

**Czech University of Life Sciences Prague**

**Faculty of Economics and Management**

**Department of Information Engineering**



**Development of a map application  
for public transportation**

**Hiromu Kato**

**© 2024 CZU Prague**

# CZECH UNIVERSITY OF LIFE SCIENCES PRAGUE

Faculty of Economics and Management

## BACHELOR THESIS ASSIGNMENT

Hiromu Kato

Informatics

Thesis title

**Development of a map application for public transportation**

---

### Objectives of thesis

The main objectives of the thesis are to analyze the existing map applications such as Google Maps, Apple Map etc. and design and implement an application which is more suitable for getting public transportation information using JavaScript and Laravel framework.

### Methodology

The thesis will consist of two parts – theoretical overview and practical part. The theoretical part of the thesis will be based on a literature review and will provide fundamentals for the practical part. It will also include analysis of existing applications, focusing on their advantages and disadvantages.

The practical part of the thesis will include design and implementation of a software system that will be built using the theoretical foundations and findings from the analysis of existing systems. The implementation of the new system will be done using JavaScript and Laravel framework.

The process of the design and implementation will be described and gained experience will be summarized. Standard software engineering tools and methods will be used in the process.

**The proposed extent of the thesis**

35-40 pages

**Keywords**

Laravel, Javascript, API, public transportation, map, system

---

**Recommended information sources**

Apple. "Apple Developer Documentation." Apple Developer, Apple, 2021,  
<https://developer.apple.com/documentation/>.

Google. "Maps JavaScript API." Google Developers, Google, 2021,  
<https://developers.google.com/maps/documentation/javascript/overview>.

JavaScript reference – MDN Web Docs. Mozilla Foundation, 2023,  
<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>

Laravel Framework Documentation. Laravel Holdings, 2023, <https://laravel.com/docs/11.x>

---

**Expected date of thesis defence**

2023/24 SS – PEF

**The Bachelor Thesis Supervisor**

Ing. Jiří Brožek, Ph.D.

**Supervising department**

Department of Information Engineering

Electronic approval: 28. 11. 2023

**Ing. Martin Pelikán, Ph.D.**

Head of department

Electronic approval: 9. 2. 2024

**doc. Ing. Tomáš Šubrt, Ph.D.**

Dean

Prague on 13. 03. 2024

## **Declaration**

I declare that I have worked on my bachelor thesis titled  
" Development of a map application  
for public transportation"

by myself and I have used only the sources mentioned at the end of the thesis. As the author of the bachelor thesis, I declare that the thesis does not break any copyrights.

In Prague on 14.03.2024

---

## **Acknowledgement**

There are many people I want to acknowledge who got me to this point in my life.

I am grateful to my supervisor Ing. Jiří Brožek, Ph.D for his understandable guidance and valuable suggestions.

There are some friends who support each other.

Finally, I am truly thankful to my family and friends in Japan for their endless support.

I had never thought I graduated from a foreign university.

Thank you for supporting me all these years.

# Development of a map application for public transportation

## **Abstract**

These days, most people own smartphones and use map navigation applications.

Additionally, map navigation applications with various features and applications that apply these technologies are in widespread use.

However, since there is no navigation application that I can use satisfactorily in Prague, I decided to make the creation of a map navigation application the theme of my bachelor's thesis and will try to develop a map navigation application that could solve the problem.

These problems are time lost due to traffic jams when riding the bus in Prague and route selection problem based on walking speed.

The purpose of this research is to understand the basic structure and technology by analyzing and comparing existing famous applications such as Google Map and Apple Map and PID Lítačka in literature review.

In the Practical part, we will utilize that knowledge to develop a better map navigation application that focuses on public transportation.

For development, I use common frameworks and programming languages.

In conclusion, this study of map navigation applications related to public transport provides an overview of complex technologies and contributes to the creation of new applications regarding them.

**Keywords:** Navigation, public transportation, Map, User location, GPS, Dijkstra's algorithm, API, Maps Javascript API, Google Map API, Laravel, JavaScript

# Vývoj mapové aplikace pro veřejnou dopravu

## Abstrakt

V dnešní době většina lidí vlastní chytré telefony a používá mapové navigační aplikace. Kromě toho se hojně používají mapové navigační aplikace s různými funkcemi a aplikace, které tyto technologie využívají.

Protože však v Praze neexistuje žádná navigační aplikace, kterou bych mohl uspokojivě používat, rozhodl jsem se, že se tvorba mapové navigační aplikace stane tématem mé bakalářské práce a pokusím se vytvořit mapovou navigační aplikaci, která by tento problém vyřešila.

Těmito problémy jsou časové ztráty způsobené dopravními zácpami při jízdě autobusem v Praze a problém výběru trasy na základě rychlosti chůze.

Cílem této práce je pochopit základní strukturu a technologii pomocí analýzy a porovnání stávajících známých aplikací, jako jsou Google Map a Apple Map a PID Lítačka v přehledu literatury.

V praktické části pak tyto poznatky využijeme k vývoji mapové navigační aplikace, která se skutečně zaměřuje na lepší veřejnou dopravu.

Pro vývoj použiji běžné frameworky a programovací jazyky.

Závěrem lze říci, že tato studie mapových navigačních aplikací týkajících se veřejné dopravy poskytuje přehled o komplexních technologiích a přispívá k tvorbě nových aplikací týkajících se těchto technologií.

**Klíčová slova:** Navigace, veřejná doprava, Mapa, Poloha uživatele, GPS, Dijkstrův algoritmus, API, Mapy Javascript API, Google Map API, Laravel, JavaScript

# Table of content

<b>1</b>	<b>Introduction .....</b>	<b>10</b>
<b>2</b>	<b>Objectives and Methodology .....</b>	<b>12</b>
2.1	Objectives .....	12
2.2	Methodology .....	12
<b>3</b>	<b>Literature Review .....</b>	<b>13</b>
3.1	Navigation .....	13
3.1.1	Navigation History .....	13
3.1.2	Personal Navigation System .....	14
3.2	Existing three main application in Prague (Google Map, Apple Map, PID Lítačka Application) .....	15
3.3	Three main components of Personal Navigation System.....	17
3.3.1	Map .....	17
3.3.1.1	Google Map .....	18
3.3.1.2	Apple Map .....	19
3.3.1.3	The litachka map.....	19
3.3.2	User's location .....	21
3.3.2.1	GPS .....	21
3.3.2.2	GPS structure .....	22
3.3.2.3	Fundamental finding location concept of GPS .....	24
3.3.3	API .....	25
3.3.4	Directions of some kind .....	27
3.3.4.1	Data structure of Dijkstra algorithm .....	28
3.3.4.2	Dijkstra algorithm .....	29
3.3.4.3	Advanced Dijkstra algorithm.....	30
<b>4</b>	<b>Practical review .....</b>	<b>32</b>
4.1	Review of the software development kit .....	33
4.2	Map.....	33
4.3	User's location.....	35
4.4	Directions of some kind .....	37
4.5	The problem solutions .....	42
<b>5</b>	<b>Result and Discussion.....</b>	<b>45</b>
5.1	Comparison .....	45



5.2	Recommendation for future development.....	46
<b>6</b>	<b>Conclusion.....</b>	<b>48</b>
<b>7</b>	<b>References .....</b>	<b>49</b>
<b>8</b>	<b>List of pictures, tables, graphs and abbreviations .....</b>	<b>51</b>
8.1	List of figures .....	51
8.2	List of graphs.....	51
	<b>Appendix.....</b>	<b>52</b>

# 1 Introduction

As I am living in an unfamiliar place in Prague as an international student, I started using navigation apps more frequently. In addition, the reasons, such as public transportation is widespread in Prague and I don't have a own car are also related to the use frequency.

At that time, I noticed a few problems with finding routes using public transportation in google map or apple map.

For example, traffic jams occur in Prague in the evening on weekdays. If you take the bus to go home from Czech University Of Life Sciences Prague around 16:00 or 17:00 in the evening on weekdays, there will be severe traffic jams near Dejvická, and you will be dropped off one bus stop before reaching the bus stop in Dejvická. In that case, if you use the tram that uses the same road halfway, you can get to your destination, Dejvická, without getting stuck in traffic jams. Also, they don't come on time at the time when I go to my university.

In addition, existing navigation apps typically use various algorithms to suggest the fastest route and then give us the first stop, such as bus stops or train stations, In addition, existing navigation apps typically use various algorithms to suggest the fastest route.

They are developed by world-famous big tech companies such as Google and Apple, and It seems that they may not a be a problem.

However, in some cases, they may be wrong for the first stop they give us to get public transportation and you may be able to reach the first stop faster by walking a little further or can choose faster route by getting another correct stop. The reason is that the walking speed for the distance to the set stop is assumed to be ambiguous for everyone from children to the elderly. In fact, it is possible to arrive much earlier than the estimated time, and in case of an emergency, it is also possible to save time by running. This will reduce the time it takes to get home and help us during Prague's cold winters, which drop to -10 degrees Celsius.

