International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Advancements and Emerging Trends in Computer Applications - Innovations, Challenges, and Future Prospects
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Design and Implementation of a Real-Time Online Learning Environment with Integrated Classroom and Assignment Features

Shrawani C. Bali

PG Student, Department of Computer Application, G. H. Raisoni University, Amravati, Maharashtra, India

ABSTRACT

Global events such as the COVID-19 pandemic have fueled the rapid growth of digital learning environments, necessitating the development of integrated, scalable, and user-friendly virtual classroom solutions. This paper presents the design and development of a cloud-based education management system that uses the HTML, CSS, JavaScript, React.js and MongoDB to integrate classroom management and assignment workflows. To enhance student-teacher interaction, the platform incorporates Jitsi Meet for live virtual class integration, real-time chat and Q&A, access to recorded sessions, and automated class scheduling and reminders.

Simultaneously, the system integrates robust assignment and homework management features, including task creation, submission tracking, plagiarism detection, grading, and resubmission capabilities. This system aims to close gaps in fragmented systems by being both learner-friendly and administratively simple.

KEYWORDS: *Cloud-Based Education System, Virtual Classroom Integration, Jitsi Meet in E-Learning, Assignment & Homework Automation, Plagiarism Detection in Education*

I. INTRODUCTION

With the rise of digital learning, online classroom management systems have become essential in modern education. These platforms help educators organize classes, keep track of attendance, and engage with students in a more flexible and efficient way. For students, they offer a convenient way to access learning materials and communicate with their teachers. This kind of seamless communication makes learning more interactive and helps create a more connected educational experience for everyone.

However, when it comes to managing assignments, traditional methods can often be a hassle. Teachers might use separate tools to create assignments, track deadlines, grade work, and give feedback. This fragmented approach can lead to confusion, missed deadlines, and inconsistent grading, making the whole process more complicated than it needs to be.

The goal of this research is to design and implement a Classroom and Assignment Management System using modern web technologies like HTML, CSS, JavaScript, React.js, and MongoDB. The aim is to improve communication between teachers and students, simplify task tracking, and create a more engaging virtual learning environment.

While building this system, the focus wasn't just on making it functional – it was equally important to make it easy and enjoyable to use. The interface is designed with simplicity and clarity in mind, so users can navigate the platform effortlessly. Whether someone is using a laptop, tablet, or phone, the layout adjusts smoothly to fit any screen size, ensuring a consistent experience across devices.

II. RELATED WORK :

Over the past few years, the shift to digital and hybrid learning has led to a surge in online classroom tools. Many platforms have attempted to solve the challenges of remote education, offering various features like video conferencing, assignment tracking, and communication tools. While each system has its strengths, most fall short in providing a seamless, fully customizable experience – especially for smaller institutions or independent educators.

- **Google Classroom:**

Google Classroom is widely used, especially in schools that are part of the Google ecosystem. It makes it easy for teachers to assign and collect work, share resources, and provide feedback. However, video classes are handled through Google Meet as a separate service, which sometimes creates a disconnected experience. It's a solid choice for basic classroom management, but it doesn't offer much flexibility or deep customization.

- **Microsoft Teams for Education:**

Microsoft Teams combines everything—chat, video meetings, file sharing, and assignments—all in one place. It's particularly useful for schools and colleges already using Microsoft Office. That said, it can be overwhelming for new users, especially younger students or those not used to Microsoft products. The learning curve and dense interface are common points of feedback.

- **Moodle:**

Moodle is one of the oldest and most customizable open-source learning management systems. It's packed with features and is used by many universities worldwide. The downside? Its interface feels a bit outdated, and it can be hard to customize without technical expertise. It's powerful but not the most user-friendly platform, especially for smaller teams.

- **Canvas LMS:**

Canvas stands out with its clean, modern interface and comprehensive feature set. It supports integrations with third-party tools like Zoom, Turnitin, and Google Drive, making it very versatile. It's popular in higher education, but for smaller setups, the pricing and complexity may be a barrier.

➢ **Edmodo (Now Discontinued):**

Edmodo was designed to look and feel like a social network for education. It was easy to use and encouraged interaction between students and teachers. However, it lacked the depth and scalability needed for long-term use in larger or more complex educational environments, and the platform was eventually shut down in 2022.

III. DATA AND SOURCES OF DATA :

To design and test our Classroom and Assignment Management System, we used a mix of data we gathered ourselves and insights from existing platforms. This helped us create a system that's both practical and user-friendly.

