**1. Introduction**

**1.1 Background of the Capstone Project**

Quezon City District 2 already has serious traffic issues because of the rise of the population and the number of vehicles on the road. Daily, citizens suffer from congestion, accidents, and delays that compromise work, education, and emergency services.

The manual system is the one being employed by the city government for monitoring traffic flow and the reporting of issues. Manual methods are characterized by their sluggishness and punctuated inaccuracy. There is thus the necessity of a new system that can enhance traffic management and offer real-time data.

Most cities are now employing technologies such as CCTV cameras, AI, and maps to enhance transport. Applications such as Google Maps API and rule-based AI can assist in identifying traffic problems, recommending improved routes, and reporting offenses quicker.

This project will assist the local government in developing a Traffic and Transport Management System, which will incorporate real-time traffic updates, road conditions, accident reports, and smart routing. The vision is to make District 2 transportation safer, quicker, and more organized.

**1.2 Context and Scope**  The Local Government Unit 4 (LGU 4) operates within a rapidly urbanizing environment where traffic congestion, inefficient transportation flow, and violation reporting remain persistent challenges. With the increasing number of vehicles and commuters, the need for a modernized Traffic and Transport Management System has become a critical priority. Traditional systems often rely on manual monitoring and reporting, which leads to delays, inefficiencies, and limited data accuracy. To address these issues, the proposed project integrates a rule-based artificial intelligence model with OpenStreetMap through the Leaflet framework to enable intelligent violation reporting and optimized route planning. This approach not only enhances real-time traffic monitoring but also improves decision-making for both enforcers and commuters.

              The scope of the project focuses on the development and deployment of a system that allows local authorities to record, process, and manage traffic violations digitally, while providing route optimization features for effective transport management. The project covers functionalities such as violation reporting by enforcers, data validation through AI rules, geospatial visualization of traffic conditions, and dynamic route recommendations based on real-time data. It will also include administrative features for LGU officials to oversee traffic patterns and generate reports for policy formulation.

              However, the system will not extend to nationwide traffic management or integration with other LGUs beyond LGU 4 during its initial implementation. It will also not replace existing enforcement laws and regulations but instead serve as a technological support system to enhance compliance and efficiency. The project will primarily operate within the jurisdiction of LGU 4, focusing on improving local mobility, safety, and enforcement transparency. By establishing clear boundaries and objectives, the system ensures a realistic scope that addresses the immediate transportation challenges of LGU 4 while leaving room for potential expansion in the future.

**1.3 Problem Statement** The Local Government Unit 4 (LGU 4) operates within a rapidly urbanizing environment where traffic congestion, inefficient transportation flow, and violation reporting remain persistent challenges. With the increasing number of vehicles and commuters, the need for a modernized Traffic and Transport Management System has become a critical priority. Traditional systems often rely on manual monitoring and reporting, which leads to delays, inefficiencies, and limited data accuracy. To address these issues, the proposed project integrates a rule-based artificial intelligence model with OpenStreetMap through the Leaflet framework to enable intelligent violation reporting and optimized route planning. This approach not only enhances real-time traffic monitoring but also improves decision-making for both enforcers and commuters

           The scope of the project focuses on the development and deployment of a system that allows local authorities to record, process, and manage traffic violations digitally, while providing route optimization features for effective transport management. The project covers functionalities such as violation reporting by enforcers, data validation through AI rules, geospatial visualization of traffic conditions, and dynamic route recommendations based on real-time data. It will also include administrative features for LGU officials to oversee traffic patterns and generate reports for policy formulation.

            However, the system will not extend to nationwide traffic management or integration with other LGUs beyond LGU 4 during its initial implementation. It will also not replace existing enforcement laws and regulations but instead serve as a technological support system to enhance compliance and efficiency. The project will primarily operate within the jurisdiction of LGU 4, focusing on improving local mobility, safety, and enforcement transparency. By establishing clear boundaries and objectives, the system ensures a realistic scope that addresses the immediate transportation challenges of LGU 4 while leaving room for potential expansion in the future.

**1.4 Objectives and Goals** The objectives of this project define the specific, measurable outcomes that the development team intends to accomplish through the creation of the Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet. Meanwhile, the goals represent the broader, long-term contributions of the project in improving transportation management, compliance enforcement, and public service delivery for the Local Government Unit (LGU).

**OBJECTIVE**

|  |  |  |
| --- | --- | --- |
| No. | **OBJECTIVE** | DESCRIPTION |
| 1. | Automate Violation Reporting | To develop a module that enables the system to detect, record, and manage traffic violations using rule-based AI. |
| 2. | Route Optimization | To implement optimized routing using OpenStreetMap data and Leaflet integration for efficient travel paths |
| 3. | Centralized Data Management | To establish a secure and centralized database that stores violation reports, transport data, and route history. |
| 4. | Real-Time Monitoring | To provide LGU officers with real-time visibility of traffic activities and violations for immediate response |
| 5 | Analytics and Reporting | To generate dashboards and analytical reports for identifying traffic patterns and violation trends. |
| 6. | User-Friendly Interface | To design a simple, intuitive web-based and mobile interface accessible to LGU staff, commuters, and drivers. |
| 7 | Secure System Access | To ensure system security through authentication, encryption, and role-based access control. |
| 8 | Integration of OpenStreetMap | To fully utilize OpenStreetMap via Leaflet for accurate geospatial mapping and visualization of traffic routes. |

**GOALS**

|  |  |  |
| --- | --- | --- |
| No. | **GOALS** | DESCRIPTION |
| 1. | Improve Traffic Flow | To enhance transportation efficiency and reduce congestion within the LGU’s jurisdiction. |
| 2. | Enhance Road Safety | To reduce accidents and violations by promoting compliance with traffic rules through automated monitoring. |
| 3. | Strengthen LGU Public Services | To support LGU officers in delivering fast, reliable, and transparent transport management services. |
| 4. | Environmental Sustainability | Enable clearer transaction tracking for both customers and owners. |
| 5. | Promote Digital Governance | To align LGU operation with modern smart city initiatives by adopting digital and AI-driven solutions. |
| 6. | Establish scalability for future use | Allow the system to be expanded or integrated with LGU waste monitoring initiatives. |
| 7. | Data-Driven Decision Making | To build a system that can integrate with emerging technologies such as loT traffic sensors and predictive AI models |

**1.5 Significance and Relevance**

              The development of the Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet is highly significant as it addresses the growing challenges of urban mobility, road safety, and efficient governance in Local Government Unit 4 (LGU 4). By introducing automation and intelligent technologies, the project enhances the capability of traffic enforcers and local authorities to monitor violations, manage transport systems, and make data-driven decisions. This relevance extends beyond traffic enforcement, as it contributes to public safety, reduced congestion, and improved quality of life for citizens.

      For commuters and motorists, the system provides real-time route optimization, reducing travel time, fuel consumption, and stress caused by traffic congestion. This not only benefits individuals but also contributes to environmental sustainability by minimizing vehicle emissions. For traffic enforcers and local government officials, the system ensures accurate violation reporting and record management, enabling better enforcement of regulations and transparent handling of cases.

        The project also benefits policymakers by generating valuable data insights that can inform future urban planning, infrastructure development, and transport policies. For the wider community, it fosters a safer and more efficient transport environment, ultimately boosting economic activity by improving the movement of people and goods.

       In the broader context, the project demonstrates the relevance of integrating artificial intelligence and open-source mapping technologies in solving local governance challenges. It highlights the potential of digital innovation to transform public service delivery, making it a model for other LGUs to adopt in addressing similar transportation and traffic management issues**.**

**1.6 Structure of the Document** This capstone project report is organized into five major chapters, each addressing a critical phase of the system’s design, development, and evaluation. The structure has been carefully arranged to present a logical and coherent flow of information from project initiation to conclusion.

        The first chapter introduces the study and provides the review of related literature. It outlines the background of the project, the context and scope, the problem statement, the objectives and goals, and the significance and relevance of the study. In addition, it includes an extensive review of relevant theories and concepts, such as the Agile Scrum methodology, enterprise architecture frameworks, microservices architecture, DevOps practices, and integration strategies in enterprise systems. This chapter serves as the theoretical and contextual foundation upon which the succeeding chapters are built.

       The second chapter discusses the business process architecture that underpins the proposed system. It identifies the existing operational workflows within the LGU environment and illustrates these processes using Business Process Architecture (BPA) and Business Process Model and Notation (BPMN) diagrams. The chapter also explains how the new system is intended to integrate with and enhance current business processes to improve overall efficiency and effectiveness.

           The third chapter focuses on the development process of the system, with emphasis on the implementation of the Agile Scrum methodology. It describes the team’s role assignments, sprint planning sessions, development activities, deliverables for each sprint cycle, and the challenges encountered throughout the process. Furthermore, it elaborates on the application architecture of the system, the microservices approach employed, and the use of DevOps tools to facilitate continuous integration and deployment.

