Smart Civic Complaint Analyzer Using Natural Language Processing

C. Ashwitha[1], K. Vani[2]
[1]*Student, Department of Computing-Decision and Computing Sciences, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu 641014, India*
[2]*Assistant Professor, Department of Computing-Decision and Computing Sciences, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu 641014, India*

***Abstract*—The Smart Civic Complaint Analyzer is an AI-enabled web interface designed to enable local citizens to find and report on city problems easier while city officials can be faster to respond. The platform incorporates Natural Language Processing (NLP) to identify complaints made in simple text by a person. The system can automatically categorize the type of complaint (Water, Roads, Electricity, or Sanitation), identify how urgent the complaint is (High, Medium, or Low) and identify the location. Local people can easily report a complaint or can upload multiple complaints in a compatible CSV format. Users can see all complaints submitted to the system, search and filter complaints to manage and track their reports and complaints. City officials can use an interactive dashboard to view the information submitted through visual charts and word clouds live and have summaries of the data to assist them to measure the most common, frequent and serious complaints in an area. The application is developed in Python and uses Streamlit to build the User Interface (UI) application. The text analysis used machine learning models including TF-IDF and Naive Bayes. The project stores data as an excel workbook file, to ensure while data is easy to access, the data can also be exported in a similar format. Municipalities can use this system to save time, better plan-prevention, and increase transparency with the public, while the public has a quicker and more open way to report on local problems. Overall, this project is designed to make smart cities smarter, faster, and more citizen friendly.**

***Index Terms*—Natural Language Processing, Machine Learning, TF-IDF, Naive Bayes, Complaint Classification, Urgency Detection, Location Extraction, Word Cloud, Citizen Engagement, Data Visualization, Municipal Management, Smart City Solutions.**

I. INTRODUCTION

The challenge of managing civic problems has become quite serious for both citizens and the government authorities in large growing cities with continuously increasing populations. People living in big cities must face several civic problems in day-to-day life, such as broken roads, spilled over garbage bins, water leakage, damaged streetlights, and blocked drainage. All these causes a lot of inconvenience among the public and, if not repaired within a short time, invite various health and safety hazards, too. Generally, the traditional process of reporting civic problems is sluggish and time consuming. It involves visiting municipal offices or making phone calls or sending emails to the concerned department, which generally leads to unwanted delays and unrecorded complaints. For that reason, many problems remain unsolved for an extended period. So, here's where the Smart Civic Complaint Analyzer comes in. The whole point is to make it easier for city officials to handle complaints. This system leans on AI and Natural Language Processing to read and break down what people are saying when they file a complaint. The moment someone submits an issue; the analyser gets to work. It figures out what kind of problem it is, maybe something with the roads, water, or sanitation it decides how urgent it feels, and picks out where it's happening. With all that info, officials don't have to waste time sorting things out. They see what's wrong, where it is, and how quickly they need to act.

Unlike traditional systems that depend on manual reading and sorting of complaints, this analyser saves a lot of time by turning text into meaningful and organized data within seconds. It is also capable of

generating visual outputs such as charts, word clouds, and reports on which problems happen most often within the city. This gives officials a clear picture of the city's condition and helps them zero in on the most serious or common issues first. The platform is built in Python and Streamlit for ease of use and accessibility across multiple devices. The system also uses machine learning techniques such as TF-IDF and Naive Bayes for correct complaint categorization. All the processed complaints are kept in Excel or CSV files, later to be used for review, tracking, or generation of detailed reports. The main aim of the project is to facilitate a faster, easier, and more efficient complaint-handling system for citizens and municipal officers. The Smart Civic Complaint Analyzer not only increases the efficiency in problem reporting but also ensures transparency and accountability within the civic system. By automating the analysis process, it reduces human error, saves time, and makes sure that no complaint goes unnoticed. The proposed system will act as an essential step toward creating smarter, cleaner, and more responsive cities where people's voices are heard, and problems are solved without delays.

II. METHODOLOGY

The Smart Civic Complaint Analyzer makes it easy for people to report issues in their city and helps government officers respond faster. Basically, the system uses AI and Natural Language Processing to read and sort through complaints on its own, so there's less manual work and things move more quickly. Everything runs in a clear, organized way from collecting reports to analysing them and showing the results in a way that makes sense.

