Hindustan Aeronautics Limited
Koraput Division, Odisha

PROJECT REPORT ON

<u>Smart Township</u>

(with Predictive Maintenance and Resident sentiment analysis)

<u>SUBMITTED BY</u>

NAME – Yubaraj Mohanty

REG.NO.- 2201110076

VT NO-(25-443)

COURSE – B.TECH IN COMPUTER SCIENCE & ENGINEERING

SEMESTER & YEAR- 6th semester & 3rd year

COLLEGE NAME – GCEK, BHAWANIPATNA

<u>UNDER THE GUIDANCE OF</u>

SHRI SWADESH BEHERA
(SR. MANAGER (IT), SED)

<u>SUBMITTED TO</u>

Training and Development Institute HAL KORAPUT

PROJECT REPORT ON

Smart Township

(with Predictive Maintenance and Resident sentiment analysis)

SUBMITTED BY

NAME – PRIYANKA DAS

REG.NO.- 2201202177

VT NO-(25-444)

COURSE – B.TECH IN COMPUTER SCIENCE & ENGINEERING

SEMESTER & YEAR-6th semester & 3rd year

COLLEGE NAME – NIST University, Berhampur

UNDER THE GUIDANCE OF

SHRI SWADESH BEHERA

(CH. MANAGER (IT),SED)

SUBMITTED TO

Training and Development Institute HAL KORAPUT

Hindustan Aeronautics Limited

Koraput Division, Odisha

PROJECT REPORT ON

Smart Township

(with Predictive Maintenance and Resident sentiment analysis)

SUBMITTED BY

NAME – PRAKRITI MAHARANA

REG.NO.- 4221050231

VT NO.-(25-440)

COURSE – B.TECH IN COMPUTER SCIENCE & ENGINEERING

SEMESTER & YEAR- 3rd SEM & 2ND Year

COLLEGE NAME –ACE, Jaipur

UNDER THE GUIDANCE OF

SHRI SWADESH BEHERA

(CH. MANAGER (IT),SED)

SUBMITTED TO

Training and Development Institute HAL KORAPUT

PROJECT REPORT ON

Smart Township

(with Predictive Maintenance and Resident sentiment analysis)

SUBMITTED BY

NAME – Abu Noor Al Saba

REG.NO.- 23110253

VT NO-(25-442)

COURSE – B.TECH IN COMPUTER SCIENCE & ENGINEERINGSEMESTER & YEAR-4TH SEM & 2ND Year

COLLEGE NAME – OUTR,BHUBANESWAR

UNDER THE GUIDANCE OF

SHRI SWADESH BEHERA

(CH. MANAGER (IT),SED)

SUBMITTED TO

Training and Development Institute HAL KORAPUT

PROJECT REPORT ON

Smart Township

(with Predictive Maintenance and Resident sentiment analysis)

SUBMITTED BY

NAME – Aditya Dash

REG.NO.- 2211100174

VT NO.-(25-441)

COURSE – B.TECH IN COMPUTER SCIENCE & ENGINEERING

SEMESTER & YEAR-6th SEM & 3rd Year

COLLEGE NAME – OUTR, BHUBANESWAR

UNDER THE GUIDANCE OF

SHRI SWADESH BEHERA

(CH. MANAGER (IT),SED)

SUBMITTED TO

Training and Development Institute HAL KORAPUT

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Mr. Basanta Kumar Nayak,**

Chief. Manager, Training And Development Institute, HAL for permitting me to carry out the internship at Hindustan Aeronautics Limited, Koraput.

I wish to express my deep sense of gratitude to my guide **SHRI SWADESH BEHERA ,CH. Manager IT Dept. (SED),** for his invaluable guidance and insightful suggestions, which were instrumental in enabling me to successfully complete the project on schedule.

Words are inadequate in offering thanks to the project assistants and all the employees in engine division and sukhoi engine division who helped me in understanding the techstacks used during the project work.

Finally, I am highly thankful to my family members and well-wishers who supported me and bestowed their good wishes for the successful completion of this project.

Yubaraj Mohanty

VT No. - (25-443)

Computer Science & Engineering

GCEK , Bhawanipatna

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Mr. Basanta Kumar Nayak,**

Chief. Manager, Training And Development Institute, HAL for permitting me to carry out the internship at Hindustan Aeronautics Limited, Koraput.

I wish to express my deep sense of gratitude to my guide **SHRI SWADESH BEHERA ,CH. Manager IT Dept. (SED),** for his invaluable guidance and insightful suggestions, which were instrumental in enabling me to successfully complete the project on schedule.

Words are inadequate in offering thanks to the project assistants and all the employees in engine division and sukhoi engine division who helped me in understanding the techstacks used during the project work.

Finally, I am highly thankful to my family members and well-wishers who supported me and bestowed their good wishes for the successful completion of this project.

Priyanka Das

Vt No.- (25-444)

Computer Science & Engineering

NIST UNIVRSITY, Berhampur

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Mr. Basanta Kumar Nayak,**

Chief. Manager, Training And Development Institute, HAL for permitting me to carry out the internship at Hindustan Aeronautics Limited, Koraput.

I wish to express my deep sense of gratitude to my guide **SHRI SWADESH BEHERA ,CH. Manager IT Dept. (SED),** for his invaluable guidance and insightful suggestions, which were instrumental in enabling me to successfully complete the project on schedule.

Words are inadequate in offering thanks to the project assistants and all the employees in engine division and sukhoi engine division who helped me in understanding the techstacks used during the project work.

Finally, I am highly thankful to my family members and well-wishers who supported me and bestowed their good wishes for the successful completion of this project.

Prakriti Maharana

Vt No.(25-440)

Computer Science & Engineering

ACE,Jaipur

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Mr. Basanta Kumar Nayak,**

Chief. Manager, Training And Development Institute, HAL for permitting me to carry out the internship at Hindustan Aeronautics Limited, Koraput.

I wish to express my deep sense of gratitude to my guide **SHRI SWADESH BEHERA ,CH. Manager IT Dept. (SED),** for his invaluable guidance and insightful suggestions, which were instrumental in enabling me to successfully complete the project on schedule.

Words are inadequate in offering thanks to the project assistants and all the employees in engine division and Sukhoi engine division who helped me in understanding the tech stacks used during the project work.

Finally, I am highly thankful to my family members and well-wishers who supported me and bestowed their good wishes for the successful completion of this project.

Abu Noor Al Saba

Vt No.- (25-442)

Computer Science & Engineering

Odisha University of Technology and

Research, Bhubaneswar

ACKNOWLEDGEMENT

I take immense pleasure in thanking **Mr. Basanta Kumar Nayak,**

Chief. Manager, Training And Development Institute, HAL for permitting me to carry out the internship at Hindustan Aeronautics Limited, Koraput.

I wish to express my deep sense of gratitude to my guide **SHRI SWADESH BEHERA ,CH. Manager IT Dept. (SED),** for his invaluable guidance and insightful suggestions, which were instrumental in enabling me to successfully complete the project on schedule.

Words are inadequate in offering thanks to the project assistants and all the employees in engine division and Sukhoi engine division who helped me in understanding the tech stacks used during the project work.

Finally, I am highly thankful to my family members and well-wishers who supported me and bestowed their good wishes for the successful completion of this project.

Aditya Dash

Vt No.-(25-441)

Computer Science & Engineering

Odisha University of Technology and

Research, Bhubaneswar

Hindustan Aeronautics Limited
Koraput Division, Odisha

GOVT.COLLEGE OF ENGINEERING,KALAHANDI BHAWANIPATNA

CERTIFICATE

This is to certify that Yubaraj Mohanty, pursuing **Bachelor of Technology** in **Government College of Engineering, Bhawanipatna has successfully undergone 4 weeks vocational training at HAL, Koraput** from 15 th JUNE 2025 to 12th JULY 2025.

He has completed the project on **"Smart Township (with Predictive Maintenance and Resident sentiment analysis)**" during his training period at HAL. During his training period he has been familiarized with the required techstacks required for developing a website.

We found him sincere, hardworking and his performance was excellent during the training period. His character and conduct was also found very good.

We wish her good luck in her future endeavor.

Shri Swadesh Behera	**Basanta Ku. Nayak**
Ch. Manager(IT) , SED	**Chief. Manager**
	Training & development Institute

Hindustan Aeronautics Limited
Koraput Division, Odisha

Hindustan Aeronautics Limited
Koraput Division, Odisha

CERTIFICATE

This is to certify that Priyanka Das, pursuing **Bachelor of Technology** in NIST **University, Berhampur** has successfully undergone **4 weeks** vocational training at **HAL, Koraput** from 15th JUNE 2025 to 12th JULY 2025.

She has completed the project on **"Smart Township (with Predictive Maintenance and Resident sentiment analysis)**" during her training period at HAL. During her training period she has been familiarized with the required techstacks required for developing a website.

We found her sincere, hardworking and her performance was excellent during the training period. Her character and conduct was also found very good.

