An Integrated Student Faculty Analytics Platform for Continuous Academic Performance Assessment

P. Mounika[1], Yenke Laxmi Prasanna[2], Pasunoori Bhanuprakash[2], Pudari Manikanta[2], Sirikonda Akshay Kumar[2]

[1]Assistant Professor, [2]UG Student, [1,2]Department of Computer Science and Engineering (Data Science)

[1,2]Vaagdevi College of Engineering (UGC-Autonomous), Bollikunta, Warangal, 506005, Telangana

Abstract

Monitoring and predicting student performance is increasingly critical in modern data-driven educational environments. Conventional academic management systems often function as static repositories for attendance and grades, lacking predictive insights and integrated communication channels necessary for timely academic intervention. This study proposes a centralized e-learning and management portal integrated with machine learning (ML) analytics to proactively track and predict student outcomes. The framework employs a role-based architecture comprising Admin, Faculty, and Student dashboards to facilitate seamless data flow and engagement. To classify student performance into *Good, Fair,* and *Poor* categories, two predictive models, Naïve Bayes and XGBoost, were implemented and evaluated. Experimental results demonstrate that the XGBoost model significantly outperforms the baseline, achieving an accuracy exceeding 93%. Beyond performance metrics, the system introduces an algorithmic faculty suggestion module to enhance student-instructor alignment. By synthesizing predictive modeling with a unified management interface, this research provides a scalable solution for early academic intervention, improved institutional transparency, and personalized pedagogical support.

Keywords: Student performance, Machine learning, XGBoost, Educational data mining, Academic management systems, Predictive analytics.

1. Introduction

Student retention and academic success are major concerns in higher education institutions worldwide. Universities continuously seek effective mechanisms to identify students who are academically at risk and provide timely interventions. One such mechanism is an Early Alert System (EAS), a structured approach used to detect students who may be struggling academically, socially, or behaviorally before their performance declines significantly. Early alert systems seek to improve relatedness by identifying students who would benefit from intervention in a timely manner. This identification allows the institution to connect that student with resources or student success staff proactively (e.g., by prompting an academic advisor to reach out to the student). This institutional outreach is essential within the theoretical framework provided by Tinto [1], as faculty interactions forward a students' social integration into campus and thus decrease the risk of institutional departure. Early alert systems are prevalent among public and private universities in the United States. A 2014 report estimated that 85–90% of higher education institutions in the United States had an early alert system in use [2]. Early alert systems have been found to be effective in global universities as well. For example, early alert systems have been found to enhance student success at universities in Europe, Africa, and Asia, and Oceana, either by accurate detection of at-risk students or connection of students to tutoring resources. Early alert systems are particularly relevant to the support ecosystem at colleges and universities because they can serve as an entry point for a student to access relevant resources. For example, an early alert can prompt a student to meet with an academic advisor, who then may suggest making an appointment with the tutoring center. Thus, although early alert systems themselves may not provide help in the form of homework support or financial aid adjustment, they can prompt awareness and action that lead the student to interface with university supports that do offer those services [3].

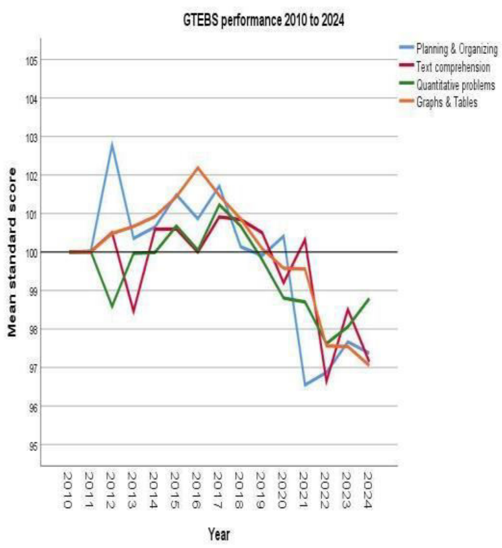

Fig. 1: Evaluating student knowledge assessment using ML techniques.

