Location-Based Alerts for Passengers in Public Transportation

Antonia Stieger

BACHELORARBEIT

eingereicht am

Fachhochschul-Bachelorstudiengang

Mobile Computing

in Hagenberg

im Februar 2025

Advisor:

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, February 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. [16] highlight, expanding the adoption of public transport is essential for minimizing environmental damage. Specifically, the United Nations [49] 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 [4] 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. [16]. 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 [22]. 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, GNSS, and geofencing. The chapter also discusses alert systems on smartphones and their restrictions, as well as speech-to-text transcription and large language models. 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 II

This chapter provides an overview of foundational 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 [12], 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. 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 in systems like WGS-84. Further details on the different positioning techniques are provided in the following sections.

2.1.1 Proximity Sensing

Proximity sensing, as described by Küpper [12], is the simplest positioning method. It works by leveraging 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 IoT device. Due to the limited range of these pilot signals, the terminal's position is assumed to correspond to the location of the base station communicating with it. Militaru et al. [15] illustrate the concept of proximity sensing in Fig. 2.1.

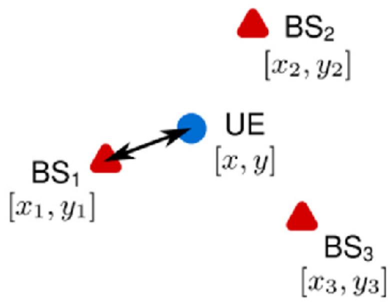


Figure 2.1: Proximity sensing [15]

In the words of Küpper [12], 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 [7] and Lee et al. [13]. Smaller cells allow for better location estimates, often reaching accuracy within 100 meters in urban areas where cell towers are densely placed. However, as the density of cells increases, the computational cost of this method also rises. 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 [7], 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. [13], 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. To determine a position fix, measurements from at least three stations are required. 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 [12]. This process, known as trilateration, utilizes the Pythagorean theorem for calculations and is illustrated in Fig. 3.2.

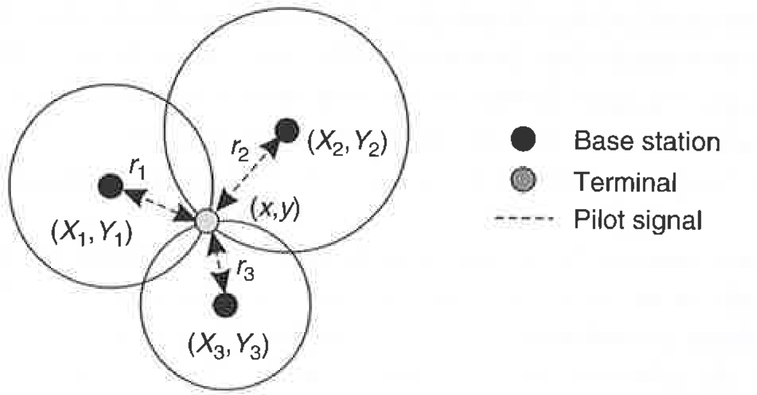


Figure 2.2: Circular lateration with three base stations [12]

In three-dimensional space, each distance measurement defines a sphere around a base station. Kolodziej and Hjelm [11] 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, ensuring a unique position fix. Additionally, the fourth base station aids in synchronizing clocks.

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, typically two base stations. Küpper [12] 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 part (a) of Fig. 3.3, 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 (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. A key advantage of hyperbolic lateration is that it only requires synchronization among the base stations' clocks, rather than between the stations and the target, simplifying timing coordination.

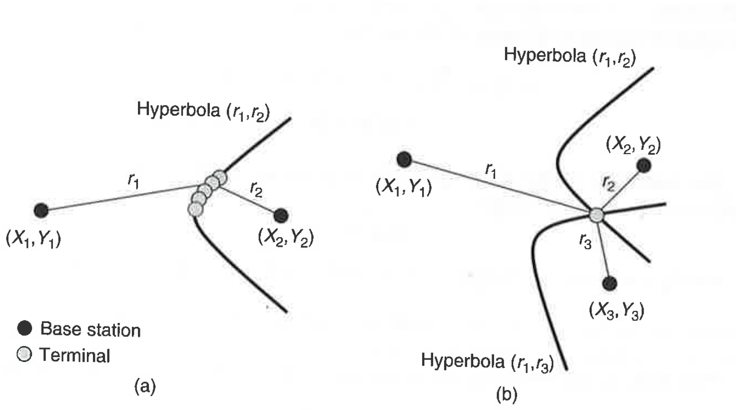


Figure 2.3: Hyperbolic lateration with two (a) and three (b) base stations [12]