We wish her good luck in her future endeavor.

Shri Swadesh Behera
Ch. Manager(IT) , SED

Basanta Ku. Nayak
Sr. Manager
Training & development Institute

Hindustan Aeronautics Limited
Koraput Division, Odisha

Hindustan Aeronautics Limited
Koraput Division, Odisha

CERTIFICATE

This is to certify that Prakriti Maharana, pursuing **Bachelor of Technology** in ACE**, Jaipur** has successfully undergone **4 weeks** vocational training at **HAL, Koraput** from 15th JUNE 2025 to 12th JULY 2025.

She has completed the project on **"Smart Township (with Predictive Maintenance and Resident sentiment analysis)** " during her training period at HAL. During her training period she has been familiarized with the required techstacks required for developing a website.

We found her sincere, hardworking and her performance was excellent during the training period. Her character and conduct was also found very good.

We wish her good luck in her future endeavor.

Shri Swadesh Behera
Ch. Manager(IT) , SED

Basanta Ku. Nayak
Sr. Manager
Training & development Institute

Hindustan Aeronautics Limited
Koraput Division, Odisha

Hindustan Aeronautics Limited
Koraput Division, Odisha

CERTIFICATE

This is to certify that Abu Noor Al Saba**,** pursuing **Bachelor of Technology** in Odisha University of Technology and research**, Bhubaneswar has successfully undergone 4 weeks vocational training at HAL, Koraput** from 15 th JUNE 2025 to 12th JULY 2025.
He has completed the project on **"Smart Township (with Predictive Maintenance and Resident sentiment analysis)**

" during his training period at HAL. During his training period he has been familiarized with the required techstacks required for developing a website.

We found him sincere, hardworking and his performance was excellent during the training period. His character and conduct was also found very good.

We wish her good luck in him future endeavor.

Shri Swadesh Behera
Ch. Manager(IT) , SED

Basanta Ku. Nayak
Sr. Manager
Training & development Institute

Hindustan Aeronautics Limited
Koraput Division, Odisha

Hindustan Aeronautics Limited
Koraput Division, Odisha

CERTIFICATE

This is to certify that Aditya Dash, pursuing **Bachelor of Technology** in OUTR**, Bhubaneswar has successfully undergone 4 weeks vocational training at HAL, Koraput** from 15 th JUNE 2025 to 12th JULY 2025.

He has completed the project on **"Smart Township (with Predictive Maintenance and Resident sentiment analysis)**

" during his training period at HAL. During his training period he has been familiarized with the required tech stacks required for developing a website.

We found him sincere, hardworking and his performance was excellent during the training period. His character and conduct was also found very good.

We wish her good luck in him future endeavor.

Shri Swadesh Behera
Ch. Manager(IT) , SED

Basanta Ku. Nayak
Sr. Manager
Training & development Institute

Contents

ABSTRACT

In the age of rapid urbanization, the need for sustainable, efficient, and intelligent township management has become critical. This project proposes a **Smart Township Web Platform** that leverages **Artificial Intelligence (AI)** and **Machine Learning (ML)** to optimize urban services and enhance the quality of life for residents.

The web-based platform serves as a central hub for township stakeholders—residents, authorities, and service providers. Through AI-driven features such as **predictive maintenance**, **smart energy management**, **automated waste management**, and **real-time security analytics**, the platform enables proactive governance and efficient resource utilization.

The web interface is designed to be user-friendly, responsive, and scalable. It integrates backend APIs with AI/ML modules to provide real-time data visualization and actionable insights. Technologies used include **Python (for AI/ML models)**, **Flask/Django for backend**, and **React/Angular for frontend**, with database support from **PostgreSQL or MongoDB**.

This smart township project demonstrates how AI and ML can transform urban living by making it more intelligent, responsive, and resident-centric.

Developed using modern web technologies like **React.js**, **Node.js/Python**, and **MongoDB/PostgreSQL**, the platform is responsive, scalable, and designed with user-friendliness in mind. It allows residents to report issues, access services, and receive updates, while providing administrators with real-time dashboards, analytics, and automation tools.

This project demonstrates how a smart township can leverage AI, ML & a web development to build a sustainable, safe, and connected community for the future.

Hindustan Aeronautics Limited
Koraput Division, Odisha

1.ABOUT HINDUSTAN AERONAUTICS LIMITED

Hindustan Aeronautics Limited (HAL) was established on October 1, 1964, through the merger of Hindustan Aircraft Limited, Aeronautics India Limited, and Aircraft Manufacturing Depot, Kanpur. The company's origins

date back to December 1940, when the visionary industrialist Seth Walchand Hirachand founded Hindustan Aircraft Limited in Bangalore in collaboration with the princely State of Mysore. The Government of India became a shareholder in March 1941 and assumed management control in 1942.

DIVISIONS OF HAL

- Bangalore Division
- Barrackpore Division
- Koraput Division
- Hyderabad Division
- Lucknow Division
- Kanpur Division
- Korwa Division

Manufactured 120 engines of the F2 series and overhaul 144 engines annually. Initially, this capacity was set for two lines of production but was increased to 160 engines in 1981-82 and further expanded to 256 engines in 1990-91.

Currently, this division manufactures RD33 engines for the MiG-29 aircraft. Additionally, it handles the annual overhaul of R25, R29B, and RD33 engines installed in MiG-21, MiG-27M, and MiG-29 aircraft, respectively.

Hindustan Aeronautics Limited (HAL), headquartered in Bangalore, India, is one of Asia's largest aerospace companies. Managed by the Indian Ministry of Defence, this state-owned enterprise specializes in the aerospace industry, encompassing the manufacturing and assembly of aircraft, navigation and communication equipment, and airport operations.

HAL made history by building South Asia's first military aircraft and continues to excel in the design, fabrication, and assembly of aircraft, jet engines, helicopters, and their components and spares. The company operates multiple facilities across India, including in Nasik, Korba, Kanpur, Koraput, Lucknow, Bangalore, and Hyderabad.

Notably, the HF-24 Marut fighter-bomber, the first fighter aircraft manufactured in India, was designed by German engineer Kurt Tank. With a rich history of collaboration, HAL has partnered with numerous international and domestic aerospace agencies, such as Airbus, Boeing, Sukhoi Aviation Corporation, Israel Aircraft Industries, RSK MIG, BAE Systems, Rolls-Royce, Dassault Aviation, Dornier Flugzeugwerke, the Indian Aeronautical Development Agency, and the Indian Space Research Organization.

HAL has successfully developed numerous R&D programs for both the Defense and Civil Aviation sectors. Some of its notable projects include:

- Advanced Light Helicopter - Weapon System Integration (ALH-WSI)
- Tejas - Light Combat Aircraft
- Intermediate Jet Trainer (IJT)
- Light Combat Helicopter (LCH)
- Various military and civil upgrades

These projects underscore HAL's substantial progress and enduring contributions to the aerospace industry.

SUKHOI ENGINE DIVISION KORAPUT (SED)

The **Sukhoi Engine Division** within Hindustan Aeronautics Limited (HAL), situated in Koraput, is a pivotal unit dedicated to the design, development, and maintenance of engines for Sukhoi fighter aircraft. Specializing in the production of engines tailored for Sukhoi aircraft models, this division plays a critical role in ensuring the operational readiness and performance excellence of these advanced fighter jets.

With state-of-the-art infrastructure and a skilled workforce, the Sukhoi Engine Division is committed to upholding the highest standards of quality and reliability in its engine manufacturing and overhaul activities. Its contributions are instrumental in supporting the Indian Air Force's fleet of Sukhoi aircraft and enhancing the country's defence capabilities.

As a key component of HAL's portfolio, the Sukhoi Engine Division exemplifies the organization's commitment to indigenous aerospace manufacturing and technological self-reliance. By delivering cutting-edge

Project

Smart Township

(with Predictive Maintenance and Resident sentiment analysis)

Introduction

In the modern age of digital transformation and smart urbanization, traditional townships face significant challenges related to infrastructure faults, resident dissatisfaction, and inefficient public service management. Most systems still rely on reactive methods, leading to delays, increased maintenance costs, and reduced quality of life.

This project introduces a **Smart Township Web Platform** that unifies **fault prediction** and **sentiment analytics** to provide administrators and residents with an intelligent, data-driven, and user-friendly solution. The aim is to proactively identify infrastructure failures and understand public sentiment, thus facilitating timely interventions and informed governance.

