**Final Project Report**

**EduNexus (Virtual Classroom Reminder App)**



**Project Supervisor**

**Noureen Hameed**

**Submitted By**

**Group ID: F2402324E1**

**Hafiz Muhammad Hammad BC230209354**

**Software Projects & Research Section,**

**Department of Computer Sciences,**

**Virtual University of Pakistan**

|  |
| --- |
|  |



**CERTIFICATE**

This is to certify that Hafiz Muhammad Hammad (BC230209354) has worked on and completed their Software Project at Software & Research Projects Section, Department of Computer Sciences, Virtual University of Pakistan in partial fulfillment of the requirement for the degree of BS in Computer Sciences under my guidance and supervision.

In our opinion, it is satisfactory and up to the mark and therefore fulfills the requirements of BS in Computer Sciences.

**Supervisor / Internal Examiner**

Noureen Hameed

Supervisor,

Software Projects & Research Section,

Department of Computer Sciences

Virtual University of Pakistan

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature)

**External Examiner/Subject Specialist**

<<External Supervisor Name>>

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature)

**Accepted By:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_**

(For office use)

**EXORDIUM**

**In the name of Allah, the Compassionate, the Merciful.**

**Praise be to Allah, Lord of Creation,**

**The Compassionate, the Merciful,**

**King of Judgment-day!**

**You alone we worship, and to You alone we pray for help,**

**Guide us to the straight path**

**The path of those who You have favored,**

**Not of those who have incurred Your wrath,**

**Nor of those who have gone astray.**

**DEDICATION**

This work is dedicated to my parents, whose unwavering love and support have been my strength throughout this journey. I also dedicate this to my teachers, whose guidance and encouragement have been invaluable. Their belief in me has always inspired me to strive for excellence.

**ACKNOWLEDGEMENT**

First and foremost, I am deeply grateful to my parents for their constant prayers, love, and motivation. Their encouragement has been the foundation of all my achievements. I extend my heartfelt gratitude to my brother, Hafiz Muhammad Moaz, for his continuous support and belief in my abilities. His reassurance kept me going during challenging times.

I also wish to thank my cousin sister, Saba Shahdin, for her help during the project. Her insights and suggestions were truly valuable. Finally, I acknowledge all those who directly or indirectly contributed to the completion of this project. Your support means everything to me.

**PREFACE**

This project represents a significant milestone in my academic journey. The aim was to design and develop a Virtual Classroom Reminder App, enhancing productivity and time management in educational settings. The scope included features like user authentication, reminder management, and a notification system to ensure effective communication of classroom activities. The project is a step toward leveraging technology for better learning experiences. Through this report, I aim to detail the processes, technologies, and methodologies involved in the project’s development.

**TABLE OF CONTENTS**

|  |
| --- |
|  |

CHAPTER NO. 1

Software requirements specification (SRS Document)

1.1 scope

1.2 Functional & non-fucntional)

1.3 Use Case Diagram

1.4 Work Plan (Use MS Project to create Schedule/Work Plan)

CHAPTER NO. 2

designing the project (Design Document)

2.1 Introduction of Design Document

2.2 Data Flow Diagram (DFD)

2.3 Entity Relationship Diagram (ERD)

2.4 Database Design

2.5 Interface Design

CHAPTER NO. 3

IMPLEMENTATION

3.1 Frontend Development

3.2 Backend Development

CHAPTER NO. 4

SYSTEM FEATURES

4.1 Functional Features

4.2 Non-Functional Features

CHAPTER NO. 5

REMINDER’s FEATURES

5.1 Create

5.2 Read

5.3 Update

5.4 Delete

CHAPTER NO. 6

CONCLUSION AND FUTURE ENHANCEMENTS

**CHAPTER 1**

SOFTWARE REQUIREMENTS SPECIFICATION (SRS DOCUMENT)

**1.1 Scope of Project:**

The scope of this project is to develop a Virtual Classroom Reminder App that facilitates seamless interaction between students, teachers, and administrators. The system provides a centralized platform for managing classroom activities, assignments, and communication. Key features include user authentication, course management, attendance tracking, and notifications. The project aims to enhance learning outcomes by improving accessibility, organization, and collaboration in educational environments.

**1.2 Functional and non Functional Requirements:**

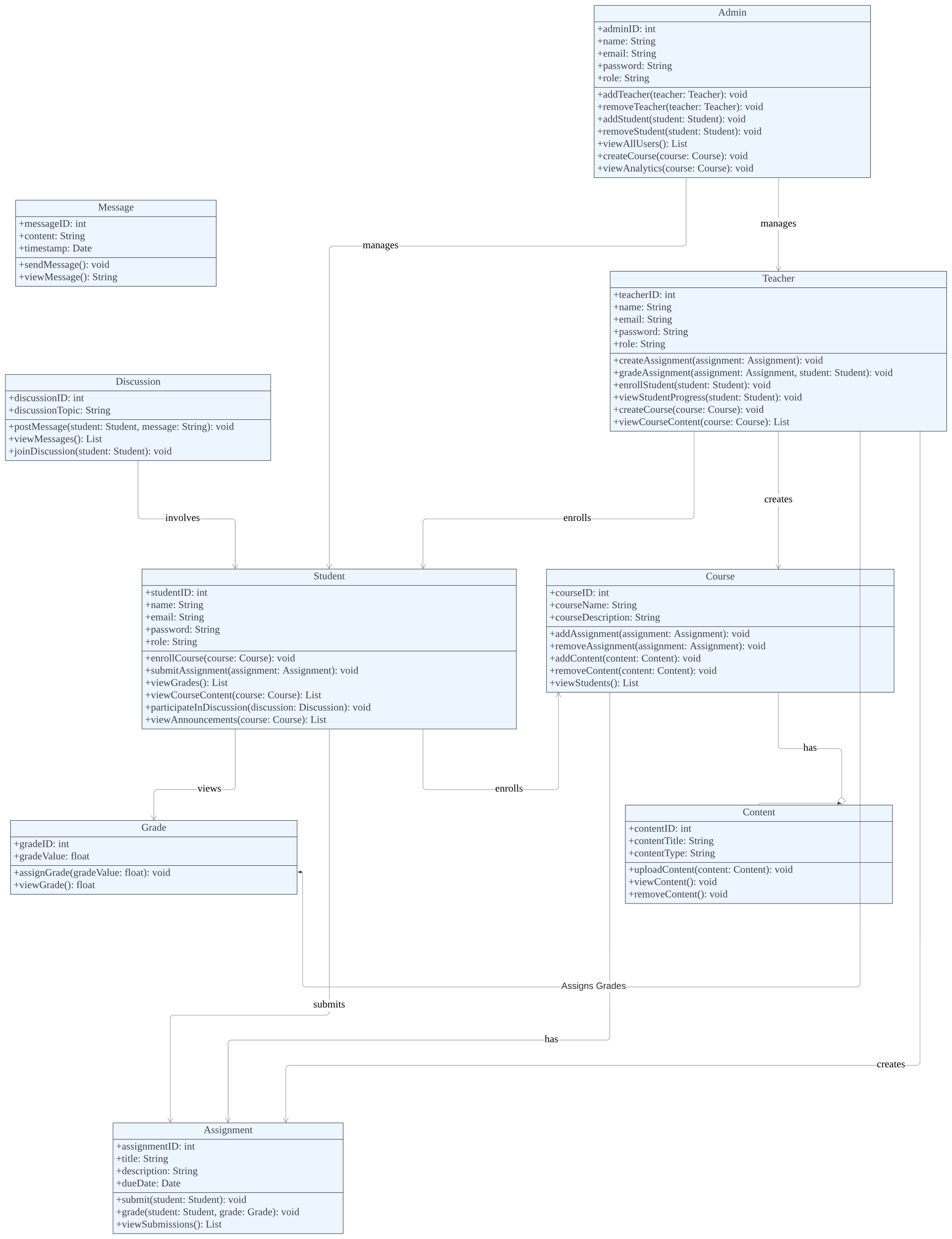
**Functional Requirements:**

* User authentication and role-based access control (Admin, Teacher, Student).
* Course creation, editing, and deletion by authorized users.
* Assignment submission and grading.
* Attendance tracking and reporting.
* Notifications and reminders for important events and deadlines.