The overall architecture of the proposed system can be divided into three layers:

1. User Interaction Layer
2. Processing and Analysis Layer
3. Data Storage and Visualization Layer

Each layer performs specific tasks that together make the system smart, fast, and user-friendly.

2.1 User Interaction Layer

This layer represents the citizen side of the system. It allows users to file complaints with ease through a simple, intuitive web interface built using Streamlit.

Here, citizens can:

Type a description of the problem, such as: "There is a water leakage near my house" or "The streetlight is not working". Upload a photo related to the issue, if they want to provide more information. Please enter the location or area name to help our system locate where the problem is occurring and the citizens can also upload their contact information for faster resolution. Once the citizen clicks the Submit button, the complaint details directly get routed to the backend system for processing. Through this approach, citizens need not visit municipal offices or make phone calls. The interface is designed to be user-friendly even for people with minimal technical knowledge.

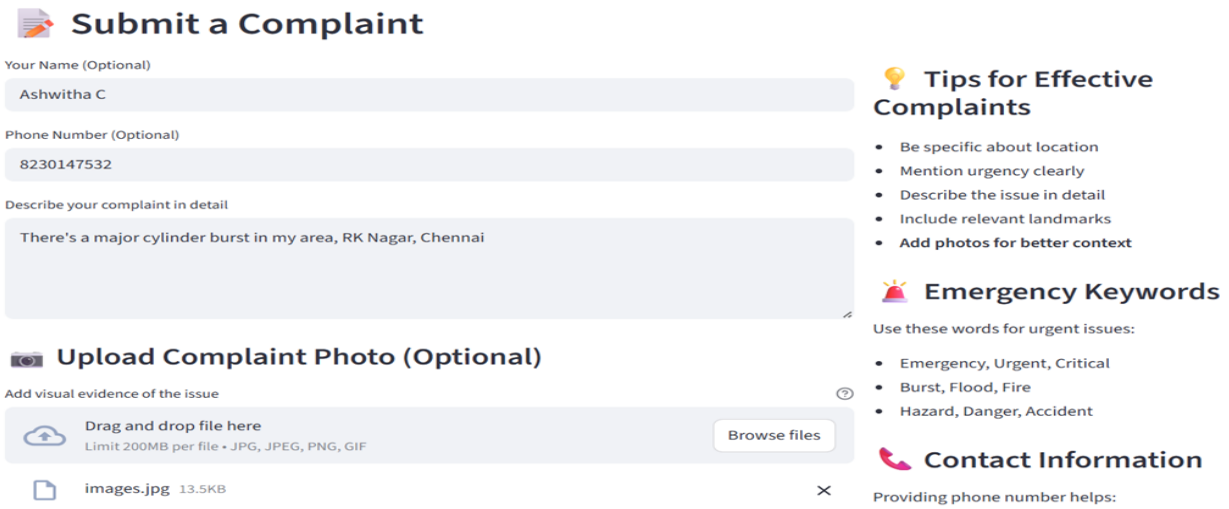

Figure 1. User Interface for Complaint Submission

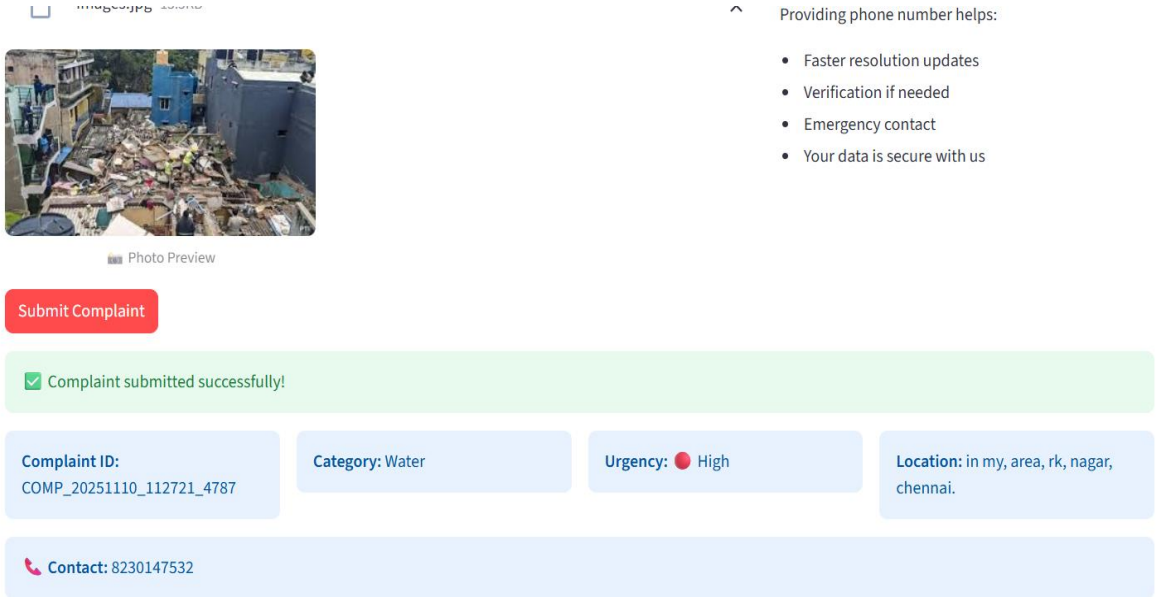

Figure 2. Instant AI Analysis and Confirmation of Submitted Complaint

2.2 Processing and Analysis Layer

This layer forms the core of the Smart Civic Complaint Analyzer, where all data processing and intelligence work are done. The backend is built in Python for reading, analysing, and classifying complaints provided by users. Once a complaint is received, the system performs several important steps:

2.2.1 Data Preprocessing

Cleaning the text submitted by the user is performed to remove unnecessary characters, punctuation, and extra spaces from the text, so that the complaint text is clear and ready for machine learning analysis.

For example:

