Glasgow on the Go: Visualising Urban Mobility in Glasgow

by

Danial Sheikh

In partial fulfilment of the   
requirements for the degree of   
Bachelor of Science

in

CS408: Individual Project



Department of

Computer and Information Sciences

March, 2023

Table of Contents

[1 Introduction 3](#_Toc161228870)

[1.1 Problem Summary 3](#_Toc161228871)

[1.2 Aims and Objectives 3](#_Toc161228872)

[1.3 Outcomes of Project 4](#_Toc161228873)

[1.4 Report Structure 4](#_Toc161228874)

[2 Background Literature 6](#_Toc161228875)

[2.1 Academic Research on Urban Mobility 7](#_Toc161228876)

[2.2 Academic Research on GeoJSON Data 7](#_Toc161228877)

[2.3 Comparison Analysis of similar applications 7](#_Toc161228878)

[3 Specification and Design 9](#_Toc161228879)

[3.1 Original Specification 9](#_Toc161228880)

[3.2 Methodology 9](#_Toc161228881)

[3.3 Requirements 10](#_Toc161228882)

[3.3.1 Functional Requirements 10](#_Toc161228883)

[3.3.2 Non-Functional Requirements 10](#_Toc161228884)

[4 System Design 12](#_Toc161228885)

[4.1 Design of Project 12](#_Toc161228886)

[4.2 High-Level Architecture 12](#_Toc161228887)

[4.3 User-Centred Design 13](#_Toc161228888)

[4.4 Prototyping and Iterations 13](#_Toc161228889)

[4.5 Design Challenges 14](#_Toc161228890)

[4.6 Compliance and Standards 14](#_Toc161228891)

[5 Detailed Design and Implementation 15](#_Toc161228892)

[5.1 Technology Stack 15](#_Toc161228893)

[5.1.1 DEVWEB 15](#_Toc161228894)

[5.1.2 Gitlab 15](#_Toc161228895)

[5.1.3 IntelliJ 15](#_Toc161228896)

[5.1.4 REST API 15](#_Toc161228897)

[5.1.5 HTML 5 16](#_Toc161228898)

[5.1.6 CSS 16](#_Toc161228899)

[5.1.7 JavaScript 16](#_Toc161228900)

[5.1.8 Bootstrap 5 16](#_Toc161228901)

[5.1.9 Leaflet 16](#_Toc161228902)

[5.1.10 Chart.js 17](#_Toc161228903)

[6 Results & Evaluation 19](#_Toc161228904)

[6.2 Results of Evaluation 19](#_Toc161228905)

[6.3 Returning to the Research Questions 19](#_Toc161228906)

[6.3.1 RQ1 19](#_Toc161228907)

[6.3.2 RQ2 19](#_Toc161228908)

[6.3.3 Objective 1 19](#_Toc161228909)

[6.3.4 Objective 2 19](#_Toc161228910)

[7 Discussion & Reflection 20](#_Toc161228911)

[7.1 Interpreting the Results 20](#_Toc161228912)

[7.2 Reflection 20](#_Toc161228913)

[7.3 Challenges 20](#_Toc161228914)

[7.4 Limitations 20](#_Toc161228915)

[7.5 Future Work 20](#_Toc161228916)

[8 Conclusion 21](#_Toc161228917)

[A Appendix 23](#_Toc161228918)

[B 23](#_Toc161228919)

[B.1 Ethical Approval Form 23](#_Toc161228920)

[B.2 Participant Information Sheet 23](#_Toc161228921)

[B.3 Consent Form 23](#_Toc161228922)

[B.4 Marking Scheme 24](#_Toc161228923)

**Abstract**

The abstract is a bird’s eye view of the project.

It should not exceed one page. Mention the scope and objectives of the project, the methodology used, the main findings, and significance of your results.

The page count of 12 is exclusive of bibliography and appendices.

Acknowledgements

Firstly, I would like to thank my supervisor for this project Dr Fredrik Nordvall Forsberg, for his advice and support throughout the project, particularly for his unwavering patience when my code would not cooperate. I would also like to thank Dr Alex Coddington, for the valuable feedback on my presentation that gave me a different perspective on how to view it.

I would also like to extend my gratitude to everyone who participated in my user study as they gave me feedback on how I could further improve my application.

# Introduction

## Problem Summary

Glasgow is a bustling and vibrant city, the way people navigate their way around it has a huge impact on important matters such as traffic jams to air pollution levels. According to the Glasgow times “Glasgow’s population has been steadily increasing since 2006, when it was 568,480. In 2019, it was recorded as 633,120” [1]. This figure is expected to continue rising putting increased pressure and demands on the existing infrastructure especially with implications for transport planning on future expansion needs. An article from STV highlighted this issue with people on record saying ““It’s the worst it’s ever been” [2]. That’s what people who rely on public transport in Glasgow are saying about the services.” Due to the rising transport costs and lack of services available this will lead to an increase in car usage. This will discourage healthier and more sustainable modes of travel like walking and cycling. Addressing the problems of urban mobility can pave the way for a sustainable Glasgow that benefits future generations to come.

