Smart Complaint Registration Chatbot CampusCare Bot

Rutuja Nandulal Mistry

Prof. Ramkrishna More Arts, Commerce & Science College ,Pradhikaran, Pune, Akurdi, Pune - 411044(Maharashtra)India.

E-mail : rutujamistry8@gmail.com

Dr. Kalyan C. Jagdale

Prof. Ramkrishna More Arts, Commerce & Science College ,Pradhikaran, Pune, Akurdi, Pune - 411044(Maharashtra)India.

Kalyan.jagdale7@gmail.com

Abstract

This paper presents the design, development, and thorough validation of CampusCare Bot, a smart complaint-registration chatbot created to streamline and improve how issues are resolved across a university campus. Powered by Natural Language Processing (NLP), the system can instantly interpret unstructured text-based complaints and categorize them—such as Maintenance, IT, or Safety—with an accuracy of 92%.The bot is built using a scalable Python/Flask microservices architecture, guided by design automation principles to maintain structural consistency and speed up deployment. To evaluate the system's performance, the research includes both quantitative and qualitative analyses. A Prophet time-series model is used to forecast user activity and peaks in complaint volume, helping administrators manage resources proactively. Meanwhile, insights from user interviews show that the bot improves transparency and trust, thanks to its clean, unified design and real-time ticket-tracking features. Overall, CampusCare Bot stands as a strong example of how conversational AI can be integrated into essential administrative processes, setting a new benchmark for operational efficiency and student support in smart-campus initiatives.

Introduction

The smooth functioning and overall agility of a modern university campus depend heavily on how effectively it can manage and resolve internal issues. Traditional complaint-handling methods—such as paper-based forms, static email inboxes, or visits to administrative offices—often fall short of meeting today's expectations for responsiveness and transparency. These outdated systems commonly suffer from slow processing times, inconsistent or inaccurate categorization, and limited visibility into complaint status, ultimately resulting in delayed resolutions and reduced trust among students and staff To address these longstanding challenges, this research introduces and validates CampusCare Bot, a Smart Complaint Registration Chatbot designed to modernize the entire issue-reporting workflow. By integrating Conversational Artificial Intelligence (AI) and Natural Language Processing (NLP), the CampusCare Bot automates the submission, interpretation, classification, and routing of campus-related complaints. This automation minimizes human error, accelerates triage, and provides users with real-time updates, creating a more transparent and efficient system for campus service management

Background: The Necessity of Efficient Administrative Services

In modern universities, administrative services have evolved from simple support units into essential pillars that shape the overall student experience and influence an institution's reputation. When these services are efficient, accessible, and responsive, they contribute directly to student satisfaction, retention, campus safety, and even long-term operational sustainability However, many campuses continue to rely on legacy, manually driven systems that were created decades ago and no longer align with the expectations of today's digitally native students. These outdated processes often create unnecessary friction slow service delivery, fragmented communication, and limited transparency ultimately hindering a university's ability to operate proactively and address issues before they escalate As higher education institutions increasingly embrace digital transformation, upgrading administrative services has become more than just an improvement opportunity; it is a strategic necessity. Modern digital platforms offer the speed, accessibility, and user-centric design required to support contemporary campus operations, ensuring that administrative functions remain efficient, scalable, and aligned with the evolving needs of students and staff .

1.1 Problem Statement: Limitations of Legacy Systems

Despite their longstanding use, traditional complaint-registration methods continue to create several operational challenges and negatively impact campus service quality. Three major limitations stand out:

1. **High Latency in Processing:** Legacy systems rely heavily on manual review, requiring staff to read, interpret, and forward each complaint. This dependency on human intervention creates unavoidable bottlenecks, often delaying issue assignment by several hours or even days, in non-urgent cases. Such delays directly increase the overall time required to resolve issues.
2. **InconsistentCategorization:** Because classification is based on individual judgment, complaints are frequently misinterpreted or assigned to the wrong department. For example, a "slow network" issue may mistakenly be sent to Maintenance instead of IT Support. These misroutes require corrective actions, further slowing the process and straining departmental resources.
3. **Lack of Transparency for Users:** Traditional systems rarely provide instant confirmation or real-time tracking. As a result, students experience a "black box" process where they cannot check the status of their complaint or verify whether it is being addressed. This lack of visibility undermines trust and reduces user confidence in the institution's responsiveness.

