RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

PERSONALIZED LEARNING AND STUDENT RETENTION

Database Management Systems

(CD252IA)

V SEMESTER

PROJECT REPORT

Submitted by

Name 1: HRUTHIK K K **USN 1: 1RV22CS071**

Name 2: KISHAN KUMAR S D **USN 2: 1RV22CS087**

Name 3: HEMANTH GOWDA C **USN 3: 1RV22CS067**

Under the guidance of

Dr. Pratiba D

Assistant Professor

Bachelor of Engineering in

Computer Science and Engineering

2024-2025

RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the **Database Management Systems Project Work** titled "**PERSONALIZED LEARNING AND STUDENT RETENTION**" is carried out by **HRUTHIK K K (1RV22CS071), KISHAN KUMAR S D (1RV22CS087) and HEMANTH GOWDA C (1RV22CS067)** who are bonafide students of RV College of Engineering, Bengaluru, in partial fulfillment for the **Internal Assessment of Course: Database Management Systems** during the year 2024-2025. It is certified that all corrections/suggestions indicated for the Internal Assessment have been incorporated in the report.

Dr. Pratiba D
Faculty Incharge,
Department of CSE,
R.V.C.E., Bengaluru –59

Dr. Shanta Rangaswamy
Head of Department,
Department of CSE,
R.V.C.E., Bengaluru–59

External Viva

Name of Examiners	**Signature with Date**
1	
2	

RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **HRUTHIK K K (1RV22CS071), KISHAN KUMAR S D (1RV22CS087) and HEMANTH GOWDA C (1RV22CS067)** the students of Fifth Semester B.E., Department of Computer Science and Engineering, R V College of Engineering, Bengaluru hereby declare that project titled **"PERSONALIZED LEARNING AND STUDENT RETENTION"** has been carried out by us and submitted in partial fulfillment for the **Internal Assessment of the Course: INTRODUCTION TO DATABASE SYSTEMS (CD252IA)** during the academic year 2024-2025. We also declare that matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma.

Place: Bengaluru

Date: 19-01-2025

Name	Signature
1. HRUTHIK K K (1RV22CS071)	
2. KISHAN KUMAR S D (1RV22CS087)	
3. HEMANTH GOWDA C (1RV22CS067)	

ACKNOWLEDGEMENT

We are indebted to our Dr.Pratiba D Assistant Professor**,Dept of CSE, RV College of Engineering** for her wholehearted support, suggestions and invaluable advice throughout our project work and helped in the preparation of this report.

Our sincere thanks to **Dr. Shanta Rangaswamy** Professor and Head, Department of Computer Science and Engineering, RVCE for her support and encouragement.

We express sincere gratitude to our beloved Principal, **Dr. K. N. Subramanya** for his appreciation towards this project work.

We thank all the **teaching staff and technical staff** of Computer Science and Engineering department, RVCE for their help.

Lastly, we take this opportunity to thank our **family** members and **friends** who provided all the backup support throughout the project work.

ABSTRACT

In the **Personalized Learning and Student Retention** project, we developed a comprehensive full-stack web application utilizing MERN (MongoDB, Express.js, React, and Node.js), MySQL, and Flask. This innovative platform integrates a machine learning model to accurately predict the dropout risk of students, thereby enabling timely interventions.

Our platform offers a suite of diversified tools to enhance the learning experience for students. It features centralized database management that allows school administrators, coordinators, and teachers to monitor student data effectively. Additionally, the application facilitates one-to-one communication between students and teachers, fostering personalized learning experiences and support.

This project represents a significant advancement in educational technology, aimed at improving student retention rates and promoting a more tailored and engaging learning environment. By leveraging the strengths of modern web technologies and machine learning, we provide an efficient and effective solution for the challenges faced in educational institutions.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In today's educational landscape, where the personalized learning experience and student retention are pivotal, the development of an efficient and user-friendly learning management system is essential. The **Personalized Learning and Student Retention** project introduces a sophisticated full-stack web application that leverages the power of modern technologies to enhance educational outcomes. This introduction delves into the significance of such a system, emphasizing its role in predicting dropout risks, personalizing learning paths, and fostering effective communication between students and teachers. By integrating advanced machine learning models, this project offers the potential to revolutionize the way educational institutions address student retention challenges, ultimately leading to improved academic success and engagement.

Moreover, a well-designed personalized learning system not only identifies at-risk students but also provides tailored interventions to support their educational journey. The centralized database management ensures that school administrators, coordinators, and teachers can efficiently monitor and manage student data, adhering to quality standards and regulatory guidelines. Additionally, this application facilitates one-to-one communication between students and teachers, promoting a supportive and interactive learning environment. By harnessing the strengths of MERN stack, MySQL, and Flask, this project exemplifies the transformative impact of technology in education, aiming to create a more responsive and inclusive learning experience.

The **Personalized Learning and Student Retention** system serves as a comprehensive solution that addresses various aspects of educational management. It encompasses functionalities such as maintaining detailed student records, providing tools to enhance the learning experience, and integrating machine learning models to predict dropout risks. Furthermore, the system captures critical academic data, enabling a holistic and integrated approach to student retention and personalized learning.

Chapter 2

Software Requirement specification

A Software Requirements Specification (SRS) serves as a comprehensive document outlining the development details for the Personalized Learning and Student Retention project. This document articulates both functional and non-functional requirements, accompanied by a set of use cases to describe user interactions, ensuring seamless functionality of the system. By defining the system's requirements early in the development process, the SRS minimizes the need for later redesign. In the context of the Personalized Learning and Student Retention project, the SRS addresses specific hardware and software requirements, as well as delineates functional and non-functional aspects critical to the success of the project.

2.1 Hardware Requirements

- **Processor:** Multi-core processor (Intel Core i5 or equivalent).
- **RAM:** 8 GB or higher for better performance.
- **Storage:** SSD for faster data access.

