JETIR.ORG
ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

ACADEMIC PROGRESS TRACKER- ACADSYNC

[1]Chirag S, , [1]Sunay Y Anchan,[1] Tanishq, [2]Ganapathi Sharma

[12]AI & DS department,
[12]Srinivas Institute of Technology, Mangalore, India

Abstract: The Academic Progress Tracker is a modern web application with the goal to make academic management easier and more transparent to all parties concerned—students, teachers, and parents. Developed using the MERN stack (MongoDB, Express.js, React.js, and Node.js), the system aggregates all the important student-related data at one handy place. Students can login to see attendance, CGPA, upcoming tests, and assignments, keeping them on their toes academically. Teachers can input student data, post announcements, and monitor academic progress without having to deal with papers. Parents get a personalized dashboard where they can monitor the child's performance, fees, and important updates, keeping them more engaged in the child's study. With secure login and a responsive, clean UI, this replaces the outdated manual systems and offers a smart, digital alternative that saves time and promotes better communication through and through.

I INTRODUCTION

Management of student data in schools and universities has always been time-consuming and laborious. From monitoring attendance and marking to reporting progress to parents, traditional methods often translate into a lot of handwork, paperwork, and poor communication.

To tackle the above-mentioned issues, we developed the Academic Progress Tracker—a web application that brings together the students, the teachers, and the parents into a single easy to navigate interface. The system uses the MERN stack (MongoDB, Express.js, React.js, Node.js) such that a responsive, quick, and scalable web application can be achieved.

The overall objective of this project is to make work pertaining to students automated and create an uninterrupted flow of communication between all the stakeholders. With various dashboards designed specifically for each category of user—student, teacher, and parent—the system makes sure that everyone can view what they need to view, when they need to do.

By automating manual processes with a digital solution, this makes student administration more efficient, organized, and accessible. It minimizes the likelihood of making errors and saves faculty and administrative staff precious time. Students get a clear picture of their performance, and parents are informed and involved in the child's education. Teachers are able to teach more and less paperwork. The system also enhances transparency and accountability throughout the institution. Overall, it provides a smarter, more collaborative learning experience. It is extremely flexible to educational institutions of any size, ranging from schools to colleges. The intuitive nature of the interface ensures that even non-technical users can use the system with ease. With room for future development such as chat support, automated reminders, and analytics, the system is built to evolve with institutional needs. Ultimately, the Academic Progress Tracker closes the communication gap and enables a more data-driven educational experience.

II.LITERATUREREVIEW

Over the last few years, the demand for efficient and scalable Academic Progress Trackers (SMS) has grown enormously because of the expansion of educational institutions and the requirement of automating academic processes. Manually managing academic information leads to delay, faults, and inaccessibility. Therefore, a number of researchers and developers have suggested digital systems to overcome these limitations with a host of technologies such as web-based frameworks, cloud environments, the Internet of Things (IoT), and artificial intelligence.

Tang [1] created a Academic Progress Tracker employing the Springboot framework with an example of a modular back-end development structure. The application had a strong focus on quick development and ease of maintenance through componentbased development. While stable, the solution was mostly backend-centric, with minimal attention towards the user interface. This signifies the requirement of complete-stack solutions like those that have been built employing MERN (MongoDB, Express.js, React.js, Node.js) to integrate seamlessly frontend with the backend.

Zhang et al. [2] proposed a visualization-assisted SMS with distributed storage infrastructure. Their paper focused on the real-time visualization of students' performance data and load balancing of the database to make it appropriate for large institutions with a high number of data. This is in line with the scalability of MongoDB in the MERN stack that processes unstructured or semistructured educational data efficiently.

Zeng and Boontasorn [3] discussed the application of IoT to student management through features like RFID-based smart attendance and mobile sensor technology. Their system introduced automation in campus life with less human intervention. This system, as promising as it showed with its advanced integration of technology, created concerns regarding cost, privacy, and infrastructure sophistication—issues usually encountered by developing nations' institutions. Conversely, MERN-based systems provide a more cost-effective and modular solution while allowing for future growth to include IoT integration.

Xiong [4] proposed a browser/server (B/S) architecture for a web-based SMS, highlighting the importance of accessibility across different devices. Their work focused on responsiveness and basic CRUD operations (Create, Read, Update, Delete) for managing student data. This is similar to the design philosophy of modern MERN stack applications, where React is used for dynamic and responsive interfaces that can adapt to various screen sizes and device types.