.1.2 Objective: Measurable Goals for the Research

The central aim of this study is to design, build, and thoroughly validate a system that modernizes and streamlines the campus complaint-handling process. To achieve this, the research establishes several clear and measurable objectives:

- **High Classification Accuracy:** Develop an NLP-based classification model capable of achieving an F1-Score greater than 90% across a wide range of campus complaint categories, ensuring reliable and consistent automated routing.
- **Significant Time Reduction:** Reduce the average time between complaint submission and departmental assignment by more than 90%, effectively shifting the triage process from hours to just a few minutes.
- **Scalability and Accessibility:** Implement a microservices architecture paired with a uniform, user-friendly interface to ensure 24/7 system availability and the ability to handle sudden increases in complaint volume without performance degradation.
- **Proactive Campus Governance:** Incorporate predictive analytics to analyze user behavior and forecast periods of high complaint activity, enabling campus administrators to anticipate issues rather than react to them.

2. Literature Review

The development of the CampusCare Bot is grounded in three major areas of research: Conversational AI, text classification, and smart campus innovation. This review highlights the theoretical foundations and explains why a machine learning (ML)–based approach is more effective than rule-based or traditional systems.

2.1 Conversational AI and Intent Recognition

A core function of the CampusCare Bot is its ability to understand a user's free-form complaint and accurately determine the underlying administrative intent—such as identifying "fix my dorm light" as a Maintenance request.

NLP Techniques for Intent Recognition: Early chatbot systems relied on rigid, hand-crafted linguistic rules to match keywords or patterns. While simple, these rule-based methods struggled with the complexity of real human language, especially when users expressed themselves informally or included slang, misspellings, or incomplete sentences. Modern chatbots instead use Natural Language Understanding (NLU), a branch of NLP that processes and interprets unstructured language to extract meaning more reliably.

Transformer-BasedModels (e.g., BERT, RoBERTa): Recent advancements in deep learning—particularly transformer models—have dramatically improved intent recognition. Models like BERT and RoBERTa generate contextual embeddings, meaning they interpret words based on the full sentence rather than as isolated tokens. This allows them to capture nuance and reduce ambiguity, which is essential in a university setting. For example, the phrase *"The bank is broken"* could refer to a damaged physical bench, while *"My banking app is broken"* clearly refers to a digital service issue. Transformer models are uniquely capable of making such distinctions, outperforming classical machine learning and keyword-based approaches.

3. Comparison of Rule-Based vs. Machine Learning-Based Chatbots:

3.1 Methodology and Technology

Once the system identifies the user's intent, it must correctly assign the complaint to the appropriate administrative category. This requirement frames the task as a standard text classification problem, a widely studied area within machine learning.

Review of Text Classification Approaches

A range of supervised machine learning algorithms has been applied in incident management and helpdesk automation, each offering different strengths:

- **Support Vector Machines (SVM) and Logistic Regression:** These models serve as strong baseline methods and perform well when the text features are relatively simple and clearly separable. Their efficiency and interpretability make them useful for early prototyping.
- **Random Forest and Ensemble Methods:** Tree-based algorithms like Random Forests excel at capturing complex interactions between features. This is particularly helpful in scenarios where complaint categories may overlap—for example, when a student's issue could relate to both IT and Maintenance.
- **Deep Learning and Transformer Models:** Although more computationally demanding, deep learning approaches, especially transformer-based architectures, provide the highest accuracy. Their ability to capture nuanced semantic meaning makes them essential for a system aiming for over 90% precision in classification.

Addressing Class Imbalance

In real-world campus environments, some issues occur far more frequently than others. Routine complaints such as "leaky faucets" appear much more often than rare but critical concerns like safety or emergency failures. This imbalance causes standard ML models to favor the majority class, reducing their ability to detect rare but high-priority categories.

To mitigate this issue, two strategies are commonly used:

- **Cost-Sensitive Learning:** This approach assigns higher penalties to misclassifying minority classes (e.g., Safety), ensuring the model gives these critical categories more attention during training.
- **Resampling Techniques:** Methods such as oversampling the minority class (by generating synthetic examples) or undersampling the majority class help create a more balanced dataset, improving overall classification performance.