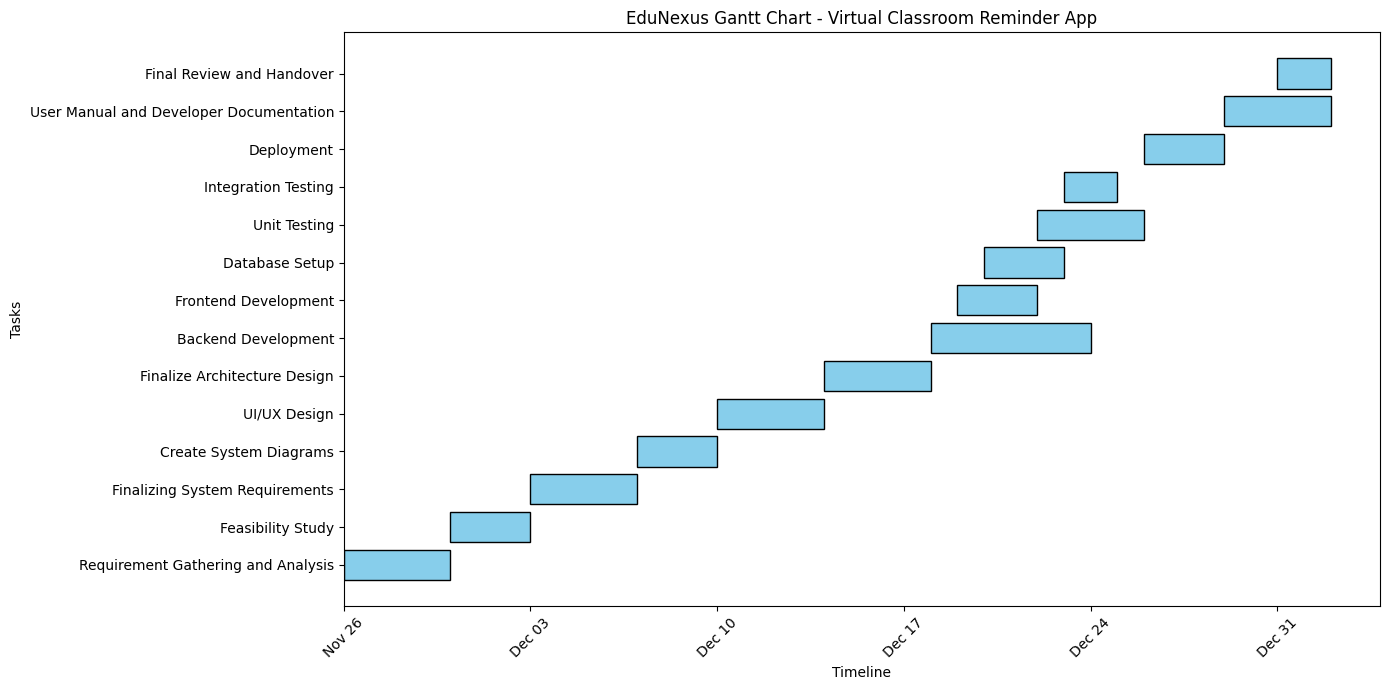
**Non-Functional Requirements:**

* The system should be scalable to support multiple users simultaneously.
* It must ensure data security and user privacy.
* The interface should be user-friendly and responsive.
* The system should provide reliable performance with minimal downtime.
* It must be compatible with common browsers and mobile devices.

**1.3 Use Case Diagram**



**1.4 Work Plan**



**CHAPTER 2**

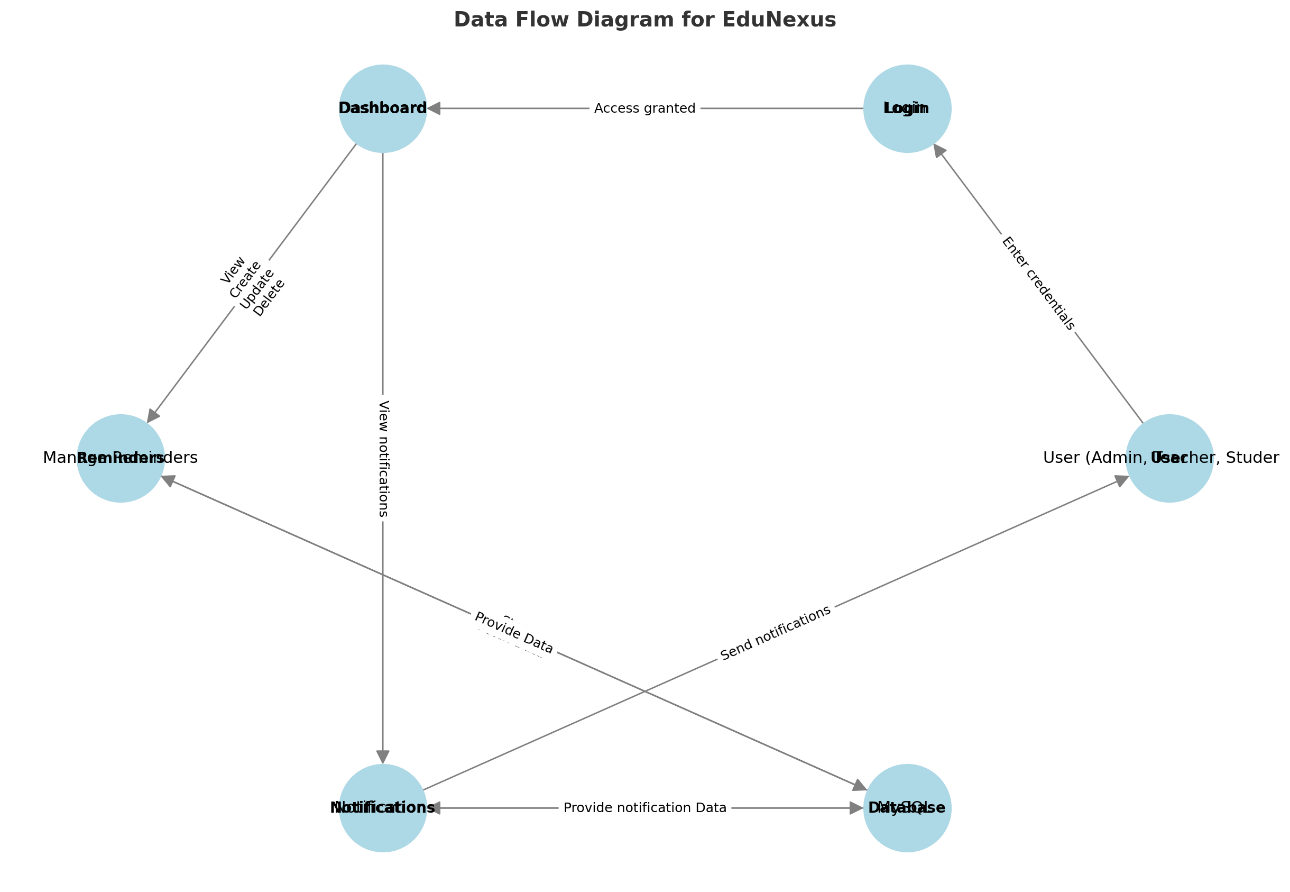
Designing the Project

(Design Document)

