IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

SOET Website with Chatbot: An Institutional Web Portal using CAMPUSAI

Nitin Pawar, Gopika Hawale, Rutvik More, Vishal Gavhad, Sarika Rathi, Dr. Madhuri Kawarkhe
Graduate Students, Department of Computer Science and Engineering
Project Guide / Assistant Professor, Department of Computer Science and Engineering
Assistant Professor, Department of Computer Science and Engineering
School of Engineering and Technology (SOET), Chh. Sambhajinagar, Maharashtra, India
MGM University, Chh. Sambhajinagar, Maharashtra, India

Abstract: *Higher education institutions increasingly require digital platforms that provide accurate academic information while offering quick and convenient assistance to students, applicants, faculty, and visitors. However, many college websites rely on static content, limited information services, and separate communication channels, making it difficult for users to find relevant information efficiently. The absence of an integrated and intelligent support system can create delays in accessing information related to courses, admissions, faculty, events, placements, notices, and other institutional activities. The main objective of this study is to develop a dynamic SOET Website with Chatbot that provides centralized college information and interactive assistance through a single web-based platform. The proposed system is developed using HTML, CSS, JavaScript, PHP, MySQL, and XAMPP. The website includes modules for departments, department-wise courses, faculty, admissions, placements, events, gallery, blogs, news, announcements, contact services, visitor tracking, and database backup and restore. Separate registration forms are provided for admission and events, while blog and news submissions follow a verification workflow in which content is reviewed by an assigned administrator before publication. A role-based administrative panel enables authorized users to manage website content and operational activities. The integrated chatbot is designed to respond to user queries using relevant website information and provide conversational assistance. The implemented system demonstrates improved organization of institutional content, centralized administration, controlled content publication, and easier access to college-related information. The proposed platform can support digital transformation in educational institutions by combining dynamic content management with an interactive chatbot in a single accessible environment.*

Keywords: Web Portal, AI Chatbot, Higher Education, Content Management System, Automated Inquiry Resolution, Retrieval-Augmented Generation (RAG).

I. INTRODUCTION

Institutional websites of engineering colleges typically host static content describing academic programs, faculty profiles, admission procedures, examination schedules, and placement records. While comprehensive, such static content places the burden of navigation and search entirely on the visitor. Prospective students, parents, and current students frequently pose the same categories of questions - fee structure, eligibility criteria, hostel facilities, timetable changes, and result declarations - through phone calls or emails to administrative staff, creating repetitive workload and delayed responses outside office hours.

Conversational AI, powered by advances in large language models (LLMs), offers a way to convert static institutional websites into interactive, always-available information sources. This paper proposes CampusAI, a chatbot widget embedded in the School of Engineering and Technology (SOET) website, designed to answer student and visitor

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

queries in real time and in the language of the user's choice. Unlike a generic web search engine, the proposed system is grounded exclusively in verified institutional data, reducing the risk of incorrect or fabricated answers.

The remainder of this paper is organized as follows: Section 2 reviews related work in institutional and educational chatbots. Section 3 describes the proposed system architecture and methodology. Section 4 discusses the methodology and system design. Section 5 presents the results and discussion of the implemented system. Section 6 outlines the proposed evaluation framework. Section 7 presents the future scope, and Section 8 concludes the paper.

2. Related Work and Literature Review

User interfaces for automated student inquiry handling have been investigated in [3] using basic rule-based chatbot models. The authors of [4] discussed the significance of natural language processing (NLP) and key intent classification algorithms in improving chat response accuracy. Researchers in [5] suggested using web-based administrative dashboards for centralizing campus event schedules and faculty directory management. The authors of

[6] discussed the role of modular database design in academic management systems and employed various backend frameworks for data retrieval. An automated chatbot pipeline integrated with institutional databases for handling real-time admission queries was implemented in [7]. The researchers in [8] used modern web development tools and Machine Learning (ML) techniques to investigate automated query resolution across various academic modules. The authors of [9] proposed a paradigm for enhancing digital student services using hybrid conversational AI architectures. The retrieval-augmented generation approach was used as a base model in [10] alongside several search algorithms to manage complex academic knowledge bases. For the categorization and handling of multifaceted student inquiries, the authors of [11] presented an integrated web framework that incorporates different functional modules, such as Admissions, Course Directories, Faculty Portals, and AI Chatbot assistants.

