NaviSense Final Report

HA001, Honors Academy Bachelor (2024-25)

Team NaviSense

Full Name	Student ID
Olaf Hermans	1902555
Teodor Krajč	1966014
Alicja Napieralska	1986024
Lucas Iovdij	2005107
Yash Rana	1815857

Coach: Yannis Koutsioukis

Eindhoven, 25/05/2025

Contents

1 Introduction **2**
- 1.1 From Interviews to Implementation . . . 2
- 1.2 Problem Description . . . 1

2 Initial Research, Planning, and Approach **2**

3 Exploring the LiDAR **4**

4 LiDAR Data Game **6**

5 Naext Visit **7**

6 Digital Twin Environment **8**
- 6.1 3D environment . . . 8
- 6.2 Feedback to users . . . 8
- 6.3 Input device . . . 9
- 6.4 Communication between the device and the environment . . . 10
- 6.5 Output to the user . . . 11
- 6.6 Further work . . . 12

7 Computer Vision Aid **13**
- 7.1 Introduction . . . 13
- 7.2 Existing work . . . 13
- 7.3 Processing and output . . . 14
- 7.4 Aspects for improvement . . . 14

8 Conclusion **16**

9 Discussion **17**
- 9.1 Digital Twin . . . 17
- 9.2 Feedback Methods . . . 18
- 9.3 Cooperation with Other Companies . . . 19

10 References **21**

A Individual Contributions **22**
- A.1 Lucas Iovdij . . . 22
- A.2 Yash Rana . . . 23
- A.3 Alicja Napieralska . . . 24
- A.4 Olaf Hermans . . . 25
- A.5 Teodor Krajč . . . 26

1 | Introduction

Public buildings, especially complex spaces such as hospitals, often create significant navigation challenges for people with visual impairments. These environments are typically designed with sighted people in mind and often lack clear tactile or audio feedback that could help individuals with visual impairments find their way. The heavy reliance on visual cues—such as signs, colors, and symbols—makes it particularly difficult for those who are hard of seeing to navigate confidently and independently, often forcing them to rely on help from others. While the lack of accessibility in public spaces should be addressed at an institutional level, until that happens, there is a pressing need for navigation systems that address these issues. This lack of accessibility leaves many people with visual impairments feeling isolated and frustrated in environments that should be accessible to everyone.

The NaviSense project was created in response to these challenges, with the goal of developing a non-intrusive, intuitive indoor navigation system that provides real-time directional feedback to help people with visual impairments navigate complex public spaces independently and safely.

1.1 | From Interviews to Implementation

Project NaviSense began last year, focusing on creating a practical implementation of this vision through user-centered design and testing. While we initially planned to build a working prototype within the first year, it quickly became clear that creating an effective technical solution would require a much deeper understanding of what end users actually need, how they interact with their environment, and what challenges they face day-to-day. As a result, we decided to focus last year's work on foundational research and validating the problem.

To build this foundation, we started with a thorough exploration of existing assistive technologies. We looked at tools like the eZwayZ app, the WeWalk smart cane, Envision glasses, and the discontinued Sunu wristband. We also visited the muZIEum in Nijmegen to get firsthand experience of navigating 'with visual impairment' and went to innovation events to connect with affected individuals and experts in the field. All this work helped us understand the problems with current solutions and identify gaps that no existing tool was properly addressing, but most importantly, it made us realize the need to conduct a qualitative study to hear directly from those who face these challenges daily and rely on existing solutions.

Thus, the core of last year's work was twenty detailed interviews with people with visual impairments from the Netherlands, Spain, the UK, and Peru. The results gave us an overview of user experiences and expectations, and created the basis for this years' major design choices carrying forward. Namely, we found that 85% of participants use a white cane as their primary mobility aid. This showed us how important it is to design solutions that work alongside, rather than replace, this main tool. Additionally, 60% reported using mobile apps for tasks such as text recognition or navigation, which suggested strong familiarity and comfort with smartphones, making them a practical platform for any assistive solution. However, 20% of participants pointed out the inconvenience of having to hold their phone while also managing a cane or bag. Regarding feedback modes, 50% of participants preferred vibration and 40% used audio-based tools, but many warned against any audio system that might block sounds coming from their surroundings, which are so crucial for navigation when sight cannot be a primary sense. This concern, mentioned explicitly by 15% of participants, indicates prioritising subtle, peripheral haptic or spatial audio cues that preserve their ability to orient in space. Participants also detailed the most significant challenges they face when navigating indoor spaces. The most common issue that was reported by 55% of respondents—was the lack of autonomy due to the need of dependence on others for assistance in unfamiliar buildings. This reinforced the core design requirement that any tool we develop must enable users to navigate independently without external help. 25% cited orientation difficulties, such as lack of reference points or clear instructions, which points toward developing dynamic, real-time directional feedback that adapts as the user moves and is able to recognize its environment. Another 25% criticized the overreliance on visual cues, especially in hospitals, and 20% noted that inconsistent or non-tactile markings on doors and hallways. The issues mentioned suggest focus on digital alternatives to visual information—such as audio or haptic guidance, integrated directly into the user's personal device. Moreover, 10% of participants pointed to the inability to read queue numbers on screens as a recurring barrier, pointing to a need for a solution that would be able to dynamically carry context-specific and location-specific information to the user.

Aside from identifying problems that inspired our design standards, the interviews also helped us to create criteria for the implementation of our future tool. As 20% of the participants expressed enthusiasm for new technologies and a willingness to adopt innovative solutions for aids, we could validate our concept of digital feedback systems going beyond traditional navigation methods. 15% pointed to the importance of simplicity and discretion, voicing a preference for tools that do not attract attention or lead to cognitive overload, guiding us to avoid big or attention-drawing hardware. Multiple participants asked for clear, intuitive instructions like "left" or "right," rather than technical coordinates, which should influence how we would convey the feedback to the user. 10% also noted that tactile reference points and straightforward routes are crucial for ease of movement, which we should keep in mind to make our feedback design to support natural, clear navigation. This year, we build directly on last year's insights, using the interviews as the set of requirements for turning our research into implementation. Our goal now is to build working prototypes, test our feedback systems, and validate our approach, for now in a digital environment.