1. **Primary Data (Data We Collected Ourselves) :**This is the data we gathered directly from people using the system (teachers, students, and admins).

➢ **Source:** We talked to teachers, students, and school staff to understand their needs and challenges with online learning tools.

- **Teachers** shared their struggles with managing assignments and feedback.
- **Students** told us they wanted an easy-to-use platform with reminders.
- **Admins** gave us insights into managing schedules, records, and communication between everyone.

➢ **Source for Testing:** We tested our system with 15 students and 3 teachers.

- **Data collected:** How often they logged in, how many assignments were created or submitted, and how well features like video calls and notifications worked.
- **Source for Feedback:** After testing, we asked users for feedback to improve the design (like adjusting button visibility or fixing issues with smaller screens).

2. **Secondary Data (Data from Other Platforms)**: **This is the data we collected by researching existing platforms to see what works well and what doesn't.**

➢ **Source:** We reviewed popular platforms in online learning to understand their strengths and weaknesses.

- **Google Classroom:** It's good for assignments, but users feel disconnected when switching between Google Meet and Classroom for video calls.
- **Microsoft Teams for Education:** It has a lot of features, but the interface can be hard to use, especially for younger students or those not familiar with Microsoft tools.
- **Moodle:** A powerful tool used by universities, but it's complex and hard to customize without technical skills.
- **Canvas LMS:** It's modern and feature-rich, but it's too expensive and complicated for smaller schools or individual teachers.
- **Edmodo:** It used to be popular but didn't scale well, and was eventually discontinued.

3. **Technical Resources (Data from Technology Guides): We also used official guides and documentation from the tools we used to build the system.**

➢ **Source:** Official documentation from:

- **React.js** for building the interface
- **MongoDB** for storing user and assignment data
- **Jitsi Meet** for live, virtual classes

IV. RESEARCH AND METHODOLOGY :

A. Research Purpose

The purpose of this project is to build a web-based system that helps manage online classrooms and assignments. It includes features like live classes using Jitsi, class schedules, reminders, recorded sessions, real-time chat, and tools for giving and checking homework.

B. Research Questions

➢ Can virtual classes improve learning and participation?
➢ Do reminders and schedules help students attend on time?
➢ Does live chat make online classes more interactive?
➢ How well does the system help with giving, submitting, and checking assignments?
➢ Can plagiarism checks improve the quality and honesty of student work?

C. Method Used

1. System Development:

We used HTML, CSS, JavaScript, React.js for the front end and Node.js with MongoDB for the back end. Jitsi is used for live classes, and extra tools are added for notifications, grading, and plagiarism checks.

2. Testing:

The system was tested by teachers and students. We collected feedback using forms and observed how they used the system.

3. Data Collection

➢ **Quantitative data:** Number of classes attended, assignments submitted, and user ratings.
➢ **Qualitative data:** Comments and suggestions from users.

4. Data Analysis:

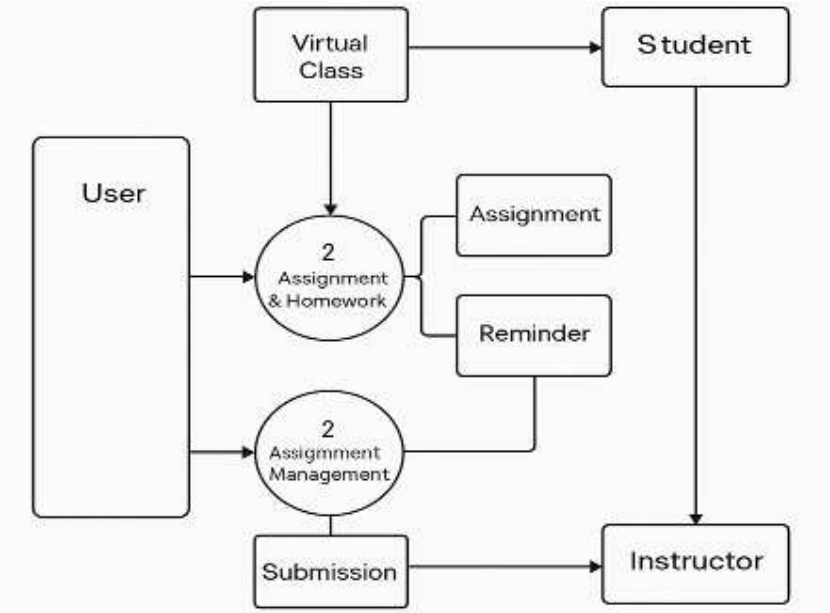
We used basic charts and summaries to understand the data and find out what worked well and what needs improvement.