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 [12] it is also called Angle of Arrival (AoA) or Direction of Arrival (DoA). Werner [26] explains that by using directional antennas or other angle-measuring equipment, the system identifies the direction of incoming signals from known base stations or access points. Once the angles from at least two base stations are known, the object's position can be triangulated by drawing lines along these angles, the point where the lines intersect reveals the target's location, as shown in Fig. 2.4.

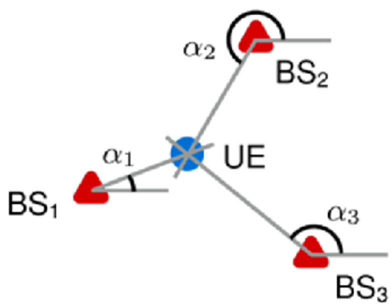


Figure 2.4: Triangulation [15]

In three-dimensional space, a third angle may be required for accurate positioning. Angulation offers an effective alternative to lateration methods, particularly in environments where measuring precise distances is challenging due to signal reflections or interference. However, it does require precise angle measurements, which can be sensitive to minor errors in directionality, especially at long distances.

2.1.4 Pattern Matching

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. When the calculated path is then compared to an initial point of reference, the position can be determined. This method represents a modern approach to dead reckoning, which was widely used by European explorers during the Middle Ages. They navigated using tools such as compasses, log lines and estimates of speed and time, as detailed by Taylor [24]. Figure 2.5 illustrates this process of approximating position through dead reckoning.

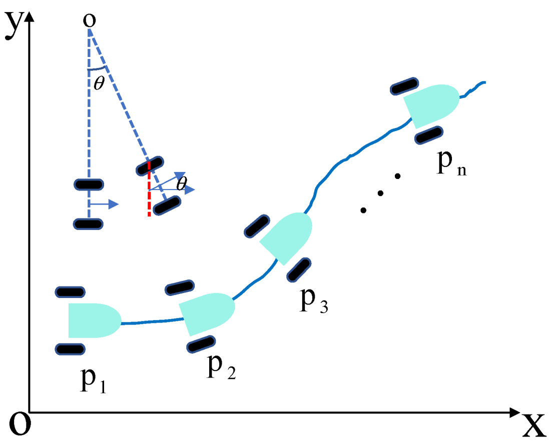


Figure 2.5: Dead Reckoning [25]

While inertial navigation shares these conceptual roots, it leverages modern technology to measure motion far more precisely. 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. 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 satellites or radio signals. This self-contained nature makes inertial navigation highly robust against signal obstructions and manipulation, unlike GNSS systems, which can suffer from signal disruptions in certain environments. Urban canyons, mountain valleys, and tunnels can create radio shadows that block satellite signals as per Zhang et al. [29]. Furthermore, as Enge [3] and Ruegamer et al. [20] highlight, GNSS is vulnerable to jamming, where interference blocks satellite signals, and spoofing, where fake signals deceive receivers into showing incorrect positions or times. In such scenarios, measurements from inertial sensors can be used as an enhancement for GNSS systems, but they still are highly susceptible to errors. El-Sheimy [21] 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 systems. 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.

2.2 Positioning Systems

2.2.1 Cellular Positioning

2.2.2 GNSS

2.2.3 WLAN and Bluetooth Positioning

2.2.4 Inertial Navigation Systems

2.2.5 Hybrid Systems

Chapter 3

Fundamentals

This chapter provides an overview of foundational concepts relevant to the thesis. It begins by reviewing various positioning methods 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.

3.1 Positioning

Positioning techniques are organized by the methods used to determine a location, with each approach offering specific benefits depending on the environment and application. The sections below introduce the core positioning methods developed over recent decades, including satellite-based positioning, cellular and WLAN-based positioning, proximity sensing, lateration and angulation. Each method has different principles for calculating location, whether through signal timing, distance, or angle measurements. The following sections will explore the fundamental concepts behind each technique, detailing how they calculate a position and discussing the advantages and limitations that make each method suitable for different scenarios.