1.2 | Problem Description

The problem we wanted to address this academic year is how to provide accurate, real-time indoor navigation to support people with visual impairments, by using a solution that is non-intrusive, intuitive, and fits seamlessly into their existing habits. Building on last year's findings, we focused specifically on designing a system that allows independent orientation and movement in complex public buildings, such as hospitals, without relying on visual cues or needing help from others. Our goal was not to completely rethink mobility aids from scratch, but to work alongside familiar tools like white canes and smartphones with a digital layer that offers directional feedback. To do this, we wanted to explore whether the built-in sensors of mobile phones, which are already widely used by our target group, could work as a reliable basis tool for navigation, and figure out how to deliver feedback, avoid cognitive overload, and respect the user's sensory environment.

2 | Initial Research, Planning, and Approach

Since most of our team was completely new to the project, we first analyzed the problem. Even though one team member had previous experience with both the topic and the project, the rest were not as aware of the user preferences and considerations that surround designing solutions for people with visual impairments. Essentially, while we understood the problem intuitively, there was little formal framing of the problem itself, so we focused on understanding it deeply through discussions. Thus, we first defined a problem statement, then defined the main user, and researched existing technologies and efforts being done in this area.

During this initial analysis, we soon realized that we did not have the slightest idea of what tech to use as a facilitator for the solution. Although we had some initial research conducted by the previous year's team, more specifically on Ultra Wide Band technology and a little on Computer Vision, it only scratched the surface of its specifications such as price, margin of erroneous positioning, and the like. We did not understand the trade-offs of using these two technologies entirely, nor how they could be utilized in the implementation of a solution. Hence we looked a bit further into the different positioning techniques that were currently available; more precisely, we conducted further research into LiDAR, UwB, Computer Vision, Wi-Fi positioning, and BLE.

Additionally, an early sparring session with one Sjoerd of TeamHART, who is also looking for methods to assist people with visual impairments localizing themselves in their environment, helped us understand that a beacon solution was not entirely feasible for our project. He let us know that beacons, such as UwB and BLEs, are extremely expensive, and time-consuming to set up both for development and actual implementation, which signaled to us that such a solution would not address our problem in a responsible manner, as such a solution would not be attractive to many administrations of public spaces. Thus, we singled out UwBs, BLEs and other beacon solutions quite early on. Another idea we discarded early on was Wi-Fi positioning, which aimed to deliver indoor positioning based on the direction and location of different routers with respect to the position of the positioning device by understanding the direction of multiple Wi-Fi signals. The issue we found with Wi-Fi was that the accuracy of its positioning had an incredibly large margin of error, usually from 1-15 meters [1]. This meant that this positioning system would frequently mischaracterize the position of the user to such an extent that it could misplace the user in an entirely different room without a clear mechanism for fixing this placement reliably within a short span of time.

With this in mind, we would further research the techniques associated with both LiDAR [16] and Computer Vision [10]. However, this proved more challenging than anticipated as most resources we found still for LiDAR did not directly address our specific needs which we admittedly preferred to work with initially (PS: explained in the next paragraph). Though we found specific techniques which we could utilize in LiDAR development such as SLAM [6], most of these methods were used in embedded systems more on the scale of robots, which did not yield well to portable personal devices that we were aiming to develop as per the designs of the prior NaviSense team.

Still, inspired by our PDPs and the desire to work with physical systems specifically and get hands-on experience, we were not entirely dissuaded initially by challenges posed by LiDAR specifically and the fact that we could not find enough satisfactory material. Instead, the team desired to indulge in a more hands-on, hardware-oriented, prototype driven approach to the problem, so that we could gain first hand experience with such tech, and gain insight as to what components were optimal to use in addressing our problem.

So our approach evolved to the desire of testing the two mentioned technologies. We decided first to explore LiDAR as it was not necessarily clear which technology was better and as we were excited to work specifically with an embedded system to hone our skills in developing them. We then made a rough roadmap and set up goalposts for the development and went over it with our client. Notably, we made our roadmap very early on, and with arguably an incomplete understanding of all the required knowledge which caused problems later on during our project. We will address this in subsequent sections of this report, but importantly, in terms of sustainable development, we did not brainstorm enough through the possibilities which the project could take, and we did not anticipate the issues we would encounter later on. Although, we wanted to explore the technology of LiDAR in positioning, we did not have a clear plan as to how our particular LiDAR of choice could have been utilised in a design for an actual solution of the problem; for instance a guiding robot which would have the LiDAR situated on top of it. Looking back, this was something we should have given more attention to early on even though we felt like we were wasting time researching, as we eventually came to the point where we needed to answer these design questions. Doing so would have avoided this issue of a lack of framing which stalled our development at one point and caused some misunderstanding in the work needed to be done.

Still our initial plan and approach for the upcoming period was to figure out the workings of the LiDAR. We thus went ahead with our initial plan, and sought to set up the LiDAR.

3 | Exploring the LiDAR

Our first development step consisted of setting up the LiDAR. We first purchased the YDLiDAR X4 PRO sensor[2] and tried connecting it to an ESP microcontroller. This decision was strategic as several team members had prior experience with using an ESP in the course of embedded systems. We hoped the ESP would act as the mediator between the LiDAR collecting the data and an external computer that would compile it into a reading of the environment around the LiDAR and hence the user, while it also allowed us to inspect and analyze the data the LiDAR generated.

