Design and Implementation of a Cycling Navigation Web Application

A red square with white text

Description automatically generated

A Dissertation Presented

By

*Gulied Abdi*

20022743

Submitted To The University of The West of England To fulfil the degree of:

*Software Engineering for Business*

April 2024

Department of Environment and Technology – Computer Science and Creative Technologies

# **Abstract**

This report outlines the design and implementation of a web application specifically targeted at cyclists, addressing the lack of dedicated tools that provide accurate and practical routing information for cycling. Focusing on the need for detailed, bike-friendly navigation, the application integrates functional requirements such as real-time location tracking, cross-device compatibility, and point-to-point navigation. Non-functional requirements include the ability to find nearby bike shops and turn-by-turn navigation tailored for cyclists are also incorporated to enhance the overall user experience. The development process employed Agile methodologies, focusing on iterative design and testing with direct input from the cycling community to ensure the application’s effectiveness and usability. This report discusses the technical challenges encountered and the solutions implemented of this navigation website on promoting safer and more enjoyable cycling experiences.

# **Table of Contents**

[**Abstract** 2](#_Toc166278327)

[**Table of Contents** 3](#_Toc166278328)

[**Introduction** 5](#_Toc166278329)

[**Project Overview** 5](#_Toc166278330)

[**Report Outline** 6](#_Toc166278331)

[**Research** 7](#_Toc166278332)

[**Secondary Research** 7](#_Toc166278333)

[**Evaluation of Sources:** 7](#_Toc166278334)

[**Requirements Analysis** 10](#_Toc166278335)

[**Project Proposal** 11](#_Toc166278336)

[**Problem statement:** 11](#_Toc166278337)

[**Proposed solution:** 11](#_Toc166278338)

[**Success criteria:** 11](#_Toc166278339)

[**Research resources:** 11](#_Toc166278340)

[**Defining requirements** 12](#_Toc166278341)

[**Functional requirements:** 12](#_Toc166278342)

[**Non-Functional requirements:** 14](#_Toc166278343)

[**User Stories** 15](#_Toc166278344)

[**Design and Methodology** 17](#_Toc166278345)

[**Methodology** 17](#_Toc166278346)

[**Agile** 17](#_Toc166278347)

[**Design** 18](#_Toc166278348)

[**Wireframes** 18](#_Toc166278349)

[**Realisation of wireframes** 19](#_Toc166278350)

[**Map Design** 20](#_Toc166278351)

[**Testing** 22](#_Toc166278352)

[**Unit Testing** 22](#_Toc166278353)

[**Deployment & Version Control** 23](#_Toc166278354)

[**Netlify:** 23](#_Toc166278355)

[**Git / GitHub:** 23](#_Toc166278356)

[**Risk Analysis** 24](#_Toc166278357)

[**Considerations** 24](#_Toc166278358)

[**Future of the project** 25](#_Toc166278359)

[**Reflection and Lessons Learned** 26](#_Toc166278360)

[**Successes** 26](#_Toc166278361)

[**Challenges** 26](#_Toc166278362)

[**Conclusion** 27](#_Toc166278363)

[**Appendix** 28](#_Toc166278364)

[**Bibliography** 28](#_Toc166278365)

# **Introduction**

## **Project Overview**

The primary objective of this project is to develop a website that can host a live map dedicated to cycling navigation. This website will be the main deliverable and is intended to evolve continually over the upcoming years. By launching this platform, it will be possible to progressively implement features directly responsive to the needs and feedback of our target audience. Initially conceived as an app that merely provided map functionality, the scope was broadened to a webpage format, which was deemed more feasible and practical for broader and more flexible use.

The outcomes anticipated from this project are diverse and significant. Primarily, the website will provide a fully functional, user-friendly interface that supports a live map enabling users to efficiently plan and navigate cycling routes. This will not just improve individual cycling experiences but will also enhance safety by offering real-time data on traffic conditions, weather updates, and route suitability. Another fundamental outcome is to cultivate a thriving community of cyclists who engage with the platform. This community aspect is vital, serving as a feedback mechanism that will inform ongoing improvements to the platform. Through regular interaction with this community, it will be possible to understand better and adapt to the evolving needs, ensuring the platform remains relevant and valuable to its users.

Documenting the development process from its conceptualisation through to its visualisation is also a key goal. This documentation will serve multiple purposes: providing transparency and serving as a resource for stakeholders interested in the project's progress and methodologies.