## Aims and Objectives

The aim of this project is to monitor and visualise urban mobility in Glasgow. I decided to do this by utilising datasets provided by the Glasgow Open Data hub and visualise them in an appealing way whilst being easy to understand. This was done by designing an application using JavaScript as the primary programming language. The three different datasets from Glasgow Open Data Hub highlight pedestrian, cyclist, and traffic flow throughout Glasgow. I decided the best approach to retrieve and access the data would be through Application Programming Interface (API) for each dataset. Users can enter two specific dates from when they want to view the data from, and this is presented to them through an interactive map and graph.

To achieve the main aim of this project the project, I followed six key objectives throughout it:

* Collect and display the data from the Glasgow Open Data hub. As the data is directly from the Glasgow City Council which is a credible and concrete source.
* The data should cover pedestrians, cyclists and vehicles to fully showcase Glasgow’s urban mobility.
* Allowing users to enter a specific date range and filter results, this ensures the data is relevant to their needs.
* Develop and implementing an interactive graph and map.
* Helping users visualise the data in a clear and concise manner.

## Outcomes of Project

The outcome of this project has resulted in a fully functional application that illustrates urban mobility in Glasgow. The application provides users access to the most current and historical data pedestrian, cyclist, and traffic data. The application consists of a homepage and three separate pages for both This offers an insight into Glasgow’s urban mobility overtime. The application displays the data through an interactive map and graph allowing giving users a different way to visualise the data. The map and graph are connected as if you click on a certain bar in the graph, it will take you to that exact location on the map. There are various filter options that allow the user to get the relevant results.

Based on user testing …

## Report Structure

Include (1) the problem you have tackled, (2) why this problem is worth addressing, (3) what you did to address it - in broad terms. Detail will come later.

i.e., issue(s) on which the research will focus, shall be clearly identified, and described. You shall refer to past research work relevant to the topic and objectives, i.e, of the study. You shall outline where applicable the potential research output with respect to research transfer and uptake by the community.

The introduction says: this is an overview of the project. This is why I did it (the problem) and how I tackled it. It is the runway into the project. It lets the marker know what to expect of your report.

If you’re doing a research project, this would be the place to include the research questions you plan to address. For example:

RQ1: Where did James Bond come from?

RQ2: Why are pumpkins orange?

If you’re doing a project of type 1, include a list of objectives. For example:

Objective 1: Provide software to allow James Bond to become invisible.

Objective 2: Provide software to keep track of all loyalty points in one place.

Provide a ‘map’ of your dissertation. For example: Section ?? reviews the background literature that was reviewed to inform this project. Then Section ??.....

# Background Literature

This section should provide the grounding for your project. You will be able to refer back to this in Section ??, where you report on design decisions and methodology.

There should be a strong link to your research questions (for project types 2 and 3). There will be a strong link to the software design decisions for project type 1.

Don’t just include a paragraph for each paper you read. Synthesize it and create a story line.

Review between 5 and 10 authoritative sources. For research based projects, some of these should be from peer-reviewed journals or conferences.

This section will reference literature like this [?]. Sloppy referencing will (1) take time and effort, and (2) lose you marks if you don’t do it.

## Academic Research on Urban Mobility

The paper “Transport Issues and Sustainable Mobility in Smart Cities” talks about the complexity of urban mobility and city sustainability [3]. It highlights the significance of using geographical data to clearly illustrate urban planning and policymaking, a concept key to my project on visualising Glasgow's urban mobility. Furthermore, the paper also advocates about the importance of about the need for tools that offer real-time, accessible information. This reinforces the aim of my project to create an interactive map that can help users observe mobility patterns and infrastructure planning in Glasgow.

## Academic Research on GeoJSON Data

For urban mobility GeoJSON is a powerful tool for visualising complex datasets. This is proven by Esri’s an analysis of human mobility patterns using this data tool [4]. This article suggests the methodological approach of creating an interactive map of Glasgow, using GeoJSON is a strong approach to depict urban mobility.