Moreover, in recent years, public transportation use and road congestion have been reduced as part of the solution to environmental pollution problems. This allows for smoother traffic flow, reducing commute time and stress. It may be cheaper to use public transportation than to own a personal vehicle. There are no fuel costs, parking fees, vehicle maintenance costs, etc. Given this social background, as a student majoring in informatics, I decided to create a

navigation app that focuses on public transportation using Information technology.

In Chapter 3, the basic components for creating navigation apps are divided into map, current location, and several directions. in the map section, I compared the differences between existing apps, the current location section explains how GPS works and the API that makes it possible to use GPS, and some Direction of some kind sections explain typical algorithms used in navigation apps and deciding the direction.

The first part of the practical part started with the Review of the Software Development Kit, and as in Chapter 3, we developed and combined the sections for the map, current location, and several directions to create a unique navigation app.

Chapter 5 of this thesis is Results and discussion, where we discover the shortcomings and differences in the existing app under Limits of Application, and provide recommendations to improve this app under Recommendation for future developers. The final chapter of the paper presents the conclusion of the study and highlights the main points.

## **2 Objectives and Methodology**

### **2.1 Objectives**

The main objectives of the thesis are to analyze the existing map applications such as Google Maps, Apple Map etc. and design and implement an application which is more suitable for getting public transportation information using Laravel.

In addition, the application would solve the challenges bellow.

- Time lost due to traffic jams when riding a bus in Prague
- Route options by setting walking speed

### **2.2 Methodology**

The thesis will consist of two parts -- theoretical overview and practical part. The theoretical part of the thesis will be based on a literature review and will provide fundamentals for the practical part. It will also include analysis of existing applications, focusing on their advantages and disadvantages.

The practical part of the thesis will include design and implementation of a software system that will be built using the theoretical foundations and findings from the analysis of existing systems. The implementation of the new system will be done using Laravel.

The process of the design and implementation will be described and gained experience will be summarized. Standard software engineering tools and methods will be used in the process.

## **3 Literature Review**

### **3.1 Navigation**

Navigation is the process of determining and checking a route to get to a destination.

It consists of various tools, data, algorithms, and technologies to get information and so on. It is used in many contexts, such as maritime navigation for ships and submarines, aviation navigation for aircraft, and land-based navigation for automobiles, bikes, bicycles, and pedestrians. Moreover, it has obvious differences in developing technologies.

#### **3.1.1 Navigation History**

In the beginning, for example, Celestial navigation and geographical navigation have been used since ancient times. In detail, Celestial navigation is also called astronavigation and uses the sun, the moon, and stars, which is called the Celestial body. Additionally, geographical navigation is also one of the rudimentary methods of navigation, as well as Celestial navigation. It measures a position using landmarks such as lighthouses, mountains, capes, islands, and so on. Given these examples, geographical navigation mostly relied on the natural features of the earth's surface and lacked precision. In addition, these methods required a deep understanding of astronomy and careful observation, making it a skill mastered by seasoned navigators. However, due to science and technology developed, the accuracy was enhanced by coastal maps and charts, celestial calendars and sextants for determining the location of celestial bodies. Moreover, magnetic compasses and chronometers had contributed to important roles then. The used compass record was found in China and it covered they had used it during wars and then foggy. (Tsui, James Bao-Yen., 2005)

In the modern era, Radio wave facilities and satellite positioning systems which are called GPS offer straightforward yet precise location information, and conventional navigation technology, which is explained above and complex and requires advanced skill tends to be obsolete except for redundancy for safety purposes or specific and emergency circumstances.

### 3.1.2 Personal Navigation System

We will see the modern-era navigation systems in more detail.

Smartphones, being highly portable, have popularized positioning applications, with the Ericsson mobile report from June 2018 indicating that 4.8 billion smartphones worldwide are equipped with GNSS chipsets. (KARKI & WON, 2020)

Consequently, smartphones have emerged as the predominant GNSS location terminal, finding extensive utility across diverse mobile services.

GNSS (global navigation satellite system) is a general term for satellite positioning systems and a system that receives signals with time information from multiple positioning satellites and measures the current position on the ground. GPS (global positioning system) is a system developed by the United States and is a type of GNSS.

In this thesis, I will use GPS examples to explain the method that finds a user's location.

Hence, we can easily get a personal navigation system as a usual individual stuff that we use daily and see the revolution of PNA which stands for personal navigation assistants.

The resource explaining PNA explanation starts with the following sentence.

There are three different types of personal navigation assistants(PNAS). (Christopher E. White, 2000)

Moreover, it explains and shows three generations of PNA.

First-generation PNAs are the simplest one. They have only a map and the function to search the map. For instance, we can utilize the address searching function, landmark searches, and so on. In addition to First-generation PNAs, Second-generation PNAs provide a map and the user's current location.

Unlike first-generation PNAs, second-generation PNAs have the location where the user is. Finally, third-generation PNA possesses Map, User's location, and Directions of some kind. In this thesis, I will analyze map applications and develop an application that is more suitable for getting public transportation information, so I divided map applications into the aspects of a map, the user's location, and directions of some kind based on third-generation PNA based on third-generation PNAs.

Hence, I analyze these aspects of the applications that were already mentioned before with three practical applications which are Google map, Apple Map, and PID Lítačka.

## **3.2 Existing three main application in Prague**

### **( Google Map, Apple Map, PID Lítačka Application)**

In selecting Google Maps, Apple Maps, and PID Lítačka as comparators for public transportation navigation applications, we took into account the wide disparity in OS usage among Prague residents in 2023. 70.14% prefer Android and 29.29% prefer iOS. (Christopher E. White, 2000)

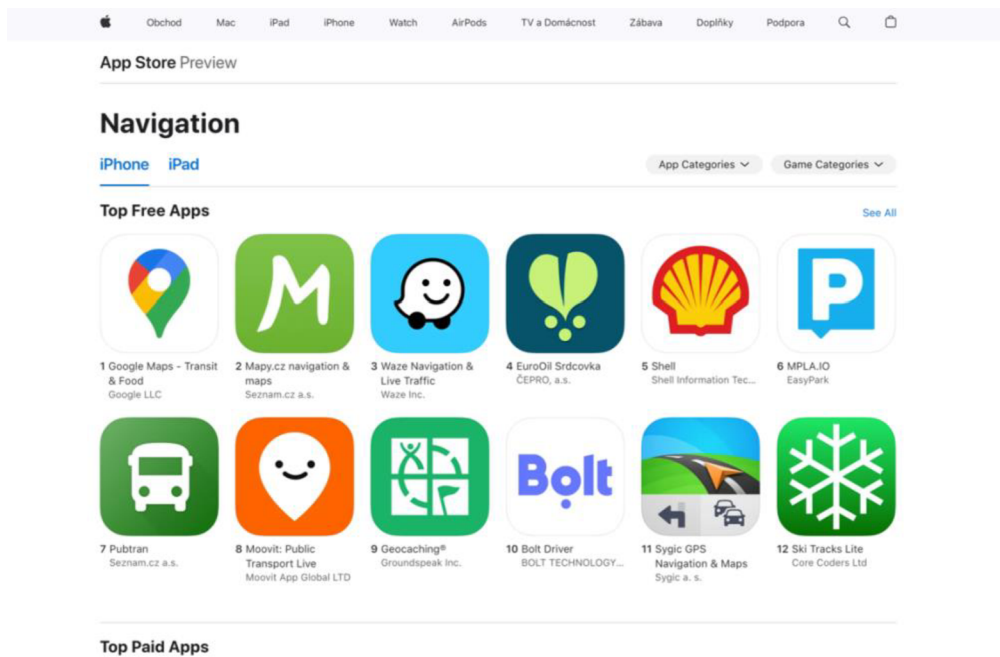
It means that 29.29% of the people who answered the survey have Apple Map because the application automatically downloaded. In addition, it is essential to evaluate applications that are compatible with both major platforms because it's not only a side answer.

Google Maps has emerged as the most important choice due to its cross-platform compatibility.

It integrates seamlessly into both Android and iOS devices, guaranteeing accessibility for the majority of Prague's population. Its dominance in the field of public transportation navigation is driven by unparalleled global coverage, real-time traffic updates, and an extensive database of points of interest.

Given the widespread adoption of Android, the universal appeal of Google Maps has made it the cornerstone of navigation for the majority of Prague residents.

As a result, in the Czech Republic Apple store and navigation ranking, it first rank on 3rd March.



**Figure 1 The Google Map Czech rank in Apple store, Source: Author himself**

Conversely, for the 29.29% of Prague's iOS users, Apple Maps holds a distinct advantage. its seamless integration with the iOS ecosystem provides a tailored experience for Apple device owners. Features such as Flyo and detailed 3D maps made by Apple Maps offer a visually captivating navigation experience. In addition, Apple's strict privacy policy resonates with a demographic that is increasingly concerned about data security. iOS users find Apple Maps to be a compelling choice that stands out and seamlessly matches their device.

While Google Maps and Apple Maps are used by a wide range of users, PID Lítačka addresses an important aspect of navigation specific to Prague. Serving as the primary electronic card for accessing public transportation, the PID Lítačka is essential to Prague residents' daily commute. Its relevance is underscored by the fact that public transportation is a lifeline for many citizens. The integration of smartphone applications will further streamline the process for both Android and iOS users by offering greater convenience in managing tickets and checking balances.