Universal design is a core part of the project meaning that the website is accessible to all users, regardless of age or ability. The importance of this inclusivity cannot be overstated, as it ensures that the benefits of the platform can be universally enjoyed, which is crucial for growing a wider audience and regular users. The interface will be designed to be intuitive and straightforward, minimizing barriers to use through clear navigation, readable fonts, and user-friendly design aesthetics.

The website will leverage the latest in web development frameworks and technologies to ensure it is robust, responsive, and capable of handling real-time data efficiently. This includes using HTML5, CSS3, and JavaScript for the frontend to create a dynamic and interactive experience, while server-side technologies will manage API integrations for live updates. The integration of these technologies will not only provide a seamless user experience but also ensure that the platform is scalable and secure. In summary, this project aims to create a cutting-edge web platform that enhances how cyclists navigate and engage with their environment.

## **Report Outline**

The report will be split in the sections:

* **Research:** a deep dive into the research that took place after the conceptualisation of the idea. This includes a critical evaluation of the literature used as source material.
* **Requirements analysis:** Explanation of how requirements were derived from the literature review and initial research.
* **Design and Development Methodology:** Discussion of the chosen development methodology. Documentation and explanation of the system design and implementation process.
* **Risk analysis:** Exploration of different technology options considered for the project. Identification and analysis of potential risks associated with each technology option. Rationale for the chosen technology, possibly using decision-making tools like a Pugh Matrix or case studies.
* **Reflections:** Insightful reflection on the development process, focusing on successes and challenges. Discussion of key lessons learned and how they could be applied to future projects.
* **Conclusion:** Recap of the main points covered in the essay. Reflect on personal growth and further development based on the project’s outcomes.

# **Research**

## **Secondary Research**

### **Evaluation of Sources:**

The first step of research into this topic was to evaluate the competition. Analyse what they excel in and what they fall short in. In doing so, I aimed to mould this platform into something that can have its own unique skill set whilst also learning from past mistakes. A study discussing the strengths and weaknesses of the most popular navigation apps ([*Samet, C. Fruin, Nutalong, 2012*](#Samet)) including the IOS Maps and Google maps. By looking into this article, it became clear which pitfalls not to fall into. A requirement (*see* ***page 10***) that has been drawn from this keeping up the quality of the data on the maps. This includes updating the data and letting users know where the data is gathered from. Another functional requirement that is derived from this article is having a map that users can dynamically interact with. A non-functional requirement that I have gained from this research is having the map to be as responsive as possible. Testing should confirm that map data loads within two seconds when accessed from a standard broadband connection. Interactive map features like panning and zooming should exhibit no noticeable lag. Ultimately, the performance of the website alongside the map API will be tested during the testing phase development cycle.

The primary concern regarding this source is its publication date in 2012, which could potentially render it obsolete for contemporary application development. Technology, especially in the field of mobile applications and mapping services, evolves rapidly, and strategies or technologies discussed nearly a decade ago might not align with current capabilities of development. Despite this, the source provides valuable historical context and insights into the initial phases of development for these applications. It offers an alternative view, which includes the challenges during the early 2010s, which can be instrumental in understanding the evolution and incremental advancements in mapping technologies over time. This historical perspective is crucial for developers and researchers who wish to trace the trajectory of technological improvements and shifts in user experience design.

This led me to researching the specific mapping service known as Google Maps. The platform is widely regarded as the most preferred navigation application on both android and android platforms. Knowing this, I gained a drive to understand what issues the platform has so I can avoid them or implement preventative measures. An article ([*Asif, Abbas, Burney, Burney, 2018*](#Burney)) dives into the security concerns of Google Maps. More specifically, the risks of allowing anonymous access including malicious activity. The article discusses the possibility of users gaining sensitive information from the service without any checks. As you can imagine, this would be a public safety risk. With this knowledge, it is vital to include a functional requirement that entails the protection of sensitive information. There is a call for continuous monitoring of how mapping services like Google Maps are used, ensuring compliance with legal standards and enhancing the overall security to prevent misuse.

When integrating a geolocation system into a website, it is critical to adhere to the established standards and regulations concerning the gathering of location data. A recent study *(*[*Cabalquinto, Hutchins, 2019*](#Calbaquinto)*)* utilized a sample of 100 train commuters to give their perspectives on real-time geolocation mapping. The findings revealed significant reservations among participants, with a large majority expressing scepticism, opposition, or indifference towards the use of their location data. This suggests a discomfort among users regarding how their location information is handled on the internet.

