INTERNATIONAL JOURNAL OF ECONOMIC SOCIAL SCIENCE AND MANAGEMENT LAW
Peer Reviewed, Referred & Indexed Journal
E-ISSN: 3070-0558
VOL.7, NO.2(1) (2026)

An Edutech Admission Consultancy Platform with AI College Predictor Module

Mr. Debabrata Behera
Dept. of CSE
GIFT Autonomous
Bhubaneswar, Odisha, India

Mr. Satyanarayan Chhualsingh
Dept. of CSE
GIFT Autonomous
Bhubaneswar, Odisha, India

Er. Jagannath Ray
Assistant Professor, Dept. of CSE
GIFT Autonomous
Bhubaneswar, Odisha, India

***Abstract*—The rapid advancement of internet technologies and intelligent web-based systems has significantly transformed the educational ecosystem, especially in the area of admission counseling and academic guidance. Students seeking higher education often face difficulties while selecting suitable colleges and courses due to scattered information, complex counseling procedures, inconsistent cutoff records, and lack of proper guidance systems. Traditional educational consultancy services mainly rely on manual operations, physical counseling sessions, and static information delivery methods, which become inefficient when handling large numbers of students during admission periods. Furthermore, most existing admission guidance systems do not provide intelligent prediction support for estimating admission probabilities based on historical data and candidate performance.**

This research presents the design, development, and implementation of a Web-Based Edutech Admission Consultancy Platform integrated with an AI-Based Linear Probability College Predictor Module. The proposed platform provides centralized educational guidance, structured college information management, inquiry handling systems, chatbot assistance, and intelligent admission prediction functionalities within a single responsive web application. The AI-Based College Predictor analyzes candidate rank, category, preferred branch, and historical cutoff trends to estimate admission probability and classify institutions into Safe, Moderate, and Dream categories.

The proposed platform is developed using HTML5, CSS3, Bootstrap 5, JavaScript, PHP, and MySQL technologies. The system supports secure database management, responsive user interfaces, chatbot communication mechanisms, and scalable administrative operations. Experimental analysis demonstrates that the platform improves counseling efficiency, reduces manual effort, enhances accessibility, and simplifies institutional selection processes for students. The modular architecture also supports future scalability for cloud deployment, machine learning integration, mobile applications, and advanced predictive analytics systems.

Keywords— Educational Technology, Admission Counseling System, AI-Based College Predictor, Linear Probability Model, Chatbot System, PHP, MySQL, Full-Stack Web Development, Responsive Web Application

I. INTRODUCTION

A. Background

The digital transformation of educational services has revolutionized the way students access academic information, counseling support, and admission-related services. In modern educational environments, students extensively depend on online platforms to explore colleges, analyze cutoff trends, compare courses, and evaluate institutional opportunities. The rapid increase in internet accessibility and digital learning environments has accelerated the demand for intelligent web-based educational consultancy systems.

Despite the availability of multiple educational websites and admission portals, students still encounter significant difficulties during college selection processes. Information related to institutional cutoffs, branch availability, reservation categories, counseling schedules, and placement records is often scattered across different websites. Such fragmented information creates confusion and increases the complexity of decision-making during admission counseling periods.

Traditional educational consultancy systems mainly operate through manual counseling methods where students interact physically with counselors and admission experts. These systems depend heavily on paperwork, offline communication, and manual analysis of historical cutoff records. As the number of engineering and higher education aspirants increases every year, traditional systems become inefficient and difficult to scale.

Modern web technologies, artificial intelligence techniques, and database management systems provide opportunities for building intelligent educational platforms capable of automating counseling operations and providing personalized admission guidance. AI-based prediction systems can analyze historical cutoff records and candidate-specific parameters to estimate admission probabilities more efficiently than manual systems.

To address these challenges, the proposed Edutech Admission Consultancy Platform introduces a centralized, responsive, and intelligent web-based solution capable of integrating educational guidance, chatbot assistance, and AI-based admission prediction functionalities into a single application environment.

B. Problem Statement

Existing educational consultancy systems face several technological and operational limitations. Most platforms mainly focus on providing static information regarding colleges and courses without offering intelligent prediction support for estimating admission possibilities. Students are often required to manually compare their ranks with historical cutoff records

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com

to identify suitable institutions, which is time-consuming and error-prone.

Traditional counseling systems also rely heavily on manual operations that consume significant administrative effort. During admission periods, counseling organizations face challenges in handling large volumes of student inquiries efficiently. The absence of centralized information management systems and automated recommendation engines further reduces operational efficiency.