In summary, the selection of Google Maps, Apple Maps, and PID Lítačka as comparators was strategic based on the distribution of iOS and Android users in Prague in 2023. The cross-platform compatibility of Google Maps is a major advantage for the majority of the



population favors Android and positioned it as a universal choice; Apple Maps specifically caters to iOS users and provides a seamless and visually appealing navigation experience; PID Lítačka addresses key areas of public transportation and meets the essential needs of Prague's urban residents. Each application plays a role in its strengths and comprehensively covers the navigations of Prague's diverse user base.

### 3.3 Three main components of Personal Navigation System

#### 3.3.1 Map

The first factor that makes up the third generation PNAs is a map. We can call it map data, when it comes to software applications. Modern maps are made using different map layers stored in Geographic Information Systems (GIS). Each layer contains its own set of location-based data points. For example, Parcels, topography, demography, and imagery.

They are stacked one above the other to create a map.

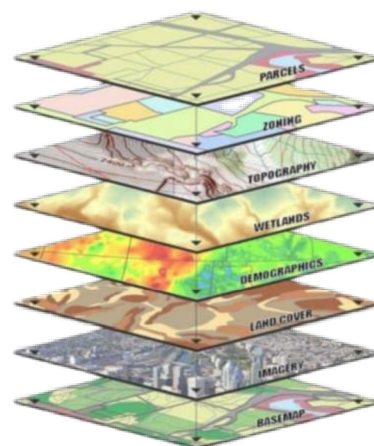


Figure 2 data layer (Werner, 2022)

In addition, This world is so huge and constantly changing. New roads, buildings, restaurants, and so on appear. On the other hand, they disappear.

We would say the quantity of data is most important because we can't get to a destination if the destination is not recognized as data on a map.

So, according to the point, I compared them based on the way to get the data.

### 3.3.1.1 Google Map

Google is gathering the map data using the combinations of methods. The first thing is third-party data. They have gathered more than 1,000 third parties from all over the world.

For example, the United States Geological Survey (USGS) and the National Institute of Statistics and Geography (INEGI) in Mexico give the data to Google and so on. (Andrew Lookingbill, 2019)

These organizations have been recognized and esteemed by this society and then It's really accurate and way too much data though, google carefully checks the data to ensure the data is accurate or not by themselves.

In addition, they are so huge and take a perspective widely thought, others, which as a local municipality, an NGO, or a housing developer, that focuses on small regions, also cooperate with Google to provide their specific data. (Andrew Lookingbill, 2019)

Moreover, google set up a new service to upload up-to-date for them, for example, new roads, the situation of construction, and disaster damage by many organizations. (Andrew Lookingbill, 2019)

It makes the map more up-to-date and gives it much information. However, it is not only from these organizations but also users. Google allows us to edit and add new map information. Furthermore, they publish them after the review of this information.

It takes a lot of time and effort though, Google Map would become reliable and we can use it with such confidence. In order to process the checks, they use machine learning and it allows them to finish them faster. Physically, they also use cars that are called "Google Street View cars" and equipped with multiple cameras. There are chances we see them on the streets and some of us already saw them. The method gives us street views, known as street-level imagery. It really shows details. My parent house is located in residential area.

However, the cars allow us to see the house in Google Map.

When we take public transportation into account, it doesn't affect them deeply though, It helps users to get to destinations easily.

### 3.3.1.2 Apple Map

Apple also gathers map data in the almost same ways. For example, third-party data from partners like Foursquare, Yelp, and TripAdvisor, to provide users with other business information, ratings and reviews. In addition, there is “Apple Business Connect”, which allows us to put our businesses on Apple map. (Perez, 2023)

Furthermore, Apple is conducting ground surveys with vehicles around the world to collect data to improve Apple Maps and to support the Look Around feature. Vehicle surveys may use equipment mounted on the outside of vehicles as well as iPhones, iPads, or other devices inside of vehicles. We may periodically revisit and re-collect data in some of those locations in an effort to maintain a high-quality, up-to-date map.

At this point, as I mentioned above, Apple and Google are gathering these information in the same way.

However, I found a source that proves they gather pedestrian data. Apple is conducting surveys with portable systems in select pedestrian areas (for example, certain streets in San Francisco that cannot be driven). Some pedestrian surveys use a backpack system to collect data that may be used directly in Apple Maps, such as in the Look Around feature.

Other pedestrian surveys use iPads, iPhones, or other devices to collect data for map improvement purposes. (Apple, n.d.)

These pedestrian surveys allow us to improve and update Apple Maps in areas where vehicles simply can't go while using the same privacy protections as the Apple Maps vehicles.

### 3.3.1.3 The litachka map

This map doesn't have an aerial photograph and they use Apple Map as we can see the logo of the Apple map on the left lower. However, they offer three functions. Firstly, they show where each stop is, such as trams, metros, buses, and so on. Secondly, Sale spots like ticket machines, information centers and these opening times are shown on their map. Finally, it shows parking spots on the streets and these prices and maximum parking times. The Sale spots and parking spots are not on Google Map and Apple Map.

In terms of the Czech mobile application, it is really a local application and useful to the people who usually use cars.

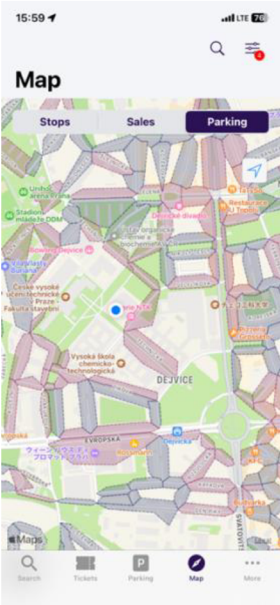


Figure 3 the Praking spots of PID Lítačka UI, Source: Author himself

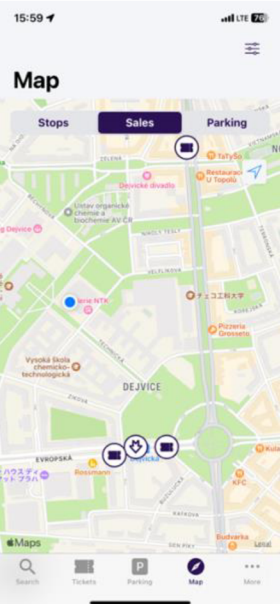


Figure 4 the sales spots of PID Lítačka UI, Source: Author himself

### 3.3.2 User's location

The second factor is User's location. This shows where the user is and can be found in many ways. In general, GPS is the most popular method. However, there are some other ways. For example, dead reckoning (DR) and ground-based beacon.

Dead reckoning (DR) is a method of navigation that estimates a current position. A previously determined position was used with the object's speed and course. By calculating how far they move and what course they used over time, The method provides us with the user's location. It is a navigation technique that is primarily used in environments where traditional methods, which GPS may be unavailable or unreliable, such as in submarines, airplanes, or remote areas. In addition, ground-based beacon uses Bluetooth to communicate with each other. Bluetooth can be used for only a short distance, which is said that the maximum is around 100m. It is mainly used indoors. Hence, we would need a lot of beacons to get a user's location and this cannot be used in the environment such as submarines and deserts. These above methods are not suitable for public transportation and most applications that needs current location use GPS and I will use GPS to get user's locations in this thesis and, in this chapter, I will explain GPS.

#### 3.3.2.1 GPS

GPS, which stands for Global Positioning System, is a navigation technology that has become a huge part of modern life, used in a myriad of applications for military, surveying agriculture, and emergency response. In the details, GPS is a satellite-based navigation system designed to provide the accurate current location of GPS receivers anywhere on Earth, although the accuracy and reliability of GPS depend on several factors, including the number of satellites in view, their geometric distribution, signal obstructions, and atmospheric conditions. However, this thesis shows the accuracy of GPS under the forest. He conducted the experiment to assess the positioning accuracy of three dual-frequency smartphones in forested environments.

Three devices are OPPO K9 (hereinafter referred to as K9), Realme GT NEO (NEO) ,and HUAWEI P50 Pro (P50).The first experiment involved the use of three smartphones midst a forest and the results showed the positioning accuracy of 6.13 meters with leaves, while 4.10 meters without leaves. (al., 2022)

In other words, the accuracy is not definitely correct and we have to consider it to create the application.

### 3.3.2.2 GPS structure

Understanding the structure and function of GPS is essential to better understand it. Next, we will elucidate the operating mechanism of GPS, focusing on its main components such as GPS satellites, Control segments, and GPS receivers.