Converting all words to lowercase.

Removing special symbols or numbers.

Tokenizing, or breaking the text into its individual words. This was done to ensure the AI model understood precisely what was intended.

2.2.2 Feature Extraction

After pre-processing, the system then changes the cleaned text into numerical form for machine learning algorithms to understand. The method in use is TF-IDF, which means Term Frequency–Inverse Document Frequency. It gives more importance to unique words which appear often in a complaint but not in others. For instance, words such as "pothole", "leakage", or "garbage" are treated as important keywords to help the system understand the problem category.

2.2.3 Complaint Classification

After turning the complaint into numbers, the system jumps right in and uses a Multinomial Naive Bayes classifier to figure out what kind of problem it's looking at. It sorts each complaint into one of several main categories, like:

- Road - think potholes or damaged streets
- Water / air - stuff like leaking pipes or supply problems
- Sanitation - missed trash pickup, drainage trouble
- Electricity or anything else

The system knows how to spot and sort just about any complaint you throw at it, just by looking at the details.

2.2.4 Urgency Detection and Location Identification

Besides identifying the issue category, the system also checks how serious or urgent the problem is. This is done through keyword matching. The text is scanned for certain words indicating urgency, such as "burst," "accident," "danger," or "immediate." Depending on the matched words, the complaint will be assigned to a priority level:

High - Requires immediate attention: burst water pipe, electrical hazard

Medium - Needs attention soon

Low - Can be addressed later

At the same time, the system identifies the location by recognizing place names or location keywords like "street," "road," "area," or "colony." This helps the officials locate where the problem exists more quickly.