Yu [5] approached from a more holistic perspective by combining teaching resource sharing in a university SMS. Such a combination enabled students and teachers to share academic content and information in a common digital space. The idea finds parallels with centralizing various academic tools into a single dashboard, as seen in MERN-based systems. The component-based architecture of React enables such modular extensions without drastic architectural changes.

Jin et al. [6] and Tang and Zhang [7] ventured into early web-based SMS utilizing classic B/S models. These systems set the stage for current SMS platforms since they were based on supporting student profiles, user roles, and managing academic records. Nonetheless, they did not support real-time updates, an aspect currently offered by latest technologies such as Node.js (utilizing WebSocket or REST APIs). These early computers also employed relational databases, although modern applications are now more using NoSQL databases such as MongoDB due to their flexibility.

Turcu et al. [8] introduced RFID and other ICT tools in educational institutions to provide security and automation. The research recommended using RFID tags to ensure identification in libraries, classrooms, and hostels. Although effective, these systems tend to need expensive hardware and regular network infrastructure. However, MERN stack systems can focus on features requiring only a web browser and internet to avoid the capital burden of implementation.

Shatnawi et al. [9] suggested a data mining-driven intelligent academic counseling system through association rule mining. Their system had the ability to suggest courses based on past performance patterns, providing a customized learning experience. The system can be incorporated into MERN-based applications with Node.js for backend and MongoDB for pattern data storage. Additionally, machine learning frameworks like TensorFlow.js can be incorporated into the stack to provide improved prediction and recommendation functionality.

Ray and Sharma [10] further developed this concept by using collaborative filtering to recommend elective classes. Their system, while in research format at the time of writing, showed how recommender systems can be used to aid academic planning. With React's versatility and Express.js's API-friendly nature, these sophisticated features can be easily integrated into contemporary SMS platforms, giving students smart assistance with academic choices.

Collectively, these researches emphasize the technological development and expansion of Academic Progress Trackers. From basic web portals to smart, IoT-based, and data-driven systems, the landscape has evolved significantly. However, most of the current systems are either too basic or excessively complicated and expensive for small and medium-sized institutions. This is where the MERN stack stands out—it offers a balanced solution with contemporary features, affordability, and space to grow.

By building upon the best practices and lessons of earlier research, this project hopes to create an extensible, intuitive, and scalable Academic Progress Tracker. It borrows the chief strengths of prior systems—role-based access, automation, real-time data, and personalization—while removing their limitations via a contemporary full-stack approach.

Key Findings and Trends

- Modern APT platforms increasingly adopt modular frameworks like Springboot and MERN for scalability and ease of development.
- Distributed storage systems enhance performance and allow efficient handling of large student datasets.
- Integration of IoT technologies enables real-time tracking and automation in campus management.
- Browser/Server (B/S) architectures support cross-platform access and simplify deployment for institutions.
- Resource-sharing features improve collaboration and accessibility to academic materials among students and faculty.
- Traditional systems lack real-time data processing, a gap now addressed by technologies like Node.js and WebSockets.
- RFID and ICT integration improves security and automation but comes with infrastructure challenges.
- Data mining and recommender systems enable personalized academic advising and course recommendations.

- MERN-based APT solutions provide a balanced mix of performance, cost-efficiency, and modularity, making them ideal for modern institutions.

III. Methodology

The proposed Academic Progress Tracker is developed using the MERN stack (MongoDB, Express.js, React.js, and Node.js), which offers a full stack JavaScript-based framework to provide modularity, scalability, and responsiveness across any device. The system is developed in phases, drawing inspiration from successful models and approaches outlined in recent literature.

1.System Design and Requirement Analysis

In the initial step, system functional and non-functional requirements were collected, including student enrolment, attendance record, result management, and role-based access by faculty, administrators, and students. The requirement specification directly follows Tang's [1] modular design approach in terms of separation of concerns and scalability, as applied to educational systems. An B/S architecture is employed, like in the models presented by Xiong [4] and Jin et al. [6], in a way that allows platform-independent access using any present browser. The architecture is easier to maintain and remote-friendly without local installations.

