University Database Management App

Content

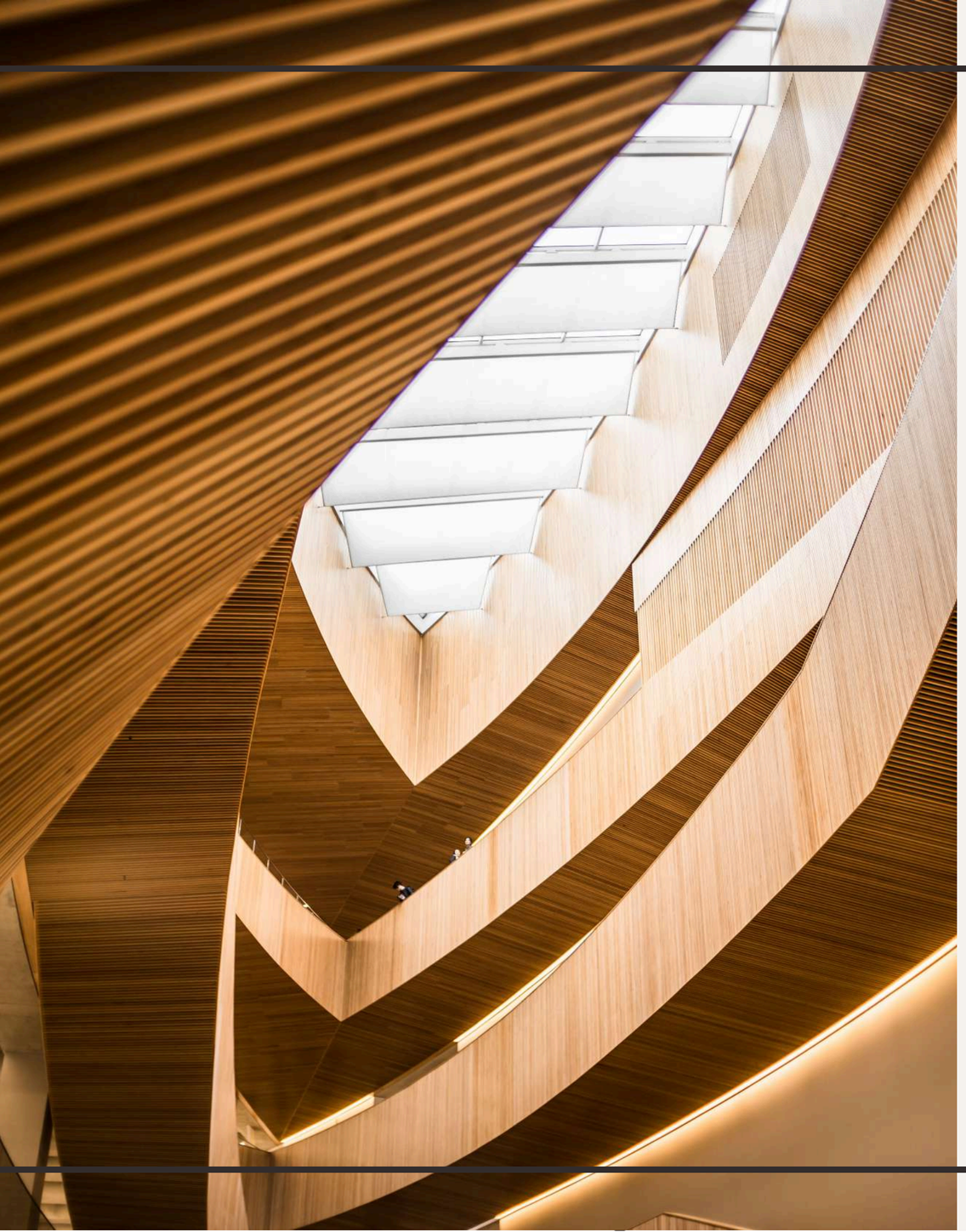
- Introduction
- Objective
- Architecture Overview
- Features
- Technology Stack
- Demo
- Conclusion
- Summary

Introduction

In today's educational landscape, efficient data management is crucial for universities to thrive. Our project focuses on developing a comprehensive system to handle student information, course structures, and departmental functionalities. Leveraging technologies like SQL, PLSQL, Java SE, and Bash scripting, we aim to streamline administrative processes and enhance academic operations.

