PHASE 3: IMPLEMENTATION OF PROJECT

TITLE: TRAFFIC FLOW OPTIMIZATION USING AI AND IOT

OBJECTIVE

The goal of phase 3 is to implement the core components of a smart traffic optimization system using AI and IoT. This includes the development of a traffic prediction model, deployment of a traffic-sensing network, implementation of control algorithms for dynamic signal adjustments, and integration of data privacy and security measures.

1.AI MODEL DEVELOPMENT

Overview

The core of the system is a machine learning model that analyses historical and real-time traffic data to optimize traffic signal timing and routing recommendations.

Implementation

- **Traffic prediction Model:** Use supervised learning to analyse traffic patterns using historical data and real-time feeds.
- **Data Source:** Datasets include city traffic logs, GPS data from vehicles, and feeds from road sensors and cameras.
- **Model functionality:** Predict congestion,Travel times, and suggest dynamic adjustments to traffic lights.

Outcome

By the end of phase 3, the AI model should be able to predict traffic build-ups and suggest optimal signal timings with a focus on reducing delays at key intersections.

2. SENSOR AND IOT INTEGRATION

Overview

To enable real-time optimization, sensors and IoT devices will collect traffic data, such as vehicle counts, speeds, and environmental conditions.

Implementation

- **Network Integration:** Connect sensors to a central server using edge devices and IoT gateways.
- **API Usage:** Data transmitted via MQTT or RESTful APIs for analysis and processing.

Outcome

The system will have real-time access to live traffic conditions, enabling dynamic adjustments to traffic flow via smart signaling.

3. ADAPTIVE TRAFFIC SIGNAL CONTROL

Overview

Based on the AI predictions, traffic signals will adapt their timings dynamically to improve flow and reduce congestion.

Implementation

- **Control Algorithms:** Implement reinforcement learning or heuristic-**based** algorithms for adjusting light cycles.
- **Intersection Management:** Prioritize high-density routes and emergency vehicle detection
- **Feedback Loop:** Integrate with live data from sensors to adjust signals every few minutes.

Outcome

Smart traffic signals will adapt in near real-time to traffic volumes, significantly improving flow efficiency.

4. DATA SECURITY AND PRIVACY

Overview

Traffic and vehicle data may contain sensitive location information; hence secure data handling is essential.

Implementation

- **Encryption:** All transmitted data will be encrypted using TLS.

- **Secure Storage:** Use secure cloud-based infrastructure with restricted access.
- **Compliance:** Ensure compliance with local data protection laws and anonymize vehicle data.

Outcome

All data collected and processed during this will be phase stored and transmitted securely.

5. TESTING AND FEEDBACK COLLECTION

Overview

Testing will be conducted in a simulated environment or controlled field deployment to evaluate system performance.

Implementation

- **Simulations:** Use tools like SUMO or VISSIM to simulate traffic conditions.
- **Test Sites:** Deploy in a small real-world intersection or city sector.
- **User Feedback:** Gather input from traffic managers and drivers on perceived improvements.

Outcome

Feedback will be used to fine-tune the AI model, improve signal control logic, and prepare the system for broader deployment.

Challenges and Solutions

1. **Real-Time Data Reliability**
    - **Challenge**: Sensor failures or connectivity issues.
    - **Solution:** Redundant sensors and fallback prediction models.
2. **Scalability**
    - **Challenge**: Expanding the system to a city-wide level.
    - **Solution:** Modular architecture and cloud-based scalability.
3. **System Integration**
    - **Challenge:** Integration with legacy traffic systems.
    - **Solution:** Use middleware and open-standard APIs for compatibility.

OUTCOMES OF PHASE 3:

1. **BASIC AI MODEL**: AI model capable of predicting and optimizing traffic flow.
2. **FUNCTIONAL CHATBOT INTERFACE**: Real-time sensor network integrated for live data collection.
3. **OPTIONAL IoT INTEGRATION**: If IoT devices are available Dynamic traffic signal control in a test environment.
4. **DATA SECURITY:** Data handling procedures compliant with security standards which gives protection for mechanism in place.
5. **INITIAL TESTING AND FEEDBACK:** Initial performance testing and stakeholder feedback collected.