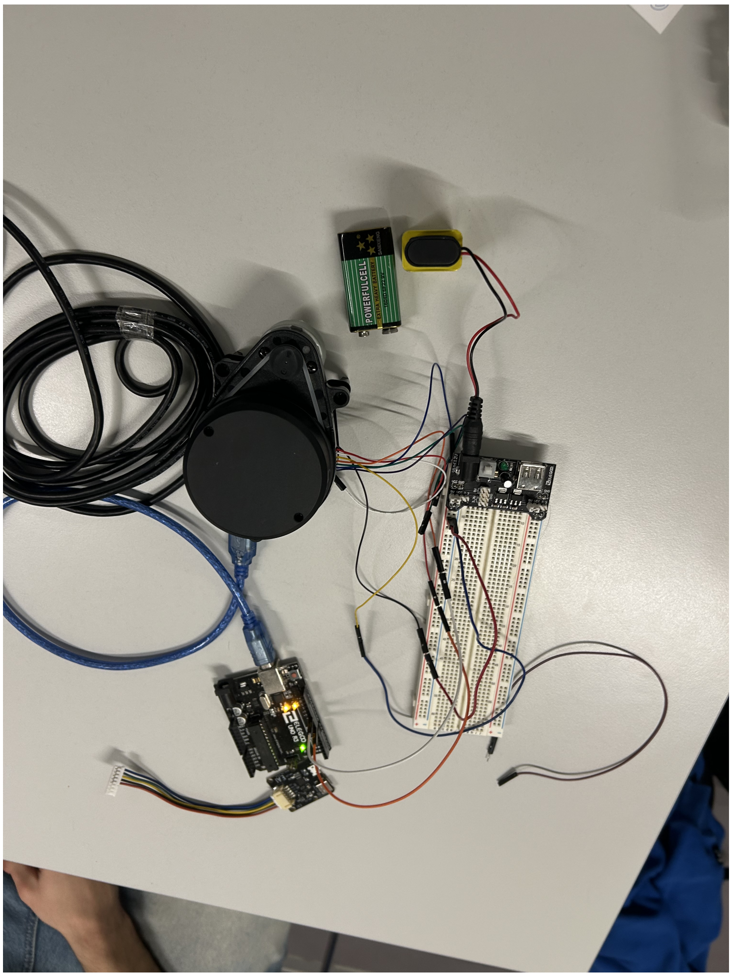


Figure 3.1: Our LiDAR Starter Kit

Unfortunately, we had a lot of issues with setting it up to work the way it is intended. Firstly, we encountered a few different C++ Arduino libraries [3][4] that were said to work with the LiDAR we bought. This turned out to not be the case. Most of these libraries did not work and the process of finding one that did turned into a game of guessing. This compounds with the fact that it was not necessarily clear to us that these libraries did not work as intended, and hence caused us to doubt the most basic of factors such as the wiring of the LiDAR. Secondly, this revealed a bigger issue however, and that was that the LiDAR and the ESP being so hands-on, i.e that actual physical interfacing with it was required to test out its outputs and whether it is working correctly, as well as the fact that it was attachable only to a single computer at a time meant that basically only one to two people could work on it together at the same time. This fact of the matter did not go well with our personal developmental goals, as whatever hands-on experience and skill development we were doing, most of it was sectioned off only to a small subset of the team.

When we finally found the correct library [8] that actually enabled us to work with the ESP and began working on transforming the LiDAR output into sensible data, our team only grew more dissatisfied with LiDAR. As we went along with development and designing the potential solution, more and more practical issues associated with our specific LiDAR. One such issue was a dimension restriction. Because we were constrained by a budget, the LiDAR we purchased only allowed us to collect data on the account of two dimensions. This meant that we were unable to properly map out the environment with the user in mind. Since lacking the ability to scan three dimensions meant that we could not detect objects properly, due to the lack of depth at which these were visible was not discernible based on this data alone, this was not the best for our design as we were unable to examine potential navigational issues for the people with visual impairments that come abound from having objects obstruct mappings of the indoor environments. Another such issue was the limited portability of the LiDAR. As our LiDAR was so heavy, so big, and required an incredible amount of stability and time to produce the scans of the environment we did not really have an idea as to how this LiDAR would yield to an actual wearable or usable device that could genuinely be used by the user. The closest we could genuinely get was some form of a helmet that used a platform to safely station the LiDAR on the top of the user's head, which is as ridiculous of an idea as it sounds. Although we had gained a better grasp of the underlying technology at this point, it became clear to us that we, ironically, lacked vision for how exactly we were going to utilize LiDAR in our actual design. In addition, due to the overall issues with the particular LiDAR we used mentioned above, as well as the incredible slowness with which the LiDAR produced map readings, it was doubly clear to us that this particular LiDAR sensor would not work the way we intended, i.e. as an immediate localization mechanism in a pre-mapped space. More specifically, given that our particular LiDAR was able to scan the environment at 6 Hz scan frequency, completing six full 360° rotations per second, and with an angular resolution of 0.43°, it can yield approximately 837 data points ($360°/0.43° \approx 837$); this results in about 5,022 points per second (6 rotations · 837 points). Unsurprisingly, this in the end yields a relatively slow scan speed compared to higher-end LiDAR systems that can produce hundreds of thousands to millions of points per second; these could be theoretically used to "immediately" locate oneself in a pre-mapped space but such is not the case with the YDLiDAR X4. Given that these higher-end, more robust, compact and quality LiDAR sensors were simply out of the scope of our budget we essentially realized that we would be unable, at this capacity, unable to develop a solid, reliable design to solve our defined problem with. Thus, the ideas which would incorporate our particular LiDAR model would be impractical and unfeasible.

Nevertheless, we aimed to utilize and visualize the data from the LiDAR at least in some manner to ground our current progress in usable, meaningful research output for future purposes of the project. As it was in line with our personal development interests, we opted to develop a visualization method that could be used to visualize LiDAR data even using different sensors (components) that use the underlying LiDAR technology. This allowed us to utilize the LiDAR we had already purchased, letting us develop a universal visualization tool that took the measurements of the distances and angles produced by generally any LiDAR and transformed those into a mapping of an environment around the LIDAR. We built the tool using the Matplotlib library with Python at first. As for the computation of the visualizations, we made mappings by taking raw data from the LiDAR, the angle and distance measures, to generate a 2D point cloud - a mass of points representing an arbitrary body - around an origin point by placing each individual point by its distance away from the origin at the measured angle. Our visualization method was adaptable overall, as these calculations generally apply to the vast majority of LiDAR sensors as they produce similar raw data, meaning we could use this computation scheme on any such sensor. It also made it possible for us to revisit LiDAR as an underlying indoor positioning tech later if it turned out that it was to be the best solution after further exploration of different methods. We would be able to opt for a different LiDAR sensor that would perhaps be better primed towards solving our given problem as opposed to the YDLiDAR we had. This decision was already an improvement to the manner in which we planned our work, as we had now considered both our progress so far and future possibilities in regards to what route the project could take, leaving us with a visualization tool we could always turn back to if needed.

4 | LiDAR Data Game

Although we initially envisioned our LiDAR to be a feasible option for real-time indoor navigation, we eventually came to the conclusion that the current hardware at our disposal was unsuitable for practical implementation. However, we still wanted to incorporate the LiDAR into our project, as we spent our time researching the technology of the component. Because we had already used the Lidar to try and create an initial scan of the rooms we wanted to locate ourselves in, we already had the 2D slices of the rooms available. Unfortunately, the quality of these scans was not sufficient to build our location system on. Instead of discarding this data, we decided to still incorporate it into the project by building an interactive navigation simulation that uses the 2D slices of the real rooms captured by the Lidar.