#### GPS satellite

The GPS satellite constellation refers to the group of at least 24 satellites and is manipulated by The United States Air Force to cover worldwide for 24 hours and Each satellite circles the earth twice a day. (A, 2000)

As of July 3, 2023, there were a total of 31 operational satellites in the GPS constellation, not including the decommissioned, on-orbit spares. (Official U.S. government information about the Global Positioning System (GPS) and related topics, 2022)

They transmit radio signals to the earth station and these signals have periodically updated data from the control segment using 1575.42 MHz which is called L1, 1.2276 GHz refers to L2 or a second SPS ranging signal. (SPS stands for Standard Positioning Service)

There are many types of data GPS satellites send. For example, satellite time of transmission, satellite position, satellite health, satellite clock correction, propagation delay effects, time transfer to UTC, and constellation status. Moreover, they send a 1.023MHz Pseudo Random Noise (PRN) Coarse/Acquisition (C/A) code sequence first to accurately uniquely identify each of the satellites, decode satellite signals, synchronize timing, and determine position. (Tsui, James Bao-Yen., 2005)

#### Control segment

The GPS control segment encompasses a global network of ground facilities responsible for monitoring the GPS satellites in the constellation, managing their communication, conducting analysis, and issuing directives and commands to facilitate navigation services. The Master Control Station (MCS) is located at Schriever AFB in Colorado, North America,

with an additional master station in California. The infrastructure includes a total of 4 ground antennas, 6 Air Force monitor stations, and 10 NGA monitor stations. (Mulla, Baviskar, Baviskar, & Bhovad, 2015)

## GPS receiver

GPS receivers have many types such as smartphones, car navigation systems, electronic smart watches for fitness or sports, and so on. These days, even professional football players use GPS to monitor their performances and workload and prevent injury and there are many types of GPS receivers though, they are composed of the software part and the hardware part.

The hardware part mainly consists of Antenna, RF chain, and ADC. Antenna is used to receive transmitted signals directly from the GPS satellites. After that, the RF chain lets the input amplified to the proper frequency which is suitable for ADC. ADC, which stands for an analog-to-digital converter, literally digitizes the result into digital information. (Tsui, James Bao-Yen., 2005) Decoding algorithms of the information GPS receiver got from satellite follows a structured process starting from Acquisition in software. The acquisition method is to find some beginnings of C/A codes and use this information to change the wide-band signal to a continuous wave signal and its carrier frequency can be found.

Generally, it takes less than one second to process the acquisition program with 1ms of digitized data to find a signal a 400 MHz with Pentium computer.

Then, these C/A codes are differentiated by a difference in starting time and Doppler frequency. Once the Acquisition is done, the information is sent and the tracking phase begins. (Tsui, James Bao-Yen., 2005)

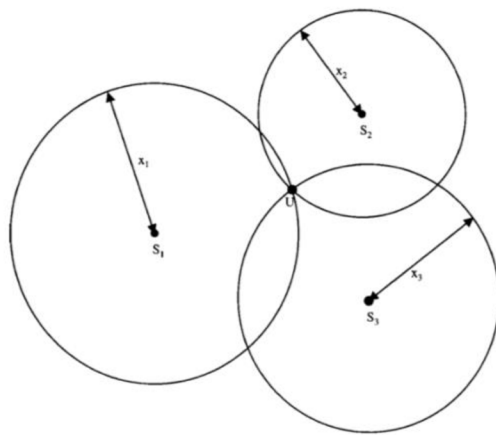
However, these following steps are way too complicated and related to physics like frequency. In addition, in fact, there are some acquisition ways and requirements and conditions to use one of them.

Our objective in this paper is not to develop GPS software that decode and proceed the information GPS satellites sent. Rather, I don't focus on these acquisitions and requirements and will describe a brief GPS structure.

Thus, here, I will explain the fundamental GPS concept to find the location after the software gets information through the tracking phase.

### 3.3.2.3 Fundamental finding location concept of GPS

The user's location can be determined by distances that are measured from the user's location to some satellite positions we already know. I will briefly explain it using an example below. When we know three distances between the user's location to the points, we can write three circles. Then, one position which is the interaction of three circles is uniquely shown in the below illustration.



**Graph 1 User location and three circles in two dimension (Tsui, James Bao-Yen., 2005)**

However, this is just a two-dimensional user position and our world has length, width, and height, so it's three dimensions. Because of that, we need one more known position and it is supposed to be the center of the earth. And then, The range indicated by that distance becomes a sphere and the satellite positions can be shown  $(x_1, y_1, z_1)$ ,  $(x_2, y_2, z_2)$ , and  $(x_3, y_3, z_3)$ .

What is more, we define each three distances and the user's location respectively as  $D_1$ ,  $D_2$ , and  $D_3$  and  $(X_u, Y_u, Z_u)$ .

As to the distances, we can get the information as ephemeris data sent by the satellites and there is the formula to get a distance. Every satellite sends a signal at a certain time  $T_{si}$ . The receiver will receive the signal at a later time  $T_{ui}$ . Then,  $T_{ui}$  minus  $T_{si}$  means the signal traveling time from the satellite to the user.

Because speed is already known, here speed is  $c$ , we just use the following formula to get the distances.



$$\text{Distance} = \text{Speed} \times \text{Time.}$$

However, each satellite has an atomic clock though, the usual GPS receiver doesn't have it and it makes time offset. In order to solve the problem, one more satellite is needed and we require at least five satellites. Thus, we can apply them to the equation of a Point and a Sphere in 3D Space and then make the three equations of them. This is just a simple example and we must consider other facts like Einstein's theory of relativity due to gravity and altitude though, as the simple example, we already know  $(x_1, y_1, z_1)$ ,  $(x_2, y_2, z_2)$ ,  $(x_3, y_3, z_3)$ , and  $D_1, D_2$ , and  $D_3$ .

When we calculate them, we can get the value of  $(X_u, Y_u, Z_u)$  and it can be matched to latitude, longitude, and altitude.

Finally, we can find a user's location. (Tsui, James Bao-Yen., 2005)

$$\begin{aligned} D_1 &= C * (Tu_1 - Ts_1) & D_2 &= C * (Tu_2 - Ts_2) \\ D_3 &= C * (Tu_3 - Ts_3) \\ D_1 &= \sqrt{(x_1 - X_u)^2 + (y_1 - Y_u)^2 + (z_1 - Z_u)^2} \\ D_2 &= \sqrt{(x_2 - X_u)^2 + (y_2 - Y_u)^2 + (z_2 - Z_u)^2} \\ D_3 &= \sqrt{(x_3 - X_u)^2 + (y_3 - Y_u)^2 + (z_3 - Z_u)^2} \end{aligned}$$

**Figure 5 formulas (Tsui, James Bao-Yen., 2005)**

Theoretically, it can be determined by these hardware and software technologies. However, as I mentioned before, that is too complicated technologies and they have the hardware part. It means It's beyond my control, Utilizing API helps me to use the GPS function.

### **3.3.3 API**

API is the abbreviation of Application Programming interface.

We use software applications and then the interface is between a user and software application via user interface.

However, different software applications also can communicate and connect with each other when we use an API.

What is more, it is a set of protocols and defines the method and data formats that different applications use to request and respond.

Moreover, it has not only connecting technology but also many advantages of API.

- Compatibility

Even if these separate software entities use different platforms or programming language, they can interact with each other.

- Standardization

These standardizations are already decided and help developers integrate different software easily.

- Functionality

APIs have a wide range of functionality, including data access, authentication, messaging, and so on. For instance, Web APIs enable applications to have access to data and services over the Internet using Internet protocol HTTP.

- Security

These separate software have the abstractions or encapsulation of the detail and structure by using API.

In addition, APIs can implement security functions such as authentication, authorization, and encryption to ensure that only authorized users and applications can request another one.

- Maintenance

As long as developers don't change the compatibility of existing two software applications, we can develop each other and provide new features are easily mounted. (BIEHL, 2016)

To sum up, API has a lot of advantages and enables us to use GPS technology seamlessly in our applications. By leveraging GPS APIs, developers can access not only User's location which is high-level and complicated technology and needs hardware and satellites, but also their location-based services, and integrate mapping functionalities in their applications. In addition, we can create our own functionalities on our platform using them.

### 3.3.4 Directions of some kind

In this part, based on public transportation navigation, we focus on directions on the ground especially on roads.

In addition, the direction depends on what kind of type routing is or what fact is the highest priority, so firstly, I will explain the routing type.

"Routing" refers to setting a starting point and destination, calculating distance and time on a road network consisting of roads, intersections, landmarks, and so on. For example, driving from the starting point to the destination in the shortest time or shortest distance.

We can say it would be what is our priority and it decided the directions.

Moreover, As regards environmental pollution, recently we can choose Eco-routing navigation and it offers the most eco-friendly route which has the minimal amount of fuel and/or emits the least amount of emissions and definitely not the shortest duration or distance route. (BORIBOONSOMSIN, 2012)

We have to consider many things, like travel times, fuel consumption and the algorithm is too complicated thought, even the worry about environmental pollution can decide our direction. Essentially, that information, such as public transportation schedules, real-time data, and distance is big data. There are many algorithms to calculate them and we must choose them to get the best directions.

Here, I pick up the most common shortest-route optimization algorithms which are called The Dijkstra algorithm with the binary heap priority queue and A\* algorithm and it's said that it is high efficiency in traffic information systems wherever it calculates the shorter road. Dijkstra algorithm is introduced by Edsger W. Dijkstra in 1956.