Though these systems interface with instructors and student success staff, the key user is the student, who decides how to respond to the alert. Thus, the current study was contextualized within frameworks of student persistence in higher education such as Tinto's model of institutional departure and self-determination theory. Both theories highlight that students who are connected to the people and resources of an institution are more likely to persist through university. We sought to understand the student experience of a particular early alert system, called Academic Status Reports (ASRs) [4]. Thus, we administered a survey to undergraduate students at a large public university to examine their knowledge of and responses to ASRs. Our primary aim was to gain an improved understanding of how college students think about and react to this type of institutional support. Early alert systems can be used to identify students who would benefit from an influx of support and notify student success staff (e.g., administrators, advisors) of these students [5]. These systems are sometimes referred to as early warning systems when they are used only to identify students experiencing academic or related challenges. However, these systems can also be used to provide positive feedback (e.g., kudos to a student for a strong start to the semester). For that reason, we use the neutral term early alert system.

2. Literature Survey

Alruwais, et al. [6] designed to create a machine learning (ML)-based system that assesses student performance and knowledge throughout the course of their studies and pinpoints the key variables that have the most significant effects on that performance and expertise. Additionally, it describes the impact of running models with data that only contains key features on their performance. Oliveira, et al. [7] introduced the implementation of a chatbot on the web portal of a higher education institution, aiming to enhance student interaction and provide seamless access to information and support services. With the increasing reliance on digital platforms for student engagement, a chatbot offers a user-friendly and efficient means of communication, catering to the diverse needs of students in a higher education setting. Imundo, et al. [8] investigated students' knowledge of and experiences with an early alert system designed to enhance academic persistence. Method: We surveyed (N = 356) undergraduates at a large public university in the U.S. The survey was researcher-created and administered online. Participants self-selected into the study and provided informed consent prior to beginning the study. Data were coded by experts, who achieved excellent IRR. The analyses were frequency-based to understand diverse student knowledge, experiences, and responses with early alert systems.

Guanin-Fajardo, et al. [9] Recognized the gaps in the study and that on-time completion of college consolidates college self-efficacy, creating intervention and support strategies to retain students is a priority for decision makers. Assessing the fairness and discrimination of the algorithms was the main limitation of this work. Syed Mustapha, et al. [10] determined the optimal methods for feature engineering and selection in the context of regression and classification tasks. This study compared the Boruta algorithm and Lasso regression for regression, and Recursive Feature Elimination (RFE) and Random Forest Importance (RFI) for classification. According to the findings, Gradient Boost for the regression part of this study had the least Mean Absolute Error (MAE) and Root-Mean-Square Error (RMSE) of 12.93 and 18.28, respectively, in the case of the Boruta selection method. Zou, et al. [11] focused on the students majoring in Computer Science in a certain university and conducted an exploration using their scores in multiple undergraduate courses. Initially, we selected the students' basic and core academic courses based on the training program and identified four groups of course combinations with strong positive correlations through correlation and cluster analysis. Luo, et al. [12] described the complexity and nonlinearity of learning behaviors in the educational process, predicting students' academic performance effectively is challenging. Nevertheless, machine learning algorithms possess significant advantages in handling data complexity and nonlinearity. Initially, a multidimensional spatiotemporal feature dataset was constructed by combining three categories of features: students' basic information, performance at various stages of the semester, and educational indicators from their places of origin (considering both temporal aspects, i.e., performance at various stages of the semester, and spatial aspects, i.e., educational indicators from their places of origin).

3. Proposed Methodology

The proposed system is a Django-based web application designed to automate student-faculty interactions and academic management. It allows students to view marks, attendance, assignments, and provide feedback, while faculty can upload materials, add marks, and manage attendance. The system integrates Machine Learning models like Naive Bayes and XGBoost to predict faculty performance and provide intelligent faculty suggestions. It also includes a messaging module with email notifications, ensuring effective communication between students, faculty, and parents. Admins can view performance analytics and generate reports for better decision-making. Overall, this system replaces manual processes with a data-driven, efficient, and user-friendly platform.