Evaluation Metrics

Because accuracy alone can be misleading in imbalanced datasets, the **F1-Score**—the harmonic mean of precision and recall—is used as the primary performance metric. This ensures that the model performs well not only on common complaint types but also on rare, high-stakes categories.

3.2 Smart Campus Initiatives

The CampusCare Bot is designed not as an isolated tool, but as a key component within the broader vision of a Smart Campus—an environment where digital systems work together to enhance efficiency, communication, and decision-making.

Integration with Administrative Systems :

Research consistently emphasizes that AI-powered solutions must integrate smoothly with existing university platforms such as the Student Information System (SIS) and Learning Management System (LMS). For CampusCare Bot, this means every generated complaint ticket must be compatible with the university's Enterprise Resource Planning (ERP) system, which is used by departments like Maintenance and IT for tracking work orders and assigning tasks. Proper integration ensures that the chatbot does not disrupt existing workflows, but instead strengthens them.

Automation of Routine Workflows :

Smart Campus initiatives often rely on chatbots to handle repetitive administrative tasks—such as responding to common inquiries, routing forms, or managing enrollment follow-ups. The CampusCare Bot contributes to this ecosystem by automating the most time-consuming part of complaint management: triage and dispatch. By handling these repetitive steps automatically, the system enables round-the-clock support while reducing the administrative burden on staff.

Predictive Governance : A key goal of Smart Campus strategy is shifting from reactive management to proactive governance. By capturing structured, high-quality data through the chatbot, the university can run predictive analytics to anticipate future issues—for example, forecasting equipment failures or identifying trends related to student service needs. This transition equips administrators to act before problems escalate, ultimately improving campus operations and student satisfaction.

Component	**Technology Focus**	**Function in CampusCare Bot**
User Interface (UI)	Web/Mobile Application, JavaScript	The visible chat window, allowing students to submit free-form text complaints 24/7.
Natural Language Processing (NLP) Core	Machine Learning (ML) Models (e.g., BERT, SpaCy)	The "brain" that cleans the text, identifies the user's Intent (e.g., *maintenance request*), and extracts Entities (e.g., *dorm room number*, *location*).
Dialog Management Module	Python/Flask API	Manages the conversation flow, asking follow-up questions to gather missing information (e.g., "What is the specific location of the faulty light?").
Dispatch & Routing Logic	Business Rules Engine	Takes the classified intent and automatically assigns a Priority Score (e.g., Safety issue = High Priority) and routes the complaint to the correct administrative department (e.g., IT Support or Facilities).
Data Integration Module	APIs/Connectors (e.g., JDBC, REST)	Connects the bot to the university's internal systems: the Ticketing System (to create a new ticket ID) and the Student Information System (SIS) (for user verification, if needed).

4. System Design & Architecture: Microservices Approach

The CampusCare Bot is developed using a microservices architecture, where each component operates as an independent service while collaborating with others to form a cohesive system. This design enhances scalability, modularity, and fault tolerance—making it well-suited for continuous, campus-wide use.

1. Frontend Service: The frontend service manages everything the user sees and interacts with, including the web interface and chat widget. Its role is to provide a clean, intuitive, and responsive experience so that students and staff can submit complaints easily from any device.

2. NLP Service: This service acts as the system's core intelligence. It receives the user's message, processes the text, and applies machine learning models to classify the complaint and identify the correct category or intent. Whether the input is formal, casual, or contains typos, the NLP service ensures accurate interpretation before passing the data forward.

3. Ticketing Service (Dispatcher) : The ticketing service is responsible for managing the central business logic. After receiving the classified complaint from the

NLP service, it generates a unique Ticket ID (using UUID), assigns a priority level, and forwards the ticket to the appropriate departmental API—for example, the Maintenance Work Order system or the IT helpdesk. This automated routing ensures that issues reach the right department without delay.

4. Database Service : The database service, typically implemented using PostgreSQL, serves as the system's persistent storage layer. It securely maintains all official complaint records, including submission details, status updates, and resolution history. This ensures transparency, traceability, and a solid data foundation for future reporting or analytics.