3.1.1 Cellular or WLAN Positioning

Cellular and WLAN positioning are two different methods used to determine a device's location, especially when GPS is unavailable or needs to be supplemented. Cellular positioning relies on signals from nearby cell towers, using trilateration or multilateration techniques to calculate the device's distance from each tower. These techniques calculate position by measuring signal travel times or signal strength to multiple towers, which enables a triangulated location, as discussed by Küpper [12]. This method is particularly effective in rural areas where WLAN networks may be sparse, though the accuracy decreases in low-density tower areas due to the increased distances between towers. Meanwhile, Kolodziej and Hjelm [11] highlight that WLAN positioning estimates a device's location by analyzing signals from nearby access points. For this approach, a database or map of known WLAN access points with their geographic locations is necessary in order to compare signal strengths and determine relative proximity. Küpper

[12] states that because of signal variability indoors, WLAN positioning often works best in urban or indoor environments where multiple access points are available for cross-referencing.

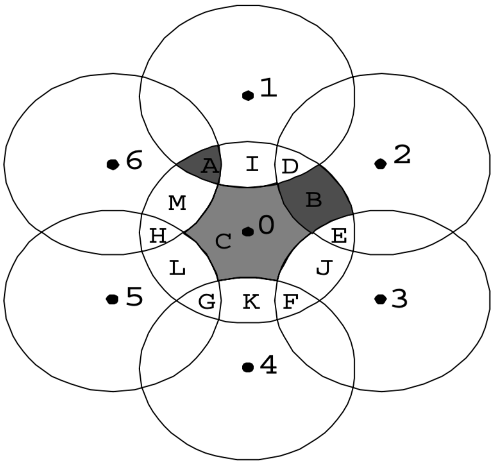
3.1.2 Proximity Sensing

Proximity sensing estimates a device's location by assessing signal interactions with nearby known transmitters, such as access points, beacons, or other devices. Depending on the infrastructure available, it can either use cellular or WLAN networks. As Küpper [12] notes, proximity sensing achieves high accuracy in locations with a high concentration of access points, such as indoor environments like malls and airports where WLAN is widely available. Outdoors, however, accuracy can vary significantly due to the reduced availability of proximity devices. Indoors, proximity sensors such as WLAN or Bluetooth beacons can achieve precision within a range of a few meters. Though this method provides great accuracy, a continuous connection to nearby beacons is required, resulting in high power consumption. Furthermore, it can sometimes cause delays in applications due to its reliance on the quality and consistency of signal strength, according to Mautz [14]. Proximity sensing methods are frequently combined with other positioning systems like GPS or inertial motion sensors to form stronger and more accurate hybrid systems. This combination improves both reliability and precision across diverse environments. In outdoor settings, GPS is often used as the primary source, with proximity sensing employed as a secondary technique. In the words of Küpper [12], proximity sensing in cellular networks is often referred to as Cell of Origin (CoO), Cell Global Identity (CGI), or Cell-ID and while it has reduced accuracy in both, cellular and WLAN networks, it is the easiest to implement.

3.1.3 Cell of Origin

Cell of Origin (CoO) is a basic mobile positioning technique used to identify the specific cell in which a device or caller is located. A "cell" is a geographic area within the cellular network, managed by a base station (BS), which 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 [7]. Smaller cells result in a more precise position, often reaching accuracy within 100 meters in urban areas where cell towers are densely placed. In rural areas, cell coverage may span several kilometers, reducing CoO accuracy.

Each cell may contain multiple overlapping service areas, meaning a device or mobile station (MS) could be within the range of multiple base stations, allowing it to detect and communicate with several base station IDs simultaneously. According to Jan, Chu et al. [8], an MS located in region A can receive signals from BSs 0, 1 and 6; in region B, it might connect to BSs 0 and 2; and in region C, it may only detect signals from BS 0. The area where a MS detects a unique combination of base station signals is called a localization region. For example, the cell of BS 0 contains 13 distinct localization regions (A-M). These corresponding localization regions are linked to specific sets of base station IDs in a centralized table or database, which is managed by a server. When a MS reports receiving signals from specific BSs, the server consults this table to determine the MS's location. For instance, if the MS detects signals from BSs 0, 1 and 6, the server can identify that the device is located in region A, allowing for accurate localization of the MS's position, as stated by Jan, Chu et al. [8]. This configuration is demonstrated in Fig. 3.1.