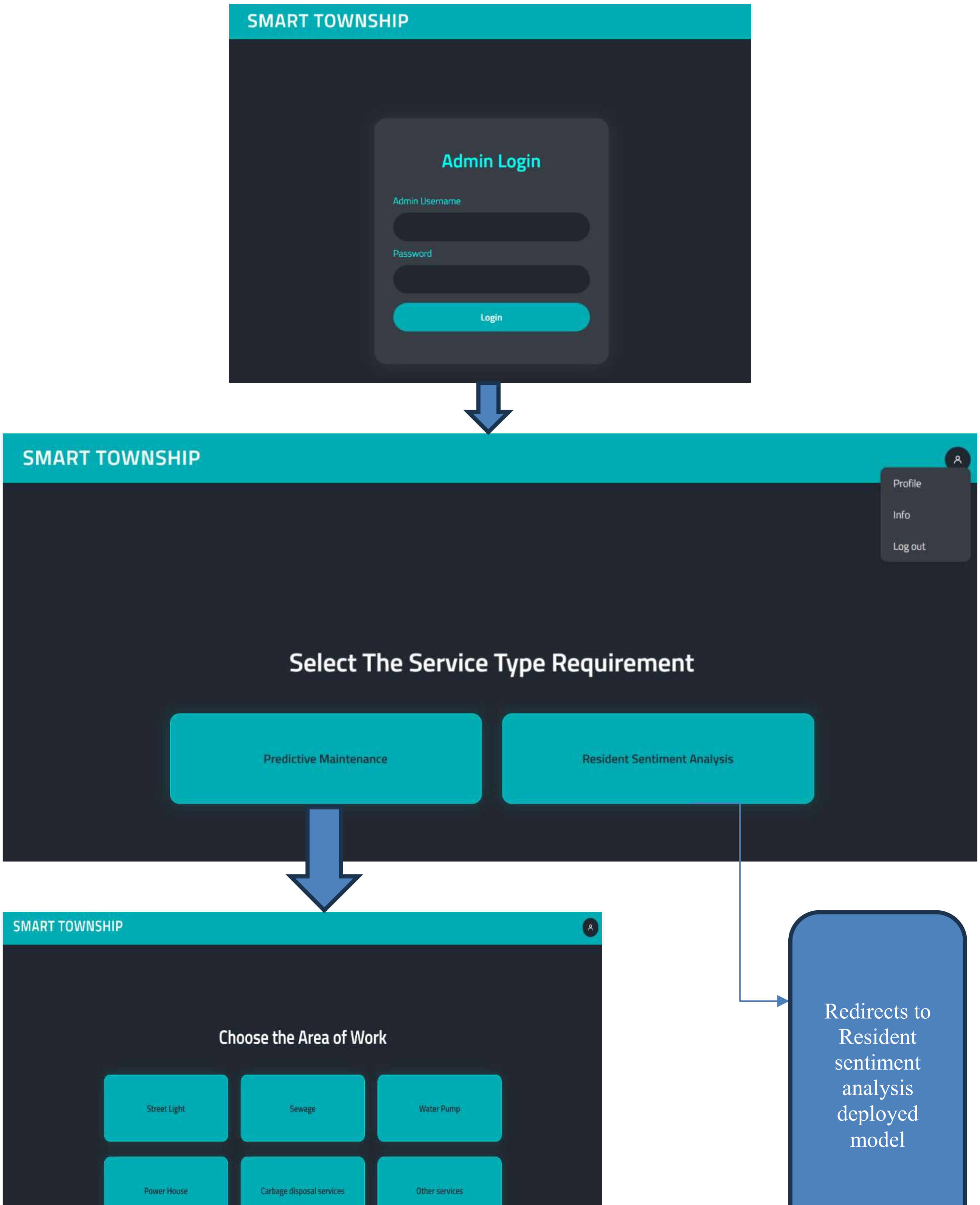
Project Overview

Combined System Components

Module	*Description*
1(a): Street Light Fault Prediction	**Predicts faults in street lights using environmental and electrical parameters with a machine learning model (Random Forest Classifier), helping reduce maintenance delays.**
1(b): Resident Sentiment Analysis	**Analyzes textual feedback from residents using NLP to classify sentiments (positive, neutral, negative), offering a sentiment heatmap across sectors.**

These modules are seamlessly integrated into a centralized dashboard where authorities can view fault predictions and citizen feedback visually and in real-time.

Visuals of Landing Page



Project – 1(A)
Resident Sentiment Analysis

Brief Introduction

This project focuses on creating a dashboard that analyzes resident feedback data to determine overall public sentiment.
The goal is to visualize sector-wise sentiments to assist authorities in decision-making and issue resolution.

By collecting responses via Google Forms and processing them with sentiment analysis, we aim to provide real-time, actionable insights through an interactive dashboard.

Literary Overview/Background:

Sentiment analysis has been widely used in social media, customer feedback systems, and product review platforms.
Several municipal and civic bodies are adopting digital tools for public engagement. This project draws inspiration from dashboards like Twitter Sentiment Analysis, and open governance projects, adapting those ideas for city feedback management.

Tools and Technologies

TechStack	Usecase
Python	for scripting and analysis
Streamlit	for dashboard creation
Pandas	for data processing
TextBlob	for sentiment analysis
Folium	for mapping
Google Forms & Sheets	for data collection
Hugging Face Spaces	for deployment
Git & GitHub	for version control

System Design / Architecture

The system consists of three major components:

1. Data Collection via Google Forms → Sheets
2. Sentiment Analysis using `TextBlob`
3. Visualization Dashboard with Streamlit

The workflow is linear and follows:
Form → Sheet → Preprocess → Sentiment Label → Visual Dashboard → Deployment

Project Directory Structure:
```
resident-sentiment-dashboard/
│
├── app.py
├── sentiment_analysis.py
├── sync_feedback.py
├── processed_feedback.csv
├── sector_coords.csv
├── requirements.txt
└── README.md
```

Implementation

Work flow

Step 1: **Data Collection**

- Source: Google Forms
- Storage: Linked Google Sheet
- Residents submit feedback with:
    - Their sector/block
    - Feedback text
    - Optional rating or complaint category

Step 2: **Sync Data**

- You create a Python script (sync_feedback.py) that:
    - Uses Google Sheets API to fetch responses
    - Stores them in a local CSV (feedback.csv)
    - You run this periodically or via a refresh button on Streamlit

Step 3: **Sentiment Analysis**

- Script (sentiment_analysis.py) reads the feedback.csv

- Using `TextBlob` to compute polarity and assign:
    - Polarity > 0.1 → Satisfied
    - -0.1 < Polarity < 0.1 → Neutral
    - Polarity < -0.1 → Frustrated

Step 4: **Dashboard Visualization**

- Streamlit app (app.py or dashboard.py) does:
    - Loads processed data
    - Displays charts using:
        - matplotlib, seaborn → bar graphs, heatmaps
        - folium → interactive map with colored sentiment markers
        - wordcloud → common frustrated words
    - CSV download & real-time refresh supported