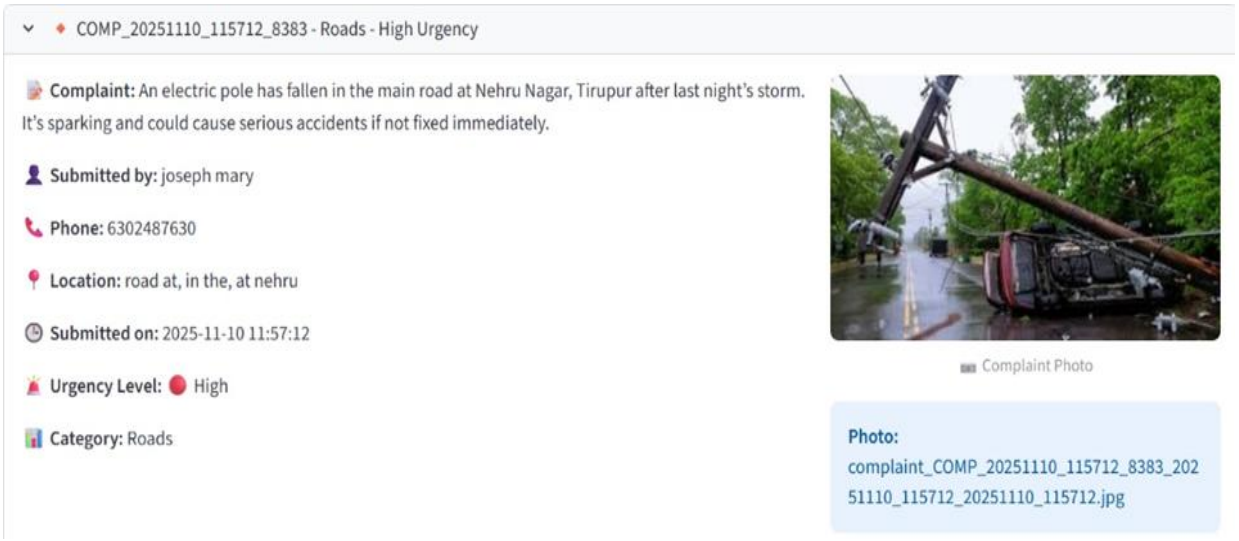

Figure 3. Detailed Complaint Record Displaying Real-time AI Classification, Urgency, and Visual Evidence

Data storage and visualization layer

After processing, all the complaint information will be automatically saved into an Excel or CSV file using Python's Pandas library: Complaint ID, Category, Urgency Level, Location, and Photo filename. This storage method makes it easy to maintain records, generate reports, and view complaint histories whenever needed. The stored data is also connected to a dashboard built in Streamlit.

This dashboard enables municipal officers to:

View all complaints in one place, Filter complaints based on type or urgency, See visual reports in the form of charts, graphs, and word clouds. For instance, a pie chart shows the percentage of different complaint categories such as road, water, sanitation, etc. Bar chart showing the total number of complaints received per week. A word cloud shows the most frequently appearing words in complaints. This visualization serves as an aid for city officials to make quick and sound decisions as it clearly shows which type of problems are frequent, and which areas need extra attention.

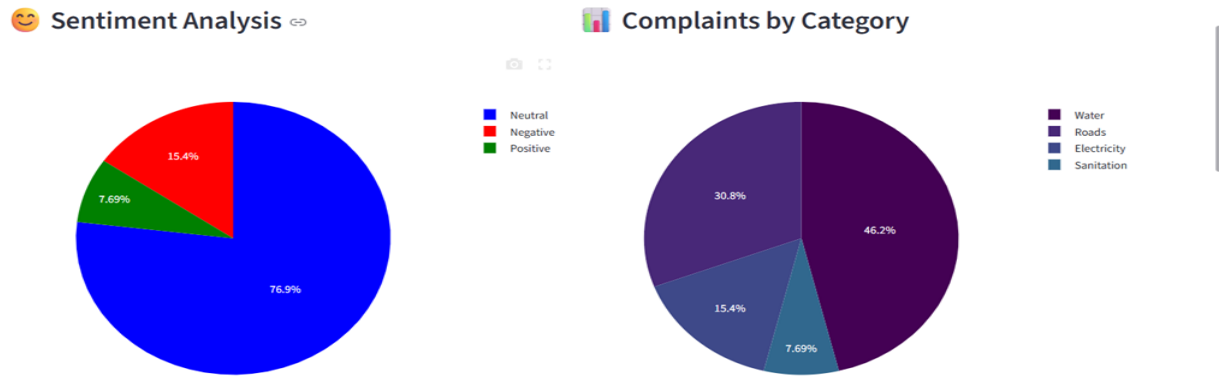

Figure 4. Analytics Dashboard Highlighting Complaint Distribution by Category and Sentiment

