Location-Based Alerts for Passengers in Public Transportation

Antonia Stieger

BACHELORARBEIT

eingereicht am

Fachhochschul-Bachelorstudiengang

Mobile Computing

in Hagenberg

im März 2025

Advisor:

FH-Prof. Dr.-Ing. Jens Krösche

Declaration

I hereby declare and confirm that this thesis is entirely the result of my own original work. Where other sources of information have been used, they have been indicated as such and properly acknowledged. I further declare that this or similar work has not been submitted for credit elsewhere. This printed copy is identical to the submitted electronic version.

Hagenberg, March 15, 2025

Antonia Stieger

Contents

Abstract

This should be a 1-page (maximum) summary of your work in English.

Kurzfassung

An dieser Stelle steht eine Zusammenfassung der Arbeit, Umfang max. 1 Seite. ...

Chapter 1

Introduction

1.1 Motivation

In today's world, climate change has emerged as a critical global issue, prompting widespread efforts to reduce carbon footprints. One effective strategy to mitigate environmental impacts is the increased use of public transportation over private vehicles. Despite the clear environmental advantages, such as reduced fuel consumption and lower emissions, many commuters still favor the convenience and flexibility of private cars. This preference persists even though mass transit can significantly alleviate urban traffic congestion and improve air quality. As Minelgaitė, Dagiliutė et al. [12] highlight, expanding the adoption of public transport is essential for minimizing environmental damage. Specifically, the United Nations [53] reports that individuals could reduce their carbon footprint by up to 2.2 tons annually by switching from personal vehicles to shared transportation systems. However, as Zheng and Krol [48] note, widespread adoption will only occur when public transport becomes the "most convenient option for getting around." Currently, challenges such as limited service coverage, infrequent schedules, and perceptions of unreliability deter potential users. Thus, making public transit more accessible and user-friendly is crucial for encouraging environmentally responsible travel behavior. This thesis aims to explore how technological innovations can enhance the appeal and functionality of public transportation, ultimately increasing rates of usage.

1.2 Challenges

One significant obstacle to this goal is the strong competition from private vehicles, particularly in suburban and rural regions, where car ownership is widespread, and public transport often struggles to provide a viable alternative. While urban areas often benefit from more comprehensive public transport networks, less populated regions face specific difficulties with limited routes, infrequent schedules, and longer distances between stops. In these areas, missing a stop can result in substantial delays or even stranding passengers. These discrepancies highlight the need for tailored technological solutions that address the distinct requirements of passengers in both urban and rural settings. Additionally, issues such as comfort, reliability, and safety significantly deter public transport use. A study by Friman and Fellesson [3] on passenger satisfaction and stress reduction states that passengers frequently express dissatisfaction with these factors, which affects their overall experience. The added anxiety of potentially missing a stop only exacerbates the situation, creating further reluctance to use public transportation according to Minelgaitė, Dagiliutė et al. [12]. Addressing these concerns through innovative systems that are providing tailored alerts for commuters is essential to overcoming the reluctance many people feel toward public transport.

In addition to the challenges related to public transport, several technological challenges must be addressed when developing an effective alert system. Managing reminders is one issue, especially within the constraints of app development guidelines, such as those set by Apple, which restrict background activity and control over alerts. Geofencing also requires constant location tracking, which not only drains the battery but also might raise privacy concerns, as highlighted by Shevchenko and Reips [19]. They also point out the issue of choosing an inappropriate radius size for geofencing, which can lead to inaccuracy or latency. Additionally, some devices may lack real-time location capabilities, meaning geofencing might not work reliably or at all for those users.

1.3 Goals

This thesis aimes to analyze and introduce a user-focused app-based alert system that integrates schedule data and geofencing. Passengers are provided with a public transport app that reduces stress and increases accessibility. This approach, particularly valuable in rural areas, ensures that regular commuters as well as tourists, children, and the elderly can navigate public transportation networks with ease, ultimately encouraging greater adoption of public transport.

The app offers users the flexibility to choose from three distinct alert modes based on their preferences. The first mode sends alerts according to the scheduled arrival times of transports. The second uses three geofences along the route to progressively notify passengers as they near their desired stop. The third relies on nested geofences around the final destination to trigger alerts. In addition to the modes, the user will be alerted with three escalating levels of urgency, ensuring passengers are notified in an effective manner. By exploring these three modes, this thesis will try to evaluate their respective advantages and disadvantages, and demonstrate their implementation within iOS.

However, Apple's app development guidelines present certain challenges, and this thesis will examine strategies to bypass these constraints. To test the system's functionality, a simulator mode will be developed using a GPX file, allowing the system to simulate journeys along desired routes without requiring on-site testing. Additionally, the app will incorporate voice control for setting up reminders.

1.4 Structure

Before diving into how the implementation of location-based alerts works, Chapter 2 introduces the core technologies used in this project, with a focus on positioning methods such as cellular network positioning, Global Navigation Satellite System (GNSS) and geofencing. The chapter also discusses alert systems on smartphones and their restrictions as well as Speech-to-Text (STT) transcription. Furthermore, the challenges specific to public transportation are highlighted. Chapter 3 outlines the app concept, providing an overview of the different alert modes and explaining their functionality. Chapter 4 details the technical implementation, discussing system architecture, geofence management, notification handling, voice-controlled alert setup, and route simulation. This thesis closes with Chapter 5, which describes what has been achieved and outlines potential next steps.

Chapter 2

Fundamentals

This chapter provides an overview of the core concepts relevant to the thesis. It begins by reviewing various positioning methods as well as systems for indoor and outdoor settings, followed by an examination of how smartphones handle alerts, including different types and influencing factors in iOS. Geofencing technologies are analyzed for their functionality, use cases and limitations. Finally, advancements in technologies such as speech-to-text and large language models are discussed in the context of voice-controlled reminder systems.

2.1 Positioning Methods

The term “positioning” refers to the ability to determine an object’s location within a defined space. According to Küpper [9], positioning relies on the measurement of observables, which describe the spatial relationship between a target and its reference points. Depending on the positioning technology used, observables can include angles, ranges, range differences or velocity and they are measured by analyzing the properties of pilot signals. Pilot signals, such as radio, infrared or ultrasound signals, are reference signals transmitted from a known source and by identifying their origin, a fixed point with known coordinates, the position of the target can be calculated using positioning techniques. These techniques vary based on the type of observable measured and include proximity sensing, lateration, angulation, pattern matching and inertial navigation. The computed position is expressed relative to a chosen reference system. Küpper [9] states that this system could be descriptive such as identifying a specific cell in a grid, or geodetic where the position is represented as two- or three-dimensional coordinates based on WGS-84 or UTM. Further details on the different positioning techniques are provided in the following sections.

2.1.1 Proximity Sensing

Proximity sensing, as described by Küpper [9], is the simplest positioning method. It leverages the limited coverage range of pilot signals to detect the presence or absence of a terminal within a specific area. In this context, a terminal refers to a device or object whose position is being determined, such as a mobile or Internet of Things (IoT) device. The nearest base station to the terminal is identified based on Received Signal

Strengths (RSS) and since pilot signals have a limited range, the terminal's position is assumed to match the location of this base station. Militaru et al. [11] illustrate this concept in Fig. 2.1 where the terminal (UE) is surrounded by three base stations BS_1, BS_2 and BS_3. The terminal receives signals from all three base stations, but BS_1 has the strongest RSS, indicating it is the closest. As a result, BS_1's coordinates are assigned as the position of the terminal.