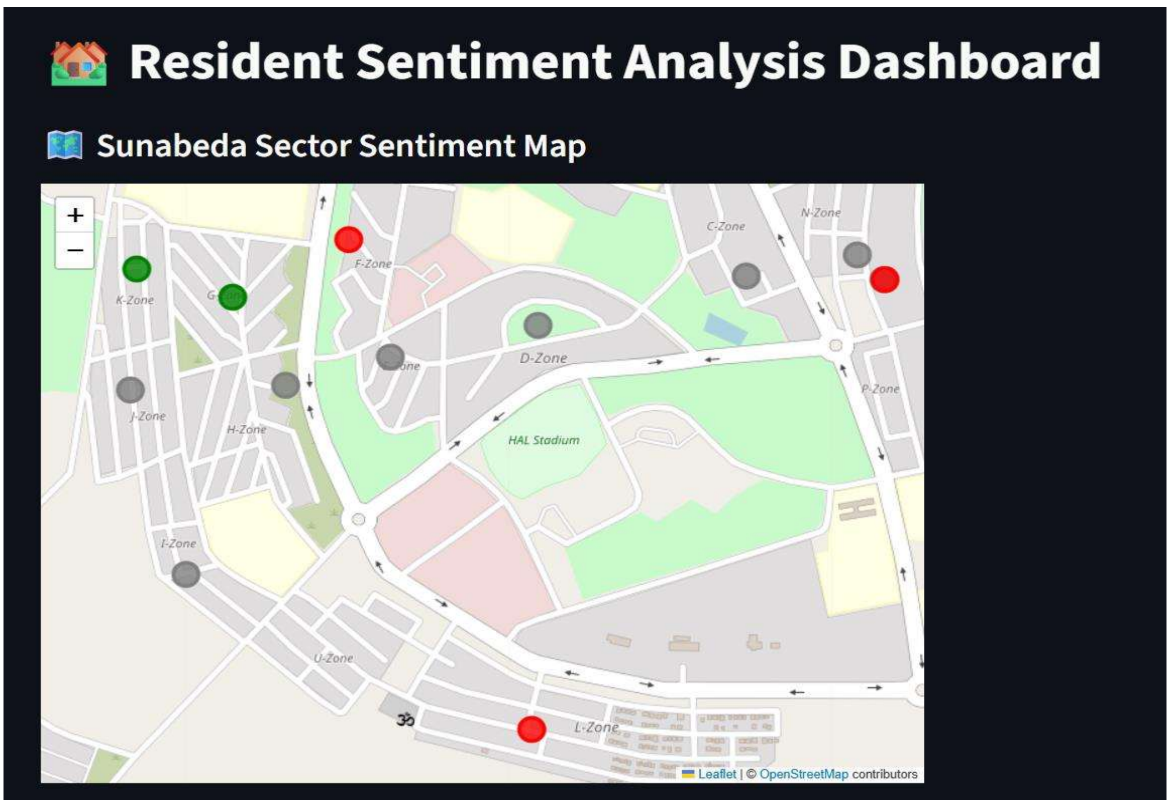
Step 5: **Deployment**

- You deploy the app on Hugging Face Spaces

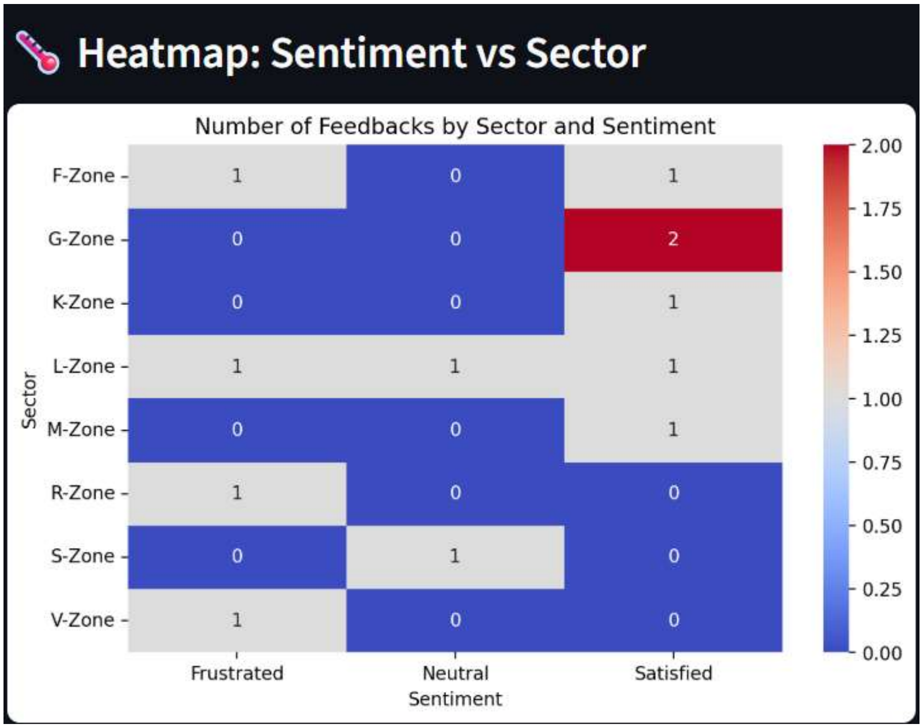
Results and Visuals

- The dashboard was successfully deployed.

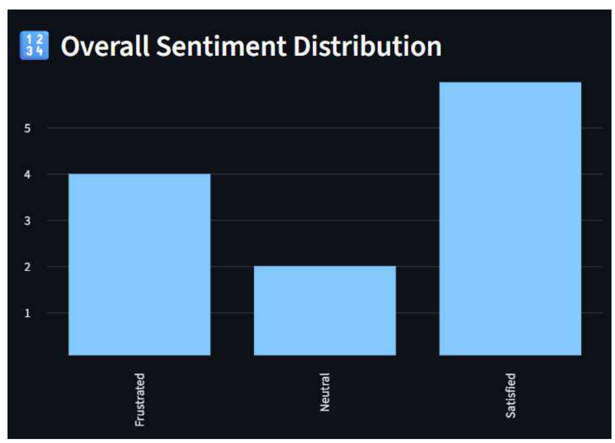
- Sector-wise color-coded sentiment map helps visualize concentration of frustrated responses.

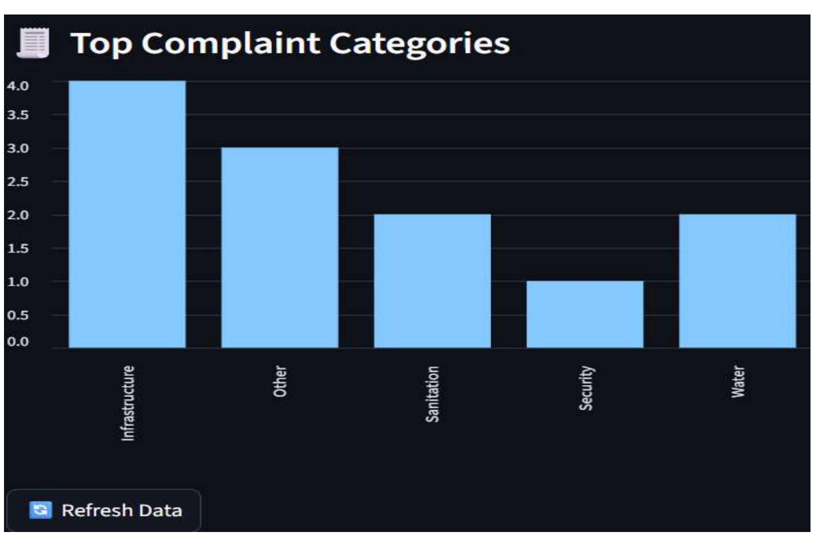


- Word cloud shows frequent complaint terms.



Sect	Frustrated	Neutral	Satisfied
F-Zone	1	0	1
G-Zone	0	0	2
K-Zone	0	0	1
L-Zone	1	1	1
M-Zone	0	0	1
R-Zone	1	0	0
S-Zone	0	1	0
V-Zone	1	0	0





📄 Full Feedback Dataset

		Full Name	Sector	Feedback	Date	Sentiment	Category
0	1:07:17	Shyama	G-Zone	The Park is clean and the security is good.	28/06/2025	Satisfied	Security
1	1:08:47	Sailesh	L-Zone	Water supply is irregular and often dirty.	29/06/2025	Frustrated	Water
2	1:47:16	Mukesh pradhan	K-Zone	Electricity supply is stable and we are happy with the service overall	27/06/2025	Satisfied	Water
3	1:47:51	Joy	F-Zone	Everything is fine. Very peaceful neighbourhood.	25/06/2025	Satisfied	Other
4	1:48:04	Santosh nayak	F-Zone	Street flooding after every rainfall. Really frustrating.	19/06/2025	Frustrated	Infrastructure
5	1:49:13	Akash	V-Zone	Garbage collection is delayed for the past week.	23/06/2025	Frustrated	Sanitation
6	1:51:04	Seeta kumari	M-Zone	Too much noise from nearby construction sites.	20/06/2025	Satisfied	Infrastructure
7	1:51:44	Nishant	L-Zone	Streetlight don't work at night in our block.	30/06/2025	Neutral	Infrastructure
8	1:51:45	Anil kumar nayak	L-Zone	Love the new community center! Very useful.	27/06/2025	Satisfied	Other
9	1:52:43	Lalua	G-Zone	Too much noise from nearby construction site.	26/06/2025	Satisfied	Infrastructure

📁 Download Feedback Data

⬇️ Download as CSV

Project – 1(B)
Predictive Maintenance

Brief Introduction

In rapidly urbanizing areas, the infrastructure of townships—including streetlights, sewage systems, water distribution, and public utilities—plays a critical role in ensuring livability and safety. However, the failure of such infrastructure due to poor maintenance or reactive responses often results in inconvenience, hazards, and increased costs.

This project proposes the use of **Artificial Intelligence (AI)** to implement **Predictive Maintenance (PdM)** in township infrastructure systems. Predictive maintenance is a proactive approach where faults and failures are anticipated before they occur, enabling timely interventions, reducing downtime, and optimizing maintenance resources.

Scope:

The pilot implementation of this system is designed for core township infrastructure components, especially **streetlight fault prediction**, with the potential to scale across:

- Drainage and sewer networks
- Public utilities and meters
- Water Pump
- Power House
- Other Services

Streetlight Fault Prediction

Now this is one of the many predictive maintenance services. The Model can be used for all other mentioned issues.
We have Highlighted this particular issue as an example and trained the model according to this dataset only.

Problem Statement Overview

The current approach to street light maintenance often relies on reactive measures, addressing faults only after they have occurred. This leads to delayed repairs, prolonged periods of reduced lighting, and potentially higher costs associated with emergency fixes.

The problem is to develop a predictive system that can analyze real-time or historical data from street lights to anticipate faults before they happen.

This would enable a transition from reactive to proactive maintenance, optimizing resources and improving the safety and functionality of the street lighting infrastructure.

