International Journal of Trend in Scientific Research and Development (IJTSRD)
Special Issue on Recent Advances in Computer Applications and Information Technology
Available Online: www.ijtsrd.com e-ISSN: 2456 – 6470

Building an ERP-Based Integrated Student Management System Website Using React, Node.js, HTML, CSS, and AWS Cloud

Vedant Gadkari, Chaitanya Karole

G H Raisoni University, Amravati, Maharashtra, India

Abstract

Enterprise Resource Planning (ERP) systems have become essential tools for managing organizational resources and processes in both business and educational environments. The development and deployment of student management systems have become more practical and scalable for institutions through the introduction of cloud technologies which incorporate frameworks like React and Node.js. This research paper explores the process of building a website for an ERP-based integrated student management system using these technologies and deploying it on the Amazon Web Services (AWS) cloud. The paper examines how ERP systems transform through modernization and cloud implementation which affects their architectural design and operational execution. The study analyzes how user experience data security and operational efficiency are impacted by current standards in ERP and cloud computing and web application development. The system connects different institutional modules through its main function which handles student enrollment and attendance tracking and examination management and fee processing and role-based dashboard access. The three-tier architectural model separates presentation application and data layers which leads to better system maintenance and scalability [1]

The system uses security measures which include role-based access control and encrypted communication and secure authentication protocols to safeguard sensitive academic information. With the rapid expansion of digital education ecosystems, scalable and secure management systems have become essential for institutional governance. This research presents a cloud-native ERP-Based Student Management System designed using modern full-stack technologies. The frontend is developed using React, while backend services are implemented using Node.js and RESTful APIs. The system is deployed on Amazon Web Services to leverage infrastructure scalability, high availability, and automated backups. A three-tier architecture ensures modular design and maintainability. The system incorporates role-based access control and encrypted communication mechanisms to enhance data security. Experimental results indicate that the cloud-based ERP model provides improved scalability, reduced infrastructure cost, and better performance under concurrent user loads. The study demonstrates that cloud computing significantly enhances the effectiveness of ERP systems in educational institutions [2].

KEYWORDS*: Enterprise Resource Planning (ERP), Student Management System, Integrated Academic System, Educational Information System, Full-Stack Web Development, React.js, Node.js, RESTful APIs, Cloud Computing, Amazon Web Services (AWS), MongoDB.*

1. Introduction

Modern educational institutions need to manage their academic and administrative functions through digital systems that require effective management. The unified student management system which operates according to Enterprise Resource Planning (ERP) standards solves these problems by delivering a central platform that enables departments to manage their data and operational workflows. The combination of React and Node.js technologies with HTML and CSS core web languages enables systems to achieve required performance standards for their complex requirements. The system achieves better reliability and security through its cloud deployment on AWS platforms which also provides scalability benefits. The research paper documents the development process for a website that implements an ERP-based student management system through React and Node.js for application development and HTML and CSS for content presentation while using AWS cloud infrastructure for system deployment [3].

The implementation of these technologies brings contemporary updates to the ERP system while following established digital transformation procedures that current research identifies as effective methods for ERP system modernization through cloud-based solutions (Rawat, 2023; Zhang, 2022). The paper investigates the factors that drive system development for cloud deployment while identifying the challenges which arise during system development and deployment. The paper investigates three core aspects which include system integration, security measures and user experience design[4].

Fig-1 diagram for cover page for the website

2. Literature Review

Enterprise Resource Planning (ERP) systems have traditionally been used by large organizations to consolidate their primary business operations into a single system which boosts productivity while enabling better access to data (Al-Mashari & Zairi, 2000). The implementation of ERP solutions has grown in educational institutions because they help solve problems which stem from data being stored in different locations and students' information needing to be recorded through physical methods and institutions not having access to real-time data (Assemi & Jazzi, 2012). The student management system which operates on an ERP platform brings together all essential functions such as admissions, attendance tracking, grading processes, timetable management, and administrative duties to create a unified system which allows different departments to share information while making operational decisions (Yadav & Rathore, 2017). The core functions of modern ERP systems now depend on web technologies. The advantages of building interactive user interfaces through component-based front-end frameworks which include React become clear through research on web application frameworks (Krochmal, 2018). React provides developers with tools to create UI components which can be reused together with state management functions and virtual DOM optimization, which enables developers to build applications that respond quickly while being easy to maintain. Node.js has emerged as a popular backend technology because it uses an event-driven system with non-blocking I/O capabilities that enable it to process many concurrent users, which makes it perfect for developing web applications which need to accommodate multiple users (Tinkov & Vinoski, 2010) [5].