Many existing platforms lack responsive user interfaces, chatbot support systems, secure administrative controls, and scalable architectures for handling large datasets and multiple users simultaneously. Security vulnerabilities, inefficient backend operations, and poor database structures also affect system reliability and maintainability.

Therefore, there is a strong need for a modern educational consultancy platform capable of integrating AI-based admission prediction systems, responsive interfaces, chatbot assistance, secure database operations, and centralized administrative management within a single scalable architecture.

C. Objectives

The major objectives of the proposed system are listed below:

- To develop a responsive and scalable web-based educational consultancy platform.
- To provide centralized information regarding colleges, branches, counseling procedures, and admission-related activities.
- To implement an AI-Based College Predictor Module for estimating admission probabilities using historical cutoff data.
- To classify colleges into Safe, Moderate, and Dream categories based on prediction analysis.
- To integrate a chatbot assistance module for handling admission-related user queries.
- To develop an admin dashboard for centralized management of institutional records, chatbot data, inquiries, and prediction datasets.
- To reduce manual counseling effort and improve decision-making efficiency for students.
- To support future scalability for machine learning integration, cloud deployment, and mobile application support.

D. Scope of the Project

The proposed Edutech Admission Consultancy Platform is designed for students, educational consultancy organizations, institutions, and counseling agencies. The platform provides centralized educational guidance, intelligent admission prediction, chatbot communication, and inquiry management functionalities.

The application supports responsive access across desktops, laptops, tablets, and smartphones. The system architecture also supports future enhancements including machine learning-based recommendation systems, cloud deployment, real-time analytics, multilingual interfaces, and mobile application integration.

II. LITERATURE REVIEW

A. Online Educational Consultancy Platforms

Educational consultancy platforms have become increasingly important in modern admission systems. These platforms allow students to access information regarding institutions, branches, counseling schedules, admission procedures, and eligibility criteria through online systems. Modern platforms reduce paperwork and simplify communication between students and institutions.

Several educational websites and counseling platforms provide college search functionalities and cutoff information. However, most systems primarily deliver static information and lack intelligent recommendation or prediction support. Students are still required to manually analyze historical cutoff trends and compare them with their entrance examination ranks.

B. Artificial Intelligence in Educational Systems

Artificial intelligence technologies are widely used in modern educational systems for recommendation analysis, student performance prediction, and intelligent counseling support. AI-based systems can analyze historical datasets and generate personalized recommendations based on user-specific parameters.

Machine learning algorithms such as regression analysis, classification models, and probability estimation methods are increasingly being used in educational applications. However, many advanced machine learning systems require large training datasets and high computational resources. Lightweight probability-based systems provide efficient alternatives for educational consultancy applications.

C. Chatbot Systems in Education

Educational chatbots are becoming popular for automating communication processes and reducing administrative workload. Chatbots can provide instant responses to student queries regarding admissions, counseling procedures, branch selection, and institutional details.

Rule-based chatbot systems are widely implemented because of their simplicity and lower computational requirements. These systems improve accessibility and reduce communication delays during high-volume counseling periods.

D. Research Gap

Although several educational platforms and counseling systems are available, significant limitations still exist. Most platforms do not integrate AI-based admission prediction systems with centralized consultancy operations. Existing systems rarely provide intelligent categorization of institutions based on student rank probability analysis.

Many platforms also lack chatbot integration, responsive interfaces, secure backend architectures, and centralized administrative dashboards. Therefore, there is a need for a

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com

modern full-stack educational consultancy platform capable of integrating AI-based prediction, chatbot assistance, and secure information management systems within a scalable architecture.

III. SYSTEM OVERVIEW

A. Proposed System

The proposed Edutech Admission Consultancy Platform is designed as a centralized educational guidance, career alignment, and objective admission analysis system. The platform integrates responsive user frontend interfaces, secure backend processing modules, deterministic AI-based prediction engines, rule-guided conversational chatbot systems, and normalized relational database management operations into a single cohesive web application. This comprehensive integration addresses the widespread challenges of information fragmentation and lack of personalized counseling infrastructure typically observed during highly competitive higher education choice-locking and verification windows.