Tools and Technologies

TechStack	Usecase
pandas	For data manipulation and analysis.
numpy	For numerical operations.
Matplotlib and seaborn	For data visualization.
Scikit-learn	For machine learning model development (Random Forest Classifier),
ipywidgets	For creating interactive elements in the notebook
joblib	For saving and loading the trained machine learning model.
kagglehub	For downloading the dataset
Git & GitHub	For version control

Implementation

Workflow

The project implementation follows these steps:

1. **Data Loading and Exploration:**
   - Download the dataset using kagglehub.
   - Load the data into a pandas DataFrame.
   - Explore the data using methods like head(), describe(), and check for data types and missing values.
   - Convert the timestamp column to datetime objects.

2. **Data Preprocessing and Feature Engineering:**
   - Handle categorical features (environmental_conditions) using one-hot encoding (pd.get_dummies).
   - Ensure numerical features are in the correct format, handling potential non-numeric characters if necessary.
   - Define the features (independent variables) and the target variable (fault_type).

3. **Exploratory Data Analysis (EDA):**
   - Visualize the correlation matrix to understand the relationships between features.
   - Analyze the distribution of the target variable (fault_type) using a count plot.

4. **Model Selection and Training:**
   - Split the dataset into training and testing sets using train_test_split.
   - Choose a suitable classification model. The Random Forest Classifier is used in this project due to its robustness and ability to handle various types of data.
   - Train the Random Forest model on the training data.

5. **Model Evaluation:**
   - Evaluate the trained model's performance on the test data using metrics like accuracy, confusion matrix, and classification report.

6. **Prediction with Manual Input:**
   - Demonstrate how to make predictions using manually provided input values to test the model with specific scenarios.

7. **Interactive Prediction Interface:**
   - Create an interactive interface using ipywidgets to allow users to easily input values and get predictions without modifying the code.
   - Map the numerical fault_type predictions to descriptive labels for better understanding.

8. **Model Saving:**
   - Save the trained model using joblib for future use without retraining.

About the Dataset

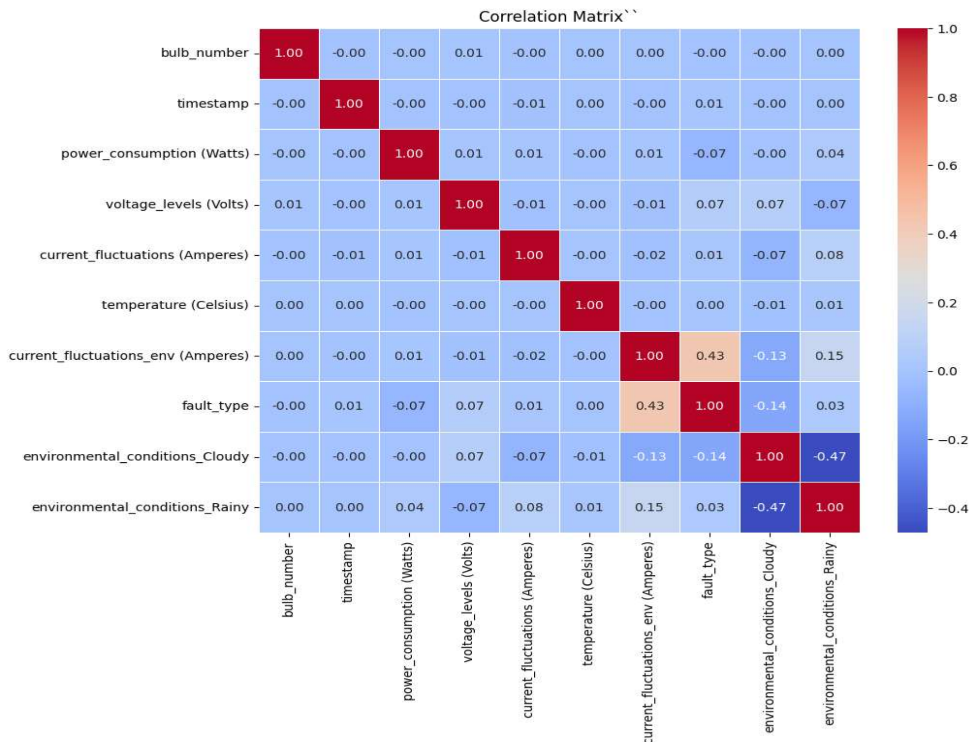
The dataset used in this project is the "**Street Light Fault Prediction Dataset**" available on Kaggle. It contains various attributes related to street lights, including:

- **bulb_number**: Unique identifier for each street light.
-
- **timestamp**: The date and time of the recorded data.
-
- **power_consumption** (Watts): The power consumption of the light.
- **voltage_levels** (Volts): The voltage supplied to the light.
- **current_fluctuations** (Amperes): Fluctuations in the current.
- **temperature** (Celsius): The ambient temperature.
- **environmental**_conditions: Weather conditions (e.g., Clear, Cloudy, Rainy).
- **current_fluctuations_env** (Amperes): Current fluctuations influenced by environmental conditions.
- **fault_type**: The type of fault observed (the target variable).

The fault_type column is a categorical variable representing different types of faults. The dataset provides a basis for training a supervised learning model to predict this target variable.

Results

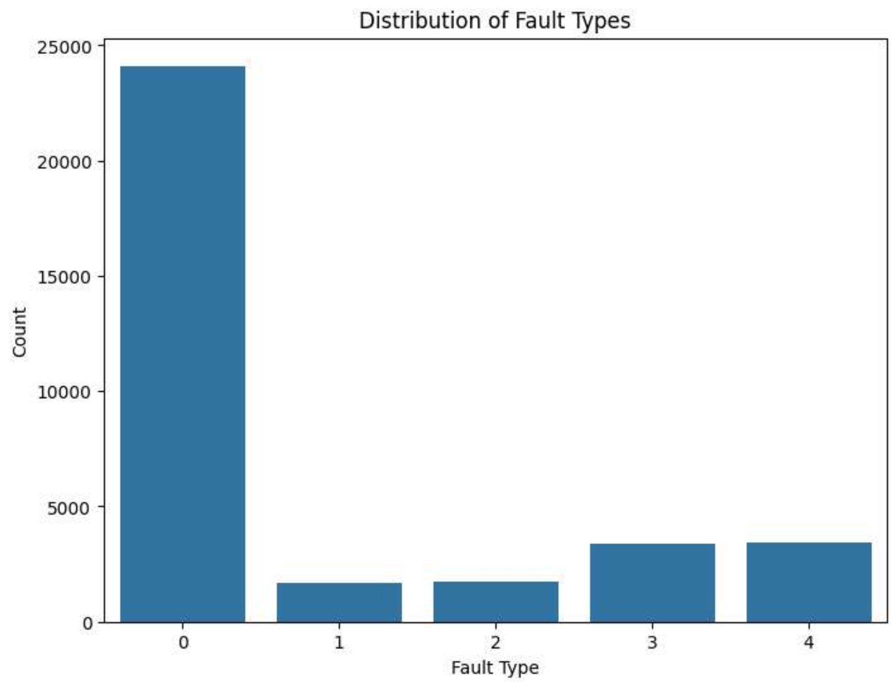
The Random Forest Classifier model achieved an accuracy of approximately 85% on the test dataset.



The confusion matrix and classification report provide a more detailed understanding of the model's performance for each fault type:

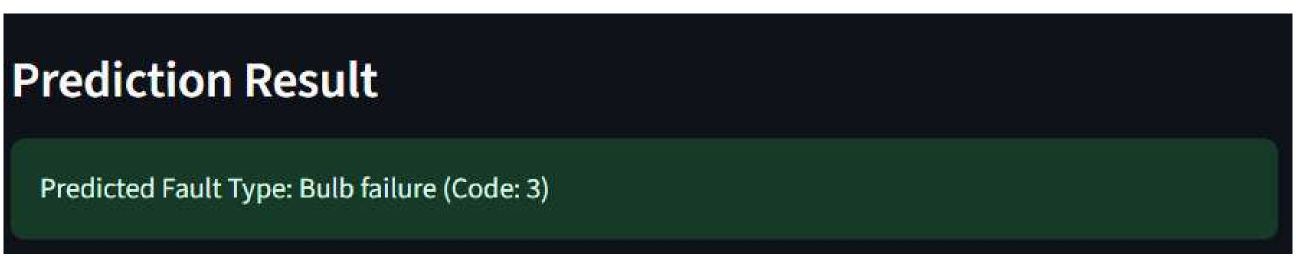
```
Accuracy: 0.85

Confusion Matrix:
[[4844    0    0   17    2]
 [  78  235    0    0    0]
 [ 160    0  154    1    0]
 [ 571    0    0   92    0]
 [ 220    0    0    0  488]]

Classification Report:
              precision    recall  f1-score   support

           0       0.82      1.00      0.90      4863
           1       1.00      0.75      0.86       313
           2       1.00      0.49      0.66       315
           3       0.84      0.14      0.24       663
           4       1.00      0.69      0.81       708

    accuracy                           0.85      6862
   macro avg       0.93      0.61      0.69      6862
weighted avg       0.86      0.85      0.82      6862
```

- The model performs very well in predicting "No fault" (Fault Type 0), with high precision and recall.
- It also shows good performance for "Short circuit" (Fault Type 1) and "Light Flickering" (Fault Type 4).
- The performance for "Voltage Surge" (Fault Type 2) and "Bulb failure" (Fault Type 3) is lower, particularly in terms of recall for Fault Type 3. This suggests that the model is less effective at identifying all instances of these fault types.



