**“****College Android Application(SGGS app) Database and API Development.”**

Submitted for the fulfilment of DBMS Project in

# Third year

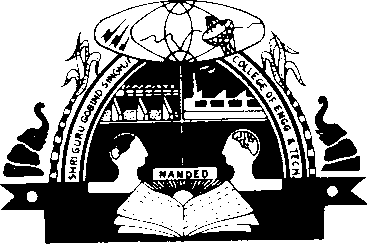
**Computer Science and Engineering**

By

* **2021BCS073**: Sushil Bhoybar

Under the Guidance of

**Mrs. Misha Nihalani**



# SHRI GURU GOBIND SINGHJI INSTITUTE OF ENGINEERING & TECHNOLOGY,

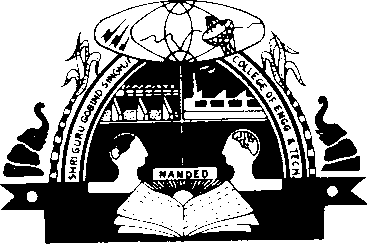
**NANDED (M.S.) 431606 ACADEMIC YEAR**

**(2023-24)**

**SHRI GURU GOBIND SINGHJI INSTITUTE OF ENGINEERING & TECHNOLOGY,**

# Nanded-431606,

**Maharashtra, India.**



# Department of Computer Science and Engineering

**CERTIFICATE**

This is to certify that Mr. Sushil Bhoybar (2021BCS073) studying in Third Year B. Tech has submitted the report for DBMS project on the topic “College Android Application(SGGS app) Database and API Development.” in the year 2023-24.

We award this certificate of completion of technical seminar as a record of bona fide work carried out by them under my supervision and guidance.

Date :04.12.2023

Guide Head Of Department

Mrs. Misha Nihalani Prof. S.M.Bansode

# ACKNOWLEDGEMENT

### We as a group would like to thank Mrs. Misha Nihalani Ma’am, our DMBS mentor for their help and guidance in completing our project “College Android Application(SGGS app) Database and API Development”. You have always taught us and guided us to understand things that we should know while studying the topic and in producing good project work.

We thank Prof. Mrs. Suwarna Bansode Ma’am (HOD CSE department) for her ongoing support and encouragement in every aspect. Finally, the entire staff of Department of Computer Engineering for guiding our thoughts and vision.

### The successful completion of our project would not have been possible without the dedicated support from all our mentors, family, and friends.

**Index**

|  |  |  |
| --- | --- | --- |
| **Sr. No.** | **Content** | **Pg. No.** |
| 1. | Introduction | 6 |
|  | 1.1 Project Definition | 6 |
|  | 1.2 Objectives | 7 |
|  | 1.3 System Analysis | 8 |
| 2. | Ch. 2. Literature Survey | 10 |
| 3. | System Developments And | 11 |
|  | Requirements |  |
|  | 3.1 Hardware Requirement | 11 |
|  | 3.2 Software Requirement | 11 |
|  | 3.3 Front-end Details | 11 |
|  | 3.4 Connectivity | 11 |
|  | 3.5 Some more technology  3.6 Diagrams | 12  13 |
| 4. | Implementation Planning | 14 |
|  | 4.1 Implementation Planning | 15 |
|  | 4.2 Important Screenshots | 17 |
| 5. | Conclusion | 19 |
|  | 5.1 Future Scope | 20 |
|  | 5.2 References | 21 |

**Abstract**

#### PROBLEM STATEMENT:

Colleges like SGGS have lacked an efficient system for students and faculty to access vital academic information stored in its database. The absence of a streamlined interface between the database and an Android application leads to a disconnect in accessing attendance records, exam results, and event notifications. The challenge is to develop a secure and user-friendly API that bridges this gap, allowing seamless data transactions between the college's database and the Android app. The solution should prioritize data security, scalability, and performance while enabling functionalities crucial for student-faculty interaction. The aim is to create a cohesive platform that fosters informed decision-making and enhances communication within the college community.

#### PROBLEM SOLUTION:

##### Objective:

The primary objective of this project is to design a comprehensive database system and effectively manage data pertaining to students and academic information within College. This database will serve as the backbone for an Android application, facilitating streamlined access to attendance records, exam results, and real-time event notifications. The focus lies in creating an API that securely connects this database with the Android app, ensuring seamless data retrieval and updates. Ultimately, the goal is to optimize data management, fostering a more connected campus community by enhancing accessibility and communication within the college environment.