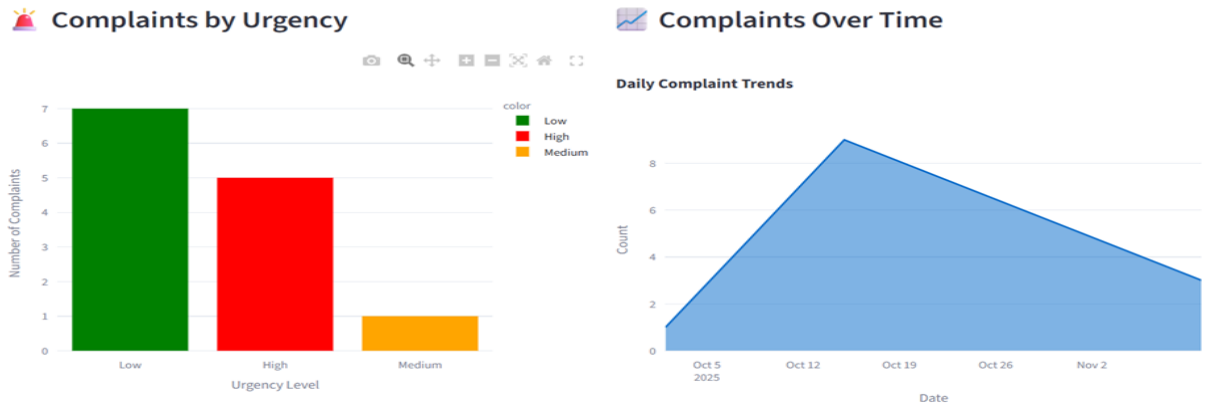

Figure 5. Distribution of Complaints by Urgency Level and Daily Reporting Trends

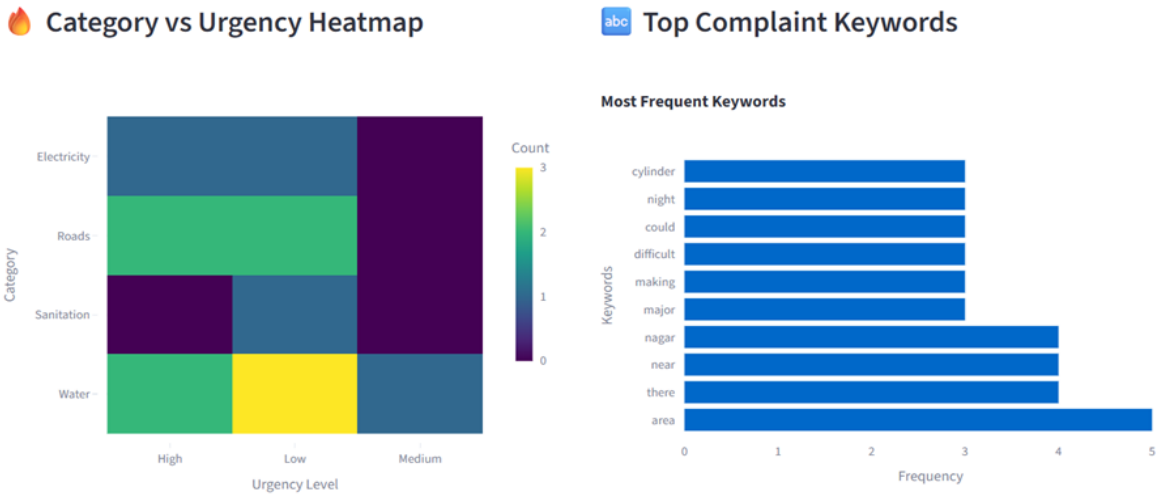

Figure 6. NLP-Powered Insights: Correlation Between Issue Type, Urgency, and Top Complaint Keywords

III. WORKING FLOW OF THE SYSTEM

1. The citizen opens the Streamlit interface and submits a complaint with details and an optional image.
2. The complaint data is sent to the Python backend.
3. The NLP engine cleans the text and processes it.
4. The AI model classifies the complaint type and determines urgency and location.
5. The processed data is saved in Excel or CSV files.
6. The dashboard updates automatically with new data for officials to view and act upon.