2.2 Software Requirements

1. **Operating System Compatibility:** The system should be compatible with major operating systems, including Windows, Linux, and macOS, to ensure widespread accessibility for users.
2. **Web Framework:** Utilize a stable and scalable web framework for the user interface.
3. **Backend:** Employ JavaScript as the primary backend language, ensuring seamless communication between various components of the system.
4. **Database Management System:** Implement a SQL-based database system to manage structured data efficiently, enabling secure and reliable storage of critical educational information.
5. **NoSQL Database:** Integrate MongoDB to capture and organize unstructured data, specifically for storing and managing personalized learning analytics.

2.3 Functional Requirements

1. **Student Management**
    - Allow registration of students with relevant details (student_id, contact, etc.).
    - Enable administrators to manage student accounts (create, update, delete).
2. **Teacher Management**
    - Allow registration of teachers with relevant details (teacher_id, contact, etc.).
    - Enable administrators to manage teacher accounts (create, update, delete).
3. **Dropout Risk Prediction**
    - Implement a machine learning model to predict the dropout risk of students based on historical data and various risk factors.
    - Display risk scores and suggest interventions to support at-risk students.
4. **Personalized Learning Tools**
    - Provide tools for creating personalized learning paths for students based on their learning preferences, performance, and needs.
    - Offer resources and activities tailored to individual student profiles.
5. **Communication System**
    - Facilitate one-to-one communication between students and teachers, including messaging and video conferencing capabilities.
    - Enable teachers to provide personalized feedback and support to students.
6. **Centralized Database Management**
    - Manage student data, including academic records, attendance, and performance metrics.
    - Ensure secure and efficient storage and retrieval of data.
7. **NoSQL (MongoDB)**
    - Store and manage unstructured data related to student engagement and learning analytics.
    - Capture diagnostic information and feedback from personalized learning activities.

Chapter 3

Entity Relationship Diagram

An Entity Relationship (ER) Diagram is a visual representation similar to a flowchart, depicting the relationships between "entities" like people, objects, or concepts within a system. ER Diagrams are primarily utilized in designing or troubleshooting relational databases across various domains such as software engineering, business information systems, education, and research. Commonly referred to as ERDs or ER Models, these diagrams employ a set of symbols, including rectangles, diamonds, ovals, and connecting lines, to illustrate the connections between entities, relationships, and their attributes. They follow a structured format resembling grammar, with entities represented as nouns and relationships as verbs.

1.Entities and Attributes:

Admin

- Represents the administrators responsible for managing the system.
- Related to designing courses and supervising departments.

Course

- Represents the various courses offered in the learning system.
- Related to Admin (designed by) and Teachers (taught by).

Teacher

- Represents the teachers responsible for teaching courses.
- Related to departments (assigned to) and courses (teaches).

Department

- Represents the organizational units within the system (e.g., academic departments).
- Related to Admin (supervised by) and both Students and Teachers (belongs to).

Student

- Represents the students enrolled in the learning system.
- Related to departments (belongs to), courses (enrolled in), and logs (records).

Logs

- Represents the tracking of student performance and learning tools used in the system.
- Related to students (records attendance and marks).

2.Relationships:

Admin Designs Course:

- Relationship: 1-to-N (One Admin designs multiple Courses).

Admin Supervises Department:

- Relationship: 1-to-N (One Admin supervises multiple Departments).

Teacher Teaches Course:

- Relationship: M-to-N (A Teacher can teach multiple Courses, and a Course can be taught by multiple Teachers).

Teacher Assigned to Department:

- Relationship: 1-to-N (A Teacher belongs to one Department, but a Department can have many Teachers).

Student Belongs to Department:

- Relationship: 1-to-N (A Student belongs to one Department, but a Department can have many Students).

Student Enrolls in Course (Derived):

- Relationship: N-to-M (Students are enrolled in multiple Courses).

Logs (Records Student's Performance):

- Relationship: 1-to-N (One Student has multiple Logs).

3.Attributes:

- **Admin:** admin_id, name, email, school_name.
- **Course:** code, name, sessions.
- **Teacher:**teacher_id, name, email.
- **Department:**name.
- **Student:**student_id, name,gender,email
- **Logs:** marks, attendance,behaviuoral score

4.Flow of Information:

A. Student Registration and Management:

- Students register on the platform, providing necessary details such as student_id, contact, and enrollment_date.
- Administrators manage student accounts, ensuring data is kept up-to-date.

B. Teacher Registration and Management:

- Teachers register on the platform with relevant details such as teacher_id and contact information.
- Administrators manage teacher accounts, keeping track of their courses and activities.

C. Course Enrollment:

- Students enroll in various courses provided by the platform.
- Teachers are assigned to teach these courses, providing them with tools and resources to enhance the learning experience.

D. Learning Tools Utilization:

- Both students and teachers use the provided learning tools and resources.
- These tools are tailored to individual student needs, fostering personalized learning paths.

E. Dropout Risk Prediction:

- The system collects and analyzes data related to student performance, engagement, and other risk factors.
- A machine learning model predicts the dropout risk of each student.
- The system displays risk scores and suggests interventions for at-risk students.

F. One-to-One Communication:

- The platform facilitates direct communication between students and teachers through messaging and video conferencing.
- Teachers provide personalized feedback and support based on individual student needs.

G. **Centralized Database Management:**

- The system maintains a centralized database to store student data, academic records, attendance, and performance metrics.
- The database ensures secure and efficient storage and retrieval of data, adhering to quality standards and regulatory guidelines.

H. **Feedback and Diagnostics:**

- The platform collects feedback and diagnostic information from learning activities.
- This data is stored in the NoSQL database (MongoDB) for further analysis and improvement of the learning process.