Data Collection: All academic data, including attendance, assignment submissions, test scores, faculty evaluations, and historical academic records, are collected from students and faculty. This ensures a centralized repository of student performance data for analysis.

Data Preprocessing: Raw data is cleaned, normalized, and missing values are handled. Feature selection and extraction are performed to create a structured dataset suitable for machine learning algorithms.

Centralized Database Management: The preprocessed data is stored in a centralized database, which maintains secure access and ensures consistency. This module also handles role-based access for students, faculty, and administrators.

Predictive Modeling: Machine learning models, such as Random Forest, Logistic Regression, or XGBoost, are trained on historical academic data to predict student performance in real-time. This module generates performance scores and identifies students who may require additional support.

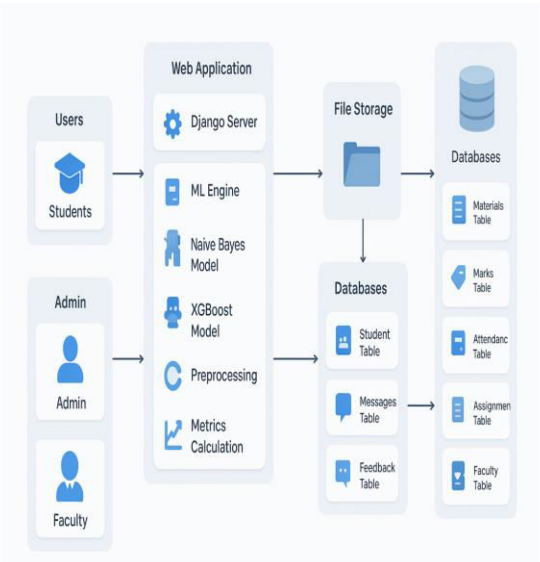

Fig. 2: System overview of the centralized student–faculty platform for real-time academic performance prediction.

Dashboard & Visualization: A dynamic dashboard presents insights from predictions, showing performance trends, at-risk students, and analytics to both faculty and students. Visualization tools enhance understanding and decision-making.

Feedback & Intervention: The system provides actionable feedback to students and faculty. Faculty can intervene early with tailored guidance, while students can track their progress and improve their academic performance proactively.

XGBoost Classifier

XGBoost (Extreme Gradient Boosting) is a powerful ensemble machine learning model based on decision trees. It builds multiple trees in sequence, where each tree learns from the errors of its predecessors. XGBoost is known for its speed, regularization techniques, and high performance on structured/tabular data.

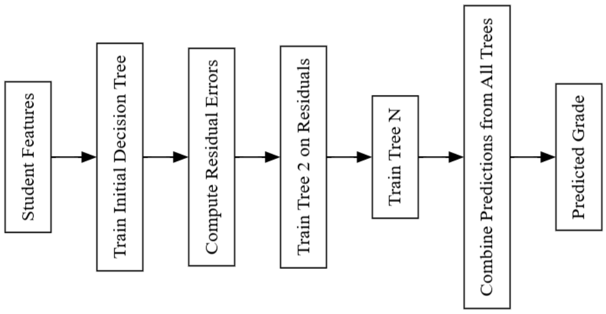

Fig. 3: Flow diagram of XGBoost classifier training process.

Initialize Base Model: XGBoost starts with a base prediction, usually a single decision tree, which provides initial guesses for student feedback. At this stage, the model might be inaccurate, but it establishes a starting point for improvement. This base tree helps the algorithm understand the basic relationship between student marks and feedback categories.

Compute Residuals: After the base tree makes predictions, XGBoost calculates the errors, called residuals, by comparing predicted feedback with actual feedback. These residuals highlight where the model is underperforming. In this project, residuals indicate which students' feedback was predicted incorrectly, guiding the next learning step.

Fit a New Tree on Residuals: A new decision tree is trained to predict these residuals, effectively learning how to correct the mistakes of the previous tree. Each new tree focuses on the hardest-to-predict instances, improving accuracy iteratively. This ensures that patterns missed in earlier trees, like subtle differences in student marks, are captured in subsequent iterations.