## Comparison Analysis of similar applications

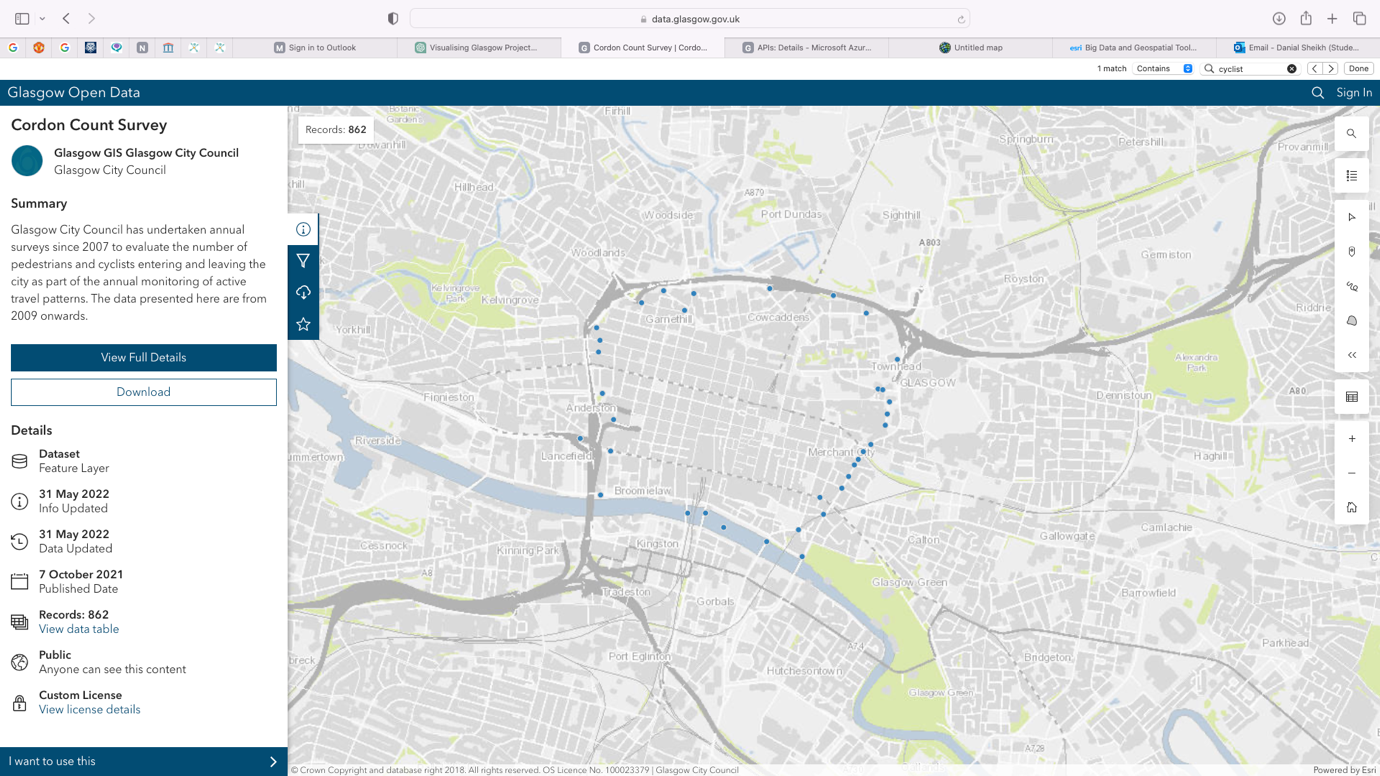
The case study carried out on visualising São Paulo's urban mobility with various techniques offers valuable insights into data visualisation [4]. The paper illustrates on how complex mobility data can be simplified to ensure users understand it. For presenting and analysing the urban mobility data the study used the bundling method. This technique is used to simplify and clarify the presentation of large sets of data. I am hoping I can apply this to present Glasgow’s mobility data on an interactive map.

A map of different types of buses

Description automatically generated

*Figure 1: "Naïve straight-line drawing of OD trails form the Metropolitan area of São Paulo. [4]"*

Figure 1 shows “OD Survey trajectories plotted over the map of the São Paulo metropolitan area, where each line represents an OD trail. The extreme clutter in this figure precludes the visualization of individual trajectories, traffic patterns, or connections between the regions of the map. [4]”.



*Figure 2: Glasgow Open Data Hub Cordon count map*

The map in figure 2 from the Glasgow Open Data Hub displays the Cordon Count Survey data, which shows the number of pedestrians and cyclists. It is critical to show Glasgow’s active travel and urban mobility. It offers data from as early as 2009. I am hoping my application is similar but is more interactive and visually appealing to users.

# Specification and Design

Describe all details of the design and procedures used to achieve the project objectives. Do this chronologically.

It should be detailed enough to allow for an assessment of the rigour of your process.

Say which software lifecycle approach you used e.g., Waterfall, Spiral, Agile. How did you gather user requirements?

## Original Specification

The foundations of this project were conceptualised from my supervisor, Dr Fredrik Nordvall Forsberg. The original project specification read as follows: “Glasgow City Council provides vast amount of data through the Glasgow Open data Hub <https://data.glasgow.gov.uk/> . The goal of this project is to select some of these datasets that you find interesting and visualise them in an appealing way. You might want to take inspiration from other visualisations such as <https://engaging-data.com/> . The choice of implementation language is up to you, but clearly a high-level language with good graphics support such as Python or JavaScript might be preferable.” When reviewing the datasets provided by Glasgow Open Data Hub, I was drawn to the untapped potential of urban mobility data. Thus, I directed my efforts towards developing an application that visualises urban mobility in Glasgow This would of making the data visually appealing and easy to understand for both public use and urban planning.