##### Features:

 **Attendance Management:**

* View and track attendance records for individual students.
* Mark attendance for classes and events through the app.

 **Exam Result Retrieval:**

* Access and display exam results for various subjects or semesters.
* Provide detailed breakdowns of grades and performance.

 **Event Notifications:**

* Push notifications for upcoming events, seminars, workshops, and academic deadlines.
* Allow users to RSVP or register for events directly through the app.

 **Course Schedule and Timetable:**

* Display personalized class schedules for students and faculty.
* Include details such as class timings, locations, and instructors.

 **Profile Management:**

* Allow users to manage their profiles, including personal information, contact details, and academic history.
* Edit profile information and upload documents (if necessary).

 **Feedback and Suggestions:**

* Allow users to provide feedback on courses, faculty, or the application itself.
* Incorporate a system for users to submit suggestions or complaints.

 **Security Features:**

* Secure sensitive data with user authentication, password protection, and encrypted connections for robust access control.

# Ch. 1. Introduction

## Project Definition

The College Android Application (SGGS app) Database and API Development project is a software initiative centered on enhancing the academic experience for college students within the SGGS community. This app-based solution aims to streamline access to vital academic information and campus-related updates. The primary goal is to develop a robust database infrastructure complemented by a purpose-built API, facilitating seamless interaction between the college's database and an intuitive Android application.

This project seeks to empower students with convenient access to critical academic data, including attendance records, exam results, and real-time event notifications. By leveraging this technology, the app aims to provide a user-friendly interface, ensuring students can effortlessly retrieve essential information on their smartphones. The API's core focus is to securely connect the Android application to the college database, enabling swift and secure data retrieval and updates.

Key functionalities embedded within the app include attendance management, exam result retrieval, event notifications, and personalized schedule access. Security measures, such as robust authentication protocols, will be implemented to safeguard sensitive student information. Additionally, scalability and performance optimizations are essential considerations to ensure the system's adaptability to future enhancements and to guarantee an optimal user experience

## Objectives

The objective of project planning is to provide a SMART:

S:- specific

M:- measurable A:- action-driven, R:- realistic

T:- time-bound framework that enables the reader to make reasonable scheduling.

These estimates are made within a limited time frame at the beginning of the project and should be updated regularly as the project progresses.

* **API Development**: Design and implement a RESTful API to handle data transactions between the college's database and the Android application.
* **Database Integration**: Connect the API securely with the college's database to retrieve and update information seamlessly.
* **Functionality Implementation**: Enable functionalities for attendance management, exam result retrieval, event notifications, etc. within the Android application.
* **Security Measures**: Implement authentication and authorization protocols to ensure data security and privacy.
* **Scalability and Performance**: Develop the API with scalability in mind to accommodate potential future enhancements and ensure optimal performance.

# System Analysis

## Existing System

The existing system lacks a unified platform for college students to access critical academic information effortlessly. Students face challenges in retrieving timely attendance records, exam results, and event notifications due to the absence of an integrated system. The current setup lacks a dedicated Android application and a seamless API connection to the college database, resulting in disjointed data accessibility and limited student-faculty interaction.

##### Disadvantages:

* Fragmented Information Access
* Limited Connectivity
* Inefficiencies in Data Retrieval
* Reduced Transparency
* Security Vulnerabilities
* Inadequate User Experience

## Proposed System

The proposed system has many advantages over the existing system. It requires less overhead and is very efficient. The proposed system deals with Limited Connectivity and Reduced Transparency efficiently.

# Ch. 2. Literature Survey

In exploring the landscape of database management in educational institutions, seminal works such as "Effective Database Management in Higher Education Institutions" by Smith et al. (2019) offer insights into optimizing database structures to accommodate the complexities of student data and administrative records. This paper emphasizes scalable architectures and efficient data models, providing a benchmark for structuring databases within academic settings. Additionally, articles like "Challenges in Managing Student Data: A Case Study of University" by Johnson (2020) elucidate the challenges universities face in managing student information, shedding light on practical solutions and frameworks for effective database management tailored to educational institutions.