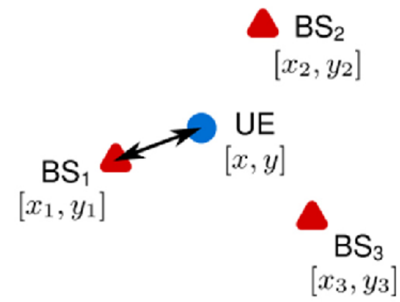


Figure 2.1: Proximity sensing with BS_1 as the nearest base station [11]

In the words of Küpper [9], proximity sensing in cellular networks is often referred to as Cell of Origin (CoO), Cell Global Identity or Cell-ID. A "cell" is a geographic area within the cellular network, managed by a base station that serves as a local signal hub for devices within that area. The accuracy of the location determined through CoO depends largely on the size and shape of the cell, as highlighted by Grejner-Brzezinska and Kealy [6] and Lee et al. [10]. Smaller cells allow for better location estimates, often reaching accuracy within 100 meters in urban areas where cell towers are densely placed. In rural areas, where cell coverage spans several kilometers, the location estimate becomes less accurate.

CoO is considered an inexpensive solution, as noted by Grejner-Brzezinska and Kealy [6] and Küpper [9], because it is compatible with the existing infrastructure. A terminal can determine its location using CoO either through an active connection to a base station or by passively receiving broadcast signals while in an idle state. When actively connected, the terminal's location is identified using the coordinates of the serving base station. In idle mode, the terminal can either query a remote database to retrieve the base station's location using its cell ID or rely on the base station to include its location directly in the broadcast signal, reducing the need for external lookups.

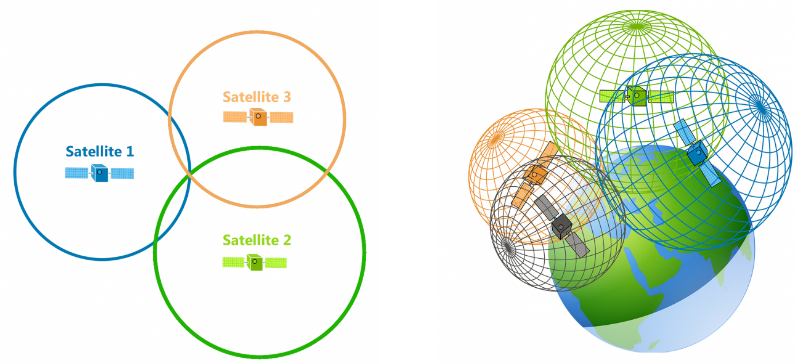
2.1.2 Lateration

Lateration, the most widely used method for localization according to Lee et al. [10], determines a target's location by measuring distances or distance differences from multiple reference stations. These measurements are referred to as "pseudoranges" because they include errors that distort the true ranges. Common error sources in lateration include clock errors, atmospheric effects or multipath propagation. Lateration techniques are categorized into circular lateration, which relies on absolute distance measurements and hyperbolic lateration, which uses differences in these distances across stations.

Circular Lateration

Circular lateration uses distance measurements to multiple base stations to derive a location. Assuming the base stations are at the same elevation, knowing the distance between the target and a single base station places the target somewhere on a circle centered on that station. Introducing a second base station allows for two possible positions where the two circles intersect. Adding a third base station resolves this uncertainty, pinpointing the target's exact location at the single intersection point of all three circles, according to Küpper [9]. This process, known as trilateration, utilizes the Pythagorean theorem for calculations and is illustrated in Fig. 2.2 (a).

In three-dimensional space, each distance measurement defines a sphere around a base station. Kolodziej and Hjelm [8] note that with only three base stations, the target's position is narrowed down to two possible points where the spheres intersect. Typically, one of these points can be dismissed as implausible, such as a location in outer space. To eliminate any ambiguity, a fourth base station is introduced as can be seen in Fig. 2.2 (b), ensuring a unique position fix. Additionally, since circular lateration determines position based on absolute distances, the fourth base station is necessary to synchronize clocks and correct for clock offset errors between the base stations and the target. Any clock offset would introduce range errors leading to incorrect positioning, which makes circular lateration more demanding in terms of synchronization requirements.



(a) **(b)**

Figure 2.2: Circular lateration with three (a) and four (b) base stations (satellites) [30]

Hyperbolic Lateration

In contrast to circular lateration, hyperbolic lateration determines a position by measuring differences in distance rather than absolute distances. A hyperbola represents all points that maintain a constant range difference relative to two fixed points. Küpper [9] explains that with the known range difference between the target and two base stations, the target's possible locations are constrained along a hyperbolic path between them. This method, illustrated in Fig. 2.3 (a), uses two base stations to determine a single hyperbolic path. However, with just two base stations, the target's precise location cannot be unambiguously determined, as it could lie anywhere along the hyperbola. To resolve this ambiguity, a third base station is introduced, as shown in Fig. 2.3 (b). By adding this third base station, a second hyperbola is created and the target's position is estimated at the intersection of these two hyperbolas. In three-dimensional space, the principle extends to hyperboloids, requiring at least three base stations for an unambiguous position fix. The most important advantage of hyperbolic lateration, according to Werner [23], is that it only requires synchronization among the base stations' clocks, rather than between the stations and the target, because it relies on Time Difference of Arrival (TDoA) measurements. With TDoA, only the relative arrival times at the base stations matter, making the exact emission time from the target irrelevant, as it is the same for all signals and cancels out in the calculation.

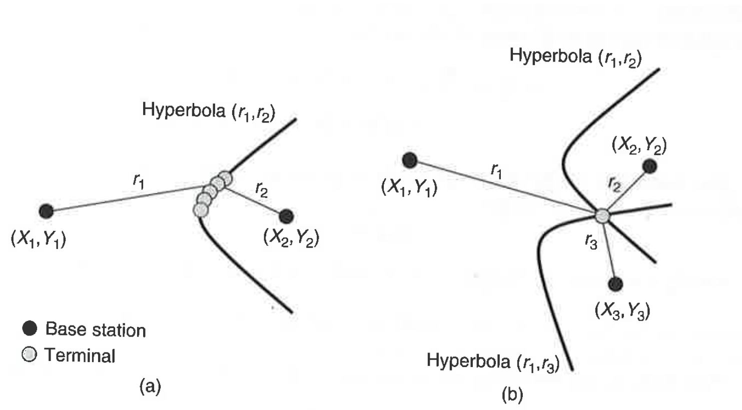


Figure 2.3: Hyperbolic lateration with two (a) and three (b) base stations [9]

2.1.3 Angulation

Angulation is a positioning technique that determines an object's location based on the angles at which signals are received from multiple reference points, rather than measuring distances. According to Küpper [9] it is also called Angle of Arrival (AoA) or Direction of Arrival (DoA). To determine the angle of incoming pilot signals, either the base station or the terminal must be equipped with antenna arrays. However, Küpper [9] highlights that angulation is predominantly used as a network-based method today, meaning that the arrays are more commonly installed at the base station rather than the terminal due to cost and complexity considerations. Once the angles from at least two base stations are known, the object's position can be determined by plotting lines along these angles, the point where the lines intersect reveals the target's location.

Technically, two angle measurements are sufficient to determine a position in 2D, but angulation is sensitive to errors caused by the resolution of antenna arrays, which can lead to approximations of the actual angle rather than precise measurements. These errors increase when the target is farther from the base station, making measurements less reliable at long distances. Additionally, in non-line-of-sight conditions, multipath propagation introduces further inaccuracies as signals reflect off obstacles and arrive at the base station from unintended directions. To mitigate these errors, measurements from at least three base stations are recommended. Fig. 2.4 illustrates how angle measurements from three base stations intersect to determine the terminal's position.

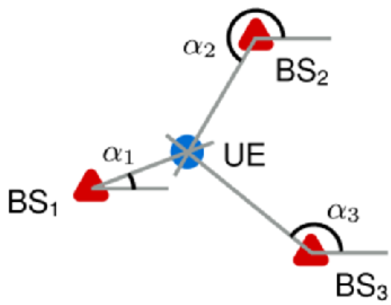


Figure 2.4: Angulation using three base stations [11]

2.1.4 Pattern Matching

Pattern matching is a method of finding a target's position by observing its surroundings and identifying patterns in the environment to determine where it is. Küpper [9] divides the matching of a target pattern with a known reference into two types: optical and nonoptical pattern matching.