Prospective students can browse verified institutional profiles, explore academic engineering branches, evaluate placement trends, submit managed inquiry forms, interact with the persistent chatbot assistance interface, and leverage AI-based prediction tools to safely identify suitable colleges aligned with their unique ranks, categories, and educational preferences. Simultaneously, authenticated system administrators are equipped with a secure, centralized master dashboard to oversee candidate records, update year-on-year historical cutoff datasets, monitor dynamic chatbot logs, and resolve user inquiries. By automating repetitive administrative overhead and eliminating manual spreadsheet tracking errors, the platform creates an efficient multi-role ecosystem that enhances student accessibility while maintaining absolute data integrity.

B. System Architecture

The proposed platform follows a highly decoupled and optimized three-tier architecture designed to enforce clean separation of functional concerns, ensure high transaction processing speeds, maximize long-term maintainability, and handle concurrent high-volume usage spikes smoothly:

1) Presentation Layer
2) Business Logic Layer
3) Database Layer

The Presentation Layer represents the user-facing portal and is implemented using structural HTML5 markup, customized CSS3 style sheets, the Bootstrap 5 fluid grid layout engine, and asynchronous JavaScript execution routines. This layer manages interface consistency, responsive multi-screen rendering across diverse mobile and desktop monitors, input constraints enforcement, and asynchronous client-side interaction loops that exchange localized JSON packets with backend server components without triggering disruptive page refreshes.

The Business Logic Layer functions as the central transactional processing unit of the platform and is built using structural PHP scripts executing on a secure web server environment. This server application layer intercepts incoming form actions, enforces rigid parameter sanitization to defend against malicious script injection exploits, handles multi-role session security tokens, executes calculation logic rules for the prediction matrices, and coordinates connection pools to fetch or update underlying persistent storage files.

The Database Layer utilizes the MySQL relational database management system to handle and protect persistent application assets and operational directories. The underlying storage design utilizes performance-tuned indexes and foreign key cascades across highly normalized tables to minimize multi-table join latencies during concurrent lookups, securely organizing administrative credentials, student contact logs, university specs, branch rank thresholds, and testimonial string parameters.

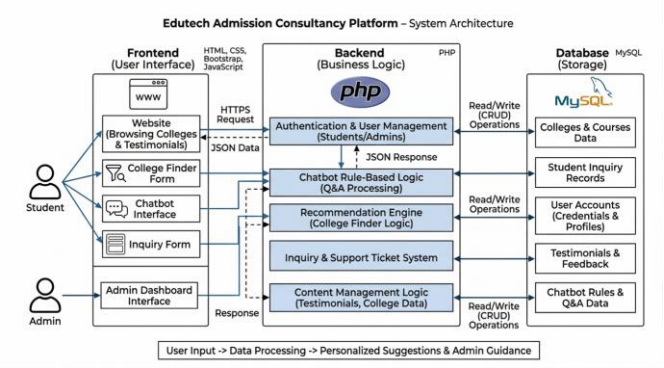

Fig. 1. Overall System Architecture

C. Key Functional Modules

The operational workflows and software tasks within the application framework are explicitly distributed across six specialized key functional modules that coordinate together to process incoming requests:

- **Student Inquiry Module:** Manages candidate onboarding and contact workflows by providing semantic data capture fields that validate, sanitize, and pass applicant queries directly into backend storage tables for immediate administrative review and ticketing.
- **College Information Module:** Renders a device-responsive data grid that enables students to discover, search, filter, and compare institutional variables including branch configurations, fee parameters, structural criteria, and geographic location tags.
- **AI-Based College Predictor Module:** Serves as the primary analytical engine of the application, parsing applicant exam ranks and demographics using a deterministic Linear Probability Model against historical seat-allocation logs to output tiered entry predictions.
- **Educational Chatbot Module:** Implements an interactive client-side conversational interface guided by structured decision-tree nodes to automate context discovery, resolve common admission queries, and route users toward targeted platform tools.

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com

- **Admin Dashboard Module:** Provides a secure, single-pane graphical management panel where authenticated supervisors can monitor application performance metrics, moderate testimonials, modify core database values, and import updated cutoff datasets.
- **Database Management Module:** Intersects all query routines to enforce data integrity, maintain transactional connection security rules, run clean backup optimizations, and handle rapid database lookups across normalized collections.

IV. METHODOLOGY

A. System Workflow

The operational workflow of the Edutech Admission Consultancy Platform is designed as an automated, multi-tiered pipeline that orchestrates student interactions, input processing, database matching, and analytical rendering. The workflow execution sequence begins when a prospective student accesses the web application's frontend presentation interface via a standard web browser. The user is presented with a responsive dashboard that serves as a gateway to various educational and counseling services. From this entry point, students can seamlessly navigate through different functional pathways: browsing verified institutional profiles, examining comprehensive branch specifications, submitting targeted academic inquiries, engaging with the rule-guided automated chatbot assistant, or initiating the predictive analysis cycle within the AI-Based College Predictor Module.