Visuals


Street Light Fault Prediction App
Enter Input Data
power_consumption (Watts)
40.00
voltage_levels (Volts)
220.00
current_fluctuations (Amperes)
4.50
temperature (Celsius)
37.00
current_fluctuations_env (Amperes)
2.30
Environmental Conditions
Cloudy
Predict Fault Type


Prediction Result
Predicted Fault Type: Bulb failure (Code: 3)

Project Smart Township

Landing Page

/code

Admin_login.html

```
 <!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Admin Login - Smart Township HAL</title>
  <link
href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&family=Titillium+Web:wght@300;60
0&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="style.css">
</head>
<body class="login-page">
  <nav class="navbar">
    <span class="navbar-brand">SMART TOWNSHIP</span>
  </nav>
  <div class="main-center">
    <div class="container login-container">
      <h1>Admin Login</h1>
      <form onsubmit= "event.preventDefault(); window.location.href='dashboard.html'; return false;">
        <label style="display:block; text-align:left; margin-bottom:0.25rem; color:var(--color-accent-bright);
font-weight:500;">Admin Username</label>
        <input type="text" placeholder="" required />
        <label style="display:block; text-align:left; margin-bottom:0.25rem; color:var(--color-accent-bright);
font-weight:500;">Password</label>
        <input type="password" placeholder="" required />
        <button type="submit">Login</button>
      </form>
    </div>
  </div>
</body>
</html>
```

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Smart Township HAL - Login</title>
```

```
  <link
href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&family=Titillium+Web:wght@300;60
0&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="style.css">
</head>
<body class="login-page">
  <nav class="navbar">
    <span class="navbar-brand">SMART TOWNSHIP</span>
  </nav>
  <div class="main-center">
    <div class="container login-container">
    <h1>Welcome to Smart Township</h1>
    <form onsubmit="event.preventDefault(); window.location.href='admin-login.html';">
    <img src="https://sdscraneservices.in/wp-content/uploads/sites/7/2021/08/unnamed.jpg" alt="Hindustan
aeronautics limited logo" style="width: 300px; height: auto; margin-bottom: 1rem; border-radius: 10px;">
    <div style="margin:1.5rem 0 0.5rem 0; color:#aaa; font-size:1rem;">Login as an admin?</div>
    <button
  class="admin-link"
  style="width:100%; background:var(--color-accent); color:var(--color-text-main); border-radius:25px;
padding:0.75rem; font-weight:bold; margin:0;"
  onclick="window.location.href='admin-login.html';"
  >
  Admin login
</button>
  </div>
  <script src="https://cdn.jsdelivr.net/npm/particles.js"></script>
  <script src="main.js"></script>
</body>
</html>
```

Dashboard.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Smart Township HAL - Dashboard</title>
  <link
href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&family=Titillium+Web:wght@300;60
0&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="style.css" />
</head>
<body class="dashboard-page">
  <nav class="navbar">
    <span class="navbar-brand">SMART TOWNSHIP</span>
    <div class="profile-menu" id="profileMenu">
      <button class="profile-icon" id="profileIcon" aria-label="Profile">
        <svg width="24" height="24" fill="none" stroke="currentColor" stroke-width="2" stroke-
linecap="round" stroke-linejoin="round" viewBox="0 0 24 24">
          <circle cx="12" cy="8" r="4"/>
          <path d="M4 20c0-4 4-7 8-7s8 3 8 7"/>
        </svg>
```

```
      <div class="dropdown" id="profileDropdown">
        <a href="#">Profile</a>
        <a href="#">Info</a>
        <a href="#" id="logoutBtn">Log out</a>
      </div>
    </div>
  </nav>
  <div class="bg-overlay"></div>
  <div class="full-height center-content container">
    <h2>Select The Service Type Requirement</h2>
    <div class="card-grid large">
      <a href="faults.html" class="card futuristic-glow">Predictive Maintenance</a>
      <a href="https://huggingface.co/spaces/ana-saba/resident-sentiment-analysis" class="card futuristic-
glow">Resident Sentiment Analysis</a>
    </div>
  </div>
  <script>
    const profileMenu = document.getElementById('profileMenu');
    document.addEventListener('click', function(e) {
      if (profileMenu.contains(e.target)) {
        profileMenu.classList.toggle('open');
      } else {
        profileMenu.classList.remove('open');
      }
    });
    document.getElementById('logoutBtn').addEventListener('click', function(e) {
    e.preventDefault();
    window.location.href = "index.html";
  });
  </script>
</body>
</html>
```

Faults.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Smart Township HAL - Fault Prediction</title>
  <link
href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&family=Titillium+Web:wght@300;60
0&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="style.css" />
</head>

<body class="faults-page">
  <nav class="navbar">
    <span class="navbar-brand">SMART TOWNSHIP</span>
    <div class="profile-menu" id="profileMenu">
      <button class="profile-icon" id="profileIcon" aria-label="Profile">
        <svg width="24" height="24" fill="none" stroke="currentColor" stroke-width="2" stroke-
```

```html
          <circle cx="12" cy="8" r="4"/>
          <path d="M4 20c0-4 4-7 8-7s8 3 8 7"/>
        </svg>
      </button>
      <div class="dropdown" id="profileDropdown">
        <a href="#">Profile</a>
        <a href="#">Info</a>
        <a href="#" id="logoutBtn">Log out</a>
      </div>
    </div>
  </nav>
  <div class="bg-overlay"></div>
  <div class="full-height center-content container">
    <h2>Choose the Area of Work</h2>
    <div class="card-grid large">
      <a href="https://huggingface.co/spaces/ana-saba/StreetLight-Predictive-Maintenance" class="card futuristic-glow" target="_blank">Street Light</a>
      <a href="#" class="card futuristic-glow" onclick="alert('Sorry! This service is not yet available')">Sewage</a>
      <a href="#" class="card futuristic-glow" onclick="alert('Sorry! This service is not yet available')">Water Pump</a>
      <a href="#" class="card futuristic-glow" onclick="alert('Sorry! This service is not yet available')">Power House</a>
      <a href="#" class="card futuristic-glow" onclick="alert('Sorry! This service is not yet available')">Carbage disposal services</a>
      <a href="#" class="card futuristic-glow" onclick="alert('Sorry! This service is not yet available')">Other services</a>
    </div>
  </div>
    <script>
    const profileMenu = document.getElementById('profileMenu');
    document.addEventListener('click', function(e) {
      if (profileMenu.contains(e.target)) {
        profileMenu.classList.toggle('open');
      } else {
        profileMenu.classList.remove('open');
      }
    });
    document.getElementById('logoutBtn').addEventListener('click', function(e) {
    e.preventDefault();
    window.location.href = "index.html";
  });
    </script>
</body>
</html>
```

Main.js

```javascript
// main.js
particlesJS("particles-js", {
  particles: {
    number: {
```

```
      density: { enable: true, value_area: 800 }
    },
    color: { value: "#00fff7" },
    shape: { type: "circle" },
    opacity: {
      value: 0.5,
      random: false
    },
    size: {
      value: 3,
      random: true
    },
    line_linked: {
      enable: true,
      distance: 150,
      color: "#00fff7",
      opacity: 0.4,
      width: 1
    },
    move: {
      enable: true,
      speed: 2,
      direction: "none",
      straight: false,
      out_mode: "out"
    }
  },
  interactivity: {
    detect_on: "canvas",
    events: {
      onhover: {
        enable: true,
        mode: "grab"
      },
      onclick: {
        enable: true,
        mode: "push"
      },
      resize: true
    },
    modes: {
      grab: {
        distance: 200,
        line_linked: {
          opacity: 0.6
        }
      },
      push: {
        particles_nb: 4
      }
    }
  },
  retina_detect: true
});
```

Style.css

```
:root {
  --color-bg-main: #222831;      /* rgb(34, 40, 49) */
  --color-bg-secondary: #393e46; /* rgb(57, 62, 70) */
  --color-accent: #00adb5;       /* rgb(0, 173, 181) */
  --color-accent-bright: #00fff5;/* rgb(0, 255, 245) */
  --color-text-main: #ffffff;
  --color-text-secondary: #7eeaff;
}

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  font-family: 'Titillium Web', 'Orbitron', sans-serif;
  color: #ffffff;
}

body, body.login-page {
  background: var(--color-bg-main);
  color: var(--color-text-main);
  font-family: 'Titillium Web', 'Orbitron', sans-serif;
  margin: 0;
  min-height: 100vh;
}

.container {
  position: relative;
  z-index: 2;
  padding: 2rem;
  max-width: 1200px;
  margin: auto;
  text-align: center;
}