Regions	BSs Id
A	0, 1, 6
B	0, 2
C	0
D	0, 1, 2
E	0, 2, 3
F	0, 3, 4
G	0, 4, 5
H	0, 5, 6
I	0, 1
J	0, 3
K	0, 4
L	0, 5
M	0, 6

Figure 3.1: A layout of the cell-based positioning network [8]

CoO is often used in scenarios where approximate location data is sufficient. It's commonly used in emergency services to approximate a caller's location or for network optimization by telecom providers. Per Grejner-Brzezinska and Kealy [7], CoO is a power-efficient solution, as it only requires a single connection to the cell tower, making it accessible across cellular networks, even in low-power settings where more intensive GPS tracking might be impossible.

3.1.4 Lateration

Lateration calculates a target's location by using distance or distance differences from multiple reference stations, typically requiring at least three to achieve accuracy. These measurements, often referred to as "pseudoranges" due to inevitable errors, are based on the signal strength loss experienced by the beacon as it travels between the target and access points. 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 is a method used in positioning systems to determine a target's location by measuring distances to multiple base stations. 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 [12].

This process, known as trilateration, utilizes the Pythagorean theorem for calculations and is illustrated in Fig. 3.2.

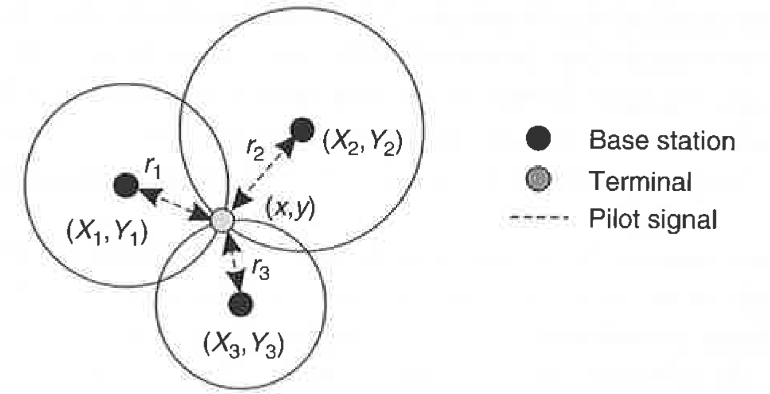


Figure 3.2: 2D Circular lateration with three base stations [12]

In three-dimensional space, each distance measurement defines a sphere around a base station. Kolodziej and Hjelm [11] 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, ensuring a unique position fix. Additionally, the fourth base station aids in synchronizing clocks, enhancing the accuracy of the positioning system.

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, typically two base stations. Küpper [12] 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 part (a) of Fig. 3.3, 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 (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. A key advantage of hyperbolic lateration is that it only requires synchronization among the base stations' clocks, rather than between the stations and the target, simplifying timing coordination.

3.1.5 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

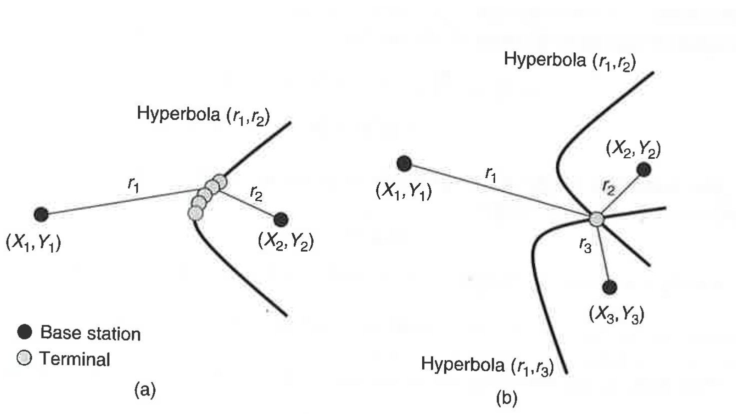


Figure 3.3: Hyperbolic lateration with two (a) and three (b) base stations [12]

measuring distances. According to Küpper [12] it is also called Angle of Arrival (AoA) or Direction of Arrival (DoA). Werner [26] explains that by using directional antennas or other angle-measuring equipment, the system identifies the direction of incoming signals from known base stations or access points. Once the angles from at least two base stations are known, the object's position can be triangulated by drawing lines along these angles, the point where the lines intersect reveals the target's location, as shown in Fig. 3.4.