In the realm of API development and integration, research papers like "API Design Patterns for Mobile Applications in Education" by Brown and Garcia (2018) offer a comprehensive overview of designing APIs to facilitate seamless data transmission between educational databases and mobile platforms. This paper outlines best practices in authentication, data retrieval, and error handling, providing guidance for robust API development crucial for the project's success. Furthermore, "Secure Integration of Educational Databases with Mobile Apps" by Lee and Patel (2021) delves into security measures and encryption techniques essential for protecting student and faculty data during API integration, offering invaluable insights into safeguarding sensitive information within educational systems. These seminal works collectively contribute to the foundational knowledge necessary for implementing a secure, efficient, and user-centric college application system.

# Ch.3. System Development and Requirement

##### Hardware Requirement:

* + - Core to Dual or above
    - 2 GB RAM
    - 25 GB Free Hard disk space
    - Network interface card or Modem
    - LAN Network

##### Software Requirements:

* + - Windows XP/7/8/10
    - Linux
    - DBMS
    - NetBeans IDE 8.0.2
    - SQL databases

##### Front-end web development details

* PHP: Often used for web development and creating APIs due to its ease of integration with databases like MySQL.
* Node.js: Known for its efficiency in handling asynchronous tasks, making it suitable for APIs with high concurrent requests.
* API Testing and Documentation:

Postman: A popular tool for testing and debugging APIs, allowing for the creation of requests, testing endpoints, and automating API testing workflows.

* JWT (JSON Web Tokens): Used for implementing authentication mechanisms in APIs, ensuring secure access to protected routes.

##### Connectivity (Front end and Back end):

**PHP is an amazing and popular language.**

It is powerful enough to be at the core of the biggest blogging system on the web (Word Press). It is deep enough to run the largest social network.

(Facebook)! It is also easy enough to be a beginner's first server-side language!

* + - PHP is an acronym for "PHP: Hypertext Pre-processor."
    - PHP is a widely used, open-source scripting language.
    - PHP scripts are executed on the server.
    - PHP is free to download and use.
    - PHP files can contain text, HTML, CSS, JavaScript, and PHP code.
    - PHP codes are executed on the server, and the result is returned to the browser as plain HTML.

##### Some more technology which is used:

* + - **XAMPP**

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages.

##### MYSQL

MySQL ("My S-Q-L", officially, but also called "My Sequel") is the world's second most widely used open-source relational database management system (RDBMS). SQL phrase stands for Structured Query Language.

The MySQL development project has made its source code available. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation

MySQL is a popular choice of database for use in web applications. Free-software-open-source projects that require a full- featured database management system often use MySQL.

# 3.6 Diagrams

# ER Diagram and Data Flow Diagram

# 

# Ch.4.Implementation Planning

## Implementation

##### SYSTEM IMPLEMENTATION :

System implementation marks the pivotal stage in the project where the meticulously designed architecture and plans are translated into a tangible, functional system. This phase involves the actual creation, development, and integration of various components, encompassing both the backend and frontend of the College Android Application (SGGS app) Database and API Development project.

Firstly, the backend implementation focuses on establishing the database infrastructure and crafting the API. This involves setting up the database system, defining schemas, tables, and relationships to accurately store and manage student data, attendance records, exam results, and other pertinent information. Simultaneously, the API development occurs, where the endpoints are created, incorporating authentication protocols, data retrieval, and update mechanisms. Attention to security measures, such as encryption and access control, is paramount during this phase to ensure the integrity and confidentiality of sensitive data.

Concurrently, the frontend implementation involves developing the Android application interface. This stage revolves around actualizing the user-centric design, integrating API calls to fetch and display data seamlessly within the app's modules. It encompasses creating intuitive screens for attendance management, exam result retrieval, event notifications, and other functionalities outlined in the project objectives. Ensuring a cohesive and user-friendly experience remains central, requiring rigorous testing and refinements to align with the initially conceptualized design and user expectations. Overall, system implementation demands meticulous attention to detail, collaboration among development teams, and iterative refinement to bring the envisioned system to life.