The insight gained from this study is that while location services are becoming abundant in digital applications, user fear about privacy remains a significant issue. The fear is primarily due to concerns over how personal data, particularly location information, is used and shared by entities that collect such data. To address these concerns, it is important for developers and businesses to ensure that obtaining user consent becomes a standard practice. This means that users should have clear, straightforward options to either opt-in or opt-out of location tracking. The importance of having a consent-based system for location data collection cannot be overstated. It not only aligns with ethical business practices but also complies with legal standards like the General Data Protection Regulation (GDPR) in the European Union, which mandates explicit consent for the processing of personal location data. Furthermore, implementing a level of transparency enhances trust between users and service providers, which can lead to increased user engagement and satisfaction.

Another study ([*Souza, Oliviera, Junior, Sales, Cesar, 2015*](#Souza)) suggests that geolocation applications could potentially enhance citizen well-being by addressing the emotional impacts of commuting. A notable experiment by Yahoo Labs called "Happy Maps" is discussed. This experiment focuses on suggesting routes that are not only efficient but also enjoyable by considering factors like beauty, quietness, and overall happiness. These routes aim to provide a more positive commuting experience by leveraging emotional responses to the surroundings. This is something which I would love to add to this service down the line.

**Mapbox and Google Maps APIs:**

My interest in integrated map APIs grew as I delved into the differences between Google Maps API and Mapbox API, focusing particularly on the accuracy of their location data. A crucial study ([*Sapurta, Furqan, Abdidin, and Yunardi, 2019*](#Sapurta)) provided a comparison of these two popular APIs, revealing that Google Maps had a lower error rate of 9.250% compared to Mapbox's 12.128%. This result suggested a higher accuracy level in distance calculations for Google Maps, although the difference was not so large to render Mapbox unusable. The study concluded that for the specific requirements of their research, Google Maps suited them best.

From a usability perspective, Mapbox Directions API can be more user-friendly and intuitive. Both APIs were considered viable for this project, emphasising that the choice of API could depend heavily on the specific needs and context of the application being developed.

In a more recent analysis by ([*Ovramenko, 2023*](#Ovramenko)), the strengths and weaknesses of each platform were discussed in greater detail. Mapbox was praised for its customization options and flexible pricing. This makes it particularly appealing for projects such as this which don’t have much financial backing. Mapbox's superior handling of large datasets and its inclusion of a studio where data can be directly deployed into maps were noted as significant advantages. This level of customisation allows developers more control over the presentation and manipulation of geographic data. On the contrary, Google Maps is preferred for its larger user base, which increases community support for new users that are looking to implement the API. It also has a larger feature set than Mapbox which may be appealing for more people.

Another pivotal difference between the two platforms is their source of data. Unlike Google Maps, which uses their own data, Mapbox relies on OpenStreetMap. Which is a community-sourced platform. This has both advantages and disadvantages. While OpenStreetMap's open-source data promotes transparency and community involvement, it may also lead to inconsistencies and gaps in data coverage, especially in less populated areas. Mapbox offers more developer tools compared to Google Maps API, which can significantly ease the process of implementation during development. These tools are really helpful for developers who want to make their work easier and add detailed map features more smoothly. This was extremely appealing for this specific project as it made the whole development process more interactive.

The choice between Google Maps API and Mapbox API involves a trade-off between accuracy, user support, customisation capabilities, and the nature of the data source. Google Maps may be preferable for developers seeking reliability and extensive support, while Mapbox offers superior tools for customisation and handling complex data, making it ideal for projects where extremely detailed mapping capabilities are needed. Each project's specific requirements will ultimately guide the decision on which mapping API to integrate, highlighting the importance of a thorough evaluation based on the criteria most critical to the project’s success.

# **Requirements Analysis**

The steps surrounding the defining of the requirements (*see Figure 1*) is below:

A diagram of a project

Description automatically generated

***Figure 1:*** *Steps taken before starting the development of the project.*

## **Project Proposal**

### **Problem statement:**

“Many navigation web services do not fully portray the intricate designs of routes in the UK. Take Bristol for example, one of the most bike-friendly cities, has various alley ways and paths that do not show on widely used maps. Services such as google Maps, Apple maps and others show great routes but mainly utilize the main roads. A lot of the time there faster and safer routes for cyclists that can be used.”

### **Proposed solution:**

“I am looking to create a website that can give routes, directions, and current information for cyclists. As Bristol has a large cycling community (I myself being one), I would like a service that has the fastest and most efficient routes. A bonus feature that I will add if there is time is one where you can track and store your journeys similar to apps such as Strava and Nike Run. I will utilize An API such as google maps or Mapbox for the Map itself and Mapbox directions API to find the best and most efficient route service available. Other application that I will research will be Open sources such as GeoServer and PostgreSQL.”

### **Success criteria:**

* Fully functioning and live website.
* Accurate location services.
* User-friendly interface.
* Real-time updates.
* Search functionality.

