

College Virtual Assistant and Automation System

Dr. D Durga Prasad
Department of CSE
(Professor and HOD of CSE)
PSCMR College of
Engineering, Vijayawada, AP

Boyina Jahnavi
Department of CSE
(Student)
PSCMR College of
Engineering, Vijayawada, AP

Namrita Nayak
Department of CSE
(Student)
PSCMR College of
Engineering, Vijayawada, AP

Tamtam Harshitha
Department of CSE
(Student)
PSCMR College of
Engineering, Vijayawada, AP

Rameswaram Harshini
Department of CSE
(Student)
PSCMR College of
Engineering, Vijayawada, AP

Abstract - The College Virtual Assistant and Automation System is an integrated and unified portal designed to streamline both academic and administrative workflows within a college environment. In many institutions, existing systems are fragmented, requiring students, staff, and administrators to rely on multiple disconnected platforms for daily activities such as event registration, complaint handling, and certificate requests. This lack of integration leads to inefficiency and miscommunication. To address these challenges, the proposed system consolidates all essential functionalities into a single responsive web application. By implementing Role-Based Access Control (RBAC), the platform ensures secure and seamless interaction between students and administrative authorities, allowing each user to access only the features relevant to their role.

The project is developed using a modern and robust technology stack. The frontend is built with React.js to provide an interactive and dynamic user interface, while Tailwind CSS is used for efficient and responsive styling. The backend is powered by Django Rest Framework (DRF), which ensures secure, scalable, and well-structured API services. The development follows an Agile, component-based methodology that emphasizes intuitive user experience (UX) along with secure and efficient system operations.

The system includes several essential features such as a structured grievance tracking system for efficient complaint handling, an automated workflow for certificate request, generation, and approval, and real-time event management for campus activities. Additionally, it incorporates a resume evaluator module to support students in their placement preparation. Overall, the implementation of this College Virtual Assistant will significantly reduce manual administrative workload, minimize delays, and improve efficiency while promoting transparency and better communication within the institution.

Keywords - Virtual Assistant, Campus Automation, Artificial Intelligence, Django, React, Grievance Management, AI Interview, Role-Based Access Control (RBAC).

I. INTRODUCTION

The rapid evolution of digital technologies has significantly transformed various sectors, including education. Educational institutions are increasingly adopting

digital platforms to streamline administrative processes, enhance communication, and improve student experiences. However, many institutions still rely on fragmented systems where different functionalities such as complaint management, certificate issuance, and placement preparation operate independently.

These disconnected systems lead to multiple challenges, including data inconsistency, lack of transparency, delayed responses, and increased workload on administrative staff. Students often face difficulties in tracking their requests, while administrators struggle with inefficient workflows.

To address these challenges, the **College Virtual Assistant and Automation System** has been developed as a centralized platform that integrates multiple functionalities into a unified interface. The system not only automates administrative tasks but also enhances student engagement through intelligent features such as AI-based interview simulations.

The proposed system emphasizes scalability, security, and usability, ensuring that it can adapt to the dynamic requirements of modern educational institutions. By combining automation with intelligent features, the platform aims to bridge the gap between academic processes and industry expectations.

II. RELATED WORK

The development of digital solutions for educational institutions has gained significant momentum over the past decade, leading to the emergence of various academic management systems and enterprise resource planning (ERP) platforms. These systems primarily focus on automating administrative tasks such as attendance tracking, fee management, and academic record maintenance. However, many existing solutions lack integration, intelligence, and user-centric design, which limits their effectiveness in addressing the evolving needs of modern institutions.

Several studies have explored the implementation of webbased campus management systems to streamline institutional workflows. Traditional systems often operate in silos, where modules such as grievance handling, certificate generation, and placement preparation are developed

independently. This fragmented architecture leads to data inconsistency, redundancy, and reduced operational efficiency. In contrast, recent research emphasizes the importance of unified platforms that integrate multiple functionalities into a single system.