##### SYSTEM TESTING:

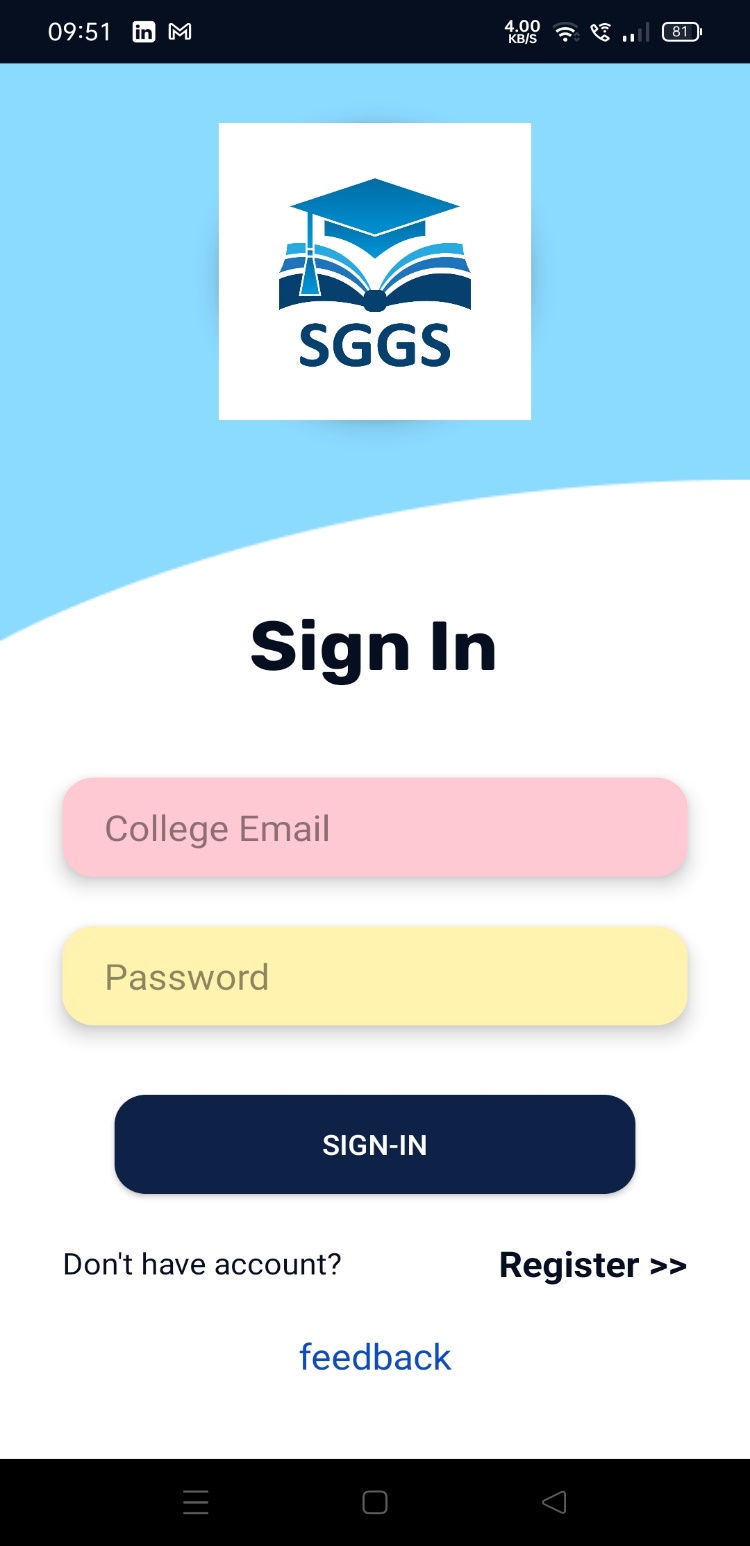
System testing constitutes a crucial phase in the project, validating the functionalities, performance, and reliability of the developed this project. This phase involves a comprehensive evaluation of the entire system, encompassing backend database operations, API functionality, and the frontend user interface.

Rigorous testing scenarios, including unit testing, integration testing, and end-to-end testing, are employed to identify and rectify any anomalies or discrepancies. This process ensures that the system operates seamlessly, meeting predefined requirements, handling expected loads, and delivering an intuitive user experience while adhering to stringent security measures.