In this environment, players can move through the rooms and have to reach an end goal to complete the game. Because the slices were initially intended to be used for navigation, the goal of the environment is to get to a specific destination within the room. As an example, one of the final destinations is placed at the door of the room, representing the challenge someone would encounter when using our envisioned navigation system. In this environment, we do not provide any form of feedback with regard to moving towards the right direction, as this is already another part of the project, and the player can easily see whether they are in the right direction while playing. This approach for the environment layout was chosen to still simulate, to a certain extent, the challenges and considerations needed for navigating complex indoor spaces.

The creation of this environment allows us to still keep the initial vision of incorporating the Lidar into the project, but transforms its usage into a more engaging experience, allowing for an easy demonstration of the complexity of navigation through indoor spaces.

The created simulation is in the form of a minimalist 2D navigation game in which players navigate a triangular avatar through a LiDAR-mapped environment using their keyboard's input, in which the avatar's orientation is determined by the cursor position. The game has been designed to mimic real navigation challenges based on three aspects. First, the use of actual LiDAR scans keeps real spatial constraints where players navigate real room spaces. Obviously, these rooms contain correct proportions and wall positions one would encounter in real life. Secondly, the intended goal of the game which is reached by getting to a designated destination, for example, having to reach the door in the room, keeps the original navigation context while converting it to a more interactive experience. Third, the absence of navigation aids like sensory feedback or direction indicators simulates real navigation constraints, although the difference is that in this game, the player has a top view of their position. Moreover, the triangle-shaped avatar fulfils two roles. The triangular shape allows for intuitive orientation as it is easy to observe which direction it is facing, whereas its simplicity allows for easy collision detection with the LiDAR point cloud. Additionally, the game design also reflects the difficulties we experienced while working with the LiDAR data. By using the raw scans as the environment, players experience firsthand how noisy or sparse sensor data can hinder navigation, like finding open doorways from incomplete data or false obstacle detection. Although rather simplified, the game still demonstrates why indoor navigation is harder than it may initially seem, translating our hardware limitations into an interactive case study.

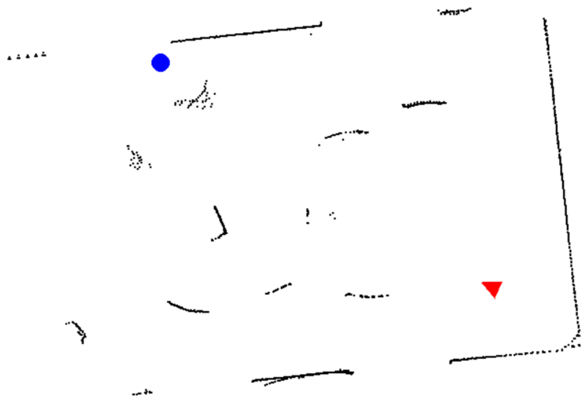


Figure 4.1: Screenshot taken from the LiDAR game, where the player (red triangle) needs to get to the objective (blue circle)

5 | Naext Visit

After concluding that the current form of LiDAR would not be suitable to build further upon for the project, it left us with the room to explore different trajectories. Coincidentally, arranged a site visit to Naext, an experience which would turn out to be a turning point for the project. Naext is a company dedicated to making public spaces more accessible for everyone, especially people with visual impairment and other health-related complications. Their platform offers real-time guidance through audio and haptic feedback, helping users to confidently navigate difficult environments like hospitals and train stations, aiming to redefine accessibility across various industries [12]. In essence, it could be said that Naext is a company making the exact product that we had envisioned our end product of the project could be.

With that in mind, our visit to Naext came at the perfect time. We began with a meeting with Victor van Dinten, who is one of the co-founders of Naext. He shared with us Naext's founding, the company's mission to make public spaces more accessible and the services they provide. He also briefly shared with us the future directions he envisions for the company, talking about how they want to distinguish themselves by being more service-oriented rather than providing 'just a product' and applications beyond healthcare and transport. Inspired by this meeting and the close resemblance between Naext's goal and ours, we were given the opportunity to experience a live demonstration of their technology. This was something we had especially been looking forward to, as it allowed us to further enrich our knowledge, as our research had been rather technical and practically limited.

Victor started by showing the technology they use to create an initial scan of the space they want to navigate. Similarly to us, they use LiDAR, although their component was way more advanced, costing around 50.000 euros compared to our 70 euro LiDAR. With this LiDAR, Victor was able to create a full scan of the room within a minute, which was then uploaded to a point cloud, which he showed on his phone. This reaffirmed our initial thoughts that it was not so much the technology of the LiDAR itself that was faulty for localisation, but rather the current hardware at our disposal that was the dealbreaker. After showing us how they created an initial scan of the room, the actual demonstration of their service could start.

Based on their service, we were guided through the building. One of the first demonstrations Victor did was to show how the system can recognise objects in its environment. When the phone's camera was pointing at an object, the service would identify the object and estimate its relative distance. This allowed for real-time control, not just for people to avoid objects, but also to allow them to get, for example, a coffee at the coffee machine. Moreover, Victor told us that, unlike most other navigation systems, their service could run locally and offline on the phone without the need for additional computing power from external servers. Another impressive feature, besides the navigation itself, was the system's ability to maintain its orientation even when the phone was not held towards the direction indications on the screen. This meant that it could still track users' position and direction for several meters even when the camera of the phone was held at an off-angle position. This avoided penalizing users for not holding their phone in the correct position, making it a thoughtful design choice that improves its usability in real-world scenarios.

Lastly, we had the opportunity to experience the navigation system ourselves. We each took turns holding the phone and tried to navigate ourselves to a destination solely based on the system's feedback. To guide us in the right direction, the system used haptic feedback, which would vibrate more intensely the more we deviated from the correct direction. This approach worked relatively well, as we were able to stay on track without relying on visual input. While the navigation itself worked well, our conversation with Victor during the demonstration revealed that the aspect of providing user feedback was still relatively underexplored. This realization brought a change in outlook for our project. Rather than trying to replicate the entire navigation pipeline, we chose to explore how we could have a beneficial impact by refining user feedback and engagement with such systems.
All in all, the visit the Naext not only validated parts of our initial approach, but also gave us the clarity and inspiration to define a new and more focused, impactful direction for our project.

6 | Digital Twin Environment

As a result of visiting the Naext office and seeing their process, we decided to focus on the development and comparison of efficient feedback methods to relay navigational instructions to users. Since we did not yet have a working method of localizing or routing a user through a building, we needed to find a way of testing and interacting with feedback methods in an environment. We observed that Naext used a game engine to visualize the data that they collected about buildings. This led us to believe that we could create a 3D environment in a game engine to place a user in, and allow them to move through the environment.