2.Technology Stack Selection

MERN stack was used due to its support for JSON data structures, asynchronous messaging, and open-source community. MongoDB was employed as the database layer to maintain semi-structured academic information such as course enrollments, marks, and feedback, in line with distributed storage conventions highlighted by Zhang et al. [2].React.js is used for frontend interface creation, offering a component-based and responsive system. This offers ease of reusability and dynamic UI updates, following principles of contemporary user interface design for academic portals [5].Node.js and Express.js form the middleware and backend stacks, handling API routes, authentication, and business logic. They deal with real-time data processing on a scalable scale, removing the performance bottlenecks associated with typical relational database-based systems [6], [7].

3. API Development and Database Schema

A NoSQL schema was utilized to model students, instructors, courses, attendance, and feedback records. The nonrelational model aligns with the scalability benefits outlined in Zhang et al. [2]. RESTful APIs were utilized to handle CRUD operations and frontend-backend communication.Role-based access control (RBAC) was utilized to provide data privacy and controlled access, a characteristic highlighted in secure academic systems by Tang and Zhang [7].

4. Inclusion of Additional Features

Based on intelligent advising systems [9] and models for choice suggestion [10], the design is that in the future, machine learning API-backed collaborative filtering-based or association rule mining-based recommendation engines can be added. .In order to remain cost-efficient and straightforward, the project does not provide IoT integrations like RFID (as proposed in [3] and [8]) but reserves hooks for future implementation in case institutions opt for physical device integration.

5. Deployment and Testing

The system was tested at unit level, integration level, and usability level to ensure correctness and smooth user experience. Emphasis was on fast page loads, form validation, and mobile responsiveness as per best practices discussed by Yu [5]. Hosting was done on a cloud infrastructure using Node.js hosting services with MongoDB Atlas as the cloud database solution due to its scalability as well as reliability.