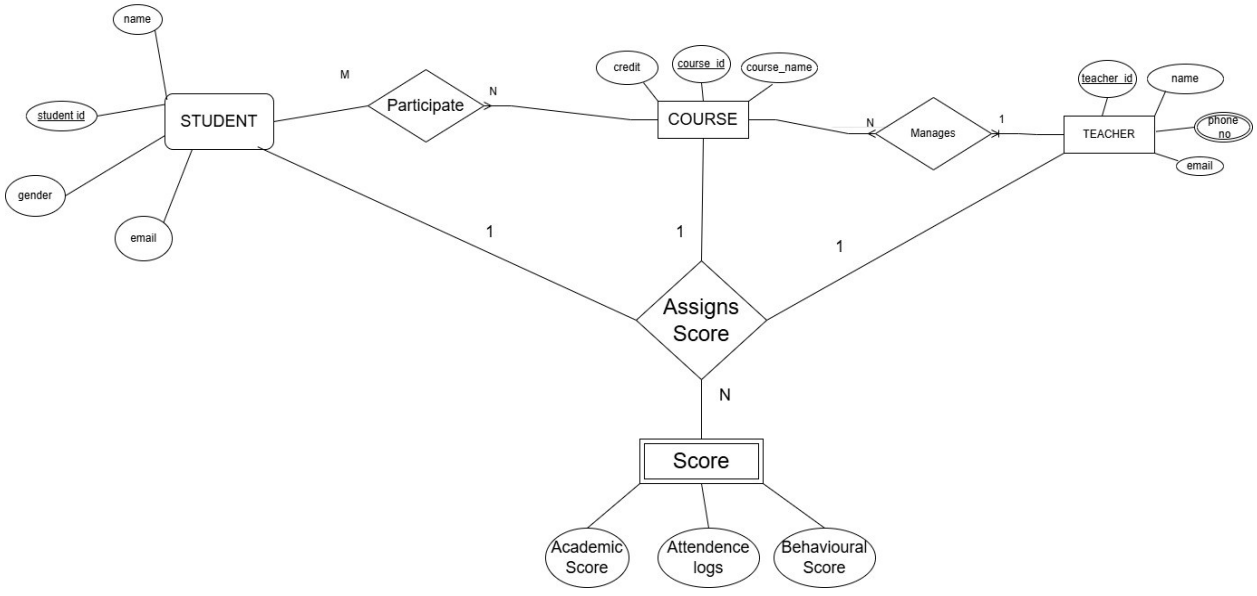

Figure 1 ER Diagram

Chapter 4

Detailed Design

A Data Flow Diagram (DFD) is a conventional visual tool that illustrates the flow of information within a system, providing a clear depiction of system requirements. It can encompass manual, automated, or mixed processes, showcasing how data enters and exits the system, undergoes modifications, and where it's stored. The primary aim of a DFD is to delineate the scope and boundaries of a system comprehensively. DFDs are hierarchical, with each layer delving deeper into the system or data. Typically, these layers are represented as Levels 0 through 2, each progressively revealing more intricate details about the system's components and operations.

4.1 DFD Level 0

A Data Flow Diagram (DFD) Level 0, also recognized as the fundamental system model or context diagram, portrays the entirety of software requirements within a single bubble. Input and output data are indicated by incoming and outgoing arrows. Subsequently, the system is broken down and depicted as a DFD with multiple bubbles. Each of these bubbles represents parts of the system, which are further decomposed and elaborated upon through more detailed DFDs as the analysis progresses.

1. **At Level 0**, all data flows into the **Admin** entity, representing the central control of the blood bank management system.
2. **Admin** processes incoming data and oversees the entire system's operations, including donor examinations, blood donations, and patient blood distribution.

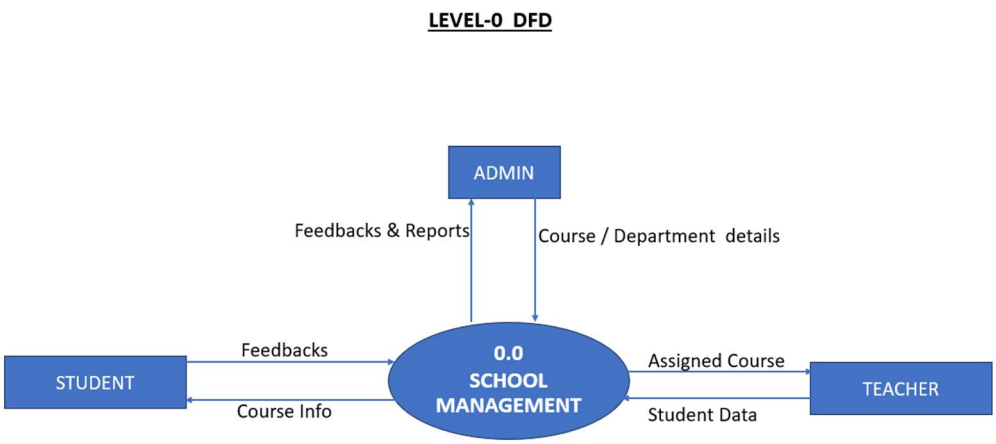

Figure 2 DFD Level 0

4.2 DFD Level 1

Level 1 of the Data Flow Diagram (DFD) provides a deeper level of detail, focusing more sharply on specific functions within the system or process. It breaks down the main functions highlighted in the Level 0 overview, providing explanations for each. At this level, the context diagram is decomposed into multiple bubbles or processes, emphasizing the main objectives of the system and breaking down the high-level processes outlined in the Level 0 DFD into subprocesses. In simpler terms, Level 1 DFD elaborates on the overall system's functions and breaks them down into more detailed subprocesses.

1. In Level 1, the process "Student Enrollment (Process 1)" entails the registration of students into the system, including verifying documents and storing student information in the student database.

2. "Grade Recording (Process 2)" involves teachers updating student grade information and storing these records in the grade database.

3. "Course Management (Process 3)" involves admins managing course details and storing this information in the course database.

4. "Manage Attendance (Process 4)" entails teachers submitting attendance records, which are then stored in the attendance database.