Optical Pattern Matching

Optical pattern matching, also known as scene analysis, determines the position of a target by capturing and comparing images of a scene using a camera. This method can be categorized into two types: static and dynamic scene analysis, as described by Küpper [9]. In static scene analysis, a single image of the scene is compared to a database of prerecorded images taken from various positions and angles. The target to be located could be the observer or another object within the scene. By matching the current image with the database, the target's position is identified. Dynamic scene analysis, on the other hand, determines the target's position by analyzing changes between consecutive images captured over time.

Nonoptical Pattern Matching

Nonoptical pattern matching identifies a target's position by analyzing measurable physical properties of the environment. When these properties are based on radio signal characteristics, the technique is commonly referred to as fingerprinting, as noted by Küpper [9] and Werner [23]. At the time of these publications, fingerprinting combined with Wireless Fidelity (Wi-Fi) was a widely used method for indoor positioning due to its inherent resilience to multipath propagation. Unlike lateration or angulation which suffer from signal reflections and interference, fingerprinting captures these effects within the radio maps themselves using them as an important part for positioning. Furthermore, alternative technologies such as Ultra-Wideband (UWB) had not yet emerged and as a result, much of the focus on indoor positioning research was on Wi-Fi-based techniques due to the widespread deployment of Wi-Fi infrastructure.

Fingerprinting consists of two main phases: the offline phase and the online phase. In the offline phase, the area is divided into a grid and at each grid point, RSS from nearby base stations are recorded. These measurements create unique “fingerprints” that are stored in a database, each linked to its respective grid location. During the online phase, the terminal collects RSS values at its current location to form a sample vector. This vector is then transmitted to a server, which compares it with the database of fingerprints recorded during the offline phase. By identifying the closest match, the system estimates the target’s position. However, a significant drawback of fingerprinting is the overhead of creating radio maps and maintaining them to account for changes in the signal landscape.

2.1.5 Inertial Navigation

Inertial navigation approximately calculates the current position by starting from a known location and using measurements of direction, velocity and time to compute the path traveled. This method represents a modern approach to dead reckoning, which was widely used by explorers during the Middle Ages. They navigated using tools such as compasses, log lines and estimates of speed and time, as detailed by Taylor [21]. Figure 2.5 illustrates this process of approximating position through dead reckoning.

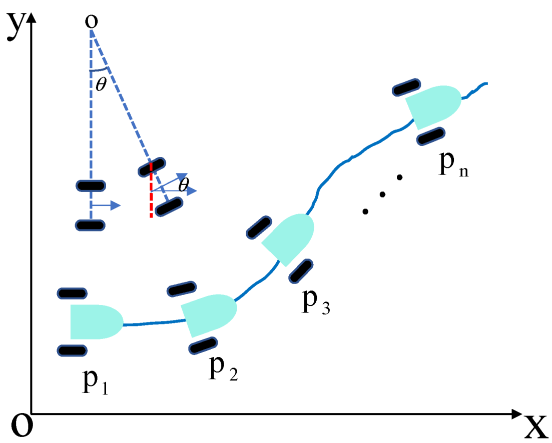


Figure 2.5: Principle of dead reckoning [22]

While inertial navigation shares these conceptual roots, it leverages modern inertial sensors to measure motion far more accurately. Unlike traditional dead reckoning which required frequent manual corrections, an Inertial Navigation System (INS) relies on data from an Inertial Measurement Unit (IMU), which contains accelerometers to measure linear acceleration and gyroscopes to track angular rotation. Inertial navigation errors increase over time due to sensor drift, so regular positions updates from the outside are needed. However, INS are covered in more detail in Subection 2.2.4.

2.1.6 Overview of Positioning Methods

The previous sections have introduced a range of fundamental positioning methods and to provide an overview, Table 2.1 summarizes the key characteristics of the methods. The table is based on the works of Küpper [9] and Werner [23] and outlines the specific features each method observes and the tools as well as techniques used for measurement.

Table 2.1: Positioning methods in comparison

Method	Observable	Measurement
Proximity Sensing	Physical proximity to a reference	Sensing pilot signals
Lateration	Distance and distance difference to reference point	Travel time and RSS of pilot signals (or their differences)
Angulation	Angle to reference point	Antenna arrays
Pattern Matching	Comparison to reference patterns	Camera and RSS
Inertial Navigation	Acceleration, angular rotation and time from a reference	Accelerometers, gyroscopes and additional sensors depending on the infrastructure

2.2 Positioning Systems

As highlighted by Küpper [9], a target cannot determine its location on its own. Instead, it relies on a distributed infrastructure that implements positioning methods to compute its position. This infrastructure typically consists of components like base stations, which are fixed points with known coordinates, and terminals, whose positions are initially unknown. Depending on the type of system, satellite, cellular or indoor, the base stations may include satellites, cellular towers, Wireless Local Area Network (WLAN) access points or tag readers. Terminals can range from mobile phones and laptops to vehicles and Radio Frequency Identification (RFID) tags, as illustrated in Figure 2.6.

In most cases, base stations either assist terminals in performing measurements or perform the measurements themselves. Additional components, such as databases and control units, may also be required for managing, processing and distributing positioning data. This section provides an overview of positioning systems, including Cellular Positioning System (CPS), GNSS, Wi-Fi and Bluetooth positioning systems, as well as INS, all of which rely on the positioning methods described in Subsection 2.1.

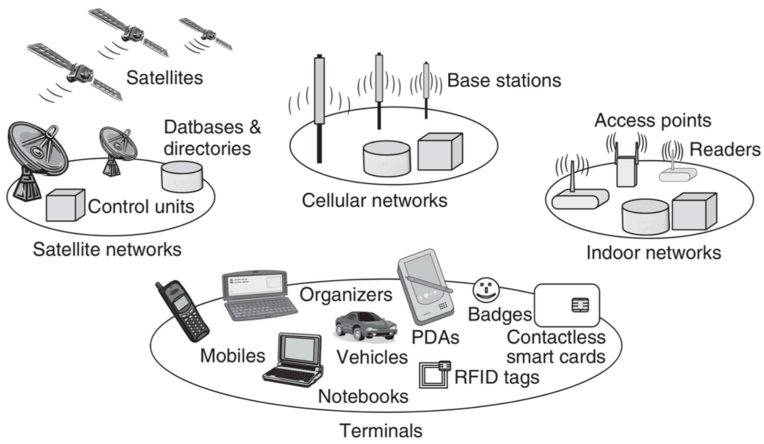


Figure 2.6: Infrastructures used in positioning [9]

2.2.1 Cellular Positioning Systems

A CPS uses the existing infrastructure of cellular mobile phone networks to estimate the location of a device by analyzing signals exchanged between terminals and base stations. Küpper [9] highlights that the first generation of Location-Based Services (LBS) mainly used proximity-based positioning methods, such as CoO, because they were simple to implement and required minimal changes to the existing network infrastructure. Another benefit of CoO is that it is network-based, so it doesn't need any special functionality on the mobile devices, meaning it works even on legacy terminals. However, it soon became clear that basic cell-based positioning was not accurate enough to meet the demands of many LBS. In 1996, the Federal Communications Commission (FCC) issued a mandate, known as Enhanced 911 (E-911), requiring mobile operators to determine emergency callers' locations and send them to Public Safety Answering Points (PSAPs). Küpper [9] further points out that this pushed the development of more accurate, lateration-based positioning methods into cellular networks.