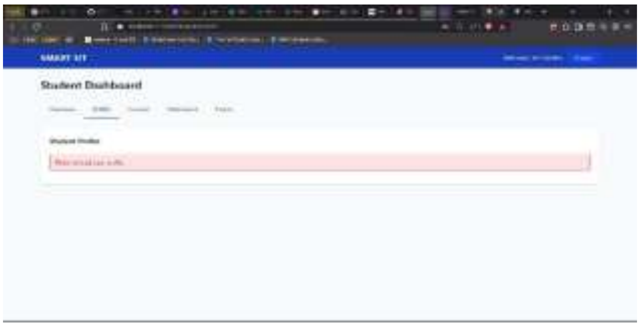

Figure 3.1:Student Dashboard

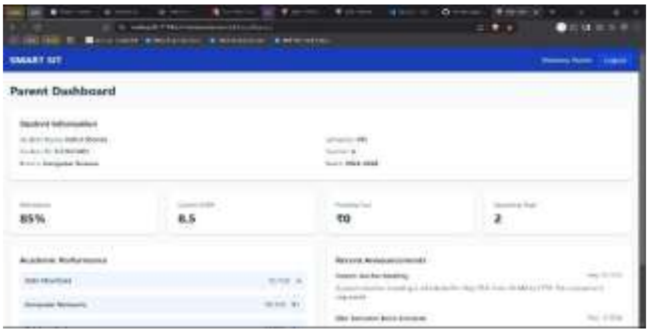

Figure 3.2:Parent Dashboard

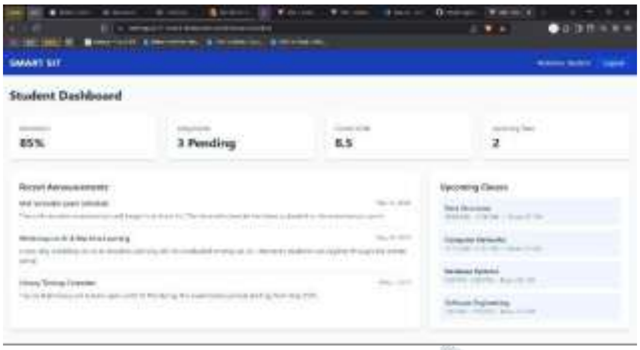

Figure 3.3: Recent Announcements

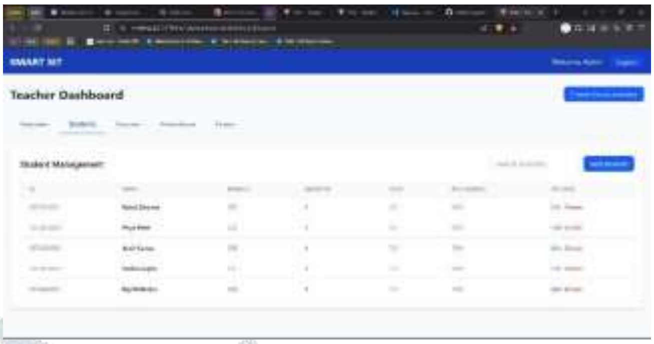

Figure 3.4: Teacher Dashboard

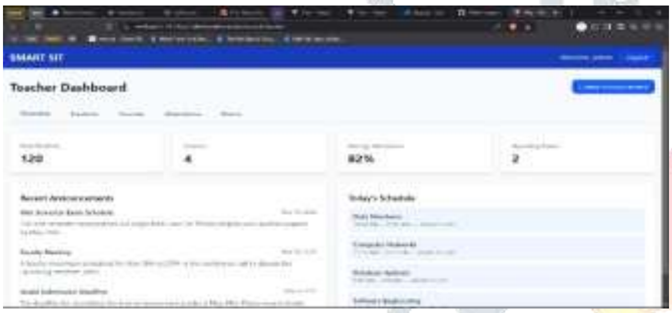

Figure 3.5:Teacher Dashboard Announcements

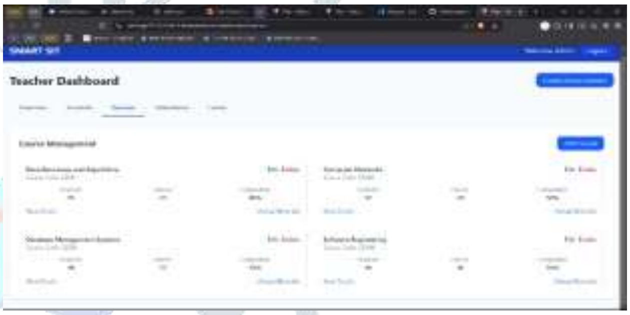

Figure 3.6:Course Management

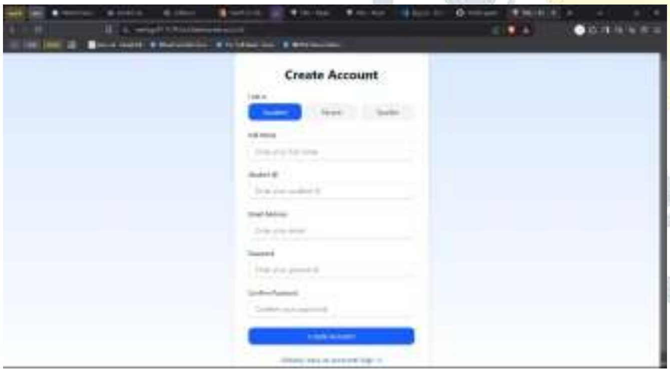

Figure 3.7:Sign In page

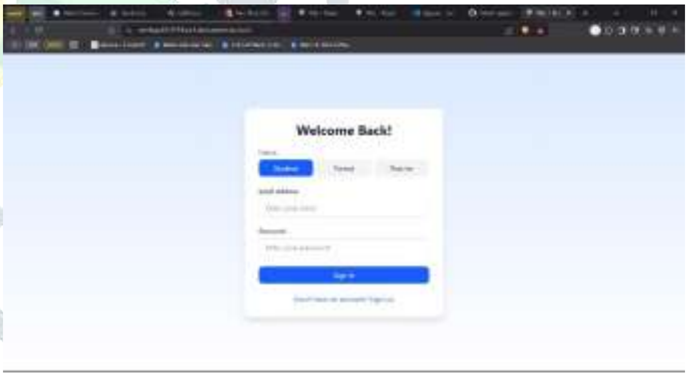

Figure 3.8:Login Page

IV. EXPECTED OUTPUT

The MERN-based Academic Progress Tracker will most probably offer a centralized, web-based platform for the efficient and secure management of student-related data. Through role-based access control, the system will provide access or updates to authorized users only for a given set of datasets, thus maintaining data integrity in overall terms. One of the outcomes is the availability of a responsive and user-friendly interface through React.js, which will facilitate easy navigation across devices. Through the integration of Node.js and RESTful APIs, real-time interaction will be supported, where the users will receive instant feedback and updates, thus enhancing the responsiveness and usability of the system.

