URCET Campus Chatbot

An AI-Powered Conversational Assistant for Usha Rama College of Engineering and Technology

1. Abstract

The URCET Campus Chatbot is an AI-powered web assistant designed for Usha Rama College of Engineering and Technology, providing a single, natural language interface to access campus information. It consolidates fragmented data on academic programs, faculty, admissions, fees, exams, and policies, reducing repetitive queries for staff and streamlining information access for students, parents, and applicants. The hybrid response system intelligently combines a curated knowledge base for routine queries with Google Gemini 2.5 Flash AI for complex or contextual questions. Built with React 19, TypeScript, and Vite, and integrated via the Google Generative AI SDK, the platform ensures a responsive, accessible, and performant experience across desktop and mobile devices without complex backend infrastructure.

2. Keywords

Campus Chatbot, Generative AI, Gemini 2.5 Flash, React 19, TypeScript, Vite, Conversational Interface, URCET, Campus Information System, Hybrid Response Architecture, Google Generative AI SDK, Natural Language Processing, Smart Query Processing, Context-Aware Conversation, Responsive Design.

3. Synopsis

The rapid digitisation of higher education has raised expectations for immediate, accurate, and 24/7 access to institutional information, which traditional channels cannot meet. Campus data scattered across portals, notice boards, and printed documents burdens users and administrative staff with repetitive queries. The URCET Campus Chatbot addresses this with a unified AI-powered interface, trained on a comprehensive knowledge base covering all aspects of campus life at Usha Rama College of Engineering and Technology. Its hybrid response system directs routine, high-frequency queries to a curated knowledge base for guaranteed accuracy, while more complex or contextual queries are escalated to Google Gemini 2.5 Flash via the Generative AI SDK. A fallback mechanism ensures uninterrupted service even if the AI model is unavailable. Quick-action buttons provide fast access to departments, faculty contacts, exams, facilities, and policies. Built with React 19 and TypeScript, the responsive interface delivers a consistent experience across desktop, tablet, and mobile devices, ensuring equitable access for all users.

4. Technologies Used

Frontend - React 19 with TypeScript – Modular, maintainable interface with type safety to prevent runtime errors.

Build Tool: Vite – Fast development server and optimized production bundles for quick loading.

AI Model: Google Gemini 2.5 Flash – Generates real-time, contextual responses to complex queries.

Hybrid Response Engine: Curated knowledge base + AI fallback – Answers common questions instantly and escalates unusual queries to AI.

Query Processing: Smart module – Understands different phrasings and correctly routes user questions.

Interface Design: Mobile-first responsive layout with quick-access buttons for frequently requested info.

5. Procedure

Step 1: Access and Setup:

Users open the chatbot on any device with no login needed. The interface loads, quick-access buttons appear, and the system connects to the AI backend.

Step 2: Submit Query:

Users type questions naturally. The system interprets intent and standardizes phrasing so different ways of asking the same thing are understood correctly.

Step 3: Answer Retrieval:

The system first checks the curated knowledge base for a matching answer. If none is found, it sends the query to Google Gemini AI for a real-time response. A fallback ensures continuity if the AI is unavailable.

Step 4: Context-Aware Conversation:

The chatbot remembers previous messages, allowing follow-up questions to be understood correctly in context. Multi-turn interactions stay coherent.

Step 5: Display Response:

Answers appear in the chat interface with timestamps and formatting. The conversation thread preserves history, keeping the session continuous and easy to review.

6. Why It Is Best

Hybrid Knowledge Architecture: The URCET Campus Chatbot integrates a curated knowledge base for high-frequency, high-stakes queries with the Gemini 2.5 Flash AI model for complex, context-dependent questions, ensuring both factual accuracy and generative flexibility.

Reliable and Continuous Service: The fallback mechanism guarantees uninterrupted access even if the AI service is temporarily unavailable, while the knowledge base provides instant, latency-free responses to predictable queries, maintaining trustworthiness for critical campus information like fees and deadlines.

Intuitive, Accessible Interface: Built with React 19 and TypeScript, the frontend delivers a responsive, mobile-friendly experience with quick-action buttons and smart query processing, allowing users to interact naturally without needing precise phrasing or prior knowledge of the system.

Efficient, Secure, Low-Overhead Deployment: By avoiding complex backend infrastructure, the chatbot reduces operational cost, security exposure, and maintenance overhead, while still enabling rapid updates to institutional content and ensuring a scalable, maintainable solution for campus-wide information delivery.

.

7. Conclusion

The URCET Campus Chatbot demonstrates that institution-specific AI assistants can be efficiently developed and deployed without complex backend infrastructure, providing a reliable, production-quality conversational interface for students, faculty, parents, and prospective applicants. By combining a curated knowledge base with Gemini 2.5 Flash AI and a robust fallback mechanism, the system balances factual accuracy with generative flexibility. Smart query processing, context-aware multi-turn conversation management, quick-action navigation, and a responsive device-agnostic interface deliver a superior user experience while reducing administrative burden and improving information accessibility. The current implementation establishes a foundation for next-generation enhancements, including real-time database integration, regional language support (e.g., Telugu), voice interaction, and an administrative analytics dashboard. Multi-campus deployment capabilities and live-synchronised knowledge updates will expand scalability, accessibility, and operational efficiency. Collectively, these improvements position the platform as a scalable, data-driven, linguistically inclusive institutional intelligence solution suitable for modern higher education administration.