Cellular networks were not originally designed for positioning and their effectiveness in determining location is constrained by factors such as cell size, signal propagation characteristics and network density. These challenges were particularly evident in earlier generations, where the cells were larger and infrastructure was sparse, especially in rural areas. However, as mobile networks evolved, the shift to higher frequency bands necessitated the deployment of smaller cells and denser infrastructure to maintain effective coverage, as noted by Mushiba [13]. Despite these advancements, cellular positioning is still often used as a complementary approach to GNSS, providing coverage where GNSS signals are unavailable or unreliable, such as indoors or in urban canyons.

2.2.2 Global Navigation Satellite Systems

A GNSS is a network of satellites providing real-time positioning and timing data to users worldwide and offering higher accuracy compared to cellular approaches. The most widely used system is the Global Positioning System (GPS) and was developed by the United States Department of Defense (DoD) in the 1970s. Küpper [9] states that GPS was initially intended for military purposes before being opened for civilian use and gaining mass adoption as low-cost GPS receivers became accessible in the 1990s. Other GNSS systems include GLONASS (Russia), Galileo (European Union) and BeiDou (China). These positioning systems consist of constellations of satellites orbiting Earth, which transmit signals to receivers that calculate their position using circular lateration and time measurements.

The constellation of GPS consists of 24 satellites distributed across six orbital planes with four satellites per orbit, each spaced 60 degrees apart. This ensures that at least four satellites are always within the receiver's line of sight from any location on Earth, according to Winters et al. [24]. Three satellites are used to calculate the receiver's 3D position, while the fourth is necessary for time synchronization between the receiver and the satellites. Since GPS applies Time of Arrival (ToA), which is based on circular lateration, accurate timing is important. Satellites are equipped with highly accurate atomic clocks, whereas receivers use lower-cost quartz-crystal clocks, which tend to drift over time, as highlighted by Küpper [9]. To compensate for this drift, the receiver treats the time offset as an unknown variable and incorporates it into its position calculations. This method assumes that the satellites are perfectly synchronized, allowing the receiver to adjust its own time to match them.

However, since satellite signals are relatively weak and struggle to penetrate walls and ceilings, a receiver must have a clear view of the sky to detect and process them. Urban canyons, mountain valleys, tunnels and dense forests can create radio shadows that block satellite signals as per Zhang et al. [26]. Furthermore, as Enge [2] and Ruegamer et al. [17] highlight, GNSS is vulnerable to jamming, where interference blocks satellite signals, and spoofing, where fake signals deceive receivers into showing incorrect positions or times.

2.2.3 Wi-Fi and Bluetooth Positioning Systems

While outdoor positioning systems such as GNSS work reliably in open environments, they struggle indoors due to the obstruction of signals caused by solid objects like walls. Hence, Indoor Positioning Systems (IPS) have been developed to provide locating and navigation capabilities within buildings using technologies like Wi-Fi and Bluetooth, both of which operate in the 2.4 GHz frequency band.

Wi-Fi-based positioning is widely used due to the ubiquity of Access Points (APs) in indoor spaces which therefore suggests that positioning comes at no extra cost. According to Küpper [9], Wi-Fi positioning relies on measuring the RSS, Signal-to-Noise Ratio (SNR) or proximity to beacons, which can either be passively received from access points in the downlink or actively transmitted by devices in the uplink. The two primary methods used in Wi-Fi positioning are lateration and fingerprinting which are further described in Subsections 2.1.2 and 2.1.4. Unlike optical technologies such as infrared, Wi-Fi does not require a direct line of sight which makes it scalable. Wi-Fi signals can also cover relatively long ranges of 50 to 100 meters, reducing the number of APs needed for coverage in large indoor spaces such as shopping malls or airports. Nonetheless, they are highly susceptible to multipath caused by obstacles like walls or structures and interferences from devices operating on the same frequency, which poses challenges for lateration as discussed by Grejner-Brzezinska and Kealy [6]. Fingerprinting, however, requires an extensive initial setup to map the environment along with periodic updates to account for changes in the signal landscape. Additionally, privacy concerns arise as users frequently connect to Wi-Fi networks which raises potential issues regarding tracking and data collection.

Bluetooth is a short-range wireless standard that typically determines a target's general location or proximity rather than absolute positions. Bluetooth Low Energy (BLE) uses beacons that continuously broadcast signals for nearby devices to receive or sensors, which detect transmissions from BLE devices. When only one beacon is available, the device's position is approximated by assuming the location of the detected beacon. If multiple beacons are detected, the position can be calculated using Received Signal Strength Indicator (RSSI) measurements and lateration which calculates distances from signal strength measurements, or AoA which determines the signal's direction. In addition, Bluetooth devices can form ad hoc networks, where they connect and exchange location information which allows them to detect their proximity to each other. However, as stated by Kolodziej and Hjelm [8], to connect the network to the physical environment, at least one terminal must have a known absolute position.

2.2.4 Inertial Navigation systems

The term "dead reckoning" comes from "deduced reckoning" which refers to estimating the current position by extrapolating from a previously known location which is adjusted by the observed direction of movement and velocity over time, as described by Grejner-Brzezinska and Kealy [6]. As discussed in Subsection 2.1.5, the core of any INS is the IMU that combines inertial sensors like accelerometers, gyroscopes and sometimes magnetometers and barometers depending on the use case. Odometers are separate sensors but can be integrated into navigation systems alongside an IMU.

Accelerometers measure linear acceleration relative to their own inertial frame. They contain a proof mass attached to a spring and when acceleration on the vertical axis occurs, inertia causes the mass to displace. This displacement is used to calculate acceleration, according to Werner [23]. However, this principle faces two challenges: the influence of the earth's gravitational pull and the measurements being limited to a single axis. Accelerometers cannot distinguish between gravitational and motion-based acceleration which is why other sensors are needed. To resolve the second problem, three accelerometers are arranged orthogonally to measure acceleration along three axes. Grejner-Brzezinska and Kealy [6] explain that in static conditions, the axis with the highest acceleration value corresponds to the direction of gravity, typically toward the ground. However, during motion the highest acceleration may appear along a different axis.

Gyroscopes measure angular velocity and their data can be combined with accelerometer data to separate movement from gravity as highlighted by Werner [23]. They work based on the conservation of angular momentum, which means a spinning object resists changes to its orientation. In a mechanical gyroscope, a rotating mass is mounted within a system of three concentric rings (gimbals) so it can rotate freely in all directions. As the device moves, the spinning mass maintains its original orientation relative to the surroundings and the rotation of the gimbals can be tracked which allows for the measurement of inertial rotational motion.

Magnetometers detect magnetic fields using Hall sensors which measure the voltage generated by the deflection of electrons when exposed to a magnetic field.

Barometers measure air pressure which can be used to estimate altitude changes or weather conditions based on the relationship between atmospheric pressure and height.

Odometers calculate the distance traveled by counting wheel rotations from a known starting point.

Once an initial position is established using methods like lateration or angulation, an INS operates independently and doesn't rely on external electromagnetic signals like radio signals. This self-contained nature makes inertial navigation highly robust against signal obstructions and manipulation, but they still are highly susceptible to errors. El-Sheimy [18] explains that position errors grow quadratically as a result of velocity errors that stem from biases or inaccuracies in accelerometer measurements during the initial integration process. He further points out that errors in gyroscopic data are even more impactful, as they first produce angular inaccuracies, which then propagate into velocity errors that grow quadratically and ultimately result in cubic error in position. These errors can accumulate over time due to the lack of external reference points. To mitigate this, inertial navigation is often only applied in short-term uses, such as to bridge periods when line-of-sight to satellites is obstructed or to refine position fixes determined by GNSS. By combining the last known location obtained via GNSS with measurements captured by an IMU chip, a vehicle's movement can still be tracked during GNSS outages, e.g. driving through a tunnel.