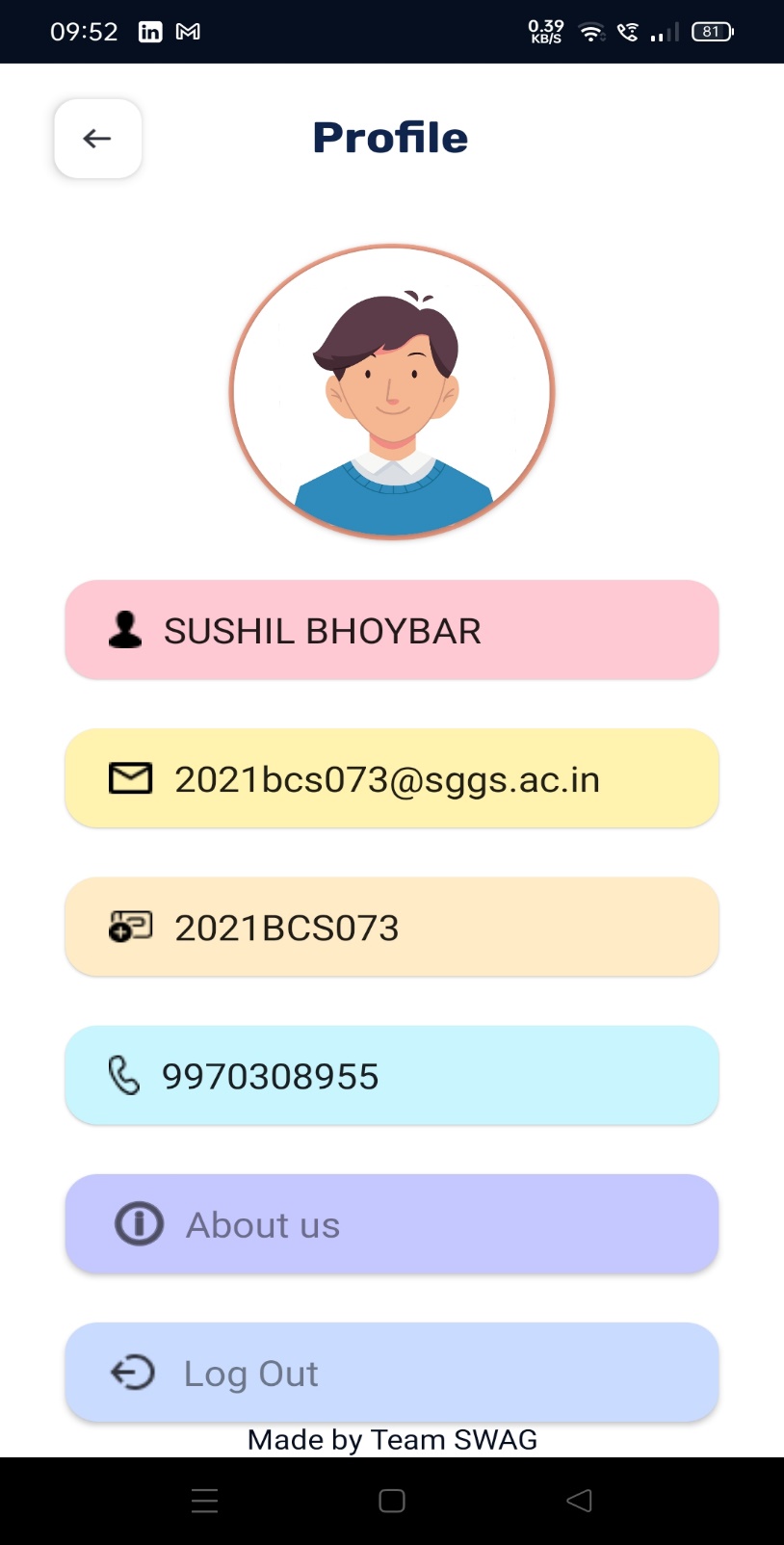
Ultimately, system testing is pivotal in ensuring the robustness and readiness of the application for deployment and usage within the college community.