Objective

Our objective is to democratize university database management, empowering every staff member to effortlessly interact with our system. Through intuitive interfaces and user-friendly controls, tasks such as adding new students, modifying courses, or updating information become accessible to all. By eliminating the need for specialized technical expertise, our solution revolutionizes data management, ensuring efficiency and inclusivity in administrative processes.

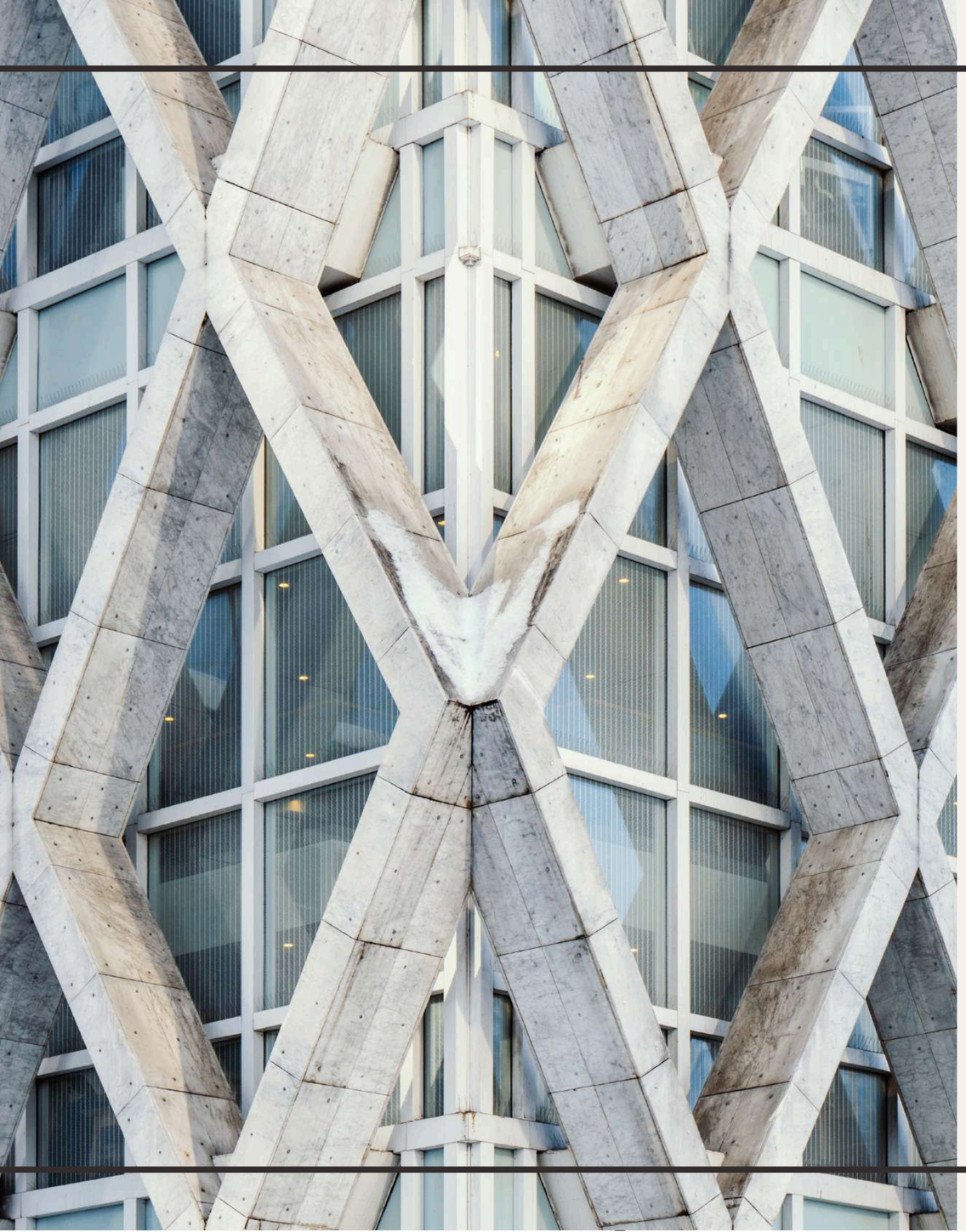


Architecture Overview

High-level overview of the application architecture:

- Utilizes the Model-View-Controller (MVC) design pattern
- Emphasizes separation of concerns:

Model: Handles data management
View: Manages the user interface
Controller: Implements application logic