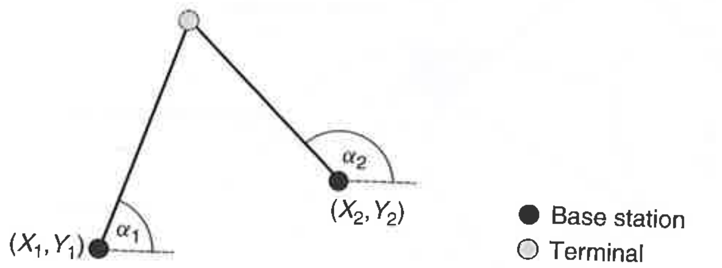


Figure 3.4: Angulation with two base stations [12]

In three-dimensional space, a third angle may be required for accurate positioning. Angulation offers an effective alternative to lateration methods, particularly in environments where measuring precise distances is challenging due to signal reflections or interference. However, it does require precise angle measurements, which can be sensitive to minor errors in directionality, especially at long distances.

3.1.6 GNSS

The Global Navigation Satellite System (GNSS) is an international network of satellites providing real-time positioning and timing data to various users worldwide. Originally developed to meet the demands for precise global navigation, GNSS aims to deliver accurate and reliable positioning for civil, commercial and scientific applications beyond military uses. GNSS systems, such as GPS (USA), GLONASS (Russia), Galileo (EU) and BeiDou (China), share a basic structure that includes space, control and user segments. Kaplan and Hegarty [10] highlight that the space segment consists of a constellation of satellites orbiting Earth. The control segment ensures that ground stations monitor the satellites and adjust their paths if needed. The user segment, however, says that users can access signals from at least four satellites at any given time in order to calculate an accurate 3D position through trilateration. GNSS satellites transmit signals containing information about their location and timing, allowing receivers on the ground to determine distance by calculating the time delay in signal arrival, which is known as time-of-arrival positioning, as stated by Enge [3]. High-accuracy GNSS is achieved with systems like Differential GNSS (DGPS), which uses reference stations on the ground to correct errors from atmospheric interference, satellite orbit irregularities and clock discrepancies.

3.2 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 or device functionality. 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 to users.

3.2.1 Alert Types

Smartphones provide users with various alert mechanisms, ranging from visual and auditory cues to haptic feedback. These feedback types are important for confirming actions, delivering critical information, or enhancing user experience. Notifications are among the most widely used feedback types and they can be divided into local and push notifications. According to Apple's documentation [41][42], local notifications are generated by the app itself to inform users of important events or reminders, even when the app is running in the background, while push notifications are server-sent, commonly used for real-time updates such as messaging notifications or breaking news alerts. Another significant alert type is haptic feedback, which alerts the user physically through vibrations. In iOS, the primary tools for haptic feedback include the UIImpactFeedbackGenerator, which provides feedback for different levels of physical impact, such as light, medium, or heavy taps. The UISelectionFeedbackGenerator is used to indicate changes in selection, such as scrolling through options in a picker and lastly the UINotificationFeedbackGenerator conveys success, warning, or error notifications to enhance the clarity of critical feedback, as highlighted by the Apple Dev-Pages [39]. In addition to haptics, sound feedback is available, which is more noticeable but also more effective in capturing users' attention. As per Apple's guidelines [31], sounds in iOS are divided into system sounds, which are used for standard actions like errors or confirmations and custom sounds, which are often used in games or specific app events, providing unique auditory experiences. For visual feedback, Apple's UIAlertController [30] offers pop-up alerts with titles, messages and action buttons and is commonly used for critical alerts or confirmations, providing a controlled interruption compared to sounds. Toasts are non-intrusive messages that appear briefly and disappear without needing user interaction. While they aren't native to iOS, third-party frameworks can enable them, offering a less intrusive way to communicate short information. Subtler forms of visual feedback, such as progress indicators and badges, are also effective for showing ongoing tasks or indicating unread notifications within app sections. With iOS 16, Apple introduced the Live Activities feature [37], which can be used to display real-time updates on the lock screen or in the Dynamic Island. This feature allows users to stay informed about ongoing events, such as tracking a food delivery or monitoring live sports scores, without needing to open the app.