2.3 Geofencing

Shevchenko and Reips [19] define geofencing as the automated triggering of actions when "virtual boundaries around specific locations" are crossed. It works by defining a point on a map with latitude, longitude and a radius to form a circular boundary. When this boundary is entered or exited, an event can be triggered. iOS provides this feature through the Core Location framework, as stated on the Apple Developer Pages [38]. This framework mainly supports geofencing with circular regions, defined by a center point and radius, but does not natively support polygonal geofences. However, developers can implement polygonal geofencing through custom solutions or third-party libraries. For instance, the React Native Background Geolocation plugin enables polygonal geofencing by approximating polygons with a minimum enclosing circle, as documented by Transistor Software [51]. Over time, geofencing has become a valuable tool across various industries.

In marketing, businesses use geofencing to send location-based advertisements or discounts to customers near their stores. In workforce management, geofencing automates attendance tracking by allowing employees to clock in and out automatically upon entering or leaving designated work zones. In IoT, geofencing can enhance home automation. For example, smart devices can adjust lighting, heating or security systems when residents enter or leave their homes. Similarly, Kadam et al. [7] explore how geofencing combined with IoT can transform agriculture by helping farmers protect crops from wild animal intrusions through real-time alerts. Geofencing is also improving safety in educational settings. Takyiwa-Debrah [20] highlights how GPS-enabled wearable devices help track children's locations to reduce the risk of abduction in schools. In logistics, geofencing ensures vehicles adhere to designated routes and enables businesses to monitor fleet locations, as stated by Reclus and Drouard [15]. In healthcare, it assists caregivers monitor dementia patients by setting up safe zones and alerting them when patients move beyond these boundaries to ensure the patients' safety, as noted by Arora and Deswal [1].

Despite its advantages, geofencing raises privacy concerns, as highlighted by Greenwald [5]. Since it relies on continuous location tracking, users' sensitive data may be vulnerable to misuse or security breaches. This is particularly concerning when monitored regions overlap with personal or sensitive areas, such as homes or healthcare facilities.

2.4 Alerts on Smartphones

Alert types, trigger conditions and restrictions define how and when users are notified of specific app events, ensuring that alerts provide value without disrupting user control. The following sections outline the types of alerts available, the mechanisms that trigger them and the specific guidelines imposed by Apple to maintain a responsible alert experience towards users.

2.4.1 Alert Types

Smartphones offer different ways to alert users, including visual, sound and vibration cues. These alerts help confirm actions, share important information and improve user experience. Notifications are one of the most common types of alerts and can be split into local and push notifications. According to Apple's documentation [42][43], local notifications are created by the app itself to inform users of events, even when the app runs in the background. Push notifications are sent from a server and are often used for messages or reminders and updates. Another type of alert is haptic feedback, which uses vibrations to notify users physically. In iOS, tools like the UIImpactFeedbackGenerator provide vibrations for various levels of physical impact such as light, medium or heavy taps. The UISelectionFeedbackGenerator signals changes in selection, like scrolling through a picker, while the UINotificationFeedbackGenerator conveys success, warning or error notifications to clarify important feedback, as noted by the Apple Dev-Pages [40]. Sound feedback is also available and is more noticeable, making it effective for capturing users' attention. According to Apple's guidelines [32], iOS sounds are divided into system sounds, used for standard actions like errors or confirmations and custom sounds, often used in games or specific app events to create unique auditory experiences. For visual feedback, Apple's UIAlertController [31] provides pop-up alerts with titles, messages and action buttons, often used for critical alerts or confirmations to ensure user attention. Toasts, which are brief messages that appear and disappear without user interaction, are another form of visual feedback. While not native to iOS, third-party frameworks can enable toasts, offering a less intrusive way to share short information. Subtler visual cues, like progress indicators and badges, are also effective for showing ongoing tasks or highlighting unread notifications within app sections. With iOS 16, Apple introduced the Live Activities feature [37], which displays real-time updates on the lock screen or in the Dynamic Island. This feature keeps users informed about ongoing events, such as tracking a food delivery or monitoring live sports scores without needing to open the app.

2.4.2 Triggers

In iOS, alerts are not limited to user input, they can also be triggered by automated or event-based conditions. Apps, for example, can use scheduled timers to generate alerts or notifications at specific times for reminders, countdowns or periodic updates. This is often implemented using Swift's Timer class [44]. Alerts can also respond to app state changes, such as when an app transitions to the background, returns to the foreground or closes. For instance, an app may display a confirmation alert if the user attempts to leave a page with unsaved data or notify them of an important update. Device sensors like accelerometers and gyroscopes detect motion or orientation changes, which can also trigger alerts. Location-based triggers, which are central to this thesis, use geographical boundaries to deliver alerts when a user moves into or out of a specified region. According to Apple's Developer Pages [33, 34], the CoreLocation framework provides tools like CLLocationManager for managing these triggers, while CoreMotion supports motion event detection. Similarly, HealthKit [36] enables alerts based on health metrics, such as notifying users if their heart rate exceeds a specified threshold. To manage in-app notifications, Apple's NotificationCenter [39] is frequently used to broadcast data changes, enabling observers to update alerts or badges in real time. For notifications, iOS provides three main trigger types. UNTimeIntervalNotificationTrigger [47] schedules alerts after a specified time interval. UNCalendarNotificationTrigger [45] triggers notifications at specific dates or times, with options for recurring alerts and UNLocationNotificationTrigger [46] sends alerts when a user enters or leaves predefined geographical areas.

2.4.3 Restrictions in iOS

When it comes to responsible app development, Apple has many guidelines to maintain user control and prevent disruptive app behavior. Generally, apps are restricted from triggering alarms or alerts when not actively running in the foreground, with exceptions for specific cases like Voice over IP, music streaming and location-tracking apps. These applications can run background audio, but this capability is not intended for alarms and comes with strict guidelines to prevent misuse. For instance, while a music app may play in the background, an alarm app that attempts to exploit this mode to ring an alarm would be rejected from the App Store. Additionally, iOS does not provide any public API for third-party apps to set, modify or access alarms in the native Clock app, which is entirely separate and inaccessible for tasks like scheduling alarms or timers. When delivering notifications, apps must obtain user permission. Users can choose to allow, mute or disable notifications from each app, and Apple discourages excessive notifications, especially those lacking immediate or relevant value. Apps that send spammy notifications risk penalties or even removal from the App Store. Custom sounds for local notifications are limited to 30 seconds, if a sound exceeds this limit, iOS defaults to the standard notification sound, as discussed on the Apple Developer website [32]. Local notifications also respect Silent Mode and Do Not Disturb settings, meaning sounds will not play if these modes are active. According to Apple's documentation

[35], critical alerts are allowed to bypass these settings, but this permission is reserved for essential use cases, such as health monitoring or emergency alerts, not for general alarms or reminders. Location-based triggers, such as geofencing, also require explicit user permission. IOS [41] limits the frequency of location-based notifications and may throttle frequent triggers. Excessive location tracking can lead to App Store rejection unless it is essential to the app's functionality. Additionally, Apple states on its Dev Pages [38] that the use of geofencing is limited to a maximum of 20 active geofences per app to allow all apps to access condition monitoring and prevent excessive battery drain or performance issues, as these features rely on shared hardware resources.

2.5 Speech-to-Text

STT is a technology that turns spoken words into written text. It works by listening to audio and using voice recognition to understand and transcribe the words. The result is a text document that captures what was said. While speaking is the most natural and efficient way to communicate, audio recordings alone can be difficult to review or reuse quickly. This makes STT an important tool across many fields. For instance, it enables record-keeping of patient notes in healthcare or court proceedings in legal services and supports transcription of lectures in education. Furthermore, it is widely used in virtual assistants to process voice commands, in closed captioning to make videos accessible to broader audiences or in creating accessibility tools for individuals with hearing impairments.