## Methodology

Which methodology did you choose?

In developing the application, I opted to use an Agile development approach. A key reason behind this was its flexibility and capacity for iterative development. I met with my supervisor Fredrik Forsberg Nordvall every fortnight for our stand-up meetings as this allows for regular feedback and continual improvement, when building the application. To help manage my workload effectively and ensure we organised it into sprints. When completing tasks for the sprint I added them to a logbook which allowed me to track my progress throughout this project. Please see figure 3.1 in appendix. This strategy ensured the setting of realistic deadlines but allowed for the flexibility to adapt to the evolving requirements of the application. The consistent dialogue ensured that my project remained align with its aim and objectives.

When selecting an agile methodology for the project over waterfall or spiral, three key advantages stood out to me:

* Flexibility and responsiveness: Agile enabled me to make quick adjustments to my project based on the user feedback and continuous insights from my supervisor. This would present a challenge if I used a waterfall method instead due to its rigid structure.
* Efficiency and speed: Agile sprints accelerated the application’s development allowing for more prototyping and sprint iterations. This was instrumental for improving the applications features particularly for visualising the data. A task that would have been delayed under the linear approach of waterfall.
* Improve Risk Management: Agile allowed me to make swift changes during development to my application based on suggestions from my advisor and issues I was facing. Whereas a spiral model while methodical in addressing risks through pre-planning, it would not offer the same adaptability for solving real time problems as they arise.

## Requirements

### Functional Requirements

The functional requirements for my final application are as follows:

* Allow users to enter specific from when they want to view the data.
* Users can filter results to match what they are specifically looking for.
* The application will allow users to view an interactive map and graph of Glasgow.
* Users will be able to see data on traffic, cyclists, and pedestrians in Glasgow.
* Users can interact with the map and graph using both of their interactive features.
* It must be easy to navigate and visually appealing.

### Non-Functional Requirements

The non-functional requirements for my final application are as follows:

* It must be compatible with modern web browsers and mobile devices.
* The application should be accessible according to WCAG standards (Web Content Accessibility Guidelines).
* The application should always update in real-time data becomes available from Glasgow Open Data Hub APIs.
* Ensure the interactive graph is downloadable for people wanting to use it for further work.
* It must be easy to navigate and visually appealing.

### User Stories

* As a commuter, I want to check traffic so I can avoid congested routes.
* As a city planner, I want to understand peak footfall areas to better plan urban infrastructure.
* As a researcher, I want to extract data for academic purposes to facilitate my study of urban trends.
* As a policy maker, I need to compare historical and current data to assess the impact of new policies.
* As an urban developer, I want to access pedestrian flow patterns to design better public spaces.
* As someone relocating to Glasgow, I would like to find the least congested parts of the city, so I can live in a noise free environment.

### System Use Cases

* A user selects a location to view traffic trends for that area.
* A user clicks on a pin on the map to get more details about footfall at a specific location.
* The developer in charge of the application (me) should ensure the information from the API is processed correctly and displayed in real time to the user.
* A user filters the data for a specific time period so they can study its impact on urban mobility.
* A user navigates between pages, and it is easy to find out what they are looking for.

# System Design

## **Design of Project**

This chapter gives an insight into the design process of my application, detailing how each design decision aligns with my project aim and objectives. It will unpack the rationale behind the selected technology stack, the architecture design of the application and the user focused approach. It will also touch upon the strategies implemented to ensure accessibility standards are met and performance is as expected.

## **High-Level Architecture**

For structuring the application, I decided to follow a Model View Controller (MVC) approach. MVC helps break up front end and back-end code into separate components making the code more manageable:

* **Model**- The model is the backend of my application. It interacts with the Glasgow Open Data hub API to fetch the data for each different page, whilst also querying logic of user requests.
* **View**- The front end of the application and presents the processing of the sorted data from the model. It presents the data in an easy to understand and user-friendly way, by generating the interactive map and graph. It updates based on the data provided by the model through the controller.
* **Controller**- The brains of the application by handling user interactions and application responses. It makes calls to the model to retrieve the appropriate data based on user requests, such as selecting certain timeframes for mobility data, and then passes this to data to the View to make it presentable.

The data flow in the application begins with the user interacts with the interface, entering two dates and clicking the submit button. The action then triggers the controller which uses data logic methods in the model to sanitise and validate user input, this ensures the dates are in the correct format. After this happens the Controller calls the Model’s “getDataFromAPI” method and passes in the two dates. The model makes an API request to Glasgow Open Data Hub and processes the received data. Once the data is ready, the Model sends it to the View where it is used to render the interactive map and graph. Visually representing the traffic, pedestrian or cyclist data depending on what page they are on.

