Title:Traffic Forecasting For Road Network

Subtitle: Optimizing Routes in London and University of Roehampton

By

Uzum Stanley Ekene

Submitted to

The University of Roehampton

In partial fulfilment of the requirements
for the degree of

Master of Science

in

Data Science

Declaration

I hereby certify that this report constitutes my own work, that where the language of others is used, quotation marks so indicate, and that appropriate credit is given where I have used the language, ideas, expressions, or writings of others.

I declare that this report describes the original work that has not been previously presented for the award of any other degree of any other institution.

Uzum Stanley Ekene

19-8-2024

Acknowledgements

Here, it is customary to thank the people who have supported this work and your studies in general. It is up to you who you thank!

Abstract

Traffic congestion in London leads to significant time delays and economic losses, while navigating specific locations, such as the University of Roehampton campus, poses challenges, particularly for newcomers. These issues affect a wide range of users, including commuters, drivers, students, staff, and visitors. Addressing these challenges is crucial to enhancing quality of life, reducing travel costs, and improving the efficiency of transportation systems.

This research presents the development and implementation of the "London Navigator," a traffic forecasting and route optimization system designed specifically for the complex road networks of London and the University of Roehampton. Leveraging Dijkstra's algorithm, renowned for its efficiency in identifying the shortest paths in graph-based networks, the system is tailored to handle the extensive and intricate roadways of London, offering a more practical solution compared to alternatives like the Bellman-Ford algorithm.

The literature review underscores the effectiveness of Dijkstra's algorithm in various transportation and urban planning applications, highlighting its suitability for solving the Vehicle Routing Problem (VRP). Existing navigation tools, such as Google Maps, often fail to provide optimal routing in specialized environments like university campuses or dense urban areas. To address these limitations, this project integrates Dijkstra's algorithm with Geographic Information Systems (GIS) and open-source technologies, creating a solution that enhances navigation within these specific contexts.

The implementation of the London Navigator utilizes open-source tools including MapServer, Leaflet, and PostgreSQL, chosen for their cost-effectiveness and robust capabilities. The system is designed using an Object-Oriented (OO) methodology, ensuring a user-friendly interface accessible to decision-makers without technical backgrounds. The application, built on a web platform, features an intuitive interface with a main window and toolbar, catering to users with moderate computer skills.

The system is structured into two main subsystems: one for managing road network data and another for displaying geographic information. This design ensures functionality and adaptability, with potential for future enhancements such as GPS tracking integration and congestion indicators. These planned features will provide real-time traffic updates and

suggest alternative routes to avoid congestion, further optimizing travel times and improving user experience.

In conclusion, the London Navigator successfully addresses the limitations of existing navigation tools by delivering precise and efficient route optimization tailored to the challenges of London's road network and the University of Roehampton campus. The project demonstrates significant potential for reducing travel times, improving navigation efficiency, and enhancing the overall transportation experience for users. Future work will focus on expanding the system's capabilities, making it an even more valuable tool for urban traffic management and personal navigation.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Traffic congestion and navigation inefficiency are pressing issues in metropolitan areas such as London. Addressing these challenges can save time and reduce economic losses significantly. According to the INRIX 2022 Global Traffic Scorecard, London is the most congested city globally, with drivers losing an average of 156 hours in traffic annually. In the UK, the average driver lost 80 hours to congestion, reflecting a 7-hour increase from 2021, though still 35 hours less than in 2019. The annual cost of congestion in London amounted to £1,377 per driver, with UK drivers losing an average of £707. Additionally, rising fuel costs added £212 for London commuters and £122 nationally [1].

Globally, drivers in the top five congested cities were: London (156 hours), Chicago (155 hours), Paris (138 hours), Boston (134 hours), and New York (117 hours). Most UK cities, including London, have met or exceeded pre-COVID congestion levels, with Londoners experiencing a 5% increase over pre-pandemic delays. This surge in traffic congestion underscores the urgent need for effective traffic management to mitigate economic losses and improve the quality of life for commuters [1].

A recent report from the leading Global Positioning System (GPS) company TomTom highlighted that traffic congestion results in significant losses in time and energy consumption, with commuters in major cities worldwide spending approximately 75% more time traveling than necessary [2]. The rapid increase in the number of vehicles and the high percentage of the global population living in urban areas are key factors contributing to this issue. Enhancing traffic congestion management and implementing optimal route-planning algorithms could benefit over 3 billion people by reducing energy consumption and aiding in mitigating global warming. One effective strategy to reduce urban traffic is optimizing traffic flow for more efficient routing. An improved vehicle routing system would lead to better utilization of all nearby road networks [3].

Maps are fundamental tools for presenting and analyzing information on the spatial distribution of business sectors, resources, and individuals requiring services, as well as for determining locations. Woodbury (1996) noted that 85% of all computerized databases globally include a location component, such as a street address, zip code, census tract, or legal description. Geographic Information System (GIS) technology is currently one of the most prominent new research tools and one of the fastest-growing high-tech monitoring systems [4].

Analyzing road networks is crucial for generating accurate and effective information about roads, aiding decision-makers in selecting optimal routes through mathematical calculations involving dynamic programming [5, 6]. Determining the optimal route is a prevalent research topic in transportation literature [7]. Most technologies used in today's applications, such as Google Maps, which assist vehicle drivers in choosing the best route—whether it is the shortest distance, least time, or most economical—utilize Dijkstra's algorithm [8].

By improving traffic management, cities can reduce economic losses and enhance productivity. This project aims to develop a system that provides optimal routes for travelers within London and aids in effortless navigation within the University of Roehampton campus. The system leverages Dijkstra's algorithm to determine the shortest route from the traveler's starting point to their destination. By offering real-time, optimized routing solutions, the system addresses traffic congestion problems by directing travelers through the shortest paths, thereby reducing travel time and enhancing overall efficiency.

1.1 Problem Description, Context and Motivation

Traffic congestion in London leads to significant time and economic losses. The INRIX 2022 Global Traffic Scorecard reports that the average London driver lost £1,377 in time due to congestion, severely impacting productivity and economic efficiency. Additionally, navigating the University of Roehampton campus presents challenges for newcomers, including students, visitors, and newly employed staff, who may struggle to find optimal routes.

The primary individuals affected are London drivers and commuters facing daily delays and increased travel costs, as well as University of Roehampton students and visitors experiencing navigation difficulties, particularly during peak times such as the start of semesters.

Solving London's traffic congestion is crucial for reducing travel costs and time, enhancing quality of life and productivity for commuters. Effective traffic management can lead to substantial economic savings and a more efficient transportation system. Similarly, improving campus navigation at the University of Roehampton is important for providing a positive experience for students, staff, and visitors, facilitating easier access to campus facilities and reducing time spent navigating the campus.

1.2 Objectives

The primary aim of this research is to develop an optimized navigation system for the London road network and the University of Roehampton campus using a path-finding algorithm. The specific objectives are as follows:

1. To map the London route network as a graph incorporating both drivable roads and footpaths:

This objective involves creating a comprehensive graph representation of the London road network, including both drivable roads and pedestrian pathways. This detailed mapping will provide the foundation for the navigation system, enabling accurate and efficient route planning.

2. To develop a system to effectively implement a path-finding algorithm to improve navigation efficiency:

This objective focuses on the implementation of Dijkstra's algorithm to find the shortest and most efficient routes within the mapped network. The system will dynamically adjust routes based on real-time traffic data to minimize travel time and reduce congestion.

3. To design a user-friendly web interface for the navigation system:

This objective aims to create an intuitive and accessible web interface for users. The interface will allow travelers to easily input their starting points and destinations, view optimal routes, and receive real-time updates on fastest paths.

1.3 Methodology

This section details the methodology that will be employed to achieve the objectives of this research project. It includes the design, testing and evaluation, project management, and the technologies and processes used.

Design

The design phase involves several key steps:

1. Data Collection:

- Obtain London road data from Transport for London (TFL). This data will provide a comprehensive overview of the road network in London, including both drivable roads and pedestrian paths.

- Digitize campus routes using Google Earth and QGIS. These tools will be used to create an accurate map of the University of Roehampton campus, including all pathways and routes.

2. Graph Modeling:

- The graph network will be digitized and cleaned using QGIS. It will be modeled with Python to represent the London road network as a graph with nodes and edges, ready for further processing.

3. Graph Densification:

- The graph will be densified by adding intermediate nodes and refining edges. This will improve the accuracy and efficiency of the path-finding algorithm, resulting in a detailed and accurate representation of the road network.

4. Path-Finding Algorithm Implementation:

- Dijkstra’s algorithm will be implemented in Python, Go, and JavaScript. It will calculate the shortest paths on the graph and be tested on smaller sections of the graph network before scaling to the entire area.

5. Validation:

- The computed paths will be validated by comparing them with real-world navigation scenarios. User tests will be conducted where participants follow the suggested routes and provide feedback to enhance the system's reliability.

Testing and Evaluation

Testing and evaluation will involve the following steps:

1. System Testing:

- Initial tests will be conducted on smaller sections of the graph network to ensure the path-finding algorithm works correctly.

2. User Testing:

- Participants will use the system to navigate both London roads and the University of Roehampton campus. Their feedback will be used to make necessary adjustments.

3. Comparison with Real-World Scenarios:

- The computed paths will be compared with actual navigation scenarios to validate the accuracy and efficiency of the system.

Project Management

Project management will employ agile methodologies, the use of a Kanban board to track progress. The project management details can be viewed at GitHubProject(https://github.com/users/uzumstanley/projects/1/views/1).

Technologies and Processes

The following technologies and processes will be used:

1. Data Collection and Graph Modeling:

- Tools: QGIS, Google Earth, Google Maps
- Language: Python
- Purpose: To digitize and clean the road network data and model it as a graph.

2. Graph Densification:

- Tool: Python

- Purpose: To add intermediate nodes and refine edges for improved path-finding accuracy.

3. Path-Finding Algorithm Implementation:

- Languages: Python, Go, JavaScript
- Algorithm: Dijkstra’s algorithm
- Purpose: To calculate the shortest paths on the graph.

4. Database Design:

- Tool: PostgreSQL with PostGIS
- Purpose: To store and manage spatial and attribute data efficiently.

5. Backend Development:

- Purpose: To handle data processing, pathfinding algorithms, and database interactions.

6. Web Development:

- Tool: React
- Purpose: To create a user-friendly web interface for displaying optimal routes on a digital map of London.