The research which compares server-side technologies shows that Node.js provides an effective method to manage real-time data transfer, which modern ERP systems need to communicate with their API-based systems (Bhardwaj & Navet, 2020). Web interfaces rely on HTML and CSS to create their front-end markup and styling, which allows users to access various devices through responsive design principles(Marcotte, 2010). The front-end and back-end components of the system connect through RESTful APIs which enable their integration. VWith the emergence of cloud computing, ERP deployment models have shifted from on-premises infrastructure to scalable cloud environments. Cloud platforms such as Amazon Web Services provide elasticity, high availability, and cost-effective resource management, making ERP systems more accessible to small and medium-sized institutions. Research indicates that cloud-based ERP systems reduce capital expenditure while improving system reliability and disaster recovery capabilities. Recent advancements in web technologies have further influenced the development of modern student management systems. Frontend frameworks like React enable responsive and interactive user interfaces, while backend environments such as Node.js support scalable API-driven architectures. Studies emphasize that integrating these technologies enhances system performance, modularity, and maintainability. However, literature also identifies ongoing challenges related to data security, user adoption, and performance optimization, indicating the need for secure, scalable, and cost-efficient ERP models tailored specifically for educational institutions [6].

3. Research Methodology

Frontend Development: React was used to build the essential user interface components of the application. The application shared state through React hooks and Redux which developers used for state management. The design created a user interface that worked well on both desktop and mobile platforms.

Backend Development: The developers created RESTful APIs through Node.js and Express to handle system functions including user login, student registration, attendance submission, and grade entry. The system used JSON as the standard data format for exchanging information between the frontend and backend systems [7]

Database Integration: AWS RDS provided a cloud-based MySQL database that was set up for use. The database schema included tables which stored information about users, students, classes, attendance records, grades, and logs. The team improved SQL query performance through optimization techniques.

Cloud Deployment: The application used AWS Elastic Beanstalk for its deployment because it provided necessary scalability. The team developed automated testing and deployment processes through continuous integration and deployment pipelines which they created using AWS Code Pipeline and Code Build [8].

The system underwent evaluation through a usability study that included a sample group of stakeholders. The study measured four essential metrics which included time taken to complete tasks, number of mistakes made by users, user satisfaction measured through Likert scale surveys, and system performance during simulated load testing. The team used participant feedback to make improvements to the user interface design and optimize system performance. The development of the ERP-Based Integrated Student Management System followed a structured and iterative approach which used the Agile software development methodology as its foundation. The analysis of institutional administrative processes served as the basis for system requirements gathering which identified essential functional modules that included student registration and attendance tracking and examination management and fee processing. The requirements established for the project led to the creation of a three-tier system architecture which used presentation and application and data layers. The frontend interface was built with React to create an interface that users can interact with easily and which adapts to different device types. The backend of the system used Node.js to create RESTful APIs which handled business logic and allowed communication with databases [9].

The relational database system was developed to store academic and administrative records in an organized manner while using normalization methods for data management. The system used Amazon Web Services cloud infrastructure for deployment because it provided the necessary components to support system growth and system dependability and system protection of customer information. The testing procedures included three testing types which were unit testing and integration testing and performance evaluation testing to assess how the system worked and how it protected data and how it managed multiple users at the same time. The ERP solution development process created a modular system which educational institutions could use because it provided secure features and scalable capabilities. The proposed ERP-Based Integrated Student Management System development process used a structured software engineering method which started with requirement analysis and system planning. The researchers examined institutional workflows to determine essential components which included student registration and attendance tracking and examination processing and fee management. The system used three-tier architecture to divide its components into presentation layer and application logic and database sections. The development team used React to build the frontend because it enables users to interact with the application in a responsive and dynamic manner while Node.js handled backend operations through business logic implementation and API dispatching [10].

The system used Amazon Web Services cloud infrastructure for deployment because it provided advantages in scalability and reliability and secure data storage. The testing process included functional tests and performance tests to confirm that the system operated efficiently and maintained stable performance. The development team built the proposed system through an iterative Agile development process which allowed them to create flexible systems that could undergo ongoing improvements. The development team created smaller development cycles from which they built separate modules that underwent independent design work, implementation processes, and testing activities before the team conducted systemwide testing. The development team built the frontend interface through a component-based architectural system, which improved system maintainability, while the backend system used RESTful APIs to support efficient communication between its different system components. The organization set up cloud deployment to enable automatic system scaling and to protect data through secure management practices. The testing process used three testing methods which included unit testing and integration testing and load testing to verify both the system's functional performance and its ability to handle actual user scenarios [11].