### **Research resources:**

* Articles / Academic reports.
* Accurate location services.
* Surveys (if applicable).

## **Defining requirements**

This section will introduce the requirements and discuss the order of importance of the functional requirements. I will be implementing the **M**o**SC**o**W** Prioritisation to present the functional requirements. The format of MoSCoW is:

* **Must** have.
* **Should** have.
* **Could** have.
* **Won’t** have.

This allows us to easily understand the order of prioritisation of the requirements.

### **Functional requirements:**

The CycleNav website **Must** have:

* The ability to display a map.
* The ability to display the location of the user (*see* ***Figure 2***).
* Ask for consent of location. As per GDPR guidelines there must be an opt-in option for users to give their location data. (*see* ***Figure 2.1***).
* The ability to navigate from A – B (*see* ***Figure 2.2***).

The CycleNav website **Should** have:

* The ability to Mark a location. To find the latitude and longitude of a location.
* ability to be displayed on different devices.
* A screenshot of a computer

  Description automatically generatedA map of a city

  Description automatically generatedHave the ability to zoom. This feature is imperative for users with visual impairments. Not only that but the feature to zoom on a map is extremely important.

***Figure 2 :*** *User location can be shown.*

***Figure 2.1:*** *The website clearly asks for consent of location.*

A screenshot of a map

Description automatically generated

***Figure 2.2 :*** *The Mapbox navigation API can clearly and easily display not only the routes from A-B but also the turn-by-turn navigation.*

The CycleNav website **Could** have:

* Turn by turn navigation (*see* ***Figure 2.2***).
* Show traffic.
* Show nearby bike shops. This is a feature that is currently in development that is being tested with the Bristol area using Open Data Bristol. (*see* ***Figure 2.3***).

A map of a city

Description automatically generated

***Figure 2.3 :*** *The custom data added to the Mapbox map using Bristol open data can be seen in this figure. As this only spans the Bristol area the feature is not fully complete.*

The CycleNav website **Won’t** have:

* A log in feature. At a glance this seems like a good feature to have however it isn’t necessary for this website. This feature may be added in the future.
* Multiple routes.
* Forums/Discussion areas.
* Change map mode to satellite/terrain or use a Satnav mode.

### **Non-Functional requirements:**

* The map should load withing two seconds or under whilst network conditions are stable. This ensures a smooth experience for the user.
* The website will be suitable for users with disabilities allowing screen readers to operate or by having a zoom feature.
* All user location data must be encrypted in transit using TLS and at rest to protect against unauthorized access.
* The system should be capable of handling increases in user numbers and data requests without degradation in performance, ensuring that the service is dependable during peak usage times.
* The codebase should be well-documented and structured to facilitate easy updates and maintenance.
* The website interface should be intuitive and easy to use, minimizing the learning curve for inexperienced users. This includes clear icons and instructions for obtaining location consent.

These non-functional requirements ensure that the CycleNav website not only meets its functional goals but also provides a secure, accessible, and reliable user experience.

## **User Stories**

These are the user stories that I have extracted from the requirements.

User story: **Location Display.**

* **As a** user,
* **I want** the website to display my current location on the map,
* **So that** I can know my location and plan my route.

The website must request and obtain user consent to access the GPS data, a requirement under privacy laws like GDPR. Ensuring accurate location tracking across different devices and browsers can be challenging due to varying support for geolocation APIs. Additionally, the website must handle scenarios where location services are disabled or unavailable, providing a graceful fallback or prompting the user appropriately. Network reliability can also affect the accuracy and speed of location updates, which is critical for a seamless user experience.

User story: **Consent for location.**

* **As a** user,
* **I want** to be asked for my consent to use my location data,
* **So that** my data is handled in compliance with GDPR.

The website must effectively design and implement a user interface that clearly communicates the request for location data consent, ensuring it is easy to understand. Ultimately, trust with the user can only come from transparency with how we use their location data.

User story: **Route Navigation.**

* **As a** user,
* **I want** to input a starting point and a destination point,
* **So that** I can be given navigation options.

The primary concern is integrating reliable mapping services like Google Maps or Mapbox APIs that support detailed routing functionalities. Additionally, the interface must be intuitive, allowing users to easily specify locations and understand the route options presented.

User story: **Zoom Functionality.**

* **As a** user with visual impairments,
* **I want** the ability to zoom in and out of the map at will,
* **So that** I can see the map details clearly according to my needs.

The map must load efficiently at various zoom levels without losing performance, which requires careful management of data and resources. It's also essential to maintain map responsiveness across different devices and screen sizes, ensuring a consistent user experience.

