

# Final Project Proposal

## Project Title:

Smart Traffic Management System + Solar-Powered Street Lighting

## Project Description:

This project aims to design and implement a Smart Traffic Management System that integrates IoT, data analytics, and machine learning to monitor and control urban traffic efficiently. The system collects real-time traffic data, predicts congestion, and automatically adjusts street lighting based on traffic density. It enhances urban safety, energy efficiency, and sustainability by combining real-time analytics, predictive modeling, and smart lighting control using solar-powered infrastructure.

## Group Members & Roles:

Name	Role Description (What each member did)
Hassan Gamal Ghanem	Managed the overall project, organized meetings, followed up on progress, and ensured all milestones were delivered on time. (Team Leader)
Maya Yaser Amin	Created and ran the Python script that simulates traffic data such as vehicle count, speed, and timestamps.
Mohamed Mohamed Elkishkey	Cleaned and preprocessed the collected data, handled missing/duplicate records, and prepared the dataset for analysis.
Habiba Ashraf Elboghdady	Developed the process that analyzes real-time traffic and sends alerts when congestion happens, as well as adjusts lighting levels automatically.
Radwa Hany Sobhy	Trained and tested the machine learning model used for predicting traffic congestion using historical data.
Amr Mohamed Youssef	Designed and built the interactive

dashboard (using Power BI or Streamlit) to visualize real-time traffic, lighting levels, and future predictions.

**Team Leader:**

Hassan Gamal Ghanem

**Objectives:**

- Simulate and collect real-time traffic data using IoT-based sensors or Python scripts.
- Preprocess and clean data for analysis and modeling.
- Build an ETL pipeline for structured data storage in a data warehouse or data lake.
- Perform real-time traffic monitoring and generate alerts during congestion.
- Develop a machine learning model to predict traffic congestion trends.
- Control street lighting dynamically based on traffic density to save energy.
- Visualize real-time metrics, lighting levels, and congestion predictions through an interactive dashboard.

**Tools & Technologies:**

- Programming Languages: Python, SQL
- Data Handling & Storage: Azure Data Factory, Azure SQL Database, or local Data Lake
- Real-Time Processing: Apache Kafka or Azure Stream Analytics
- Machine Learning: Scikit-learn, Pandas, NumPy, Matplotlib
- Visualization: Power BI, Streamlit, or Grafana
- IoT Simulation: Python MQTT or Kafka producers
- Version Control: GitHub
- Cloud Platform (Optional): Microsoft Azure

**Milestones & Deadlines:**

Milestone	Description	Deadline
-----------	-------------	----------

1. Traffic Data Simulation	Develop Python script to generate and ingest traffic data (vehicle count, speed, timestamp, etc.)	Week 13
2. Batch Data Processing (ETL)	Process and clean data, calculate averages, detect congested streets	Week 16
3. Real-Time Analytics & Alerts	Set up live monitoring with alerts for congestion, integrate smart lighting control	Week 19
4. Traffic Prediction Model	Train and evaluate ML model to predict congestion	Week 22
5. Dashboard & Final Report	Develop interactive dashboard and prepare final report	Week 25

### Key Performance Indicators (KPIs):

#### 1. Data Preprocessing (Python Script, Cleaned CSV)

- ✓ Missing/duplicate data correctly handled — Target: 100%
- ✓ Script efficiency — Execution time within expected threshold

#### 2. SQL Integration (Schema, Queries)

- ✓ Query accuracy — Results match expected outputs, Target:  $\geq 95\%$
- ✓ Query performance — Average execution time under 2 seconds for test queries

#### 3. Visualization (Charts, Dashboard)

- ✓ Dashboard load time — Target:  $< 3$  seconds
- ✓ Required KPIs/metrics visualized — Target:  $\geq 90\%$

#### 4. Presentation (Report, Slide Deck)

- ✓ Report completeness — All required sections delivered, Target: 100%
- ✓ Stakeholder clarity/feedback score — Target:  $\geq 4/5$