Update Predictions: Predictions are updated by adding the contributions of the new tree to the existing predictions. This additive process gradually refines feedback predictions, making them closer to the actual categories. Over multiple iterations, this step allows the model to capture complex patterns in student performance data.

Apply Learning Rate and Regularization: XGBoost adjusts the new tree's contribution using a learning rate to prevent overshooting the solution. Regularization terms penalize complex trees to avoid overfitting, ensuring the model generalizes well to unseen student data. These techniques help maintain accuracy while keeping the model robust and reliable.

Iterate Until Convergence: Steps 2–5 repeat for a predefined number of trees or until the model's improvement becomes minimal. Each iteration incrementally reduces prediction errors. In this project, multiple iterations allow the model to learn intricate relationships between marks across different subjects and the corresponding feedback.

Final Prediction: Once all trees are trained, the final prediction for a student's feedback is obtained by summing contributions from all trees. The numeric prediction is converted back into the feedback category (Excellent, Good, Average). This step produces a highly accurate, interpretable output for automated student feedback evaluation.

4. Result Description

The system produced accurate and meaningful results in managing academic data as well as predicting student performance using machine learning. Below screens are able to view the marks, attendance, assignments, study materials, and academic updates through their personalized dashboard, while faculty and administrators efficiently performed academic management tasks such as entering marks, uploading material, and reviewing student progress.

Fig. 4 demonstrates the functionality where administrator or authorized personnel upload teacher information into the system. The uploaded data includes teacher names, subjects assigned, and contact information. This process ensures the system maintains a structured record of teaching staff for academic management. Fig. 5 illustrates the process of adding student information into the system. The data includes student names, roll numbers, class, and relevant academic details. Uploading these details enables the system to track individual student performance and link them to their respective courses and teachers. Fig. 6 shows the teacher marking or uploading attendance for students. Attendance data is recorded and associated with specific dates and subjects. This feature ensures accurate tracking of student presence and supports performance analysis.

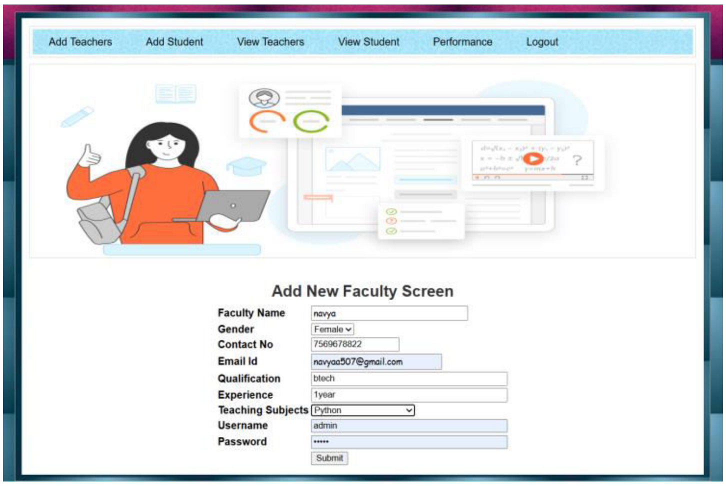

Fig. 4: Uploading teacher details

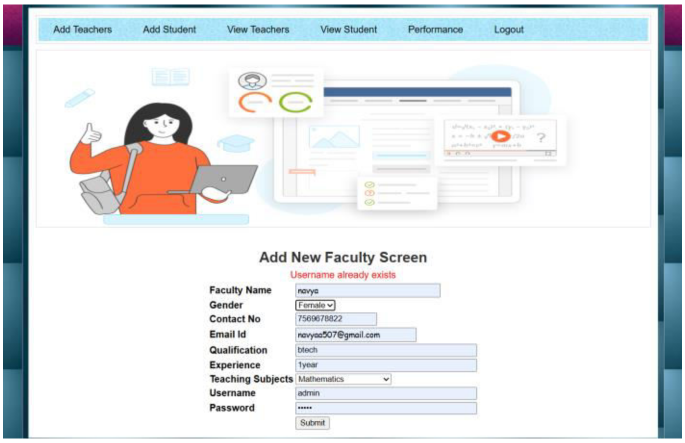

Fig. 5: Uploading student details.