3.2.2 Triggers

In iOS, alerts aren't only triggered by user input, they can also be set off by automated and event-based conditions. For instance, apps can use scheduled timers to trigger alerts or notifications at specific times, like for reminders, countdowns, or regular updates, easily implemented with Swift's Timer class [43]. Additionally, alerts can be triggered based on app state changes, such as entering the background, coming to the foreground, or closing. For example, an app may display a confirmation alert when the user tries to leave a page with unsaved data or notify them of an important update using UIApplicationDelegate [44]. Device sensors, like the accelerometer and gyroscope, can detect movement or orientation changes and trigger alerts accordingly. As noted by Apple [33][34], location-based triggers can also activate alerts or notifications when the user enters or exits a specific region, which is a key point in this thesis. For this purpose, CLLocationManager is useful for handling location-based triggers, while CoreMotion supports detecting motion events. Additionally, Apple's HealthKit [36] allows apps to create alerts based on user health data. For example, a notification can be sent if a user's heart rate exceeds a certain threshold. To manage in-app alerts, Apple [38] recommends using NotificationCenter is often used to broadcast data changes so that observers can display alerts or badges when specific data updates. For notifications, iOS provides three main trigger types. UNTimeIntervalNotificationTrigger [47] schedules a notification after a set time interval, while UNCalendarNotificationTrigger [45] delivers notifications at specific dates and times, either once or repeatedly. Most relevant to this thesis is UNLocationNotificationTrigger [46], which sends notifications when the user enters or exits a designated geographical area.

3.2.3 Restrictions in iOS

When it comes to responsible app development, Apple has many guidelines to maintain user control, ensure privacy and prevent disruptive app behavior. Generally, apps are restricted from triggering alarms or alerts when they are not actively running in the foreground, with exceptions for specific cases like VoIP (Voice over IP), music streaming and location-tracking apps. These apps can run background audio, but this is not intended for alarms and comes with strict guidelines to prevent misuse. For instance, per Apple's guidelines [32], while a music app may play in the background, an alarm app that tries to exploit this mode to ring an alarm would be rejected. This limitation makes it challenging to release an app that plays sounds solely for alerts, though this is not the goal of this thesis. 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 [31]. 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 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 function. Additionally, Apple [40] restricts the use of geofencing to a maximum of 20 active geofences per app to conserve battery life and optimize performance.

3.3 Geofencing

Shevchenko and Reips [22] define geofencing as the automated triggering of actions when "virtual boundaries around specific locations" are crossed. It relies on multiple location-based technologies, including GPS, Wi-Fi, and cellular networks, to establish and monitor these predefined geographic boundaries. Over time, it has become a useful tool with applications across different industries.

Geofencing enhances user experiences by providing personalized services, such as sending targeted advertisements or adjusting smart home settings when people enter or leave specific areas. It is also widely used in business operations, including logistics, healthcare, and workforce management, where it helps streamline processes and improve efficiency. However, as Greenwald [6] highlights, geofencing raises significant concerns, particularly regarding privacy. Since it relies on continuous location tracking, users' sensitive data may be vulnerable to misuse or security breaches, especially when the technology overlaps with personal or sensitive areas such as homes or healthcare facilities. Another common challenge is battery drainage, as location-tracking technologies like GPS consume substantial power, reducing device performance over time [1]. Furthermore, geofencing accuracy can be unreliable in urban areas with limited connectivity or physical obstructions like tall buildings. These inaccuracies, as discussed by Reclus and Drouard [18], can lead to false alerts or missed triggers, undermining user trust and the reliability of geofencing systems.