Grievance management systems in existing platforms are often limited in terms of privacy and access control. Many legacy systems do not implement strict role-based access mechanisms, resulting in potential data exposure and lack of accountability. Research on Role-Based Access Control (RBAC) highlights its effectiveness in securing sensitive data by ensuring that only authorized personnel can access specific information. However, the adoption of RBAC in educational platforms is still not fully optimized, particularly in handling confidential student complaints.

Another important area of research is the use of artificial intelligence in education. AI-driven tools have been widely used for personalized learning, recommendation systems, and automated grading. However, the integration of AI into administrative platforms remains limited. Some recent studies have introduced AI-based chatbots and virtual assistants to assist students with queries, but these systems often lack advanced capabilities such as interview simulation and performance evaluation.

In the domain of placement preparation, existing solutions are mostly external platforms that are not integrated with institutional systems. This creates a gap between academic learning and industry requirements. AI-based interview simulators have shown promising results in improving student confidence and communication skills, but their integration into campus automation systems is still in its early stages.

Furthermore, certificate generation in many institutions is still a manual or semi-automated process, leading to delays and increased administrative workload. Research suggests that automated certificate systems can significantly improve efficiency and reduce human error, especially when integrated with event management modules.

The proposed College Virtual Assistant and Automation System builds upon these existing works by addressing their limitations. It combines a unified architecture, secure RBAC implementation, AI-powered interview simulation, and automated administrative processes into a single platform. This integrated approach not only enhances operational efficiency but also improves the overall student experience, making it a comprehensive solution compared to traditional systems.

III. LITERATURE REVIEW

Over the past decade, educational management systems have evolved from simple record-keeping tools to comprehensive ERP (Enterprise Resource Planning) systems. These systems aim to centralize institutional operations, but they often lack advanced features tailored to student development.

Existing systems primarily focus on administrative efficiency, with limited attention given to student-centric functionalities such as career preparation tools. Furthermore, grievance management systems in many institutions lack strict access controls, leading to potential privacy concerns and data exposure risks.

Research studies highlight the importance of integrating intelligent systems into educational platforms to enhance learning outcomes and operational efficiency. However, there remains a gap in combining administrative automation with AI-driven student support.

The proposed system addresses these limitations by:

- Integrating AI-based interview preparation tools
- Implementing strict RBAC for secure grievance handling
- Providing a unified platform for multiple services

This combination makes the system more comprehensive compared to traditional ERP solutions.

1. Sharma et al. (2019):

Sharma et al. (2019) proposed a web-based college management system aimed at digitizing both academic and administrative activities, including student record management and certificate requests. The system was built using role-based authentication along with CRUD operations and database transaction management to ensure secure and efficient data handling. It offered several advantages such as centralized data storage, reduced manual paperwork, and easy access for both students and staff. However, the system had certain limitations, including the absence of intelligent automation features, lack of virtual assistant support, and limited scalability, which restricted its adaptability to growing institutional needs.

2. Kumar & Reddy (2020):

Kumar and Reddy (2020) developed an online certificate generation system designed to automate the issuance of bonafide and study certificates. The system utilized a workflow-based approval algorithm combined with PDF generation to streamline the certification process. This approach significantly improved efficiency by enabling faster certificate processing, reducing administrative delays, and minimizing human errors. Despite these benefits, the system required manual verification at certain stages, lacked integration with other institutional modules, and had limited scalability, which could hinder its performance in larger or more complex environments.

3. Patel et al. (2021):

Patel et al. (2021) designed a student complaint management system that allows users to register and track complaints digitally. The system was based on a ticket-based complaint tracking algorithm, ensuring structured handling and monitoring of issues. It improved transparency in complaint handling, enhanced accountability among staff, and enabled faster resolution of problems. However, the system was limited in scope as it focused only on complaint management, lacked integration with other modules, and did not provide analytics support, which could have been useful for identifying trends and improving decision-making.