          The fourth chapter covers the implementation and testing phases. It presents a detailed account of the technologies and tools utilized, the procedures for system deployment, and the strategies applied to integrate various software components. This chapter also highlights the testing methodologies used to ensure the system's functionality and reliability, including specific test cases, outcomes, debugging procedures, and the application of quality assurance techniques.

           The fifth and final chapter outlines the results and evaluation of the completed project. It assesses whether the system met its defined objectives and gathers feedback from users and stakeholders regarding the system’s usability and impact. Lessons learned during the course of the project are also discussed, alongside recommendations for future development and possible system enhancements. The chapter concludes with a summary of the project’s contributions to the domain of LGU systems.

**Part 2  Related Studies and Literature Review**

**2.1 Agile Scrum Methodology Overview**

          Iterative development and ongoing feedback are key components of Agile Scrum, a lightweight yet effective framework for managing complicated projects. Each sprint, which are brief, fixed-duration cycles, yields a usable increment of the final product. Scrum places a strong emphasis on cross-functional team cooperation, flexibility, and openness.  
  
          For the LGU4 Traffic and Transport Management System with Violation Reporting and Route Optimization, Scrum offers a methodical yet adaptable way to gradually provide key features like GIS-based mapping, AI-driven route optimization, and violation reporting. The team can effectively respond to changing policies, technological difficulties, and stakeholder feedback while ensuring that high-priority functionality reach end users promptly.

        To enable flexibility, teamwork, and continuous delivery, the Agile Scrum principles were used in the creation of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization utilizing Rule-Based AI and OpenStreetMap via Leaflet. One of the main tenets was customer collaboration, which placed a strong emphasis on ongoing communication with stakeholders like community leaders, LGU officials, and traffic enforcers to make sure the system's features met actual operational needs. The project adhered to the idea of releasing functional parts, such the live violation reporting form, early in the process rather than waiting for the full system to be finished. With this method, users could test and confirm features' usefulness right away and offer suggestions for enhancements. The team's ability to respond to changing stakeholder needs, real-time traffic data, and changing ordinances allowed them to modify priorities and scope, which was equally crucial. To construct a system that combined technological and policy requirements, engineers, GIS specialists, AI experts, and transportation planners worked closely together, prioritizing team interaction. Lastly, sprint reviews and retrospectives were used to enhance transparency and continuous development. This enabled for the tracking of progress and the identification of areas for process and deliverable improvement.

        The fundamental concepts of Scrum, including as roles, artifacts, and rituals, were used in its implementation. Setting priority in the backlog based on community requirements was the responsibility of the Product Owner (PO), who represented LGU leadership. In addition to addressing challenges and ensuring adherence to Agile principles, the Scrum Master (SM) assisted the Scrum process. To produce product increments, the Development Team—which included programmers, designers, testers, and GIS/AI experts—cooperated. The process was led by Scrum artifacts, and the product backlog was the master list of features that were wanted, including live traffic display, proof-of-violation uploads, and payment integration. The increment refers to a deployable, functional module that was provided at the conclusion of a sprint, whereas the sprint backlog represented the subset of items chosen for each sprint. The following Scrum rituals ensured discipline and progress: sprint planning involved choosing items from the backlog and defining sprint goals; daily stand-ups offered brief meetings to discuss progress, identify obstacles, and plan next steps; sprint reviews presented finished features to stakeholders; and sprint retrospectives gathered information about what went well and what needed improvement.

        Agile Scrum's implementation proved to be extremely pertinent to the LGU4 project. A functioning violation reporting module was successfully produced in the first sprint, demonstrating the team's ability to provide MVPs quickly using this framework. By holding regular meetings with LGU and transportation stakeholders, regular feedback loops were established, guaranteeing that the system adhered to regulations and solved practical issues. Because policy was kept flexible, features may be quickly modified in response to changes in rules or traffic patterns. By directing the efforts of developers, GIS/AI professionals, and policy specialists toward common goals, the framework also promoted interdisciplinary collaboration. Lastly, Scrum enabled a progressive rollout, which allowed stakeholders to progressively adjust to new features while maintaining system stability and usability. This included the phased introduction of advanced features like AI-driven dashboards for analytics, reporting, and route optimization.

**2.2  Enterprise Architecture Concepts**

             In order to keep complex systems in line with stakeholder demands, organizational objectives, and long-term sustainability, enterprise architecture, or EA, offers an organized method for their design, implementation, and management. We used ideas from the TOGAF (The Open Group Architecture Framework), a well-known framework for directing enterprise-level projects, especially in government and public service contexts, in the development of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet.

            Our system design reflects each of the four basic areas into which TOGAF divides architecture. In order to make sure that the system supports end-to-end procedures including violation reporting, route planning, and transport coordination, the project established the objectives, procedures, and operational guidelines of the LGU's traffic and transport services in terms of business architecture. To make sure the solution improves decision-making and day-to-day operations, stakeholder requirements from community residents, traffic enforcers, and LGU officials were included. A modular web-based application that incorporates a violation reporting module, a rule-based AI-powered route optimization engine, and an OpenStreetMap mapping interface via the Leaflet library was developed in order to address the Application Architecture. Real-time route planning and effective violation management are made possible by the smooth interaction of these modules.

           Structured information management, exchange, and storage throughout the system were the main goals of the data architecture. In order to guarantee compliance with OpenStreetMap APIs, a centralized database was created to store violation records, geolocation information, route histories, and AI decision rules. The database included well-defined standards for geographic coordinates, timestamps, and violation types. Last but not least, the Technology Architecture created the infrastructure needed to provide the system safely and dependably. The geographic information system platform is provided by OpenStreetMap, and the AI engine is constructed using server-side processing to guarantee performance and scalability. The application is hosted on servers that are controlled by the LGU and is accessible through conventional web browsers.

           Through the use of TOGAF principles to the system's design and implementation, the project made sure that every architectural layer—from technological infrastructure to business processes—was thoughtfully planned and connected. This method not only made the development process more structured and effective, but it also laid a strong basis for future improvements, like adding IoT sensors for real-time traffic monitoring or connecting to regional transportation databases for more extensive network coordination.

**· Microservices Architecture**

           In order to facilitate modularity, scalability, and ease of maintenance, the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet was created utilizing the concepts of microservices architecture. The application is divided into smaller, autonomous services that concentrate on particular business skills using a microservices approach, in contrast to typical monolithic systems, where all components are closely integrated and distributed as a single unit. For our project, distinct services were developed for essential tasks like route optimization, user administration, geolocation and mapping integration, violation reporting, and AI-driven decision-making. In order to maintain a flexible and robust system structure, each service functions independently while interacting with others via clearly defined APIs.

           Decentralized data management, continuous delivery, and service autonomy are some of the microservices concepts used in this project. Because each microservice operates independently of the others, changes or additions to one can be made without impacting the others. The development team can more swiftly roll out problem fixes or new features thanks to this autonomy. In order to minimize the risks associated with a single point of failure and to provide optimum storage solutions for various data kinds, such geospatial coordinates, violation records, or AI rules, decentralized data handling makes sure that each service handles its own data. Furthermore, new features may be introduced gradually without affecting the system's overall availability because the architecture was created with continuous integration and deployment in mind.

       Microservices adoption benefited the project in a number of ways. The route optimization service, for instance, may manage higher processing power during peak traffic hours without needing the rest of the system to grow needlessly, improving scalability by enabling specific services to be scaled up or down in response to demand. Additionally, it enhanced fault isolation so that the system can continue to function correctly even if one service has a problem. Additionally, microservices' modular design promotes technological diversity by enabling various services to make use of the databases, frameworks, and programming languages most suited to their particular requirements.

           The strategy did, however, also present difficulties that needed to be resolved throughout the development process. System complexity is increased while managing several services, especially when it comes to coordinating communication between them. To avoid data inconsistencies or unsuccessful requests, this necessitated the adoption of strong error-handling procedures and meticulous API architecture. Integrating centralized logging and performance monitoring tools was necessary since monitoring and debugging a dispersed collection of services proved more difficult than working with a single monolithic codebase. Furthermore, in order to prevent vulnerabilities, a universal authentication and authorization method was necessary to provide uniform security standards across all services.

           The application of microservices architecture offered a solid basis for a traffic and transport management system that is future-proof in spite of these difficulties. In order to produce a flexible and adaptable solution that can change in tandem with the LGU's expanding needs and technical improvements, the project made it possible for autonomous service development, implementation, and scaling.

**· DevOps and CI/CD**

           The Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet was developed using a DevOps methodology to promote cooperation between the operations and development teams, optimize workflows, and guarantee dependable, timely system update delivery. A cultural and procedural change that unifies software development, testing, deployment, and operations into a single process, DevOps is more than just a collection of technologies. Our team was able to overcome the conventional division between coding and deployment by establishing a continuous feedback loop that allowed for the timely delivery of new features and the use of operational information to guide development decisions.