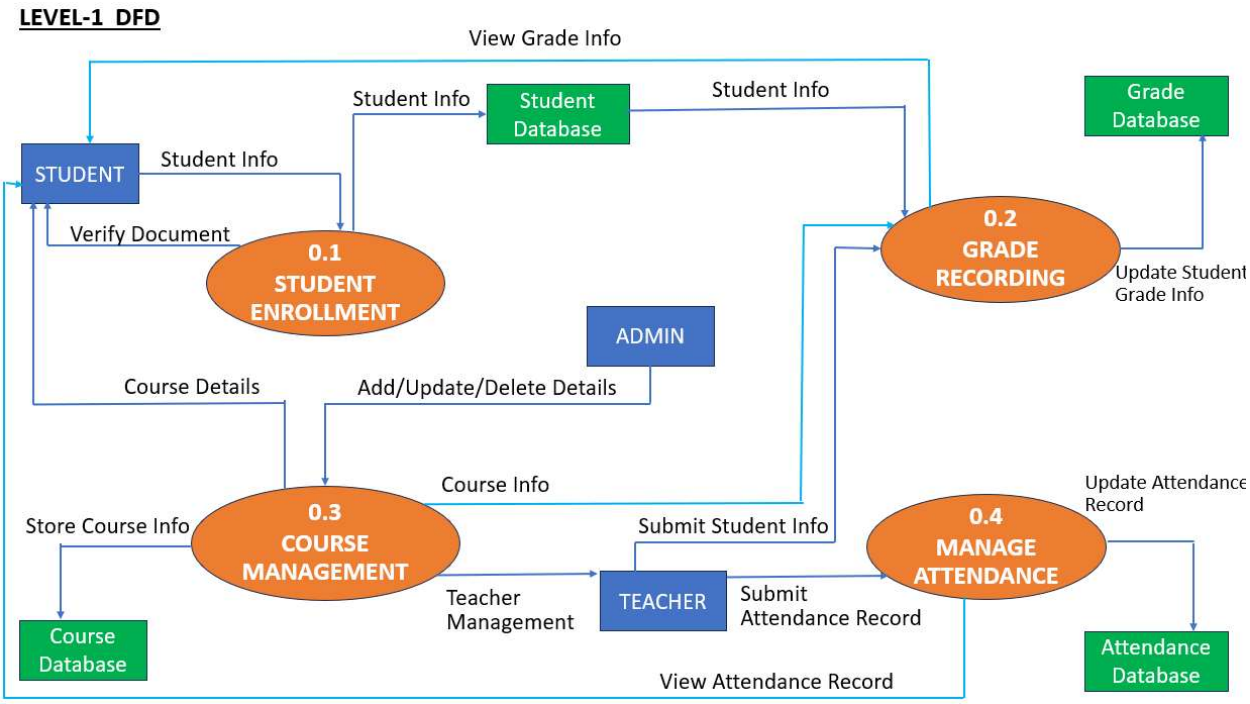

Figure 3 DFD Level 1

Chapter 5

Relational schema and Normalization

Relation schema defines the design and structure of the relation like it consists of the relation name, set of attributes/field names/column names. Every attribute would have an associated domain.

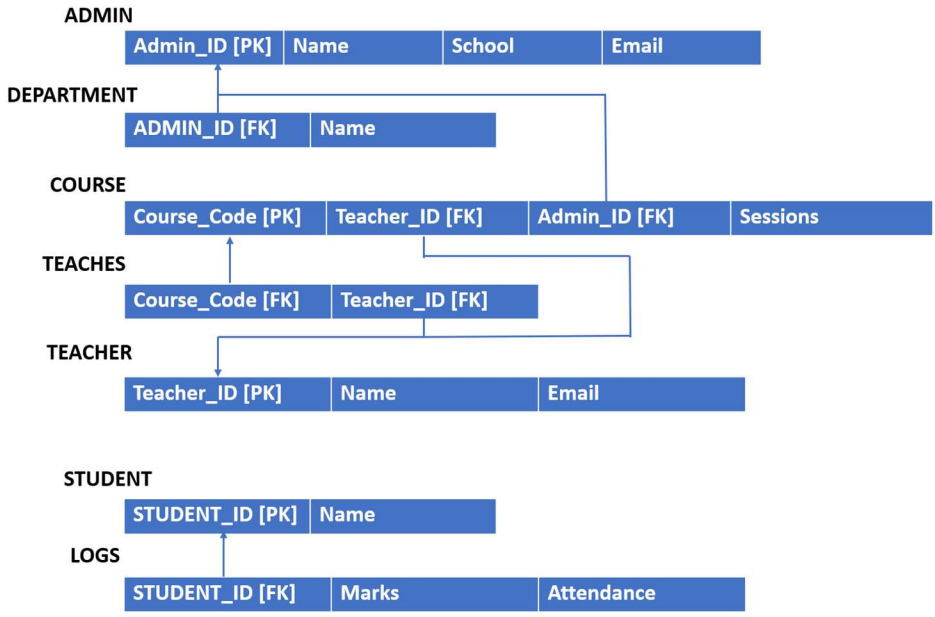

Figure 4 Schema Diagram

Normalization

Normalization is the process of organizing data in a database. It includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.

Normalization works through a series of stages called Normal forms.