Initially designed for military and government use in the 1990s, geofencing has since expanded into a variety of domains due to advancements in GPS-enabled smartphones and mobile technologies [27]. In marketing, for instance, businesses use geofencing to send location-based advertisements or discounts to customers near their stores, creating personalized shopping experiences and increasing sales [6]. Workforce management benefits from geofencing by automating employee attendance tracking, where staff can clock in and out automatically upon entering or leaving work zones, reducing manual errors and administrative burdens [1]. Geofencing is also transforming home automation in the Internet of Things (IoT) domain. For example, smart devices can adjust lighting, heating, or security systems based on residents' proximity to their homes [1]. Similarly, it has proven valuable in agriculture, where geofencing combined with IoT helps protect crops from wild animal intrusions by sending farmers real-time alerts to minimize damage [9]. In educational settings, Takyiwa-Debrah [23] highlights how geofencing improves child safety by integrating GPS-enabled wearable devices, which provide real-time location tracking to prevent abductions in schools. In logistics, geofencing ensures vehicles adhere to designated routes and allows businesses to monitor fleet locations in real time, improving efficiency and security [18]. In healthcare, it helps caregivers monitor dementia patients by setting up safe zones and alerting them when patients move beyond these boundaries, enabling prompt interventions and ensuring patient safety [2].

3.4 Converting Speech to Text

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 speech-to-text an essential tool across many fields. For instance, it enables record-keeping of patient notes in healthcare or court proceedings in legal services, supports transcription of lectures in education, streamlines workflows in media and customer service by converting interviews or calls into text. Furthermore, it is widely used in virtual assistants to process voice commands, as well as in closed captioning to make videos accessible to broader audiences. It also plays a key role in creating accessibility tools for individuals with hearing impairments.

Rabiner and Juang [17] 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 [19], 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, speech-to-text 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 [28] highlight that conversations with multiple speakers, especially when voices overlap, present difficulties in separating speech accurately. 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. [5]. Lastly, technical vocabulary can be challenging for the system to recognize without prior training.

In Fig. 3.5 by Rabiner and Juang [17] the speech-to-text pipeline is illustrated by breaking down the process into different stages, starting with signal processing, where speech is captured and analyzed to differentiate between voiced, unvoiced and silent segments. Next, feature extraction transforms the raw audio into meaningful data points, which are then segmented into smaller units for easier processing. The labeling and sound merging steps classify individual sounds and combine them into coherent units, guided by sound classification and phonotactic rules. Sound classification identifies sounds, such as vowels or consonants, based on their unique acoustic patterns, while phonotactic rules ensure that recognized sounds follow the natural language patterns, enhancing transcription accuracy. Word verification and sentence verification ensure the transcription is contextually meaningful by leveraging lexical access and language models. Throughout the pipeline, knowledge sources such as phonotactic rules and linguistic models play an important role in improving the transcription and producing a recognized utterance.

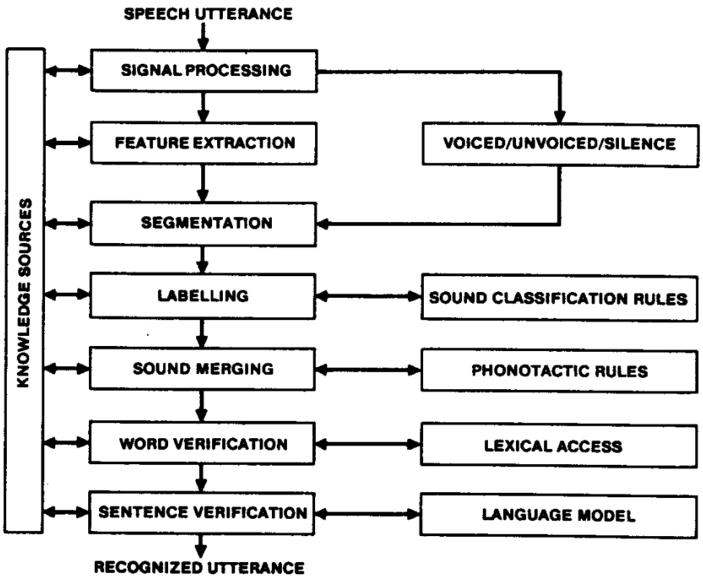


Figure 3.5: Knowledge integration in speech recognition [17]