User story: **Contact for support.**

* **As a** user,
* **I want** the ability to contact the support team,
* **So that** I can let them know if there is an issue with the map, my experience or feedback about the service.

Implementing a contact feature for user support on a website involves addressing both front-end and back-end technical challenges. On the front end, the design must be user-friendly, providing an intuitive contact form or other communication methods that are easy to access and use. This form should collect essential information without overwhelming the user, such as name, email, and what the issue is. On the back end, the system must manage and route these inquiries efficiently. This feature is critical but not yet fully developed. Completing it will require careful testing to ensure that it works seamlessly across all platforms and does not become a point of frustration for users seeking assistance. This is vital for maintaining high levels of user satisfaction and fostering a supportive user community.

# **Design and Methodology**

## **Methodology**

### **Agile**

The Agile methodology's flexible, iterative approach, which prioritises constant improvement makes it the perfect fit for the development of this project.

Agile encourages an iterative development process that uses sprints or cycles (*See* ***Figure 3***) to make small, incremental changes. This strategy is perfect for this project since it will allow me to immediately release a basic version of the platform and gradually improve it in response to real user input and interactions, emphasising the user centric nature of this service. Agile makes ensuring that every change can incorporate new features and improvements in line with the project's ongoing evolution throughout the years.

A screenshot of a computer

Description automatically generatedAtlassian Jira is a great option for those wishing to deploy Scrum and Kanban boards because it is specifically made to assist these Agile approaches. Users can easily schedule sprints, assign tasks to team members, and build user stories thanks to its intuitive design.  My experience with the product from previous projects is the reason I implemented it. For this project, it became simple to implement.

***Figure 3 :*** *The image shows the documentation of the sprint cycles by using Jira.*

A core objective of the project is to develop features that are directly responsive to user feedback. Agile methodology involves regular check-ins with users, which aligns well with the project's aim to engage with a community of cyclists. By incorporating feedback mechanisms into each phase of the development process, I can better understand and adapt to users' evolving needs, ensuring the platform remains valuable to its intended audience.

By breaking the project into manageable units, Agile allowed me to identify and address risks early in the development process (*see* ***Risks****, pg.24*). This proactive approach to risk management is crucial for a project involving real-time. Frequent testing (*see* ***Testing****, pg*.20) inherent in Agile practices helps in identifying potential technological and usability issues before they become significant problems.

As the initial project scope was changed from an app to a more comprehensive web platform, Agile’s flexibility is a significant advantage. It supports and encourages rapid and flexible responses to change, which is essential for a project expected to adapt its functionalities in response to user demands and technological advancements.

All of these reasons made resulted in Agile becoming the methodology for the project. adopting Agile methodologies will facilitate a more dynamic, user centric development process. It aligns with the project's long-term goals of adaptability, user engagement, and continual improvement, making it the ideal approach to handle the expected scale of the platform in the future.

## **Design**

### **Wireframes**

A screenshot of a computer

Description automatically generated**A screenshot of a computer

Description automatically generated**A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedWireframes serve as the blueprint for the cycling navigation website. Creating wireframes for this website is crucial because it allows me to understand the basic structure before any detailed design work or coding is initiated.

***Figure 4.2 :*** *The Wireframe of the “Download” page.*

***Figure 4.3 :*** *The Wireframe of the “Contact us” page.*

***Figure 4:*** *This is the wireframe of the main landing page.*

***Figure 4.1 :*** *This is the wireframe of the “About Us” page.*

### **Realisation of wireframes**

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generatedAfter creating the wireframes, I designed the website (*See figures below)* using Bootstrap Studio, a software that lets web developers customize sites with an extensive Bootstrap library. This tool simplifies designing responsive and visually appealing websites by offering a variety of customizable components. Adjusting this required significant html, CSS and JavaScript programming.

***Figure 4.6 :*** *The final design of the “Contact us” page.*

***Figure 4.5***  *Final design of the “Download” page.*

***Figure 4.4 :*** *Final design of the landing page.*

### **Map Design**

With Mapbox, I created a design of a map that not only looked unique but also incorporated layers of custom data. This customisation is facilitated through Mapbox Studio, where users can customise the aesthetics of the map by adjusting colours, fonts, and layers to their liking. It is not a free service however there is a free plan which I am currently on. As the website scales up there will be changes in the plan however for now it will suffice.

A map of a city

Description automatically generatedI created a custom map style and used Open Data Bristol to add nearby cycling shops (*see* ***figure 5***). This was done by uploading a GeoJSON file that contained a dataset.