The shortest-path tree is generated from Dijkstra's Algorithm by determining the shortest path from a node to all other nodes with the minimum weight in the graph. In essence, this algorithm identifies the minimum weight path between any given pair of nodes. (Anie Lusiani1, 2021)

Speaking of public transportation, nodes are destinations, some stops, such as bus stops, tram stops, metro stops, even boat and cable car stops in Prague and weight is time through the track.

### 3.3.4.1 Data structure of Dijkstra algorithm

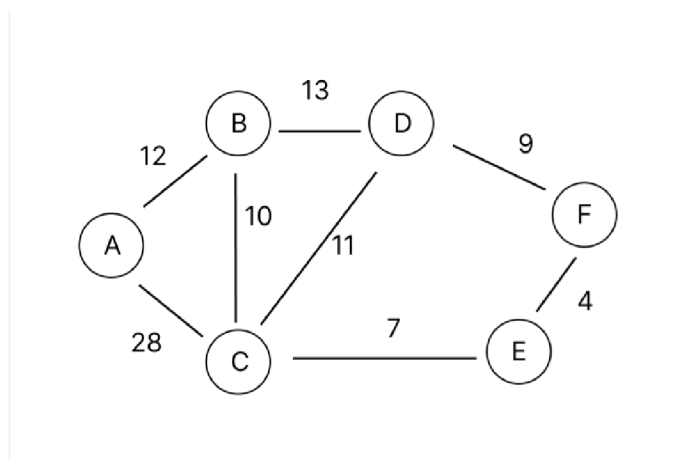
Before I explain the algorithm, I will explain what type of data structure it needs.

Graph 1 shows a graph of the map, where the intersections indicated by circles are called nodes, and the numbers written on the lines mean the weight between nodes and are called costs. In addition, edge is a connection between nodes. The starting point is represented by a double circle S, and the destination is represented by a double circle G. Representing a map in this format is called map graphing. Furthermore, a map of a road network expressed in this way is called a calculation map. [独立行政法人 工業所有権情報 研修館, 2005]

However, for the Dijkstra algorithm, the algorithm needs the data as an adjacency matrix.

For this reason, we should adjust any data type to the adjacency matrix data.

The Graph 2 is a example of weighted graph and the below PHP code shows the adjacency matrix data of graph 2 and other data definitions.



**Graph 2 Weighted Graph, (kiiimiis, 2015)**

```
define('STATION_NUMBER', 6);
define('START_STATION', 0);

$stations = array('A', 'B', 'C', 'D', 'E');

//adjacencyMatrix
//Each array has the time from the each station to the data id
numbers's stations
$adjacencyMatrix = array(
    //one', 'two', 'three', 'four', 'five', 'six'
    array(0, 12, 28, 0, 0, 0), //from A
```

```

    array(12, 0, 10, 13, 0, 0), //from B
    array(28, 10, 0, 11, 7, 0), //from C
    array(0, 13, 11, 0, 0, 9), //from D
    array(0, 0, 7, 0, 0, 4), //from E
    array(0, 0, 0, 9, 4, 0) //from F
);

```

### 3.3.4.2 Dijkstra algorithm

From here, I will explain the algorithm.

First thing First, we select a starting point. In the navigation system, mostly it would be User's current location. It's also recorded to be visited or not, which usually has a Boolean datatype as a state, and gets 0 weight.

After the first step, we set the arrival cost of every node to infinity without the starting point.

```

for ($i=0; $i < STATION_NUMBER; $i++) {
    $currentCost[$i] = -1; //Here, minus one is infinity
    $fix[$i] = false;
}

//we set 0 as the starting point
$currentCost[START_STATION] = 0;

```

Secondly, we call this direction step two, which calculates the cost of reaching each node connected by an edge from the selected node, and updates the cost if it is lower than the current cost of reaching each node. After that, we will move to the node that is connected and has the shortest distance to the points that have not been traced yet from the current node. After moving, we give the "true" variable to current node to prove we reached. When every node has "true" value for the visitation, it's finished. Otherwise, we repeat the step two until the destination node finally has true. (kiiimiis, 2015)

```

//this loop finish until every node is checked
while (true) {
    //the times are set -1 as infinity
    $minStation = -1;
    $minTime = -1;
    for ($i = 0; $i < STATION_NUMBER; $i++) {
        if (!$fix[$i] && ($currentCost[$i] != -1)) {

```

```

        //Check for stations where the pattern has not yet
        been identified and where the travel time is short.
        if ($minTime == -1 || $minTime > $currentCost[$i])
    {
            $minTime      = $currentCost[$i];
            $minStation = $i;
        }
    }
}
if ($minTime == -1) {
    //whole stations are checked or the first time is
infinity
    break;
}

    // check for times between curent station to whole next
stations
    for ($i = 0; $i < STATION_NUMBER; $i++) {
        if (!$fix[$i] && $adjacencyMatrix[$minStation][$i] >
0) {
            // new time is the time where we came from and
next station
            $newTime      =          $minTime      +
$adjacencyMatrix[$minStation][$i];
            //comparing the time and time which is alraedy
checked and from another route
            if ($currentCost[$i] == -1 || $currentCost[$i] >
$newTime) {
                $currentCost[$i] = $newTime;
            }
        }
    }
    // mark as we already checked it
    $fix[$minStation] = true;
}
for ($i = 0; $i < STATION_NUMBER; $i++) {
    echo ($stations[START_STATION] . "→" . $stations[$i] . " : "
. $currentCost[$i] . "分");
    echo '<br>';
}
}

```

### 3.3.4.3 Advanced Dijkstra algorithm

When we choose the node with the shortest distance and which is not traced ever, we can use “priority Queue”.

The above normal Dijkstra algorithm is linear search and the calculation time is  $O(n)$  when

we choose the node with the shortest distance and which is not traced ever. However, instead of linear search, we can replace it with “priority Queue”.

Here, as an example, I will use Heap map for the priority Queue.

Priority queues require  $O((E+V)\log N)$  computation for insertion and retrieval and the computation for determining the minimum vertex is faster than linear search  $O(n)$ .

The Advanced Dijkstra algorithm is a bit different from the above code, so I will explain the difference between the original and advanced one with a heap map and write the code below.

Firstly, we prepare three data structures which are composed of two arrays and one heap map.

One of the arrays is called parent and has the node which is the shortest distance. Thus, the final data of this is supposed to be the best routing answer. Another one, called Distance, has the arrival cost from a starting point. Thus, the final data is supposed to be the best shortest distance answer.

In addition, we select a starting point and give 0 to the node value in Heap map and an infinity value to others.

Secondly, we pick up the smallest node in Heap map ( firstly, a starting point is supposed to be chosen ) and give the node the record which shows It has been visited.

Thirdly, we will add the smallest value which is from the node we pick up in the second step to the distance array.

Fourthly, we calculate the arrival cost which is from the node to every node that is connected and record it in the Heap map.

Fifthly, we put the last node of every node which is connected to the node we pick up in the second step in parent array.

Until the last node which is the final goal in Heap map is picked up, we repeat it from the second step. (QURESHI, 2021)

## 4 Practical review

In this practical part, I will make a navigation app based on the knowledge from the literature review and explain the implementation process and structure of the application with the same flow of the literature review such as first thing is a map, second is current location, and third is some kind of directions.

Once I have completed this step, I will complete creating a basic map navigation application. After that, the implementation will be completed by addressing and resolving the following issues in “the problem solutions” chapter.

I expect the created navigation application to develop navigation solutions that not only provide accurate wayfinding, but also incorporate real-time traffic information, traffic data, and a user-friendly interface.

In terms of user interface, I used Figma to create the prototype below.

Finally, I will evaluate the results in the “Results and Discussion” chapter.

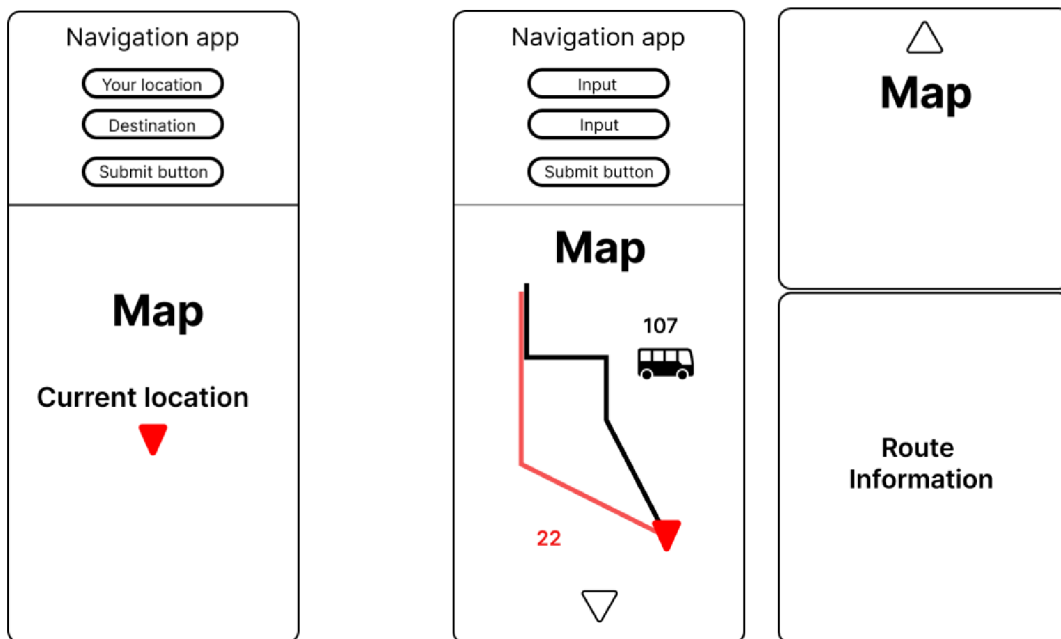


Figure 6 UI prototype design, Source: Author himself



## **4.1 Review of the software development kit**

This navigation application is implemented in Laravel using various Google Map APIs. Laravel is one of the most convenient and famous PHP frameworks and is used for web applications. PHP is an open source and server-side scripting language, which means it allows to manage dynamic contexts and databases. In addition, it is embedded in an html file and supports CSS and JavaScript. Google Maps Platform mainly offers three platforms consisting of Android, IOS, and JavaScript. This application is a web application, then I chose the JavaScript platform called Maps JavaScript API.