Architecture Overview

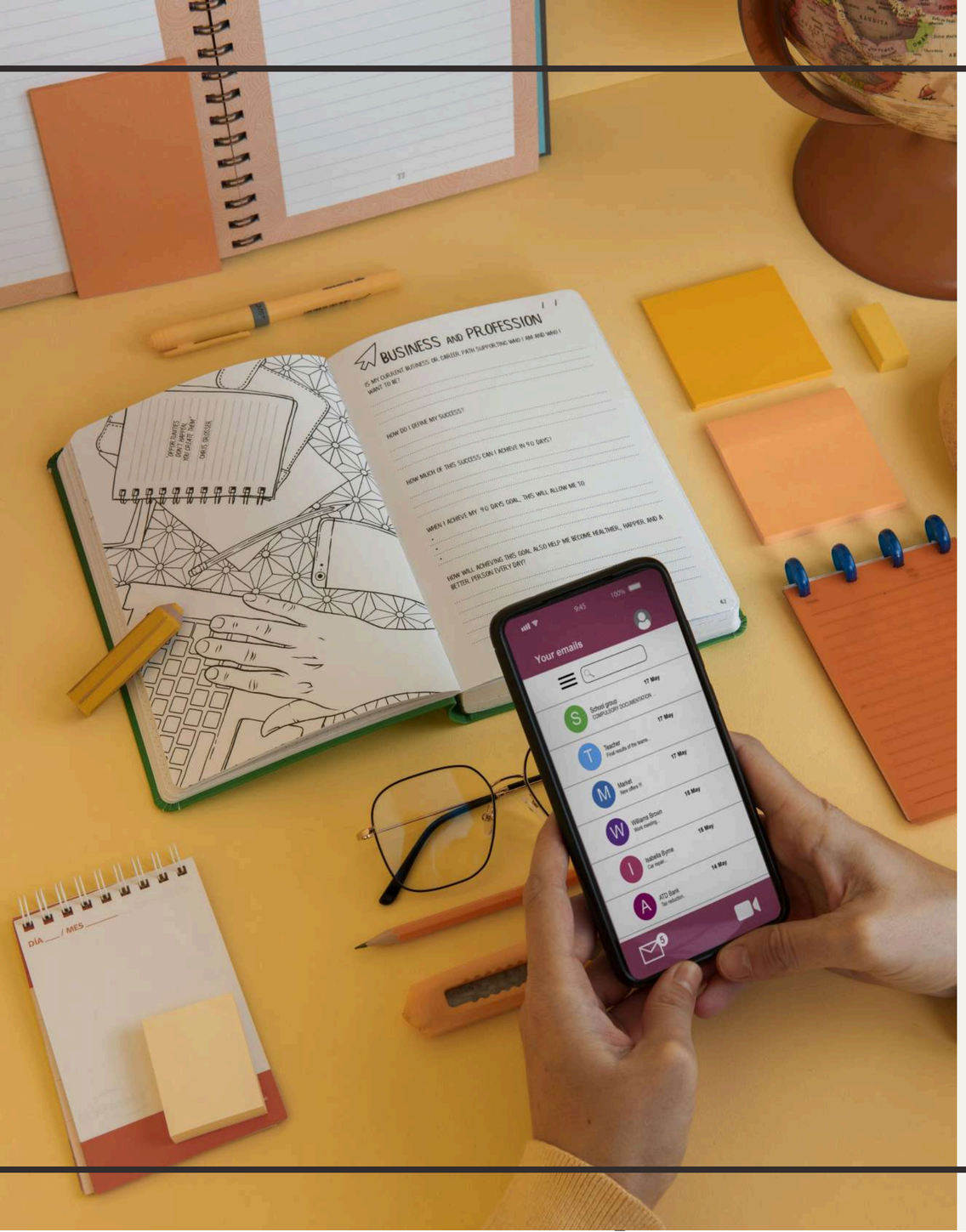
Cont'd

Advantages:

•Clear division of responsibilities enhances code maintainability and scalability.