1 NF

A relation will be 1NF if it contains an atomic value. It states that an attribute of a table cannot hold multiple values. It must hold only single-valued attribute**.**

In our schema all the attributes for all the entities are atomic in nature. So it is already in 1NF and need not be reduced**.**

Student_id	Student_name	**Sclass_id**	**Admin_id**
3	122student	2	1
4	123student	2	1
5	124student	4	1

Table 1 Sample Student Table

Admin_Id	**Admin_Name**	**Admin_email**	**School_Name**
1	KISHAN KUMAR	kishankumar1@gmail.com	Schoolb
2	HRUTHIK K K	hruthikkk@gmail.com	SCHOOL 1
3	KIRAN V	kiranv@gmail.com	School
4	A	a@gmail.com	KV

Table 2 Sample Admin Table

2 NF

In the 2NF, relational must be in 1NF. In the second normal form, all non-key attributes are fully functional dependent on te primary key

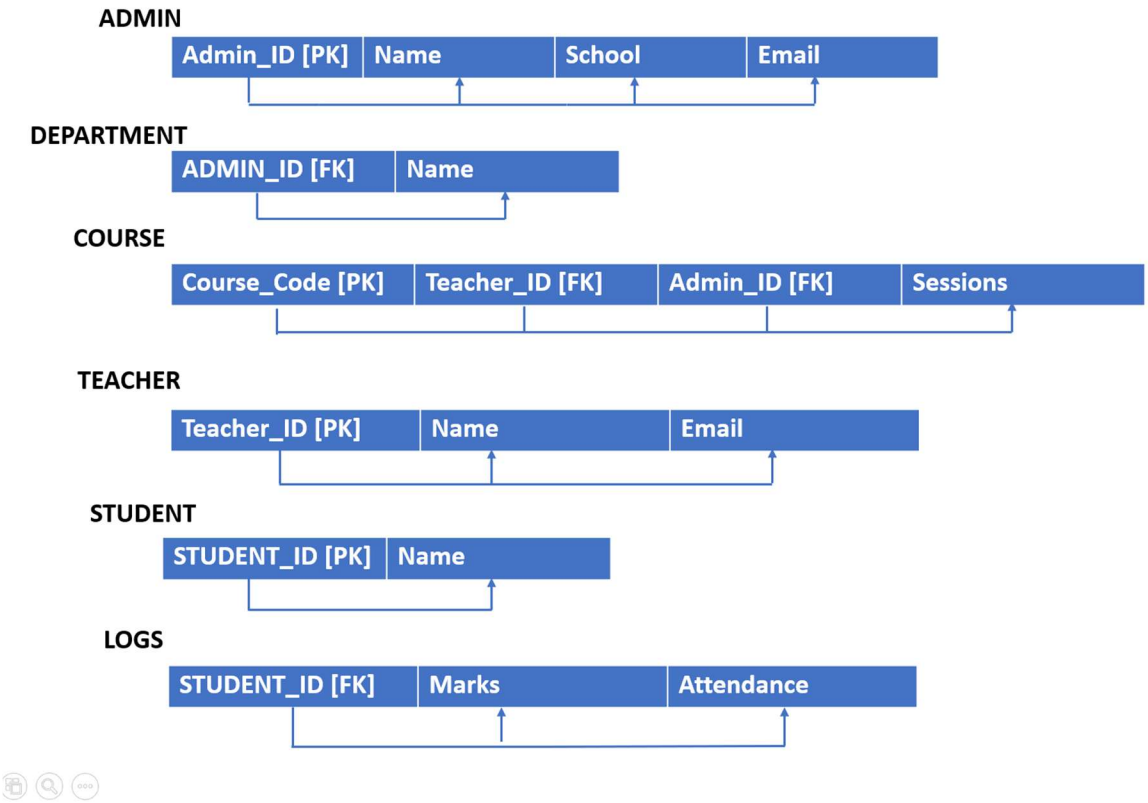

Figure 5 - 2 NF

3 NF

A relation will be in 3NF if it is in 2NF and not contain any transitive partial dependency. 3NF is used to reduce the data duplication. It is also used to achieve the data integrity. If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.