Algorithms Name	Accuracy	Precision	Recall	F1-Score
Naive Bayes Classifier	86.92	94.03	81.79	80.55
XGBoost Classifier	93.07	93.66	93.56	92.84

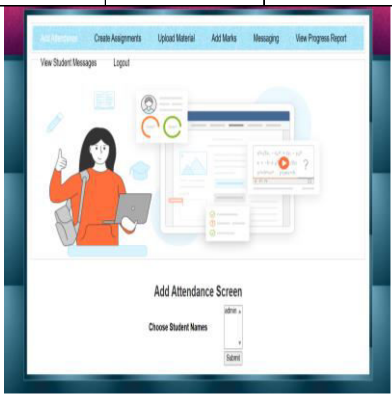

Fig. 6: Teacher uploading student attendance.

Fig. 7 presents the process where teachers create assignments for students. Teachers input assignment details, including title, description, and submission deadlines. The system stores these assignments, making them accessible for students to review and submit. Fig. 8 illustrates the student interface for viewing assignments assigned by teachers. Students can access assignment instructions, deadlines, and submission status. This feature facilitates timely completion and submission of academic tasks.

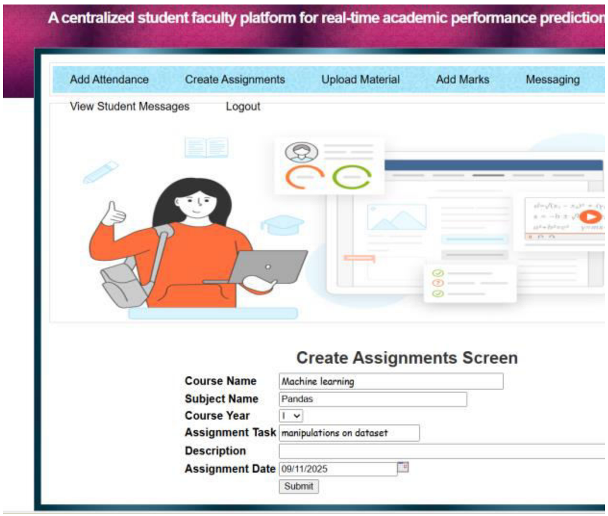

Fig. 7: Teacher creating assignments.

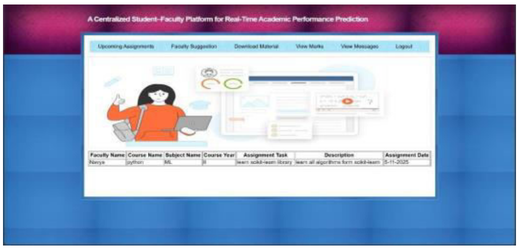

Fig. 8: Student checking assignments.

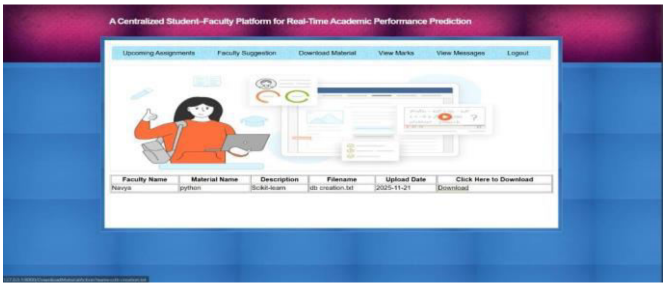

Fig. 9: Student downloading learning materials.

Fig. 9 displays the functionality where students download uploaded study materials. Materials such as notes, presentations, and reference documents are organized by subject. This feature ensures students have easy access to resources required for learning and assignment completion. Table 1 presents a performance comparison between the Naive Bayes Classifier and the XGBoost algorithm. The Naive Bayes Classifier achieved an accuracy of 86.92%, with precision, recall, and F1-score values of 94.03%, 81.79%, and 80.55%, respectively. In contrast, the XGBoost Classifier outperformed Naive Bayes across all metrics, achieving an accuracy of 93.07%, precision of 93.66%, recall of 93.56%, and an F1-score of 92.84%.