To maintain system efficiency, reduce unnecessary backend server load, and protect data processing pipelines from structural errors, all user inputs are immediately subjected to rigorous validation checks. Client-side validation routines, written in modular native JavaScript, intercept form submission events in real time to enforce format compliance, field length restrictions, pattern matching (such as telephone and email structures), and strict data type constraints. Payloads that fail to pass these local validation gates are blocked instantly, and the interface dynamically displays context-aware error indicators to guide the user.

Once the frontend validation engine confirms payload integrity, the client-side script serializes the attributes into clean data structures and securely transmits them to the server-side backend layer. This communication is executed using non-blocking HTTP POST or asynchronous AJAX design patterns, which preserve application state and allow results to update dynamically without requiring a full browser page refresh. Upon reaching the backend container, server-side PHP processing scripts intercept the incoming variables. The backend engine applies a second tier of validation, parsing input strings through strict sanitization protocols and escaping routines to eliminate any malicious patterns before database interaction begins.

Following input cleansing, the business logic layer evaluates the request routing context to determine whether to perform persistent transactional data writes or to launch algorithmic lookup operations. For core analytical queries, the backend engine initializes a secure connection to the persistent data store. It maps candidate profile criteria directly against indexed relational tables within the MySQL database to isolate matching datasets. The prediction module then executes computational calculations, formats the analytical results into a structured data array, converts the payload into a clean JSON string, and returns it to the web view. Finally, frontend script handlers catch the incoming JSON response, clear the designated container element, and use CSS transition animations to render interactive recommendation cards directly onto the user's screen.

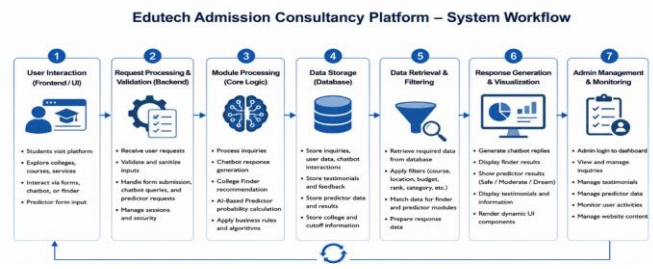

Fig. 2. System Workflow Diagram

B. AI-Based College Predictor Methodology

The AI-Based College Predictor Module implements a deterministic evaluation model engineered to replace arbitrary, subjective human guess-work with structured, mathematical probability metrics. The analytical engine evaluates the candidate's competitive academic standing by mapping their objective merit score directly against historical seat-allocation boundaries recorded for specific institutional majors across previous counseling cycles. This mathematical framework eliminates the uncertainty and cognitive overhead that students traditionally experience during highly complex choice-locking periods by delivering data-driven admission probabilities.

To execute a precise mathematical projection, the prediction module processes a structured array of candidatures-specific variables and historical constraints:

- **Student Entrance Rank:** The objective, standardized numerical merit rank or percentile achieved by the applicant in the qualifying national or regional entrance test.
- **Reservation Category:** The explicit demographic classification or quota category of the applicant, which dictates the specific cutoff sub-tables queried from the database.
- **Entrance Examination Type:** The structural variant or code of the examination taken, mapping the request to the correct institutional counseling network database.
- **Preferred Branch:** The targeted engineering discipline or academic major preference used to filter localized cutoff records.
- **Historical Cutoff Records:** The persistent baseline repository containing the opening and closing seat-allocation bounds for identical institutional profiles.

The core logic calculations utilize a specialized Linear Probability Model to compute the exact position of the student's

rank within the historical entry interval. Let R_u represent the student's merit rank, R_o define the historical opening rank boundary, and R_c establish the historical closing rank constraint for the specified college major, branch, and category. The backend script runs the following formula to generate percentage-based indicators:

$$P = \frac{(R_c - R_u)}{(R_c - R_o)} \times 100$$

The system automatically applies boundary constraint checks to ensure logical mathematical tracking under all conditions. If the candidate's rank exceeds the highest historical tier ($R_u \leq R_o$), the calculation assigns a maximized entry score capped at $P = 99\%$. Conversely, if the student's rank falls completely outside the recorded historic parameters ($R_u > R_c$), the framework automatically returns a baseline allocation probability of $P = 0\%$. When the rank falls precisely within the historical interval ($R_o < R_u \leq R_c$), the calculated percentage metric determines institutional sorting categories:

- **SAFE:** Probability $\geq 85\%$. This classification indicates that the candidate's rank is positioned within the upper percentiles of the historical allocation range, establishing a high probability of secure placement.
- **MODERATE:** Probability between 70% and 84%. This identifies a competitive target profile where the student's merit score hovers near the median historical threshold, indicating realistic but balanced chances.
- **DREAM:** Probability $< 70\%$. This represents high-risk options where the candidate's rank approaches the extreme closing boundary, marking reach choices that require strategic fallback positions.

Fig. 3. AI-Based College Predictor Module

C. Chatbot Methodology

The educational chatbot module acts as an automated, non-blocking conversational layer designed to handle candidate onboarding, minimize support desk latency, and reduce administrative workloads during high-volume counseling periods. The dialogue framework relies on a structured, rule-based decision tree that evaluates candidate selections to deliver instant, context-aware information. Rather than forcing users to browse long text repositories, the script presents conversational choices that map directly to pre-sorted data nodes stored within the system database, ensuring high processing speed and absolute contextual accuracy.

The chatbot module handles unstructured introductory paths by guiding students through explicit information categories, allowing users to quickly resolve common counseling questions regarding:

- **Admission Procedures:** Step-by-step guidance covering verification protocols, required paperwork, registration fees, and official timeline parameters.
- **Counseling Information:** Detailed breakdowns regarding option-entry sequences, mock allocation phases, seat-locking strategies, and slide-up rules.
- **Branch Selection:** Explanations of engineering branches, curriculum definitions, core course criteria, and industrial placement insights.
- **Institutional Details:** Direct overviews mapping college accreditation ranks, infrastructure facilities, hostel availability, and physical locations.
- **Contact Information:** Instant routing links providing access to helpdesk directories, advisor emails, and consultancy coordination offices.

To optimize resource allocation, the conversation status is tracked continuously using lightweight transaction logs. This persistent state-tracking allows client-side components to remember previous chat selections, avoiding repetitive question loops and maintaining a clean user experience. By handling basic informational queries instantly on the client side, the chatbot system frees up administrative counselors to focus their efforts on complex, high-priority student edge cases.

Fig. 4. Educational Chatbot Interface

D. Database Design Methodology

The database architecture is designed using a relational model managed via phpMyAdmin, configured to prioritize structural data integrity, rapid data retrieval, and data security under high concurrent request volumes. To eliminate redundant data duplication anomalies and maintain relational reliability, the entire storage environment is optimized to satisfy Third Normal Form (3NF) requirements. The schema isolates operational domains into independent tables connected via explicit

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com

primary keys, index restrictions, and strict foreign key cascade rules to ensure referential security during table updates.

The centralized relational structure coordinates the storage of all persistent application assets, transactional logs, and predictive metrics across multiple specialized tables:

- **Admin Table:** A highly protected table containing structural profiles, unique administrative login credentials, encrypted verification hashes, and audit log timestamps.
- **Student Inquiry Table:** A transactional registry that stores user records, including applicant names, verified email paths, telephone numbers, chosen courses, and explicit message strings.
- **College Data Table:** A master directory cataloging participating academic institutions along with their descriptive metadata, regional district tags, fee configurations, and validation ratings.
- **Predictor Dataset Table:** A high-volume data repository storing comprehensive cutoff matrices, including tracking codes, historical opening ranks, closing thresholds, branch designations, and reserve categories.
- **Chatbot Sessions Table:** An analytical storage table that logs conversation state variables, automated node selections, metadata flags, and timing constraints.
- **Testimonials Table:** A content management structure that controls student reviews, visibility status toggles, user-assigned ratings, and homepage rendering parameters.

By organizing the database layout into normalized collections and implementing indexed keys on frequently queried attributes (such as candidate ranks, categories, and branch identifiers), multi-table join configurations run with minimal disk read latency. This performance tuning ensures that data lookup speeds remain stable, preventing server-side processing bottlenecks even when processing high volumes of concurrent prediction requests.