* 1. **Introduction of Design Document**

This design document outlines the architecture, data flow, and interface design of the EduNexus App. This phase defines the system’s blueprint, providing a foundation for the development process. It includes Data Flow Diagrams (DFD), Entity-Relationship Diagrams (ERD), and Database Design. This phase ensures clarity in system requirements and facilitates efficient implementation. The design phase is critical as it minimizes errors and aligns the development with project objectives.

* 1. **Data Flow Diagram (DFD)**



User (Admin, Teacher, Student)

Notifications

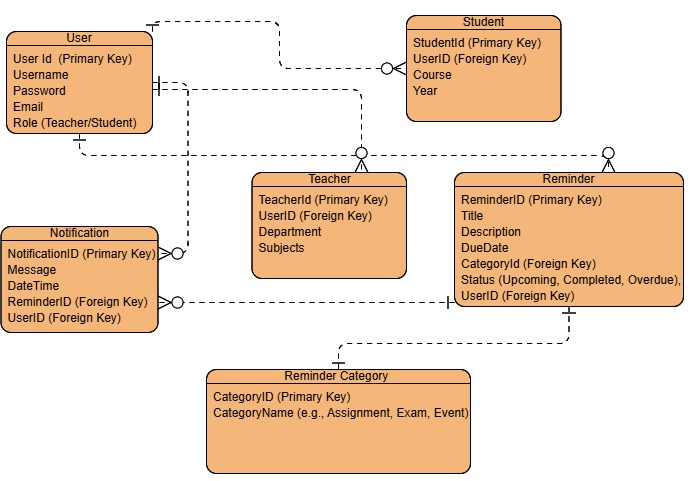
Database