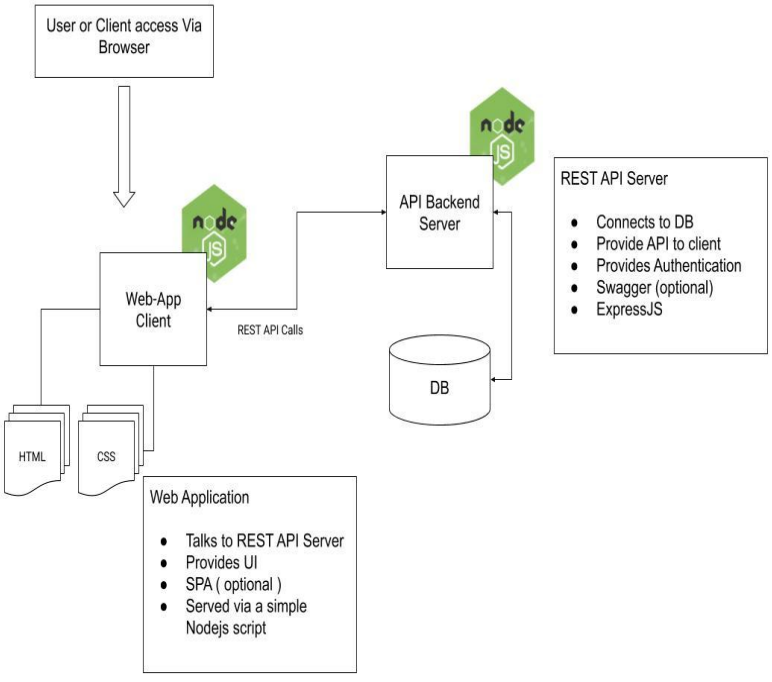

Fig-2 Block diagram of ERP management system

+Security Implementation: The implementation of authentication and authorization security measures used JSON Web Tokens (JWT) as the authentication method. The combination of AWS security groups and IAM roles created access restrictions that protected cloud resources from unauthorized access.

4. Result

The literature identifies these factors as essential components which must be present to achieve successful implementation of an ERP-based integrated student management system operated through cloud technology. The new system must achieve technical compatibility by establishing links with its existing components and all historic data systems (Rawat 2023). The institution needs to implement system best practices for process alignment to avoid expensive custom development work. The organization should view modernization as a perpetual operation that requires continuous system updates and user feedback evaluation. The presence of a strong executive leadership team combined with an established change management framework enables organizations to decrease resistance while achieving successful implementation. The system usage becomes more effective through the establishment of precise performance targets and the delivery of complete training resources to users [12].

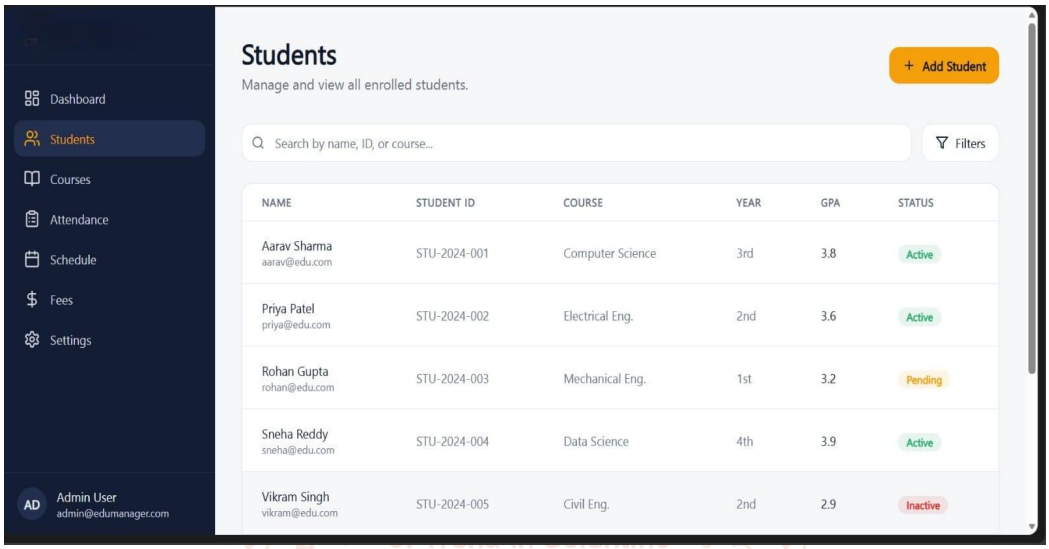

Fig-3 Above image represent the output of the website

5. Conclusion

The implementation process of an ERP-based student management system demonstrates The combination of web technologies with cloud-based systems enables institutions to develop operational systems which secure their operations and provide users with simple access and expandable functions. The literature shows that successful outcomes depend on organizations that study their integration challenges and security needs and their requirements for stakeholder involvement how digital transformation has transformed educational technology through its use of current React and Node.js and AWS technologies. The combination of web technologies with cloud-based systems enables institutions to develop operational systems which secure their operations and provide users with simple access and expandable functions.