D. Tools Used

Features	Tools/Tech Used
Frontend	HTML, CSS, JavaScript, React
Backend	Node.js, MongoDB
Live Class	Jitsi
Notifications	Email, In-app alerts
Plagiarism Check	Online API

E. Ethics :

All participants agreed to be part of the study. Their data was kept private and used only for research.



Data flow Diagram

V. RESULTS AND DISCUSSION :

Results:

- The Classroom and Assignment Management Module worked really well overall. It was fast and responsive, with actions like creating assignments and submitting work taking only 0.5 to 2 seconds. Even as more students and assignments were added, the system kept up without any noticeable slowdowns.
- Teachers and students both found the system easy to use. Teachers were able to manage assignments and grades quickly, while students could easily track their work and deadlines. The system was also accessible across different devices, whether on a computer, tablet, or phone.
- Data security was a top priority, and everything was stored securely using MongoDB. Only authorized users could access sensitive information.
- The system also proved scalable – no matter how many students or assignments were added, it continued to perform well. However, some users, especially those less familiar with technology, had a bit of a learning curve. There were also occasional slowdowns during busy times, like when everyone submitted assignments at once.
- In the end, the system met its main goals: it made life easier for teachers and more organized for students. There are a few areas to improve, like better user training and server optimization, but overall it's a solid tool with a lot of potential.

Discussion:

The Classroom and Assignment Management Module has made a real difference for both teachers and students by simplifying the way assignments and grades are managed. Teachers have saved time with automated processes for creating assignments, grading, and providing feedback. This has allowed them to focus more on teaching. Students, on the other hand, have enjoyed the convenience of having all their assignments, deadlines, and grades in one place, helping them stay organized and on track.

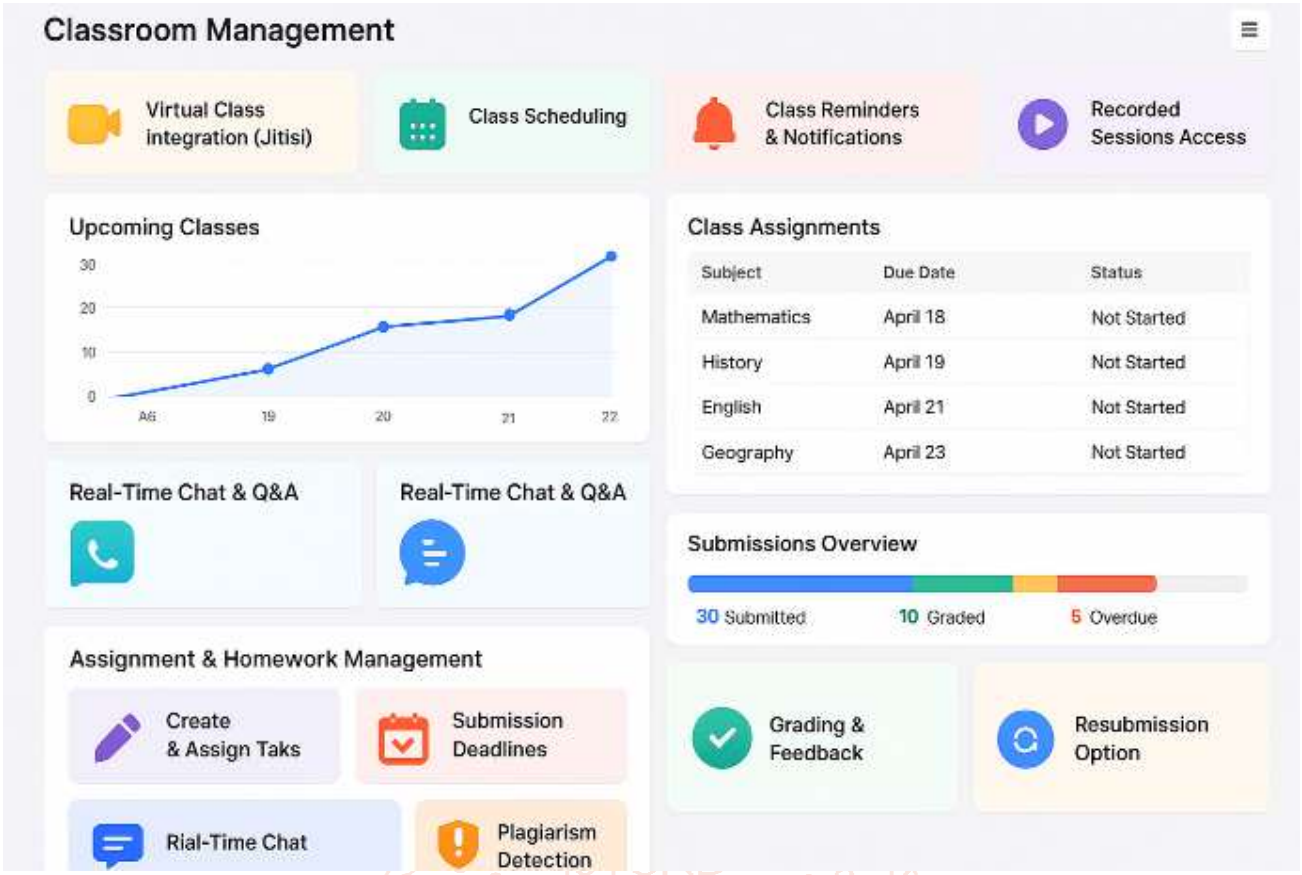
While most users found the system easy to use, there was a bit of a learning curve for some, particularly those who aren't as comfortable with technology. Offering more onboarding resources, like video tutorials or help guides, could help users get the hang of things more quickly.