Manage Reminders

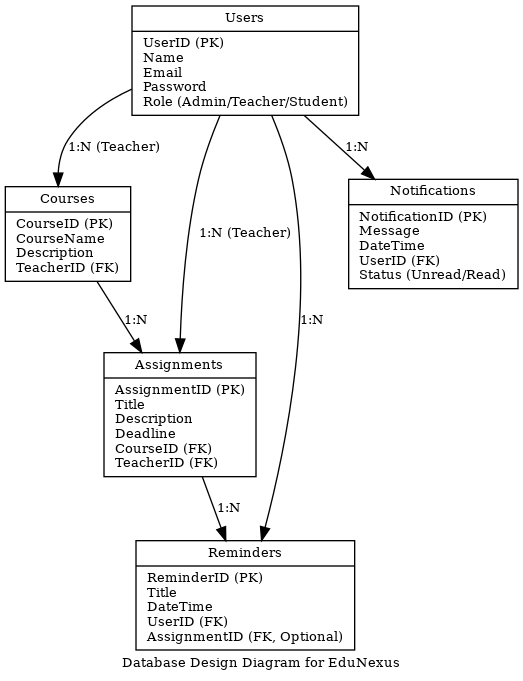
Login

Dashboard

* 1. **Entity Relationship Diagram (ERD)**



* 1. **Database Design**



* 1. **Interface Design**

Figure 2: Login Page

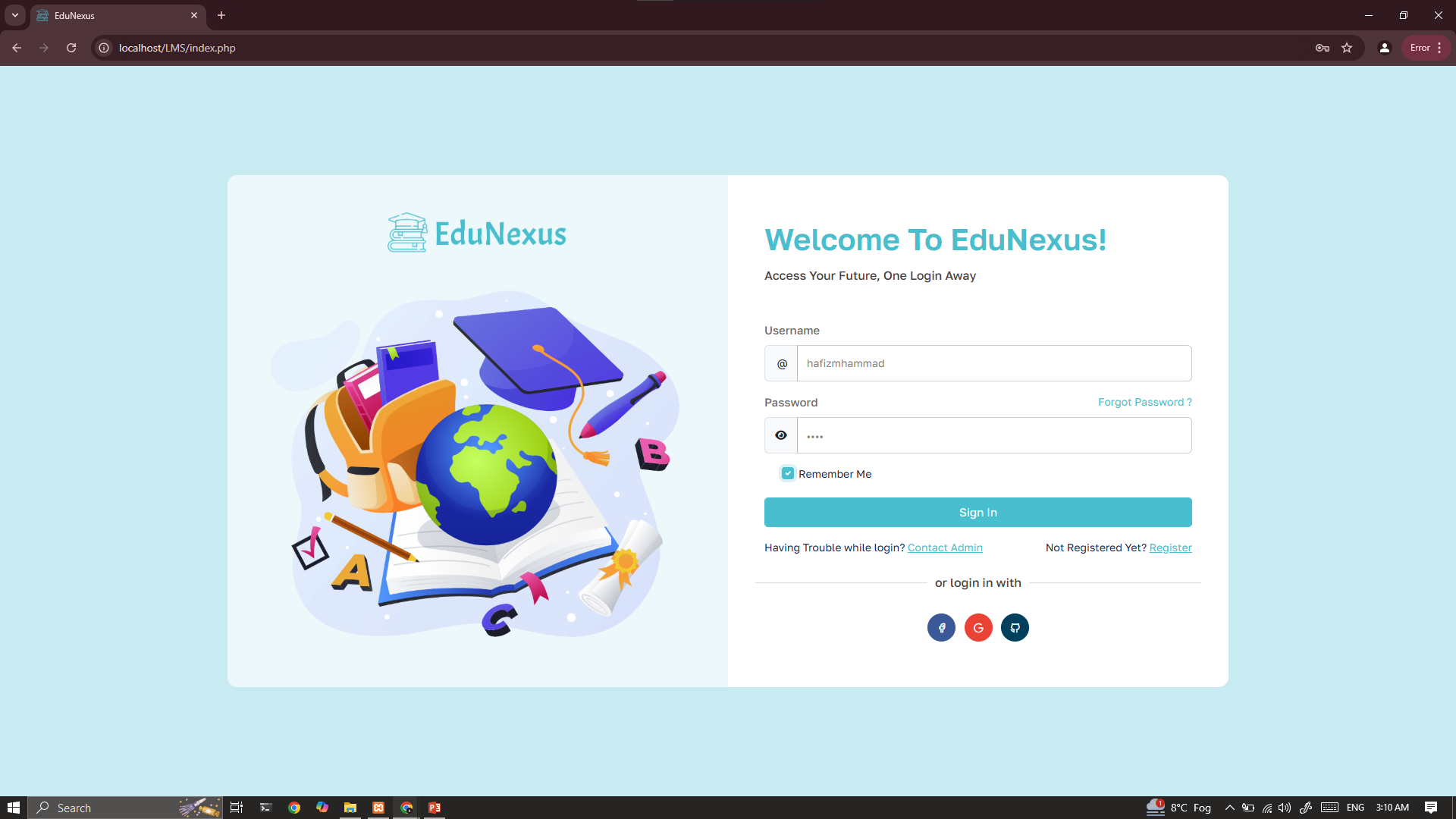


Figure 1: Registeration Page

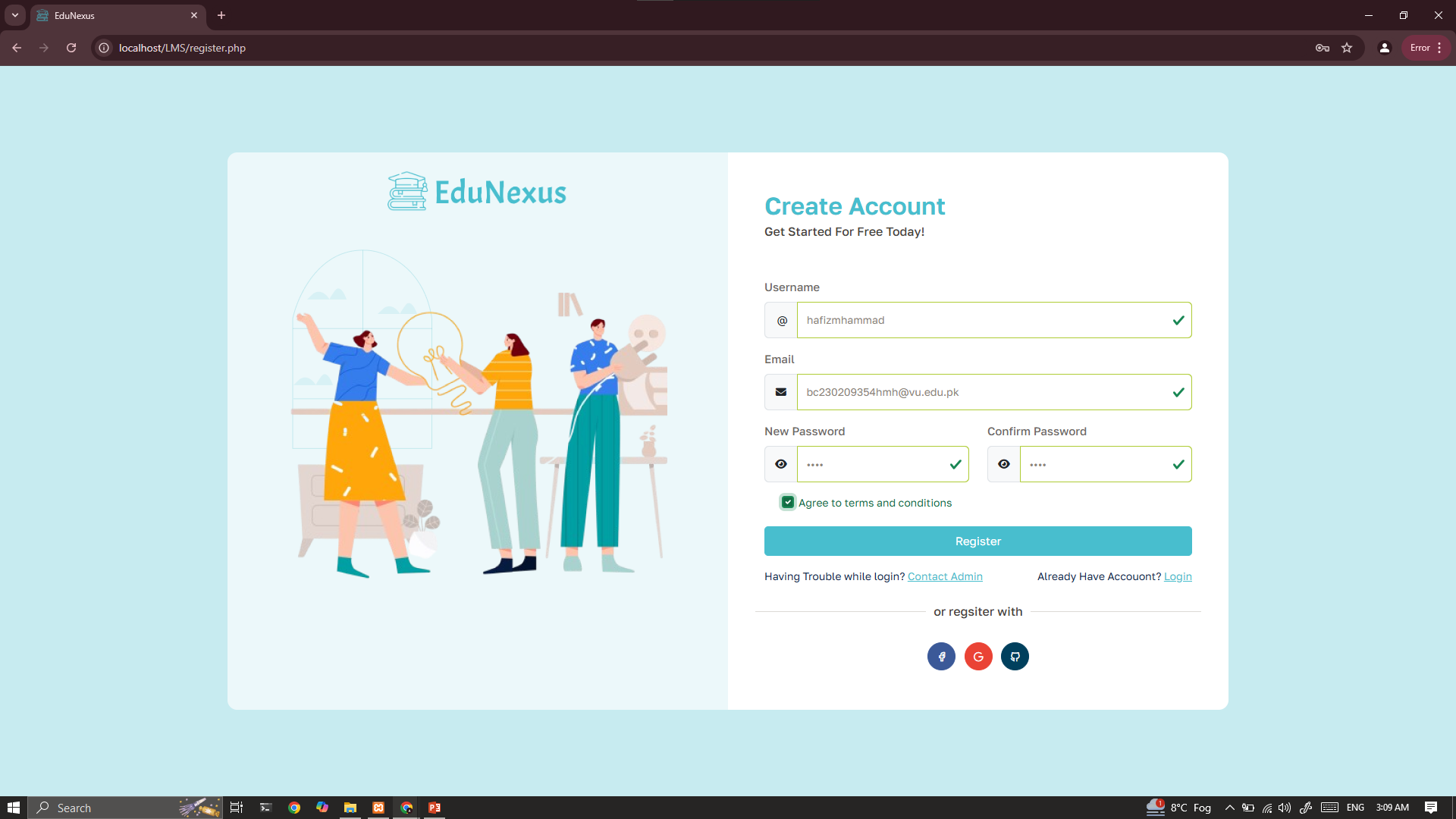


Figure 3: Dashboard

