Date of publication September 10, 2025, date of current version September 17, 2025.
Digital Object Identifier 10.1109/ACCESS.2017.DOI

TrafficWiz: A Database-Driven Tool for Analyzing Congestion and Safety in Nashville

HAIDER BAIG [1], MERHAWIT MEHARI[1], BENJAMIN SCHNEIDER[1], ZELALEM TEWELDE[1], DEMARICK WEBB-RIVERA[1]
[1] Middle Tennessee State University, Murfreesboro, TN
Corresponding author: Team TrafficWiz.

This IEEE document is part of the overall semester project for CSCI 4560 Fall 2025.

ABSTRACT Traffic congestion in Nashville and other major cities has become a constant challenge for city planners and commuters, often leading to wasted time and economic losses. There is an increased rate of accidents, resulting in higher stress levels for drivers and passengers alike. There are several existing approaches that try to help manage the problem. However, they often fail to integrate multiple data sources and/or provide dependable insights for both residents and city planners. To address this problem, we came up with the solution, our project, *TrafficWiz*, which aims to develop a database-driven analysis tool that combines real-world traffic data sets from the National Highway Traffic Safety Administration (NHTSA) and the Department of Transportation (DOT). The analysis tool uses MySQL for structured data storage, Python for queries and analysis, and JavaScript visualization libraries to highlight congestion patterns and accident hotspots. Our approach enables the creation of an interactive dashboard that presents traffic insights and predictive models in an accessible format. By integrating the multiple data sets from trusted sources into an intuitive dashboard, *TrafficWiz* aims to help decision makers improve traffic safety, reduce delays for common drivers, and improve traffic congestion to increase urban mobility in Nashville.

INDEX TERMS Traffic analysis, database systems, congestion, accident hotspots, MySQL, Python, visualization, Nashville

I. INTRODUCTION

URBAN traffic congestion in densely populated cities has become a topic of study in transportation research for smart city planning. Studies consistently show that traffic delays and accident frequency increase as road networks become saturated, leading to billions of dollars in lost productivity each year [1]. Data-driven approaches, such as predictive modeling and traffic simulation, have been used to understand congestion trends with the end goal of guiding policy interventions to improve the flow of traffic. Research has shown that integrating real-time traffic conditions with historical accident data improves the accuracy of traffic flow forecasts, enabling more effective transportation management [2]. However, many current systems are fragmented, focusing on safety metrics or congestion analytics, without combining them into a unified user-accessible platform.

There have been several tools and projects that have attempted to provide traffic insights at the city or regional level. For example, applications such as Google Maps and Waze use crowd-sourced data and sensor data from their applications to provide route recommendations, while government agencies such as the Tennessee Department of Transportation (TDOT) release periodic reports on traffic patterns [3]. Other academic projects, such as those that use the Federal Highway Administration datasets, focus on large-scale analysis but often lack interactive dashboards tailored to specific communities [4]. Our project, *TrafficWiz*, builds on these efforts by integrating open data sets from the National Highway Traffic Safety Administration (NHTSA) and the Department of Transportation into a single relational database. Unlike existing solutions, *TrafficWiz* emphasizes displaying transparent facts regarding both safety (accident hotspots) and efficiency (congestion trends) through a dashboard interface, allowing local stakeholders, as well as Nashville residents, to visualize and make decisions using traffic insights from *TrafficWiz* in real time.

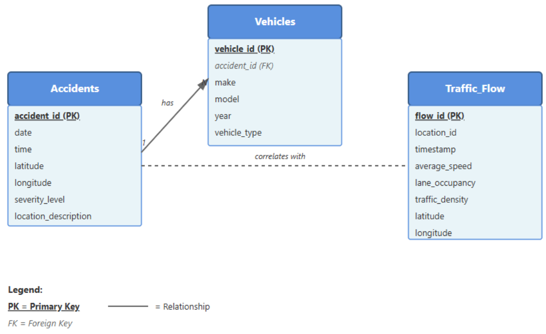

FIGURE 1. Entity–Relationship Diagram (ERD) generated from the TrafficWiz MySQL schema. The schema integrates accidents, vehicles, and traffic flow data through relational links.

II. DATABASE SCHEMA

The database was implemented using MySQL and is made up of several tables, including *Accidents*, *Vehicles*, and *Traffic_Flow*. These tables are interconnected through primary and foreign keys to ensure that the data remains consistent and is able to support complex analytical queries.

The *Accidents* table holds some of the most used information, such as accident ID, date, time, location coordinates, and severity level. The *Vehicles* table stores vehicle details, including make, model, and associates them with the accident table using a foreign key. The *Traffic_Flow* table captures congestion data from the Department of Transportation (DOT), including average speed, lane occupancy, and traffic density per time interval.

The relationships between these tables allow for an in-depth analysis using SQL queries. For example, a join between *Accidents* and *Traffic_Flow* can reveal correlations between congestion levels and the frequency of accidents in certain zones or time periods.