3.5 Large Language Models

Chapter 4

Related Work

4.1 Google Maps

4.2 Citymapper

4.3 Moovit

4.4 Transit

Chapter 5

Concept

5.1 Overview

5.2 Types of Alert Systems

5.2.1 Time-Based Mode

5.2.2 Station-Based Mode

5.2.3 Distance-Based Mode

Chapter 6

Implementation

6.1 System Architecture

6.2 UI Prototypes

6.3 Storing and Accessing User Data

6.4 iOS GeofenceManager using CLLocationManager

6.4.1 Permissions and Capabilities

6.4.2 Start Monitoring Region

Geofence Level 1: Notification

Geofence Level 2: Vibration

Geofence Level 3: Alarm

6.4.3 Stop Monitoring Region

6.5 Notification Management in iOS

6.5.1 Permissions and Capabilities

6.5.2 Scheduling Notifications Locally

6.6 Alarm Management in iOS

6.7 Displaying Routes on Map

6.8 Simulation of Routes

6.9 Voice-Controlled Alert Setup

6.9.1 Transcribing Speech to Text

6.9.2 Generating AI Response from Speech and Prompt

6.9.3 Scheduling Alert

Chapter 7

Conclusion

7.1 Resume

7.2 Next Steps

References

Literature

[1] S Amudha et al. “Enhancement of smart home automation system using secure geofence concept in IoT”. In: *AIP Conference Proceedings*. Vol. 2112. 1. AIP Publishing. 2019 (cit. on pp. 17, 18).

[2] Parul Arora and Suman Deswal. “Location Tracking Mechanisms for Dementia Patients”. *Recent Patents on Engineering* 17.5 (2023), pp. 69–80 (cit. on p. 18).

[3] Per K Enge. “The Global Positioning System: Signals, Measurements, and Performance”. *International Journal of Wireless Information Networks* 1 (1994), pp. 83–105 (cit. on pp. 8, 15).

[4] 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).

[5] 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. 19).

[6] 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. 17).

[7] 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, 11, 12).

[8] Rong-Hong Jan, Hung-Chi Chu, and Yi-Fang Lee. “Improving the accuracy of cell-based positioning for wireless networks”. *Computer Networks* 46.6 (2004), pp. 817–827 (cit. on pp. 11, 12).

[9] 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. 18).

[10] Elliott D Kaplan and Christopher Hegarty. *Understanding GPS/GNSS: Principles and Applications*. Artech house, 2017 (cit. on p. 15).

[11] Krzysztof W Kolodziej and Johan Hjelm. *Local positioning systems: LBS applications and services*. CRC press, 2017 (cit. on pp. 6, 10, 13).

[12] Axel Küpper. *Location-based services: fundamentals and operation.* John Wiley & Sons, 2005 (cit. on pp. 4–7, 10–14).

[13] 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).

[14] Rainer Mautz. “Indoor positioning technologies” (2012) (cit. on p. 11).

[15] 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, 7).

[16] 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).

[17] Lawrence Rabiner and Biing-Hwang Juang. *Fundamentals of speech recognition.* Prentice-Hall, Inc., 1993 (cit. on pp. 18, 19).

[18] 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 pp. 17, 18).

[19] Henning Reetz. *Phonetics: Transcription, production, acoustics, and perception.* Wiley-Blackwell, 2020 (cit. on p. 18).

[20] 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. 8).

[21] Naser El-Sheimy. “Inertial techniques and INS/DGPS integration”. *Engo 623-Course Notes* (2006), pp. 170–182 (cit. on p. 8).

[22] 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, 17).

[23] Akosua Takyiwa-Debrah. “A geofence and IoT-based solution for child abduction in Ghanaian schools” (2023) (cit. on p. 18).

[24] 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. 8).

[25] 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. 8).

[26] Martin Werner. *Indoor Location-Based Services: Prerequisites and Foundations.* Springer, 2014 (cit. on pp. 7, 14).

[27] Tegg Westbrook. “The Global Positioning System and Military Jamming: The geographies of electronic warfare”. *Journal of Strategic Security* 12 (July 2019), pp. 1–16. DOI: 10.5038/1944-0472.12.2.1720 (cit. on p. 17).

[28] 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. 19).

[29] 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. 8).

Online sources

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

[31] 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, 17).

[32] Apple Inc. *Configuring Your App for Media Playback*. Accessed: 2024-11-02. 2024. URL: https://developer.apple.com/documentation/avfoundation/media_playback/configuring_your_app_for_media_playback (cit. on p. 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. 16).

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

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

[40] 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).

[41] 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).

[42] 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).

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

[44] Apple Inc. *UIApplicationDelegate*. Accessed: 2024-11-02. 2024. URL: https://developer.apple.com/documentation/uikit/uiapplicationdelegate (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] 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).

Check Final Print Size

— Check final print size! —

width = 100mm
height = 50mm

— Remove this page after printing! —