The system performed well, even when more users and assignments were added. However, during busy times (like when everyone submitted assignments at once), there were some slowdowns. This could be improved by optimizing the server or considering cloud-based solutions for better performance during peak hours.

Security-wise, the system did a great job protecting user data with secure logins and proper data storage. However, adding extra layers of protection, like encryption for grades and personal info, would make it even safer.

Looking ahead, adding a mobile app could make the system even more accessible, especially for students and teachers who prefer using their phones or tablets. Adding features for direct communication between teachers and students (like messaging or forums) would also be helpful. Integrating with other tools, like Google Drive or plagiarism checkers, could further improve the experience and reduce the need for switching between platforms.

In the end, the module has successfully made classroom and assignment management easier and more organized. With a few enhancements, it has the potential to become an even more valuable tool for both teachers and students.



Dashboard

VI. CONCLUSION :

The Classroom and Assignment Management Module has made a significant difference in how both teachers and students manage assignments and track progress. For teachers, it has taken the load off many time-consuming tasks, like creating assignments, grading, and giving feedback, which has allowed them to focus more on teaching.

Students have also benefited from the module, as it offers them a clear and organized way to keep track of assignments, deadlines, and grades, making it easier for them to stay on top of their studies.

The system has been reliable, with fast response times and solid security, ensuring that only authorized users can access their data. Thanks to the use of React.js and MongoDB, it has performed well, even as the number of users and assignments grew.

That said, there were some challenges—particularly with users adjusting to the new system and occasional downtime during peak times. These issues can be tackled by offering better user training and improving server performance.

Overall, the module has been a valuable tool in making classroom and assignment management easier and more efficient. There's still room to grow, though. Future improvements—like a mobile app, better communication features, and integration with other tools—could make the system even more useful and enhance the experience for both teachers and students.

VII. REFERENCES :

- **Books:**

[1] *"Web Development with Node and Express"* – Server-side handling with MongoDB.

[2] *"Learning React"* – Building interactive UIs.

[3] *"MongoDB: The Definitive Guide"* – MongoDB for backend data management.

- **Online Resources:**

[4] MDN Web Docs for frontend tech (HTML, CSS, JS).

[5] React Docs for building UIs.

[6] MongoDB University for MongoDB courses.

- **Research Papers:**

[7] "Design and Implementation of School Management System" – Covers classroom and assignment management.

[8] "A Cloud-Based School Management System" – Cloud-based architecture.

- **Examples:**

[9] Google Classroom, Trello, and Schoology for features and design patterns.

- **GitHub Repositories:**

[10] Search for open-source school management systems built with React, Node, and MongoDB.

- **Tech Blogs:**

[11] Dev.to and Medium for tutorials and guides.