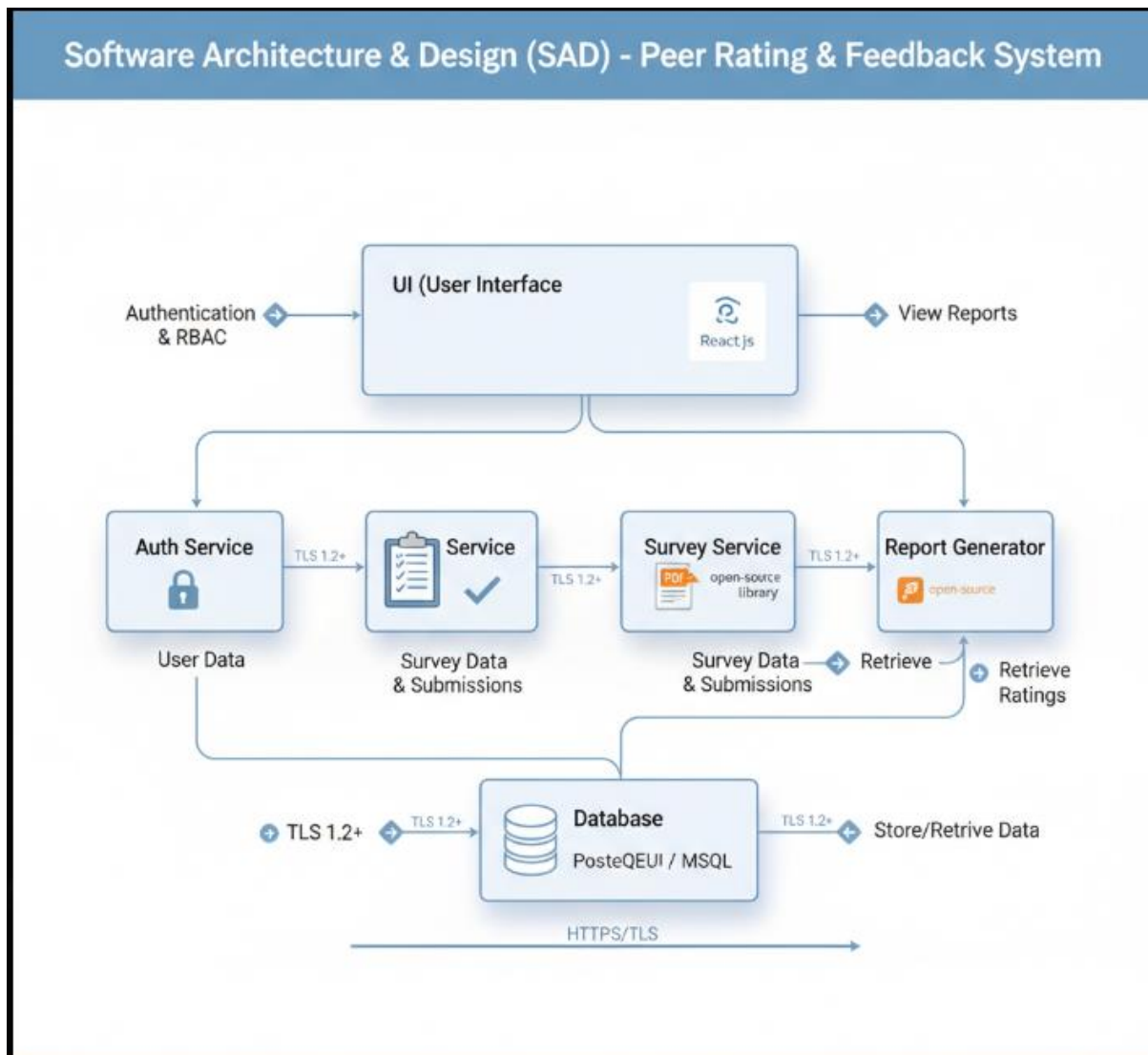
- **Goals:** The system aims to be **secure, reliable, scalable, and user-friendly**.
- **Constraints:** It must comply with **data privacy regulations** and securely handle **PII**. All data in transit must be protected by

HTTPS/TLS encryption. The PDF export must use a designated open-source library.

3.2 Stakeholders & Concerns

- **Students:** Concerned with usability and the confidentiality of their feedback reports.
- **Instructors:** Concerned with the accuracy of reports and the functionality of the dashboard.
- **Admins:** Concerned with security and system reliability.
- **Regulators:** Concerned with compliance with data privacy standards.
- **Developers:** Concerned with the modularity of the system's design.