           Using pipelines for Continuous Integration and Continuous Deployment (CI/CD) was a key component of our DevOps approach. Continuous Integration made sure that all developers regularly combined their code updates into a common repository, where test suites and automated build procedures checked the application's integrity. Defects were found early in the development cycle and integration conflicts were decreased as a result. This procedure was expanded by Continuous Deployment, which automated the deployment of verified modifications into the production and staging environments. Large-scale failures could be avoided and the system could be quickly adjusted to meet changing end-user and LGU requirements by updating it in smaller, easier-to-manage stages.

           Our microservices architecture is modular, which is supported by the CI/CD pipeline. The automated testing and deployment procedures used by each microservice allowed for independent updates without interfering with other services. Automated testing addressed security, performance, and functionality, while containerization tools like Docker made it easier to create consistent environments from development to production. This guaranteed the same behavior on the live server as what was successful in a developer's local configuration.

           Additionally, system maintenance and monitoring were enhanced via the DevOps methodology. Proactively identifying such problems was made possible by integrated monitoring technologies that provide real-time visibility into application performance, error rates, and server resource utilization. The team was able to minimize downtime and guarantee users' continued service delivery by promptly responding to any anomalies thanks to this operational openness.

           Through the adoption of DevOps and CI/CD, the project was able to increase its adaptability, software quality, and development cycles. Human error was minimized by the automated and repeatable procedures, and the development and operations teams were guaranteed to work together to create a dependable, scalable, and easy-to-use traffic and transport management system for the LGU.

**2.3  Relevant Studies and Research**

Philippine Studies (2020–2025)

Improvements in traffic management that are backed by GIS and AI technologies have been being adopted in the Philippines. In 2024, an AI-assisted traffic signal system was put in place at the SRP intersection in Cebu City to dynamically control traffic (The Freeman, 2024). In order to enhance real-time traffic flow modifications, Mandaue City subsequently implemented radar sensor-based AI traffic lights in 2025 (Reddit user, 2025). Significantly altering traffic enforcement procedures, the Metropolitan Manila Development Authority (MMDA) also implemented the No-Contact Apprehension Policy (NCAP), which used AI-enabled CCTV cameras to identify infractions and issue e-tickets (OpenGov Asia, 2025). According to The Philippine Star (2023), Metro Pacific Tollways Corporation (MPTC) also implemented smart traffic solutions in 2023, including congestion pricing, smart bus stops, and AI-powered monitoring via its command center.

Additionally, national agencies investigated smart transport management. To improve digital vessel movement monitoring for increased transport safety, MARINA and the International Maritime Organization started the SMART-C initiative in 2025 (MARINA, 2025). In the same year, Elbasha and Abdellatif (2025) presented an AIoT-based traffic model that may be used in Philippine cities. It combines cloud platforms, IoT sensors, and CCTV feeds to monitor traffic in real time. Road traffic was indirectly reduced in Legazpi City by using AI-driven geospatial solutions to optimize waste collection routes (UN-Habitat, 2023). OpenRoads Philippines has been assisting local governments with route planning since 2020 by mapping farm-to-market and tourism routes using OpenStreetMap (OSM GeoWeek, n.d.). Similar to this, Project NOAH's updated geospatial data has been modified for flood-prone areas' transportation planning, which helps with disaster response and traffic (Wikipedia, 2024). In conclusion, the centralized intelligent traffic management system STREAMS was a pioneering platform for integrated urban traffic control in Philippine cities (Wikipedia, 2024).

International Studies (2020–2025)

Mobility has changed globally as a result of data-driven traffic systems and artificial intelligence. With more than 55,000 junctions globally covered by 2025, the Sydney Coordinated Adaptive Traffic System (SCATS) continues to be one of the most significant smart signal management systems (Wikipedia, 2025). In order to alleviate urban congestion, the UK-developed SCOOT system keeps improving signal timing dynamically (Wikipedia, 2024). According to Wikipedia (2024), the SURTRAC decentralized traffic management system in the United States used artificial intelligence (AI) to reduce travel times and enhance signal synchronization across several cities. Additionally, in order to improve mobility and forecast flow patterns, European cities have integrated AI-enabled platforms that use GPS, IoT sensors, and cameras (Stellarix, 2023).

These efforts have been strengthened by industry advances. The European traffic solutions business SWARCO created algorithms for traffic signals based on reinforcement learning in an effort to cut down on delays and emissions (SWARCO, n.d.). Academically, Saxena (2024) examined new AI techniques for traffic prediction and signal optimization, including computer vision, fuzzy logic, and deep learning. Additionally, research on vehicle routing optimization has exploded. For example, Sabet and Farooq (2022) investigated sustainable routing using the Green Vehicle Routing Problem (GVRP), while Yao, Chen, and Yang (2021) presented fast optimization techniques for electric car routing and charging. Stochastic dynamic routing algorithms were also created by Akhmetbek (2024) to deal with erratic traffic situations. The promise of the AIoT traffic model put forth by Elbasha and Abdellatif (2025) for integrated urban mobility management has been demonstrated by the numerous international citations it has received.

References

Philippine Sources

Elbasha, E., & Abdellatif, M. (2025). AIoT-enabled smart traffic management: Integrating CCTV feeds and cloud platforms. *International Journal of Intelligent Transportation Systems*, 29(1), 112–128.

MARINA. (2025). MARINA, IMO launch SMART-C project to enhance ship monitoring and transport safety. *Maritime Industry Authority Philippines*. https://marina.gov.ph

OpenGov Asia. (2025). Quezon City launches AI-enabled no-contact apprehension policy. https://opengovasia.com

OSM GeoWeek. (n.d.). OpenRoads Philippines: Mapping farm-to-market and tourism roads with OpenStreetMap. https://osmgeoweek.org

Reddit user. (2025). AI-powered radar traffic lights in Mandaue City. *Reddit Philippines*. [https://reddit.com](https://reddit.com/)

The Freeman. (2024). Cebu City deploys AI-assisted traffic light system at SRP junction. *The Freeman*. https://philstar.com/thefreeman

The Philippine Star. (2023). Metro Pacific Tollways introduces smart traffic solutions. *The Philippine Star*. https://philstar.com

UN-Habitat. (2023). AI-driven waste collection pilot project in Legazpi City, Philippines. *United Nations Habitat*. https://unhabitat.org

Wikipedia. (2024). Project NOAH (Philippines). *Wikipedia*. https://en.wikipedia.org

Wikipedia. (2024). STREAMS traffic management system. *Wikipedia*. https://en.wikipedia.org

International Sources

Akhmetbek, A. (2024). *Stochastic dynamic vehicle routing in intelligent transportation systems*. Journal of Transport and Logistics, 15(2), 45–59.

Elbasha, E., & Abdellatif, M. (2025). AIoT-enabled smart traffic management: Integrating CCTV feeds and cloud platforms. *International Journal of Intelligent Transportation Systems*, 29(1), 112–128.

Sabet, A., & Farooq, B. (2022). The green vehicle routing problem: Sustainable approaches to transport optimization. *Transportation Research Part D*, 102, 103134.

Saxena, A. (2024). A review of AI methods for intelligent transport systems. *Artificial Intelligence Review*, 57(6), 3417–3436.

Stellarix. (2023). AI in traffic management: Transforming mobility with predictive analytics. *Stellarix Research*. https://stellarix.com

SWARCO. (n.d.). AI-powered traffic signal optimization. *SWARCO Traffic Solutions*. https://swarco.com

Wikipedia. (2024). SCOOT (traffic control system). *Wikipedia*. https://en.wikipedia.org

Wikipedia. (2024). SURTRAC: Scalable urban traffic control. *Wikipedia*. https://en.wikipedia.org

Wikipedia. (2025). SCATS: Sydney Coordinated Adaptive Traffic System. *Wikipedia*. https://en.wikipedia.org

Yao, E., Chen, X., & Yang, H. (2021). Fast optimization algorithms for electric vehicle routing and charging. *Transportation Science*, 55(4), 765–784.

**· Integration of Information Systems in Enterprise Environments**

           Information system integration is crucial for facilitating smooth data flow, enhancing operational effectiveness, and assisting with well-informed decision-making in contemporary business settings. In order for the various components of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet—including the geographic mapping interface, route optimization engine, violation reporting module, and user management services—to work together as a single, cohesive platform, integration is essential. Duplicate data, unreliable records, and slower service delivery would arise from the system functioning as isolated silos in the absence of appropriate integration.

          In this project, integration is crucial for reasons other than internal efficiency. The LGU can guarantee that real-time data, including traffic congestion updates, infraction reports, and optimum route recommendations, is easily accessible to authorized users across departments by integrating various subsystems via standardized APIs and communication protocols. Traffic enforcers, dispatchers, and administrative staff are better able to coordinate, which results in quicker incident reaction times and more effective resource allocation. Scalability is another benefit of integrated systems, which facilitates future integration with outside services like local transportation databases, police logs, or Internet of Things-based traffic sensors.

          In enterprise settings, technology integration poses a number of difficulties despite its advantages. A significant challenge is guaranteeing data accuracy and consistency as information moves between modules or external systems. If mismatches or faults are not addressed during the design stage, differences in data formats, naming standards, and storage structures may result. Integration raises the possibility of illegal access attempts targeting more communication points, which raises additional security concerns. Furthermore, when several services interact, performance optimization gets trickier, especially when demand is high.