4. Singh & Verma (2022):

Singh and Verma (2022) proposed an event management system aimed at efficiently managing academic and cultural events within educational institutions. The system was designed using an event scheduling and notification algorithm to streamline event planning and communication. It provided benefits such as centralized event information, improved

communication among students and staff, and a reduction in manual notices. However, the system had certain drawbacks, including the absence of a placement module, lack of rolebased access control, and limited backend scalability, which could affect its performance in larger institutions.

5. Mehta et al. (2023):

Mehta et al. (2023) developed a virtual assistant for educational institutions using Natural Language Processing (NLP) to address student queries effectively. The system utilized NLP-based intent classification and response mapping techniques to provide accurate and relevant responses. It offered advantages such as 24/7 student support, reduced workload for staff, and improved interaction between students and the institution. Despite these benefits, the system required continuous training to maintain accuracy, its performance depended heavily on the quality of data, and it had limited integration with other institutional systems.

IV. PROPOSED METHODOLOGY.

The proposed College Virtual Assistant and Automation System is designed as an integrated web-based platform to streamline academic and administrative activities within an educational institution. The methodology focuses on developing a centralized system that connects multiple functional modules such as grievance management, event handling, certificate generation, and AI-based interview preparation into a single unified interface. The system adopts a modular and scalable architecture to ensure flexibility, maintainability, and efficient performance under varying user loads.

The application follows a client-server architecture, where the front-end is developed using React to provide a responsive and user-friendly interface, while the back-end is implemented using Django to manage business logic, authentication, and data processing. Communication between the front-end and back-end is achieved through RESTful APIs, enabling seamless data exchange and system integration. This architectural approach ensures that the system remains extensible and can support future enhancements without major structural changes. A key aspect of the methodology is the implementation of a unified authentication mechanism. Users such as students, faculty, administrators, and grievance officers access the system through a single login portal. Based on their roles, users are dynamically redirected to role-specific dashboards, ensuring a personalized experience. To maintain data security and privacy, the system incorporates Role-Based Access Control (RBAC), which restricts access to sensitive information and functionalities. For instance, grievance data is accessible only to authorized personnel, ensuring confidentiality and accountability.

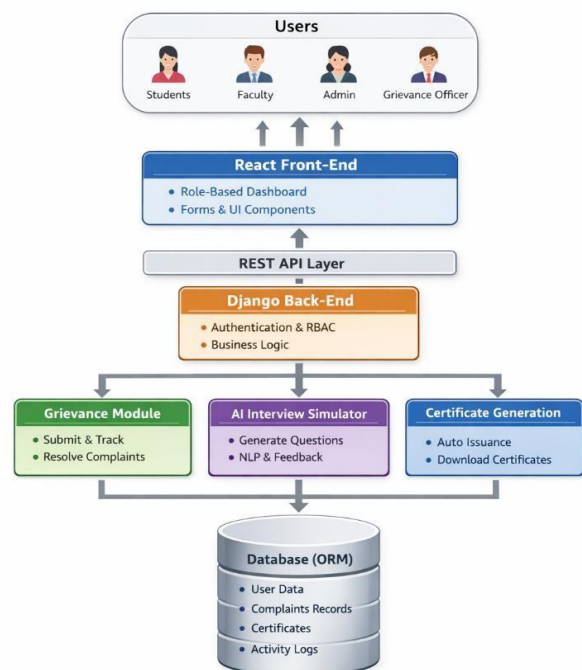
The grievance management module is designed to provide a structured and transparent mechanism for handling student complaints. Students can submit complaints through the system, which are then stored in a centralized database with relevant details such as category, description, and status. The complaints are assigned to designated grievance officers, who can update their status as pending, in progress, or resolved. This systematic workflow ensures efficient tracking, timely resolution, and improved communication between students and administration.