***Figure 5 :*** *The map studio with the data from Open Data Bristol loaded on to the map.*

***Figure 5.1 :*** *As you can see the map locates the user location showing the closest stores.*

A screenshot of a map

Description automatically generated

Thanks to the Mapbox-gl-Directions API, There is an integrated routing system that allows users to locate the closest cycling distance (*See* ***Figure 5.2***). There are limitations to this API. Including a maximum of 10,000 kilometres per route. This is not an issue as the target demographic is cyclists who are cycling within the city. There is also a maximum of 300 request per minute. This is standard as it is nearly impossible to do this. All of these limitations would not hinder the use cases for this service.

For now, this API serves well as the base skeleton for the future of the service. However, I plan to add custom routes and changing the design depending on user feedback. This can be done in a multitude of ways such as creating a flutter application which uses any open-source map such as Mapbox or Leaflet. These solutions would come down the line when developing the mobile application however for now, it is fine to keep the map as an integrated part of the website. This reduces costs and implications down the line. As emphasised in a book ([*Horton, S, 2013*](#Horton)) on Web development, knowing your audience is extremely important. As I am a cyclist myself It further solidifies the relationship between me and my audience.

By taking the time to learn from sources ([*Krause, J, 2016*](#Kraus)) I understood how to design responsive webpages a lot easier and it gave me the confidence in laying the foundation that this integrated map could fit into.

A map with a blue line

Description automatically generated

***Figure 5.2 :*** *The map can navigate from A to B*

## **Testing**

### **Unit Testing**

I conducted unit tests for these specific components:

* Map loading / Data rendering.
* Consent of location.
* User location.
* User interaction e.g. Zoom functionality.
* Api calls e.g., Route generation.

Here are the test cases:

Test Case 1**: Initial Load**:

* **Objective**: Make sure that the map loads in a timely manner. With a default view set within the code.
* **Steps**: Simulate a map load and observe how fast the map appears.
* **Expected result**: The map loads in in a timely manner. And allows the user to navigate straight away with no issues.

Test Case 2**: Zoom**:

* **Objective**: Make sure that the ability to zoom works and all the buttons work fine.
* **Steps**: trigger the function by clicking on the zoom in and out features
* **Expected result**: The map loads in in a timely manner. And allows the user to navigate straight away with no issues.

Test Case 3**: Fetch routes**:

* **Objective**: ensure that the Navigation API works correctly.
* **Steps**: Simulate a route and analyse how accurate it is and how fast it generates it.
* **Expected result**: A route will be created for all the modes of navigation.

Test Case 4**: Get location**:

* **Objective**: Make sure that the location of the user can be shown.
* **Steps**: click on the location button.
* **Expected result**: The location of the user will be shown.

Test Case 4**: Consent**:

* **Objective**: The browser must ask for permission of the user before taking location data from them. And it must not collect data if the user refuses.
* **Steps**: Click the location button.
* **Expected result**: If the user allows the location data to be gathered it will track their live location. If not, the user will not be able to see their location and no data will be gathered.

Because the components are interactive and complicated, unit testing is an essential practice for developing this website. Throughout the project's lifecycle, this method helps maintain high-quality code standards and drastically lowers the chance of introducing bugs. Unit testing also makes it easier to find and fix problems quickly, which promotes more stable releases and dependable user experiences early in the development process.

## **Deployment & Version Control**

### **Netlify:**

When deploying the website, I chose to use Netlify (*see* ***Figure 5***) to host it. The benefits of using Netlify was the pricing, ease of use and security features. Netlify is an excellent platform choice for hosting and managing the cycling navigation website due to its powerful features that align well with the project's goals of continuous evolution, user engagement, and robust technological implementation.

Netlify provides a scalable and secure environment that supports agile, continuous development, making it an ideal choice for this project. Its feature set not only aligns with the technical needs of the project but also supports the strategic objectives of iterative improvement and user-centric development.

A blue and white text on a black background

Description automatically generated

***Figure 5 :*** *The figure shows the deployment of the website on Netlify.*

### **Git / GitHub:**

A screenshot of a computer program

Description automatically generatedFor version control I have chosen GitHub to store my code and update it. It serves as a great platform for documenting the entire development process. Each commit can be accompanied by detailed messages that explain why changes were made, providing a clear history of the project’s evolution. Also, GitHub’s issue tracker allows teams to document bugs and tasks directly within the repository, making it easy for anyone to follow the project's progress and understand decision-making processes.