7. Integration of Custom Tile Map Service:

- Purpose: To ensure seamless and efficient map rendering using high-resolution drone imagery.

Why These Methods?

(i). Dijkstra's Algorithm:

- Proven efficiency in finding the shortest path.

- Widely used in interactive maps with extensive documentation and support.

(ii). PostgreSQL with PostGIS:

- Provides robust and scalable data management for spatial data.

(iii). React:

- Offers a responsive and interactive web interface for users.

1.4 Legal, Social, Ethical and Professional Considerations

Legal Considerations

Developing a navigation system for public use involves several legal considerations, including data privacy, data security, and compliance with local regulations:

1. Data Privacy and Protection:

- The system will handle sensitive data related to users' locations and travel patterns. Compliance with data protection regulations such as the General Data Protection Regulation (GDPR) is crucial. User data will be anonymized and encrypted to ensure privacy and security.

- Explicit consent will be obtained from users before collecting any personal data.

2. Intellectual Property:

- Ensuring that all software components, including the path-finding algorithm and web interface, respect intellectual property rights. Open-source libraries and tools will be used in compliance with their respective licenses.

3. Compliance with Local Traffic Laws:

- The system will provide route suggestions in line with local traffic regulations, avoiding restricted or prohibited areas.

Social Considerations

The project aims to positively impact society by reducing traffic congestion and improving navigation efficiency. Key social considerations include:

1. Accessibility:

- Designing the system to be inclusive and accessible to all users, including those with disabilities. This includes ensuring the web interface is compliant with accessibility standards (e.g., WCAG 2.1).

2. Public Awareness and Acceptance:

- Educating users about the benefits of the system and encouraging its adoption through public awareness campaigns. Engaging with local communities to gather feedback and improve the system based on user experiences.

Ethical Considerations

Ethical considerations in this project primarily involve the responsible use of data and ensuring the system's benefits are equitably distributed:

1. Responsible Data Usage:

- Ensuring that data collected from users is used solely for the purpose of improving navigation efficiency and not for any unauthorized or unethical purposes.

2. Equity and Fairness:

- Ensuring that the system benefits all users equally and does not favor certain groups over others. Special attention will be given to providing accurate and efficient routes for all areas, including less affluent neighborhoods.

Professional Considerations

Professionalism in the development and deployment of the navigation system includes adhering to industry standards and ethical guidelines:

1. Adherence to Industry Standards:

- Following best practices in software development, including thorough testing, code reviews, and documentation. Ensuring the system is reliable, secure, and user-friendly.

2. Ethical Guidelines:

- Adhering to the ethical guidelines set forth by professional bodies such as the Association for Computing Machinery (ACM) and the Institute of Electrical and Electronics

Engineers (IEEE). This includes maintaining honesty, integrity, and transparency throughout the project.

Managing Ethical Considerations

To manage ethical considerations effectively, the following actions will be taken:

1. Ethics Review:

- Conducting an ethics review to identify potential ethical issues and develop strategies to address them. This includes obtaining approval from relevant ethics committees if required.

2. User Consent:

- Implementing a transparent user consent process, ensuring that users are fully informed about the data being collected and how it will be used.

3. Continuous Monitoring:

- Continuously monitoring the system for any ethical concerns that may arise during its operation. Establishing a feedback mechanism for users to report any ethical issues.

4. Data Security Measures:

- Implementing robust data security measures, including encryption, access controls, and regular security audits to protect user data.

By addressing these legal, social, ethical, and professional considerations, the project aims to develop a navigation system that is not only effective and efficient but also responsible and respectful of users' rights and societal norms.

1.5 Background

Advancements in information technology have proven effective in addressing real-world problems, such as monitoring road traffic levels and determining the shortest travel routes. Maps are crucial for presenting and analyzing spatial distribution data across various sectors, including business, resources, and service needs. According to Woodbury, 85% of all computerized databases worldwide contain a location component like street addresses or zip codes. Geographic Information System (GIS) technology has emerged as one of the most significant research tools and rapidly growing high-tech solutions for data monitoring and management [9].

GIS technology integrates diverse datasets swiftly, enabling users to analyze and visualize information efficiently. In service management, GIS is used extensively to display large volumes of data relevant to local and regional planning activities [9]. Location and routing techniques in GIS support decision-making processes by presenting solutions clearly and allowing for process exploration, thus enabling informed judgments. Web-based architectures, such as the World Wide Web (WWW), facilitate deploying databases via the Internet, providing access to central data storage, remote data collection, and interaction between GIS and optimization software [10].

The motivation for this project arises from the need to provide accurate and effective information for road network analysis and traffic management. Road network analysis produces reliable information that aids decision-makers in selecting optimal routes through mathematical calculations and dynamic programming [11], [12]. Determining the optimal route is a common research topic in transportation literature, focusing on minimizing travel time and cost [13].

Effective traffic conditions management and accurate information provision are critical for monitoring road networks and navigation systems. This study aims to implement and evaluate a GIS-based methodology for determining optimal routes in road networks, utilizing key information based on the cost of distance. An integrated approach to location and routing, supported by a robust application, offers a significant competitive advantage. The project's goal is to develop a decision tool for monitoring road networks and facilitating efficient navigation within the University of Roehampton campus.

GIS technology in transport management has proven effective in analyzing road networks and optimizing routes. Chen [9] highlights the flexibility and power of GIS in various fields, including retail tourism and teaching curriculum. Calvoa et al. [10] demonstrate GIS's potential in solving daily carpooling problems. Boulaxis and Papadopoulos [11] highlight the use of dynamic programming techniques with GIS for optimal feeder routing in distribution systems. Monteiro et al. [12] illustrate the application of GIS spatial analysis in optimizing electric line routing, showcasing GIS's versatility in various optimization scenarios. Finally, Ahn and Rakha [13] explore the effects of route choice decisions on vehicle energy consumption and emissions, providing insights into the environmental benefits of optimal routing decisions.

Effective traffic management through advanced traffic forecasting and GIS technologies is essential for reducing economic losses and improving urban mobility. This project aims to leverage real-time data and Dijkstra algorithm to provide a comprehensive solution for traffic management in London and navigation within the University of Roehampton campus, significantly reducing travel time, fuel costs, and overall congestion, contributing to a more efficient and sustainable urban environment.

1.6 Structure of Report

The report is organized into five chapters, each focusing on a specific aspect of the project. Chapter 2: Literature and Technology Review provides a comprehensive review of existing literature and technologies relevant to traffic forecasting and road network optimization. Chapter 3: Implementation with Dijkstra's Algorithm details the methodology and steps taken to implement the project using Dijkstra's algorithm. Chapter 4: Evaluation and Results presents the evaluation of the implemented system and discusses the results obtained. Chapter 5: Conclusion summarizes the findings of the project and discusses its implications.

Chapter 2 Literature – Technology Review

Traffic congestion remains a significant issue, especially in urban areas [14]. The primary cause is road congestion or exceeding road capacity, leading to a supply shortage and increased accidents and traffic jams [15]. Inefficient vehicle routing and the waiting time at intersections, compared to travel time on road segments, also contribute to this problem [14]. These challenges are often referred to as Vehicle Routing Problems (VRP), a heavily researched topic in operations research due to the rising costs of emissions in recent years [16]

2.1 Literature Review

Path Planning

Extensive research has developed effective global path planning algorithms, including graph search, random sampling, and bionic algorithms. Notably, the Dijkstra Algorithm is a key method in graph search [17], recognized for solving optimization problems, including path planning [18]. Path planning algorithms are primarily divided into heuristic and optimal path planning. Optimal path planning techniques calculate all costs from one graph point to the next to determine the shortest route to each vertex [19]. The shortest route problem involves finding the most direct path from a given starting point to a desired destination, typically represented in graph form [20].

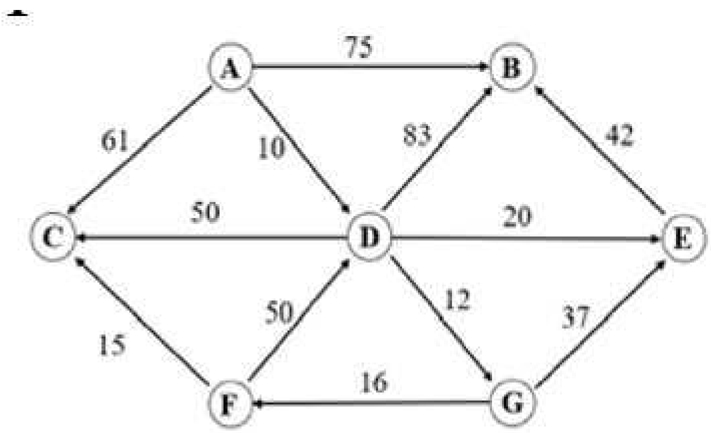

Fig. 1. Shortest path graph example

The optimal path planning algorithm explores all possible solutions to find the globally optimal path. Incremental graph algorithms can be integrated into optimal algorithms to enhance their efficiency. Conversely, heuristic path planning algorithms iteratively refine a subset of solutions, aiming for a feasible but potentially suboptimal outcome. Although these solutions are not guaranteed to be globally optimal, they often achieve comparable quality. Heuristic algorithms require a destination point to begin the process [19].

Routing Algorithms

Numerous vehicle routing algorithms exist, including Bellman-Ford, Floyd Warshall, and the A* search algorithm [11]. However, the Dijkstra Algorithm stands out as the most well-known and fastest labeling technique [21]. Developed by Dutch computer scientist Edsger Wybe Dijkstra in 1959 [22], it is a greedy algorithm that determines the shortest route between an origin node and a destination vertex [23]. According to Suardinata et al., the Dijkstra algorithm finds the minimum length from vertex a to z in a weighted graph with positive weights, as negative nodes cannot be used [24]. The key concepts include:

- The length of a trajectory in a weighted graph is the sum of the weights, denoted by $\mathrm{l(u,v)}$.

- The shortest path from initial node v_0 to end node v_k is the path with the minimum length, expressed as $\mathrm{l}(v_0, v_k) = \mathrm{min}_{\mathrm{i}=1,\ldots}\mathrm{l_i}(v_0, v_k)$ if there are m different paths.

The algorithm is briefly represented as $\mathrm{G = (V, E)}$, where V is a set of vertices and E is a set of edges [25].