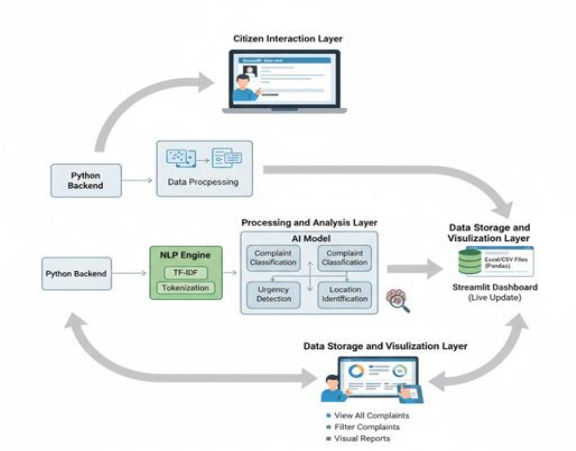

Figure 7. Comprehensive Working Flow of the Smart Civic Complaint Analyzer System

Algorithm: Smart Civic Complaint Analyzer
Input:

- Complaint tex0t and image uploaded by the citizen
- Citizen details (name, location, contact, etc.)

Output:

- Classified complaint type (Water / Road / Sanitation)
- Urgency level (High / Medium / Low)
- Extracted location and stored report in Excel or CSV file

```
Procedure SmartCivicComplaintAnalyzer()
Begin
   Initialize an empty complaint database.
  While system is active do
     Receive complaint input from citizen through Streamlit interface.
     If image is uploaded then
        Store image with complaint details.
     End If
    Preprocess the complaint text:
        Remove special characters and stop words.
        Convert text to lowercase.
        Tokenize the words.
    Apply NLP model to analyse the complaint:
        Use TF-IDF to extract key features.
        Use Naive Bayes classifier to identify issue type (Water / Road / Sanitation).
        Predict urgency level (High / Medium / Low).
        Extract location details using text pattern matching.
    Save results in Excel/CSV file using Panda's library.
     Update dashboard with charts and word clouds showing complaint trends.
     Generate confirmation message for citizen submission.
  End While
End Procedure
```

IV. RESULTS AND DISCUSSION

The Smart Civic Complaint Analyzer was designed to identify and classify different civic complaints such as garbage issues, road damage, water leakage, electricity failure, and drainage blockage. The model was able to identify and classify most of the complaints correctly during testing whenever a text description was provided. For instance, when entering a complaint like "there is water leaking from the main pipe near my street," the model was able to label it under Water Issue. Similarly, a complaint such as "the streetlights are not working at night" was classified as an Electricity Problem which proved that TF-IDF and Naïve Bayes are a good combination for text classification. It could also identify the level of urgency for each complaint for example, drainage blockages or anything to do with road damage would fall under high priority, while minor complaints about garbage would be marked as medium or low priority.

A dashboard was created to make the system more useful for authorities, where all the complaints were displayed according to their categories and priority levels. Graphs and charts were generated indicating how many complaints belonged to each type. For instance, garbage issues were highlighted as the most common type of complaint in the dashboard, followed by problems pertaining to roads. This helped in understanding which areas needed more attention from the municipal department. The system also included a word cloud feature: it visually showed the most frequently used words in the complaints. Words like "road," "garbage," "water," and "leakage" appeared larger, showing that these were the main issues faced by the public.

The analyser was tested by providing various inputs, which it processed efficiently and returned results very

fast. For any complaint to be processed and categorized, it takes only a few seconds. A simple and user-friendly interface was built with Streamlit that allowed citizens to input complaints smoothly without any confusion. Overall, the Smart Civic Complaint Analyzer provides accurate, fast, and meaningful insights into civic complaints. The model could be taken up by the local government bodies to help reduce the manual effort needed, provide quicker response times, and cleaner and more efficient cities. The project demonstrated how machine learning and natural language processing can be applied to real life civic management systems.

V. CONSIDERATIONS AND CHALLENGES IN SYSTEM IMPLEMENTATION

Various practical and technical challenges had to be overcome while building the Smart Civic Complaint Analyzer to ensure it works smoothly in real time. The main aim was to create a fast, reliable, and easy to use platform by both the citizens and city officials without confusion. The main challenge was to make the system accurately comprehend each complaint. Everyone describes a problem using different words. Therefore, it was solely necessary to ensure that the NLP model correctly identifies the type of issue the person is reporting. For enhancing accuracy, the system was trained carefully and defined rules were set for main categories such as Road, Water, Electricity, and Sanitation. This would reduce the chance of mistakes and keep every complaint sorted in the right type. Another challenge was the detection of how urgent the problem is and finding the correct location from the text. People express urgency differently for example, one might say “please fix it immediately” while another might write “urgent repair needed.” The system was trained to understand such words and assign the right urgency level as High, Medium or Low. Similarly, the system was also designed to pick up location names from plain text, even when people don’t follow a fixed format, which made it more flexible and useful.