Rabiner and Juang [14] explain that speech can be understood as an audio signal characterized by its frequency, amplitude and temporal patterns, which define the unique qualities of spoken language. They further emphasize that speech is made up of different components that work together to create meaning and structure. One of the most basic elements is phonemes, which are the smallest units of sound in a language like the "b" in "bat". These phonemes combine to form syllables, which are the building blocks of words, such as "bat" having one syllable and "better" having two. Beyond these, intonation patterns refer to the rise and fall of pitch in speech, which help convey emotions, emphasis or the difference between a question and a statement. According to Reetz [16], the process of turning speech into text depends on acoustic properties and linguistic understanding. Acoustic models focus on the sound itself, breaking down the audio into patterns like pitch, tone and duration to identify specific sounds or words. Meanwhile, language models go a step further by applying rules of grammar, vocabulary and sentence structure to make sense of those sounds in context. Together, these models ensure that the transcription is not only accurate in identifying the sounds but also meaningful and grammatically correct in written form.

However, STT faces several challenges that can impact its accuracy. Background noise, such as conversations, traffic or music can interfere with speech clarity and make transcription difficult. Regional accents and dialects add variability in pronunciation, which can lead to misinterpretations. Additionally, Zechner and Waibel [25] highlight that conversations with multiple speakers, especially when voices overlap, present difficulties in separating speech accurately. Lastly, spontaneous speech with its non-standard grammar, fillers like "um" or "uh" and frequent self-corrections further complicates transcription, as stated by Furiu et al. [4].

2.6 Conclusion

Chapter 3

Related Work

Public transportation navigation apps help commuters plan, track and complete their journeys more efficiently. While many apps provide route tracking, the challenge of knowing exactly when to get off at the right stop persists for passengers. Without clear alerts, passengers must constantly monitor their route which can be inconvenient or impractical, especially in unfamiliar areas.

This chapter examines three popular transit apps and how they handle user alerts for exiting public transportation. While all three apps provide detailed navigation and tracking, their approaches to stop notifications differ. Some rely on Live Activities for glanceable updates on the lock screen while others offer explicit stop alerts.

3.1 Google Maps

Google Maps was developed by Google and is a mapping service that offers route planning across different transportation modes including driving, walking, cycling and public transit. The platform has partnered up with numerous public transportation providers globally to integrate their data while also collecting information independently to facilitate trip planning for users. Google Maps provides real-time transportation information that allows users to view live arrivals for buses, metros and subway systems and alerts them to canceled routes. According to 9to5Google [27], Google Maps began testing the Live Activities feature on iPhones in 2023 to enable users to receive navigation updates directly on their lock screens and in the Dynamic Island. Live Activities offer glanceable directions, estimated time of arrival and upcoming turns without the need to unlock the device. However, only a limited number of users have reported seeing this feature appear intermittently while navigating with Google Maps. Figure 3.1 (left) shows Google Maps' route planning and detailed tracking in Figure 3.1 (middle). The Live Activity feature, also referred to as "Glanceable Directions" by Google, can be toggled in the settings but appears inconsistently with no known way to manually enable it. When it activates automatically, it appears as shown in Figure 3.1 (right).

Figure 3.1: Public transport route planning in the Google Maps app (left) and its Live Activity feature (right)

3.2 Citymapper

Citymapper was founded in 2011 by former Google employee Azmat Yusuf, who aimed to simplify navigation for London's public transport system, as highlighted by TechRound [52]. The app was initially launched as Busmapper and focused on London's bus network. Over time, it expanded to include the underground, trams and other transport modes and eventually became Citymapper. Today, the platform operates in over 100 cities worldwide including New York, Paris, Tokyo, Vienna and Sydney, according to their website [29]. Additionally, it supports entire regions like Scotland, the Basque Country and the Balearic Islands.

The app uses real-time data from multiple transit services to provide route planning and live tracking. It suggests the fastest and most convenient routes while considering delays and disruptions and offers users the possibility to receive notifications for service alerts. Furthermore, users are notified when approaching their designated stops if they enable this feature, as stated on their website [50]. Since iOS 16.1, Citymapper introduced Live Activities allowing users to follow the progress of their trip without unlocking their iPhone, as stated on Citymapper's Website [28]. Figure 3.2 (left) shows Citymapper's route planning. When "Go" is pressed in the detailed navigation in Figure 3.2 (middle) and the user is near the route, the Live Activity starts and displays real-time trip progress on the lock screen as in Figure 3.2 (right).

Moreover, the app allows users to track their CO_2 savings when using public transport, walking or cycling. It also provides the option to save favorite locations and label important places such as home and work for quick access. Additionally, calendar synchronization enables trip planning based on upcoming events and Siri integration allows users to receive commute updates via voice commands. Alongside public transport routes, Citymapper integrates with ride-hailing services like Uber and Gett to provide fare estimates.

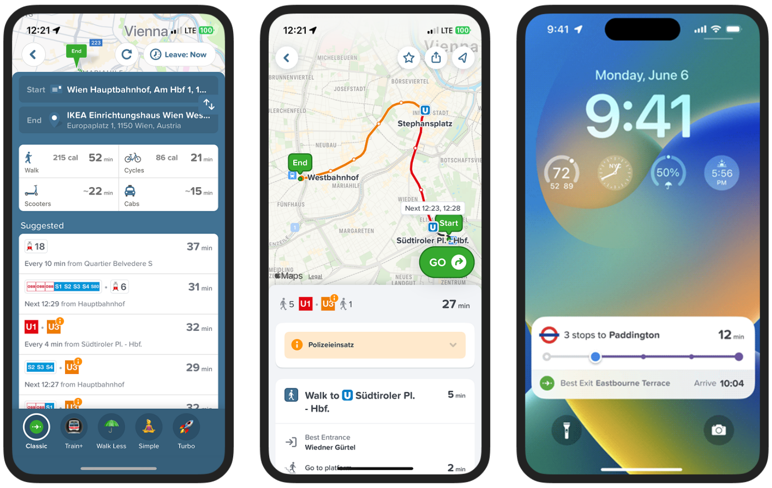


Figure 3.2: Public transport route planning in the Citymapper app (left) and its Live Activity feature (right) [28]

Citymapper has won Apple and Google's App of the Year awards multiple times and has over 50 million users worldwide. The company briefly operated a night bus network in London to test transport alternatives. In 2023, Citymapper was acquired by Via, a company specializing in digital transit infrastructure. This acquisition, as highlighted on Via's website [54], suggests a shift in focus from a standalone tool to integration within a larger mobility ecosystem. However, Via has stated that the acquisition will not impact the app's availability to its users and that they plan to expand its features and reach.

3.3 Moovit

Moovit is a public transportation navigation app launched in 2012 for iOS, Android and web browsers. It aims to help users navigate cities by integrating multiple forms of transport including public transit, shared bikes, ride-hailing services such as Uber and Lyft, car-sharing and scooters into a single application. Over the years, Moovit has grown significantly and now serves over 1.7 billion users in 3,500 cities across 112 countries and supports 45 languages, as stated on their website [49].

Moovit provides route planning and navigation as shown in in Figure 3.3 and displays nearby stops as well as arrival times for buses and trains. Live tracking of transit lines and approaching line notifications are available through a premium subscription. Other premium features include ad removal and traffic condition updates. Users can save favorite locations, set home and work labels for quick access and sync their Apple Calendar for trip planning based on upcoming events. Moovit also provides alerts for delays, service changes and reminders when to get off a bus or train. The app includes a Way Finder feature, which uses augmented reality to help locate bus or train stops. Simlarily, commuters can upload photos to assist other users in recognizing stops or report station conditions and suggest updates to transit data.

Moovit [49] describes its transit data repository as one of the largest in the world.