NEXT STEPS FOR PHASE 4:

In phase 4, the team will focus on:

1. **Wider Deployment:** Expand to more intersections and integrate public transport data.
2. **User Interface Development:** Develop a dashboard for traffic controllers and a mobile app for drivers.

CODE FOR TRAFFIC FLOW AND OPTIMIZATION:

PICTURE1:

```python
import streamlit as st
import pandas as pd
import numpy as np
import plotly.express as px
import plotly.graph_objects as go
import json
import os
from openai import OpenAI
import re

class Chatbot:
    """
    Class for natural language processing of data queries
    """

    def __init__(self):
        # Initialize OpenAI client with API key from environment
        self.openai_api_key = os.environ.get("OPENAI_API_KEY")

        if self.openai_api_key:
            self.client = OpenAI(api_key=self.openai_api_key)
        else:
            self.client = None

    def process_query(self, query, df):
        """
        Process a natural language query about the data

        Args:
            query: String with the user's query
```

PICTURE 2:

```python
    # Calculate some basic statistics for numeric columns
    numeric_columns = df.select_dtypes(include=[np.number]).columns
    if not numeric_columns.empty:
        summary['numeric_stats'] = {
            col: {
                'min': float(df[col].min()) if not pd.isna(df[col].min()) else None,
                'max': float(df[col].max()) if not pd.isna(df[col].max()) else None,
                'mean': float(df[col].mean()) if not pd.isna(df[col].mean()) else None
            }
            for col in numeric_columns
        }

    # Information about categorical columns
    categorical_columns = df.select_dtypes(include=['object', 'category']).columns
    if not categorical_columns.empty:
        summary['categorical_stats'] = {
            col: {
                'unique_values': df[col].nunique(),
                'top_values': df[col].value_counts().head(3).to_dict()
            }
            for col in categorical_columns
        }

    # Information about datetime columns
    datetime_columns = df.select_dtypes(include=['datetime64']).columns
    if not datetime_columns.empty:
        summary['datetime_stats'] = {
            col: {
                'min': str(df[col].min()),
                'max': str(df[col].max())
            }
```

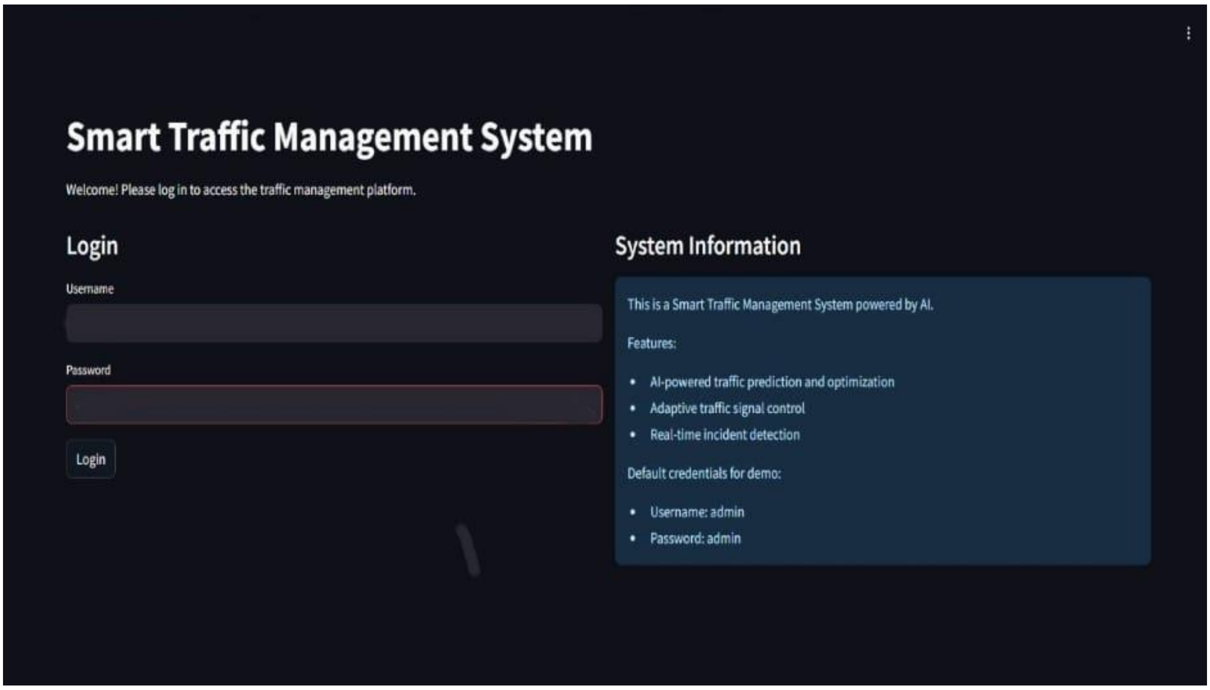
PICTURE 3:

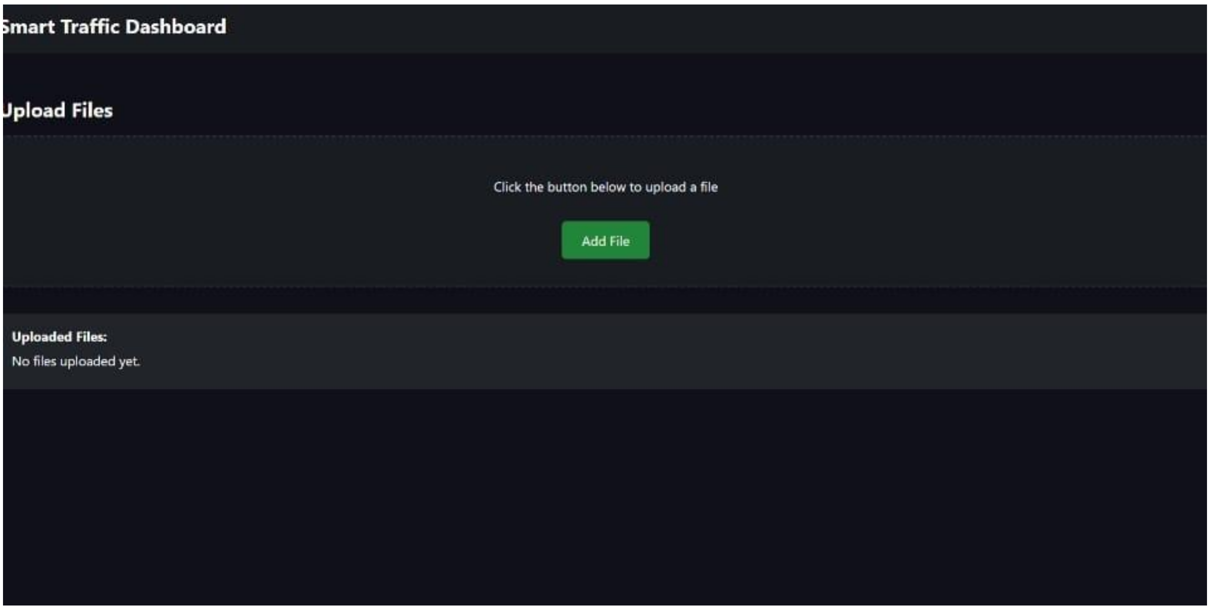
```
        Args:
            query: String with the user's query
            df: pandas DataFrame
            data_summary: String with data summary

        Returns:
            String with response
        """
        # the newest OpenAI model is "gpt-4o" which was released May 13, 2024.
        # do not change this unless explicitly requested by the user
        response = self.client.chat.completions.create(
            model="gpt-4o",
            messages=[
                {
                    "role": "system",
                    "content": (
                        "You are a data analysis expert. Your task is to answer questions about data. "
                        "Provide clear, concise, and accurate answers. Include relevant statistics or "
                        "calculations when appropriate. If the query can't be answered with the available data, "
                        "explain why."
                    )
                },
                {
                    "role": "user",
                    "content": f"Query: {query}\n\nData Summary: {data_summary}\n\nPlease answer this question about the data."
                }
            ]
        )

        return response.choices[0].message.content
```

OUTPUT:




Smart Traffic Dashboard
Upload Files
Click the button below to upload a file
Add File
Uploaded Files:
No files uploaded yet.