

Accreditation and data management assistant

CAPSTONE PROJECT

By: Vidya Sinha 92200133021
Dhruvi Patel 92200133029

Table Content

- Introduction
- Problem Statement
- Objective
- Technical solution
- System design
- Outcome
- Project demo

INTRODUCTION

- Bodies like NBA, NAAC, Qos, NIRF, COE rank institutions on multiple criterias.
- As of now, all the work done till now is currently done manually, from data entry , data collection and scoring them for self assessment which eventually lead to errors and inconsistency and especially time taking and inefficiency.
- So there was a need for an automated accreditation and data management system which does this and simply the whole process which also be more transparent.

Problem Statement

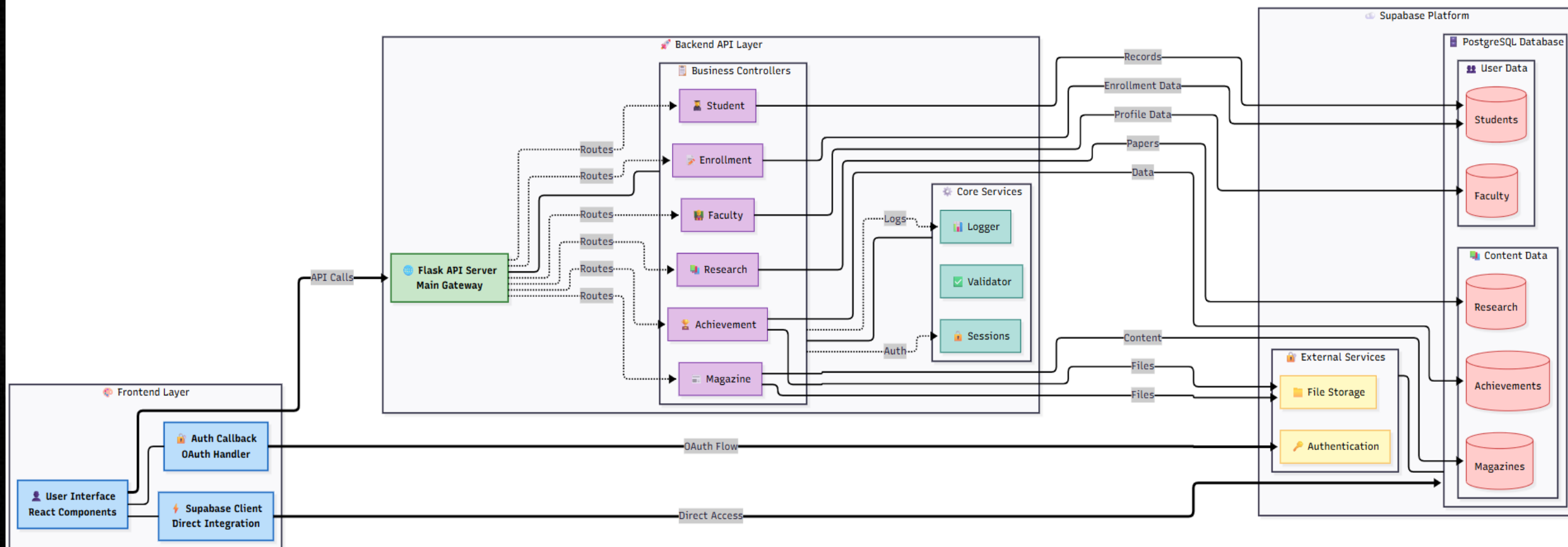


- Faculties used to rely on spreadsheets and manual data entry to prepare the reports which again creates challenges in accuracy since there is large set of data which comprises of many students + faculties details which needs to be arranged in a way the criterias are listed and have to do this same repetitive task every time an accreditation bodies visits institution.
- Even the accreditation bodies struggle to verify all the manual data records across different criterias. This creates a need for a centralized, automated solution that can calculate accreditation scores, manage institutional data securely, and generate reliable reports.

Objective

- 1, To develop an automated system that calculates scores self assessment by checking all the criteria for both accreditation bodies and institution
 2. To design a centralized database for secure storage and easy retrieval of institutional data for all past years.
 3. To provide standardized reporting dashboards for institutions(HOD, Admin, dean) and accreditation bodies(naac, nba etc). 4.To ensure data privacy and secure access by adopting encryption and user authentication mechanisms and role based access.
-

System Architecture



Outcome

Accreditation and Data Management Assistant successfully addresses the persistent challenge faced by higher-education institutions and accreditation bodies in handling complex data and calculating scores efficiently and accurately.

By developing an automated, centralized platform, the system replaces reliance on repetitive manual work, reducing the likelihood of human error, improving data reliability, and making complex compliance reporting significantly simpler and more transparent for all stakeholders.

The current outcome delivers key innovations, including support for multiple accreditation body formats (like NBA and NAAC), real-time score computation, and the provision of live verification dashboards for accreditation officers, along with robust security measures such as Role-Based Access Control

Future scope

For future scope, the system is designed for extensibility and long-term scalability. The modular and cloud-native architecture, leveraging Flask Blueprints and SQLAlchemy ORM, ensures modular code organization, making it easier to maintain the code and integrate new features without disrupting existing functionality.

The RESTful API architecture also supports easy integration with other systems or third-party services. Future scalability strategies are already considered for large data growth, including using Read replicas for high-volume read operations (like dashboard queries) and Partitioning/Sharding if multi-year institutional data tables grow very large.

Additionally, the system can easily switch to stronger authentication methods like JWT tokens or Supabase Auth to prepare for secure, multi-user use as the institution grows.

Technical Solution

Frontend : Enforces type safety, reduces runtime errors, and ensures responsive, modular UI (forms, dashboards, file uploads).

Backend (Flask Framework): Lightweight API framework with **Blueprint pattern** for modular routing and clear separation of concerns

Database & Storage: PostgreSQL for relational data, Supabase for file storage & authentication, SQLAlchemy for clean and maintainable queries.

Authentication & Security: Session-based authentication, role-based access control, token validation, input sanitization, and content-type validation for uploads.

Enough talking

**NOW LETS CHECK
PROJECT DEMO**