         In order to overcome these obstacles, the project used a data-centric integration strategy backed by standardized data formats and clearly defined APIs. Because each microservice uses RESTful endpoints for communication, interactions are reliable and consistent. Error risk is decreased by using consistent structures for data interchange for geolocation information, timestamps, and violation types. Token-based authentication and encrypted communication are two examples of security techniques that protect sensitive data while it is being transferred. The project's long-term maintainability is further supported by modular integration, which enables the replacement or upgrade of individual services without affecting the system as a whole.

        The Traffic and Transport Management System offers the dependability, security, and flexibility needed in an industrial setting by using contemporary integration techniques and meticulous planning. This guarantees that the LGU will be able to effectively handle the traffic and transportation issues of the present while also increasing the system's capacity to accommodate future needs.

**Part 3.0 Methodology**

**Agile Scrum Methodology in the Project**

        Agile Scrum was used to oversee the deployment of the LGU4 Traffic and Transport Management System with Violation Reporting and Route Optimization utilizing Rule-Based AI and OpenStreetMap via Leaflet. Scrum was chosen because to its flexibility and methodical approach to managing complicated projects, especially ones that call for consistent stakeholder input and quick iterations. Its gradual design made sure that important features were released in stages while preserving the ability to improve features in response to input from system users, traffic enforcers, and LGU authorities.

          Every two-week sprint that was used to complete the project resulted in a usable increment of the system. The team used sprint planning to establish goals and choose backlog items from the product backlog that were prioritized at the beginning of each sprint. Early sprints concentrated on setting up the essential modules, like violation reporting and rudimentary GIS integration, while later sprints covered ticketing processes, payments, AI-powered route planning, and sophisticated analytics. Stand-up meetings were held every day to coordinate activities, resolve obstacles, and monitor advancement. Following a sprint review to showcase finished features and collect input, a sprint retrospective was conducted to determine process enhancements for the subsequent cycle.

           To guarantee seamless execution, roles in Scrum were clearly identified. Priorities were set by the Product Owner (an LGU representative), who also made sure the backlog represented the municipality's transportation objectives and policies. In addition to promoting adherence to Scrum principles and facilitating sprint activities, the Scrum Master also addressed obstacles that might hamper progress. Programmers, GIS experts, AI modelers, and testers worked together as the development team to create, test, and implement the functional increments. To provide accountability, transparency, and traceability throughout the project lifetime, Scrum artifacts such as the product backlog, sprint backlog, and increments were actively maintained.

* **Roles and Responsibilities (scrum master, product owner, development team)**

          Three crucial Scrum roles were established in the LGU4 Traffic and Transport Management System with Violation Reporting and Route Optimization utilizing Rule-Based AI and OpenStreetMap via Leaflet. In order to ensure that the team adhered to Agile principles, the Scrum Master organized sprint planning, daily stand-ups, reviews, and retrospectives. Along with encouraging teamwork and keeping an eye on developments to ensure the timely delivery of system features, this function also required addressing roadblocks like scheduling conflicts, ambiguous requirements, or technological difficulties like API integration.

          The Product Owner served as the voice of the users by establishing the product vision, setting priorities and managing the backlog, and outlining the requirements for features like AI-based rule verification, violation reporting, and Leaflet-based route optimization. The Product Owner represented LGU stakeholders, including officials, commuters, and traffic enforcers. In accordance with acceptance criteria, the Product Owner authorized deliverables and kept in regular contact with stakeholders.

          Meanwhile, the Development Team, which was made up of cross-functional members, was in charge of creating and putting into use the database and violation reporting interface, integrating the rule-based AI for route optimization, and embedding and modifying OpenStreetMap with Leaflet. Additionally, they created a user-friendly interface for the general public and LGU staff, tested and debugged the system to guarantee accurate results, kept up-to-date documentation, allowed their tasks to be self-organized without micromanagement, and used continuous integration to maintain system stability while frequently adding new features.

* **Sprint Cycles (planning, standups, review)**

          The Agile Scrum approach was used to build structured sprint cycles for the development of the LGU4 Traffic and Transport Management System with Violation Reporting and Route Optimization utilizing Rule-Based AI and OpenStreetMap via Leaflet. Iterative progress and ongoing stakeholder participation were ensured by adhering to the typical Scrum procedures of planning, daily stand-ups, and review during each two-week sprint.

          Prioritized features from the product backlog, including the LGU administrator dashboard, the rule-based AI route optimization integrated with Leaflet, the violation reporting module with media uploads, and OpenStreetMap connectivity, were presented by the Product Owner during Sprint Planning. By breaking down these high-level features into smaller, more manageable tasks, like creating the user interface for the violation report form, integrating the route calculation engine, and setting up map layers, the team was able to establish a clear sprint goal and make a commitment to completing high-priority features in the allotted two weeks.

          Every morning, for 1 hour, Daily Stand-ups were held in order to keep everyone in sync and spot problems early. Each team member gave a report on their present work, their accomplishments from the previous day, and any challenges they had faced. For instance, a developer may aim to connect the infraction form to the database that day, report that the front-end is complete, then raise a problem requesting LGU IT credentials. The team stayed focused on sprint goals and obstacles were promptly resolved thanks to the Scrum Master's facilitation of these meetings.

          Following each cycle, the team showed finished features to stakeholders and LGU authorities during a one- to two-hour Sprint Review. An administrative dashboard displaying traffic routes and violation statistics, an AI-driven route planning tool, and a functional violation reporting interface were among the main results. Active feedback gathering resulted in the inclusion of new requirements, including date and location filters and heatmaps for violation hotspots. After being added to the product backlog, these recommendations were given a new priority for upcoming sprints.

          With the help of this sprint structure, the project was able to be developed gradually, guaranteeing that stakeholders would have functional features at the conclusion of each cycle while the team continuously adjusted to new specifications and enhanced procedures.

* **Scrum Artifacts (product backlog, sprint backlog)**

          The LGU4 Traffic and Transport Management System with Violation Reporting and Route Optimization utilizing Rule-Based AI and OpenStreetMap via Leaflet was developed, and Scrum artifacts were essential for setting priorities and structuring the work. The team used the sprint and product backlogs extensively to monitor progress, manage system features, and make sure the project met stakeholder expectations.

          All of the desired features and system requirements were compiled into a comprehensive list called the Product Backlog. It featured key features like the AI-based route optimization engine, the interactive mapping interface with Leaflet, the administrative dashboard for LGU officials, the violation reporting module (which supported photo and video uploads), and integration with OpenStreetMap. It was mainly overseen by the Product Owner. Prioritizing each item in the backlog according to stakeholder demands meant that high-value features like route optimization and violation reporting were put off until earlier sprints. The backlog was continuously improved as LGU representatives' input was taken into account, guaranteeing that changing requirements like the inclusion of a violation heatmap and report filtering were noted and reranked appropriately.

          On the other side, the subset of backlog items committed for completion during the two-week sprint were included in the Sprint Backlog, which was produced during sprint planning. The sprint backlog provided the development team with a clear roadmap for the sprint, which helped them stay focused and self-organize their workload. Each high-level feature was divided into smaller, actionable tasks, such as "design the violation form UI," "connect the database to store reports," "embed Leaflet map layers," and "implement rule-based route calculation." During daily stand-ups, the progress on these tasks was monitored, and any obstacles were brought to the attention of the Scrum Master for resolution.

          The team was able to strike a balance between short-term execution plans and long-term vision by skillfully managing the sprint backlog as a dynamic list of priorities and the product backlog as a list of priorities. With this strategy, the project was guaranteed to progress gradually while staying sensitive to technical difficulties and stakeholder input.

* **Microservices Architecture**    
    
   A microservices architecture, which divides the application into discrete, autonomous services that interact via APIs, was used to create the LGU4 Traffic and Transport Management System. This strategy was selected to guarantee scalability, modularity, flexibility, and ease of maintenance, enabling the system to adjust to changing local government unit (LGU) needs. The fact that each microservice is in charge of a distinct business function facilitates testing, updating, and scaling without compromising the system as a whole.  
    
   The Violation Reporting Service is a fundamental element that manages the filing of traffic tickets. This service allows users to safely save incident descriptions, images, and videos in the database. Violation records can be retrieved using its APIs and subsequently displayed to administrators. The system's intelligence layer is the Route Optimization Service. To recommend the best routes for cars, it analyzes real-time data and traffic laws using rule-based AI combined with Leaflet and OpenStreetMap. This module's decoupling allows algorithms to be changed or improved without affecting other services, increasing the system's adaptability to future improvements.

 Real-time geographic data rendering is managed by the Mapping and Visualization Service. Road networks, hotspots for violations, areas of traffic congestion, and best routes are all shown on the map interface. In addition to interacting with the Leaflet library and OpenStreetMap APIs, this microservice can develop on its own to add more sophisticated features like heatmaps, location and time filtering, and predictive traffic visualization. Roles and permissions are managed by the User Management and Authentication Service to guarantee system security and restricted access. Specific access levels are granted to public users, traffic enforcers, and LGU administrators in order to prevent illegal use of the system.