6.1 | 3D environment

Using the Godot 3D engine, we created a 3D map, initially consisting of four walls and a target object. Essentially, the map forms a room. The target object in this case is simply a cube in the environment. At this stage, the environment is not very complex, and is supposed to serve as a simplification of real indoor spaces. Further ideas for expanding on the environment are discussed in section 9.

In the environment we place a player, an entity in Godot that can be moved around by the user, along with a camera object attached to the player. In Godot, the camera specifies how the environment is viewed by user—the user's point of view.

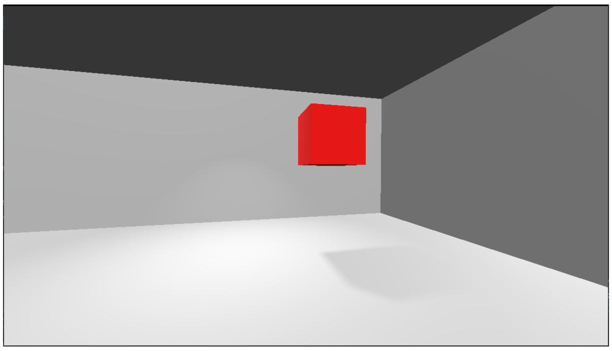
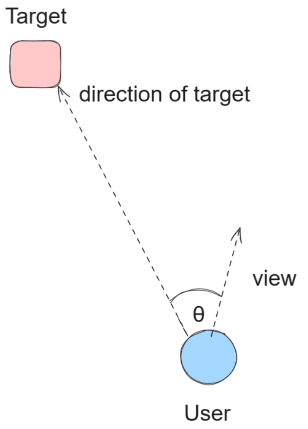


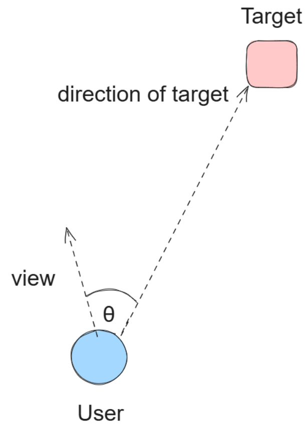
Figure 6.1: The 3D environment. The red cube signifies the target destination. The scene is viewed through the perspective of the user.

6.2 | Feedback to users

Our next step was establishing the parameters that had to be communicated to users. As a result of testing different instructions on a blindfolded folded team member, we decided that a feasible starting point would be to communicate to the user if they should move to the right or left, in other words, the direction of the target destination. The quantity underlying this would be the angle between the user and the target destination.



(a) Target is to the "left" of the user. In this case, θ is a negative quantity.



(b) Target is to the "right" of the user. In this case, θ is a positive quantity.

Figure 6.2: A simplified environment viewed from the top down. Here *direction of target* is a vector from the user to the target destination, and *view* is where the user is facing. θ is the angle between the user and the target destination.

6.3 | Input device

We chose to use mobile phones as the input device for the user, specifically targeting the Android platform. Mobile phones are highly ubiquitous devices, and are minimally intrusive to people's day to day lives. In order to accurately convey the user's position into the digital environment, while still accurately reflecting the form and feel of the final complete product, we estimate the relative position of the device. We use the gyroscope and linear velocity sensors to calculate the orientation and position of the phone.

The gyroscope provides the angular velocity of the phone, or the instantaneous change in orientation expressed in radians around the three axes of rotation, the x, y, z axes as illustrated in the figure below. As we want to estimate the actual orientation of the device in our digital environment, before we use the gyroscope data in the environment, we include a preprocessing step to convert the instantaneous angular velocity to the orientation of the device in radians. The exact conversion steps are described in the following section.

The velocity sensors, similarly, express the velocity of the phone (hence also the user) in the x, y, z axes as shown here. Consequently, a preprocessing step is also required here, and is described in the next section.

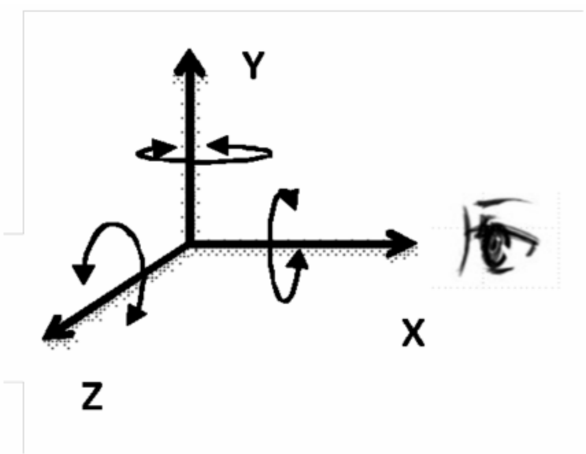


Figure 6.3: The z, y, and z axes, and rotations around them, visualized [9]

6.3.1 | Preprocessing of sensor data

Let the angular velocities be denoted as $\omega_x, \omega_y, \omega_z$ for the angular velocities around the x, y and z axis, respectively. Similarly, let the angle of the device along each axis be $\theta_x, \theta_y, \theta_z$. Since angular velocity is the instantaneous rate of change of the angle over time, the angle of the device at time the following equations can calculate t:

$$\theta_x(t) = \int_{s=0}^{t} \omega_x(s)ds$$

$$\theta_y(t) = \int_{s=0}^{t} \omega_y(s)ds$$

$$\theta_z(t) = \int_{s=0}^{t} \omega_z(s)ds$$

Where $\theta_x(t), \omega_x(t)$ denote the value of θ_x at time t and ω_x at time t respectively.

We follow the same logic for the velocity data. The velocity sensors express the instantaneous velocity at a moment in time; the relative position of the device s_x, s_y, s_z is the integral of the velocity v_x, v_y, v_z along the corresponding axis. This can be expressed in the following formulas:

$$s_x(t) = \int_{u=0}^{t} v_x(u)du$$

$$s_y(t) = \int_{u=0}^{t} v_y(u)du$$

$$s_z(t) = \int_{u=0}^{t} v_z(u)du$$

Where $s_x(t), v_x(t)$ refer to the value of s_x, v_x at time t respectively.

6.4 | Communication between the device and the environment

In order to relay data between the user's phone and the digital environment, we chose to use a server using the UDP protocol [13]. UDP is a network protocol that essentially the base upon which the TCP and HTTP protocol are built upon. UDP messages called "datagrams" have minimal headers (extra information about the message) and hence do not require much processing. It is not as reliable as TCP, but is more applicable to real time applications, like video games or streaming.