.full-height {
  height: 100vh;
  display: flex;
  flex-direction: column;
  justify-content: center;
}

.center-content {
  display: flex;
  flex-direction: column;
  align-items: center;
}

/* HEADINGS */
h1{
  color: var(--color-accent-bright);
  font-size: 2rem;
  margin-bottom: 2rem;
  font-weight: 600;

}

.login-container {
  background: var(--color-bg-secondary);
  box-shadow: 0 0 25px rgba(0, 255, 255, 0.08);
  padding: 3rem 2rem;
  border-radius: 20px;
  max-width: 400px;
  margin: auto;
  text-align: center;
```

```css
  z-index: 2;
}

input, button {
  display: block;
  width: 100%;
  padding: 0.75rem;
  margin: 0.5rem 0;
  border-radius: 25px;
  border: none;
  background: var(--color-bg-main);
  color: var(--color-accent-bright);
  font-size: 1rem;
  font-family: inherit;
  outline: none;
  box-sizing: border-box;
}

input::placeholder {
  color: var(--color-text-secondary);
  opacity: 1;
}

button {
  background: var(--color-accent);
  color: var(--color-text-main);
  font-weight: bold;
  cursor: pointer;
  transition: all 0.3s;
  margin-top: 1rem;
}
/* Add to style.css */
.navbar {
  width: 100vw;
  background: var(--color-accent);
  padding: 0.5rem 2rem;
  box-sizing: border-box;
  display: flex;
  align-items: center;
}

.navbar-brand {
  color: #fff;
  font-size: 2rem;
  font-weight: bold;
  letter-spacing:1px;
}
body.login-page {
  display: flex;
  flex-direction: column;
  min-height: 100vh;
}

body.login-page > .navbar {
  flex-shrink: 0;
}

body.login-page .container.login-container {
  flex: 1 0 auto;
  display: flex;
  flex-direction: column;
  justify-content: center;
  min-height: 0;
  margin: auto;
  position: relative;
  z-index: 2;
}
```

```css
.main-center {
  min-height: calc(100vh - 64px); /* Adjust 64px to your navbar height */
  display: flex;
  align-items: center;
  justify-content: center;
}

button:hover {
  background: var(--color-accent-bright);
  color: var(--color-bg-main);
  transform: scale(1.05);
  box-shadow: 0 0 10px var(--color-accent-bright);
}

.admin-link {
  display: inline-block;
  margin-top: 1rem;
  font-size: 1rem;
  color: var(--color-accent);
  text-decoration: underline;
  background: none;
  border: none;
  cursor: pointer;
}

.card-grid {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(240px, 1fr));
  gap: 2rem;
  width: 100%;
  max-width: 1000px;
}
.card, .login-container {
  border: none;
  box-shadow: 0 0 25px rgba(235, 236, 236, 0.08);
}

.card-grid.large .card {
  min-height: 140px;
  font-size: 1.2rem;
  padding: 2rem;
  color: #222831;
  background-color: #00adb5;

}

.card {
  background: rgba(0, 255, 255, 0.05);
  border: 1px solid #00fff7;
  padding: 2rem;
  border-radius: 15px;
  box-shadow: 0 0 20px rgba(0, 255, 255, 0.15);
  transition: all 0.3s ease;
  text-decoration: none;
  color: #ffffff;
  display: flex;
  justify-content: center;
  align-items: center;
  font-weight: 600;
  text-align: center;
}

.card:hover {
  transform: scale(1.05);
  box-shadow: 0 0 25px rgba(0, 255, 255, 0.4);
  background: rgba(0, 255, 255, 0.08);
```

```css
h2{
  margin: 50px;
  font-size: 40px;
}

.profile-menu {
  position: relative;
  margin-left: auto;
  display: flex;
  align-items: center;
}

.profile-icon {
  width: 40px;
  height: 40px;
  border-radius: 50%;
  background: var(--color-bg-main);
  display: flex;
  align-items: center;
  justify-content: center;
  color: var(--color-bg-main);
  font-size: 1.5rem;
  cursor: pointer;
  border: none;
  outline: none;
  transition: box-shadow 0.2s;
}

.profile-icon:hover {
  box-shadow: 0 0 0 3px var(--color-accent);
}

.dropdown {
  display: none;
  position: absolute;
  right: 0;
  top: 48px;
  background: var(--color-bg-secondary);
  box-shadow: 0 2px 8px rgba(0,0,0,0.15);
  border-radius: 10px;
  min-width: 140px;
  z-index: 10;
}

.dropdown a {
  display: block;
  padding: 0.75rem 1.25rem;
  color: var(--color-text-main);
  text-decoration: none;
  font-size: 1rem;
  border-radius: 10px;
  transition: background 0.2s;
}

.dropdown a:hover {
  background: var(--color-accent);
  color: #fff;
}

.profile-menu.open .dropdown {
  display: block;
}
```

Project 1(B)

/code

Frontend(Using Streamlit)

```
# app.py

import streamlit as st
import pandas as pd
import joblib

# Load the trained model
rf_model = joblib.load('model.pkl')

# Features based on the model
numerical_features = [
    'power_consumption (Watts)',
    'voltage_levels (Volts)',
    'current_fluctuations (Amperes)',
    'temperature (Celsius)',
    'current_fluctuations_env (Amperes)'
]

environmental_features = [
    'environmental_conditions_Cloudy',
    'environmental_conditions_Rainy'
]

all_features = numerical_features + environmental_features

# Fault type mapping
fault_type_mapping = {
    0: 'No fault',
    1: 'Short circuit',
    2: 'Voltage Surge',
    3: 'Bulb failure',
    4: 'Light Flickering'
}

# Streamlit UI
st.set_page_config(page_title="Street Light Fault Predictor")
st.title("Street Light Fault Prediction App")

st.markdown("### Enter Input Data")

with st.form(key='prediction_form'):
    user_input = {}
    for feature in numerical_features:
        user_input[feature] = st.number_input(f"{feature}", value=0.0)

    env_condition = st.selectbox("Environmental Conditions", ['Clear', 'Cloudy', 'Rainy'])

    submit_button = st.form_submit_button(label='Predict Fault Type')

if submit_button:
    # Encode environmental condition as one-hot
    user_input['environmental_conditions_Cloudy'] = 1 if env_condition == 'Cloudy' else 0
    user_input['environmental_conditions_Rainy'] = 1 if env_condition == 'Rainy' else 0

    # Convert to DataFrame and reorder columns
```

```
    # Prediction
    pred_fault_num = rf_model.predict(input_df)[0]
    pred_fault_desc = fault_type_mapping.get(pred_fault_num, "Unknown Fault Type")

    # Display result
    st.markdown("### Prediction Result")
    st.success(f"Predicted Fault Type: {pred_fault_desc} (Code: {pred_fault_num})")
```

Backend

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix


import os
for dirname, _, filenames in os.walk('/kaggle/input'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

df.describe()

df1 = pd.get_dummies(df, columns=['environmental_conditions'], drop_first=True)
correlation_matrix = df1.corr()
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm', fmt=".2f", linewidths=.5)
plt.title("Correlation Matrix``")
plt.show()

plt.figure(figsize=(8, 6))
sns.countplot(x='fault_type', data=df)
plt.title("Distribution of Fault Types")
plt.xlabel("Fault Type")
plt.ylabel("Count")
plt.show()

import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report
features = ['power_consumption (Watts)', 'voltage_levels (Volts)', 'current_fluctuations (Amperes)',
'temperature (Celsius)', 'current_fluctuations_env (Amperes)']
target = 'fault_type'

for feature in features:
    df[feature] = pd.to_numeric(df[feature].astype(str).replace('[^0-9.]', '', regex=True), errors='coerce')

df = pd.get_dummies(df, columns=['environmental_conditions'], drop_first=True)

X_train, X_test, y_train, y_test = train_test_split(df[features +
list(df.columns[df.columns.str.startswith('environmental_conditions')])], df[target], test_size=0.2,
random_state=42)

rf_model = RandomForestClassifier(random_state=42)
```

```
y_pred = rf_model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)
conf_matrix = confusion_matrix(y_test, y_pred)
class_report = classification_report(y_test, y_pred)

print(f"Accuracy: {accuracy:.2f}")
print("\nConfusion Matrix:")
print(conf_matrix)
print("\nClassification Report:")
print(class_report)

print("Feature names during model training:")
print(rf_model.feature_names_in_)

manual_values = {
    'power_consumption (Watts)': 48.423,
    'voltage_levels (Volts)': 218.93,
    'current_fluctuations (Amperes)': 3.01,
    'temperature (Celsius)': 28.34,
    'current_fluctuations_env (Amperes)': 1.95,
    'environmental_conditions_Cloudy': False,
    'environmental_conditions_Rainy': False
}

manual_df = pd.DataFrame([manual_values])

for feature in features:
    manual_df[feature] = pd.to_numeric(manual_df[feature], errors='coerce')

manual_pred = rf_model.predict(manual_df)

print("Manual Prediction:")
print("Predicted Fault Type:", manual_pred[0])

import joblib

# If scikit-learn, XGBoost, RandomForest etc.
joblib.dump(rf_model, 'model.pkl')
```