* 1. **Implementation Screenshots**

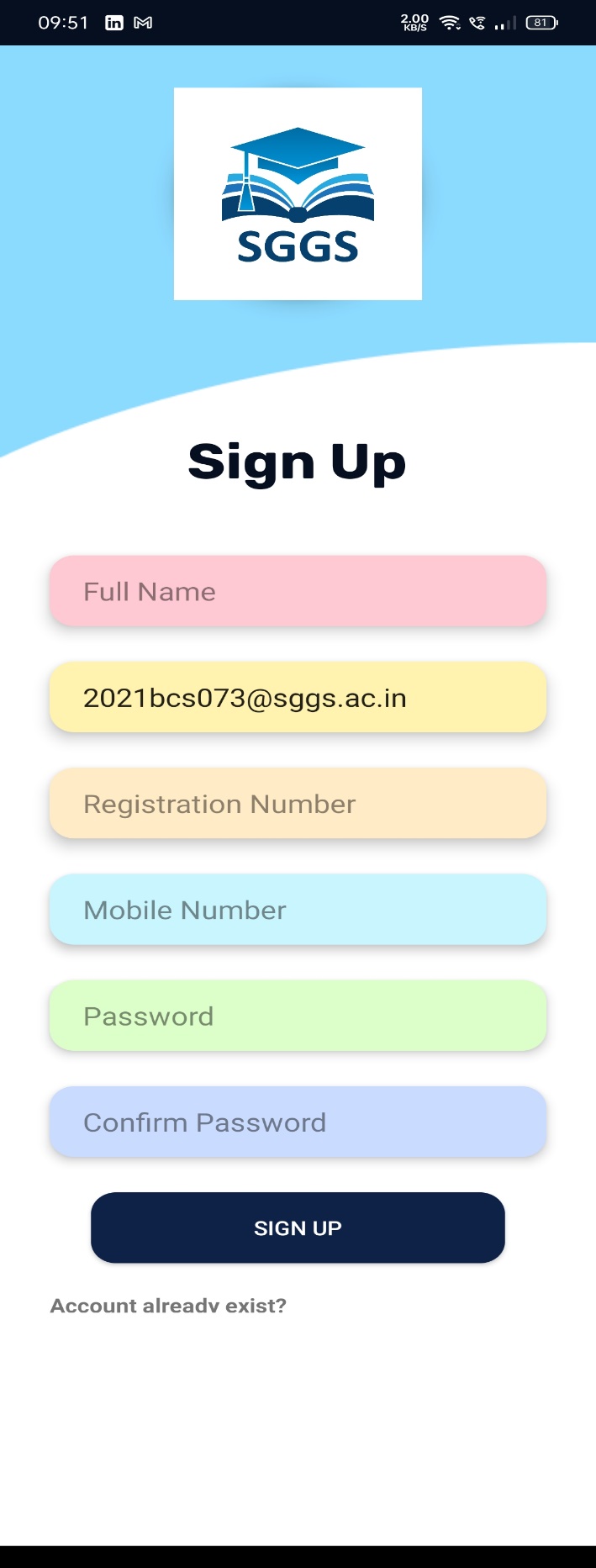
##### Admin login



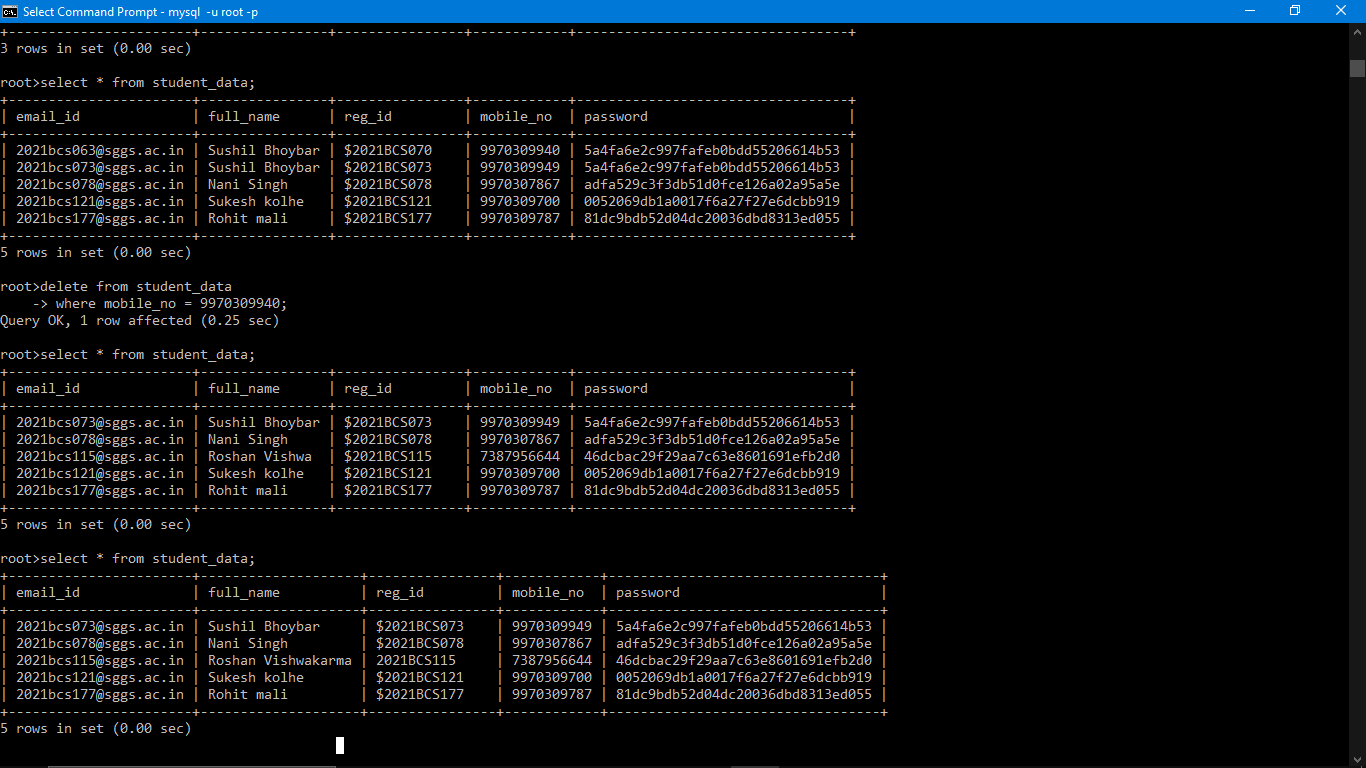
* + 1. **Update profile**

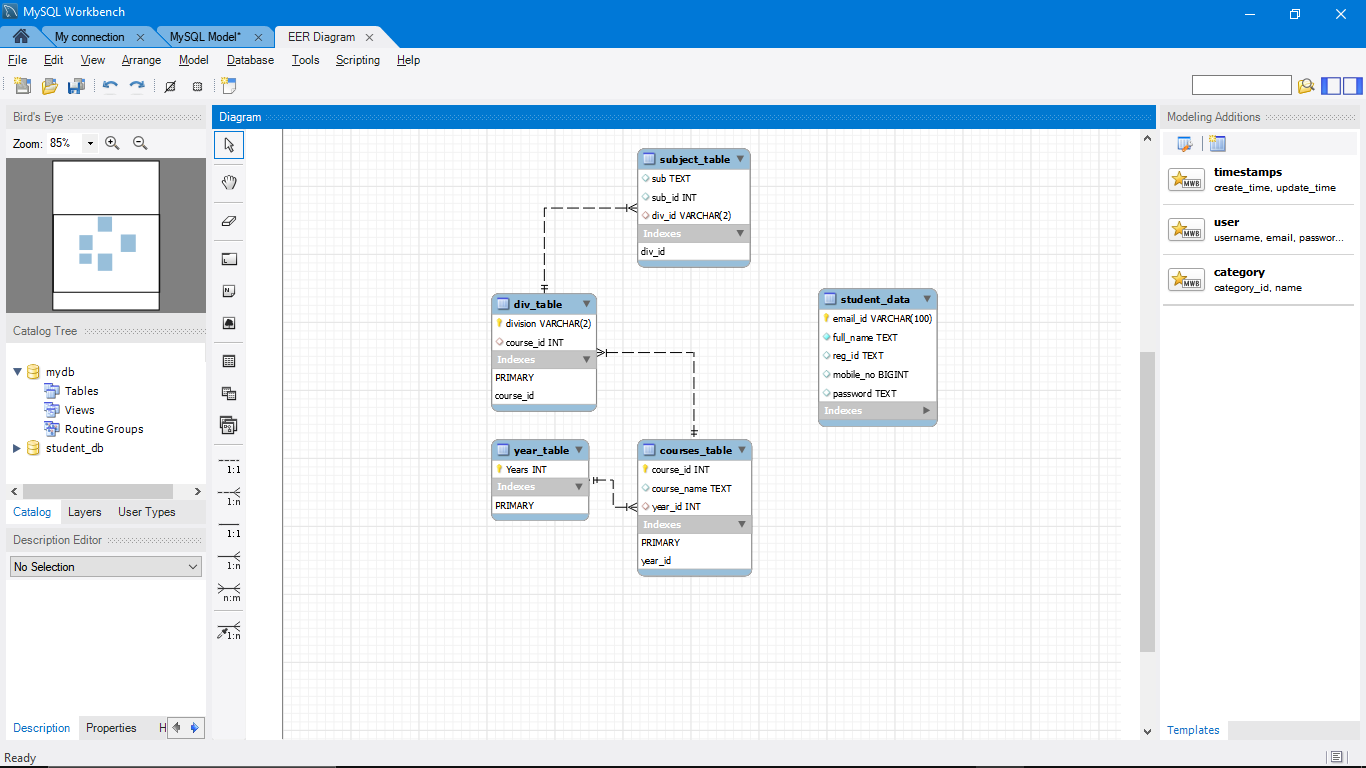


##### Registration Window



* + 1. **Manage registered students.**

****

****

# Ch.5.Conclusion

In conclusion, the College Android Application (SGGS app) Database and API Development project culminates as a testament to efficient data management and enhanced connectivity within the college community. This endeavor successfully bridges the gap between the expansive college database and the students' and faculty's access to critical academic information through the development of a robust API and an intuitive Android application interface.

By prioritizing seamless data accessibility and security, this project establishes a framework that empowers students to effortlessly retrieve attendance records, exam results, and real-time event notifications. The implementation of stringent authentication protocols and encryption measures ensures the confidentiality and integrity of sensitive data, fostering a secure environment for information exchange.

Furthermore, the system's testing phase rigorously validates its functionalities, performance, and user-centric design, guaranteeing a reliable and intuitive experience for end-users. The culmination of this project signifies a milestone in promoting informed decision-making, improved communication, and streamlined data access within the college community, laying the foundation for a more connected and engaged campus environment. Ultimately, this initiative sets the stage for scalable and efficient information exchange, fostering a community-driven approach to education and communication within the college ecosystem.