Dijkstra initiated research in grid-map-based path planning by examining the shortest path between two nodes on a map [26]. The Dijkstra Algorithm, a graph search method, efficiently determines the shortest paths between two points [27]. This approach is particularly useful for solving the Vehicle Routing Problem. Road networks are modeled using graph theory, where intersections are represented as nodes connected by edges, with cost values assigned from departure to destination. Computer algorithms then efficiently find vehicle routes within this framework [12].

Dijkstra's method employs a priority queue to trace the highest-priority point. It assigns values to each point and maintains these values until the inspection point is reached, allowing comparison with newly determined route values [18].

The advantage of using the Dijkstra Algorithm is that it requires every edge's weight to be non-negative; otherwise, the path cannot be accurately followed, leading to incorrect results. The algorithm starts from the initial point, expanding outward in layers until the target node is found [28].

The Dijkstra algorithm is commonly used in real-time applications, such as geographic mapping, for optimal path planning, including considerations like traffic light durations. Additional research has integrated the time dimension into the algorithm [29].

2.2 Technology Review

Dijkstra's Algorithm and Its Relevance

Dijkstra's algorithm, introduced by Edsger W. Dijkstra in 1959, is a seminal method for finding the shortest paths in a graph [16, 17]. Extensively utilized in transportation networks, computer science, and urban planning, it calculates the shortest path between nodes in a graph with non-negative weights, making it ideal for road networks. Studies highlight its effectiveness in optimizing routes. For example, Botsis and Panagiotopoulos applied it at the International Hellenic University in Serres, Greece, finding the shortest routes between campus locations [14]. Their findings confirm the algorithm's efficiency in solving complex routing problems, motivating its use in traffic forecasting and navigation.

Choosing Dijkstra's Algorithm over Bellman-Ford

The Bellman-Ford algorithm, developed by Richard Bellman and Lester Ford Jr., calculates shortest paths in a graph and can handle graphs with negative edge weights. However, its higher time complexity ($O(|E||V|)$) makes it less efficient than Dijkstra's algorithm, which has a time complexity of $O(|E| \log |V|)$. Given the extensive and complex road network of London and the University of Roehampton, Dijkstra's algorithm's efficiency makes it a more practical choice.

While Bellman-Ford's ability to handle negative weights is advantageous in certain scenarios, this feature is unnecessary for navigating London's road network, where negative edge weights do not exist. Therefore, the additional complexity of Bellman-Ford is unwarranted for this application.

Practical Applications of Dijkstra's Algorithm

Dijkstra's algorithm has been successfully employed in various practical scenarios, such as optimizing urban rail networks and facilitating building evacuations, demonstrating its versatility and reliability [14].These applications underscore the algorithm's suitability for traffic forecasting in road networks, including the complex and dynamic environment of London's road infrastructure.

Limitations of Existing Navigation Tools

Current navigation tools, such as Google Maps, provide broad route planning services but often fall short in specific contexts, such as University of Roehampton campuses or intricate urban road networks[14, 17]. These tools may not always provide the most optimal routes or account for all the nuances of local road layouts .

This research aims to address these limitations by implementing a tailored approach using Dijkstra's algorithm, specifically designed for the navigation needs of London's road network and the University of Roehampton. By integrating on-site drone surveys for precise data collection and employing advanced software tools, the proposed system aims to deliver an efficient and user-friendly navigation solution.

Advancing Current Research

The study builds on previous research, such as the work by Botsis and Panagiotopoulos, which applied Dijkstra's algorithm to campus navigation [14]. By densifying the graph model and developing a web application, this research seeks to enhance user experience and improve the routing algorithm's effectiveness. This focused approach aims to fill existing gaps and extend the applicability of Dijkstra's algorithm to more complex and dynamic traffic forecasting scenarios within road networks.

2.3 Summary

Findings:

- **Dijkstra's Algorithm:** Proven effective for finding shortest paths in graphs. Ideal for navigation in complex environments like university campuses and road networks [16, 17].

- **Real-World Applications:** Successfully used in optimizing campus routes (Botsis & Panagiotopoulos, 2020) and urban networks [14]. Highlights its suitability for detailed, constrained environments.

- **Limitations of Existing Tools:** Generic tools like Google Maps often fail in providing optimal routes for specific layouts and intricate networks [14, 17].

Goal: To reduce traffic congestion and enhance campus navigation through an optimized, tailored navigation system.

Chapter 3 Implementation

To enhance road routing and navigation in London, especially within the University of Roehampton, a traffic forecasting system is proposed to avoid inefficiencies and traffic jams. The system's architecture, depicted in Fig.1, comprises three main elements: a server, a database, and the users, refers to Appendix C for the devlopment codes/scripts and the dataset.

3.1 System's Architecture

A. Geographic Information System(GIS)

Geographic Information Systems (GIS) have been advancing since the 1970s. GIS serves as a crucial tool for location mapping, visualizing dynamic conditions, and aiding decision-making processes [32-34]. Geospatial data are instrumental in monitoring responses to accidents. During the response phase, GIS enables the analysis of real-time data, facilitating visualization and automation for more efficient decision-making. Research in GIS has concentrated on areas like shortest path analysis [35, 36]. This highlights the significant potential of GIS applications to reduce response times if geospatial information is utilized at the initial stages of accident response.

B. Web Service

A web service is an internet-based technology defined by the W3C as "a software system designed to support interoperable machine-to-machine interaction over a network" [37].

C. Web GIS

Web GIS initially offered only client access through browsers, incorporating basic GIS functions to minimize system costs. Browsers are an optimal choice because they can extensively utilize GIS, reduce software expenses, simplify operations, and provide a user-friendly interface [38, 39].

D. System Component

The web framework is based on a three-tier architecture consisting of the client layer, middleware layer, and database layer. These components provide a unified interface for data consultation, requests, and decision-making. The PostgreSQL database, enhanced with PostGIS, supports both relational and spatial queries. Users interact with the database via the Internet, using a web browser for navigation and spatial analysis. User requests are processed by the Apache Web Server and passed to the GIS server, which queries the database[40]. The system supports system administrators, who manage databases, and teleoperators, who handle data entry, editing, and deletion through a GUI. The system utilizes geographic data for mapping and semantic data for routing and traffic flow information.

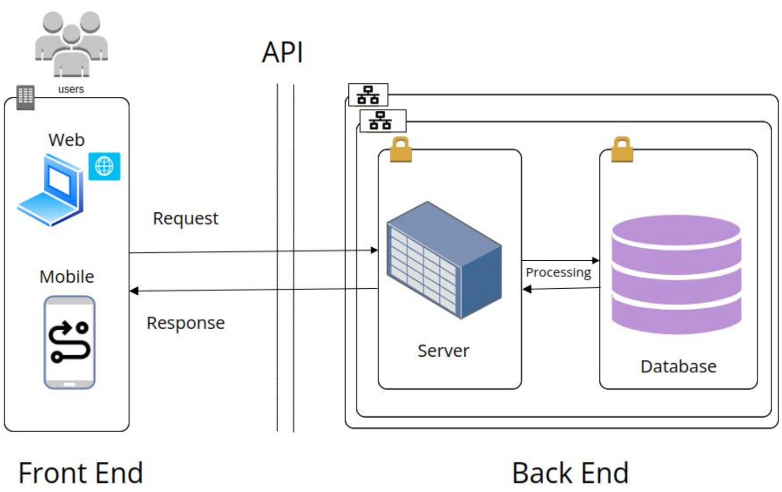

Figure 1. System Component

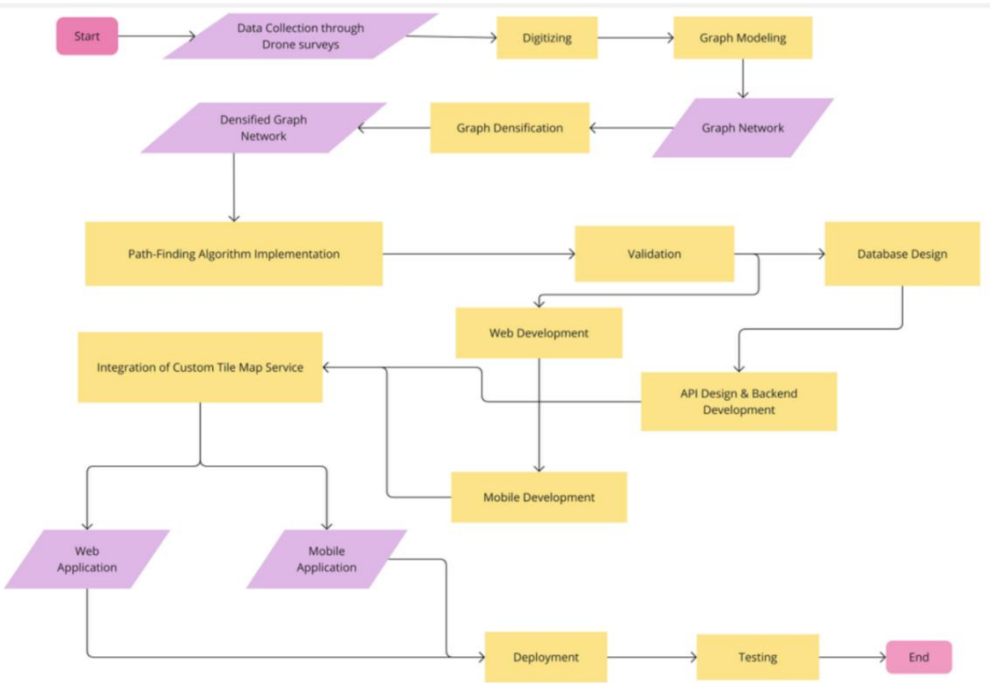

Figure 2. System Development Flow Chart

3.2 Data Structure

A. Study Area

The study focused on London roads and the University of Roehampton for designing and implementing a GIS-based road network navigation system. The dataset was sourced from TFL, see Appendix C1 for more details. University of Roehampton locations were digitized using QGIS.