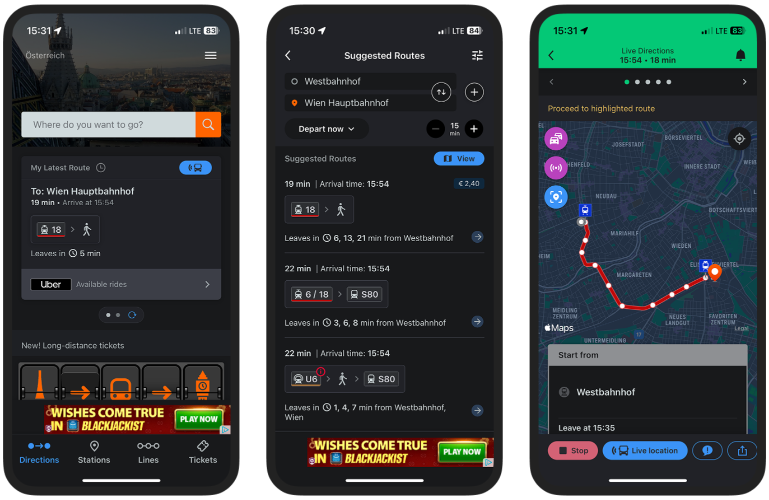


Figure 3.3: Public transport route planning and tracking in the Moovit app

The app collects data from over 7,500 public transit operators and 360 micro-mobility providers and includes more than 6 million stops and stations. However, a great part of its data collection comes from the Mooviter Community, a network of 875,000 local editors who contribute to updating transit information. This crowdsourced approach has helped Moovit expand to hundreds of new cities each year. Since 2022, Moovit has been part of Mobileye to advance autonomous transportation solutions and it has received several awards such as Google's "Best Local App" in 2016 and Apple's "Best Apps of 2017".

3.4 Conclusion

The apps reviewed in this chapter all provide route planning and tracking, but they differ in how they notify users when to exit public transportation. Google Maps offers Live Activities on iPhones which can show navigation updates on the lock screen. However, this feature is not consistently available and does not provide explicit stop notifications which means users must monitor their route on their own. Citymapper goes a step further by offering "Get Off Alerts" in the form of notifications, which must be activated by the user. Additionally, it provides Live Activities, similar to Google Maps, to help users track their journey progress. Moovit also includes stop notifications, but none of the apps offer vibrational alerts or sound alarms, as proposed in this thesis. The project aims to enhance user awareness by introducing a multi-step alert system, ensuring passengers receive both passive and active reminders when approaching their stop.

Chapter 4

Concept

4.1 Overview

4.2 Types of Alert Systems

4.2.1 Time-Based Mode

4.2.2 Station-Based Mode

4.2.3 Distance-Based Mode

Chapter 5

Implementation

5.1 System Architecture

5.2 UI Prototypes

5.3 Storing and Accessing User Data

5.4 iOS GeofenceManager using CLLocationManager

5.4.1 Permissions and Capabilities

5.4.2 Start Monitoring Region

Geofence Level 1: Notification

Geofence Level 2: Vibration

Geofence Level 3: Alarm

5.4.3 Stop Monitoring Region

5.5 Notification Management in iOS

5.5.1 Permissions and Capabilities

5.5.2 Scheduling Notifications Locally

5.6 Alarm Management in iOS

5.7 Displaying Routes on Map

5.8 Simulation of Routes

5.9 Voice-Controlled Alert Setup

5.9.1 Transcribing Speech to Text

5.9.2 Generating AI Response from Speech and Prompt

5.9.3 Scheduling Alert

Chapter 6

Conclusion

6.1 Resume

6.2 Next Steps

Appendix A

Acronyms

APs	Access Points
AoA	Angle of Arrival
BLE	Bluetooth Low Energy
CoO	Cell of Origin
CPS	Cellular Positioning System
DoA	Direction of Arrival
DoD	Department of Defense
E-911	Enhanced 911
FCC	Federal Communications Commission
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IoT	Internet of Things
IPS	Indoor Positioning Systems
LBS	Location-Based Services
PSAPs	Public Safety Answering Points
RFID	Radio Frequency Identification
RSS	Received Signal Strengths
RSSI	Received Signal Strength Indicator
SNR	Signal-to-Noise Ratio
STT	Speech-to-Text
ToA	Time of Arrival
TDoA	Time Difference of Arrival
UWB	Ultra-Wideband

Wi-Fi Wireless Fidelity

WLAN Wireless Local Area Network

References

Literature

[1] Parul Arora and Suman Deswal. “Location Tracking Mechanisms for Dementia Patients”. *Recent Patents on Engineering* 17.5 (2023), pp. 69–80 (cit. on p. 15).

[2] Per K Enge. “The Global Positioning System: Signals, Measurements, and Performance”. *International Journal of Wireless Information Networks* 1 (1994), pp. 83–105 (cit. on p. 12).

[3] Margareta Friman and Markus Fellesson. “Service Supply and Customer Satisfaction in Public Transportation: The Quality Paradox”. *Journal of Public Transportation* 12 (Dec. 2009). DOI: 10.5038/2375-0901.12.4.4 (cit. on p. 2).

[4] S. Furui et al. “Speech-to-text and speech-to-speech summarization of spontaneous speech”. *IEEE Transactions on Speech and Audio Processing* 12.4 (2004), pp. 401–408. DOI: 10.1109/TSA.2004.828699 (cit. on p. 17).

[5] Amie Greenwald et al. “An economically viable solution to geofencing for mass-market applications”. *Bell Labs Technical Journal* 16.2 (2011), pp. 21–38. DOI: 10.1002/bltj.20500 (cit. on p. 15).

[6] Dorota Grejner-Brzezinska and Allison Kealy. “Positioning and tracking approaches and technologies”. *Advanced Location-Based Technologies and Services* (2004), p. 1 (cit. on pp. 5, 13).

[7] Ashwini L Kadam, Hoon Lee, and Mintae Hwang. “Implementation of IoT application using geofencing technology for protecting crops from wild animals”. *Asia-pacific Journal of Convergent Research Interchange* 6.6 (2020), pp. 13–23 (cit. on p. 14).

[8] Krzysztof W Kolodziej and Johan Hjelm. *Local positioning systems: LBS applications and services.* CRC press, 2017 (cit. on pp. 6, 13).

[9] Axel Küpper. *Location-based services: fundamentals and operation.* John Wiley & Sons, 2005 (cit. on pp. 4–8, 10–12).

[10] Junggoo Lee et al. “An Efficient Localization Method Based on Adaptive Optimal Sensor Placement”. *International Journal of Distributed Sensor Networks* 2014 (Aug. 2014). DOI: 10.1155/2014/983618 (cit. on p. 5).

[11] Andreea-Valentina Militaru et al. "Positioning Key Elements for Increasing Localization Precision of the VRUs in 5G NR Environments". *Bulletin of the Polytechnic Institute of Iași. Electrical Engineering, Power Engineering, Electronics Section* 68 (Feb. 2024), pp. 57–76. DOI: 10.2478/bipie-2022-0022 (cit. on pp. 5, 8).

[12] Audronė Minelgaitė, Renata Dagiliūtė, and Genovaitė Liobikienė. "The Usage of Public Transport and Impact of Satisfaction in the European Union". *Sustainability* 12.21 (2020). DOI: 10.3390/su12219154 (cit. on pp. 1, 2).

[13] Abel Mushiba. *ADVANCED TELECOMMUNICATIONS TECHNOLOGIES - A Comprehensive Guide to GSM, UMTS, LTE, and 5G*. Jan. 2024 (cit. on p. 11).

[14] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition.* Prentice-Hall, Inc., 1993 (cit. on p. 17).

[15] Fabrice Reclus and Kristen Drouard. "Geofencing for fleet and freight management". In: *2009 9th International Conference on Intelligent Transport Systems Telecommunications, (ITST)*. 2009, pp. 353–356. DOI: 10.1109/ITST.2009.5399328 (cit. on p. 15).