***Figure 6 :*** *A snippet of the final update of the code*

# **Risk Analysis**

***Risk table :*** *Describes the risks and preventative measures installed.*

|  |  |  |  |
| --- | --- | --- | --- |
| **Risk** | **Impact** | **Likelihood** | **Preventative measures** |
| Technical difficulties with API | High | Medium | Choose reliable APIs and have a vetting process in place. |
| Issues with server handling user load | High | Low | Implement cloud hosting solutions and monitor traffic of the site regularly. |
| Data privacy and GDPR issues | Very High | Medium | Implement data protection measures and make sure to understand the data protection laws |
| Dependency of third-party services | High | High | Have contingency plans in place and research more service providers. |
| Map data being inaccurate | High | Medium | Regularly update the map data |
| Lack of skilled developers | Medium | Medium | Consider contracting outside sources |
| Budget constraints | Medium | High | Prioritise spending money on essential features of the website. Monitor spending. |

## **Considerations**

Given the dangers listed in the project for creating this website, choosing the right technology and approaches is essential in reducing these risks. Here is a breakdown of the technology possibilities that were planned, along with the reasons behind the approaches that were selected:

Mapbox API was the chosen platform due to its superior customization capabilities which aligns well with the project’s need for specific map features tailored to cyclists. It also provides documentation and developer support which reduces the risk of technical difficulties during implementation.

Because of its scalability, which reduces the danger of insufficient server capacity, I chose Netlify for hosting. The website can withstand greater traffic without experiencing a decline in performance thanks to Netlify's flexible scaling based on user demand. Its extensive worldwide distribution network also aids in speeding up loading times, solving user experience issues brought on by sluggish performance.

To make sure the interface is clear and easy to use, an iterative design process backed by frequent user testing sessions was selected. This method lowers the possibility of a poorly received user interface by allowing for ongoing improvement based on real user interactions.

The main objective is to implement a mobile application using the Mapbox navigation API through coding a Flutter Application using JavaScript. Other options like using Java to create it is also possible however there is a lack of skill I have as a developer so I would like to stick to Front end development.

## **Future of the project**

The ongoing evolution of this project will extend well beyond the initial submission date, ensuring continuous improvements and expansion. Creating a smartphone application that makes use of the website's APIs is a major component of this technique. Considering my current level of mobile programming knowledge, this shift to mobile platform development provides a new difficulty.

I'm thinking about making the project a community-driven initiative to address this. The project can gain from varied experience and cooperative problem-solving by involving other talented developers in the development process. This will speed up the process and improve the quality of the mobile application.

Keeping the source code updated on a public GitHub repository is also essential to this community-driven methodology. In addition to encouraging transparency, this welcomes contributions from developers all over the world who may assist enhance the code, add new features, and solve issues. The usage of the Mapbox API and other open-source data, which are by their very nature meant to foster community participation and creativity, lends further credence to the choice to keep the source code publicly available.

# **Reflection and Lessons Learned**

## **Successes**

One of the defining achievements of this project was the successful creation of a fully operational and accessible website featuring a live map tailored for cycling navigation. This functionality marked a significant stride towards realising the project’s core mission—enhancing the cycling experience through technology. The integration of real-time data played a critical role in amplifying both the utility and attractiveness of the platform to its user base. The real-time data integration is particularly commendable because it goes beyond traditional static mapping. By incorporating dynamic updates directly into the navigation experience, the platform offers a more accurate and responsive service.

Moreover, the way this dynamic data blends with the static user interface elements of the website deserves special mention. The design and development teams worked meticulously to ensure that the real-time data did not overwhelm or clutter the user interface. Instead, it complements the existing design by seamlessly integrating into the map view without disrupting the user experience. This thoughtful integration aids cyclists in making informed decisions swiftly without having to navigate away from the map or deal with intrusive updates.

## **Challenges**

One of the key challenges faced during the development was the initial lack of expertise in mobile development, which was a setback as it made me reconsider the direction of the project. However, the transition from mobile application to a web-based service served as a learning experience and developed my character. Overcoming this involved a steep learning curve and required seeking external help from more experienced people and sources online.

Another challenge was ensuring data privacy and security, particularly with the integration of location-based services. Complying with GDPR and other privacy laws required a meticulous approach to data handling and user consent management, which was both time-consuming and complex.

Ensuring universal design and accessibility posed its own set of challenges, especially in making the site intuitive and easy to navigate for all users, regardless of their technical abilities.