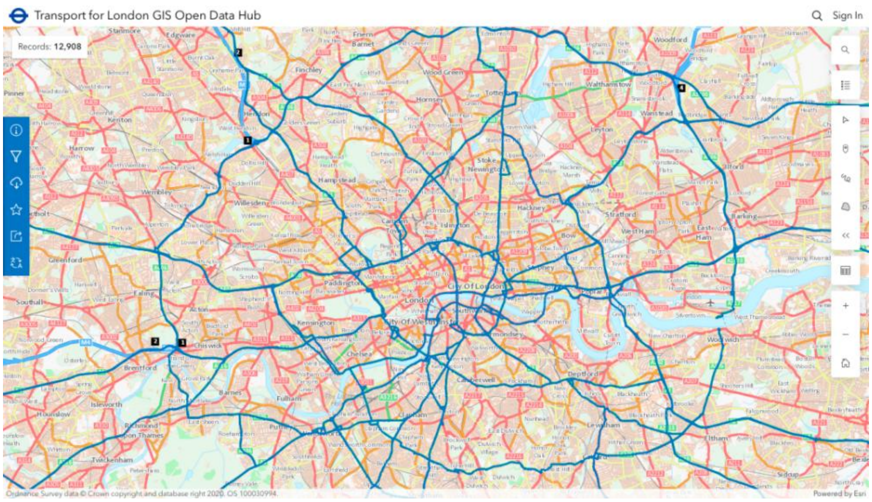

Figure 3. London Road Network(TFL)

B. GIS Layers

The Global Positioning System (GPS) is used to collect longitude (X) and latitude (Y) values, identifying the geographic locations of monitoring sites. The application is based on real data embedded in maps as layers. The spatial data include roads, street names, and monitoring site locations. There are several distinct layers: the road layer, city layers, and the road connection network layer. Roads are represented by polylines, while street names and monitoring stations are represented by points. City names, also represented by points, are placed on the road line to enable plugin routing operations. The "cities" table currently contains a few test names, with more to be scanned and added in future application updates.

Figure 4. London Road and Cities(QGIS)

Figure 5. University of Roehampton routes(Google Earth)

C. Graph Modeling

The graph network, digitized and cleaned using QGIS, is modeled with Python to provide a comprehensive representation of the London road network. This model includes both visual and data aspects, capturing the intricacies of the network through nodes (representing intersections) and edges (representing road segments). The resulting graph is ready for further processing and analysis, enabling advanced functionalities such as route optimization, traffic forecasting, and network efficiency studies. This detailed graph model serves as a foundational tool for enhancing navigation and traffic management systems in London, offering precise and actionable insights into the road network.

D. Graph Densification

The graph will be densified by adding intermediate nodes and refining edges to enhance path-finding accuracy and efficiency. This process results in a more detailed and precise representation of the road network, ensuring better navigation and traffic management. By incorporating additional nodes and adjusting edges, the graph model can capture the finer details of the network, leading to improved route calculations and more reliable traffic predictions. The densified graph thus offers a robust foundation for various applications, including advanced GIS analyses, real-time traffic updates, and optimized route planning, significantly benefiting road users and urban planners.

3.3 System Development

A. Database Design

A comprehensive database design using PostgreSQL with PostGIS will be developed to efficiently store and manage spatial and attribute data. This approach ensures robust and scalable data management, accommodating the complexities of spatial data such as road networks, intersections, and monitoring sites. By leveraging PostgreSQL's powerful database capabilities and PostGIS's spatial data handling features, the system will support advanced queries, spatial analyses, and real-time updates. This design will facilitate seamless integration with other GIS tools, enhance data integrity, and provide a solid foundation for ongoing system enhancements and data-driven decision-making processes.

B. Backend Development

The backend will be developed to manage data processing, implement pathfinding algorithms, and handle database interactions. This will involve setting up servers and API endpoints to facilitate secure and efficient data transactions. The backend infrastructure will be designed to support real-time data updates, query execution, and integration with the frontend, ensuring a seamless user experience. By focusing on robust server architecture and optimized data handling, the backend will enable efficient processing of spatial data, accurate route calculations, and reliable communication with the database, ultimately enhancing the overall performance and scalability of the navigation system.

C. Web Development

A web interface using React will be developed to display optimal routes on a digital map of London. This interface will be designed for accessibility via computers, ensuring ease of use for all campus members. The goal is to provide an interactive and responsive experience, allowing users to view and interact with real-time traffic data and optimal routes efficiently. The web application will feature intuitive navigation, real-time updates, and user-friendly design elements, making it a valuable tool for navigating the campus and surrounding areas. By leveraging React's capabilities, the interface will offer a seamless and engaging user experience.

D. Integration of Custom Tile Map Service

The application will integrate a custom tile map service to provide high-resolution drone imagery, ensuring seamless and efficient map rendering. This integration will enhance user interaction by offering detailed and accurate visual representations of the area. The high-resolution imagery will facilitate better route visualization and navigation, improving the overall user experience. By utilizing a custom tile map service, the application will maintain fast loading times and smooth performance, allowing users to interact with the map effortlessly and access up-to-date visual information with ease. This feature will significantly enhance the application's functionality and appeal.

3.4 Implementation

The chosen tools are open-source solutions like MapServer, Leaflet, and PostgreSQL, which provide functionalities comparable to commercial options but at a reduced setup cost. This application is designed with the assumption that users (decision-makers) lack a background in modeling or implementation. Therefore, the information presented to them is straightforward and user-friendly, avoiding technical or conceptual details.

To develop the application, an Object-Oriented (OO) methodology, Python Dijkstra's algorithm will be implemented in Python, Go, and JavaScript to calculate the shortest paths on the graph SQL for the database. The application features a main window where all functionalities are accessible via a toolbar or menu, making it easy for users to navigate, see Appendix C2 for more details.

The target users are decision-makers with moderate computer skills, emphasizing the need for an intuitive interface. The tool is divided into two subsystems:

- A subsystem for entering location name, editing, or delete road network data

- A subsystem for managing geographic data display

The tool includes a set of graphical user interfaces and is designed for web platforms. Its open architecture facilitates the easy addition of new functionalities.

A. Graphical User Interface (GUI)

The primary goal of the GUI was to facilitate easy and efficient access. According to [41], the following characteristics are essential:

- Easy to learn: Ensures that any user can intuitively use the tool.

- Robustness: Allows users to recover from unintended situations.

- Interactivity: Enables effective information flow between the user and the system.

- Event-based: Keeps users constantly aware of the tasks they are performing.

B. **Tools**

The semantic database is managed by the PostgreSQL database management system. Graphic layers obtained with GIS from the digitization of University of Roehampton routes and the London road network are imported into the Oracle spatial DBMS: PostGIS. Utilizing this Oracle spatial DBMS allows interaction with the mapping module of the web server: Leaflet.

C. Decision Model

The web serves as the primary interface for connecting users to the system. The teleoperator uses a Graphical User Interface (GUI) to simplify the interpretation of information, enabling them to (Figure 6):

- Control Layers: Toggle the visibility of certain layers.

- Map Operations: Perform actions like zooming in, zooming out, expanding, planning, and querying specific areas of the map to gain a better understanding of the data in those regions.

- Focus on London: Directly center on the London road networks.

- Scale Visualization: View the map at different scales by changing the representation scale.

- Map Reference: Display a reference map for easier location identification.

- Full Caption Layers: Show comprehensive caption layers.

- Database Query: Utilize drop-down lists and checkboxes to query the database, or directly use SQL (Structured Query Language) statements for a combination of relational and spatial queries.

- Spatial Queries: Perform spatial queries in various ways, such as clicking the search box to query map objects or by entering the identifier, designation, or X and Y coordinates of a post, with results transmitted to the user as an XML document via the web.

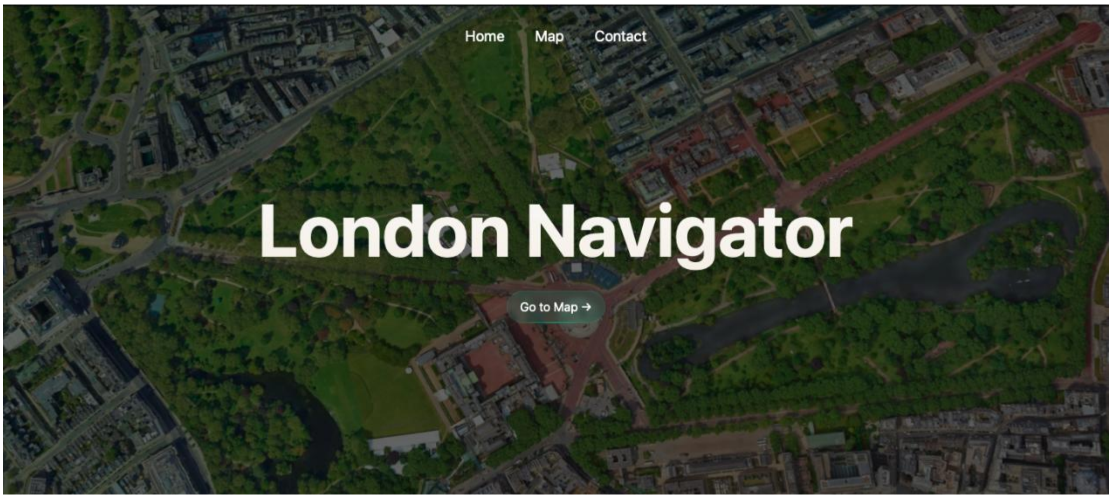

Figure 6. Welcome window((GUI)https://london-navigator.netlify.app

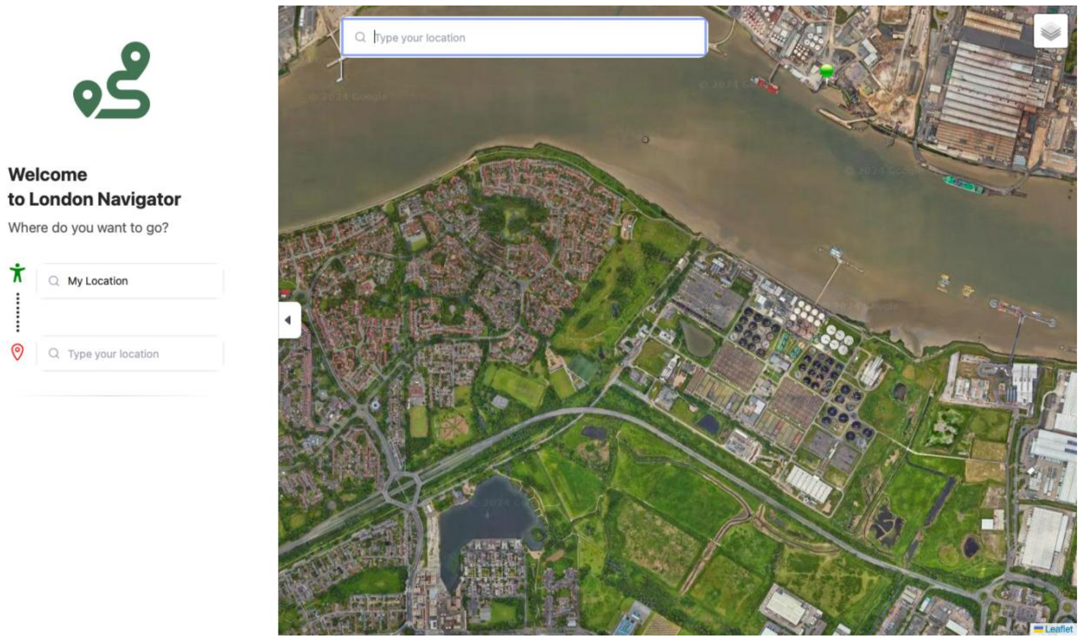

Figure 7. Operation window(https://london-navigator.netlify.app/map)

3.5 Implementation of Dikstral's Algorithm

A. Dijkstra's algorithm

Incorporating an efficient shortest-path algorithm within the GIS will reduce time spent on the road network and address navigation inefficiencies. The optimal routes are those with the minimum required transportation times. A real-time system must be capable of providing prompt responses to such queries.