The literature shows that successful outcomes depend on organizations that study their integration challenges and security needs and their requirements for stakeholder involvement. The AWS cloud services provide both enhanced system reliability The literature shows that successful outcomes depend on organizations that study their integration challenges and security needs and their requirements for stakeholder involvement. The AWS cloud services provide both enhanced system reliability and scalability benefits while delivering advanced security features and monitoring tools. Educational institutions use technology as a means to fulfill the current requirements of students and educators who operate in digital environments [13].

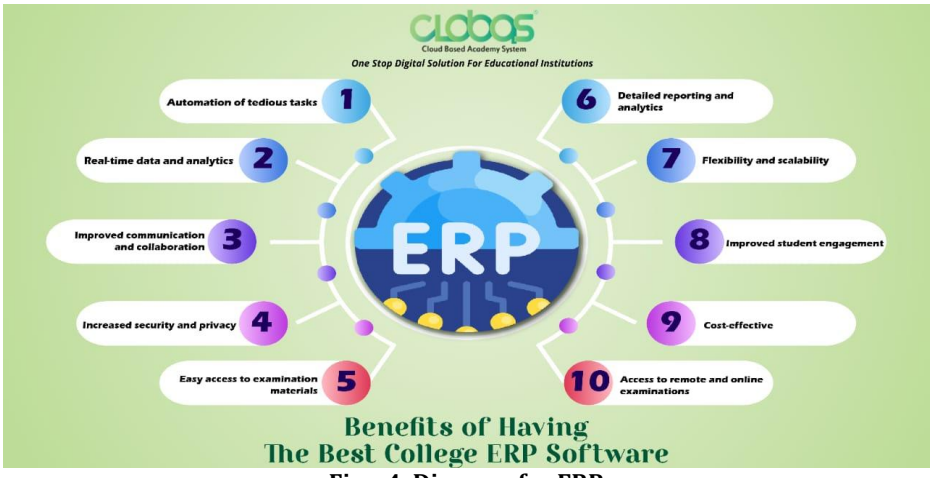

Fig – 4. Diagram for ERP

Reference

[1] C. Rawat, "Role of ERP modernization in digital transformation: PeopleSoft insight," International Journal of Computer Trends and Technology, vol. 71, no.2, pp. 61–67, 2023.

[2] M. Soltys, "Cybersecurity in the AWS cloud," arid preprint arXiv:2003.12905, 2020.

[3] X. Zhang, "Drivers affecting cloud ERP deployment decisions: An Australian study," Monash University, 2019.

[4] X. Zhang, "A hybrid cloud ERP framework for processing purchasing data," arid preprint arXiv:2202.10786, 2022

[5] Ananthakrishnan, C. M. & Hadadi, M. Cloud-based ERP systems for Higher Educational Institutes — review of ERP and cloud ERP implementation in HEIs and the advantages of cloud (2023)

[6] Badru, L. O., Vasudevan, V., Ishwar, G., Lingam, M. G. M. K. MERN Stack Web-Based Education Management Information Systems for Pacific Island Countries — discusses using MERN stack (MongoDB, Express, React, Node.js) to build dynamic education information systems (2022)

[7] Le Duc, M. Adoption of Enterprise Resource Planning (ERP) Systems in University Education Curricula — case study on adopting ERP in higher education contexts (2015)

[8] Patil, A. et al. Student Academic Management System Using Cloud Environment — a cloud-deployed student management system hosted on AWS to handle attendance and performance (2022)

[9] Abugabah, A., & Sansoni, L. Enterprise Resource Planning (ERP) system in higher education: A literature review and implications. International Journal of Human and Social Sciences, 5(6), 395–399 (2011)

[10] Beekhuyzen, J., Goodwin, S., & Nielsen, S. ERP implementation in universities: A case study. Proceedings of the Informing Science Conference, 2, 553–567 (2022)

[11] Pollock, N., & Cornford, J. ERP systems and the university as a "unique" organization. Information Technology & People, 17(1), 31–52 (2022)

[12] Allen, D., & Kern, T. Enterprise resource planning implementation: Stories of power, politics, and resistance. ACM SIGMIS Database, 32(3), 24–37(2001)

[13] Sife, A. S., Loge, E. T., & Sanga, C. New technologies for teaching and learning: Challenges for higher learning institutions in developing countries. International Journal of Education and Development using ICT, 3(2), 57–67 (2022)

[14] Rabaa'i, A. A. Identifying critical success factors of ERP systems at the higher education sector. Proceedings of the International Conference on Computer Science and Information Technology, 407–411(2009).

[15] Noudoostbeni, A., Yasin, N. M., & Jainabad, H. S. To investigate the success and failure factors of ERP implementation within Malaysian small and medium enterprises. International Journal of Business and Management, 4(7), 157–168 (2009).