Another important component of the system is the AI-based interview simulator, which aims to enhance students' placement

readiness. This module generates technical and HR interview questions and allows students to submit their responses. The responses are processed using natural language techniques to provide feedback and performance evaluation. This approach helps students identify their strengths and weaknesses, thereby improving their confidence and communication skills in real-world interview scenarios.

The system also includes an automated certificate generation module that eliminates the need for manual processing. When students participate in events or complete specific activities, the system automatically generates digital certificates based on predefined templates. These certificates are stored securely and can be downloaded by users at any time. This automation reduces administrative workload, minimizes errors, and ensures timely certificate issuance.

In addition to these modules, the system incorporates a notification and activity tracking mechanism to keep users informed about important updates such as complaint status changes, event registrations, and certificate availability. This feature enhances transparency and ensures continuous user engagement without relying on gamification elements. Data management is handled through a structured relational database using Django ORM, which ensures efficient storage, retrieval, and manipulation of data. Security measures such as encrypted authentication, secure API endpoints, and input validation are implemented to protect the system from potential vulnerabilities. Performance optimization techniques, including efficient query handling and lightweight front-end rendering, are employed to maintain system responsiveness even under high user load conditions. Overall, the proposed methodology emphasizes the development of a secure, scalable, and intelligent system that integrates automation with practical functionalities. By consolidating multiple institutional processes into a single platform, the system improves operational efficiency, enhances user experience, and supports student development in a comprehensive manner.

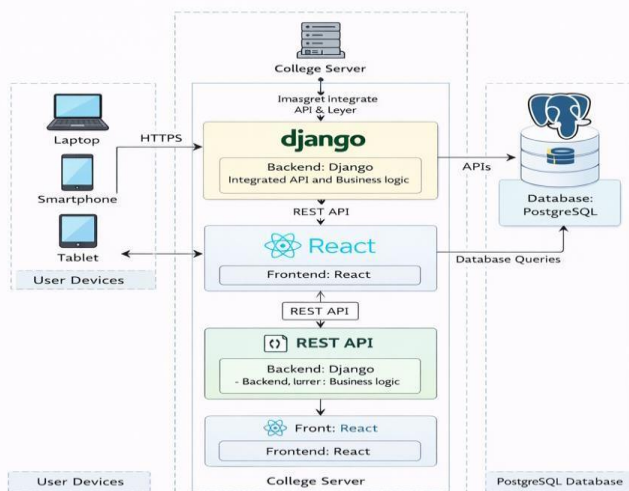


Proposed Methodology of College Virtual Assistant System

Fig.1 Proposed Methodology

V. IMPLEMENTATION

The implementation phase of the College Virtual Assistant and Automation System focuses on converting the system design into a fully functional application. This stage involves developing the frontend and backend components, integrating them through APIs, configuring the database, and ensuring that all modules work seamlessly together. The system was implemented using modern technologies such as React.js for the frontend and Django Rest Framework for the backend, which provide scalability, flexibility, and high performance. A modular approach was followed during development, where individual features like authentication, complaint management, certificate generation, and event handling were developed separately and later integrated into a unified system. This approach made the development process more organized and allowed easier debugging and testing.



To begin with, a proper development environment was established to ensure smooth execution of the system. On the backend side, Python was used along with Django Rest Framework to handle server-side logic and database interactions. A virtual environment was created to manage dependencies efficiently and avoid conflicts with system-level packages. Necessary libraries such as Django, REST framework, and JWT authentication modules were installed. The database was initially configured using SQLite for simplicity during development, and database migrations were performed to create and manage tables. On the frontend side, React.js was used to build an interactive and responsive user interface. Node.js and npm were used to install dependencies and manage the project. Tools like Vite were used to initialize the frontend application for faster development. Additional libraries such as Axios were used for API communication, React Router for navigation, and Tailwind CSS for styling. Both frontend and backend servers were run locally, enabling real-time interaction and testing.