The MERN stack's scalable and modular architecture will also facilitate future updates, such as adding more sophisticated analytics or IoT functionality. The ease of adaptation will allow the system to evolve and expand with institutional growth and changing needs. Repetitive educational and administrative tasks like attendance tracking, result processing, and notification will be automated, reducing manpower and the potential for errors. Cloud hosting will also offer users simple, remote access without local installations. These features together offer a cost-effective, accessible, and modern solution for educational institutions to automate and digitize student management processes.

Additionally, the system should allow for increased student and teacher interaction through enabling quicker exchange of academic data and updates. The NoSQL database architecture behind the system will allow for flexible data models, thus ensuring that the system is adaptable to accommodate various academic structures. Since the system is open-source, it encourages continuous improvement and contribution by the community. Through its real-time features and simplified processes, the system will improve the transparency and accountability of institutions. Finally, the project will establish a platform for a wiser, data-driven academic community.

REFERENCES

[1] Y. Tang, "Design and Implementation of a Academic Progress Tracker Based on Springboot Framework Technology," in *Proceedings of the First International Conference on Science, Engineering and Technology Practices for Sustainable Development (ICSETPSD)*, Coimbatore, India, Nov. 2023, doi: 10.4108/eai.17-11-2023.2342765.

[2] F. Zhang, X. Chen, and N. Tong, "Research on Visualization of Academic Progress Tracker Based on Distributed Storage Technology," in *Innovative Computing Vol 1 - Emerging Topics in Artificial Intelligence*, Lecture Notes in Electrical Engineering, vol. 1044, Springer, Singapore, 2023, pp. 1003–1010, doi: 10.1007/978-981-99-20921_100.

[3] Y. Zeng and N. Boontasorn, "Academic Progress Tracker Based on Intelligent Technology of Internet of Things," in *International Conference on Cognitive based Information Processing and Applications (CIPA 2021)*, Lecture Notes on Data Engineering and Communications Technologies, vol. 84, Springer, Singapore, 2022, pp. 1105–1112, doi: 10.1007/978-981-16-5857-0_110

[4] Z. Xiong, "Design and Implementation of Academic Progress Tracker Based on B/S," *International Journal of New Developments in Education*, vol. 5, no. 13, pp. 30–35, 2023, doi: 10.25236/IJNDE.2023.051306.

[5] Y. Yu, "Design of University Student Information Management System Based on Teaching Resource Sharing," in *Proc. SPIE 12922, Third International Conference on Electronics, Electrical and Information Engineering (ICEEIE 2023)*, Xiamen, China, Oct. 2023, doi: 10.1117/12.3008932.

[6] M. Jin, C. Qiu, and J. Li, "The Designment of Student Information Management System Based on B/S Architecture," in *2012 2nd International Conference on Consumer Electronics, Communications and Networks (CECNet)*, Yichang, China, Apr. 2012, pp. 2153–2155, doi: 10.1109/CECNet.2012.6201529.

[7] Y.-F. Tang and Y.-S. Zhang, "Design and Implementation of College Student Information Management System Based on Web Services," in *2009 International Symposium on IT in Medicine & Education*, Jinan, China, Aug. 2009, vol. 1, pp. 481–484, doi: 10.1109/ITIME.2009.5236230.

[8] C. Turcu, C. Turcu, V. Popa, and V. Gaitan, "ICT and RFID in Education: Some Practical Aspects in Campus Life," *arXiv preprint arXiv:1503.04286*, Mar. 2015. [Online]. Available: https://arxiv.org/abs/1503.04286arXiv

[9] R. Shatnawi, Q. Althebyan, B. Ghalib, and M. Al-Maolegi, "Building A Smart Academic Advising System Using Association Rule Mining," *arXiv preprint arXiv:1407.1807*, Jul. 2014. [Online]. Available: https://arxiv.org/abs/1407.1807arXiv

[10] S. Ray and A. Sharma, "A Collaborative Filtering Based Approach for Recommending Elective Courses," *arXiv preprint arXiv:1309.6908*, Sep. 2013. [Online]. Available: https://arxiv.org/abs/1309.6908arXiv

.