•Facilitates collaboration among developers by providing a structured framework.

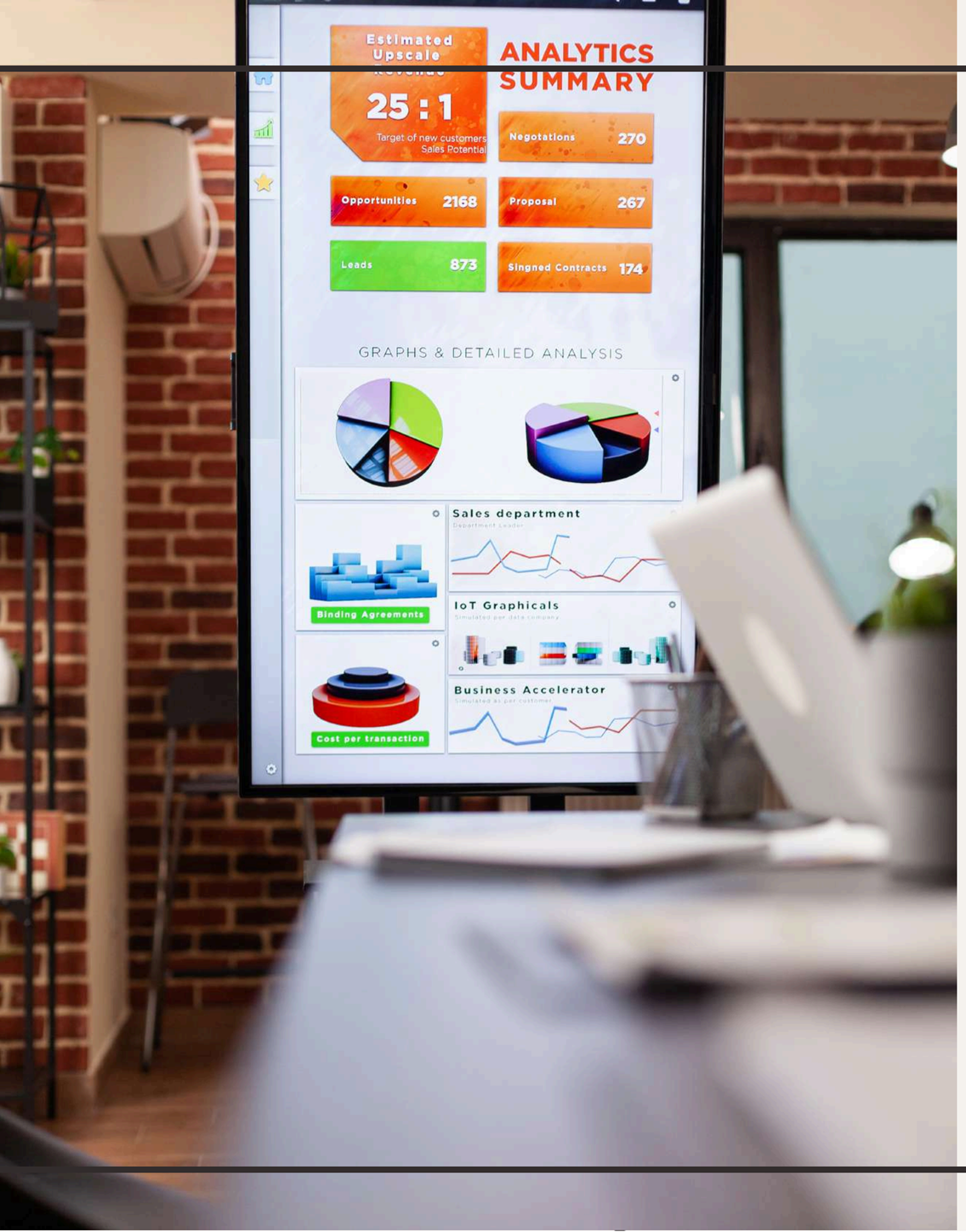
•Simplifies understanding and modification of code for new developers.