[16] Henning Reetz. *Phonetics: Transcription, production, acoustics, and perception.* Wiley-Blackwell, 2020 (cit. on p. 17).

[17] Alexander Ruegamer, Dirk Kowalewski, et al. "Jamming and spoofing of gnss signals–an underestimated risk?!" *Proc. Wisdom Ages Challenges Modern World* 3 (2015), pp. 17–21 (cit. on p. 12).

[18] Naser El-Sheimy. "Inertial techniques and INS/DGPS integration". *Engo 623-Course Notes* (2006), pp. 170–182 (cit. on p. 14).

[19] Yury Shevchenko and Ulf-Dietrich Reips. "Geofencing in location-based behavioral research: Methodology, challenges, and implementation". *Behavior Research Methods* 56.7 (2024), pp. 6411–6439 (cit. on pp. 2, 14).

[20] Akosua Takyiwa-Debrah. "A geofence and IoT-based solution for child abduction in Ghanaian schools" (2023) (cit. on p. 14).

[21] E. G. R. Taylor. "Five Centuries of Dead Reckoning". *Journal of Navigation* 3.3 (1950), pp. 280–285. DOI: 10.1017/S0373463300034871 (cit. on p. 9).

[22] Xiaofeng Wei et al. "Positioning algorithm of MEMS pipeline inertial locator based on dead reckoning and information multiplexing". *Electronics* 11.18 (2022), p. 2931 (cit. on p. 9).

[23] Martin Werner. *Indoor Location-Based Services: Prerequisites and Foundations.* Springer, 2014 (cit. on pp. 7, 8, 10, 13).

[24] Phil Winters, Sean Barbeau, and Nevine Georggi. "Smart Phone Application to Influence Travel Behavior (TRAC-IT Phase 3)" () (cit. on p. 12).

[25] Klaus Zechner and Alex Waibel. "DiaSumm: flexible summarization of spontaneous dialogues in unrestricted domains". In: *Proceedings of the 18th Conference on Computational Linguistics - Volume 2*. Saarbrücken, Germany: Association for Computational Linguistics, 2000, pp. 968–974. DOI: 10.3115/992730.992786 (cit. on p. 17).

[26] Guohao Zhang et al. "Prediction on the Urban GNSS Measurement Uncertainty Based on Deep Learning Networks With Long Short-Term Memory". *IEEE Sensors Journal* 21.18 (2021), pp. 20563–20577. DOI: 10.1109/JSEN.2021.3098006 (cit. on p. 12).

Online sources

[27] 9to5Google. *Google Maps begins testing Live Activities on iPhone*. Accessed: 2025-02-08. 2024. URL: https://9to5google.com/2024/04/24/google-maps-live-activity-iphone-testing/ (cit. on p. 19).

[28] Citymapper. *Citymapper Introduces iOS Lock Screen Navigation*. Accessed: 2025-02-07. 2022. URL: https://citymapper.com/news/2553/citymapper-introduces-ios-lock-screen-navigation (cit. on pp. 20, 21).

[29] Citymapper. *Citymapper Supported Cities*. Accessed: 2025-02-08. 2025. URL: https://citymapper.com/cities (cit. on p. 20).

[30] GIS Geography. *Trilateration vs Triangulation - How GPS Receivers Work*. Accessed: 2025-01-05. URL: https://gisgeography.com/trilateration-triangulation-gps/ (cit. on p. 6).

[31] Apple Inc. *Alerts*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/design/human-interface-guidelines/alerts (cit. on p. 15).

[32] Apple Inc. *Audio Services: Play Alert Sound*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/audiotoolbox/1405202-audioservicesplayalertsound (cit. on pp. 15, 16).

[33] Apple Inc. *Core Location*. Accessed: 2024-11-03. 2024. URL: https://developer.apple.com/documentation/corelocation (cit. on p. 16).

[34] Apple Inc. *Core Motion*. Accessed: 2024-11-03. 2024. URL: https://developer.apple.com/documentation/coremotion (cit. on p. 16).

[35] Apple Inc. *Critical Alert Setting*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/usernotifications/unnotificationsettings/criticalalertsetting (cit. on p. 17).

[36] Apple Inc. *HealthKit*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/healthkit (cit. on p. 16).

[37] Apple Inc. *Live Activities*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/activitykit (cit. on p. 15).

[38] Apple Inc. *Monitoring the user's proximity to geographic regions*. Accessed: 2025-02-14. 2024. URL: https://developer.apple.com/documentation/corelocation/monitoring-the-user-s-proximity-to-geographic-regions (cit. on pp. 14, 17).

[39] Apple Inc. *Notification Center*. Accessed: 2024-11-03. 2024. URL: https://developer.apple.com/documentation/foundation/notificationcenter (cit. on p. 16).

[40] Apple Inc. *Playing Haptics*. Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/design/human-interface-guidelines/playing-haptics (cit. on p. 15).

[41] Apple Inc. *Region Monitoring.* Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/corelocation/monitoring_the_user_s_proximity_to_geographic_regions (cit. on p. 17).

[42] Apple Inc. *Scheduling a Notification Locally from Your App.* Accessed: 2024-11-02. 2024. URL: https://developer.apple.com/documentation/usernotifications/scheduling-a-notification-locally-from-your-app (cit. on p. 15).

[43] Apple Inc. *Setting Up a Remote Notification Server.* Accessed: 2024-11-02. 2024. URL: https://developer.apple.com/documentation/usernotifications/setting_up_a_remote_notification_server (cit. on p. 15).

[44] Apple Inc. *Timer.* Accessed: 2024-11-02. 2024. URL: https://developer.apple.com/documentation/foundation/timer (cit. on p. 16).

[45] Apple Inc. *UNCalendarNotificationTrigger.* Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/usernotifications/uncalendarnotificationtrigger (cit. on p. 16).

[46] Apple Inc. *UNLocationNotificationTrigger.* Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/usernotifications/unlocationnotificationtrigger (cit. on p. 16).

[47] Apple Inc. *UNTimeIntervalNotificationTrigger.* Accessed: 2024-11-04. 2024. URL: https://developer.apple.com/documentation/usernotifications/untimeintervalnotificationtrigger (cit. on p. 16).

[48] Massachusetts Institute of Technology. *Public Transportation.* Accessed: 2024-08-26. 2023. URL: https://climate.mit.edu/explainers/public-transportation (cit. on p. 1).

[49] Moovit Inc. *About Moovit.* Accessed: 2025-02-10. 2024. URL: https://moovit.com/about-us/ (cit. on p. 21).

[50] Moovit Inc. *Citymapper'S Get Off Alerts.* Accessed: 2025-02-10. 2024. URL: https://citymapper.com/news/823/get-off-alerts (cit. on p. 20).

[51] Transistor Software. *Geofence Interface — React Native Background Geolocation.* Accessed: 2025-02-14. 2024. URL: https://transistorsoft.github.io/react-native-background-geolocation/interfaces/geofence.html (cit. on p. 14).

[52] TechRound. *Startup Profile: Citymapper.* Accessed: 2025-02-08. 2023. URL: https://techround.co.uk/startups/startup-profile-citymapper/ (cit. on p. 20).

[53] United Nations. *Act Now: Transport.* Accessed: 2024-08-26. 2023. URL: https://www.un.org/en/actnow/transport#:~:text=If%20your%20destination%20is%20too,tons%20of%20carbon%20emissions%20reduced. (cit. on p. 1).

[54] Via. *Via Acquires Citymapper to Expand its End-to-End TransitTech Solution.* Accessed: 2025-02-08. 2023. URL: https://ridewithvia.com/news/via-acquires-citymapper-to-expand-its-end-to-end-transittech-solution (cit. on p. 21).

Check Final Print Size

— Check final print size! —

width = 100mm
height = 50mm

— Remove this page after printing! —