2. LITERATURE SURVEY & COMPARATIVE ANALYSIS

2.1 Traditional University Portals vs. Proposed Portal

Standard academic web solutions rely heavily on monolithic frameworks or third-party plugin ecosystems. Table 2.1.1 provides a comparative overview highlighting the functional and structural differences between traditional institutional web platforms and the proposed School of Engineering and Technology (SOET) academic portal.

Table 2.1.1: Comparative Analysis of Institutional Web Systems

Feature / Metric	Standard University Portals	Proposed SOET Academic Portal
Framework Base	Built on generic, heavy CMS templates (e.g., WordPress, Drupal)	Developed on lightweight, native PHP 8.x (OOP) architecture
Search Functionality	Inflexible, static list matching and keyword search	Integrated 24/7 AI Academic Advisor with dynamic Context Retrieval
System Resilience	High failure rate when third-party services or APIs go offline	Automatic Local-to-Cloud smart fallback logic to guarantee uptime
Data Tracking	Relies on scattered Google Sheets and paper-based tracking	Native relational MySQL database tracking for admission requests
Database Security	Vulnerable to SQL Injection via legacy or unpolished dynamic queries	Enforces PDO secure prepared statements across all system modules

2.2 Security and Architectural Gaps

Legacy higher education web systems frequently run unvalidated raw SQL queries, creating critical security risks such as SQL Injection vulnerabilities. The proposed SOET portal architecture mitigates these vectors by implementing PHP Data Objects (PDO) with prepared statements and strict parameter binding across all database interactions. Furthermore, administrative operational workflows replace paper-based tracking methods with native database state management, using a centralized dashboard with Role-Based Access Control (RBAC) modules to isolate and protect student-facing data streams.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

2.3 Research Gap

Existing college and university websites generally provide basic information such as courses, departments, faculty, admissions, events, notices, placements, and contact details. However, many such platforms do not provide a single, well-organized and fully dynamic system for managing all these services together. Information may be distributed across different pages, making it difficult for students and visitors to find and access relevant information quickly.

Another gap is the limited availability of centralized content management. In many existing systems, updating faculty information, courses, events, blogs, news, notices, and other institutional content may require technical intervention. There is also a need for separate registration processes for admissions and events and better management of submitted information.

Furthermore, content such as blogs and news requires controlled verification before publication. A structured workflow in which submitted content is reviewed and approved by an assigned administrator can improve the management and reliability of published institutional information.

Therefore, the research gap identified in this work is the need for a centralized, dynamic, user-friendly, and role-based college website that integrates academic information, admission and event registration, faculty and course management, placements, gallery, blogs, news, announcements, visitor tracking, content verification, and database backup within one platform. The proposed SOET Website addresses these gaps by bringing the required institutional services together into a single web-based system, with an integrated conversational AI chatbot.

III. PROPOSED SYSTEM ARCHITECTURE AND METHODOLOGY

The proposed system architecture is designed as a Retrieval-Augmented Generation (RAG) framework tailored to deliver dynamic, multilingual institutional information to end users. The system decouples real-time user query processing from administrative knowledge management, enabling non-technical SOET staff to update facts independently of application codebase deployments. The architecture is composed of four principal layers: (i) the presentation layer (chat widget UI), (ii) the language and intent-processing layer, (iii) the retrieval and grounding layer, and (iv) the LLM response-generation layer.

3.1 Key Architectural Subsystems

The platform architecture coordinates user interactions, content publication, administrative oversight, and data persistence through dedicated, decoupled subsystems.

• 1. User Interaction Layer (Public Actors): The system caters to four primary actor roles: Students, General Visitors, Faculty, and Heads of Departments (HODs). Public actors interact with the front-end interface to consume institutional information, navigate departmental profiles, register for academic courses, and engage with the integrated AI chatbot.