In the London Navigator, the experimental system developed for this project, the routing service is implemented using Dijkstra's algorithm. This spatial optimization algorithm is widely used in GIS software for finding the shortest routes. Its performance relies on the data structures used to implement the graph representing the spatial network [42]. The road network is represented by a graph (non-oriented in this case), where each intersection on the physical road is depicted as a node.

Let G = (N, A) be a graph consisting of a set of nodes (N) and a set of arcs (A), each with a non-negative cost C. This graph is designed for tracing the least-cost path (route) in G. For a given destination node in the network, Dijkstra's algorithm calculates the least accumulated cost between the destination node and every other node, then finds the least-cost path from any origin node to the destination node. The logical procedure of Dijkstra's algorithm is as follows [43]:

1. Let the node at which we are starting be called the source node. Assign to the source node an initial value of zero and to all other nodes an initial value of infinity. Mark all nodes as unvisited. Set the source node as current.

2. For the current node, consider its unvisited neighbors directly connected by links having cost values and calculate their accumulated costs from the source node. If the new accumulated cost is less than the previously recorded cost, overwrite the cost.

3. When all neighbors directly connected to the current node are completely considered, mark the current node as visited. A visited node will not be checked again, ensuring its accumulated distance is final and minimal.

4. If all nodes have been visited, finish. Otherwise, set the unvisited node with the least accumulated cost to the source node as the next "current node" and continue from step 2.

Dijkstra's algorithm is very similar to the A* algorithm. The cost function c used to evaluate shortest paths in the Dijkstra algorithm is augmented by an estimator function, which estimates the shortest path between two given graph nodes. This is represented as $c(s, d) = g(s, v) + h(v, d)$, where $g(s, v)$ is the cost from source s to v, and $h(v, d)$ is the heuristic estimated cost from v to the destination d. The estimator function is a heuristic that can be chosen arbitrarily. If the estimator function is 0, A* turns into Dijkstra's algorithm [43].

B. **Simulated test cases and routing**

The algorithm was implemented in Python within a web environment hosted on [Render](https://render.com), which manages the backend, including the database and server operations. Render not only facilitates the handling of road maps online through user-friendly interfaces, but also supports the implementation of the "pgrouting" plugin, which runs on both the client and server sides. The frontend, responsible for the user interface, was hosted on Netlify and developed using Go language. The interface is designed with an intuitive template for easy and effortless use, as illustrated in the provided image[figure 6].

Users can view specific locations on the map by entering the name of the location. Additionally, they can determine the shortest route to a destination by specifying the starting and ending points. The algorithm then computes and displays the shortest route on the map. Whenever a routing operation is performed, the location names are passed as parameters from the client interface to the server. The server connects to the database, processes the request, computes the shortest path between the specified points, and highlights it on the map with a distinct color, see figure 7.

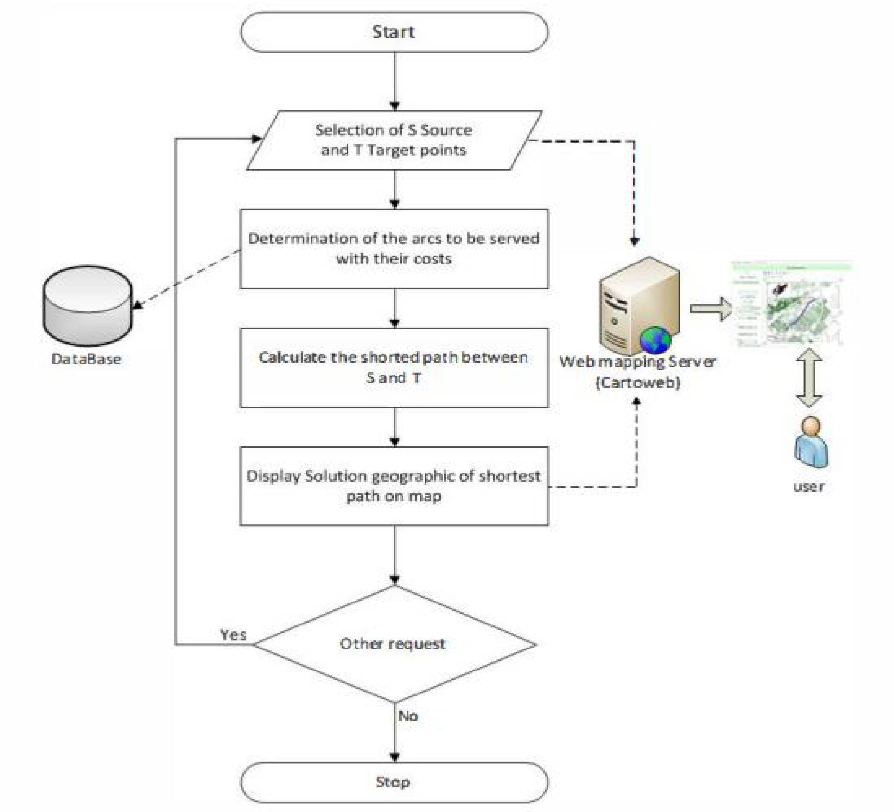

Figure 8. The proposed routing solution with Dijkstra's algorithm.

I. Test Case 1: Routing Problem at the University of Roehampton

In this scenario, we have a user named Aman Richard, a 29-year-old MSc Data Science student at the University of Roehampton, London. Aman resides at Southlands Chapel, located within the university campus. One day, he decided to go for a workout at the university gym. His fellow student recommended the Oliver Garnet-ROEactive Gym, which is situated on campus. However, Aman was unfamiliar with its exact location.

To solve this problem, Aman decided to use the "London Navigator," a tool recommended by his peer for optimizing the shortest routes on London roads, including those within the University of Roehampton campus. Using the system, Aman entered his starting point (Southlands Chapel) and his desired destination (Oliver Garnet-ROEactive Gym). The system quickly computed and provided the shortest route, complete with clear path-guiding hints displayed on the map.

Thanks to the London Navigator, Aman was able to navigate effortlessly to the gym, something that Google Maps could not achieve due to its limitations in recognizing specific campus locations. Consequently, Aman's problem was resolved efficiently with accurate routing provided by the system, see figure 8.

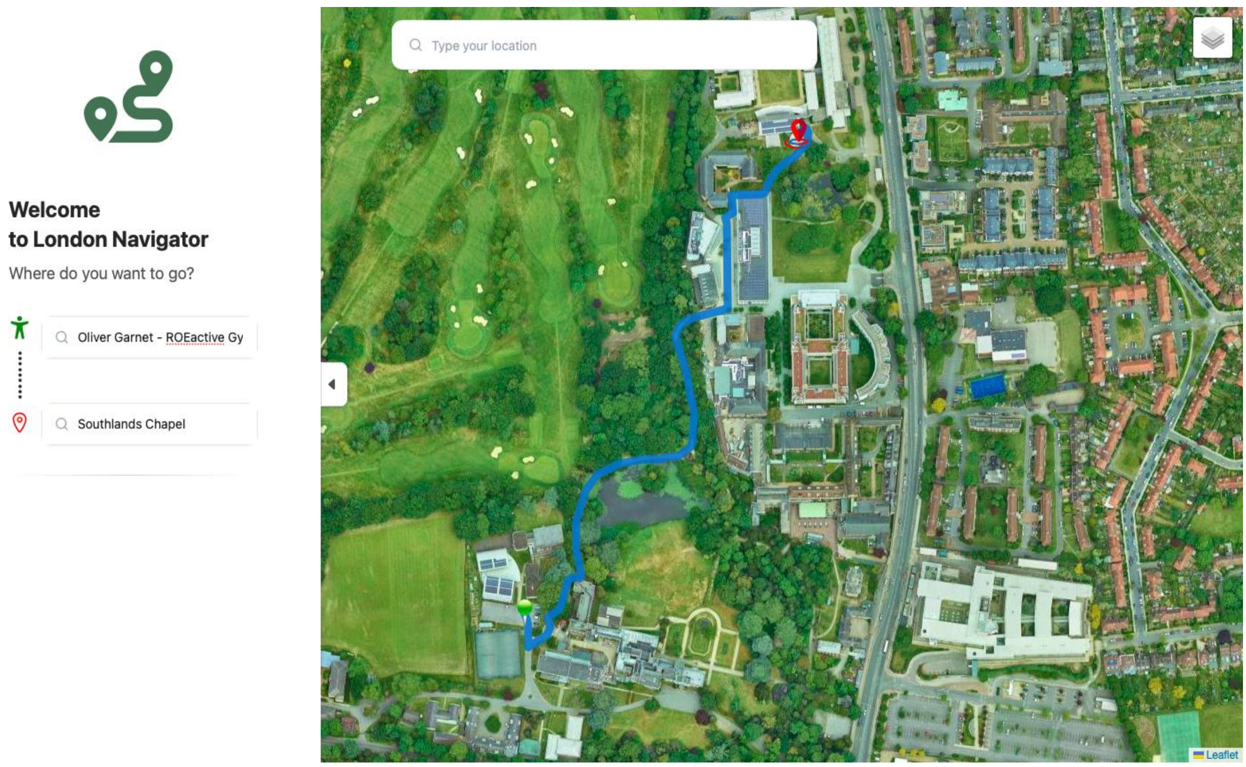

Figure 9. Southlands Chapel to Oliver Garnet-ROEactive Gym

II. In **Test Case 2: Locating a Lecture Class Building at the University of Roehampton**

this scenario, Dr. John Carter, a newly employed lecturer at the University of Roehampton, faces a challenge. On his first day, Dr. Carter arrives at the university's main entrance, ready to deliver a lecture. However, he struggles to locate the Sir David Bell Building, where his lecture is scheduled to take place. Unfortunately, Google Maps does not provide the precise location of this building within the campus.