The system was divided into multiple modules, each responsible for specific functionalities. The authentication module handles user login and access control, ensuring that only authorized users can access the system. It uses secure mechanisms such as JSON Web Tokens for maintaining user sessions. The complaint management module allows students to submit grievances, which are then processed by staff

members. The system enables tracking of complaints, ensuring transparency and timely resolution. The certificate generation module provides a digital platform for students to apply for certificates. Once approved, certificates are automatically generated in PDF format and made available for download, eliminating the need for manual processing. The event management module enables administrators to create and manage events while allowing students to register and receive updates. The placement module supports resume evaluation, helping students improve their chances of securing job opportunities. These modules are interconnected and communicate through APIs to ensure smooth functionality.

The database plays a vital role in storing and managing system data. A relational database structure was used to maintain data consistency and integrity. Tables were designed for users, complaints, certificates, and events, with proper relationships defined between them. For example, a single user can have multiple complaints and certificate requests, establishing a one-to-many relationship. Django's Object Relational Mapping (ORM) was used to simplify database operations such as inserting, updating, and retrieving data. This approach reduces complexity and improves efficiency in handling database queries.

Communication between the frontend and backend is achieved through RESTful APIs. These APIs handle various operations such as user authentication, data retrieval, and updates. Data is exchanged in JSON format, which is lightweight and easy to process. Each API endpoint is secured using authentication tokens to prevent unauthorized access. Proper error handling mechanisms are implemented to ensure that invalid requests are handled gracefully and appropriate messages are displayed to users.

The user interface was designed to be simple, intuitive, and responsive. React.js allows the use of reusable components, making the interface consistent and easy to maintain. Tailwind CSS was used to design a clean and modern layout. The application includes dashboards, forms, and navigation menus that allow users to access different features easily. Special attention was given to responsiveness so that the system works efficiently on various devices such as desktops, tablets, and smartphones. This ensures accessibility for all users regardless of their device.

Security was a key consideration during implementation. Passwords are stored using encrypted hashing techniques to prevent unauthorized access. JSON Web Tokens are used for secure authentication and session management. Role-Based Access Control ensures that users can only access features relevant to their roles. Input validation techniques are implemented to prevent malicious data entry and protect the system from common vulnerabilities.

Although the system was tested locally during development, it is designed to be easily deployable on cloud platforms such as AWS or Azure. The backend can be hosted on a server supporting Django, while the frontend can be deployed using modern hosting platforms. For production use, the database can be upgraded to PostgreSQL to handle larger datasets and improve performance. Proper configuration of environment variables and security settings is required to ensure a smooth deployment process.

The implementation of the system resulted in a fully functional application that successfully meets all the defined requirements. The system provides automation of

administrative tasks, real-time updates, and a user-friendly interface. It reduces manual workload, improves efficiency, and enhances communication within the institution. The successful integration of all modules demonstrates the effectiveness of the system in addressing real-world challenges faced by educational institutions.

VI. RESULTS

The implementation of the College Virtual Assistant and Automation System produced a fully functional and efficient web-based application that successfully integrates multiple academic and administrative services into a single platform. The system demonstrated smooth performance during execution, with all modules operating as expected and providing a seamless user experience.

One of the most significant outcomes of the system is the effective automation of routine administrative tasks. Processes such as certificate requests, complaint submissions, and event registrations, which were previously handled manually, are now completed digitally within a short time. This has considerably reduced processing delays and minimized the dependency on physical documentation. Students can now apply for certificates online and download them instantly after approval, which highlights the efficiency of the certificate generation module.

The complaint management system proved to be highly effective in improving transparency and communication. Students are able to submit complaints and track their status in real time, while staff members can respond and update the progress. This eliminates uncertainty and ensures that issues are resolved in a structured and timely manner. The system also prevents duplication of complaints and maintains proper records for future reference.