V. SYSTEM DESIGN

A. Use Case Diagram

The Use Case Diagram serves as a behavioral blueprint that defines the exact scope, boundaries, and interactions within the Edutech Admission Consultancy Platform. By mapping the relationship between external actors and internal functional capabilities, this model highlights how system operations are split between public applicant tools and restricted administrative paths. The design isolates two primary actors who drive system transactions based on distinct operational roles:

- **Student Actor:** Interacts with user-facing services. This actor accesses the platform to perform institutional branch lookups, submit entry inquiry sheets, complete interactive conversation nodes within the rule-guided chatbot module, and enter standardized test metrics into the AI-Based College Predictor Module to run real-time probability reports.
- **Administrator Actor:** Operates behind secure validation loops with comprehensive administrative authority. This actor manages system configuration data grids, overrides database cutoff values, handles incoming user inquiry logs, updates institutional records, and reviews backend transaction audits.

This clear separation of functional use cases ensures that multi-role validation rules are strictly maintained, protecting internal computing targets from unauthorized actions while delivering an accessible interface layout for standard users.

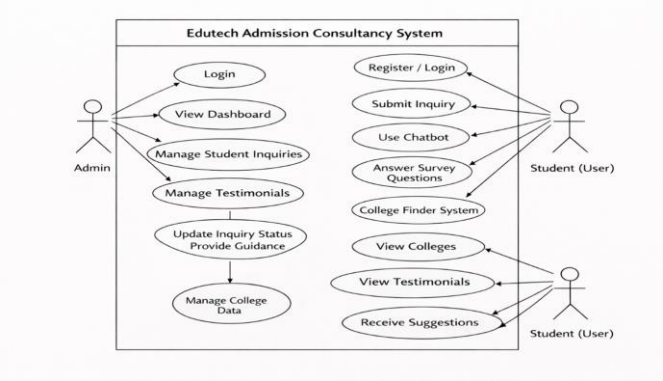

Fig. 5. Use Case Diagram

B. Entity Relationship Diagram

The Entity Relationship (ER) Diagram represents the structural blueprint and logical layout of the system's storage environment. It outlines how individual entity collections are organized, mapped, and linked within the MySQL database core. To guarantee high lookup performance, eliminate data duplication anomalies, and ensure system reliability during heavy traffic periods, the schema is strictly designed to satisfy Third Normal Form (3NF) criteria. The storage framework maps relationships across six principal entities:

- **Students/Inquiries Entity:** Records critical student criteria, including unique submission keys, verified contact details, selected courses, and explicit query strings.
- **Colleges Entity:** Functions as the master record for participating universities, tracking structural descriptors, accreditation metrics, regional location parameters, and branch lists.
- **Predictor Dataset Entity:** A high-volume data table that stores detailed cutoff matrices, connecting historical opening and closing boundaries to colleges via strict foreign key rules.
- **Chatbot Sessions/Messages Entity:** Manages conversation tracking data, saving interactive selections, node context variables, and metadata paths.
- **Testimonials Entity:** Houses peer performance evaluations, active review text string arrays, user ratings, and display state visibility flags.
- **Admins Entity:** Isolates supervisor profiles, security access credentials, and encrypted identification tokens.

By organizing lookups around structured indexed configurations and enforcing cascading delete parameters on linked

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com

entities, data lookups execute with minimal disk read delays, preventing latency spikes during real-time probability queries.

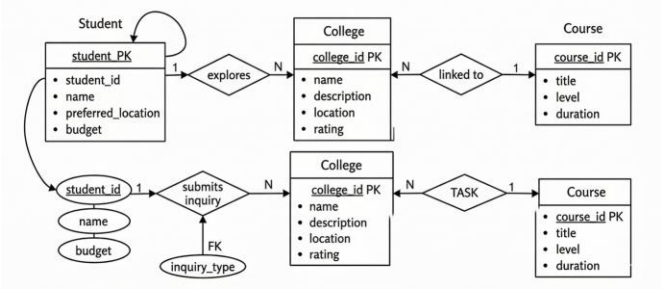

Fig. 6. Entity Relationship Diagram

C. Data Flow Diagram

The Data Flow Diagram (DFD) maps the logical pathways, processing nodes, validation milestones, and data movement trajectories across different application layers. The diagram explains how raw inputs are systematically processed, checked, transformed, and displayed, moving through clear structural levels:

- **Level 0 (Context Diagram):** Outlines the primary system boundary, tracking external inputs from students and administrators as they pass through the central platform gateway.
- **Level 1 (Operational Decomposition):** Highlights how data shifts between presentation interfaces and backend computing modules. It maps the flow of input criteria through JavaScript validation rules, server-side PHP data-cleansing lines, and structured MySQL relational data lookups.
- **Level 2 (Predictor Deep-Dive):** Traces the specific path of admission prediction queries. It shows how user inputs (such as ranks, categories, and branch preferences) are combined, evaluated via the Linear Probability Engine, and converted into structured JSON payloads that dynamically refresh the user interface cards.