Project 1(A)

/code

Frontend(Using Streamlit)

Dashboard.py

```
import streamlit as st
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from wordcloud import WordCloud
import seaborn as sns
import matplotlib.pyplot as plt
import folium
from streamlit_folium import folium_static



# Page settings
st.set_page_config(layout="wide")
st.title("🏘 Resident Sentiment Analysis Dashboard")

# Load the processed data
df = pd.read_csv("processed_feedback.csv")

# Load coordinates from sector_coords.csv
coord_df = pd.read_csv("sector_coords.csv")

# Convert to dictionary format
sector_coords = {
   row["Sector"]: [row["Latitude"], row["Longitude"]]
   for _, row in coord_df.iterrows()
}

st.subheader("🗺 Sunabeda Sector Sentiment Map")

# Creating the base map centered on Sunabeda
m = folium.Map(location=[18.695, 82.855], zoom_start=14)

# Adding a marker for each sector based on sentiment
for sector, coords in sector_coords.items():
   # Filter feedback for this sector
   sector_df = df[df["Sector"] == sector]

   if len(sector_df) == 0:
      sentiment = "No Data"
      color = "gray"
   else:
      # Getting the most common sentiment in that sector
      sentiment = sector_df["Sentiment"].mode()[0]
      color = {
         "Satisfied": "green",
```

```
        }.get(sentiment, "gray")

    # Adding circle marker
    folium.CircleMarker(
        location=coords,
        radius=10,
        color=color,
        fill=True,
        fill_opacity=0.8,
        popup=f"{sector}: {sentiment}"
    ).add_to(m)

# Showing the map in Streamlit
folium_static(m)



st.subheader("🌡 Heatmap: Sentiment vs Sector")



# Pivot data
heatmap_data = df.pivot_table(index="Sector", columns="Sentiment", aggfunc="size", fill_value=0)



# Plot using seaborn
fig, ax = plt.subplots(figsize=(8, 5))
sns.heatmap(heatmap_data, annot=True, fmt="d", cmap="coolwarm", ax=ax)
plt.title("Number of Feedbacks by Sector and Sentiment")

st.pyplot(fig)


# Section: Sentiment Distribution
st.subheader("🔢 Overall Sentiment Distribution")
sentiment_count = df["Sentiment"].value_counts()
st.bar_chart(sentiment_count)

# Section: Sector-wise Sentiment Breakdown
st.subheader("📍 Sector-wise Sentiment")
sector_sentiment = pd.crosstab(df["Sector"], df["Sentiment"])
st.dataframe(sector_sentiment)

# Section: Word Cloud for Frustrated Feedback
st.subheader("☁ Common Complaints (from Frustrated Feedback)")
frustrated_text = " ".join(df[df["Sentiment"] == "Frustrated"]["Feedback"])
if frustrated_text:
    wordcloud = WordCloud(width=800, height=400, background_color="white").generate(frustrated_text)
    st.image(wordcloud.to_array())
else:
    st.info("No frustrated feedback yet 😊")

# Section: Raw Feedback Table
```

```
# CSV Download
st.subheader("📁 Download Feedback Data")

csv_data = df.to_csv(index=False).encode('utf-8')

st.download_button(
    label="⬇️ Download as CSV",
    data=csv_data,
    file_name="resident_feedback.csv",
    mime="text/csv"
)


st.subheader("□ Top Complaint Categories")
category_count = df["Category"].value_counts()
st.bar_chart(category_count)

if st.button("🔄 Refresh Data"):
    import os
    os.system("python sync_feedback.py && python sentiment_analysis.py")
    st.success("Data updated!")
    st.rerun()
```

Backend

sync_feedback.py

```
import gspread
import pandas as pd
from oauth2client.service_account import ServiceAccountCredentials

# Setting up Google Sheets
scope = ["https://spreadsheets.google.com/feeds", "https://www.googleapis.com/auth/drive"]
creds = ServiceAccountCredentials.from_json_keyfile_name("credentials.json", scope)
client = gspread.authorize(creds)

# Connecting to Google Sheet
SHEET_ID = "1dHS-IZz8BUbnWGlq9jF5-b2QEjZHCNeceTbbaKs-Xv8"  # <-- Replace this
SHEET_NAME = "Form responses 1"  # <-- Replace this if you've renamed it

worksheet = client.open_by_key(SHEET_ID).worksheet(SHEET_NAME)

# Getting all records as a list of dictionaries
records = worksheet.get_all_records()

# Converting to pandas DataFrame
df = pd.DataFrame(records)


# Save to CSV
df.to_csv("resident_feedback.csv", index=False)

print("Synced latest Google Form responses to resident_feedback.csv")
```

sentiment_analysis.py

```
import pandas as pd
from textblob import TextBlob

# Load raw feedback
df = pd.read_csv("resident_feedback.csv")

#  Sentiment Analysis
def get_sentiment(text):
    polarity = TextBlob(text).sentiment.polarity
    if polarity > 0.1:
        return "Satisfied"
    elif polarity < -0.1:
        return "Frustrated"
    else:
        return "Neutral"

df["Sentiment"] = df["Feedback"].apply(get_sentiment)

#  Defining Categories and Keywords
category_keywords = {
    "Water": ["water", "drinking", "supply", "tap", "pipeline"],
    "Electricity": ["electricity", "power", "voltage", "cut", "load shedding"],
    "Sanitation": ["garbage", "waste", "trash", "cleaning", "dustbin"],
    "Infrastructure": ["road", "street", "drain", "pothole", "construction", "sidewalk"],
    "Security": ["security", "theft", "guards", "safety", "robbery"],
    "Noise": ["noise", "loud", "sound", "disturbance"],
    "Other": []  # Default fallback
}

#  Categorization Function
def categorize_feedback(text):
    text = text.lower()
    for category, keywords in category_keywords.items():
        if any(keyword in text for keyword in keywords):
            return category
    return "Other"

df["Category"] = df["Feedback"].apply(categorize_feedback)

# Save updated file
df.to_csv("processed_feedback.csv", index=False)
print("✅ Sentiment + Category added and saved in 'processed_feedback.csv'")
```

filecoderunner.py

```
import pandas as pd
coord_df = pd.read_csv("sector_coords.csv")
sector_coords = {row["Sector"]: [row["Latitude"], row["Longitude"]] for _, row in coord_df.iterrows()}
print(sector_coords["Sector 1"])  # Output: [18.7002, 82.8608]
```

Future Prospects

To evolve into a full-fledged Smart City Management Suite, several enhancements and extensions can be envisioned:

1. Integration with IoT Devices

- **Use of smart sensors on street lights for real-time data (current, motion, light levels)**
- **Edge computing for immediate fault detection and logging**

2. Sentiment Categorization by Topic

- **Classify feedback not just by sentiment but also by subject (e.g., cleanliness, noise, power outage)**
- **Use LLMs (like BERT or GPT) for deeper contextual analysis**

3. Scalability Enhancements

- **Migrate from CSV to cloud-based databases (e.g., Firebase, PostgreSQL on AWS)**
- **Containerize using Docker and deploy via Kubernetes for microservice management**

4. Alerting & Automation

- **Send SMS/email alerts to maintenance staff based on predicted faults or recurring negative sentiment**
- **Auto-generate maintenance tickets linked with task management systems**

5. Mobile App Companion

- **Build a cross-platform mobile app (using Flutter or React Native) for residents to submit feedback and get live updates**

6. Public Transparency Portal

- **Allow residents to view aggregated sentiment and fault statistics sector-wise, enhancing civic trust and involvement**

7. Integration with GIS Systems

- **Overlay sentiment and fault data on real-world maps using tools like Leaflet.js or Mapbox for spatial analytics**

8. Energy Efficiency Suggestions

- **Use AI to suggest optimal operation schedules or upgrades for reducing power consumption in street lighting**

Conclusion

The Smart Township project represents a significant step toward intelligent, responsive, and citizen-centric urban management. By integrating machine learning and natural language processing into a unified web platform, this project enables authorities to transition from reactive to proactive service delivery.
The Street Light Fault Prediction System empowers city managers to forecast technical issues before they occur, minimizing downtime and optimizing maintenance schedules. Simultaneously, the Resident Sentiment Analysis Module captures and interprets the voice of the community, providing valuable insights into public satisfaction and sector-wise feedback.
The dashboard-driven interface provides a centralized, real-time view of both technical and social indicators, enhancing transparency and operational efficiency. This dual-module integration demonstrates how AI and web technologies can work in harmony to transform traditional infrastructure into a smart ecosystem.
This project not only solves specific challenges in lighting maintenance and feedback analysis but also lays the foundation for a broader smart city framework that can scale across sectors like water supply, sanitation, traffic, and healthcare.