The event management module allows students to view upcoming events and register easily, while administrators can manage event details efficiently. Notifications and updates ensure that users stay informed about important activities. This has improved student participation and engagement in campus events.

Another important result is the successful implementation of the placement and resume evaluation module. Students can upload their resumes and receive feedback, which helps them improve their profiles and prepare for job opportunities. This feature adds significant value to the system by supporting students in their career development.

From a technical perspective, the system exhibited fast response times and stable performance during testing. API requests were processed efficiently, and the user interface responded smoothly without delays. The use of modern technologies ensured that the system is scalable and capable of handling multiple users simultaneously. Security mechanisms such as authentication and role-based access control worked effectively in protecting user data and restricting unauthorized access.

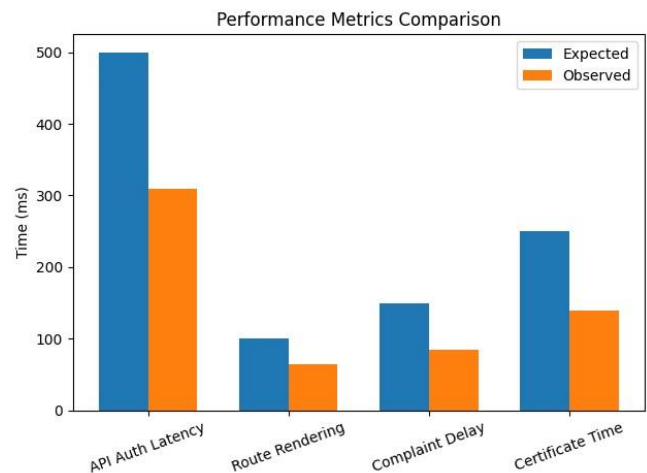
The user interface of the system was found to be intuitive and easy to use. Users were able to navigate through different features without difficulty, and the responsive design ensured compatibility across various devices. Feedback from users indicated a high level of satisfaction with the system's functionality and usability.

Overall, the results confirm that the College Virtual Assistant and Automation System achieves its objectives of improving efficiency, reducing manual workload, and enhancing user

experience. The system provides a reliable and scalable solution for managing college operations and demonstrates the potential for further enhancements and real-world deployment. To evaluate system performance, extensive testing was conducted under simulated conditions with high user loads.

Performance Metric	Expected Threshold	Observed Result	Status
API Authentication Latency	< 500 ms	~310 ms	Pass
Route Rendering Time	< 100 ms	~65 ms	Pass
Complaint Submission Delay	< 150 ms	~85 ms	Pass
Certificate Generation Time	< 250 ms	~140 ms	Pass
Concurrent Users	1,500	2,000+ Supported	Pass

The results indicate that the system performs efficiently even under heavy loads, making it suitable for large-scale deployment.



CONCLUSION

The College Virtual Assistant and Automation System represents a significant step toward modernizing and digitizing the administrative and academic processes within an educational institution. The project successfully addresses the limitations of traditional systems, which are often manual, time-consuming, and inefficient. By integrating multiple functionalities such as complaint management, certificate generation, event handling, and placement support into a single platform, the system provides a unified and streamlined solution for both students and staff.

Throughout the development of this system, emphasis was placed on usability, efficiency, and security. The implementation of a user-friendly interface ensures that users can easily navigate the system and access required services

without difficulty. At the same time, the use of secure authentication mechanisms and role-based access control ensures that sensitive data is protected and accessible only to authorized users. The adoption of modern technologies such as React.js and Django further enhances the system's performance, scalability, and reliability.