**Lessons Learned**

* One of the most important lessons learned was the value of active and ongoing engagement with the user community. Regular feedback is essential not just for identifying and fixing issues but also for innovating features that truly meet user needs.
* Starting with a minimal viable product (MVP) and then scaling based on user feedback and demand proved to be an effective strategy. This approach allowed for focusing resources on features that provided the most value while ensuring a solid foundation for adding more complex functionalities over time.
* The flexibility offered by Agile methodologies was extremely important in managing the requirements of the project. The ability to adapt quickly to changes and incorporate feedback iteratively helped in maintaining project momentum and alignment with objectives.
* Integrating privacy and security measures from the initial stages of development is non-negotiable, especially for applications dealing with sensitive data like location tracking. This not only helps in complying with legal standards but also builds trust with users.
* Utilising open-source technologies and maintaining an open-source project encouraged broader collaboration and innovation. This approach also helped in keeping development costs down while benefiting from community-driven improvements and features.

# **Conclusion**

The primary objective of this project is to develop a website that can host a live map dedicated to cycling navigation. This website will be the main deliverable and I intend to continually over the upcoming years. By launching this platform, it will be possible to progressively implement features directly responsive to the needs and feedback of our target audience. Initially conceived as an app that merely provided map functionality, the scope was broadened to a webpage format, which was deemed more affordable and practical for wider and more flexible use. The outcomes anticipated from this project are diverse and significant. Primarily, the website will provide a fully functional, user-friendly interface that supports a live map enabling users to efficiently plan and navigate cycling routes.

Furthermore, the ability to optimise routes also contributes significantly to the broader goal of promoting cycling as a sustainable mode of transportation. By providing a dependable and user-friendly navigation tool, the project supports the growth of cycling culture, encouraging more people to consider cycling as a viable alternative to motorized transport, thereby contributing to environmental conservation and healthier lifestyles.

Moving forward, these lessons will guide future development, particularly in the expansion into mobile applications and further globalisation of the platform. Emphasising these areas will ensure that the project remains sustainable and continues to grow with the evolving digital landscape.

Additionally, maintaining the project on GitHub and keeping it open source fostered a collaborative environment that will allow continuous improvements and community contributions. This approach will not only improve the quality of the software but will also accelerate the development process by leveraging community knowledge and skills.

# **Appendix**

## **Bibliography**

List of all sources cited in the essay, formatted according to UWE Harvard referencing style.

Burney, A., Asif, M., Abbas, Z. and Burney, S. (2018) Google Maps Security Concerns [online]. <https://www.scirp.org/journal/paperinformation?paperid=81614> [Accessed 15 March 2024].

Cabalquinto, E. and Hutchins, B. (2020) Investigating Smartphone Use and the Contending Attitudes of Commuters Towards Geolocation Data Collection [online]. <https://www.sciencedirect.com/science/article/abs/pii/S0736585320300629> [Accessed 15 March 2024].

Krause, J. (2016) Introducing web development. [Online]. Berkeley, CA: Apress. <https://learning-oreilly-com.ezproxy.uwe.ac.uk/library/view/introducing-web-development/9781484224991/>

[Accessed 03 March 2024].

Horton, S, Quesenbery, W (2013) A Web For Everyone [online]. <https://books.google.co.uk/books?id=Dno3DwAAQBAJ&printsec=copyright&redir_esc=y#v=onepage&q&f=false> [Accessed 15 March 2024].

Ovramenko, V, (2023) Mapbox Vs. Google Maps Pricing: What You Need to Know [online]. <https://mappitall.com/blog/mapbox-pricing#:~:text=Here%20are%20some%20of%20the,more%20customization%20options%20than%20Google> [Accessed 20 March 2024].

Samet, H., Fruin, B.C.F. and Nutalong, S.N. (2012) Duking It Out at the Smartphone Mobile App Mapping Api Corral: Apple, Google, and the Competition [online]. <https://dl.acm.org/doi/abs/10.1145/2442810.2442818> [Accessed 12 March 2024].

Sapurta, K, Furqan, M., Abidin, T.F. and Yunardi, D.H (2019) Google Maps and Mapbox Api Performance Analysis on Android-based Lecture Attendance Application [online]. <https://www.researchgate.net/publication/337372073_GOOGLE_MAPS_AND_MAPBOX_API_PERFORMANCE_ANALYSIS_ON_ANDROID-BASED_LECTURE_ATTENDANCE_APPLICATION> [Accessed 13 March 2024].

Souza, R., Oliviera, I.P., Junior, F., Sales, L. and Cesar, F.F. (2015) Beyond Efficiency: How to Use Geolocation Applications to Improve Citizens Well-being [online]. <https://personales.upv.es/thinkmind/dl/conferences/smart/smart_2015/smart_2015_3_10_40079.pdf> [Accessed 02 March 2024].

**GitHub Repository:** <https://github.com/Gulzey/Final-Year-Project>