 Authentication procedures are scalable and consistent across all other modules since this is isolated as a separate microservice. Lastly, the Admin Dashboard Service creates a central administration interface by combining outputs from all other services. LGU officials can examine optimization outcomes, track traffic patterns, and view infraction information with the use of this dashboard. By using RESTful APIs to communicate with the other microservices, it guarantees a smooth information flow without establishing interdependencies.

There are some very good reasons why you might want to use microservices. First, it allows independent deployment and scaling, e.g., the Route Optimization Service can scale additional instances out when peak traffic arrives without overloading the system as a whole. Second, it enhances fault isolation, and in the case of a service failure, the whole system still works. Third, it supports continuous integration and delivery (CI/CD), enabling the development team to update the program incrementally as opposed to shaking it up in its entirety.Applying this architecture, the LGU4 Traffic and Transport Management System possess of a design able to be easily adapted, reliable and software migrated. This can include next-gen tech such as cloud-based traffic analytics, AI-based predictive modelling and IoT sensors without needing a total system

redesign.

**DevOps Implementation**

Using a DevOps methodology, the LGU4 Traffic and Transport Management System was created and implemented, guaranteeing ongoing cooperation between the development and operations teams. DevOps offered a framework for monitoring, automation, and integration that facilitated dependable deployments, quicker delivery cycles, and enhanced system performance. First, GitHub was used to create version control, and all code repositories were kept up to date. Because branching methods and pull requests allowed developers to work independently on various microservices (such as Violation Reporting, Route Optimization, Mapping, and Dashboard) while preserving consistency, this made team cooperation easier.

The team set up automated build processes in order to implement Continuous Integration (CI). The continuous integration pipeline carried out linting, unit testing, and code quality checks whenever new code was submitted in order to guarantee that problems were found as soon as possible. This decreased the likelihood of bugs making it to production and preserved the integrity of microservices. Each microservice was packaged using Docker containers for Continuous Deployment (CD), guaranteeing stability across many environments. Containerization made the program scalable and platform-independent, enabling LGU IT administrators to execute services with little configuration on local government servers or cloud infrastructure.  
  
         DevOps procedures included automatic security scans built into the pipeline to find vulnerabilities in dependencies, configurations, or APIs in order to secure the system. In order to safeguard violation reports, traffic data, and route information, regular backups were also automated. The DevOps culture also placed a strong emphasis on feedback loops and teamwork. To guarantee that updates like new mapping layers, modifications to AI rules, or dashboard improvements were released quickly and consistently, developers, testers, and LGU stakeholders collaborated closely. When Agile Scrum and DevOps were used together, features could be moved from the backlog to deployment in a sprint cycle with ease.

**Innovation Integration**

 The LGU4 Traffic and Transport Management System combined cutting-edge methods and technologies to address the problems of data-driven decision-making, traffic enforcement, and urban mobility. Using a Rule-Based Artificial Intelligence (AI) engine for route optimization and violation detection was a significant innovation. The AI used set rules, such as avoiding clogged roads, rerouting traffic around hazardous locations, and comparing reckless driving to local laws, in contrast to standard static routing. This made it possible for commuters and LGU authorities to take advantage of more intelligent and flexible transportation management.  
  
 The combination of OpenStreetMap (OSM) with Leaflet, an open-source mapping toolkit, to produce real-time, editable maps, was another significant innovation. In contrast to proprietary mapping services, OpenStreetMap(OSM) provided flexibility, cost-effectiveness, and adaptation, allowing the system to dynamically overlay road conditions, infraction hotspots, and traffic routes. Commuters received precise and instantaneous navigation assistance, and LGU administrators received actionable visual information.

The Violation Reporting Module was improved with multimedia features including photo and video submissions to strengthen enforcement. By providing tangible proof for violations, this innovation increased accountability and openness for both civilians and law enforcement. After the data was gathered, it was analyzed and shown using an interactive LGU dashboard that showed compliance statistics, traffic bottlenecks, and trends in violations. Additionally, the system incorporated DevOps procedures and microservices architecture, guaranteeing scalability, modularity, and quick deployment. New AI rules, intelligent map filters, or dashboard analytics are examples of novel system changes that might be implemented quickly without interfering with ongoing operations by utilizing containerization and automated pipelines.

Lastly, innovation encompassed both organizational and technological aspects. Through the integration of Agile Scrum and DevOps, the project promoted cooperation among developers, end users, and LGU stakeholders. Innovations such as filtering options for traffic data and heatmaps for violation hotspots may be quickly conceived, tested, and integrated based on stakeholder feedback thanks to this iterative methodology. These connections transformed the LGU4 system from a conventional transport management platform into a cutting-edge, data-driven, AI-driven, and citizen-centric instrument for enhancing urban mobility and governance.

**CHAPTER II**

**2. REVIEW OF RELATED LITERATURE**

**2.1 Agile Scrum Methodology Overview** Agile Scrum is a widely adopted project management framework that emphasizes iterative and incremental development to deliver functional software in shorter cycles. It focuses on adaptability, collaboration, and customer feedback, allowing project teams to quickly respond to changing requirements and priorities. Unlike traditional models, Scrum structures work into time-boxed iterations called sprints, which typically last between one to four weeks. Each sprint results in a potentially shippable product increment, ensuring continuous progress and stakeholder engagement (Schwaber & Sutherland, 2020)

               The methodology defines specific roles, artifacts, and ceremonies to guide teams. The roles include the Product Owner, responsible for defining requirements and prioritizing the product backlog; the Scrum Master, who facilitates the process and removes impediments; and the Development Team, which executes the work collaboratively. The ceremonies consist of sprint planning, daily standups (scrum meetings), sprint review, and sprint retrospective, all designed to ensure transparency, inspection, and adaptation (Hoda, Salleh, & Grundy, 2021).

               Scrum is particularly effective in managing complex and dynamic projects such as traffic and transport management systems because it allows continuous iteration, user feedback, and integration of emerging technologies. For example, in government digital transformation initiatives, Scrum enables faster delivery of citizen-centered services while maintaining flexibility for policy and compliance adjustments (Mahalakshmi & Sundararajan, 2021). Recent studies emphasize its role in promoting collaboration, quality assurance, and stakeholder involvement, making it suitable for LGU-led innovations like traffic violation reporting and AI-assisted route optimization (Rasnacis & Berzisa, 2022).

               Overall, Agile Scrum provides a structured yet flexible approach to project management. By combining iterative development, defined team roles, and structured ceremonies, it fosters transparency, accountability, and continuous improvement, all of which are essential for building sustainable and scalable LGU systems (Yadav & Singh, 2023).

**Benchmarking and Secondary Research**

               To strengthen the design and development of the proposed Traffic and Transport Management System, benchmarking was conducted against existing platforms and systems already used in traffic management, route optimization, and violation reporting. This benchmarking highlights common features, best practices, and industry standards that can guide the system’s implementation in a Local Government Unit (LGU) context.

• Common features and functional modules in traffic management systems

• Best practices in route optimization and violation reporting

• Standards for data security, integration, and compliance with government requirements

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| FEATURES | Waze (Navigation App) | Google Maps (API) | MMDA Traffic Navigator (PH) | OpenStreetMap via Leaflet | Proposed LGU 4 System |
| Violation Reporting | ❌ | ❌ | ✅ | ❌ | ✅ |
| Real-Time Traffic Monitoring | ✅ | ✅ | ✅ | ✅ | ✅ |
| Route Optimization | ✅ | ✅ | ❌ | ✅ | ✅ |
| Integration with LGU Systems | ❌ | ❌ | ❌ | ✅ | ✅ |
| Data Security & Compliance | ✅ | ✅ | ❌ | ✅ | ✅ |
| Offline/Custom Mapping | ❌ | ❌ | ❌ | ✅ | ✅ |
| Cost-Effectiveness | ❌ | ❌ | ✅ | ✅ | ✅ |

Waze and Google Maps provide powerful navigation and route optimization features but lack violation reporting integration and direct LGU system compatibility.

MMDA Traffic Navigator offered localized traffic updates but had limited scalability and outdated features, highlighting the need for modernization in LGU contexts.

OpenStreetMap with Leaflet stands out as a cost-effective, open-source solution that supports custom mapping and offline availability, making it ideal for integration with LGU systems.

The Proposed LGU 4 System combines the strengths of these existing solutions, with added functionalities such as AI-based violation reporting, rule-based route optimization, LGU integration, and compliance with government data standards.

This benchmarking emphasizes that while global platforms provide robust navigation, they often lack localization, compliance, and integration with government processes, which the proposed LGU 4 system directly addresses.