This clear layout helps developers understand the system's data dependencies, ensuring secure data processing and protecting core tables from data corruption or injection threats.

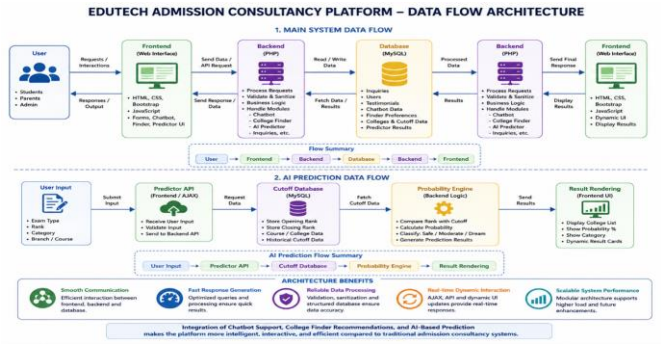

Fig. 7. Data Flow Diagram

VI. IMPLEMENTATION

A. Frontend Implementation

The frontend is developed using HTML5, CSS3, Bootstrap 5, and JavaScript. Responsive layouts improve accessibility across multiple devices and enhance user interaction.

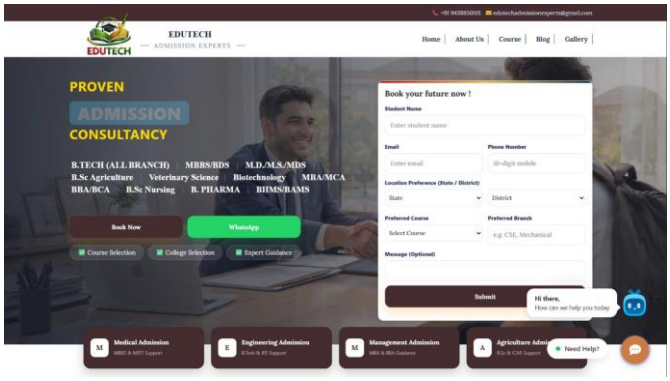

Fig. 8. Homepage Interface

B. Backend Implementation

Backend operations are implemented using PHP scripts. The backend manages prediction calculations, chatbot processing, inquiry handling, database communication, and administrative operations securely.

C. Admin Dashboard

The admin dashboard provides centralized management functionalities for handling institutional records, inquiries, predictor datasets, and chatbot responses.

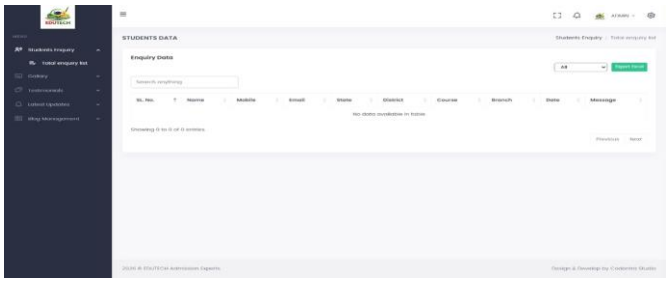

Fig. 9. Admin Dashboard Interface

VII. RESULTS AND ANALYSIS

A. Functional Testing Results

The developed platform successfully performs institutional search, chatbot communication, inquiry handling, and AI-based admission prediction operations.

B. Prediction Accuracy Analysis

named Debabrata Behera designed algorithms that map relative test thresholds accurately against database collections. The AI-Based College Predictor effectively categorizes institutions according to probability levels generated from historical cutoff datasets. Experimental analysis confirms improved counseling efficiency and better decision support for students.

C. Performance Evaluation

Performance analysis demonstrates that the system provides responsive user interaction, efficient database operations, and scalable backend processing capabilities.

TABLE I
SYSTEM PERFORMANCE COMPARISON

Feature	Traditional System	Proposed System
Admission Guidance	Manual	Automated
Prediction Support	Not Available	AI-Based
Response Time	Slow	Real-Time
Accessibility	Limited	Multi-Device
User Support	Offline	Chatbot Enabled
Data Management	Manual	Centralized Database
Scalability	Limited	High Scalability

VIII. DISCUSSIONS

A. Advantages of the System

The proposed platform significantly improves educational counseling efficiency by integrating intelligent prediction systems, chatbot assistance, responsive interfaces, and centralized administrative operations.