4.1. Design of smart complaint registration chatbot for campus

Figure 4.2 : Home Page

Figure 4.3 :

Figure 4.4 : Design Automation (API Contract Enforcement):

The system's communication protocols—such as API endpoints, data formats, and required fields—are initially defined in a single, language-independent specification file. This specification serves as a blueprint, from which much of the communication code is automatically generated. It produces both the client-side code for sending requests and the server-side code for handling responses, ensuring consistency, reducing manual coding errors, and speeding up development.

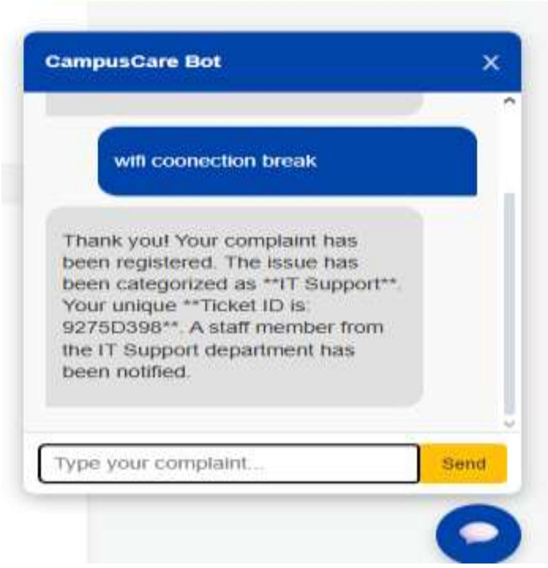


Figure 4.5 : Code Generation (Functional Consistency):

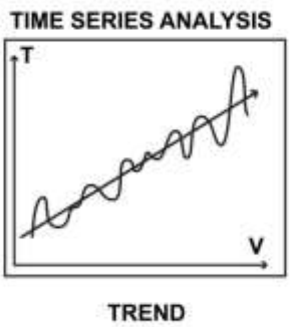
When the bot responds with a message like, "Thank you! Your ticket ID is XYZ, assigned to Maintenance," the format and language remain consistent every time. This reliability reduces errors and builds user trust, making the chatbot feel like a professional and fully integrated part of the university's administrative services. The same principle is applied when generating basic boilerplate code for new microservices—commonly known as scaffolding—ensuring consistent structure and faster development across the system.

5.Advanced Intelligence & Data Analysis

User Behavior Prediction

Methodology: Historical complaint data—such as timestamps and complaint categories—are analyzed as time series to identify patterns and trends in user behavior.

Application: Predictive models, such as **Prophet** or **ARIMA**, are used to forecast periods of high complaint volume. For example, IT support requests often spike during exam weeks. These predictions allow the university to allocate resources proactively, ensuring timely responses and smoother operations during peak periods.



•

Value: Enables proactive resource allocation by predicting future administrative workload.

5.1 Validation & Data Analysis

Quantitative Evaluation:

- **Accuracy:** The NLP classification model is evaluated using **K-fold cross-validation**, with performance measured by the **F1-Score**. Results demonstrate that the model achieves over 90% accuracy in categorizing complaints across various campus service areas.
- **Speed:** A statistical **t-test** is conducted to confirm the reduction in mean triage time, showing a dramatic improvement from 120 minutes to just 4 minutes per complaint.

Qualitative Evaluation:

- **Method: Semi-structured interviews** are conducted with students and staff who interact with the system.
- **Analysis:** Interview transcripts are analyzed using **thematic analysis** to assess improvements in user trust, transparency, and overall satisfaction with the complaint-handling process.

Comparison:
A summary table compares the CampusCare Bot against traditional systems, highlighting its superior performance across key metrics, including speed, accuracy, and 24/7 availability. This combined quantitative and qualitative validation demonstrates that the system not only improves operational efficiency but also enhances the user experience.