A diagram of a computer

Description automatically generated

*Figure 3: This shows the high-level architecture of the application.*

## **User-Interface** Design

This part of my application examines the user interface design process of the application. It shows how I focused on user centred design principles which guided the creation of a responsive, accessible, and aesthetically pleasing user experience. Then will discuss about the creation of user interface mock-ups to help visualise the layout of the page elements.

### User Interface Mockups and Role in Design

During the initial phase of the design process, various user interface (UI) mock-ups were created to conceptualise the layout and interaction flow of the application. These mockups were created using an online software called Figma. These mock-ups served as a visual guide, allowing for early detection of potential usability issues, and ensuring the functional and non-functional requirements of the application are met.

A screenshot of a green screen

Description automatically generated*Figure 4: This shows the Homepage Mockup*

A screenshot of a computer

Description automatically generated*Figure 5: This shows the Traffic Page Mockup (besides the title and description at the top, has the exact same layout/ functionality for other pages.*

In figure 4 the homepage serves as a getaway to the application. It embodies the essence of the system use cases by providing an intuitive navigation system that allows users to commence their interaction effortlessly. Also inline with the non-functional requirement, the application adjusts across devices screen size and adheres to WCAG 2.1AA standards, ensuring it is accessible to all users. The prominence of navigation features in the mockup underscores the application's commitment to intuitive user experience, a core principle of user-centred design.

In figure 5 shows the Traffic Page Mockup which illustrates the application’s core functionally the visual representation of traffic data. The mockup shows the user interface for selecting dates and filtering options which are in align in meeting the two key functional requirements. The layout of this page is a testament to user stories such as understand peak footfall areas to better plan urban infrastructure and extract data for academic purposes.

For both cases the system uses case of user navigating between pages, and it is easy to find out what they are looking for, has been successfully done by the responsive navbar at the top of the page. Together, these mockups signify an adherence to a user-centred approach, ensuring that the final application is not only about data visualization but also about delivering a coherent and effortless user journey.

### Compliance with Accessibility Standards

In today’s digital landscape, accessibility is not just a courtesy but a crucial requirement. This has been accounted for when developing the application following the Web Content Accessibility Guidelines (WCAG) to cater to users with diverse abilities. The applications adhere to WCAG 2.1 level AA standards, which ensures accessibility for a wide range of users with different needs. (google WCAG standards 2.1 AA). I did this by adding ARIA (Accessible Rich Internet Applications). This provides screen readers users with meaningful description of interactive elements especially when trying to use the interactive map and graph. The use of ARIA labels combined with a clear font ad layout ensures that all users especially with visual impairments, can navigate and interact with the application effectively.

The application compliance with accessibility was evaluated on google chrome via the WAVE tool, a respected web accessibility evaluation tool. The WAVE tool ensured I had complied with accessibility Standards. As you see in the figure below it passed all tests, and no issues were found.

A screenshot of a computer

Description automatically generated

*Figure 6: This shows the Wave Tool applied to the top half of the Pedestrian Page which shows there are no issues and it is compliant with accessibility standards.*

A screenshot of a computer

Description automatically generated

*Figure 6: This shows the Wave Tool applied to the second half of the Pedestrian Page which shows there are no issues, and it is compliant with accessibility standards*

### Responsive Navigation Bar

The navigation bar design was engineered to be fully responsive, which is a key feature that ensures that the content is adapted for various screen sizes and resolutions in alignment with mobile-first design principles. For smaller screens, the navigation options turn into a ‘hamburger’ icon- a familiar element that users can tap to expand the menu. This ensures that the navigation remains accessible without crowding the screen, ensuring users have a seamless experience on smaller screen sizes.

### Aesthetic Design Choices

For the application I have decided to choose a green colour scheme, it reflects a commitment to creating a visually soothing experience while still promoting focus and readability. This colour also reinforces the theme of urban mobility with association with progress and sustainability.

The application adapts the font size dynamically to ensure legibility on various screens, from large desktop monitors to compact mobile displays. This responsive typography ensures that the content on the screen is easy to digest, regardless of the device.

Interactivity is a key aspect of the user experience. A notable feature is the integration between the data representations clicking on a bar with the interactive bar chart opens the corresponding detailed pinpoint on the map. This interaction allows users to have a better analysis experience by diving into deeper specific data points without losing context.

User interface elements include a forms, button and fields are designed with a clear hierarchy and ample spacing. They are responsive to user screen size and accommodate a wide range of user interactions ensuring the application remains approachable and intuitive across all devices.

# Detailed Design and Implementation

Once the system design had been decided, the actual development of the application then took place. Since the user interface mockups had been created, it made the implementing and layout of the application a lot more straightforward.

## **Technology Stack**

A numerous number of technologies were used to ensure the project was completed to a high standard and ensure the aims and objectives were met.