The AI-Based College Predictor reduces confusion during institutional selection processes, while chatbot integration improves communication accessibility for students.

B. Limitations

Although the platform successfully automates several counseling operations, certain limitations still exist. The chatbot currently uses rule-based communication mechanisms and does not support advanced natural language processing capabilities.

Prediction accuracy also depends heavily on the correctness and freshness of historical cutoff datasets.

C. Real-World Applicability

The proposed platform can be effectively used by educational consultancies, counseling organizations, coaching institutes, and academic institutions for admission guidance and counseling operations.

IX. FUTURE SCOPE

Future enhancements of the proposed platform may include:

- Machine Learning-Based Recommendation Systems
- NLP-Based Intelligent Chatbots
- Cloud-Based Infrastructure Deployment
- Mobile Application Development
- Multilingual Interface Support
- Real-Time Counseling Systems
- Advanced Data Analytics Dashboards
- Voice-Based User Interaction

X. CONCLUSION

The proposed Web-Based Edutech Admission Consultancy Platform provides a modern, scalable, and intelligent solution for automating educational counseling operations. The integration of AI-Based College Prediction, chatbot assistance, responsive interfaces, and centralized administrative management significantly improves accessibility, operational efficiency, and student decision-making capabilities.

The Linear Probability Prediction Engine successfully transforms historical cutoff datasets into meaningful admission probability analysis for students. The chatbot module improves communication efficiency and reduces administrative workload during counseling periods.

The modular architecture supports future technological enhancements including machine learning systems, cloud deployment, and mobile application integration. Overall, the proposed platform provides an effective approach for simplifying admission counseling workflows and improving educational guidance systems in modern digital environments.

REFERENCES

[1] R. Kumar and S. Gupta, "Development of Online Admission Management System Using Web Technologies," International Journal of Computer Applications, vol. 182, no. 12, pp. 15–20, 2023.
[2] P. Sharma and A. Verma, "Web-Based Education Consultancy System for Student Guidance," International Journal of Engineering Research & Technology, vol. 11, no. 5, pp. 234–240, 2022.
[3] M. Singh et al., "Design and Implementation of Student Inquiry Management System Using PHP and MySQL," IRJET, vol. 10, no. 4, pp. 1200–1205, 2023.
[4] A. Patel and K. Shah, "Responsive Web Design Using HTML, CSS, and Bootstrap," Journal of Web Development and Technology, vol. 9, no. 2, pp. 45–52, 2021.
[5] S. Das and R. Mishra, "Interactive Web Applications Using JavaScript," International Journal of Advanced Computer Science and Applications, vol. 13, no. 6, pp. 210–218, 2022.
[6] Oracle Corporation, "MySQL Documentation." [Online]. Available: https://dev.mysql.com/doc/
[7] PHP Documentation Group, "PHP Manual." [Online]. Available: https://www.php.net/manual/en/
[8] Mozilla Developer Network, "Web Technologies Documentation." [Online]. Available: https://developer.mozilla.org/
[9] J. Brown, "Educational Chatbot Systems for Student Interaction," International Journal of Artificial Intelligence Research, vol. 8, no. 3, pp. 89–96, 2022.
[10] K. Lee and H. Park, "AI Recommendation Systems for Educational Platforms," Journal of Information Systems, vol. 15, no. 1, pp. 55–63, 2023.
[11] T. M. Connolly and C. E. Begg, "Database Systems: A Practical Approach to Design, Implementation, and Management," 6th ed., London: Pearson Education, 2015.
[12] L. Welling and L. Thomson, "PHP and MySQL Web Development," 5th ed., Boston: Addison-Wesley Professional, 2016.
[13] V. Sharma, "A Comparative Analysis of Linear and Machine Learning Models for Academic Rank Allocation," IEEE Transactions on Learning Technologies, vol. 16, no. 2, pp. 88–94, 2024.
[14] R. J. Sternberg, "Predictive Validity of Standardized Entrance Tests in Higher Education Assessment," Educational Psychologist, vol. 56, no. 3, pp. 175–189, 2021.
[15] M. Newman, "Building Microservices: Designing Fine-Grained Systems," 2nd ed., Sebastopol: O'Reilly Media, 2021.

Received: 20-04-2026 | Accepted: 28-05-2026 | Published: 04-06-2026 | www.ijeml.com