• 2. Core Application & Public Modules: The centralized web portal encapsulates diverse functional modules including: Informational Interfaces (Home, About Us, Departments, Faculty Profiles, Gallery, and Course Outlines); Dynamic Event & Media Subsystems (real-time updates for Notices, Announcements, News, and Campus Events); and an Interactive AI Chatbot Module (automated contextual assistant providing immediate responses to user queries).

• 3. Administrative Management Subsystem: The administrative control unit enables SOET staff and administrators to manage institutional parameters independently of the underlying codebase. Key administrative functionalities include full CRUD (Create, Read, Update, Delete) operations over departments, faculty, and academic programs; Role-based Access Control (RBAC) to govern user privileges; and database backup, restoration, and visitor log auditing.

• 4. Content Verification & Approval Workflow: To prevent unauthorized or inaccurate public postings, the system implements a managed submission pipeline: (a) Submission: Public users submit content (e.g., student blog posts, news stories, event registrations) via dedicated web forms; (b) Queueing: Submissions are stored in an unverified state within the database; (c) Verification: Assigned administrators inspect the submissions and execute approval, rejection, or edit actions; (d) Publishing: Approved content is updated in status and dynamically rendered on the public portal.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

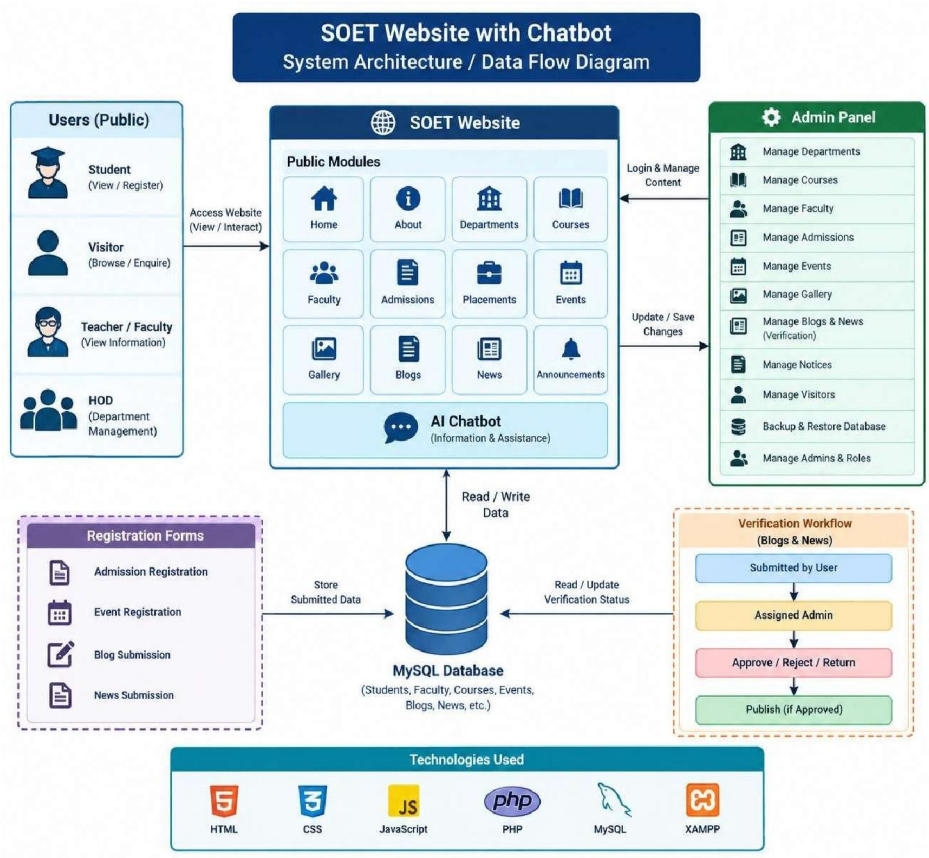


Figure 3.1.1: SOET Website with Chatbot System Architecture and Data Flow Diagram

3.2 Data Persistence Layer and CampusAI Pipeline

A centralized MySQL Database acts as the persistence engine, housing structural entities including user accounts, faculty directories, transactional registration forms, event records, and moderation queues.