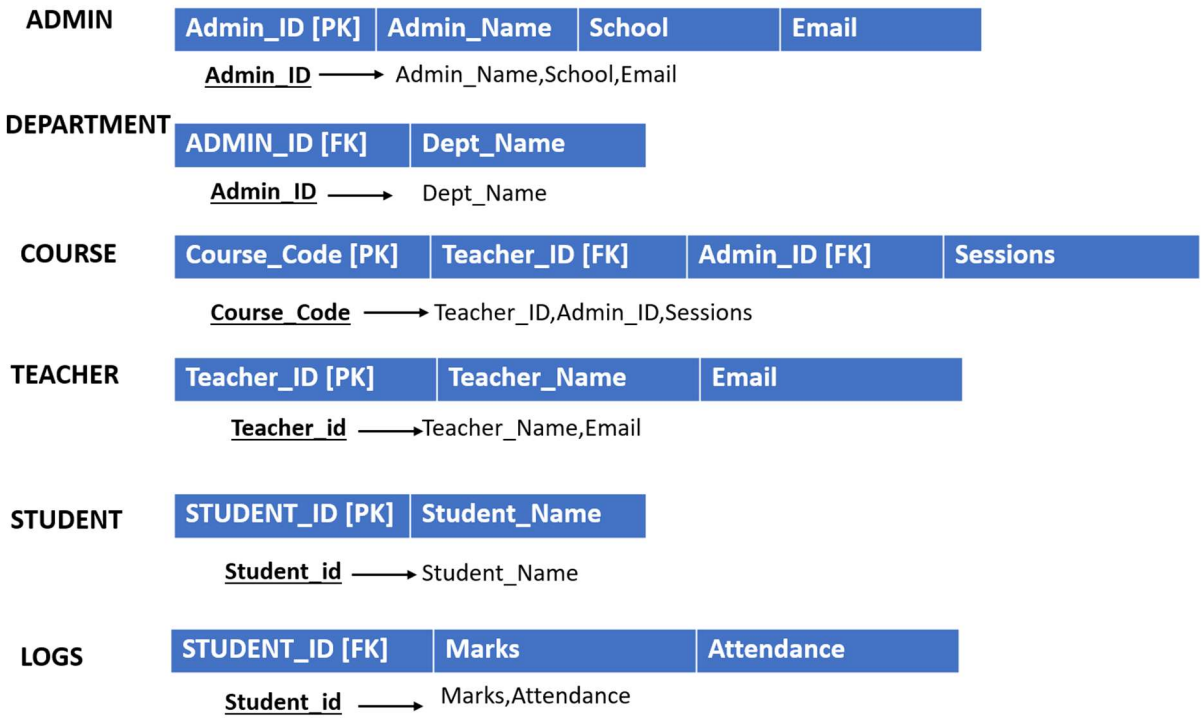

Figure 6 - 3 NF

Chapter 6

NOSQL

The integration of MongoDB as our chosen NoSQL database in the Student Management System underscores our commitment to addressing the diverse and unstructured nature of School data. This strategic decision is driven by the recognition that traditional relational databases may encounter challenges in efficiently handling the variability and volume of information inherent in customer reviews. MongoDB's flexible schema and scalability align seamlessly with the dynamic requirements of our project, ensuring optimal storage, retrieval, and management of unstructured data. This deliberate selection of a NoSQL approach emphasizes our dedication to creating a versatile and adaptive system capable of robustly analyzing and responding to the intricacies of customer feedback in the ever-evolving pharmaceutical landscape.

Integrating NOSQL database to SQL with complete system architecture

In our integrated database system, the entity **"student"** serves as a crucial bridge between two distinct database platforms: MySQL and MongoDB. The primary key of the **"student"** entity, aptly named **"student_id,"** acts as the linchpin for synchronizing data between these systems. MySQL houses the personal information of students, ensuring secure and structured storage of sensitive details such as names, addresses, and contact information.

Conversely, MongoDB is leveraged for storing dynamic data related to the personalized learning analytics and engagement of students. This includes a comprehensive repository of academic performance, learning preferences, and interaction records. By utilizing the shared primary key, data in MongoDB can be precisely matched to its corresponding student entry in MySQL, facilitating seamless retrieval and correlation of information across the integrated environment.

Through this hybrid approach, we maintain the integrity and scalability of our database system while capitalizing on the strengths of both SQL and NoSQL technologies. MongoDB provides the flexibility and scalability required to accommodate the evolving nature of learning analytics and personalized data. Together, they form a cohesive foundation for efficiently managing and accessing comprehensive student profiles within our integrated database ecosystem

Chapter 7

Conclusion & Future Enhancement

In conclusion, our model represents a significant advancement in personalized learning and student retention systems, offering robust features for efficiently managing student information, personalized learning paths, and teacher-student interactions. Through the integration of MySQL and MongoDB, we have achieved a balanced approach, leveraging the relational integrity of MySQL for structured data while harnessing MongoDB's flexibility and scalability for unstructured and evolving datasets. The system's capabilities extend beyond mere data storage, with real-time analytics, personalized learning recommendations, and high availability mechanisms ensuring reliable and insightful management of student learning experiences. Overall, our model stands as a testament to the synergy between traditional relational databases and modern NoSQL solutions, providing a comprehensive platform for educational institutions.

Looking towards future enhancements, there are several avenues for further refinement and expansion of our model. One potential area of improvement lies in enhancing the user interface and experience, with features such as personalized dashboards, mobile optimization, and intuitive data visualization tools. Additionally, exploring the integration of machine learning algorithms could enable predictive analytics for student performance forecasting, retention strategies, and personalized learning recommendations. Moreover, continued efforts towards data security and compliance with educational regulations will remain paramount, necessitating regular audits, encryption enhancements, and adherence to evolving standards. By fostering innovation and adaptability, we aim to continuously enhance our personalized learning and student retention system, empowering educators with cutting-edge tools to ensure the timely and efficient delivery of quality education to all students.

References

[1] Personalized Learning and Student Retention Project Team, "A Comprehensive Full-Stack Web Application for Personalized Learning and Student Retention using MERN, MySQL, and Flask," unpublished, Jan. 2025.

[2] A. Al-Shaikhli, M. A. Hussain, and S. Zubair, "Predicting Student Dropout Using Machine Learning Models," in *2023 IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2023.

[3] B. D. McCloskey and C. L. McMahon, "Leveraging Technology for Personalized Education," in *IEEE Transactions on Learning Technologies*, vol. 16, no. 2, pp. 234–245, 2024.

[4] J. Smith, "Educational Interventions Using Predictive Analytics," in *Proceedings of the IEEE Global Engineering Education Conference (EDUCON)*, 2023.

[5] S. Gupta and P. Sharma, "Full-Stack Development for Adaptive Learning Systems," in *2023 IEEE International Conference on Smart Technologies (EuroCon)*, 2023.

[6] R. Chen and X. Liu, "MERN-Based Educational Platforms for Student Engagement," in *2024 IEEE International Conference on Big Data (BigData)*, pp. 812–820.

[7] T. Nguyen, A. Patel, and M. George, "Data-Driven Decision Making in Education Using MySQL and Flask," in *IEEE Access*, vol. 11, pp. 112–123, 202