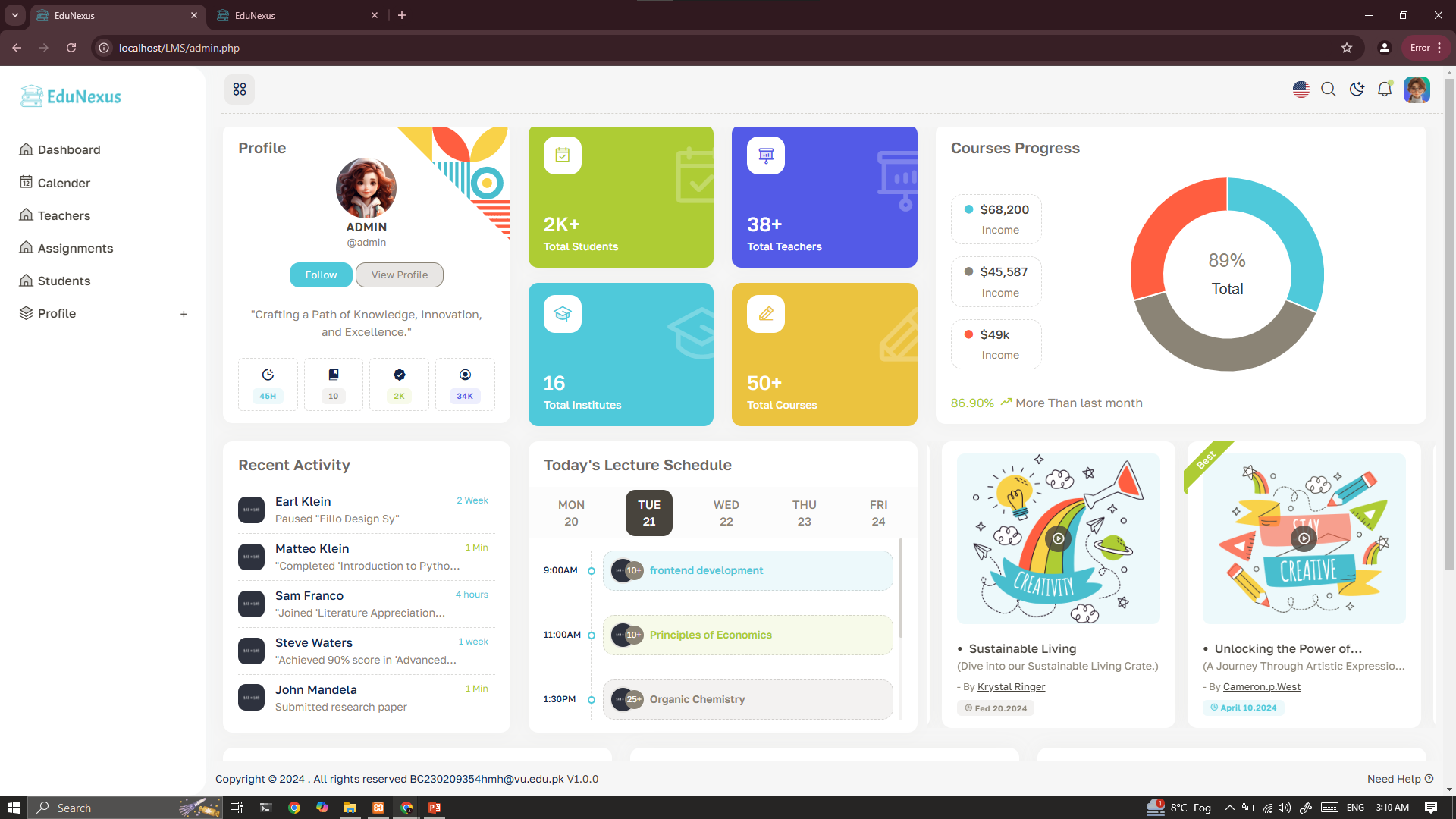




Figure 4: Notification Panel

Figure 6: Calendar

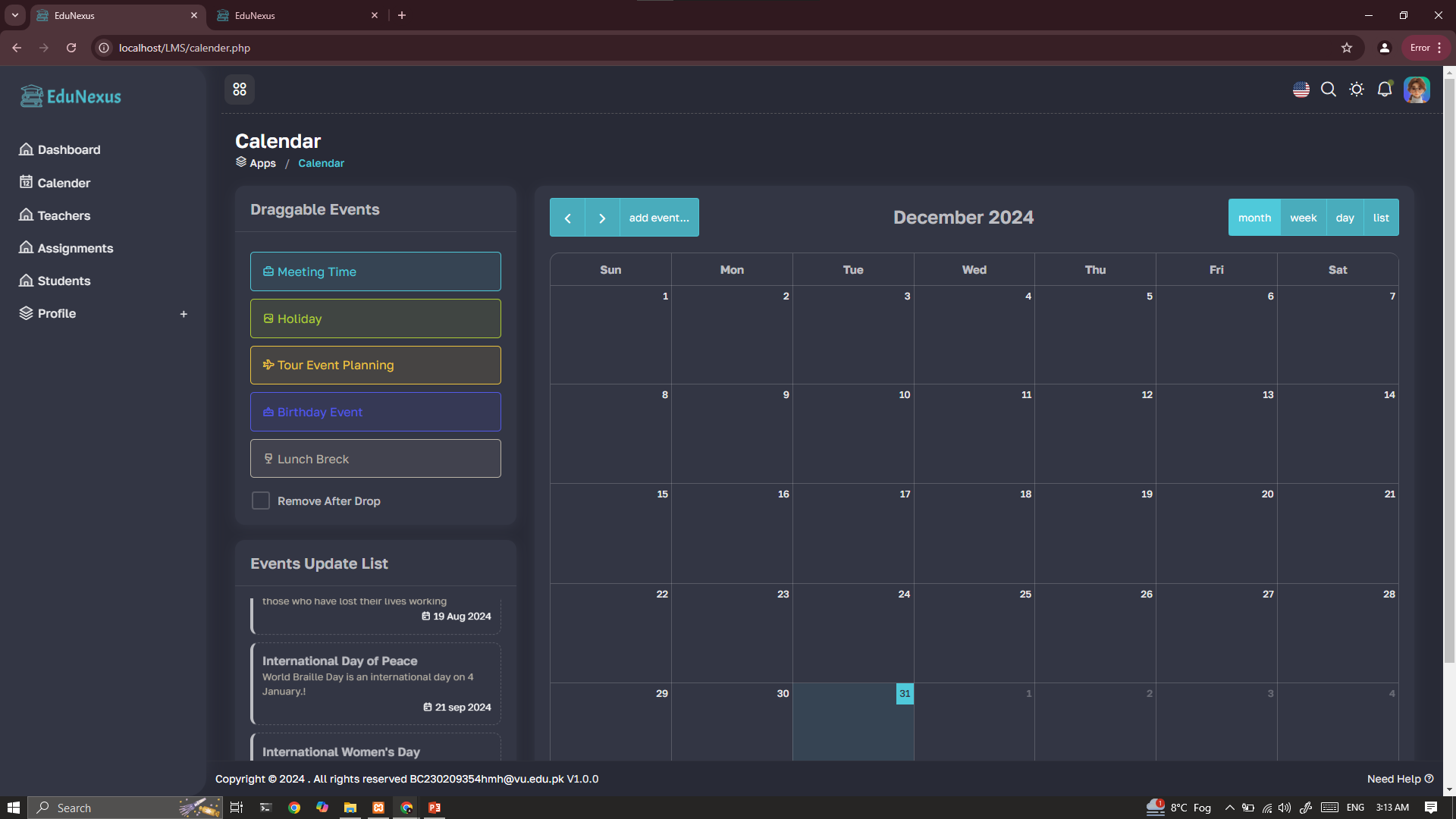


Figure 5: Assignments

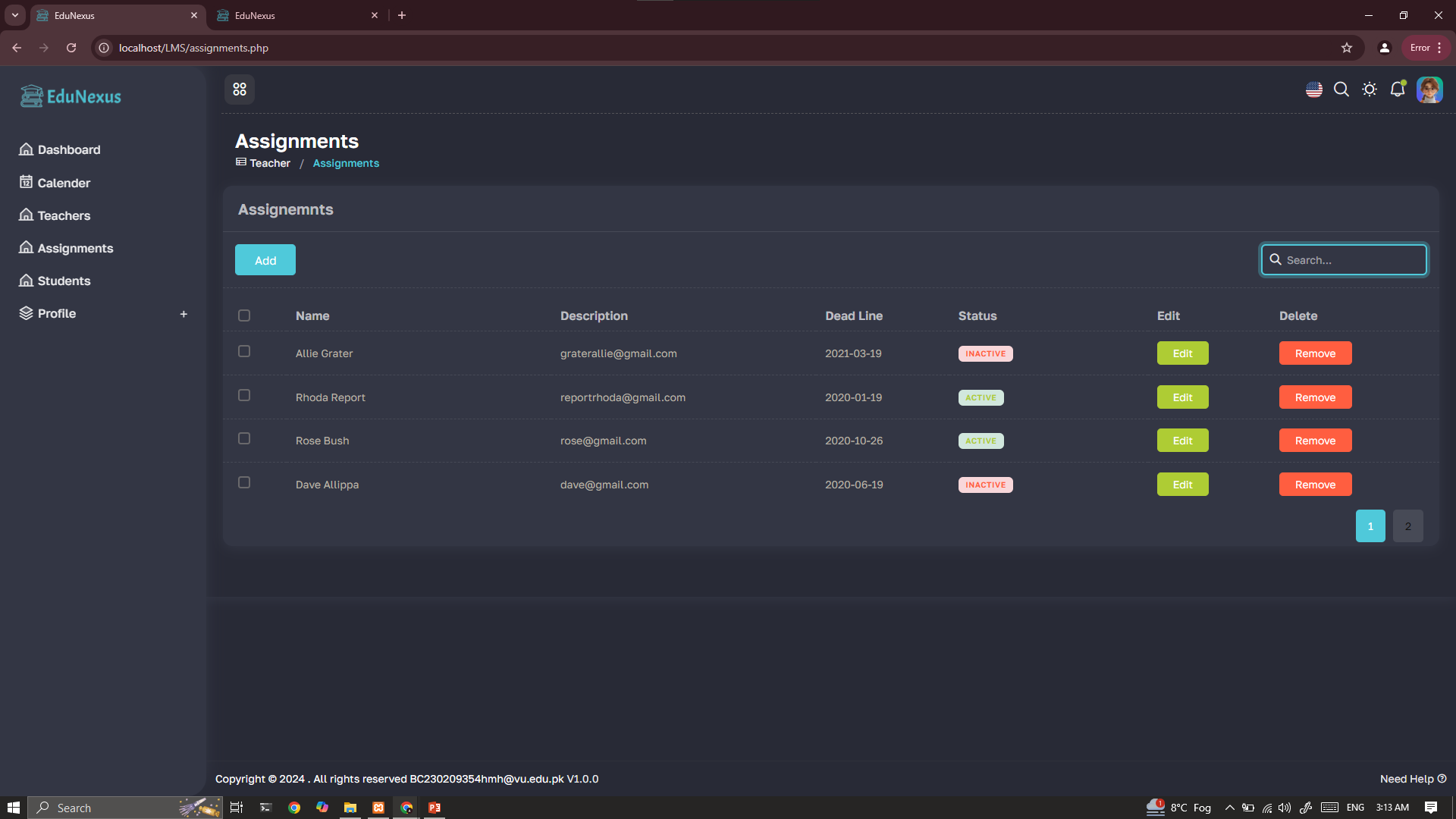


Figure 7: Notification Settings

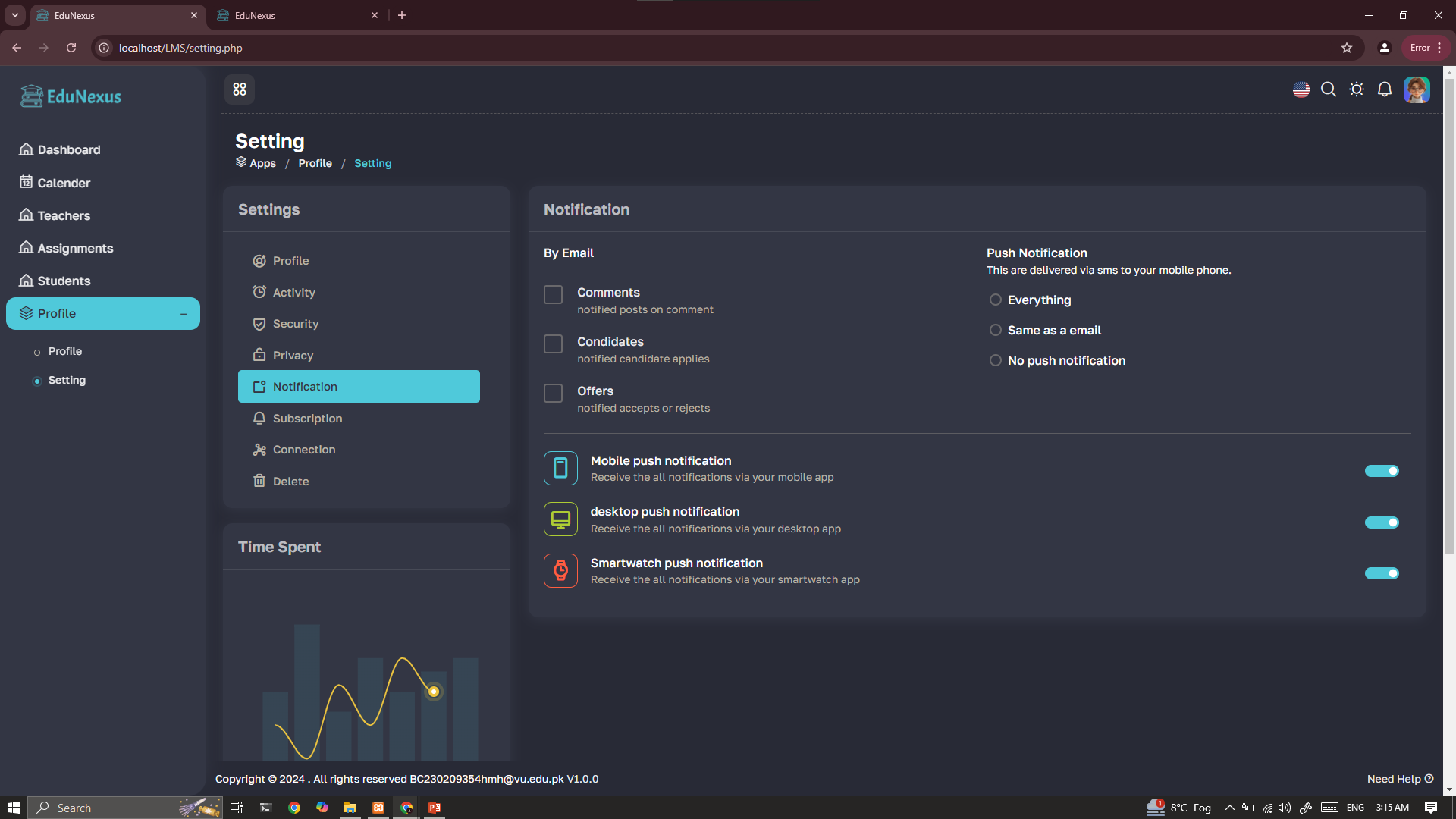
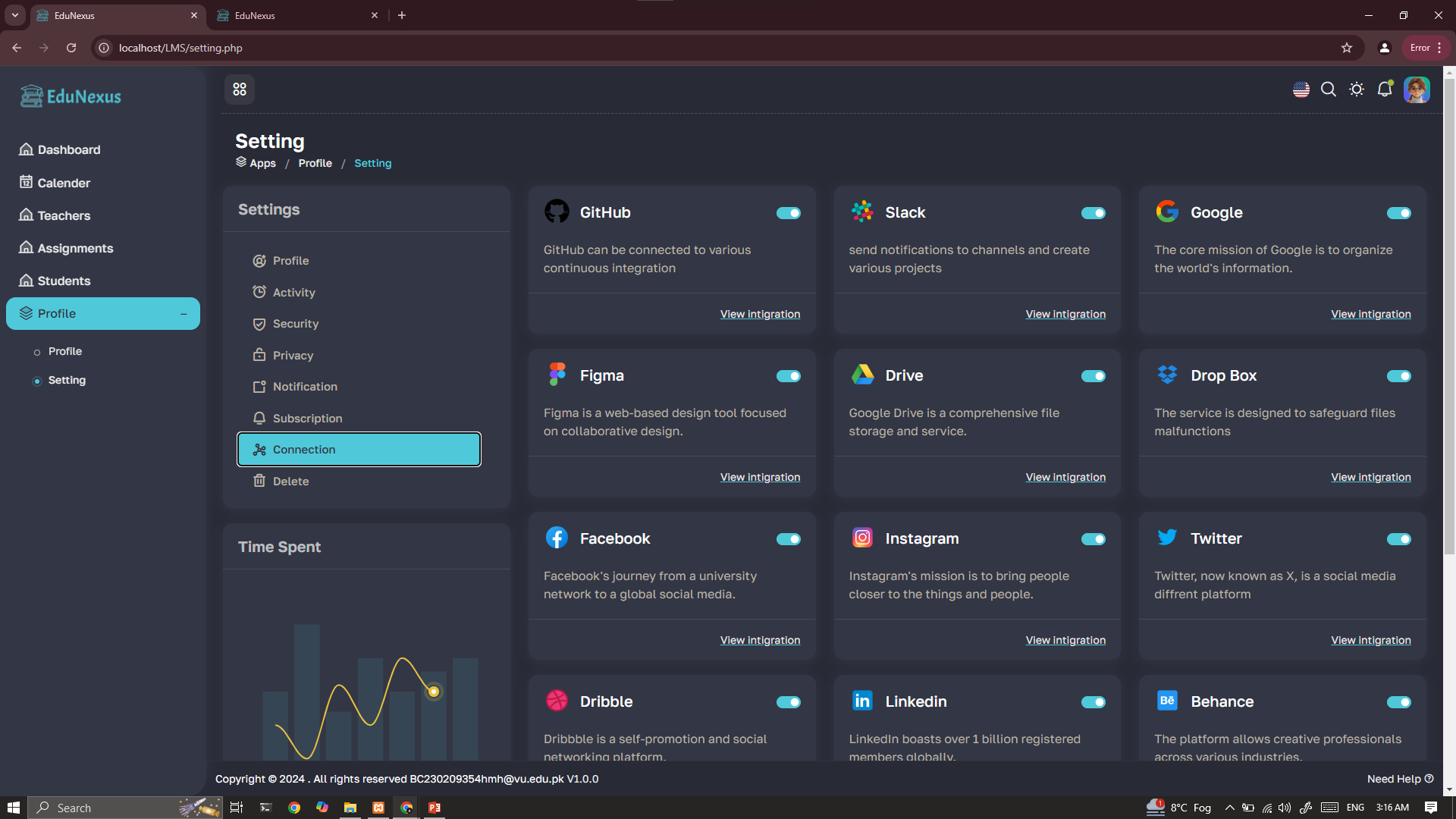