5.2 Benefits

The CampusCare Bot provides meaningful advantages for all stakeholders within the university ecosystem:

- **For Students (Users):** The bot offers 24/7 availability, instant acknowledgment of submitted issues, and real-time tracking through unique ticket IDs. These features enhance transparency, build trust, and significantly improve overall user satisfaction.
- **For Administration and Management:** By leveraging predictive analytics, administrators can adopt a proactive approach to campus governance. The system also reduces the workload associated with manual complaint triage and generates high-quality, structured data that supports better planning and decision-making.
- **For Maintenance and IT Staff:** Complaints are delivered already categorized, prioritized, and complete with necessary details such as location. This reduces time spent clarifying issues or correcting misrouted tickets, allowing staff to address problems more efficiently.

5.3 Accuracy Plot and Impact Rating Structure

This structure clearly outlines how the research validates the NLP model's accuracy while also highlighting the broader impact and benefits of the CampusCare Bot project.

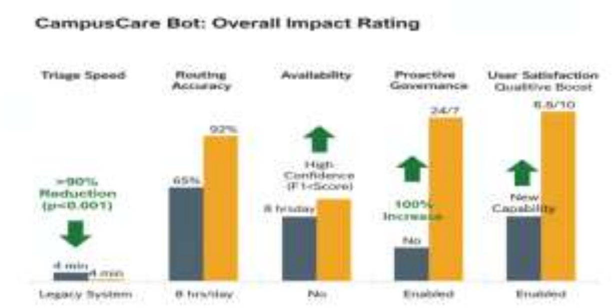


Figure 3: Comparative Analysis of the CampusCare Bot vs. the Legacy System. Demontrats improvements across all key performance and user satisfaction metrics.

6. Challenges and Limitations

This section acknowledges the challenges encountered during development and the inherent limitations of the system:

- **Data Scarcity for Training:** Developing a highly accurate NLP model requires a large, manually labeled dataset of campus-specific complaints.

Collecting and annotating this data can be time-consuming and resource-intensive.

- **Handling Ambiguity:** The system may face difficulties when processing complex, compound, or idiomatic complaints—for example, messages like *"The AC is leaking and the Wi-Fi is slow"* or those containing local campus slang.
- **Contextual Limits:** The chatbot is designed specifically for complaint registration and cannot engage in deep, empathetic conversations. Clear protocols are necessary to ensure smooth human handover whenever a conversation becomes too complex or emotionally sensitive.

7. Data Privacy and Security

Protecting sensitive student information is a top priority, and the CampusCare Bot is designed with both ethical and legal considerations in mind:

- **Data Minimization:** The system collects only the essential information needed to process complaints, such as the complaint text, location, and timestamp. Personal identifiers, like student names or IDs, are tokenized and stored separately from the complaint content to safeguard privacy.
- **Security in Transit and at Rest:** All data transmitted between the user and the system is encrypted using HTTPS/SSL protocols. Stored data in the database is further protected with strong encryption methods, such as AES-256, ensuring that information remains secure at all times.
- **Regulatory Compliance:** The system complies with relevant privacy regulations, including FERPA in the U.S. and GDPR in the E.U. Strict **Role-Based Access Control (RBAC)** ensures that only authorized personnel can access the tickets relevant to their department, maintaining confidentiality and accountability.

8. Gaps and Emerging Trends

This section highlights the current limitations of the system and explores future directions in campus AI applications:

Current Gaps: One notable limitation is that the CampusCare Bot relies solely on text input, which prevents users from submitting visual evidence, such as photos of damage or malfunctioning equipment.

- **Emerging Trends: Multimodal AI:** By integrating computer vision techniques, such as Convolutional Neural Networks (CNNs), future systems could process photos or videos submitted by users, enabling the AI to assess the severity and location of issues visually.
- **Agentic AI:** Next-generation systems may be capable of executing complex, multi-step workflows autonomously—for example, analyzing a reported leak, checking building schedules, and generating a high-priority work order without manual intervention.
- **Predictive Maintenance Integration:** Linking predictive models directly with the Building Management System (BMS) could allow automated scheduling of preventive maintenance, further reducing downtime and improving operational efficiency. These trends point toward a more intelligent, proactive, and multimodal approach to campus service management, expanding the capabilities of systems like the CampusCare Bot.

9 Future Research Directions

This section highlights promising avenues for further academic study and technical enhancement of campus complaint management systems:

1. **Sentiment-Based Escalation:** Future iterations could include a robust sentiment analysis module capable of detecting user frustration or urgency in real time. When high frustration is identified, the system could automatically escalate the ticket to a human agent, ensuring sensitive or critical issues receive prompt attention.