Features

CRUD operations :
Enable creation, retrieval, updating, and deletion of student, department, and course records.

Data Validation and Error Handling:
Ensures data integrity by validating inputs and providing appropriate error messages for incorrect entries.



Features

Cont'd

User-friendly Interface:
Offers an intuitive interface with easy-to-navigate menus and controls for seamless interaction with the application.

Courses Report Generation:
Generates detailed reports on available courses, course enrollments, and academic performance.

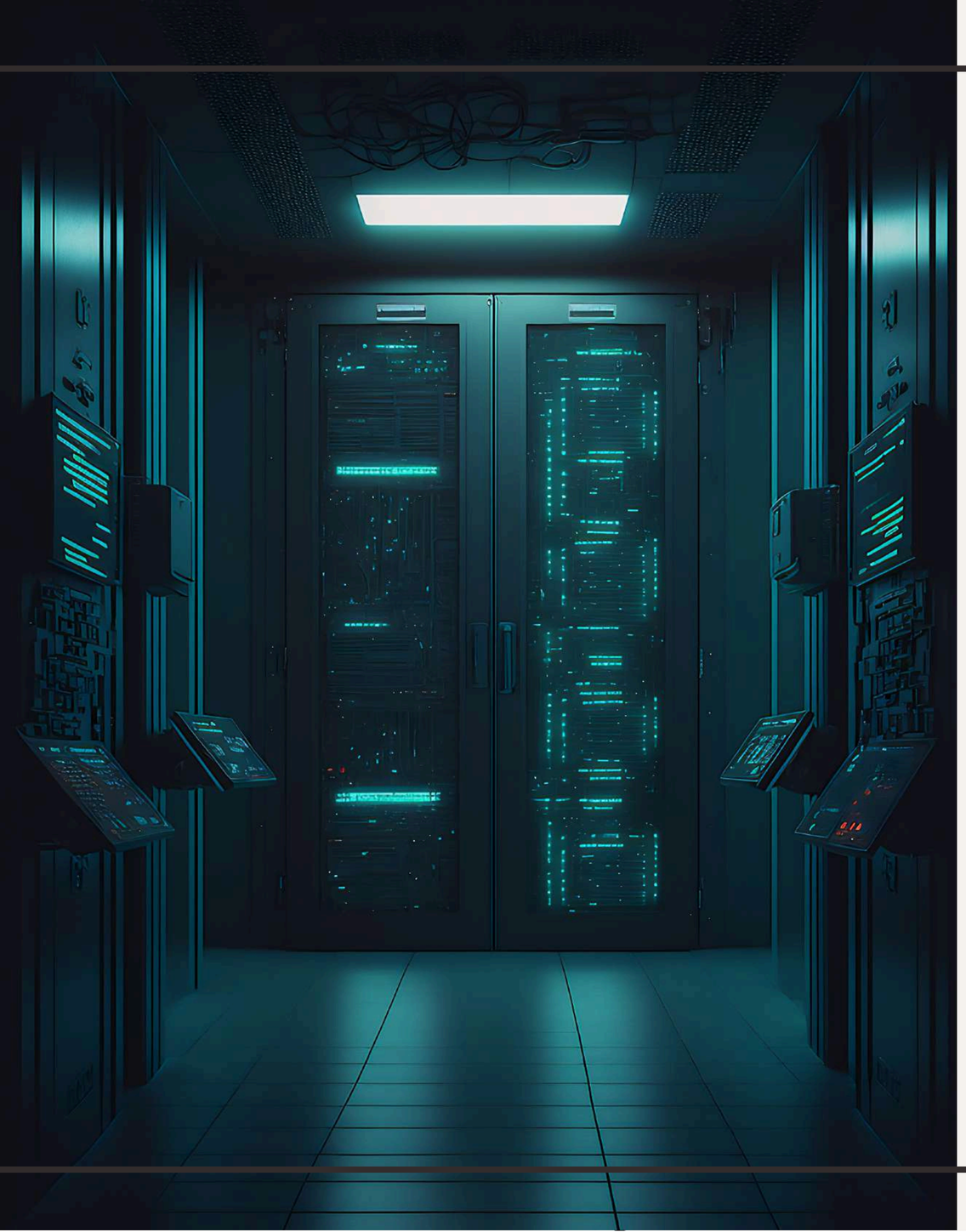
Features

Cont'd

Color-Coded Labels for Data Clarity:

- Utilizes color-coded labels to enhance data comprehension and input validation.
- Dark yellow denotes primary key fields mandatory for data retrieval, light yellow indicates secondary fields optional for retrieval.
- Black labels represent regular data, while white labels clarify information from related tables, such as displaying department names alongside department IDs for improved understanding.


Student
Phone
Department
Course
Assigned
Courses Report
Student_ID
First_Name
Last_Name
Street
City
State
Gender
Email
Birthdate
Dept_ID
Dept_Name
Cum_GPA
Enter Student_ID
Enter First Name
Enter Last Name
Enter Street
Enter City
Enter State
Enter M or F
Enter Email
This format 2002-09-22
Enter Dept_ID
For Retrieving only
For Retrieving only
Insert Student
Retrieve by ID
Update Student by ID
Delete Student by ID
Clear