The Maps JavaScript API offers extensive customizability, allowing developers to tailor maps according to their specific requirements. With a wide range of customization options, developers can modify map types, add custom content and imagery, and enhance user experience through the integration of layers, styles, controls, and events.

In that respect, there is no need to manage large amounts of map data, and the complex processing of handling real-time traffic information and traffic data to provide route guidance can be avoided. In addition, Laravel has excellent database management technology. At that point, you can create highly scalable applications by adding other data to these functions.

Taking them into consideration, using Laravel (In this application, Laravel Framework 10.43.0 version and PHP 8.1.7. version ) is reasonable and understandable.

## **4.2 Map**

In this section, we prioritize the amount of map data, the integration of real-time data sources, and mechanisms for seamless updates to the map database. Leveraging Maps JavaScript API's extensive functionality and meeting the conditions, we aim to deliver responsive and reliable Map data that adapts to evolving user needs and changing urban landscapes.

Furthermore, it offers intuitive user interfaces and rich map visualizations. It means that, in terms of map design, we utilize these advantages.

Also, as discussed in the literature review, map data, referred to as Geographic Information Systems (GIS) are composed of multiple layers, each containing distinct sets of location-based information such as parcels, topography, demography, and imagery.

The API also serves some special and additional layers such as traffic, transit, bicycling, and heat map layers. In the details, they are real-time traffic information, local transit networks, bicycling route information, and the intensity of data. Moreover, they support the KML and GeoRSS data formats for using more geographic information and these data supports allow us to add more distinctive data layers.

I explained how to load the map using Maps JavaScript API below.

```
<script                                async                                defer
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyDebRyC
5KKR__t48vDf1D9n6ZdToztsES4&libraries=places&callback=initMap
">
</script>
```

```
const { Map } = await google.maps.importLibrary("maps");
// to down load map module
map = document.getElementById("map");
  opt = {
    zoom: 13,
    center: latlng,
  };

// show google map on device screen
mapObj = new google.maps.Map(map, opt);
```

First of all, we must load the Maps JavaScript API.

This above code in home.blade.php file allows us to use the functions and call the initMap global function after the Maps JavaScript API is loaded completely.

After loading the API, the codes in the result.js file load the map module into the const variable and the next code shows the map meeting options in the div section in the PHP file. the function requires two parameters. The first parameter is the div element that has map id here and The second parameter is options.

At this point, utilizing the Maps JavaScript API, we integrate real-time data sources and

ensure seamless map database updates, offering intuitive UIs and rich visualizations.

### 4.3 User's location

Getting the current location, building upon the theoretical foundations outlined in the literature review, requires many technologies.

Here, I will use and explain geolocation API to get current location information.

Thus, this application and device must meet the W3C Geolocation standard.

In addition, Maps JavaScript API allows us to connect and use Geolocation API with their code.

#### Geolocation API

The Geolocation API provides a scripting interface for retrieving geographic position information associated with the hosting device. It defines a high-level interface to location data specifically linked to the device executing the implementation. Common sources of location data include the Global Positioning System (GPS) and location inferred from network signals such as IP addresses, RFID, WiFi and Bluetooth MAC addresses, and GSM/CDMA cell IDs, as well as user input. The API itself is neutral regarding the underlying location data sources, and it does not guarantee that the API will return the actual location of the device.

The code has a conditional branch and checks if we can meet the W3C geolocation standard and use the API, we set some options and start the `getCurrentPosition` function.

In addition, we will call the callback function which is called `setLocation`.

If not, we get the error and then it shows "Your browser can't meet W3C Geolocation standard".

```
// check u can use the geolocation
if ("geolocation" in navigator) {
  var opt = {
    "enableHighAccuracy": true,
```

```
"timeout": 10000,
"maximumAge": 0,
};
navigator.geolocation.getCurrentPosition(setLocation, showErr, opt);
}
else {
  alert("Your browser can't meet W3C Geolocation standard");
}
```

The setLocation function has a pos parameter, which is the GeolocationPosition object.

The GeolocationPosition interface represents the location of a device at a given time and is represented by GeolocationCoordinates.

The GeolocationCoordinates interface provides the position and altitude of the device on Earth, as well as the accuracy with which these properties are calculated. The geographic position information is provided in terms of World Geodetic System coordinates (WGS84). (mdn, n.d.)

Using them, we get the current position of latitude and longitude respectively and the LatLng object is created using the current latitude and longitude coordinates obtained earlier. Moreover, we use google.maps.Map class and google.maps.Marker in order to set the marker of the current location.

In summary, this code fetches the user's current location, creates a Google Map centered at that location, and adds a marker to indicate the user's position on the map.

```
function setLocation(pos) {

  // get latitude and longitude
  var lat = pos.coords.latitude;
  var lng = pos.coords.longitude;

  // set up latitude and longitude
  latlng = new google.maps.LatLng(lat, lng);
  map = document.getElementById("map");
```

```
opt = {
    zoom: 13,
    center: latlng,
};
mapObj = new google.maps.Map(map, opt);
// set marker
marker = new google.maps.Marker({
    position: latlng,
    map: mapObj,
    title: 'Current position',
});
}
```

#### 4.4 Directions of some kind

Here, I focus on the direction of a destination.

First, we need a form to get places of departure and destination.

The following HTML code shows the form and @csrf which works as a Cross-site request forgeries attack protection. (Laravel Holdings Inc, n.d.)

To make the form responsive, I wrote the CSS code below and the code also ajust others, such as font sizes, the button design. In addition, The google.maps.places.Autocomplete class in the result.js file integrates a predictive input into your application. It dynamically suggests places based on user input. The function responds to text changes in real-time, presenting a dropdown list of predictions for enhanced usability such as search efficiency and user interaction. The inputs are adjusted to google map data and has a unique identifier of a place as well as the unique identifier the Google Maps database has.

In addition, google document says the placeId is the most accurate way of identifying a place. (Google, n.d.)

Hence, we don't need to create geocoding functions and manage a complecated and big data database.

```

<form class="form" action="">
    @csrf
    <input type="text" class=""
placeholder="current location" id="source">
    <input type="text" class=""
placeholder="destination" id="des">
</form>
<button onclick="showRoute()">Get Direction</button>

```

```

html,
body {
  height: 100vh;
  margin: 0;
  padding: 0;
}

h1 {
  text-align: center;
  color: #3191d1;
}

.container {
  margin: 10px 20px;
}

.box-design {
  height: 40px;
  width: 200px;
  margin-bottom: 20px;
}

.form {
  width: 60%;
  margin: 20px auto 0px auto;
  display: flex;
  justify-content: space-around;
}

.btn {
  border-radius: 0.5rem;
  display: block;
  width: 200px;

```

```

height: 50px;
margin: 0 auto;
color: #fff;
background-color: #4ea2da;
border-bottom: 3px solid #4ea2da;
border-color: #4ea2da;
}

.btn:hover {
color: #fff;
background: #3191d1;
border-bottom: 2px solid #3191d1;
}

btn-shadow {
-webkit-box-shadow: 0 3px 5px rgba(0, 0, 0, .3);
box-shadow: 0 3px 5px rgba(0, 0, 0, .3);
}

@media screen and (max-width:480px) {

h1 {
font-size: 1.8em;
}
.form {
flex-direction: column;
}
}

#map {
height: 80%;
}

```

```

sourceAutocomplete = new google.maps.places.Autocomplete(
    document.getElementById('source')
)
desAutocomplete = new google.maps.places.Autocomplete(
    document.getElementById('des')
)

```

After we get the places with auto compliment service, we click the button to submit them and then the showRoute() function starts to work.

The following code in route.js matches the calculation of routing and the official document shows that the highest prioritization is travel time and other factors, such as distance, number of turns, and many more, may be taken into account. (Google, n.d.)

The function is embedded by google.maps.DirectionsService object and executed by route() function. In addition, this application focuses only on public transportation. Therefore, I set the option to transportation output as travelMode: google.maps.TravelMode.TRANSIT, Next, The routing information is reflected in the map by setDirections() function of google.maps.DirectionsRenderer object.