### **DEVWEB**

The application was hosted on the University of Strathclyde devweb server. The server allowed me to see how the application functioned in a real time environment whilst also quickly uploading the changes I was making during development. In a later chapter for my user testing, it allowed multiple users to use the application through a URL without having to go through the set up on their machine.

### **Gitlab**

This was the platform of the version control provided by the university. I used this to commit, pull and push versions of my application’s source code. I picked this as I had used it before, and it was intuitive to use. This and the DEVWEB sectioned mentioned above gave me another independent source to back up my project files.

### **IntelliJ**

IntelliJ is an integrated development environment, that was used to develop and program my application. I chose it as I had previous experience of using it for projects and allowed for the easy integration of frameworks and plugins throughout my project. Examples of this being the easy upload of my code to the DEVWEB server or to Gitlab.

### **CORS Proxy Sever**

In web development, Cross-Origin Resource Sharing (CORS) is a security feature that restricts web applications that make cross-origin HTTP requests. During development I could not make a request to the Glasgow Open Data Hub API because of this security problem. To overcome this, I utilised a CORS proxy server (<https://corsproxy.io/>). This proxy acts as an intermediary which adds the necessary CORS headers to the responses from the external APIs hosted on the Glasgow Open Data Hub.

let urlPedestrian = `https://corsproxy.io/?` + encodeURIComponent(`https://api.glasgow.gov.uk/mobility/v1/footfall/historical?format=json&startDate=${startDate}&endDate=${endDate}`);  
let urlCyclist = `https://corsproxy.io/?` + encodeURIComponent(`https://api.glasgow.gov.uk/mobility/v1/Mobility\_Measurements?format=json&period=day&type=bicycle&date=${startDate}&end=${endDate}`);  
let urlTraffic = `https://corsproxy.io/?` + encodeURIComponent(`https://api.glasgow.gov.uk/traffic/v1/movement/query?start=${startDate}&end=${endDate}&period=Day`);

*Figure 7 shows ensuring User ‘GET’ requests are going through the CORS proxy server.*

The CORS proxy server played a pivotal role in my application as without its users could not fetch any urban Mobility Data. While this approach facilitated the development process by allowing API requests to be made without direct CORS support, it is recognised that using a proxy may introduce security considerations that would not be suitable for a production environment.

### **REST API**

When it came to accessing the data from the Glasgow Open data Hub I did so through a REST API (Application Programming Interface). Representational State Transfer (REST) is an architectural style that allows communication between a client and a server in a scalable and stateless manner, primarily using HTTP requests [2]. This approach allows for efficient data retrieval from Glasgow Open Data Hub via HTTP GET requests, aligning with REST principles to provide a fast and seamless user experience.

The GET request, when successful, returns a JSON response containing the data relevant to the user's query, which includes the count and location from the sensor, as well as the date range specified by the user. This data is then parsed and manipulated to extract the precise information needed for display from the response. The process involves filtering the JSON response to isolate specific values and transforming them into a format that can be easily used to plot on the map and graph. This step is crucial as it ensures that users receive a tailored view of the urban mobility data they are interested in.

### **HTML 5**

Hypertext Mark-up Language 5 is used to structure website design. I used this to structure the four webpages and manipulate the elements through JavaScript.

### **CSS**

Cascading Style Sheets are used to style a HTML document, thus making it more visually appealing to users.

### **JavaScript**

JavaScript is a scripting language that allowed me to add client-side functionality to my webpages so users could interact with them. This played a significant part in allowing me to make all my webpages responsive. Close to all my functionality relied on JavaScript throughout developing my application.

### **Bootstrap 5**

Bootstrap is a CSS framework that played a key role in assisting to make the application responsive and compatible with both mobile and desktop devices. The bootstrap framework was selected to help with the responsiveness of the application through its container system. Bootstrap's container system utilises a series of responsive, fixed-width, and fluid-width settings that serve as the foundation for the layout of the application [3]. Alongside media queries this ensured content alignment and consistency across different screen sizes.

### **Leaflet**

Leaflet is a light weight open-source JavaScript library for the development of interactive maps (https://leafletjs.com/). The key features of leaflet were that its selection in this project include its lightweight nature, vast documentation, and variety of plugins. It being also desktop and mobile friendly to ensure user friendliness and accessibility. Leaflet allowed me to ensure the completion of my project objective of building an interactive map.

### **Chart.js**

Chart.js is a versatile open-source JavaScript library that specializes in creating interactive and animated charts with a simple API (<https://www.chartjs.org/docs/latest/>) . Its integration into the project was pivotal for presenting urban mobility data in an engaging and easily digestible format. The utilisation of Chart.js contributed to the achievement of the project's objectives by making statistical data on urban mobility both accessible and understandable to a broad audience. This aligns with the aim of not only visualising data but also facilitating its interpretation and analysis through interactive elements.