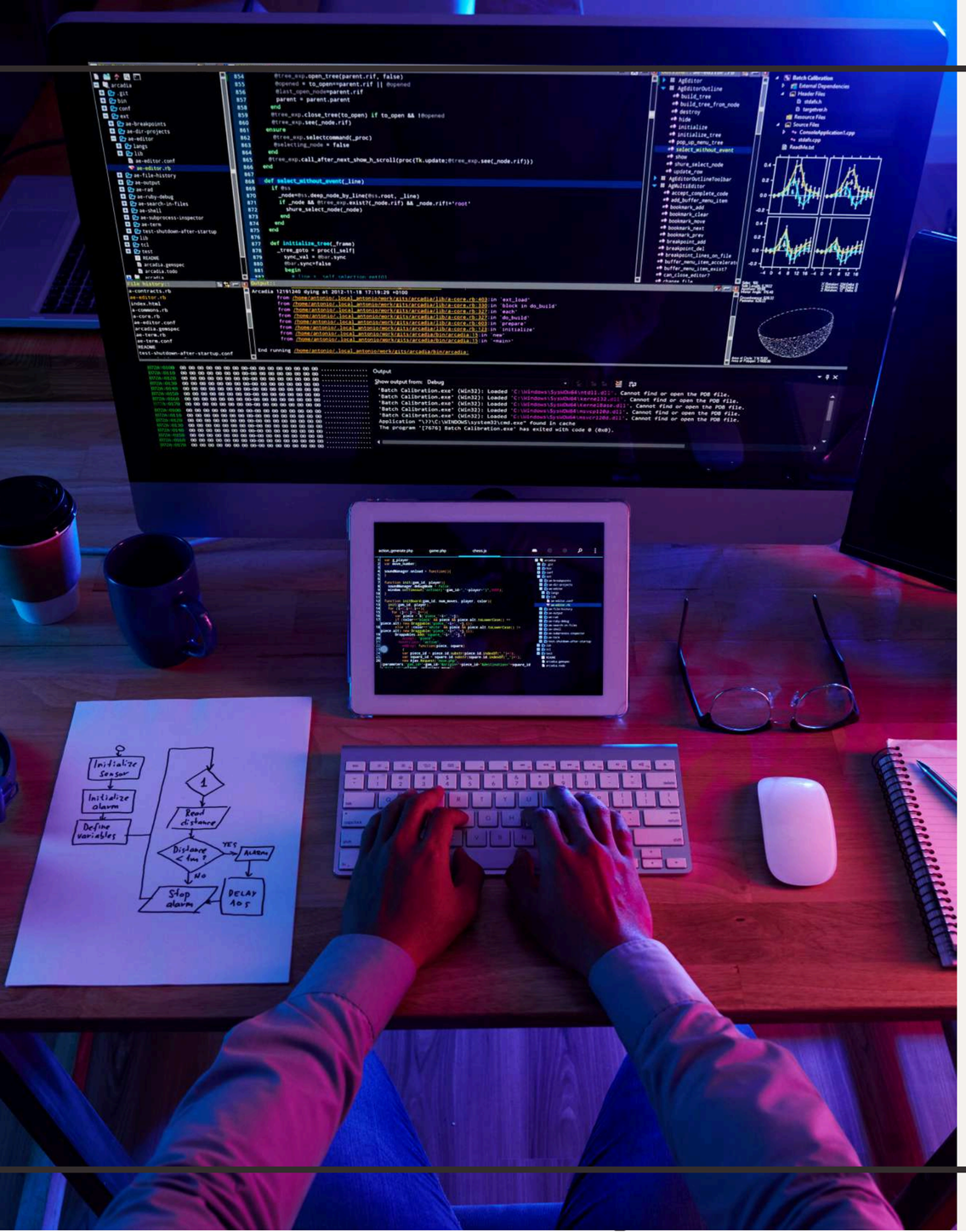


Technology Stack

JavaFX for User Interface (UI):
•Provides a robust framework for developing interactive and visually appealing UI components.

MySQL Database:
•Utilized as the backend data storage solution for efficient data management.

JDBC for Database Connectivity:
•Enables seamless interaction between the Java application and the MySQL database, facilitating CRUD operations.



Technology Stack Cont'd

Bash Scripting for Scheduled Backups and Disk Space Monitoring:

- Implements automation for scheduled backups, ensuring data integrity and security.
- Monitors disk space to prevent potential issues and maintain system stability.

MVC Architecture for Code Organization:

- Adheres to the Model-View-Controller design pattern, ensuring clear separation of concerns and maintainable codebase.