3.3 Component Diagram



3.4 Component Descriptions

- **UI:** A responsive, web-based user interface that provides distinct dashboards and functionalities for different user roles (Administrator, Instructor, Student).
- **Auth Service:** Manages user authentication and enforces **Role-Based Access Control (RBAC)** to restrict features based on user roles.
- **Survey Service:** Manages the creation and distribution of surveys and the submission of ratings and feedback. It also handles the enforcement of submission deadlines.
- **Report Generator:** Aggregates peer ratings, calculates scores, and creates individual, anonymized feedback reports in PDF format.
- **Database:** Stores all system data, including user accounts, projects, teams, surveys, and submitted ratings.

3.5 Chosen Architecture Pattern and Rationale

The system will use a **layered architecture** with modular services. This pattern was chosen to provide a clear separation of concerns, which makes the system easier to maintain and scale.

3.6 Technology Stack & Data Stores

- **Frontend:** React.js
- **Backend:** Node.js/Express
- **Database:** PostgreSQL or MySQL
- **PDF Generation:** A designated open-source library
- **Encryption:** TLS 1.2+ for all network communication

3.7 Risks & Mitigations

Risk	Mitigation
Data leaks of PII	Use TLS encryption, secure data handling, and RBAC to restrict access.
Survey manipulation	Implement immutable logs and audit trails to prevent and track changes after submission.
Unauthorized access	Use strong authentication, RBAC, and session timeouts.

3.8 Traceability to Requirements

- **User registration (PRS-F-001):** Handled by the **Auth Service**.
- **Survey creation (PRS-F-020):** Handled by the **Survey Service**.
- **Report generation (PRS-F-032):** Handled by the **Report Generator** component.

3.9 Security Architecture

Threats are addressed using the **STRIDE** model:

- **Spoofing**: Mitigated by **RBAC** and strong authentication.
 - **Tampering**: Mitigated by using hashed/salted passwords, immutable logs, and audit trails.
 - **Information Disclosure**: Mitigated by enforcing **HTTPS/TLS** for all data in transit.
 - **Denial of Service (DoS)**: Mitigated by implementing rate limiting.
 - **Elevation of Privilege**: Mitigated by strict role validation.
-

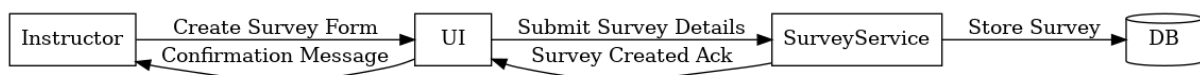
4. Design

4.1 Design Overview

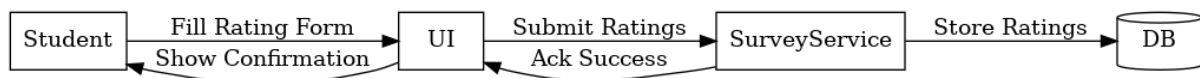
The system is designed with a layered architecture, which ensures a clear separation of concerns. This approach allows for easier maintenance and development, as each module is independent.

4.2 UML Sequence Diagrams

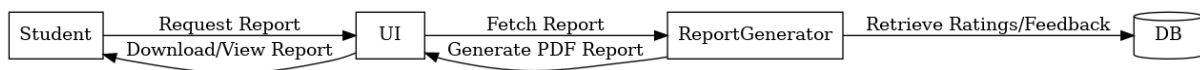
Sequence for survey creation:



Sequence for rating system:



Sequence for report generation:



4.3 API Design

- **Endpoint**: POST /api/survey/submit
- **Description**: Allows a user to submit their peer ratings and feedback for a specific survey.
- **Request**: A JSON payload containing the userId, surveyId, and ratings.
- **Response**: A JSON response indicating the status and a descriptive message.

4.4 Error Handling, Logging & Monitoring

The system will use **centralized error handling** to ensure consistency. Logs will be maintained, but they will not contain sensitive information. Monitoring will focus on key metrics such as submission success rates and system uptime.

4.5 UX Design

The user interface will be **responsive** and feature **role-based dashboards**. The design will conform to **Web Content Accessibility Guidelines (WCAG) 2.1 Level AA**.

4.6 Open Issues & Next Steps

- **Open Issues:** Integration with external LMS (Learning Management Systems).
 - **Next Steps:** Future enhancements may include biometric authentication and AI-based rating analysis.
-

5. Appendices

- **Glossary:** Definitions for PRS, UI, API, PDF, PII, RBAC, HTTPS.
- **References:** Not specified in source documents.
- **Tools:** **draw.io** for diagrams, **Swagger** for API design, and **PlantUML** for UML diagrams.