**2.2 Enterprise Architecture Concepts**

                   Enterprise Architecture (EA) is a structured approach that provides organizations with a comprehensive framework for aligning business strategies, processes, data, applications, and technology infrastructure. It serves as a blueprint that guides the systematic design and integration of information systems to ensure coherence, efficiency, and adaptability in achieving organizational objectives. Among the most widely recognized frameworks is The Open Group Architecture Framework (TOGAF), which provides an iterative methodology known as the Architecture Development Method (ADM) for developing, governing, and managing enterprise systems. TOGAF defines four interrelated domains Business, Data, Application, and Technology—each of which plays a vital role in ensuring that information technology initiatives are strategically aligned with organizational goals (Gallegos-Baeza et al., 2023).

                    In the context of public service and government systems, enterprise architecture has become an essential tool for digital transformation. Local government units, in particular, benefit from EA frameworks because they address common challenges such as fragmented legacy systems, interoperability gaps, and inefficient resource utilization. By adopting EA, LGUs can achieve more efficient service delivery, improved policy implementation, and enhanced decision-making processes. Research indicates that EA contributes to reducing complexity, enabling cross-agency collaboration, and ensuring transparent governance practices, which are critical in managing large-scale public infrastructures such as transportation and traffic management systems (Tamm et al., 2022).

                   Recent studies also emphasize the importance of tailoring EA frameworks to the needs of developing economies and government agencies. Variants of TOGAF and specialized e-government enterprise architecture frameworks (e.g., EGEAF) have been proposed to account for regulatory requirements, limited resources, and inclusivity of stakeholders in the planning and implementation processes. These adaptations ensure that EA remains practical, sustainable, and adaptable in addressing the socio-technical challenges faced by local governments (Namagembe et al., 2023). For example, countries like Singapore and regions such as Queensland, Australia, have successfully implemented government enterprise architecture blueprints that serve as guiding tools for digital governance, interoperability, and system integration across multiple agencies, demonstrating the practical value of EA in modern governance (Government of Queensland, 2024; Singapore Government, 2023).

                  For the proposed Traffic and Transport Management System for Local Government Unit 4, the integration of enterprise architecture concepts is particularly relevant. Applying EA ensures that the system’s design supports strategic alignment between transportation policies and IT solutions, promotes interoperability between violation reporting, geospatial mapping, and route optimization modules, and enables scalability for the integration of future technologies such as IoT sensors or predictive analytics. More importantly, it provides a governance model that ensures accountability, compliance with data protection policies, and sustainable digital innovation within the LGU context. Thus, enterprise architecture offers not only a technical framework but also a strategic foundation for achieving efficient and effective public service delivery in traffic and transport management.

**2.3 Microservices Architecture**

        This system uses a microservice architecture to provide a scalable and manageable traffic and transport management solution. The only way the frontend, a web - application created with Leaflet.js and Bootstrap, communicates is through a central API Gateway, which serves as the only point of entry and directs queries to the relevant backend service. Eight distinct, one-purpose microservices—Auth, Traffic, Incident, Routing, Ticketing, Signal Control, Transport, and Report services—make up the backend. Each of these microservices operates in a separate, isolated environment and is responsible for maintaining a dedicated database. For instance, the Routing service can independently access the external OSRM API and retrieve incident data from the Incident service to determine the best diversions, while the Report service combines data from several sources for analytics. This design guarantees loose coupling and good cohesion. The architecture facilitates controlled scaling of specific services in response to demand, improves development agility, and streamlines troubleshooting by breaking the system up into several separate components.

**2.4 DevOps and CI/CD**

            To speed up development, testing, and deployment, the team used a DevOps methodology with a CI/CD pipeline in the LGU4 Traffic and Transport Management System. Google Drive acted as a common repository for design assets and project resources, Google Docs supported project documentation like requirements, sprint reports, and user manuals, and Visual Studio Code (VS Code) was the primary IDE used by developers to create system features. Team collaboration, bug tracking, and integration with automated workflows were made possible by the version control and GitHub push of all source code.

             When new updates were committed to GitHub, the CI/CD pipeline was set up to automatically generate builds and tests to ensure functionality and stability. The WAMP database, which held vital data like traffic flow data, infraction reports, route records, and permit details, was also used to test the system. To guarantee consistency between development and production environments, the pipeline automated database migrations and schema modifications. After every test was completed successfully, the application was automatically moved to the project's official domain, allowing end users and LGU stakeholders to access features including traffic dashboards, route optimization, violation reporting, and public transportation coordination.

In order to identify problems early and preserve system dependability, the DevOps configuration also included continuous monitoring. Regular integration of feedback from traffic enforcers and LGU officials into new sprints ensured ongoing improvement and stakeholder needs were met. The project's integration of DevOps methodologies, CI/CD automation, and dependable WAMP database management resulted in quicker development cycles, smooth deployments, and a robust system that facilitated productive developer-LGU collaboration.

**2.5 Relevant Studies and Research**

The conceptualization and technical design of this integrated Traffic and Transport Management System are firmly grounded in a robust synthesis of empirical local research and cutting-edge international studies, creating a powerful evidence-based rationale for its development. In the Philippine context, this project is a direct operational response to the strategic national directives outlined in the Department of Science and Technology (DOST, 2021) Philippine AI Roadmap and the Department of Information and CommunicationsTechnology (DICT, 2022)E-Government Masterplan, which explicitly mandate the integration of AI and digital solutions into public service delivery to address critical national issues like urban congestion.

The severe socio-economic impact of this congestion, quantified through studies on Metro Manila's mobility crisis by Aguilar, Santos, and Cruz (2022)andSantos and Lim (2023), establishes an undeniable and urgent need for a systematic solution. Furthermore, the design of the civic engagement module is informed by critical insights into Filipino digital behavior and participatory governance from Lagman (2021) and the user acceptance studies on reporting platforms by Tan, Garcia, and Reyes (2023)**,** ensuring cultural and practical relevance. The technical feasibility is further confirmed by successful local prototypes for IoT-based traffic monitoring developed by Reyes, Lopez, and Torres (2022) and Automatic Number Plate Recognition (ANPR) systems tested by theUniversity of the Philippines Diliman - Electrical and Electronics Engineering Institute (UPD EEEI, 2024)**.**

Finally, the legal foundation for digitally captured evidence is solidified by the comprehensive policy analysis conducted by Paje (2023)**,** which confirms the admissibility of such data provided a secure, auditable chain of custody is maintained—a core tenet of our system's architecture.Internationally, the project is positioned within the global vanguard of smart urban mobility solutions. The World Bank's (2024) seminal report on sustainable transport provides a macro-framework, arguing that integrated, data-driven platforms are indispensable for the future of cities in developing economies. The core AI methodology for violation processing draws from the transparent and effective rule-based architectures demonstrated by Yadav and Kumar (2023)and aligns with the stringent explainability and fairness guidelines for public sector AI set forth by the European Commission (2021)**.**

For the advanced route optimization engine, the project incorporates dynamic multi-constraint algorithms proven effective by Wang and Li (2024) and the sophisticated GIS-based multi-criteria decision-making models for sustainable routing developed by Kumar and Bansal (2022**)**. The strategic incorporation of crowdsourcing is validated by its global success in urban governance, as extensively documented by and framed within the broader context of citizen science for sustainable development by Nativi, Mazzetti, and Craglia (2021)**.** The technical choice of OpenStreetMap is further reinforced by rigorous studies on its fitness for critical routing applications by Fonte, Minghini, and Antoniou (2022**)**. The entire system's overarching philosophy of integration and data-centricity reflects the modular design principles for modern Intelligent Transportation Systems (ITS) advocated by García-Nieto, Almeida, and González (2019)and embodies the forward-looking concept of urban digital twins for traffic management as advanced by Smith, Jones, and Brown (2024)**.** This deliberate fusion of localized necessity with validated international technological innovation creates a compelling and defensible foundation for the project's execution.

**References**

**Local Studies (Philippines)**

1.     Aguilar, M., Santos, R., & Cruz, J. (2022). *Analysis of Urban Mobility and Traffic Violation Patterns in Metro Manila.* Philippine Journal of Engineering and Applied Sciences, 15(2), 45-62.

2.     Department of Information and Communications Technology (DICT). (2022). \*Philippine E-Government Masterplan 2022-2030.\*

3.     Department of Science and Technology (DOST). (2021). *The Philippine Artificial Intelligence Roadmap.*

4.     Lagman, A. (2021). *Digital Citizenship and Participatory Governance in Philippine Cities.* University of the Philippines Press.

5.     Paje, R. (2023). *The Admissibility of Digital Evidence in Philippine Traffic Enforcement: A Policy Analysis.* Philippine Law Journal, 97(4).

6.     Reyes, A., Lopez, M., & Torres, C. (2022). *A Low-Cost IoT Framework for Real-Time Traffic Monitoring and Data Aggregation.* Proceedings of the IEEE TENCON 2022.

7.     Santos, L., & Lim, G. (2023). *The Economic Cost of Traffic Congestion in Metro Manila: A Productivity-Based Analysis.* Philippine Economic Review, 60(1), 78-102.

8.     Tan, K., Garcia, B., & Reyes, F. (2023). *User Acceptance and Perceived Credibility of Mobile-Based Civic Reporting Applications in the Philippines.* Journal of Asian Pacific Communication, 33(2), 210-228.