The UDP protocol has the unique advantage that it enables us to send smaller pieces of data at a faster rate than a typical HTTP connection. This is crucial since the latency between movements in the device and the corresponding changes in the digital environment has to be minimized to enable a smooth user experience. Although other protocols do exist that can enable similar functions, such as Web Socket [11], gRPC [5], and WebRTC [17]. These require more elaborate setups; hence, for simplicity, UDP was chosen.

The server operates by creating two separate connections with (1) the game engine, and (2) the user's phone (numbering is arbitrary). The server begins operation by establishing connections with both clients (the game engine and, user's phone). Since UDP does not have protocols for certain features, e.g., establishing connections, these features had to be implemented by us. As the server continues operation, data is transferred from one of these connections to the other through a message queue. Both these connections run in parallel to ensure that the latency is minimized.

The server was implemented in Python 3.10, and as mentioned above, is a barebones implementation. As such, it contains only the necessary features, and some extra utilities for testing purposes such as graceful stopping (termination of the server with keyboard commands).
Through this server, the device sends the orientation and position of the user to the environment. In the game engine, we set the orientation and position of player/camera representing the user based on the received data.

6.4.1 | Relaying data from the environment to the device

Based on the direction that the user is facing relative to the target destination, we calculate the angle between the direction the user is facing and where the target object is. Since humans walk and travel mostly on one plane (the ground), in order for the angle to be accurate and useful, we only consider the view vectors and the direction of target vector in zx-plane. In other words, before calculating the angle between these two vectors, we first project them to the zx-plane (see 6.4). We then calculated the signed angle between these vectors, this operation can be performed by taking the dot products of these two vectors. Both of these operations can be performed quite simply in Godot through the `slide()` (projects the vectors onto the zx-plane) and the `signed_angle_to()` (returns the angle between two vectors). Specifically, `signed_angle_to()` returns the angle while paying attention to the position of vectors relative to each other, in our case returning the negative angle if the target is to the left, and the positive angle if the target is to the right.

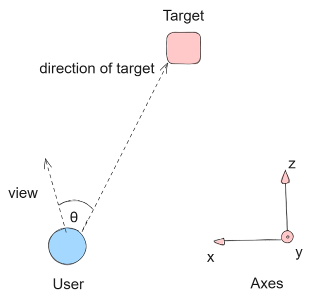


Figure 6.4: Modified version of 6.2b, showing how the axes look in our environment (viewed from top down).

Once the angle to the target destination has been computed, it is then sent to the user's phone.

6.5 | Output to the user

Having received the angle to the target destination, we must navigate the user to the target destination. To make this feedback as unintrusive as possible, we developed three feedback methods. Two of these methods use haptic feedback in the form of vibration using the user's phone, while the third method uses binaural audio.

6.5.1 | Off-path vibration

When the user is facing the target destination ($\pm 15°$), no feedback is provided. However, when the absolute value of the angle between the user and the target exceeds 15°, the user feels vibrations, scaling with how far off the target they are facing (the further away that they face, the stronger the vibrations).

We based this feedback method on our experiences when testing the product offered at Naext. We believe that this scheme is the least intrusive, however it can feel unintuitive when the angle difference is extremely large, for example, greater than 90°.

6.5.2 | On-path vibration

Similar to the previous method, this path also uses vibration, however it is an inversion of the off-path feedback scheme. When the absolute value of the angle between the user and the target is atmost 15°, the user feels a vibration. Otherwise no feedback is communicated.

While this method is feasible, and could navigate users, we found that it was too intrusive. The user would experience nonstop vibrations as they move around (if they are following the correct path). A weakness that affects both of the methods discussed is that when a user is off the right direction, it is not immediately clear whether the user should move to the right or to the left.

6.5.3 | Binaural audio

The binaural audio feedback method uses stereo audio, playing sounds on the left and right speakers separately, to convey the direction of a target. When the object is to the left of the user, the audio is played mostly from the left speaker, and conversely, when the object is to the right. This gives the user a sense of where the object is.

From our experience with this method, this seems to be the most expressive. However, it suffers from the disadvantage that it requires the use of stereo headphones. This impedes our users' ability to use their hearing for their normal tasks. We also knew from previous interactions, as detailed in last year's report, that users prefer not to have their hearing disconnected from the environment.

6.6 | Further work

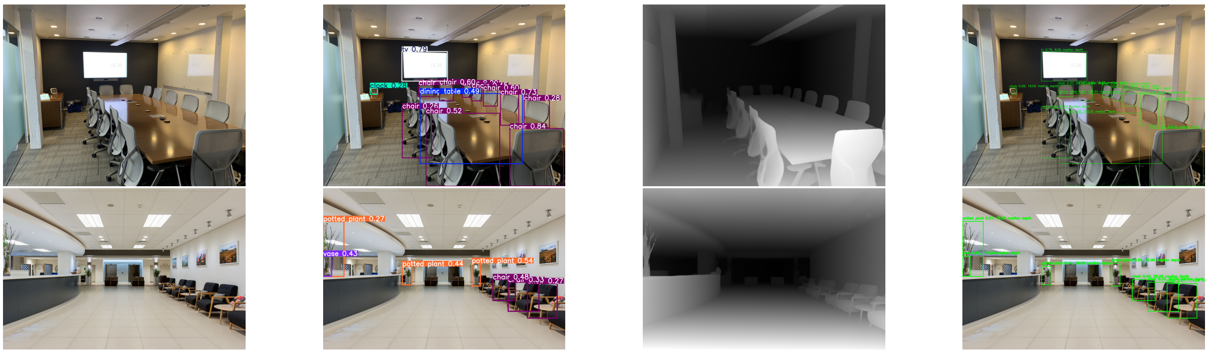
These methods form a great base for feedback, but they can still be improved on much more. A key aspect would be to bring in visually impaired users to test these methods and provide us with feedback about which features/methods are appreciated and which should be reworked. Another aspect of improvement would be with the vibration methods. Using a phone vibrator is convenient; however, it often does not offer a great degree of control in communicating more precise information.

Furthermore, the binaural audio method can be expanded to spatial audio, providing more nuanced information, such as distance from the target, reverberations from surfaces (giving users a clearer picture of where they are in the environment) [14]. This would not entirely overcome the obstructiveness of audio output, but it may perhaps mitigate the issue to a large degree.