Table 1: Comparison for the Naive Bayes Classifier and XGBoost algorithms.

This comparison highlights the stronger predictive capability of XGBoost in handling the dataset used for this project. The results demonstrate that XGBoost provides more balanced performance between precision and recall, ensuring higher reliability in classification tasks. Overall, XGBoost offers a significant improvement in accuracy and effectiveness compared to the Naive Bayes approach.

5. Conclusion

The research successfully developed an automated educational management system integrating teacher and student functionalities. Teachers were able to upload student details, mark attendance, and create assignments efficiently, while students could access assignments and download study materials seamlessly. The implementation of machine learning models, specifically NBC and XGB, enhanced the system's predictive and analytical capabilities. XGB delivered superior performance with an accuracy of 93.07%, high precision, recall, and F1-score, proving the model's effectiveness in handling educational data. The system improved data management, reduced manual workload, and provided a structured, efficient interface for both teachers and students.

REFERENCES

[1] Kezar, A.; Kitchen, J.A.; Estes, H.; Hallett, R.; Perez, R. Tailoring programs to best support low-income, first-generation, and racially minoritized college student success. *J. Coll. Stud. Retent. Res. Theory Pract.* 2023, *25*, 126–152.

[2] Wilmer, E. Student support services for the underprepared student. *Inq. J. Va. Community Coll*, *13*, 4–19.

[3] Andrews, R.J.; Imberman, S.A.; Lovenheim, M.F. Recruiting and supporting low-income, high-achieving students at flagship universities. *Econ. Educ. Rev.* 2020, *74*, 101923.

[4] Hoyt, J.E. Student connections: The critical role of student affairs and academic support services in retention efforts. *J. Coll. Stud. Retent. Res. Theory Pract.* 2023, *25*, 480–491.

[5] Kuh, G.D. Student success. In *Student Services: A Handbook for the Profession*, 5th ed.; Schuh, J.H., Jones, S.R., Harper, S.R., Eds.; Jossey-Bass: Hoboken, NJ, USA, 2011; pp. 257–270.

[6] Alruwais, N.; Zakariah, M. Evaluating Student Knowledge Assessment Using Machine Learning Techniques. *Sustainability* 2023, *15*, 6229. https://doi.org/10.3390/su15076229.

[7] Oliveira, P.F.; Matos, P. Introducing a Chatbot to the Web Portal of a Higher Education Institution to Enhance Student Interaction. *Eng. Proc.* 2023, *56*, 128. https://doi.org/10.3390/ASEC2023-16621.

[8] Imundo, M.N.; Goldshtein, M.; Watanabe, M.; Gong, J.; Crosby, D.N.; Roscoe, R.D.; Arner, T.; McNamara, D.S. Awareness to Action: Student Knowledge of and Responses to an Early Alert System. *Appl. Sci.* 2025, *15*, 6316. https://doi.org/10.3390/app15116316.

[9] Guanin-Fajardo, J.H.; Guaña-Moya, J.; Casillas, J. Predicting Academic Success of College Students Using Machine Learning Techniques. *Data* 2024, *9*, 60. https://doi.org/10.3390/data9040060.

[10] Syed Mustapha, S.M.F.D. Predictive Analysis of Students' Learning Performance Using Data Mining Techniques: A Comparative Study of Feature Selection Methods. *Appl. Syst. Innov.* 2023, *6*, 86. https://doi.org/10.3390/asi6050086.

[11] Zou, W.; Zhong, W.; Du, J.; Yuan, L. Prediction of Student Academic Performance Utilizing a Multi-Model Fusion Approach in the Realm of Machine Learning. *Appl. Sci.* 2025, *15*, 3550. https://doi.org/10.3390/app15073550.

[12] Luo, Z.; Mai, J.; Feng, C.; Kong, D.; Liu, J.; Ding, Y.; Qi, B.; Zhu, Z. A Method for Prediction and Analysis of Student Performance That Combines Multi-Dimensional Features of Time and Space. *Mathematics* 2024, *12*, 3597. https://doi.org/10.3390/math12223597.