III. SYSTEM ARCHITECTURE

The overall design of the system follows a three-layer architecture: the *Database Layer*, *Application Layer*, and *Presentation Layer*, as shown in Fig. 2.

- **Database Layer:** Contains the structured MySQL relational database, which stores all accident, vehicle, and congestion data. This layer ensures data integrity and supports concurrent access.
- **Application Layer:** Written mainly in Python, this layer performs SQL searches, cleans data, and aggregates data. It also serves as a binder, helping link the database and the front-end.
- **Presentation Layer:** Implements JavaScript and Web-based visualization frameworks to display traffic metrics in real time. The dashboard provides interactive maps and charts that are useful for city officials, traffic engineers, and residents alike.

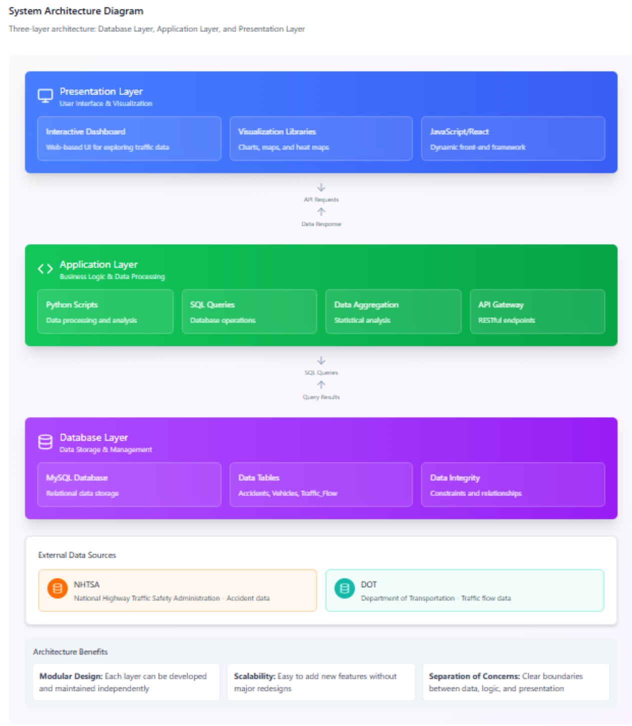

FIGURE 2. System architecture diagram showing database, application, and presentation layers of TrafficWiz.

The architecture allows for modular development—future versions can integrate APIs for real-time traffic data or predictive AI modules without major redesigns.

IV. SYSTEM WORKFLOW

The system workflow begins with importing traffic and accident datasets from NHTSA and DOT sources. These files, often in CSV or JSON format, are pre-processed and normalized prior to being inserted into the MySQL database. Python scripts are used to clean null values, remove duplicates, and structure fields for consistency.

After processing, SQL queries and Python analytics scripts aggregate the data into usable insights, such as:

- Average accident severity per highway.
- Peak congestion hours by location.
- Correlation between vehicle types and accident occurrence.

The final results are visualized through an interactive dashboard that allows users to filter by date range, location, or vehicle type. Users can explore heat maps of accident hotspots, visualize traffic flow trends, and view predictive congestion forecasts.

REFERENCES

[1] Texas A&M Transportation Institute, *2021 Urban Mobility Report*. College Station, TX, USA, 2021.
[2] Y. Zheng, L. Capra, O. Wolfson, and H. Yang, "Urban Computing: Concepts, Methodologies, and Applications," *ACM Transactions on Intelligent Systems and Technology*, vol. 5, no. 3, pp. 1–55, 2014.

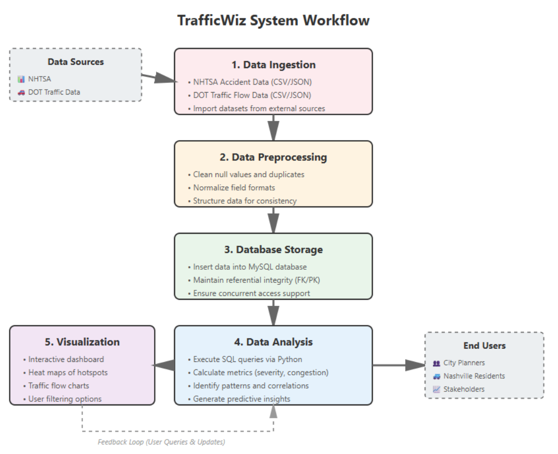

FIGURE 3. Workflow diagram for TrafficWiz, showing the data ingestion, preprocessing, storage, analysis, and visualization stages.

[3] Tennessee Department of Transportation, "Traffic and Safety Reports," 2023. [Online]. Available: https://www.tn.gov/tdot

[4] Federal Highway Administration, "Highway Statistics Series," 2023. [Online]. Available: https://www.fhwa.dot.gov/policyinformation/statistics.cfm