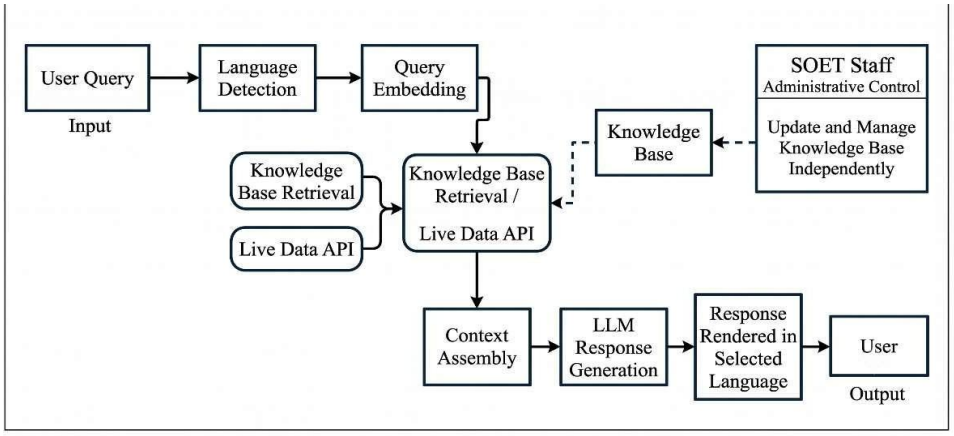


Figure 3.1.3: CampusAI Conversational Workflow and Grounding Pipeline Architecture

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

3.2.1 Sequential Process Flow

1. Front-End Ingestion & Query Vectorization: The user submits a raw natural language text prompt via the web interface. The system automatically determines the source language to configure language consistency across downstream generation stages. The input string is converted into a vector representation capturing semantic intent.
2. Dual Retrieval Mechanism: To prevent information staleness and support both static institutional context and live operational data, the architecture employs a hybrid retrieval model: (a) Knowledge Base Retrieval: Executes semantic similarity searches against stored institutional documents, course outlines, and academic regulations; (b) Live Data API: Performs dynamic API lookups to fetch real-time operational data (e.g., active event updates, notices, admission statuses).
3. Administrative Knowledge Orchestration: SOET staff access administrative controls to modify, upload, or prune stored knowledge base records independently without requiring codebase redeployment or server restarts, guaranteeing high availability and accurate factual baselines.
4. Context Synthesis & Multilingual Response Rendering: Relevant retrieved context chunks from both the static Knowledge Base and Live Data APIs are merged and injected into an enriched system prompt. The Large Language Model (LLM) generates an accurate, hallucination-free answer grounded strictly in the assembled context payload. The output is rendered into the user's primary detected language and delivered back to the client interface.

3.3 Data Sources and Preprocessing

The knowledge base is constructed from the SOET website's own content - admissions criteria, course curricula, faculty directories, fee structures, examination notices, and placement statistics - supplemented by administrative documents such as circulars and academic calendars. Documents are cleaned, segmented into passages of bounded length, and converted into vector representations for semantic search. Structured data such as fee tables and timetables are additionally exposed through direct API endpoints so that the chatbot can return exact, current values rather than approximations drawn from static text.

3.4 Multilingual Query Handling

User queries in English, Hindi, or Marathi are first normalized to a working language for retrieval, matched against the knowledge base, and the resulting answer is generated directly in the user's selected language rather than machine-translating an English answer, in order to preserve accuracy for numerical and policy-specific content.

3.5 Retrieval-Augmented Response Generation

When a query is received, the system retrieves the top-k most relevant passages from the knowledge base and supplies them as context to the LLM along with the user's question. The model is instructed to answer strictly from the retrieved context, and to state clearly when information is unavailable rather than generating an unsupported answer. For time-sensitive structured data (e.g., current fee deadlines, live notices), the system additionally invokes function calls to the institution's live backend rather than relying on the static knowledge base, ensuring the response reflects the current state of the data.