At this point, we created the fundamental map navigation application and the application meets responsive design.

```
function showRoute() {
  var source = document.getElementById('source');
  var dest = document.getElementById('dest');

  directionsService = new google.maps.DirectionsService();
  directionsRenderer = new google.maps.DirectionsRenderer();
  directionsRenderer.setMap(mapObj);

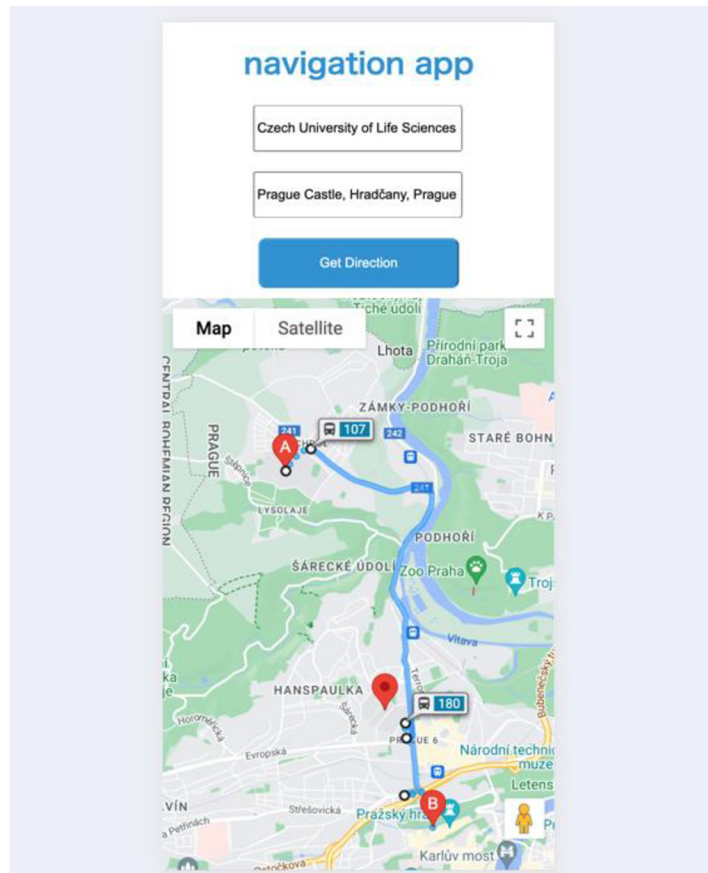
  directionsService.route(
    {
      origin: {
        query: document.getElementById("source").value,
      },
      destination: {
        query: document.getElementById("des").value,
      },
      travelMode: 'TRANSIT',
      provideRouteAlternatives: true,
    })
    .then((response) => {

      directionsRenderer.setDirections(response);

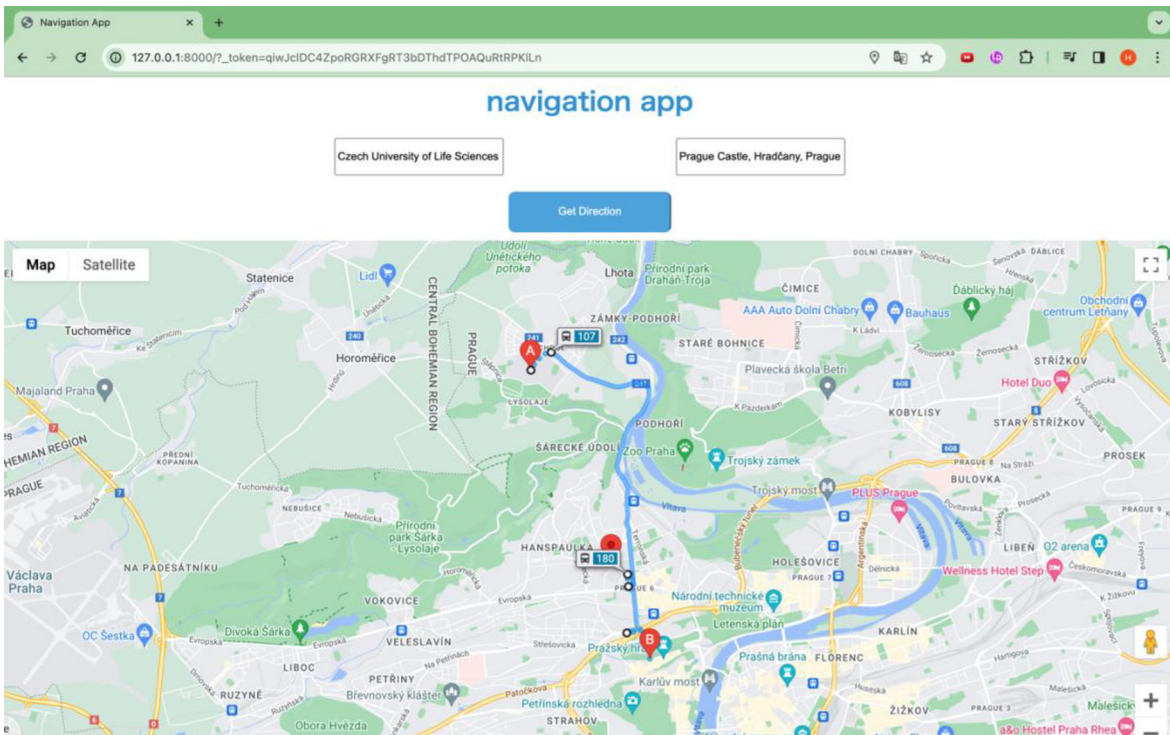
    })

    .catch((e) => window.alert("Directions request failed
due to " + directionsService.DirectionsStatus));
}
```





**Figure 7 the fundamental development result Smartphone version, Source: Author himself**



**Figure 8 the fundamental development result Web version, Source: Author himself**

## 4.5 The problem solutions

As I mentioned in the Introduction chapter, Traffic jams tend to occur in Prague in the evening, and trams and metros tend to be less likely to get stuck in traffic jams than buses since they have their own route that common cars are not allowed to use.

In addition to this, existing navigation apps usually suggest the shortest stops, such as bus stops or train stations, but sometimes a short walk can get you to your destination faster.

To address the first challenge, I will incorporate real-time traffic information and transit layer into the map and implement the ability to select trams and metros to avoid busy roads. This allows users to avoid traffic jams and travel more efficiently.

TrafficLayer object has real-time information and the Transit layer object allows us to display a city's public transit network on the map with thick colored lines.

To address the second challenge, we provide the options of multiple routes from nearby stops and display a transit layer on the map so users can visually see how far they have to walk and choose the best route for themselves because the API doesn't have the function to set up the own walk speed. This allows users to choose their public transportation options according to individual circumstances and preferences, resulting in a smoother move.

By combining these features, the navigation app can effectively address Prague's traffic congestion problem and the challenges of existing apps, providing a convenient transportation option for users.

The bellow code gives us real-time traffic information and a transit layer.

```
const transitLayer = new google.maps.TransitLayer();
const trafficLayer = new google.maps.TrafficLayer();

trafficLayer.setMap(mapObj);
transitLayer.setMap(mapObj);
```

In order to get multiple routes, we must set `provideRouteAlternatives` to `true`, and then, we get the return as a `routes` instance. In the detail, the route instance has `overview_path` and `overview_path` stores an array of data of `LatLngs` indicating the approximate path of the resulting routes. In addition, `Polyline` class uses the `overview_path` as an option to connect each array data and show a route.

```
provideRouteAlternatives: true,
```

```
var routes = response.routes;
    var polyline = new google.maps.Polyline({
        path: route.overview_path,
        strokeColor: '#'+
Math.floor(Math.random()*16777215).toString(16),
        strokeOpacity: 0.8,
        strokeWeight: 4,
        map: mapObj
    });
```

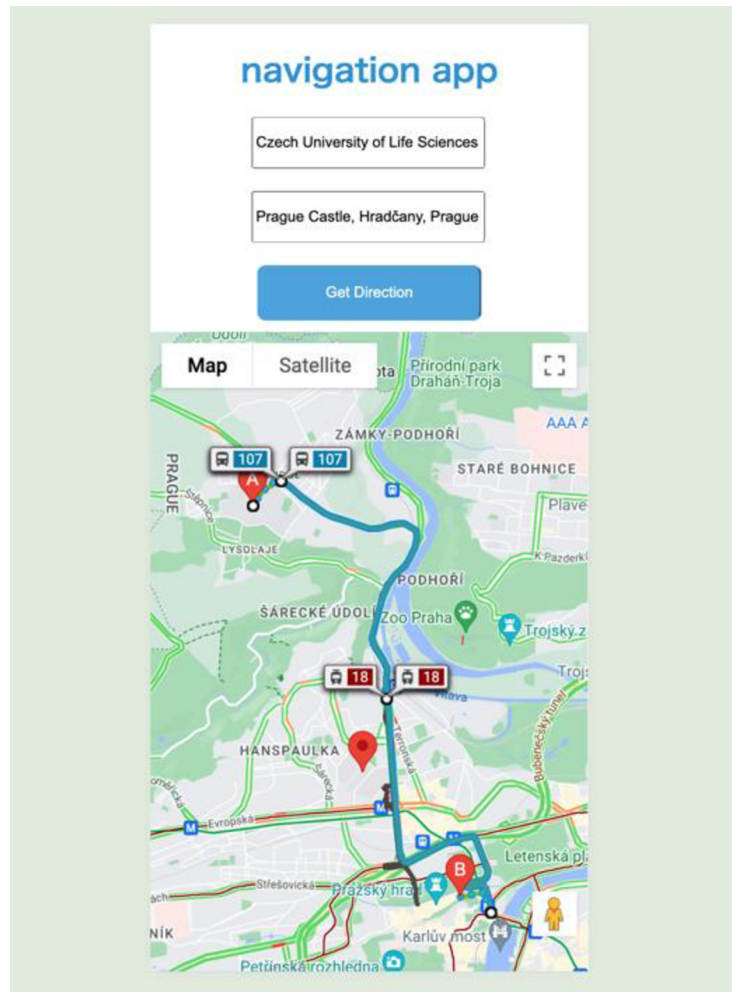


Figure 9 multiple routings traffic layer ,and transit layer, Source: Author himself