Figure 8: Connections Section



**CHAPTER 3**

IMPLEMENTATION

1. **Frontend Development**

The frontend of the Virtual Classroom Reminder App was developed using HTML, CSS, and JavaScript. The design ensures simplicity, ease of use, and responsiveness. Bootstrap was used to achieve a modern and adaptive layout.

1. **HTML** was used to structure the content and layout of the app, ensuring that all necessary elements such as forms, buttons, navigation menus, and reminder lists are well-organized.
2. **CSS** was utilized to style the app, making it visually appealing while maintaining a consistent design. Custom styles were applied to buttons, cards, forms, and other UI elements to make the interface user-friendly.
3. **JavaScript** was used to add interactivity and dynamic elements to the app, such as reminder creation, updates, and real-time notifications. JavaScript was also used for form validation to ensure data integrity.
4. **Responsive Design**: The app was built to be mobile-first, ensuring that users can seamlessly interact with it on any device. The layout adjusts dynamically based on the screen size, providing a consistent experience across different devices and browsers.
5. **Backend Development**

The backend was developed using PHP, with MySQL as the database. XAMPP provided the development environment, leveraging Apache for server-side processing. The backend handles user authentication, reminder management, and notification logic.

1. **PHP**: The backend logic was primarily written in **PHP**, which interacts with the frontend through AJAX calls to dynamically update the user interface without reloading the page. PHP was used for managing user sessions, handling form submissions, and implementing the reminder logic.
2. **User Authentication**: PHP handles user authentication by verifying login credentials, managing user sessions, and ensuring role-based access control (Admin, Teacher, and Student). The login system uses **password hashing** for added security.
3. **Reminder Management**: The backend also manages all reminder-related operations. It processes requests to create, update, view, and delete reminders. It checks if the logged-in user has the necessary permissions to perform these actions (e.g., only teachers and admins can create or modify reminders).
4. **Notification Logic**: PHP also manages the logic for sending reminders and notifications. The backend ensures that notifications are sent to users (students and teachers) at the appropriate times based on the reminder settings.
5. **MySQL Database**: The app uses **MySQL** to store data such as user credentials, reminder details, courses, assignments, and notifications.
   1. The database schema includes tables for users (Admin, Teacher, Student), reminders, notifications, and course assignments.
   2. Relationships between users, courses, and reminders are stored in relational tables to ensure efficient data retrieval and management.
6. **XAMPP Development Environment**: The backend development was carried out using **XAMPP**, an open-source platform that provides all the necessary tools for PHP development. It includes:
   * 1. **Apache** as the web server for processing PHP scripts and serving the frontend content.
     2. **MySQL** as the database engine to store application data.
     3. **phpMyAdmin** to facilitate database management and testing during the development phase.
7. **Server-Side Processing**: The backend is responsible for processing requests sent by the frontend, including user registration, login, reminder creation, assignment submissions, and real-time notifications. Data is dynamically fetched and processed based on the role of the user, ensuring that only authorized users can access specific features.
8. **Notification System**: The backend also manages the notification system, which sends real-time alerts to users about their reminders. Notifications are triggered at set times, based on the reminder date and time. PHP scripts periodically check for upcoming reminders and push notifications to users.

**CHAPTER 4**

SYSTEM FEATURES

1. **Functional Features**
2. **User authentication for role-based access.**

* EduNexus includes a robust user authentication system that ensures proper access control based on user roles (Admin, Teacher, and Student).
* Admins can manage all users, while teachers have access to course materials, reminders, and student progress, and students can view assigned tasks and reminders.
* Users authenticate via username and password, with session management to ensure security during usage.

1. **Creation and management of reminders.**

* Users (Admin, Teacher, and Student) can create and manage reminders for various activities like assignment deadlines, class schedules, and quizes.
* Teachers and admins can create reminders, while students can only view or acknowledge them.
* Reminders can be updated or deleted by authorized users, ensuring that they are always accurate and relevant.

1. **Notifications for scheduled tasks.**

* EduNexus features a real-time notification system that sends alerts to users about upcoming tasks, assignments, quizes, or class schedules.
* Notifications can be customized by users to be sent via email, SMS, or in-app messages.
* Admins and teachers can configure notification settings, ensuring that students and other stakeholders are aware of their responsibilities.

1. **Dashboard summarizing upcoming reminders.**

* The dashboard provides a clear and concise summary of all upcoming reminders, such as classes, assignments, and quizes.
* Each user (Admin, Teacher, or Student) has a personalized view of their tasks and deadlines.
* Users can filter reminders by date, course, or type, allowing for easy tracking of upcoming events.