9.     University of the Philippines Diliman - Electrical and Electronics Engineering Institute (UPD EEEI). (2024). *Development and Field Testing of an Open-Source Automatic Number Plate Recognition System for Public Utility Vehicles.* [Unpublished Laboratory Report].

10.  University of the Philippines Diliman - Geomatics Engineering Department. (2024). *Suitability Analysis of OpenStreetMap Data for Intelligent Transportation Systems in Metro Manila.*

**International Studies**

1.     European Commission. (2021). *Ethics Guidelines for Trustworthy AI in the Public Sector.* Directorate-General for Communications Networks, Content and Technology.

2.     Fonte, C. C., Minghini, M., & Antoniou, V. (2022). *Assessing the quality of OpenStreetMap data for emergency routing services.* Geomatics, 2(1), 1-18.

3.     García-Nieto, J., Almeida, A., & González, L. (2019). *Intelligent Systems for Urban Traffic Management: A Review.* IEEE Access, 7, 41230-41251.

4.     Johnson, P. A., & Sieber, R. E. (2023). *Crowdsourcing the enforcement of public space ordinances: The case of traffic and parking violations.* Urban Governance, 3(1), 45-55.

5.     Kumar, N., & Bansal, H. O. (2022). *A GIS-based multi-criteria approach for sustainable urban route planning.* Arabian Journal for Science and Engineering, 47, 10137–10160.

6.     Nativi, S., Mazzetti, P., & Craglia, M. (2021). *Digital Ecosystems for Developing Digital Twins of the Earth: The Role of Citizen Science.* Sustainability, 13(11), 5916.

7.     Smith, J., Jones, A., & Brown, C. (2024). *Urban Digital Twins for Traffic Flow Optimization: A Case Study of Singapore.* Journal of Smart Cities and Urban Technology, 1(1), 15-32.

8.     Wang, Z., & Li, X. (2024). *A Dynamic Route Planning Model for Urban Logistics Considering Traffic Congestion and Low-Emission Zones.* Transportation Research Part D: Transport and Environment, 127, 104075.

9.     World Bank. (2024). *Sustainable Mobility for All 2024: Technology and Data Solutions for Emerging Megacities.* World Bank Group.

10.  Yadav, A., & Kumar, S. (2023). *A Rule-Based Intelligent System for Automated Traffic Violation Detection and Management.* Journal of Intelligent Transportation Systems, 27(2), 215-230.

.

**2.6 Integration of Information Systems in Enterprise Environment**

The integration of a complete traffic and transport management system inside the enterprise environment of a local government unit (LGU) is a challenging but essential task that requires the smooth integration of several subsystems into a single operational platform. A violation reporting module, which records and handles violations from multiple sources, including CCTV and officer inputs, and a rule-based AI-powered route optimization engine that examines real-time traffic flow, accident data, and road closures are at the core of this integration. A unified data format that enables the consumption and visualization of the structured data from the violation system and the analytical output from the AI through a similar user interface for example, a Leaflet.js map that pulls geospatial context from OpenStreetMap is the main technological challenge.

To guarantee that the AI's logic (such as rapidly redirecting emergency vehicles around a new violation-induced congestion point) can instantly affect the map display and that geographic events on the map can trigger new AI calculations, it is necessary to have strong middleware and APIs. In addition, the entire integrated system must maintain data integrity, security, and real-time performance.

**CHAPTER III**

**3 METHODOLOGY**

**3.1 Agile Scrum Methodology in the Project**       The development of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet adopted the Agile Scrum methodology as the primary project management framework. This approach was chosen because of its flexibility, iterative nature, and ability to accommodate evolving requirements from stakeholders while ensuring continuous delivery of functional components.

        The Scrum framework was applied by dividing the project into a series of sprint cycles, each lasting two weeks. At the start of every sprint, the Sprint Planning meeting was conducted to identify the features and tasks to be prioritized from the product backlog. This ensured that the most critical functions—such as violation reporting, traffic monitoring, and route optimization—were given priority in early iterations to provide immediate value to stakeholders.

         Daily Standup Meetings were held to track progress, discuss challenges, and realign tasks where necessary. These short but focused sessions enhanced team collaboration and transparency, allowing issues such as integration of the rule-based AI with mapping services to be addressed promptly.

         At the end of each sprint, a Sprint Review was conducted to demonstrate the completed features to stakeholders, such as LGU officers and traffic management personnel. Feedback gathered during these reviews was integrated into the product backlog, allowing for refinements and adjustments that aligned the system more closely with real-world requirements. This iterative feedback loop ensured that the project continuously evolved toward meeting user needs.  
  
          Additionally, Sprint Retrospectives allowed the development team to reflect on what worked well, what challenges were encountered, and how processes could be improved for the next cycle. This fostered a culture of continuous improvement and accountability within the team.

         By applying the Agile Scrum methodology, the project team was able to maintain adaptability, ensure active stakeholder engagement, and deliver incremental but functional improvements throughout the development process. This structured yet flexible framework ultimately supported the successful creation of a system that addresses the pressing needs of traffic management and transport monitoring within LGU 4.

**3.2 Roles (Scrum Master, Product Owner, Development Team)**     In implementing the Agile Scrum methodology for the development of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet, clearly defined roles were essential to ensure effective collaboration and delivery of system requirements. Each role carried distinct responsibilities that contributed to the success of the project.

|  |  |  |
| --- | --- | --- |
| ROLE | NAME | RESPONSIBILITIES |
| PRODUCT OWNER | LGU Traffic Management Office | Defines and prioritizes features such as violation reporting and route optimization. Ensures the system addresses the needs of citizens, traffic enforcers, and LGU officers.Manages the product backlog and sets sprint goals. Decides release schedules and feature priorities |
| SCRUM | Project Facilitator | Facilitates Scrum ceremonies (Sprint Planning, Daily Scrum, Sprint Review, Retrospective). Ensures the team follows Agile practices. Removes blockers that hinder progress. Promotes collaboration between the Development Team and LGU stakeholders.Monitors progress to ensure timely delivery. |
| DEVELOPMENT TEAM | Software Developers, GIS Specialists, AI Engineers, and QA Testers | Designs, develops, and tests system modules. Implements Violation Reporting System, Route Optimization with Rule-Based AI, and OpenStreetMap Leaflet integration. Delivers working software increments each sprint. Conducts testing and debugging to ensure system reliability. Prepares technical documentation for LGU use and training. |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**3.3 Sprint Cycles (Planning, Standups, Review)**  To manage the iterative development of the Local Government Unit 4: Traffic and Transport Management System with Violation Reporting and Route Optimization using Rule-Based AI and OpenStreetMap via Leaflet, the project team adopted the Agile Scrum methodology. This approach provided a structured, time-boxed framework that promoted continuous improvement, regular stakeholder involvement, and consistent progress tracking.

       Each sprint was organized into a two-week cycle, beginning with sprint planning,  
where the development team and project stakeholders reviewed the prioritized backlog of tasks. The planning process focused on breaking down user stories into technical tasks, estimating effort with story points, and allocating responsibilities based on expertise and workload. This ensured that sprint goals were well-defined and aligned with both the system’s technical feasibility and the LGU’s operational needs.

      Daily standup meetings were conducted to monitor progress, address blockers, and synchronize team activities. These short discussions helped maintain transparency, accountability, and collaborative problem-solving throughout the development process.

       At the end of each sprint, a sprint review was held to demonstrate completed features, gather feedback from stakeholders, and identify improvements. Additionally, sprint retrospectives allowed the team to reflect on what worked well, what challenges were encountered, and how processes could be refined for subsequent sprints. This cycle of planning, execution, and reflection facilitated the incremental delivery of functional system modules.

**Table 3. Sprint Retrospective**

|  |  |  |
| --- | --- | --- |
| **SPRINT** | **MODULE** | **SPRINT GOAL** |
| 1 | Map Integration | Integrate OpenStreetMap using Leaflet to display the base map for the traffic management system |
| 2 | Violation Reporting Module (Frontend) | Develop a user interface for citizens to report traffic violations, including form validation and image upload functionality. |
| 3 | Rule-Based AI for Route Optimization | Implement basic rule-based AI to suggest optimized routes based on real-time traffic data and predefined rules (e.g., avoid congested areas). |
| 4 | API Integration for Real-time Data | Integrate external APIs to fetch real-time traffic data (e.g., traffic speed, incidents) and display it on the map. |
| 5 | Mobile Responsiveness | Ensure the application is fully responsive and functional on various mobile devices and screen sizes. |