To resolve this issue, Dr. Carter decides to use the "London Navigator," a tool designed for optimizing routes both within London and specifically on the University of Roehampton campus. At the main entrance, Dr. Carter inputs his current location and sets the Sir David Bell Building as his desired destination.

The London Navigator quickly calculates the shortest route, providing Dr. Carter with a clear and detailed path to follow. Thanks to the accurate guidance from the system, Dr. Carter is able to navigate through the campus and arrive at the Sir David Bell Building promptly, ensuring that he is on time for his lecture.

This case highlights the effectiveness of the London Navigator in addressing routing challenges that conventional map services, like Google Maps, may not fully support within specialized environments such as a university campus, see figure 9.

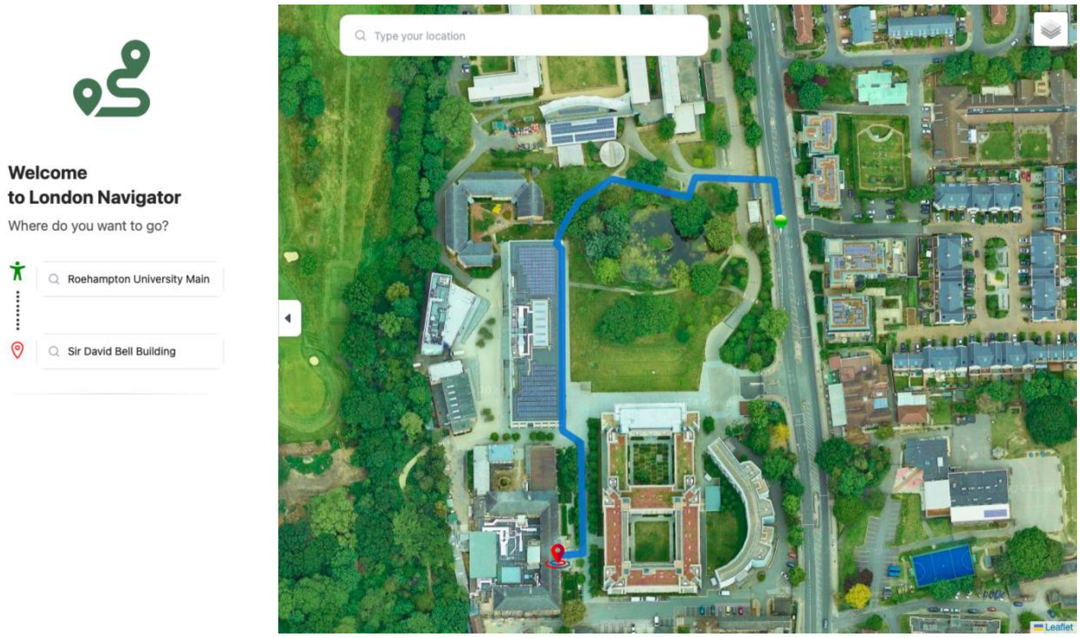

Figure 10. University of Roehampton Main Entrance to Sir David Bell Building

III. **Test Case 3: Delivering Pizza to the University of Roehampton**

In this scenario, Tom Bennett, a dispatch delivery man, is tasked with delivering a pizza to a student waiting at the main entrance gate of the University of Roehampton. Tom is currently in Putney, a nearby area, and needs to find the quickest and most efficient route to the university's main entrance.

To ensure timely delivery, Tom decides to use the "London Navigator," a specialized tool designed to optimize routes within London, including routes leading to the University of Roehampton. Tom inputs his current location in Putney as the starting point and the University of Roehampton main entrance gate as his destination.

The London Navigator promptly calculates the shortest route, offering precise turn-by-turn directions. This guidance allows Tom to avoid potential traffic delays and navigate directly to the university with ease. As a result, he is able to deliver the pizza to the student on time, ensuring customer satisfaction.

This scenario demonstrates the utility of the London Navigator in aiding delivery personnel by providing them with the most efficient routes, particularly when conventional navigation tools might not offer the same level of detail or accuracy for specific locations, see figure 10. see [Appendix C 2]

Figure 11. Putney to University of Roehampton Main Entrance Gate

Chapter 4 Evaluation and Results

To validate the effectiveness of the designed traffic forecasting system, a demonstration was conducted using a web-based client interface. This interface serves as a proof of concept and can be adapted for end-user applications, such as mobile apps, web browser extensions, or integrations with GPS navigation platforms. The client interface grants access to the user's geolocation and allows for the creation of a route by specifying either the current location or a different starting point, along with the desired destination.

The system dynamically generates a route based on the real-time traffic situation at the moment of the request, integrating google map API to forecast potential congestion by impementing dijkstra algorithm. The generated route aims to minimize travel time by choosing the optimal path under current and predicted traffic conditions. Initially, the system was trained using a manually set historical dataset to simulate traffic conditions and refine the the routing by digitising some missing network link within the system.

One of the system's key features is its ability to adapt to changing traffic conditions during a trip. If the traffic situation alters while the driver is on the road, the system can recalculate the route in real time to maintain the shortest possible travel time. The accuracy of the forecasts is continually evaluated by comparing predicted traffic conditions with the actual outcomes. This feedback mechanism allows the system to learn and improve its predictive accuracy over time, ultimately enhancing route optimization to achieve reduced travel times and alleviate congestion.

To quantify traffic congestion, the system utilized the condition $14 \leq n \leq 20$ for $d = 60$, where n is the number of vehicles and d represents the length of the road segment under study. This criterion assumes that a traffic jam forms when there are between 14 and 20 vehicles within a 60-meter stretch of road.

Additionally, the system leveraged the Google Maps API to calculate the number of markers (vehicles) in a specific neighborhood of the London road network. This data on inflow and outflow across the network aids in forecasting routes that result in reduced travel times. The application of this system within the London road network, including the University of Roehampton area, demonstrated its potential to optimize routes, thereby contributing to a more efficient transportation system.

To numerically assess the benefits of the designed traffic forecasting system, the system was tested on approximately 10 different routes under varying traffic conditions, including scenarios with different levels of congestion. The evaluation aimed to quantify the time savings achieved by using the forecasting system compared to not using it.

The following metrics were used:

- t1: Time required to complete the route without using the traffic forecasting system.

- t2: Time required to complete the route using the traffic forecasting system.

- Δt: The time difference between completing the route without and with the forecasting system, expressed as a percentage.

$\Delta t = \frac{t1-t2}{t1} \times 100$ ....(1)

This formula calculates the percentage of time saved by utilizing the forecasting system. The results from the test routes provided a range of time gains, highlighting the effectiveness of the system in optimizing travel times under different traffic conditions.

Using table 1 which contains the test result of 10 different route by using the forecasting system and without forecasting system. To calculate change of time of travel (Δt) at the minimum fixed gain in time travel within the network by using equation(1),

t1 = 25, t2 = 15

$$\Delta t_{min} = \frac{t1-t2}{t1} \times 100 = \frac{25-15}{25} \times 100$$

$\Delta t_{min} = 40\%$ ....(2)

To calculate change of time of travel (Δt) at the maximum fixed gain in time travel within the network by using equation(1),

t1 = 60, t2 = 45

$$\Delta t_{max} = \frac{t1-t2}{t1} \times 100 = \frac{60-45}{60} \times 100$$

$\Delta t_{max} = 25\%$ ....(3)

The above calculations demonstrate the efficiency of using the developed system of road traffic forecasting, namely, the following gains in time from 25% to 40% (average values).

Nodes	Routes	Total Time with Routing System	Total Time without Routing System
N1-N2	Acton- Algate	20	25
N2-N1	Algate- Acton	18	27
N2-N3	Algate - Anerly	45	58
N3-N2	Anerly - Algate	42	60
N3-N4	Anerly - Balham	25	40
N4-N3	Balham - Anerly	23	38
N4-N5	Balham - Battersea	15	30
N5-N4	Battersea - Balham	17	38
N5-N6	Battersea - Bayswater	20	30
N6-N5	Bayswater - Battersea	18	35

Table 1

In the histogram shown in Fig. 12 presents the time gain for overcoming a route laid without a forecasting system and with a forecasting system (10 routes were considered), while Fig.13 represent the total time gain by using the forecasting system in the 10 test experiment. The obtained results show the expediency of using the designed system for vehicles in urban conditions.

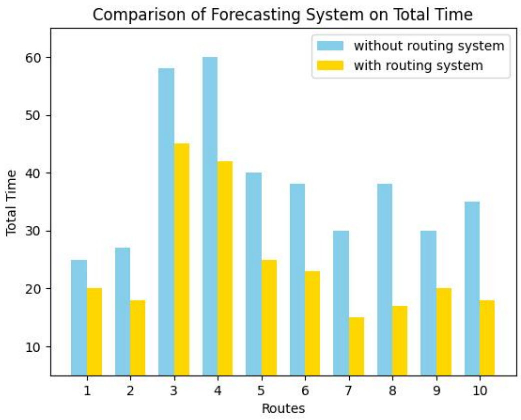

Fig. 12

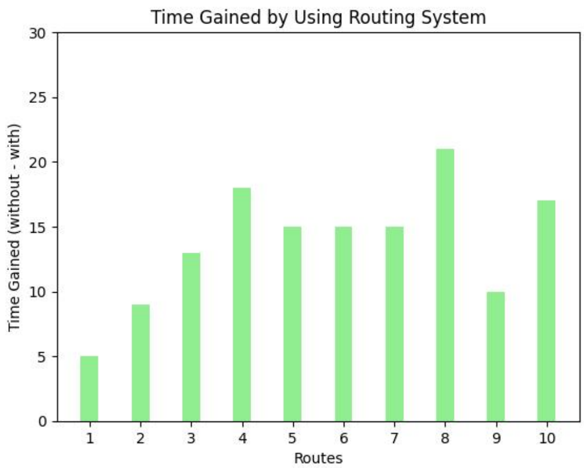

Fig. 13

The selected test cases included routes with both minimum and maximum time gains. These cases illustrated the variability in the system's performance, depending on the severity of traffic congestion. In scenarios with heavy congestion, the time savings were more pronounced, demonstrating the system's capacity to significantly reduce travel times by predicting and avoiding traffic jams.