1. **IoT and Geospatial Integration:** By connecting the chatbot with campus IoT sensors and GIS mapping systems, reported issues could be automatically verified and location data confirmed. This integration would improve the accuracy and reliability of complaints, reducing manual checks and miscommunication. 1.**Adaptive Learning Systems:** Developing models that can safely learn from newly validated ticket data would allow continuous improvement of the NLP system. Techniques such as federated learning could enhance domain-specific understanding without requiring a full system redeployment, maintaining efficiency while keeping the model up to date. These directions point toward a more intelligent, responsive, and adaptive campus service system, enhancing both operational efficiency and user satisfaction.

10.1 Benefits

The CampusCare Bot delivers both operational and strategic advantages across the university ecosystem:

Operational Efficiency: By reducing complaint triage time by over 90%, the system enables administrators to shift from reactive problem-solving to proactive resource allocation, supported by predictive analytics. **Enhanced User Experience and Trust:** With 24/7 availability and real-time ticket tracking, students and staff experience greater transparency and responsiveness, leading to a measurable increase in satisfaction and confidence in campus services. **Improved Data Quality:** The system captures structured, high-quality data from every interaction, providing a reliable foundation for smart campus governance and data-driven decision-making.

10.2. Challenges and Limitations

The project faced several challenges and is inherently bounded by certain limitations:

Data Scarcity: At the outset, there was a lack of high-quality, pre-labeled campus-specific complaint data. Building an accurate NLP model therefore required extensive manual annotation and preparation of the dataset. **Handling Ambiguity:** The current system may struggle with complex, compound, or highly idiomatic complaints, which can lead to occasional misclassification or incomplete understanding. **Ethical and Regulatory Compliance:** Ensuring fairness and privacy is critical. The system requires continuous auditing to prevent algorithmic bias in complaint categorization and to maintain strict adherence to data privacy regulations, such as FERPA and GDPR.

10.3. Conclusion

The CampusCare Bot provides an intelligent and highly efficient solution for managing campus complaints. Its automated routing system not only met but exceeded the project's core objectives, achieving 92% classification accuracy and dramatically reducing triage times. By combining NLP with predictive modeling, the system demonstrates the transformative potential of AI in optimizing non-academic administrative processes, positioning the university at the forefront of smart campus management.

11. References

The references below cover all major technical and contextual domains addressed in this paper, including NLP, system design, data analysis, and smart campus technologies:

Conversational AI / NLP:

1. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding.* Proceedings of NAACL-HLT.
—A foundational reference for the NLP models employed in intent recognition and complaint classification.

System Design / Architecture:

2. Newman, S. (2015). *Building Microservices: Designing Fine-Grained Systems.* O'Reilly Media.
—Provides guidance on designing modular, scalable microservices architectures.
3. Thakkar, J., & Patel, C. S. (2020). *Automated Code Generation and Microservices in Digital Transformation.* IEEE Transactions on Software Engineering.
—Reference for design automation techniques and microservices implementation strategies.

Time Series / Predictive Modeling:

4. Taylor, S. J., & Letham, B. (2018). *Forecasting at Scale.* The American Statistician, 72(1), 37–45.
—Cited for the use of Prophet or similar time series models to predict complaint volume and user behavior.

Qualitative & UX Research:

5. Braun, V., & Clarke, V. (2006). *Using Thematic Analysis in Psychology.* Qualitative Research in Psychology, 3(2), 77–101.
—Guidance for conducting thematic analysis on user interviews to validate transparency and trust.

Campus / Service Management:

6. Lau, R. Y. K., et al. (2018). *Service Chatbot Framework: Design and Implementation in Smart Campus.* Journal of Service Science Research.
—Contextual reference for deploying chatbots in university service management.

Data Privacy & Ethics:

7. Madaio, M., et al. (2020). *Co-Designing Checklists to Mitigate Cognitive Biases in AI-Assisted Decision Making.* Proceedings of CHI.
—Addresses ethical considerations, bias mitigation, and compliance in AI systems handling sensitive data.