IV. METHODOLOGY AND SYSTEM DESIGN

The methodology of the proposed SOET Website with Chatbot focuses on the systematic analysis, design, development, implementation, and testing of a dynamic college website. The system is developed using HTML, CSS, JavaScript, PHP, MySQL, and XAMPP. The methodology emphasizes centralized information management, user-friendly access, role-based administration, content verification, and reliable database management.

4.1 Requirement Analysis and System Design

The first stage involves identifying the requirements of students, visitors, teachers, HODs, and administrators. The required website modules are analyzed and organized into public and administrative sections. The public section

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

contains information about departments, courses, faculty, admissions, placements, events, gallery, blogs, news, announcements, and contact services. The administrative section provides facilities for managing website content, admissions, events, faculty, courses, departments, fees, gallery, placements, blogs, news, notices, visitors, and database backups. The system architecture and database structure are designed before implementation. MySQL tables are planned according to the different modules, with appropriate relationships between departments, courses, faculty, admissions, events, blogs, news, and administrative records.

4.2 System Development and Implementation

The frontend is developed using HTML, CSS, and JavaScript to create a responsive and interactive interface. PHP is used for server-side processing, while MySQL stores and manages the system data. XAMPP provides the local Apache and MySQL environment for development and testing. The website is implemented as a dynamic system in which authorized administrators can add, update, and manage institutional information. Separate forms are provided for Admission Registration, Event Registration, Blog Submission, and News Submission. The Blog and News modules follow a controlled verification process. Submitted content is first sent to the assigned administrator for verification. The administrator can approve, reject, or return the content for correction. Only approved content can be published on the website.

4.3 Testing and Validation

After implementation, each module is tested individually and then integrated with the complete system. Functional testing is performed for website navigation, forms, authentication, database operations, CRUD operations, admission processing, event registration, faculty and course management, blog and news verification, notice management, visitor tracking, chatbot interaction, and database backup and restore. Input validation and security checks are also performed to identify incorrect inputs and unauthorized access. The website is tested on different screen sizes to verify responsive behavior. Errors identified during testing are corrected and the affected modules are tested again to ensure that the final system operates correctly and consistently.

V. RESULTS AND DISCUSSION

The developed SOET Website with Chatbot was implemented as a dynamic web-based platform using HTML, CSS, JavaScript, PHP, MySQL, and XAMPP. The system integrates different college-related modules into a centralized platform and provides separate facilities for users and administrators. The implemented system was tested to verify functionality, data management, content verification, and overall usability.

5.1 System Implementation

• Dynamic Web Application: The SOET website was successfully developed as a dynamic web application integrating major administrative modules using PHP and MySQL.
• Centralized College Information: The system provides unified access to academic programs, faculty credentials, admissions, and campus events.

5.2 Module Performance

• Functional Modules: Department, course, faculty, admission, event, placement, gallery, notice, and contact modules were implemented and validated.
• Dedicated Registration Forms: Separate registration pipelines were provided for student admissions and campus event participation.
• Managed Publication Workflow: Blog and News modules support submission, administrative review, correction return, and verified publication.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

5.3 Admin Management & Content Verification

• Role-Based Administration: Multiple administrator roles can be managed according to assigned responsibilities, enabling department-wise administration.

• Database Safeguards: Database backup and restore functionality was integrated to preserve institutional records and operational continuity.

• Verification Pipeline: Blog and News submissions must be approved by the designated administrator before appearing on the public portal.

5.4 Testing Results and Discussion

Website navigation, forms, CRUD operations, authentication, database transactions, and registration processes were comprehensively tested. Responsive behavior was verified across desktop, tablet, and mobile displays. To evaluate the operational readiness and structural completeness of the platform, a modular deployment analysis was conducted across all core units of the SOET ecosystem.