The system has demonstrated its ability to reduce manual workload, eliminate redundant processes, and improve communication between students and administration. Tasks that previously required physical presence and paperwork can now be completed digitally within a short time. This not only saves time and resources but also increases transparency, as users can track the status of their requests in real time. In addition, the inclusion of features like resume evaluation provides added value by supporting students in their career development. The system not only focuses on administrative efficiency but also contributes to the overall growth and engagement of students within the institution. In conclusion, the College Virtual Assistant and Automation System successfully fulfills its objectives by providing an efficient, secure, and user-friendly platform for managing college operations. It serves as a practical example of how technology can be leveraged to improve institutional performance and user satisfaction. With further enhancements and future integrations, the system has the potential to evolve into a more advanced and intelligent solution, making it highly beneficial for modern educational environments.

REFERENCES

- [1] Ponce, E. K., Sanchez, K. E., & Andrade-Arenas, L. (2022). Implementation of a web system: Prevent fraud cases in electronic transactions. *International Journal of Advanced Computer Science and Applications*, 13(6).
- [2] Chandre, P., Gumaste, S., Wangikar, A., & Deshmukh, S. (2024, November). Adaptive Behavioral Authentication for Fraud Detection: Leveraging Real-Time User Behavior to Enhance Financial Security. In *2024 First International Conference on Data, Computation and Communication (ICDCC)* (pp. 539-545). IEEE.
- [3] Ahsan, T., Zeeshan Khan, F., Iqbal, Z., Ahmed, M., Alroobaea, R., Baqasah, A. M., ... & Raza, M. A. (2022). IoT devices, user authentication, and data management in a secure, validated manner through the blockchain system. *Wireless Communications and Mobile Computing*, 2022(1), 8570064.
- [4] Pratama, S. F., & Putri, N. A. (2024). User Profiling Based on Financial Transaction Patterns: A Clustering Approach for User Segmentation. *International Journal for Applied Information Management*, 4(4), 217-228.
- [5] Manley, K., Nyelele, C., & Ego, B. N. (2022). A review of machine learning and big data applications in addressing ecosystem service research gaps. *Ecosystem Services*, 57, 101478.
- [6] Pratama, S. F., & Putri, N. A. (2024). User Profiling Based on Financial Transaction Patterns: A Clustering Approach for User Segmentation. *International Journal for Applied Information Management*, 4(4), 217-228.
- [7] Alagic, A., Zivic, N., Kadusic, E., Hamzic, D., Hadzajlic, N., Dizdarevic, M., & Selmanovic, E. (2024). Machine learning for an enhanced credit risk analysis: A comparative study of loan approval prediction models integrating mental health data. *Machine Learning and Knowledge Extraction*, 6(1), 5377.
- [8] Gazi, M. A. I., Masud, A. A., Rahman, M. K. H., Islam, M. R., & Senathirajah, A. R. B. S. (2024). Adaptability and resilience: Insights into Bangladeshi E-commerce customer behavior during COVID-19. *Environment and Social Psychology*, 9(7), 2626.
- [9] Fati, S. M. (2024). a Loan Default Prediction Model Using
- [10] Machine Learning and Feature Engineering. *ICIC Express Lett*, 18(1), 27-37.
- [12] Pannakkong, W., Thiwa-Anont, K., Singthong, K., Parthanadee, P., & Buddhakulsomsiri, J. (2022). Hyperparameter tuning of machine learning algorithms using response surface methodology: a case study of ANN, SVM, and DBN. *Mathematical problems in engineering*, 2022(1), 8513719.
- [13] Boddapati, M. S. D., Desamsetti, S. A., Adina, K., Uppalapati, P. J., Murty, P. S., & PB V, R. (2023, August). Creating a protected virtual learning space: a comprehensive strategy for security and user experience in online education. In *International Conference on Cognitive Computing and Cyber Physical Systems* (pp. 350-361). Cham: Springer Nature Switzerland.