In summary, the evaluation confirmed that the traffic forecasting system consistently provided time savings, with the magnitude of these savings dependent on the traffic conditions encountered on each route. This demonstrates the practical value of the system in real-world applications, offering potential improvements in commute efficiency and reductions in overall travel time.

4.1 Related Works

In the domain of traffic forecasting and route optimization, numerous approaches have been proposed to enhance the accuracy and efficiency of transportation systems. A noteworthy contribution is the work by Souad El Houssaini and Abdelmajid Sadri, titled "A Web-Based Spatial Decision Support System for Effective Monitoring and Routing Problem." Their research presents an integrated framework that leverages Geographic Information Systems (GIS) combined with a Relational Database Management System (RDBMS) to improve the monitoring and management of road networks[44] .

El Houssaini and Sadri's system utilizes GIS technology to collect and analyze geographic data related to road networks and monitoring stations. Their framework not only identifies optimal routes based on minimum cost using Dijkstra's algorithm but also supports the strategic placement and orientation of monitoring stations in response to road incidents. The integration of diverse datasets and advanced information retrieval within a GIS environment demonstrates the system's potential to manage complex spatial data and optimize routing decisions effectively.

The effectiveness of their system was tested on the road network of Mohammedia City in Morocco, where it demonstrated significant improvements in response to road accidents through simulated test cases. The proposed system, described as scalable and well-structured, serves as a comprehensive tool for traffic monitoring and routing, particularly in scenarios requiring rapid and efficient intervention.

When compared to the current project on traffic forecasting and route optimization in London and the University of Roehampton, El Houssaini and Sadri's work offers valuable insights into the use of GIS and algorithmic methods for managing and optimizing road networks. While their research focuses on the integration of GIS with RDBMS and the application of Dijkstra's algorithm for routing, the current project builds on these methodologies by incorporating real-time traffic data and Google Maps API. This extension aims to dynamically predict traffic patterns and optimize routes, addressing the specific challenges of London's intricate road network.

Another significant contribution to the field is a study presented at the 1st International Conference of Information Technology to Enhance E-Learning and Other Applications (IT-ELA 2020), hosted by Baghdad College of Economic Sciences University. This study explored the application of parallel computation methods to solve the shortest path problem in vehicular networks [45].

In their paper titled "A Parallel Implementation Method for Solving the Shortest Path Problem for Vehicular Networks," Emad H. Al-Hemiary and Salim A. Mohammed Ali from Al-Nahrain University in Baghdad, Iraq, investigated the complexities of finding the shortest paths in mobile and densely populated vehicular networks. The authors applied Hedetniemi's Algorithm, a well-known method for calculating multi-source multi-destination shortest paths, and enhanced its computational efficiency using OpenCL (Open Compute Language).

Their research demonstrated a substantial acceleration in path calculation, achieving a 40x speedup when utilizing parallel computation compared to a sequential approach. This acceleration was particularly evident when handling high-dimensional input data, making the system more responsive and scalable for real-world applications. This study highlights the importance of leveraging parallel computing techniques to address the computational challenges inherent in vehicular networks, a goal that aligns closely with the objectives of the traffic forecasting system developed in the current research.

These related works underscore the ongoing advancements in traffic management and the potential of integrating advanced computation methods and real-time data to further optimize route planning systems. Such innovations contribute significantly to the overall efficiency and responsiveness of modern transportation networks.

Chapter 5 Conclusion

The "London Navigator" system, designed for traffic forecasting and route optimization within the London road network, including the University of Roehampton, has proven to be an effective tool for addressing specific routing challenges that conventional map services may not fully support. Through a series of test cases, the system demonstrated its capability to provide precise and efficient route guidance in scenarios that involved navigating within complex environments like university campuses, see [Appendix C 1] for more details.

Test Case 1 highlighted the system's effectiveness in helping Aman Richard, an MSc Data Science student, navigate from Southlands Chapel to the Oliver Garnet-ROEactive Gym within the University of Roehampton. The London Navigator successfully provided a short and clear route, overcoming the limitations of conventional tools like Google Maps, which struggled to recognize specific campus locations. This case illustrated the system's practical value in enhancing campus navigation.

Test Case 2 involved Dr. John Carter, a new lecturer at the University of Roehampton, who needed to locate the Sir David Bell Building for his lecture. The London Navigator again proved its effectiveness by quickly guiding him through the campus, ensuring he arrived at his destination on time. This case emphasized the system's utility for university staff and visitors, ensuring timely and accurate navigation across the campus.

Test Case 3 demonstrated the system's broader applicability by assisting Tom Bennett, a delivery man, in finding the quickest route from Putney to the University of Roehampton's main entrance. The London Navigator enabled Tom to avoid traffic delays by choosing route that is not congested and deliver his pizza on time, underscoring the system's relevance for logistics and delivery services operating within the London area.

The system's effectiveness was further validated through a rigorous evaluation process, where it consistently provided time savings, particularly in scenarios with heavy traffic congestion. The test results confirmed that the London Navigator could significantly reduce travel times by dynamically predicting traffic patterns and optimizing routes. The system's integration with real-time traffic data and its use of advanced algorithms, such as Dijkstra's algorithm, allowed it to outperform conventional navigation tools, particularly in specialized environments.

In conclusion, the London Navigator represents a significant advancement in traffic forecasting and route optimization within urban settings. Its ability to adapt to real-time traffic conditions and its successful application in diverse test cases within the University of Roehampton highlight its potential as a valuable tool for improving navigation efficiency in complex environments. The system not only offers practical benefits for everyday users but also contributes to reducing congestion and improving overall transportation efficiency within the London road network.

5.1 Future Work

Building on the successful implementation and evaluation of the London Navigator system, several enhancements are planned to expand its functionality and further improve its effectiveness in real-world applications.

One of the key areas for future development is the integration of GPS capabilities into the system. By incorporating GPS technology, the London Navigator can be transformed into a robust device tracker, allowing users to monitor the real-time location of their devices within the network. This feature would be particularly useful for applications such as fleet management, where it is crucial to track the location and movement of vehicles to optimize routes and improve overall efficiency.

Another significant enhancement involves the addition of a feature to identify congested regions within the network. The system will be upgraded to provide real-time indications of traffic congestion, enabling users to visualize problem areas on the map. Alongside this, the system will suggest alternative routes to help users navigate around congested zones effectively. This feature aims to reduce travel times further and alleviate traffic congestion by guiding users away from high-traffic areas and onto less congested paths.

These planned enhancements will significantly extend the utility of the London Navigator, making it not only a powerful tool for route optimization but also an essential asset for real-time traffic management and device tracking. The integration of GPS and the ability to indicate and avoid congested regions will ensure that the system remains at the forefront of urban traffic management solutions, offering users a comprehensive and adaptable tool for navigating complex road networks.

5.2 Reflection

Working on this project has been a deeply rewarding experience, offering both challenges and opportunities for growth. The process of developing the London Navigator system not only allowed me to explore an area of personal interest but also significantly expanded my technical skills and knowledge in the field of traffic forecasting and route optimization.

One of the most valuable aspects of this project was the hands-on experience I gained with Geographic Information Systems (GIS) tools, particularly QGIS. Through this, I learned how to digitize road networks, a critical skill that enabled me to construct accurate representations of London's complex road system. This experience was pivotal in laying the foundation for the project, as it allowed for the precise mapping of routes and the integration of real-time traffic data.

I also acquired practical knowledge in graph construction using data from Transport for London (TfL). This involved creating a network model that could be efficiently navigated by algorithms, further deepening my understanding of the underlying structures that drive traffic management systems. Implementing Dijkstra's algorithm on the London road network was another significant milestone in this project. This task challenged me to apply theoretical concepts in a practical context, enhancing my problem-solving skills and reinforcing my understanding of algorithmic efficiency.

The project goals were met largely because of my passion for this area of study. My previous work on a smaller-scale project in the same domain provided a strong foundation, enabling me to expand the scope to a larger network system with confidence. This continuity allowed me to build on my earlier successes and apply lessons learned to this more ambitious undertaking.

In hindsight, while I am pleased with the outcomes, there are a few aspects I would approach differently if given the chance. Although the project was successful, I realized that more extensive initial planning around data acquisition and integration could have streamlined some of the later stages of the project. Additionally, incorporating user feedback earlier in the development process might have helped refine the user interface and overall functionality of the system.

Overall, this project has been a significant learning journey, equipping me with new skills and deepening my understanding of traffic forecasting and route optimization. The experience has reinforced my passion for this field and has provided a strong foundation for future work and research in this area.

References

[1] INRIX, Inc., "INRIX 2022 Global Traffic Scorecard: UK," Jan. 10, 2023. [Online]. Available: https://inrix.com/press-releases/2022-global-traffic-scorecard-uk/. [Accessed: Jul. 31, 2024].

[2] L. Zhi, X. Zhou, and J. Zhao, “Vehicle Routing for Dynamic Road Network Based on Travel Time Reliability,” IEEE Access, vol. 8, pp. 190596–190604, 2020, doi: 10.1109/ACCESS.2020.3030654.

[3] P. Udhan, A. Ganeshkar, P. Murugesan, A. R. Permani, S. Sanjeeva, and P. Deshpande, “Vehicle Route Planning using Dynamically Weighted Dijkstra’s Algorithm with Traffic Prediction,” May 2022.

[4] R.J.C. Chen, "Geographic infonnation systems (GIS) applications in retail tourism and teaching curriculum"',Journal of Retailing and Consumer Services,vol. 14,pp. 289-295,2007.

[5] N.G. Boulaxis, M.P. Papadopoulos, "Optimal feeder routing in distribution system planning using dynamic programming technique and GIS facilities'",IEEE Trans. Power Deliv,vol. 17,pp. 242-247,2002.

[6] C. Monteiro, 1.1. Ramirez-Rosado, V. Miranda, P.J. Zorzano Santamaria, E. Garcia-Garrido, L.A Fernandez-Jimenez, "GIS spatial analysis applied to electric line routing optimization'", IEEE Trans. Power Deliv,vol. 20,pp. 934-942,2005.

[7] K.G Ahn,H. Rakha,"The effects of route choice decisions on vehicle energy consumption and emissions'", Transport. Res. Part 0-Transport. Environ,vol. 13,pp. 151-167,2008.