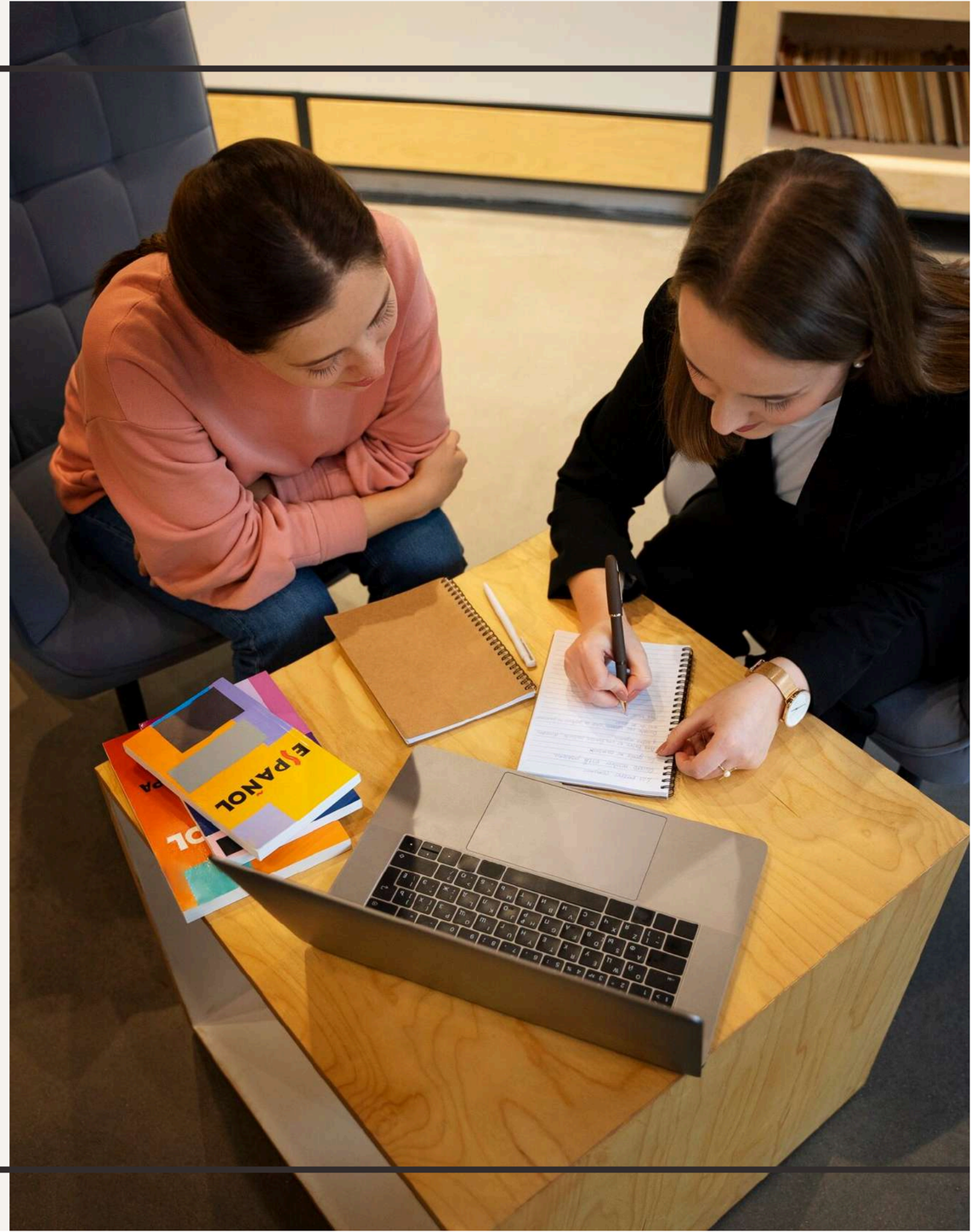
Demo

Live demonstration of the application's main functionalities in 2 minutes:

- Inserting, updating, and deleting different tables records.
- Testing data validations, database constraints and triggers.
- Demonstrate change in cumu_GPA after insertion, updating and deleting.
- Generating reports on course assignments and their students performance.

Conclusion

The **Efficient University Database Management App** is a vital tool for modernizing academic data management. By addressing the challenges of data redundancy, complexity to manage and time-consuming administrative tasks, the app significantly contributes to the overall efficiency and effectiveness of university operations.



Summary

- Objective
- Architecture Overview
- Features
- Technology Stack
- Demo

Any Questions?

Thanks!