Appendix

Screenshots with descriptions

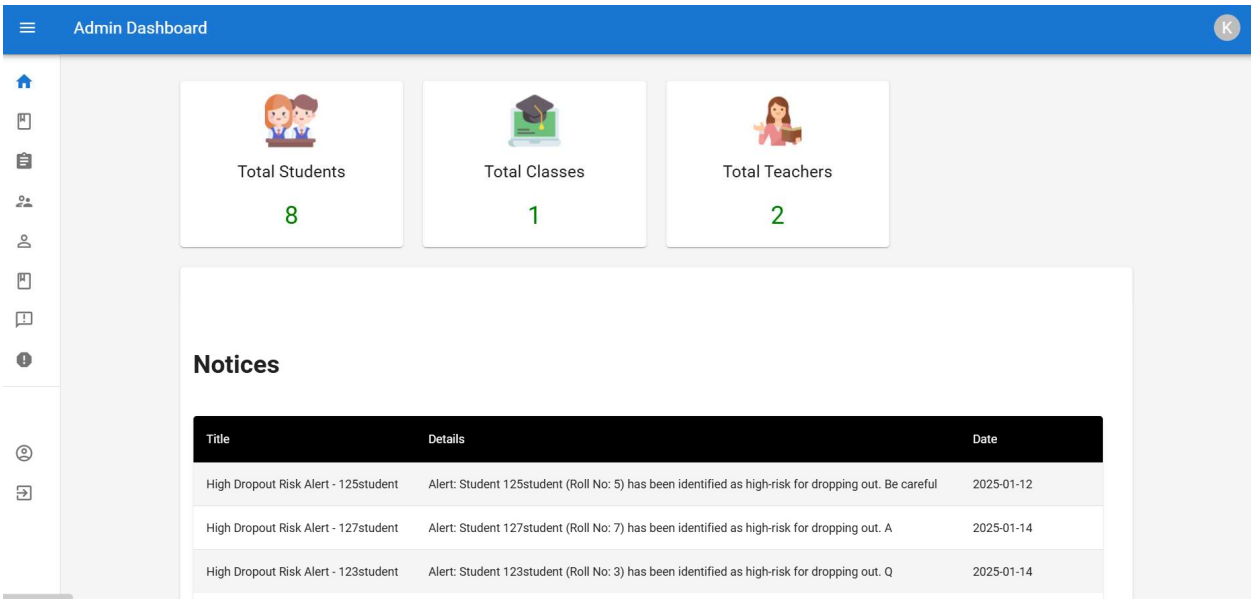

Figure 7 Home Page

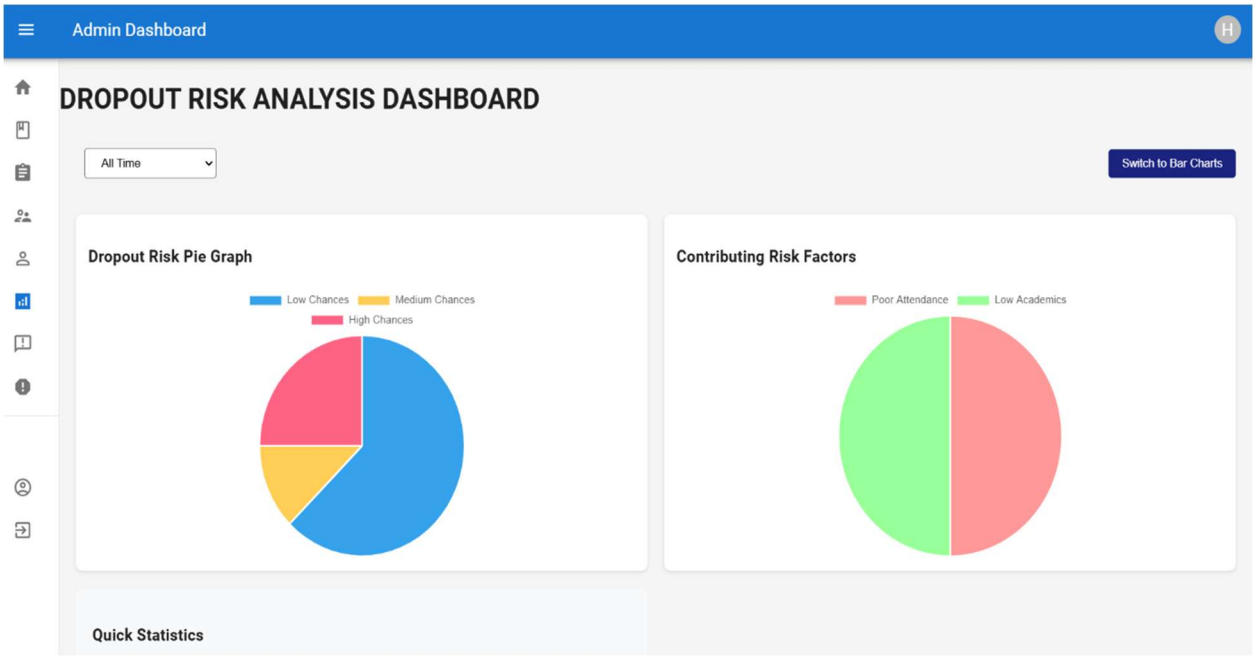

Figure 8 Dropout Prediction

This is the Machine Learning enabled prediction system