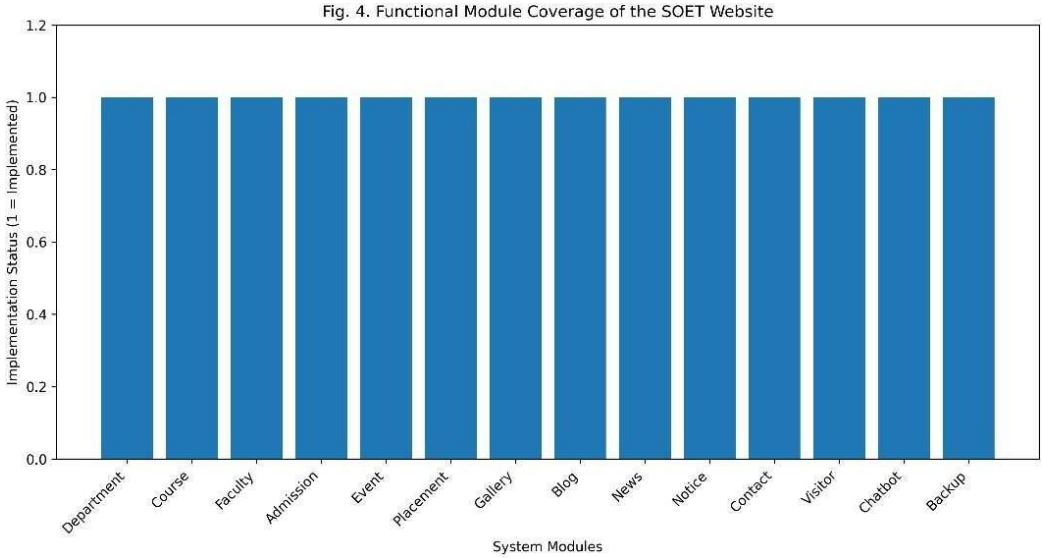


Figure 5.6: Functional Module Coverage of the SOET Website Platform

5.5 Evaluation Breakdown

• 1. Core Academic Subsystems: Department & Course Management provides complete structural mapping of academic programs, curriculum syllabi, and departmental hierarchies. The Faculty Directory dynamically renders individual faculty profiles, research areas, and contact details. Admissions & Placements facilitates student intake forms and dynamic placement drive reporting.

• 2. Communication & Engagement Subsystems: Event, Blog, & News Modules integrate public event announcements alongside managed blog/news publishing with strict administrative verification. Notices & Contact Portals deliver real-time notice board updates and automated enquiry message handling. The Gallery Module hosts campus visual media archives.

• 3. Intelligent & Administrative Subsystems: The AI Chatbot Module acts as an embedded contextual assistant serving real-time automated queries to public visitors. The Visitor Tracking & Backup Engine maintains visitor analytics logs and administrative database backup/restore workflows.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

VI. PROPOSED EVALUATION FRAMEWORK

As this paper presents a comprehensive system design and implementation platform rather than an isolated conversational benchmark, we propose the following empirical evaluation framework to be applied during the live deployment of CampusAI on the SOET portal, ensuring its effectiveness can be measured consistently with academic conversational agent literature.

Table 6.1: Proposed Evaluation Metrics for CampusAI

Metric	Description
Answer Accuracy	Proportion of test queries answered correctly against a curated ground-truth question set.
Grounding Rate	Proportion of answers directly supported by retrieved knowledge-base passages (i.e., not hallucinated).
Response Latency	End-to-end time from query submission to displayed response, measured in seconds.
Language Accuracy	Correctness of responses generated in Hindi and Marathi, evaluated by native speakers.
User Satisfaction	Post-interaction survey score (e.g., Likert scale) collected from students and staff.
Fallback Rate	Proportion of queries for which the system correctly declines to answer rather than guessing.