7 | Computer Vision Aid

7.1 | Introduction

We conducted an exploration of computer vision as a way to provide a representation of the real world. We hope that the computer vision algorithm can provide an alternative to LiDAR, while also allowing us to navigate users through environments without performing a preliminary pass (requiring the building to first be scanned and then navigate a user through the building). The computer vision algorithm is a combination of two well known models in the world of computer vision, namely the YOLOV8 object detection model and the Depth Anything V2 depth estimation model. By combining these models, we can use pictures, such as those taken by phones, to identify objects in the environment and also how far away those objects are. Examples are shown in Figure 7.1. We are able to perform this with a monocular setup (only one camera), compared to a setup with multiple cameras, which would be impractical for mobile devices.



(a) (b) (c) (d)

Figure 7.1: (a) Input picture, (b) Output from the object detection model, (c) Output from the depth estimation model, (d) Final output, (each object displays a depth measurement, the larger the depth the closer the object.

7.2 | Existing work

In last year's report we observed that users suggested that they would benefit from us highlighting reference points, important objects in the environment. This algorithm allows us to identify those objects in the environment and provide us with an estimation of that object's position in a 3-dimensional environment. We can then communicate this information to the user in different forms of output, such as spatial audio.

As previously mentioned, this algorithm was based on two existing models, which are detailed in the sections below. For both models the smallest version of the model was used, as the processing power available on mobile devices would be quite low. Furthermore, as the model may be used in real time video analysis, it would need to be quite fast.

7.2.1 | Object detection

The YOLOV8 object detection model, is a computer vision classifier trained on the COCO object dataset [7]. As such it detects and classifies objects in an image into one of 80 classes. These classes are common objects such as "chair", "dining_table", etc. The YOLOV8 model outputs a bounding box for each of these objects (a rectangle around the object detected in the image), a confidence level for the classification, and the predicted class label.

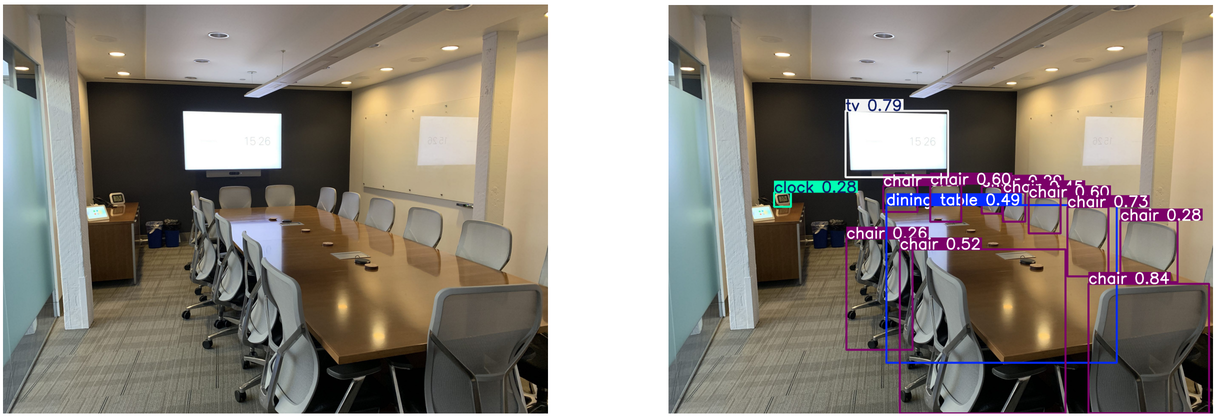


Figure 7.2: Output from the object detection model. Different color boxes denote different object classes.

7.2.2 | Depth estimation

The Depth Anything V2 model generates a depth map for an input picture [18]. A depth map is an estimate of the relative depth of each input pixel. Note that this measure of depth is relative, and not absolute, hence the depth measurement doesn't have any units. This model has been trained on 595,000 labeled synthetic images, and around 62,000,000 real, unlabeled images.

Figure 7.3: Output from the depth estimation model. Lighter regions indicate pixels estimated to be closer to the camera (at a smaller depth).

7.3 | Processing and output

For the processing of the outputs from this model, we used the OpenCV and Numpy libraries from python. Since the output image from both the models was scaled to the same size, for each bounding box (hence each object), the depth of each pixel in the box could be estimated. In order to estimate the depth of the object as a whole, initially the minimum depth in the bounding box was used. However, this was highly susceptible to being skewed by outliers, estimating some objects closer than they appeared. In order to mitigate this, we used the median of the depth of the pixels in the bounding box. As the median is a measure of central tendency that is resistant to outliers, we got significantly more intuitive results using this method. After this the bounding boxes labels were appended with the median depth measurement.

7.4 | Aspects for improvement

Since the object detection model is trained on a generic dataset (the COCO dataset), many of the classes are not as relevant for navigation and hence are not specific to our use case. A future step that could be taken is fine-tuning the object detection model to recognize more important objects, such as doors, elevators, staircases, signs. This would enable a significantly better navigation experience for users.

Additionally the computation and the processing of the two models can be pipelined more efficiently, and extended to real time video. This would enable users to get feedback at a much faster rate than is currently possible. At the current moment, processing an input picture takes around 5 seconds, which would be too slow for a live video analysis.

A further improvement would be communicating this information to the user in an effective way. This seems to be a prime opportunity to use the spatial audio generation [14] feedback method. This is also quite similar to a project led by Microsoft called Soundscape [15](no longer being offered), which would use spatial audio in an outdoor setting to tell users where certain stores, locations, buildings, etc. were in an outdoor setting. The computer vision model described above could provide the same form of feedback, but more granular and in an indoor setting.

8 | Conclusion

In conclusion, this year, our team managed to successfully put together a proof-of-concept prototype that combines sensors from mobile devices, real-time feedback, and its flexible, digital testing environment. At its core is an Android app that continuously reads gyroscope and linear-velocity data, and integrate those inputs into estimates of device orientation (θ x, θ y, θ z) and position (sx, sy, sz) in space and then sends the data thorough a lightweight Python 3.10 UDP server, keeping the latency low and allowing for sub-second updates between the phone and our digital twin. We were able to asses our solution in Godot—based virtual environment modeled in Blender and imported as meshes. Since the user's camera view mirrored the phone's orientation, we were able to thus create a realistic indoor "digital twin" for testing.

Inside this environment, we implemented three different feedback modes right on the phone. The off-path vibration, that turns on once the user's heading deviates more than ±15°, the on-path vibration buzzes only when they're within that same angular range, and binaural audio cues, that use stereo separation to point you left or right.