1. **Non-Functional Features**
2. **Scalability to support multiple users.**

* The platform is designed to handle an increasing number of users without compromising performance.
* As the number of students, teachers, and admins grows, EduNexus will be able to efficiently scale by optimizing the underlying infrastructure and server resources.
* The system supports multiple concurrent users, ensuring smooth performance during peak usage.

1. **Secure data encryption and storage.**

* EduNexus ensures the protection of sensitive data through encryption techniques, both during data transmission (using SSL/TLS) and at rest.
* User credentials, personal information, and other confidential data are encrypted to prevent unauthorized access and data breaches.
* The app uses secure databases (MySQL) with advanced security protocols, including hashed passwords and user authentication tokens.

1. **High reliability with minimal downtime.**

* The system is designed to be highly reliable, with minimal downtime, ensuring users can access their reminders and related features at any time.
* EduNexus utilizes redundancy measures, including load balancing and failover systems, to prevent service disruption in case of hardware or software failures.
* Regular backups are conducted to prevent data loss and ensure quick recovery in case of an emergency.

1. **Compatibility with multiple devices and browsers.**

* EduNexus is built to be responsive and compatible with various devices, including desktops, laptops, tablets, and smartphones.
* The platform supports all major web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge, ensuring accessibility for all users.
* The user interface (UI) adapts to different screen sizes and resolutions, providing an optimal experience regardless of the device or browser used.

**CHAPTER 5**

REMINDER’s FEATURES

1. **Features:**

The EduNexus application allows users to set, manage, and receive reminders for important tasks related to their courses, assignments, and quizes. Below are the key features associated with the reminder functionality:

1. **Create**

* **Title**: Users can create a reminder by providing a clear and descriptive title. For example, "Assignment Submission Deadline" or "Upcoming Quiz."
* **Date and Time**: Reminders can be scheduled for specific dates and times, ensuring that students, teachers, and admins are notified at the appropriate time.
* **Additional Details**: Users can include additional information, such as the course name, subject, or the assignment associated with the reminder.
* **Reminder Type**: Users can select whether the reminder is for a class, assignment, quiz, or any other activity, allowing for better organization and prioritization.

1. **Read**

* **Dashboard View**: Users can view their active reminders on a central dashboard. This dashboard displays reminders sorted by date, with the option to filter by course, assignment, or other categories.
* **Reminder Alerts**: A notification system informs users of any upcoming reminders, ensuring they never miss important deadlines or events.
* **Reminder History**: Users can access past reminders to track previous activities and deadlines, providing an organized history of tasks.

1. **Update**

* **Edit Existing Reminders**: Users can update existing reminders by modifying the title, date, time, and additional details. This ensures that reminders are always accurate and up to date.
* **Adjust Time/Date**: If there is a change in the deadline or event schedule, users can easily adjust the reminder to reflect the new date and time.
* **Course/Assignment Changes**: Teachers or admins can update the reminder based on changes in the course schedule or assignment deadlines.

1. **Delete**

* **Remove Unnecessary Reminders**: Users can delete reminders that are no longer needed, such as past assignment deadlines or canceled events, to maintain a clutter-free dashboard.
* **Bulk Deletion**: Admins and teachers can delete multiple reminders at once, especially when a large number of reminders are no longer relevant (e.g., after a quiz or assignment completion).

1. **User Roles and Permissions**

In EduNexus, there are three primary user roles: Admin, Teacher, and Student. Each role has specific access to various features, including reminders, courses, assignments, and notifications.

1. **Admin**

* **Full Access**: Admins have complete control over the app, including managing all users, creating and editing reminders, assigning courses, and assigning teachers to courses.
* **Add/Delete Users**: Admins can add students to courses, assign teachers, and remove users as necessary.
* **Notification Management**: Admins can configure global notification settings for all users.
* **Manage Assignments**: Admins have the authority to assign, update, and delete assignments.

1. **Teacher**

* **Create and Manage Reminders**: Teachers can create, view, edit, and delete reminders related to their courses and assignments. They can set reminders for upcoming classes, assignment deadlines, and quiz dates.
* **Assign Students to Courses**: Teachers can add students to their courses and track their progress.
* **Assignment Creation**: Teachers can assign assignments to students, setting deadlines and providing additional details for each task.

1. **Student**

* **View Reminders**: Students can view reminders created by teachers for their courses, including class schedules, assignment deadlines, and quizes.
* **Receive Notifications**: Students receive notifications for upcoming reminders, assignments, and any changes made by the teacher or admin.
* **Mark Reminders as Completed**: Students can mark reminders as completed once they have fulfilled the task, such as submitting an assignment.

1. **Notification System**

The EduNexus notification system ensures that users receive timely reminders about their courses, assignments, quizes, and other events:

* **Real-Time Notifications**: Users will be notified of reminders and important updates in real time via email, SMS, or in-app notifications.
* **Customizable Alerts**: Users can customize how and when they want to receive notifications (e.g., a day before an assignment is due).
* **Automatic Reminders**: As the date and time for a reminder approaches, the system will automatically send out reminders, ensuring users stay informed.

**CHAPTER 6**

CONCLUSION AND FUTURE ENHANCEMENTS

1. **Conclusion**

EduNexus successfully addresses the need for managing classroom activities efficiently. Through its user-friendly interface and robust features it enhances productivity for both students and teachers. Theis ensures timely reminders and effective management of schedules.

1. **Future Enhancements**
2. **Integration with Calendar Applications:** The app can be enhanced by allowing users to sync reminders with Google Calendar or Outlook.
3. **Mobile Application Development:** Developing mobile apps for Android and iOS platforms will improve accessibility.
4. **Multi-Language Support:** Adding support for multiple languages will cater to a broader user base.
5. **AI-Powered Recommendations:** AI algorithms could suggest optimal scheduling times or task prioritization based on user behavior.
6. **Enhanced Analytics:** Providing detailed analytics and reports on user activity and task completion will offer valuable insights..

**REFERENCES**

* **W3C.** (2018). "HTML5 Specification." *World Wide Web Consortium (W3C)*. Retrieved from <https://www.w3.org/TR/html5/>.
* **Eich, B.** (2019). "JavaScript: The Good Parts." O'Reilly Media.
* **Duckett, J.** (2011). HTML and CSS: Design and Build Websites. Wiley. ISBN: 978-1-118-20691-1.
* **Zeldman, J.** (2017). "Designing with Web Standards." New Riders Publishing.
* **PHP Group.** (2020). "PHP: Hypertext Preprocessor Manual." PHP.net. Retrieved from <https://www.php.net/manual/en/>
* **Schildt, H.** (2020). "JavaScript: A Beginner’s Guide." McGraw-Hill Education.
* **XAMPP Development Team.** (2020). "XAMPP: The Open-Source Cross-Platform Web Server Solution Stack." Apache Friends. Retrieved from <https://www.apachefriends.org/>
* **Beighley, L., & Morrison, M.** (2012). "Head First PHP & MySQL: A Brain-Friendly Guide." O'Reilly Media.
* **W3C.** (2021). "Cascading Style Sheets (CSS) Specification." World Wide Web Consortium (W3C). Retrieved from <https://www.w3.org/TR/css-2021/>
* **Jones, D.** (2021). Learn to Code with JavaScript. SitePoint. eText ISBN: 9781098124786, Print ISBN: 1098124782.

**APPENDIX**

Supplementary Material