VII. FUTURE SCOPE

• Mobile Application: Develop a dedicated native SOET mobile application for Android and iOS.
• Online Admission Processing: Extend admission services to support complete online verification, document processing, and digital fee collection.
• Cloud-Based Redundancy: Provide secure, automated cloud database backup and synchronized failover.
• Push Notifications: Integrate automated SMS and push notifications for urgent admission deadlines, exam timetables, and campus event notices.
• Broader Multilingual Support: Expand the conversational pipeline to support additional regional languages.
• Granular Access Control: Implement more fine-grained administrative permissions and audit trails for departmental coordinators.
• Comprehensive Student Services: Incorporate digital transcript requests, grade card downloads, and integrated alumni networking services.

VIII. CONCLUSION

The SOET Website with Chatbot was developed as a centralized and dynamic platform for managing and accessing college-related information. The system brings together departments, courses, faculty, admissions, placements, events, gallery, blogs, news, announcements, contact services, and visitor tracking in one unified platform. The use of HTML, CSS, JavaScript, PHP, MySQL, and XAMPP provides a structured and efficient environment for developing and maintaining the web ecosystem. Separate admission and event registration forms simplify user information collection, while the Blog and News verification workflow enforces controlled, administrator-supervised publication. The admin panel supports multi-role administration and department-wise management. Database backup and restore functionality provides a dependable mechanism to safeguard stored records. Overall, the implemented system establishes organized institutional information management and delivers a convenient, interactive experience for students, prospective applicants, faculty, and administrators.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the guidance of Mrs. Sarika Rathi and the support provided by the Department of Computer Science & Engineering, School of Engineering & Technology (SOET), MGM University, Chhatrapati Sambhajinagar.

Funding

No external funding was received for the development of this project.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 7, October 2026
Impact Factor: 8.2

Author Contributions

• Nitin Pawar: Conceptualization, System Architecture Design, Project Administration, and Final Manuscript Review.
• Gopika Hawale: Methodology, Dataset Curation/Preprocessing, Algorithm Design, and Primary Manuscript Drafting.
• Rutvik More: Software Implementation, Model Integration, System Optimization, and Interface Development.
• Vishal Gavhad: System Validation, Performance Testing, Data Analysis, and Result Visualization.
• Sarika Rathi: Project Mentorship, Structural Supervision, Technical Review, and Quality Assurance.
• Dr. Madhuri Kawarkhe: Drafting of Research Paper, Research Evaluation, and Final Manuscript Editing.

Conflict of Interest

The authors declare that there are no conflicts of interest.

REFERENCES

[1] School of Engineering and Technology, MGM University, "SOET - School of Engineering and Technology," MGM University.
[2] S. R. Rathi, Project Guidance and Supervision, Dept. of Comput. Eng., School of Engineering and Technology (SOET), 2026.
[3] E. Adamopoulou and L. Moussiades, "Chatbots: History, technology, and applications," Machine Learning with Applications, vol. 2, p. 100006, 2020.
[4] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020, pp. 9459-9474.
[5] R. Mirshahvalad and N. A. Zanjani, "Diabetes prediction using ensemble perceptron algorithm," in Proc. 9th Int. Conf. Comput. Intell. Commun. Netw. (CICN), 2017, pp. 112-115.
[6] D. Dutta, D. Paul, and P. Ghosh, "Analysing feature importances for diabetes prediction using machine learning," in Proc. IEEE 9th Annu. Inf. Technol., Electron. Mobile Commun. Conf. (IEMCON), 2018, pp. 924-928.
[7] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), Long Beach, CA, USA, 2017, pp. 5998-6008.
[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. Conf. N. Amer. Chapter Assoc. Comput. Linguistics: Hum. Lang. Technol. (NAACL-HLT), Minneapolis, MN, USA, 2019, pp. 4171-4186.
[9] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," Journal of Machine Learning Research, vol. 3, pp. 1137-1155, Mar. 2003.
[10] R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, Dept. Inf. Comput. Sci., Univ. California, Irvine, Irvine, CA, USA, 2000.
[11] E. Gamma, R. Helm, R. Johnson, and J. Vlissides, Design Patterns: Elements of Reusable Object-Oriented Software. Reading, MA, USA: Addison-Wesley, 1994.