[8] D. R. Lanning, G. K. Harrell, and J. Wang, "Dijkstra's Algorithm and Google Maps,", in Proceedings of the 2014 ACM Southeast Regional Conference (ACM SE'14), New York, NY, United States, March 2014, No. 30, pp. 1-3.

[9] R. J. C. Chen, "Geographic information systems (GIS) applications in retail tourism and teaching curriculum," Journal of Retailing and Consumer Services, vol. 14, pp. 289-295, 2007.

[10] R. W. Calvoa, F. D. Luigi, P. Haastrup, and V. Maniezzo, "A distributed geographic information system for the daily car pooling problem," Computers & Operations Research, vol. 31, pp. 2263-2278, 2004.

[11] N. G. Boulaxis and M. P. Papadopoulos, "Optimal feeder routing in distribution system planning using dynamic programming technique and GIS facilities," IEEE Trans. Power Deliv., vol. 17, pp. 242-247, 2002.

[12] C. Monteiro, I. I. Ramirez-Rosado, V. Miranda, P. J. Zorzano-Santamaria, E. Garcia-Garrido, and L. A. Fernandez-Jimenez, "GIS spatial analysis applied to electric line routing optimization," IEEE Trans. Power Deliv., vol. 20, pp. 934-942, 2005.

[13] K. G. Ahn and H. Rakha, "The effects of route choice decisions on vehicle energy consumption and emissions," Transport. Res. Part D-Transport. Environ., vol. 13, pp. 151-167, 2008.

[14] L. Zhi, X. Zhou, and J. Zhao, “Vehicle Routing for Dynamic Road Network Based on Travel Time Reliability,” IEEE Access, vol. 8, pp. 190596–190604, 2020.

[15] P. Sembiring, A. S. Harahap, and K. S. Zalukhu, “Implementation of Dijkstra’s algorithm to find an effective route to avoid traffic jam on a busy hour,” J Phys Conf Ser, vol. 1116, p. 022042, Dec. 2018.

[16] L. Costa, C. Contardo, and G. Desaulniers, “Exact Branch-Price-andCut Algorithms for Vehicle Routing,” Transportation Science, vol. 53, no. 4, pp. 946–985, Jul. 2019.

[17] L. Liu et al., “Path Planning for Smart Car Based on Dijkstra Algorithm

and Dynamic Window Approach,” Wirel Commun Mob Comput, vol. 2021, pp. 1–12, Feb. 2021.

[18] A. A. Zubir, Andriansyah, and S. Derisma, “Determining the Distribution Route Using the Clustered Generalised Vehicle Routing Problem Model and Dijkstra’s Algorithm,” Journal of Science, Technology, and Visual Culture, vol. 2, no. 2, pp. 232–240, 2022.

[19] S. Kim, H. Jin, M. Seo, and D. Har, “Optimal Path Planning of Automated Guided Vehicle using Dijkstra Algorithm under Dynamic Conditions,” in 2019 7th International

Conference on Robot Intelligence Technology and Applications (RiTA), IEEE, pp. 231–236, Nov. 2019.

[20] R. D. Gunawan, R. Napianto, R. I. Borman, and I. Hanifah, “Implementation of Dijkstra’s Algorithm in Determining the Shortest Path (Case Study: Specialist Doctor Search in Bandar Lampung),” IJISCS (International Journal of Information System and Computer Science), vol. 3, no. 3, p. 98, Dec. 2019.

[21] P. Udhan, A. Ganeshkar, P. Murugesan, A. R. Permani, S. Sanjeeva, and P. Deshpande, “Vehicle Route Planning using Dynamically Weighted Dijkstra’s Algorithm with Traffic Prediction,” May 2022.

[22] S. Kirono, M. Irfan Arifianto, R. Eka Putra, and A. Musoleh, “GraphBased Modeling and Dijkstra Algorithm for Searching Vehicle Routes on Highways,” IAEME Publication, vol. 9, no. 8, pp. 1273–1280, Aug. 2018.

[23] A. Sedeño-noda and M. Colebrook, “A biobjective Dijkstra algorithm,” Eur J Oper Res, vol. 276, no. 1, pp. 106–118, Jul. 2019.

[24] A. Fitriansyah, N. W. Parwati, D. R. Wardhani, and N. Kustian, “Dijkstra’s Algorithm to Find Shortest Path of Tourist Destination in Bali,” J Phys Conf Ser, vol. 1338, no. 1, p. 012044, Oct. 2019.

[25] Y. Sun, M. Fang, and Y. Su, “AGV Path Planning based on Improved Dijkstra Algorithm,” J Phys Conf Ser, vol. 1746, no. 1, p. 012052, Jan. 2021.

[26] Suardinata, R. Rusmi, and M. A. Lubis, “Determining Travel Time and Fastest Route Using Dijkstra Algorithm and Google Map,” SISTEMASI: Jurnal Sistem Informasi, vol. 11, no. 2, pp. 496–505, May 2022.

[27] O. Khaing, Dr. H. H. Wai, and Dr. E. E. Myat, “Using Dijkstra’s Algorithm for Public Transportation System in Yangon Based on GIS,” International Journal of Science and Engineering Applications, vol. 7, no. 11, pp. 442–447, Nov. 2018.

[28] Y. Singh, S. Sharma, R. Sutton, D. Hatton, and A. Khan, “Feasibility study of a constrained Dijkstra approach for optimal path planning of an unmanned surface vehicle

in a dynamic maritime environment," IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC), IEEE, pp. 117–122, Apr. 2018.

[29] M. Luo, X. Hou, and J. Yang, "Surface Optimal Path Planning Using an Extended Dijkstra Algorithm," IEEE Access, vol. 8, pp. 147827– 147838, 2020.

[30] B. Li et al., "Path planning of mobile robots based on an improved A* algorithm," in Proceedings of the 2020 4th High Performance Computing and Cluster Technologies Conference & 2020 3rd International Conference on Big Data and Artificial Intelligence, New York, NY, USA: ACM, pp. 49–53 Jul. 2020.

[31] Yu, Jiang, and Hua, "Anti-Congestion Route Planning Scheme Based

on Dijkstra Algorithm for Automatic Valet Parking System," Applied

Sciences, vol. 9, no. 23, p. 5016, Nov. 2019.

[32] M. Kataoka,"GIS for Homeland Security'",ESRI Press,Redlands,CA, 2007.
[33] AR. Pradhan, D.F. Laefer, W.J. Rasdorf. "Infrastructure management information system framework requirements for disasters'",ASCE,Journal of Computing in Civil Engineering,vol.21,pp.90-101,2007.
[34] M.I(. Lindell, C.S. Prater, "A hurricane evacuation management decision support system (EMDSS)"" Natural Hazards,vol. 40,pp. 627-634, 2007.

[35] M.P. Kwan, 1. Lee, "Emergency response after 9/11: the potential of real-time 3d GIS for quick emergency response in micro-spatial environments'", Computing Environment and Urban Systems, vol. 29, pp. 93-113,2005.

[36] J. Han,Z. Yong,K.W. Dai,"The approach for shortest paths in fire succor based on component GIS technology'", SPIE-The International Society for Optical Engineering, Geoinformatics 2007, Geospatial Information Technology and Applications,vol. 6754,2007.

[37] Y.M. Fang, L.Y. Lin, C.H. Huang, T.Y. Chou, "An integrated information system for real estate agency-based on service-oriented architecture",Expert Systems with Applications,vol. 36,pp. 11039-11044, 2009.

[38] D. Nebert, A Whiteside, "OpenGIS catalogue services specification version2.0'", Technical report OGC 04-02Ir2, Open GIS Consortium Inc, 2004.
[39] S. Tu,E. Normand,S. Kuchimanchi,V. Bizot,S. Shu,M. Abdelguerfi, and all. "Integrating web services into map image applications", In Computers and communications, Proceedings of ISCC 2004. Ninth international symposium,pp.44-49, 2004.

[40] K.X. Tanga,N.M. Waters,'The internet,GIS and public participation in transportation planning'",Progress in Planning,vol. 64,pp. 7-62, 2005.

[41]. Malczewski, "GIS and Multicriteria Decision Analysis, John Wiley and Sons, New York'", 1999.

[42] G. Derekenaris, Garofalakis, C. Makris, Prentzas, S. Sioutas, A Tsakalidis, "Integrating GIS, GPS and GSM technologies for the effective management of ambulances'", Computers, Environment and Urban Systems,

vol. 25, pp. 267-278, 2001.

[43] Y. Choi, A Nieto, "Optimal haulage routing of off-road dump trucks in construction and mining sites using Google Earth and a modified leastcost path algorithm"', Automation in Construction, 2011.

[44] S. E. Houssaini and A. Sadri, "A web-based spatial decision support system for effective monitoring and routing problem," in Proc. of the International Conference on Geographic Information Systems (GIS), Mohammedia, Morocco, pp. 1-6, 2023.

[45] E. H. Al-Hemiary and S. A. Mohammed Ali, "A Parallel Implementation Method for Solving the Shortest Path Problem for Vehicular Networks," in Proc. 1st Int. Conf. Inf. Technol. Enhance E-Learning Other Appl. (IT-ELA 2020), Baghdad College of Economic Sciences University, Baghdad, Iraq, pp. 1-6, 2020.

Appendices

Appendices appear after references. Your appendices depend on the nature of your project. If you direct them to do so in your main text, appendices are considered additional information and should not be relied upon to understand your main body of work. Refer readers to an appendix using a phrase such as see Appendix A for further details.

The following documents must be included as references:

- Your Project Proposal.
- Evidence of your use of a project management tool.
- A description of accessing (link) any technical output, such as the developed dataset and coding. It is strongly recommended you use GitHub or something similar to do this.
- Provide a link to the video recording for demonstrating your artefact

For any important communications between you and external stakeholders, please remove private data and anonymise communications.

Use Heading back matter style for the headings of appendices.

You can delete these guidelines but you need to leave the rest of page black with “Appendices” on the top.

Appendix A: Project Proposal

Appendix B: Project Management

https://github.com/users/uzumstanley/projects/1/views/1

Appendix C: Artefact/Dataset

[1] https://github.com/uzumstanley/Traffic-Forecasting-For-Road-Network

[2] https://london-navigator.netlify.app/

Appendix D: Screencast

https://drive.google.com/file/d/1HY_bgT6l_k6twtfgnvQvg6aKiqKb4Qav/view?usp=sharing