# Future Scope

The College Android Application (SGGS app) Database and API Development project lays a robust foundation for future enhancements and expansions, offering several avenues for continuous improvement and broader implementation within the college ecosystem.

Firstly, the project's modular structure enables the addition of new features and functionalities to meet evolving student and faculty needs. Future iterations might include enhanced collaboration tools, such as discussion forums or chat functionalities, fostering greater interaction and knowledge exchange among users. Moreover, expanding the app to incorporate features like course enrollment, fee payment, or academic resource access could further streamline administrative processes, benefiting both students and staff.

Additionally, scalability remains a key aspect for future development. As the college community grows, the system can be optimized to handle increased user loads and data volumes without compromising performance. Implementing efficient caching mechanisms and optimizing database queries can ensure responsiveness and reliability even during peak usage times.

Furthermore, the project's success lays the groundwork for potential integration with other educational platforms or systems. Seamless integration with learning management systems (LMS) or library databases could enrich the app's offerings, providing a comprehensive educational experience within a unified platform.

Lastly, continuous user feedback and iterative improvements are pivotal for the project's future. Incorporating user feedback loops allows for continuous refinement of features, user interface, and overall usability, ensuring that the application remains aligned with the evolving needs and expectations of the college community.

In essence, the project's future scope encompasses feature expansions, scalability enhancements, integration opportunities, and a commitment to iterative improvements based on user feedback, aiming to create a dynamic, responsive, and indispensable tool for the entire college ecosystem.

# References

* + - <http://docs.oracle.com/cd/B12037_01/server.101/b10739/create.ht> m
    - <http://docs.oracle.com/javafx/index.html>
    - <http://stackoverflow.com/questions/15482623/javafx-1-3-listview>
    - <http://stackoverflow.com/questions/8392119/javafx-1-3-crud-> using-listview
    - <http://docs.oracle.com/cd/E17802_01/javafx/javafx/1.3/docs/api>
    - https://[www.tutorialspoint.com/php/index.htm](http://www.tutorialspoint.com/php/index.htm)
    - https://[www.w3schools.com/php/DEFAULT.asp](http://www.w3schools.com/php/DEFAULT.asp)
    - “Database System Concepts” by Henry Korth.
    - “Mysql” by Paul Dubois.
    - James R Groff and Paul N Weinberg, “Complete reference SQL” Second Edition
    - Agnes Molnar and Adam Granicz , “ .Net with Visual C# 2010”.
    - Css files, from W3 schools.
    - http//w3 school.com/css\_file
    - From SGGS app
    - [http://www.en.wikipedia.org/wiki/Wireless\_security.](http://www.en.wikipedia.org/wiki/Wireless_security)