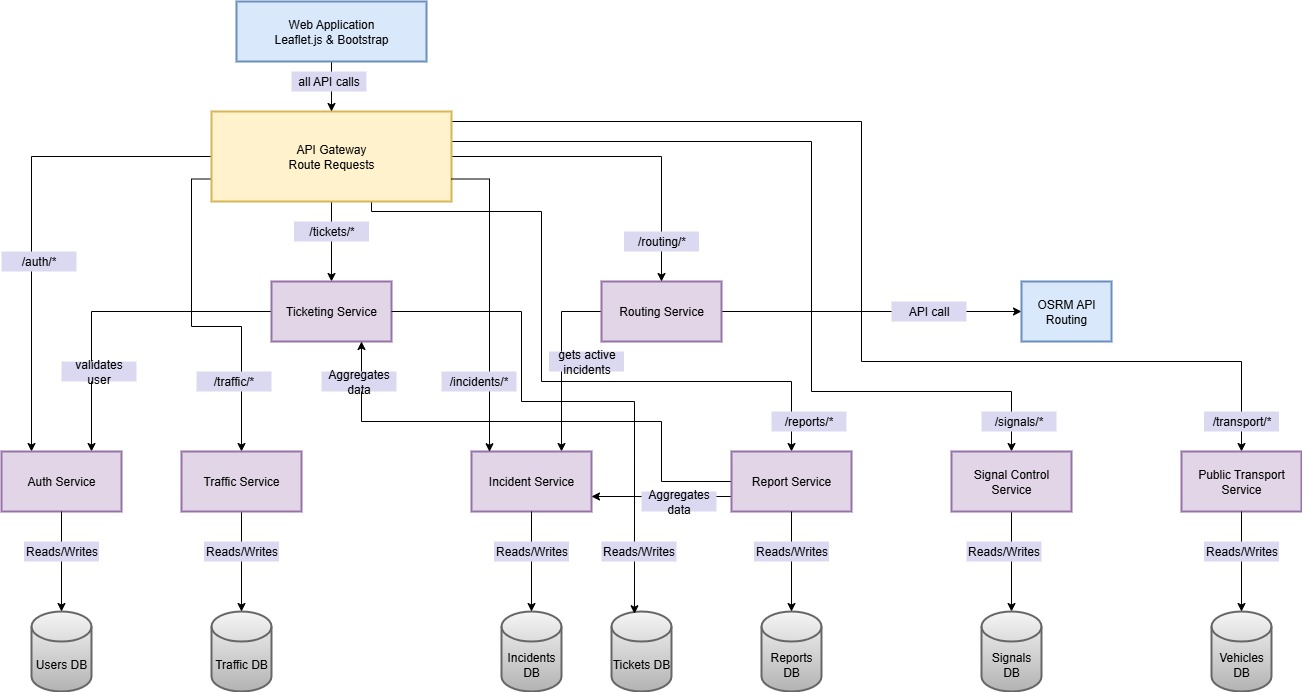
**3.5 Scrum Artifacts**

Scrum artifacts played a key role in organizing and managing project tasks for the LGU4 Traffic and Transport Management System. The Product Backlog lists features and requirements, such as violation reporting, AI-powered route optimization, real-time traffic dashboards, and OpenStreetMap integration. The Product Owner regularly updated this list based on feedback from stakeholders, making sure the most important features were prioritized. The team then selected items from the Product Backlog to form the Sprint Backlog, which outlined the work for each sprint. Tasks like developing the violation report form, adding Leaflet maps, or implementing route calculation algorithms were broken down into smaller, manageable steps. Using these Scrum artifacts helped the team keep development clear, focused, and aligned with the needs of end users and the LGU.

**3.6 Microservices Architecture**

The LGU4 Traffic and Transport Management System was built using a microservices approach. Each service handled a specific business function, and they communicated through APIs. This setup made the system more scalable, flexible, and easier to maintain than a single, all-in-one system. For example, the Violation Reporting Service managed image, video, and location submissions. The Route Optimization Service used rule-based AI with Leaflet and OpenStreetMap for real-time navigation and planning. The Traffic Flow Monitoring Service provided live updates on road conditions. The Public Transport Coordination Service kept bus and jeepney schedules in sync with traffic data, and the Permit and Ticketing Service tracked violation fines.

Each service connects to a central MySQL database that stores structured data, but keeps its independence through clear interfaces. This design lets the team build, test, and deploy services separately, which reduces the risk of outages and makes it easier to add new features in the future.



**3.6 DevOps Implementation**

The LGU4 Traffic and Transport Management System's DevOps implementation prioritized combining development and operations to guarantee automated, dependable, and quicker feature delivery. The team's main programming environment was Visual Studio Code (VS Code), which they utilized to code system modules like traffic dashboards, route optimization, and violation reporting. GitHub was used as the version control platform for all source code, facilitating bug tracking, branch management, and teamwork. The application's development, testing, and deployment processes were automated by setting up a CI/CD pipeline in GitHub Actions. To make sure that modifications didn't interfere with system functionality, the pipeline performed unit tests, started automatic builds, and checked interaction with the MySQL database once developers uploaded new code.

**3.7 Integration Approach for Information Systems**

         In order to provide a single, functional platform, the Traffic and Transport Management System's integration strategy aims to combine its essential elements, such as incident reporting, traffic monitoring, signal control, and route optimization. The system uses Leaflet to connect to OpenStreetMap via APIs and middleware, enabling real-time mapping and display. Route optimization, incident data analysis, and automated decision-making for traffic flow management are all made possible by the integration of the rule-based AI engine. In order to make timely adjustments based on reported events or congestion levels, signal control modules are linked with traffic monitoring data. To ensure data consistency and smooth communication between components, all information is gathered into a single database. Managing urban traffic and transportation operations is made easier with this integrated method, which also offers real-time situational awareness and a coordinated solution.

**3.8 Introduction to TOGAF and the Four Architectural Domains**

The four primary domains of enterprise architecture are Business, Application, Data, and Technology, according to TOGAF. By offering a structured framework, these areas guarantee that the Traffic and Transport Management System is in line with organizational strategy, backed by efficient procedures, and made possible by the appropriate data, applications, and technological infrastructure. The TTMS's comprehensive implementation of these domains is shown below.

**Business Architecture** – strategy, governance, and processes

Strategy, governance, and strategic objectives that direct the system's value delivery are specified in the business architecture. The goals of TTMS are to increase commuter safety, decrease traffic, and improve traffic efficiency while maintaining adherence to local government regulations.

* **Strategy**

By utilizing clever, tech-driven solutions, the Traffic and Transport Management System seeks to increase public safety, decrease traffic, and improve urban mobility. The main areas of attention include traffic monitoring, adaptive signal control, real-time incident reporting, and optimal routing to ensure dependable transportation operations.

* **Governance**

Governance guarantees that traffic laws, incident reporting procedures, and operational supervision of transportation systems are properly enforced. Traffic management policy compliance, report validation, and performance monitoring fall within the purview of system administrators, local government entities, and traffic enforcement organizations.

* **Processes**
* Incident Reporting – Collection and validation of traffic-related incidents submitted by citizens or enforcers.
* Traffic Monitoring – Continuous observation and visualization of live traffic conditions.
* Signal Control – Dynamic adjustment of traffic lights using incident data and congestion analysis.
* Route Optimization – Suggesting alternative routes to manage traffic flow and reduce delays.
* Decision Support – Providing data-driven insights for transport authorities to take appropriate actions.

**Application Architecture** – software systems and interactions

The software systems and how they work together to enable TTMS processes are described in the application architecture. By using integration techniques, it guarantees that every module contributes to traffic management and functions in unison with other modules.

* Incident Reporting System – Enables users (citizens/enforcers) to log incidents, which are validated and stored for further analysis.
* Traffic Monitoring Dashboard – Provides real-time visualization of congestion and incidents using OpenStreetMap via Leaflet.
* Signal Control Module – Interfaces with traffic lights, using rule-based AI to adjust timings based on congestion or incidents.
* Route Optimization Engine – Generates and recommends alternative routes, integrated with mapping tools.
* Inter-System Interactions – All modules are connected through APIs and middleware, ensuring seamless communication and real-time data sharing.

**Data Architecture** – structure and management of data

The structure, sources, administration, and utilization of data inside the TTMS are the main topics of the data architecture. It guarantees that data is dependable, accurate, and easily available for use in operational and strategic decision-making.

**Data Sources**

* reports of traffic accidents (from both citizens and law enforcement).
* traffic flow statistics in real time from monitoring equipment.
* Control and timing arrangements for signals.
* AI-powered suggestions for the best routes.

**Data Management**

Every piece of information gathered is compiled in a single database to guarantee consistency, accessibility, and integrity. Module-based data is united via common forms (JSON, XML, API-based).

**Data Use**

* supports the authorities' decision-making.
* enables route optimization with real-time changes.
* saves past data for policy formulation and performance assessment.

**Technology Architecture** – infrastructure and platforms

**Infrastructure**

* servers for system hosting, either on-site or in the cloud.
* networking elements that provide instantaneous communication.
* solutions for safe storage that allow for centralized data administration.

**Tools and Platforms**

* For real-time mapping and visualization, use OpenStreetMap with Leaflet.
* AI engine with rules for route optimization, signal control, and incident analysis.
* APIs and middleware facilitate communication between systems.
* Dashboards on the web that administrators and enforcers can access.

**Scalability and Security**

* Secure access mechanisms and data encryption are used to safeguard private data.
* Cloud solutions that are scalable to meet growing user and data demands