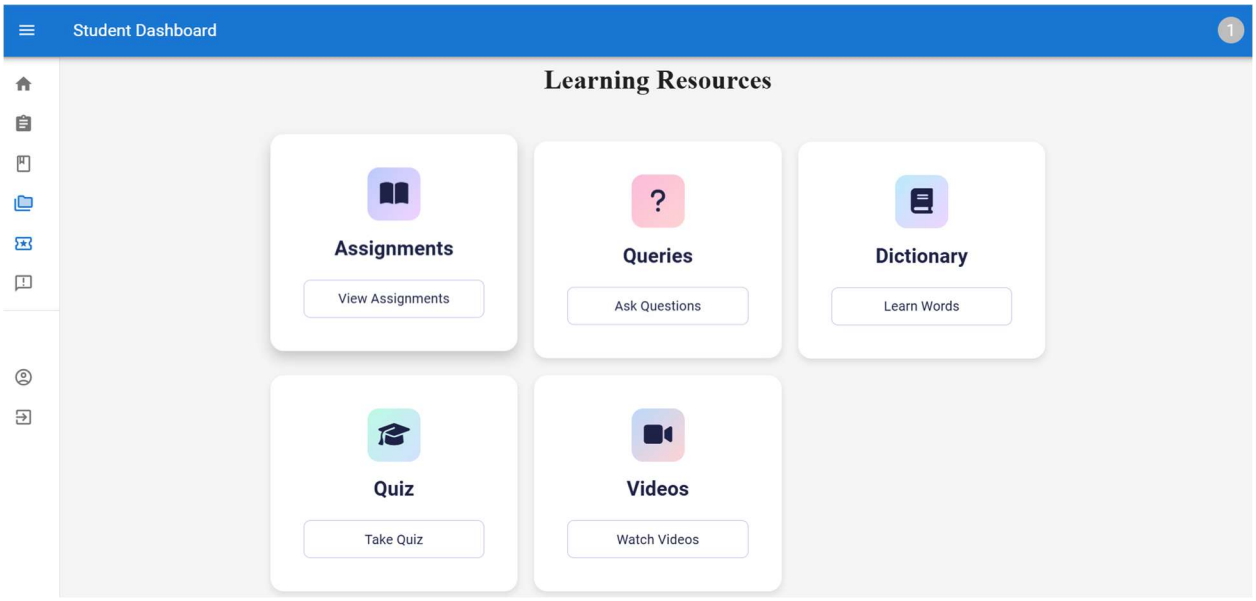

Figure 9 Personalized Learning tools

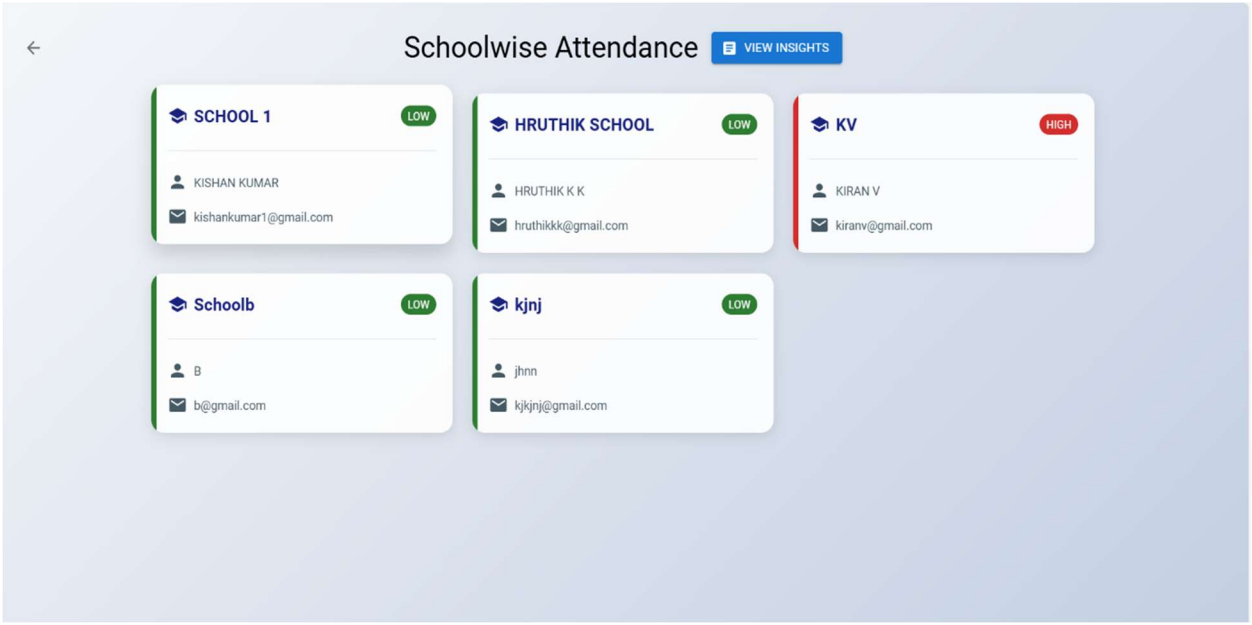

Figure 10 Schoolwise Dropout analysis

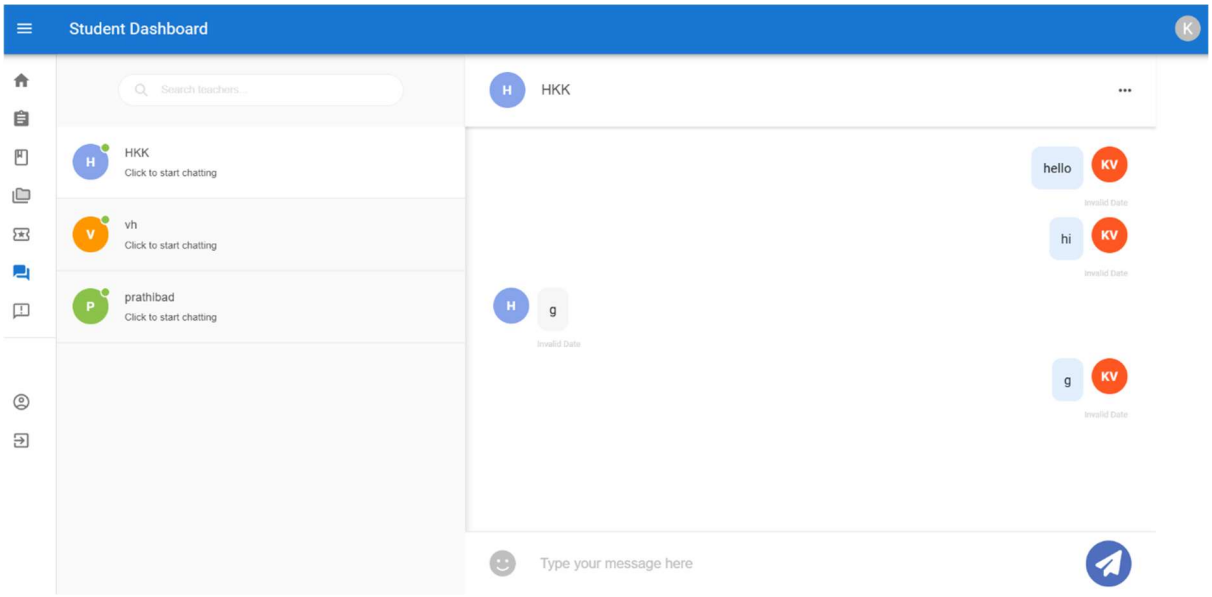

Figure 11 One to One Guidance