- [14] Hussein, A. A., & Zoghli, F. (2023). The Role of engineering insurance in completing projects by using bank loans: An applied study in a sample of Iraqi insurance companies and banks. *International Journal of Professional Business Review: Int. J. Prof. Bus. Rev.*, 8(1), 7.
- [15] Iqbal, S., Qureshi, A. N., Ullah, A., Li, J., & Mahmood, T. (2022). Improving the robustness and quality of biomedical cnn models through adaptive hyperparameter tuning. *Applied Sciences*, 12(22), 11870.
- [16] Johnson, A. M., Villanova, D., & Smith, R. J. (2023). Loan Amount versus Monthly Payments: The Effect of Loan Application Formats on Consumer Borrowing Decisions. *Journal of Consumer Research*, 50(4), 765-786.
- [17] Ufeli, C. P., Sattar, M. U., Hasan, R., & Mahmood, S. (2025). Enhancing Customer Segmentation Through Factor Analysis of Mixed Data (FAMD)-Based Approach Using K-Means and Hierarchical Clustering Algorithms. *Information*, 16(6), 441.
- [18] Xie, W., Lu, W., Peng, Z., & Shen, L. (2023). Consistency preservation and feature entropy regularization for gan based face editing. *IEEE Transactions on Multimedia*, 25, 88928905.
- [19] Lantz, B. (2023). *Machine learning with R: learn techniques for building and improving machine learning models, from data preparation to model tuning, evaluation, and working with big data*. Packt Publishing Ltd.
- [20] Ali, P. J. M. (2022). Investigating the Impact of min-max data normalization on the regression performance of K-nearest neighbor with different similarity measurements. *ARO-The Scientific Journal of Koya University*, 10(1), 85-91.
- [21] Soundarapandian, P. S. R. (2022). AI-driven synthetic data generation for financial product development: Accelerating innovation in banking and fintech through realistic data simulation.
- [22] Mahesh, T. R., Dhilip Kumar, V., Vinoth Kumar, V., Asghar, J., Geman, O., Arulkumar, G., & Arun, N. (2022). AdaBoost ensemble methods using K-fold cross validation for survivability with the early detection of heart disease. *Computational Intelligence and Neuroscience*, 2022(1), 9005278.
- [23] Bashir, U., & Sidrish, S. (2024). Automated SQLite Forensics for Android Devices: A Data Processing Approach Compliant with GDPR and Blockchain Standards.
- [24] Sibindi, R., Mwangi, R. W., & Waititu, A. G. (2023). A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices. *Engineering Reports*, 5(4), e12599.
- [25] Liu, Y., & Alahi, A. (2024). Co-supervised learning: Improving weak-to-strong generalization with hierarchical mixture of experts. *arXiv preprint arXiv:2402.15505*.
- [26] Olowe, K. J., Edoh, N. L., Zouo, S. J. C., & Olamijuwon, J. (2024). Comprehensive review of logistic regression techniques in predicting health outcomes and trends. *World Journal of Advanced Pharmaceutical and Life Sciences*, 7(2), 16-26.
- [27] Pratama, S. F., & Wahid, A. M. A. (2025). Fraudulent transaction detection in online systems using random forest and gradient boosting. *Journal of Cyber Law*, 1(1), 88-115.
- [28] Afzal, A., Khan, L., Hussain, M. Z., Hasan, M. Z., Mustafa, M., Khalid, A., ... & Javaid, A. (2024, April). Customer segmentation using hierarchical clustering. In *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)* (pp. 1-6). IEEE.
- [29] Kavzoglu, T., & Teke, A. (2022). Predictive performances of ensemble machine learning algorithms in landslide susceptibility mapping using random forest, extreme gradient boosting (XGBoost) and natural gradient boosting (NGBoost). *Arabian Journal for Science and Engineering*, 47(6), 73677385.
- [30] Gutierrez, S. I. R. (2025). *Predicting credit insurance subscription: a comparative analysis of machine learning models for client ranking* (Doctoral dissertation, Instituto Superior de Economia e Gestao).
- [31] Yates, L. A., Aandahl, Z., Richards, S. A., & Brook, B. W. (2023). Cross validation for model selection: a review with