## **Application Development Process**

### **Early Development and Focus on the Pedestrian Footfall**

const aggregatedData = {};  
if (dataType === 'pedestrian') {  
  
  
 data.forEach(item => {  
 const street = item.location.description;  
 const coordinates = [item.location.longitude, item.location.latitude];  
 const date = new ***Date***(item.processDate);  
 const day = String(date.getUTCDate()).padStart(2, '0');  
 const month = String(date.getUTCMonth() + 1).padStart(2, '0');  
 const year = date.getUTCFullYear();  
 const monthYear = `${year}-${month}-${day}`;  
  
 if (!aggregatedData[street]) {  
 // If the street key doesn't exist, create it with initial values  
 aggregatedData[street] = {  
 pedestrianCount: 0,  
 coordinates: coordinates,  
 description: street,  
 dates: [] // Array to store all dates for this street  
 };  
 }  
  
 // Aggregate the pedestrian counts for the street key  
 aggregatedData[street].pedestrianCount += item.pedestrianCount;  
 if (!aggregatedData[street].dates.includes(monthYear)) {  
 aggregatedData[street].dates.push(monthYear);  
 }  
 });  
  
 // Convert the aggregated data into an array of GeoJSON features  
 const features = ***Object***.keys(aggregatedData).map(key => {  
 const item = aggregatedData[key];  
 return {  
 "type": "Feature",  
 "properties": {  
 "pedestrianCount": item.pedestrianCount,  
 "dates": item.dates.join(', '), // Join dates as a string  
 "description": item.description  
 },  
 "geometry": {  
 "type": "Point",  
 "coordinates": item.coordinates  
 }  
 };  
 });  
 return {  
 "type": "FeatureCollection",  
 "features": features

*Figure 8: Shows the conversion of the Pedestrian JSON data to GeoJSON.*

The initial phase of my development journey began with a concentrated effort on the pedestrian footfall feature. It began with making a http get request to footfall API. Then the server returns with a JSON response if everything goes smoothly which I then parse to get the pedestrian count and location for the start date and end date the user has inputted. Then the server responds with the return of JSON data which is converted into GeoJSON. The GEOJSON data is then plotted on the interactive map of Glasgow using leaflet. This established the foundational functionality and gave me a platform to build off going forward.

### **Using a MVC structure moving forward**

The successful implementation of pedestrian data visualisation in the application led me to refactor my codebase. Adopting the Model View Controller (MVC pattern), I enhanced the code’s structure and maintainability for the application for long term use going forward.

### **Decision to Diversify Pages**

A screenshot of a computer

Description automatically generated

*Figure 9: Highlights the Cyclist tab as you are on the cyclist page.*

Originally, the application was envisioned as a single page. However, accessibility concerns and user design considerations steered it towards a multi-page approach. A homepage which introduces what the application does and dedicated pages for cyclist, traffic, and pedestrian data. Each consistent in style and interactivity as well as a user-friendly navigation bar to link them all together. In figure 9 you can see the cyclist tab highlighted which correlates to the user being on the cyclist page.

### **Graph Integration and Interactivity**

A screenshot of a map

Description automatically generated

*Figure 10: Shows clicking on a bar on the graph opens the pop-up marker of the same location on the map.*

Between January to March, I focused on adding a graph (using chart JS) to accompany the map, thus providing a dual approach on data visualisation. This met many functional and non-functional requirements but also provided users with the flexibility to choose their preferred data analysis method. An interactive feature was added where if a user clicked on a data point on the graph it would highlight the corresponding location on the map, which gives the user the choice of using both simultaneously. This is illustrated in figure 10.

Incorporating real-time user feedback mechanisms, I integrated a loading spinner using Bootstrap and custom CSS. This feature reassured users during data processing and retrieval, also will changing pages. After this I added in a data filtering option, meaning users could sort results by varying criteria, including the ability to hide zero counts, thus fine-tuning data displayed according to their preferences and relevance.

### **Styling with Accessibility in Mind**

A screenshot of a computer

Description automatically generated

*Figure 11: Uses Chrome’s to simulate how an application would look on an iPad.*

The visual design phase included a combination of Bootstrap and CSS while complying with WCAG 2.1 AA standards. This ensured that the design of the application was visually appealing but also accessible to a broad spectrum of users. I utilised Bootstrap for its robust grid system and responsive features that allowed for a visually cohesive layout across different devices. I tested this for various device screen sizes using Chrome’s developer tools to emulate different screen sizes. As you can see in figure 11 it shows how the application would look on a smaller iPad screen. Elements such as colour contrast, font size and button spacing were meticulously fine-tuned to meet accessibility guidelines.