Next, the system had to be fast and responsive. Since complaints are processed in real time and dashboarded, it is important to ensure that the system works without any delay. Optimizing both the backend and the frontend kept the system quick, even with many users submitting complaints all at the same time. Another huge focus was ease of use. The web page was designed using Streamlit, and it was made very simple and neat so that even people who do not know much about technology could easily report issues. It would feel comfortable for everyone, including elderly citizens, to clearly see buttons, simple forms, and visual charts. Another important factor was making the system flexible for future updates. The design has been made quite modular, meaning one can later add new features, categories, or even better machine learning models, without rebuilding the whole system. In this way, the platform can grow and adapt with time, meeting the changing needs in the city.

Finally, the issue of transparency and record keeping was of utmost importance. All the grievances and replies are stored in either an Excel or a CSV file which enables the officials to easily access the old records and monitor the progress of the solution. This, in addition, strengthens the relationship of trust between the public and the government.

REFERENCES

[1] E. D. Madyatmadja, H. Nindito, and D. Pristinella, “Citizen Behavior: The Evaluation of Complaint Application that Connected to Smart City,” Advances in Science, Technology and Engineering Systems Journal, vol. 5, 2020.

[2] M. Silpa Raj, P. Sambasiva Rao, G. K. Monica Nandini, S. Sureshkumar, A. Kumar Mishra, and G. Saritha, “Artificial Intelligence Driven Predictive Analytics for Real-Time Civic Engagement and Smart Decision Making in Future Urban Governance,” Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024)

[3] A. Reethika, B. Sai Divya, B. Tharun Teja, D. Adhitya, and U. S. Bharathi, “SMART CITY COMPLAINT ANALYZER,” International Research Journal of Modernization in Engineering Technology and Science (IRJMETS), vol. 07, August 2025.

[4] S. Kumar, S. Atreja, A. Singh, and M. Jain, “Adversarial Adaptation of Scene Graph Models for Understanding Civic Issues,” arXiv:1901.10124, Jan. 2019

[5] “Digital Grievance Redressal for Cleaner, Smarter India,” International Journal of Current

Trends (IJCT), vol. 12, June 2025. Authors: S. Raj and A. Kumar.

[6] Alok P. Singh, Ankur Goel, Aakansha Goel & Diksha Arya. "NLP based Grievance Redressal System." International Journal of Computer Applications, Vol. 184, May 2022.

[7] M. Silpa Raj, P. Sambasiva Rao, G. K. Monica Nandini, S. Sureshkumar, Amit K. Mishra & G. Saritha. "Artificial Intelligence Driven Predictive Analytics for Real Time Civic Engagement and Smart Decision Making in Future Urban Governance." Proceedings of the International Conference on Sustainability Innovation in Computing and Engineering (ICSICE 2024), Atlantis Press, May 2025.

[8] Gowda, N. D., Bhavya V., Hindasagatti M., Monika T. S., & Pallavi D. R. "Urban-Echo: Smart reporting for smart cities." American Journal of Sustainable City and Society, Vol. 15, 2025.

[9] Verma, D. & Thevar, M., "Complaint Classification Model Using NLP." IJRASET Journal for Research in Applied Science and Engineering Technology, Vol. (2025).

[10] Dwivedi, R. K., Nand, P., & Pal, O., "NLP-reliant Neural Machine Translation techniques used in smart city applications." Information System and Smart City, Vol. 3, 2023.

[11] Nosratabadi, S., Mosavi, A., Keivani, R., Ardabili, S., & Aram, F., "State of the Art Survey of Deep Learning and Machine Learning Models for Smart Cities and Urban Sustainability.", arXiv:2010.02670 (2020).

[12] Nimase, S., Kanawade, A., Kamble, S., & Khemnar, S., "Public Complaint Sorting for Smart City using Image Processing." International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE), 2021.

[13] Classifying Crowdsourced Citizen Complaints through Data Mining: Accuracy Testing of k-Nearest Neighbors, Random Forest, Support Vector Machine, and AdaBoost - E. D. Madyatmadja et al., Informatics, vol. 10, 2023.