[13] Todupunuri, A. (2023). The Role of Artificial Intelligence in Enhancing Cybersecurity Measures in Online Banking Using AI. International Journal of Enhanced Research in Management & Computer Applications, 12(01), 103–108. https://doi.org/10.55948/ijermca.2023.0 1015Mahesh Ganji. (2025). Enhancing Oracle Cloud HR Reporting Through AI-Driven Automation. Journal of Science & Technology, 10(6), 28–36. https://doi.org/10.46243/jst.2025.v10.i06.pp28-36

[14] Todupunuri, A. (2025). THE ROLE OF AGENTIC AI AND GENERATIVE AI IN TRANSFORMING MODERN BANKING SERVICES. American Journal of AI Cyber Computing Management, 5(3), 85–93. https://doi.org/10.64751/ajaccm.2025.v5.n3.pp85-93

[15] Sushma Babburi. (2025). Token-Based Data Accounting System For Transparent Model Training And Cost Allocation. American Journal of AI Cyber Computing Management, 5(4), 463–474. https://doi.org/10.64751/ajaccm.2025.v5.n4.pp463-474

[16] Snigdha Gaddam. (2025). SOFTWARE STACK PREPARED FOR AI TRANSITIONING FROM MODULES TO MODELS. American Journal of AI Cyber Computing Management, 5(4), 451–462. https://doi.org/10.64751/ajaccm.2025.v5.n4.pp451-462

[17] Bhagwat, V. B. (2024). A simplified transition from EBS Payroll to Cloud Payroll: Benefits and Drawbacks. Journal of Computational Analysis and Applications, 33(6).

[18] Immadi, S. K. (2025). Optimizing ERP for Human Capital Management. Applied Research for Growth, Innovation and Sustainable Impact, 377–384. https://doi.org/10.1201/9781003684657-63

[19] Prodduturi, S. M. K. To Secure Your Paper as Per UGC Guidelines We Are Providing A ElectronicBar code.

[20] Doragacharla, V. R. (2026). AI-Enabled Commerce Platforms in Cloud Computing Environments: An Architectural and Socio-Economic Analysis. Journal of Computational Analysis & Applications, 35(1).

[21] Henry Cyril. (2025). AI-DRIVEN ANOMALY DETECTION, OUTAGE PREDICTION, AND SELF-HEALING IN TELECOM PROVISIONING SYSTEMS. International Journal of Applied Mathematics, 38(12s), 2817–2832. https://doi.org/10.12732/ijam.v38i12s.1589

[22] Jay Bharat Mehta. (2025). AUTONOMOUS PATCH VALIDATION FOR ZERO-DAY EXPLOITS IN ENTERPRISE CLOUDS. International Journal of Applied Mathematics, 38(4s), 1270–1285. https://doi.org/10.12732/ijam.v38i4s.685

[23] Reddy, S. K. R. (2024). Designing Blockchain Architecture to Transform Loyalty Rewards into Cryptocurrency Investments.

[24] Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.

[25] Poojari, R. (2026). Privacy-Preserving Generative AI in Healthcare Systems Using Federated Learning Approaches. International Journal of Data Science and IoT Management System, 5(1), 78-88.

[26] Kalae, U. K. (2021). Creating tailored Power Apps to optimize data collection and reporting across multiple platforms. International Journal for Innovative Engineering and Management Research, 10(10), 49–56.

[27] Saikumar, B. (2024). Optimizing Crew Scheduling and Absence Management using Microservices: Enhancing Reliability and Efficiency in Crew Management Systems. International Journal of Enhanced Research in Management & Computer Applications, 13(11), 50–55. https://doi.org/10.55948/ijermca.2024.0116

[28] Saikumar, B. (2023). Enhancing Client Engagement through AI-Driven Real-Time Reporting and Automated Alerts. International Journal of Enhanced Research in Science, Technology & Engineering, 12(11), 111–117. https://doi.org/10.55948/ijerste.2023.1115