# Verification and Validation

## Testing Strategy

This chapter delves into the rigorous testing strategy used to ensure my final application met the highest standards of quality and functionality. It encompasses an in-depth exploration of the multi-layered testing process, which includes unit testing to validate individual components, user testing to assess the user experience, and continuous testing throughout the development cycle to maintain code integrity and address issues in real-time.

## Testing During Development

During development user input for inputting the start and end date was tested to ensure the input was valid and in line with the timeline available for the data depending what page they are on. If the user types in a date that is not in range or an invalid format then clicks the get data button a window. alert message will tell them what to change in their query. This is done through various functions in the Model class.

## User Testing

## Testing During Development

# Results & Evaluation

Results are often presented in tables, figures and other relevant illustrations. Include text that refers to these figures/tables.

**6.1 Evaluation Process**

If you involved humans in the evaluation, how many did you have? What can you say about the demographics of your participants? (If you did collect these)

In terms of the user interface, how did you carry out a usability evaluation, how did you go about doing this? How did you recruit participants?

In terms of delivering functionality, did you carry out user acceptance testing? (see attached guidance).

### 6.2 Results of Evaluation

This Section includes a direct interpretation of the gathered data and evaluation processes.

### 6.3 Returning to the Research Questions

Return to research questions or objectives as appropriate.

### 6.3.1 RQ1

It is clear from our findings that James Bond was born in Wigtown in Scotland. However, he grew up in Diss, in Norfolk. We know this because ....

### 6.3.2 RQ2

We were not able to answer this question from our studies, although some suggestions were made. These could not be proven.

### 6.3.3 Objective 1

### 6.3.4 Objective 2

# Discussion & Reflection

## Interpreting the Results

Here you will discuss your findings. This is especially relevant for research projects. You might interpret what the data and evaluation implies, both for future research and for practice (if appropriate).

The discussion is not a review of literature. You should try to compare research findings with previous work, provide explanations for your findings, discuss research findings, in terms of their contribution.

## Reflection

Look back and think about what you would do differently if you were going to start the project with the knowledge you have now. Be honest about your mistakes or missteps.

## Challenges

This is not the place to mention personal circumstances but rather challenges related to the work involved in the project.

## Limitations

Acknowledge things like small number of participants, software wasn’t completely debugged, or whatever else went wrong and affected your project. Include as appropriate.

## Future Work

If someone else wanted to build on your project’s product, what would be cool to do next?

# Conclusion

The conclusion is similar to when a plane lands. You don’t rewrite the introduction. You say something like - I addressed the problem outlined in the introduction, and I built some software to do this. I tested the software like this ADD FEW WORDS.

Summarize main findings drawn from the project work. Mention the objectives or research questions. Do not repeat points raised in the discussion and reflection Section. If applicable, you can make recommendations. The conclusion should NOT contain any references.

References

|  |  |
| --- | --- |
| [1]  [2] | D. Sandelands, "Latest figures show rise Glasgow's population due net-migration," Glasgow Times, 29-Jun-2021. [Online]. Available: <https://www.glasgowtimes.co.uk/news/19404825.latest-figures-show-rise-glasgows-population-due-net-migration/> . [Accessed: 21-Nov-2023].  [2] P. Bartlett, "Central Glasgow public transport the worst it's ever been say residents," STV News, 12-Feb-2022. [Online]. Available: <https://news.stv.tv/west-central/glasgow-public-transport-the-worst-its-ever-been-say-residents> [Accessed: 21-Nov-2023]. |

[3] Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of California, Irvine.

[4] https://getbootstrap.com/docs/5.0/layout/containers/

[5] https://leafletjs.com

[6]https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/

[7]https://routledge.com/Data-Visualization-Charts-Maps-and-Interactive-Graphics/Grant/p/book/9781138707603

# Appendix A- Model Checks for Start and End Date

* 1. A screenshot of a computer program

     Description automatically generatedModel Class Methods

The method on line 392 ensures the start date is before the end date and the other method on line 401 ensures for the data type (i.e. pedestrian cyclist or traffic) there is data recorded between that timeline.

A screenshot of a computer program

Description automatically generatedThe isValidDate(dateString) ensues the user has inputted the date in the expected format and isDateNotInFuture(startDate,endDate) checks to ensure the date inputted is not in the future.

* 1. Ethical Approval Form
  2. Ethical Approval Form

If your project required you to do any evaluation with humans, you MUST include this. It can be downloaded from the Ethics system.

https://local.cis.strath.ac.uk/wp/extras/ethics/index.php

* 1. Participant Information Sheet

If your project required you to do any evaluation with humans, you MUST include this

https://www.strath.ac.uk/ethics/informationsheetandconsentform/

* 1. Consent Form

If your project required you to do any evaluation with humans, you MUST include this.

<https://www.strath.ac.uk/ethics/informationsheetandconsentform/>

* 1. Marking Scheme

REMEMBER TO DELETE THIS. IT IS ONLY INCLUDED FOR your INFORMATION.

Table

Description automatically generated