## **5 Result and Discussion**

We completed creating a basic navigation application by utilizing the Maps JavaScript API. The user interface turned out to be the application I expected. However, by leveraging a combination of features in the Maps JavaScript API I was able to begin to solve the problem, but the result wasn't perfect.

As a result, although this application is capable of handling the accurate directions, real-time traffic information, and traffic data that I expected, some issues remained in terms of application being better than existing applications.

### **5.1 Comparison**

#### Advantage

Google Map and Apple Map display only one traffic layer or transit layer.

On the other hand, this application allows you to display two and can provide multiple routes.

#### Disadvantage

There are some various limitations and disadvantages were found.

The first is that although it is now possible to display multiple routes, the numbers for buses, trams, etc. will only be displayed for one of the best routes google map decides.

The downside is that if you choose the best route provided by this app, which is not the best route displayed by Google Maps, you won't know the bus number.

Second, since the transportation schedule is not listed, it is not possible to know the allowable time until the stop, and you may end up losing more time by choosing a different stop than the best route.

Third, when selecting a destination, these existing applications allow you to easily set a destination by selecting a convenient categorized icon or pointing to a pin, but this application allows you to input by typing. It can be said that it is not user friendly as it only has the auto-complement functions.

## 5.2 Recommendation for future development

For developers interested in developing this navigation application related to maps JavaScript API, there are several valuable recommendations to consider:

- This created application suggests multiple routes based on the traffic layer and transit layer. However, by using the schedule data and displaying schedules, it will be able to make better suggestions to the user.

In this case, there are two ways to do this: save the data of transportation schedules provided by PID (Pražské integrované dopravy) in a database and create an algorithm to display it, or use the API to call time schedule data and display only the necessary data.

In the detail, the response.routes instance, which is the return of directionsServices(), has the multiple routings information as a route type array. (Google, n.d.)

- Keeping up to date with the latest versions is crucial, as new techniques and tools may become relevant to the topic.

- As it is convenient to use the maps JavaScript API, you will basically be creating applications based on it, so knowledge of JavaScript is mainly required.

- Additionally, I recommend mobile type application of this, not web application.

This is especially important for navigation applications that need quick access to maps and directions while on the move. In this case, knowledge of programming languages and frameworks compatible with iOS and Android platforms is required.

Hence, developers need the knowledge of creating mobile applications.

- Expanding the scope of data collection beyond Google map API to include more diverse source of information like Apple Map API can leads us to create the more accurate application.

By incorporating these recommendations, future developers can develop applications using these technologies or implement a better application of map navigation application and build upon the existing body of work in this field.

## 6 Conclusion

In this thesis, we have developed the map navigation application for public transportation. For this purpose, the Maps JavaScript API of Google is employed and JavaScript programming is executed. The application is intended to be used to give the best route of public transportation for each user.

Firstly, we surveyed a navigation application structure and I tried to compare the three existing applications at the same time.

However, according to the comparison parts, there is only a little technical information about them and PID Lítačka uses Apple API.

As a result, the literature review shows mainly navigation application structure.

The structure is way too complicated and each section has also high-level technologies, especially GPS.

In the practical part, I created the application and owned Maps JavaScript API. These functions are really convenient and also give tips for solving problems.

Actually, navigation applications and the applications that use these technologies are created by smart many workers in big tech companies, such as Google, Apple, and Yahoo!

I recognized navigation application is a really high-level application through the literature review.

For further research, this developed application can overcome these limitations to have more functions and be more useful in the application. To do that, we must study algorithms and data structures and compare these process times without Google Map APIs.

The reason why it is without Google Map APIs is that Google doesn't allow developers to make money using the functions of APIs under some conditions. For example, free downloads can permit developers to use them.

Unfortunately, designing and implementing those functionalities in this way is too much time demanding in this bachelor's thesis and I decided to conclude this thesis at this point.



## 7 References

- A, J. (2000). *System, Global Positioning*. Volpe National Transportation Systems Center (U.S.).
- al., J. H. (2022). Evaluation of Positioning Accuracy of Smartphones under Different Canopy Openness. *Forests*.
- Andrew Lookingbill, E. R. (22. July 2019). *Google Maps 101: how we map the world*. Načteno z google: <https://blog.google/products/maps/google-maps-101-how-we-map-world/>
- Anie Lusiani1, \*, E. (2021). *Determination of the Fastest Path on Logistics Distribution by Using Dijkstra Algorithm*. ATLANIS PRESS.
- Apple. (nedatováno). *Apple Maps Image Collection*. Načteno z <https://maps.apple.com/imagecollection/us>
- BIEHL, M. (2016). *RESTful Api Design*. API-University Press.
- BORIBOONSOMSIN, K. (2012). *Eco-routing navigation system based on multisource historical and real-time traffic information*. IEEE Transactions on Intelligent Transportation Systems.
- Christopher E. White, D. B. (2000). *Some map matching algorithms for personal navigation assistants*. Transportation research part c: emerging technologies.
- Google. (nedatováno). *Maps JavaScript API*. Načteno z Google Maps Platform: <https://developers.google.com/maps/documentation/javascript/reference/>
- KARKI, B., & WON, M. (2020). *Characterizing power consumption of dual-frequency GNSS of smartphone*. GLOBECOM 2020-2020 IEEE global communications conference.
- mdn. (nedatováno). *Mmdn\_*. Načteno z GeolocationCoordinates: <https://developer.mozilla.org/en-US/docs/Web/API/GeolocationCoordinates>
- Mulla, A., Baviskar, J., Baviskar, A., & Bhovad, A. (2015). *GPS assisted Standard Positioning Service for navigation and tracking: Review & implementation*. IEEE.
- Official U.S. government information about the Global Positioning System (GPS) and related topics*. (28. 6 2022). Načteno z GPS.GOV: <https://www.gps.gov/systems/gps/space/>

- Perez, S. (2023). *Apple Maps' business listings are about to get more detailed with launch of 'Apple Business Connect'*. TechCrunch.
- TÓTH, R., BALLA, D., & ZICHAR, M. (2020). *Analysis and optimization of KML files*. IATED.
- Tsui, James Bao-Yen. (2005). *Fundamentals of global positioning system receivers: a software approach*. John Wiley & Sons.
- Werner, P. (2022). *Introduction to Map Layers for Backcountry Navigation*. SectionHiker.
- Zhang, A., Qi, Q., & Jiang, L. (2007). *GeoRSS based emergency response information sharing and visualization*. IEEE.
- 独立行政法人 工業所有権情報 研修館. (2005). カーナビ経路探索技術.  
独立行政法人 工業所有権情報 研修館.

## 8 List of pictures, tables, graphs

### 8.1 List of figures

Figure 1 The Google Map Czech rank in Apple store, Source: Author himself .....	16
Figure 2 data layer (Werner, 2022).....	17
Figure 3 the Praking spots of PID Lítačka UI, Source: Author himself.....	20
Figure 4 the sales spots of PID Lítačka UI, Source: Author himself.....	20
Figure 5 formulas (Tsui, James Bao-Yen., 2005).....	25
Figure 6 UI prototype design, Source: Author himself .....	32
Figure 7 the fundamental development result Smartphone version, Source: Author himself .....	41
Figure 8 the fundamental development result Web version, Source: Author himself.	42
Figure 9 multiple routings traffic layer ,and transit layer, Source: Author himself .....	44

### 8.2 List of graphs

Graph 1 User location and three circles in two dimension (Tsui, James Bao-Yen., 2005) .....	24
Graph 2 Weighted Graph (kiiimiis, 2015) .....	28

## **Appendix**

KML.....Keyhole Markup Language (KML) is XML-based and a popular geovisualization data format. For example Google Earth. (Zhang, Qi, & Jiang, 2007)

GeoRSS..... GeoRSS is also XML-based data format and the data are Geographically Encoded Objects for RSS (Really Simple Syndication) that incorporates geographic data (i.e. latitude/longitude coordinates). This is useful for plotting any data that might need to be placed on a map. (TÓTH, BALLA, & ZICHAR, 2020)