Although our budget LiDAR was not great for real-time positioning, we still built a universal visualization pipeline, from a CSV parser that converts raw angle-and-distance data into 2D maps to a standalone "LiDAR game" that uses such created maps for a controlled through avatars maze-like rooms. It illustrates how noisy or sparse data can affect navigation in space and keept LiDAR as a useful tool we could explroe this year. We also investigated a vision-based approach by combining YOLOv8 object detection with the Depth Anything V2 depth estimator to generate bounding-box medians to approximate real-world distances to actual objects.
Thus, by the end of this year, our deliverables are:

- The Android prototype app, complete with sensor pre-processing, UDP communication, and all three feedback routines.
- A Python UDP server for bidirectional data transmission.
- The Godot digital twin environment, with imported custom Blender meshes where we can render real-time user movement.
- Three feedback modules (off-path haptics, on-path haptics, and binaural audio).
- A LiDAR visualization model with an interactive 2D navigation game.
- A proof-of-concept computer-vision demo combining YOLOv8 and depth estimation.

All those deliverables, taken together, form an proof-of-concept prototype for assesing user feedback methods in indoor navigation. A solid practicatand tangible foundation for the user testing and iterative improvements that can be undertaken next year.

9 | Discussion

Our project currently stands in a unique position. Throughout this Honor's year, we explored multiple different ideas and paths we could take, ultimately opting to focus on the method of feedback to a user of an indoor navigation device. This is the portion of the indoor navigation problem that we found was largely untackled, and also where we deemed we could provide the most value from our position as an Honors team. The general idea is that once perfected, we would use this method in conjunction with indoor mapping and localization technologies developed by companies such as NAEXT. This combination would then constitute a complete product that could be used as an indoor navigation device, in line with the original vision of NaviSense. Thus, the next steps for our project should constitute further iteration on the digital twin and feedback methods, as well as exploring how to incorporate our product into a complete navigation system as mentioned above.

9.1 | Digital Twin

We currently treat the digital twin environment as the core of our testing framework. Moving forward, this environment needs to evolve in two key directions. Firstly, we must make improvements to realism and fidelity to improve correspondence to real-world spaces. Secondly, we must implement enhancements to testability and usability to aid rapid development and experimentation.

9.1.1 | Improving Fidelity and Correspondence

Lateral motion implementation remains our highest priority. At present, we rely on gyroscope data for rotation, which has proven reliable enough for testing orientation-based feedback. However, lateral translation is fundamental for guiding users through space, and its absence limits our testing severely.

We previously attempted to implement this by streaming velocity data from the IMU over a socket connection and applying it to the player's position in the Godot engine. Unfortunately, this approach yielded erratic behavior. The inconsistency stems from the noisy nature of mobile phone velocity data — especially without sensor fusion or filtering. Future efforts should focus on smoothing or filtering this data or integrating accelerometer readings over time with drift correction.

Another critical aspect is addressing rotation drift. As mentioned in Section 5.3.1, the orientation is calculated by integrating angular velocity. This method accumulates error over time and results in drift. We need to implement a calibration routine, potentially using a fixed reference direction or compass data, to allow users to reset their heading periodically. In addition to this, we envision a recalibration prompt mechanism. This system would detect inconsistencies between expected and actual orientation, and request a manual reset. For example, if the system detects an angle delta inconsistent with user movement patterns, it could prompt a recalibration. Another means through which we could address this is by asking users to recalibrate the device periodically.

Lastly, to combat observed latency issues (although minor), we plan on evaluating the existing codebase for bottlenecks and exploring lower-latency networking options (e.g., more optimized UDP payloads or socket reuse). We also want to investigate whether hardware-induced delays (such as low vibration module responsiveness on some mobile devices) might be contributing factors.

9.1.2 | Testing & Quality of Life Improvements

One important requirement of our envisioned indoor navigation device is that it will function in a variety of different environments. Thus, one of the more pressing needs is to implement a level editor-style tool to create and save environments effectively and in an organized manner. Although Godot does support some built in level building, we found that it was not really tailored to our specific use case, which has a focus on real-world correspondence. A tool of our own will allow faster iteration and testing, especially when conducting user studies on a variety of different environments. Overall, this should contribute to greater sustainability of the project moving forward.

The editor should support placement of walls, objects such as pillars and other obstacles, and reference markers through a simple GUI. Furthermore, users should be able to scale these environments and the objects within them freely, adjusting their dimensions as needed. To increase development speed, the tool should also allow copying and pasting of objects, and saving custom objects. To make real-world correspondence easier, we also plan to include reference objects (e.g., a standard 1 m-wide door frame), and a functionality to visualize the exact measurements of everything.

Another priority is the implementation of actual navigational routing capabilities. To achieve this, we envision using our already implemented "target blocks" in conjunction with a pathfinding system, similar to something you would have in a video game. Upon starting a demo, the tester could define the starting position of the user within the environment as well as a destination point. Our pathfinding system would then generate a sequence of waypoints that would form a path to the destination. We envision this as a dynamic system that only activates the waypoint nearest to the user at any given time, guiding them step-by-step. In addition, if the user were to accidentally stray from the predefined path, the system would adjust and correct for this. The goal is to allow realistic testing of end-to-end navigation through multi-step environments.

As a first step cross-team collaboration, we plan to explore an import system for 3D meshes within the environment editor. Given potential future cooperation with companies such as NAEXT that provide indoor mapping, this would allow importing real indoor scans into our system and transforming them into a navigable environment formed by a mesh.

As a last quality of life feature, we plan to introduce a settings menu with toggles for different feedback modes and experimental options. This modularity will make testing and user preference recording significantly easier.

9.2 | Feedback Methods

As outlined earlier, the current focus of our project is to refine and compare various feedback methods for guiding users. In the coming Honor's years, the majority of our work should focus on testing, evaluating, and iterating on these methods.

9.2.1 | In Depth Evaluation

Once we have internally developed and refined a set of feedback methods through initial experimentation, the next step is to perform structured evaluations. These evaluations should include individuals with visual impairments of varying severity and duration, allowing us to compare feedback modality effectiveness across user groups.

We plan to use a hybrid evaluation strategy. Part of it will entail quantitative performance trials, measuring metrics such as task completion time, path deviation, and number of incorrect turns for a given navigation action. We plan to then combine this with qualitative data from post-trial surveys and semi-structured interviews.

The test environments will be varied in complexity, with trials of varied difficulty and scale. We plan on progressing them from traversing simple environments with a few obstacles to traversing more complex environments with more obstacles or potential for vertical movement (staircases). Eventually, we want to conduct tests where the user is tasked with traversing an entire building successfully.

To manage our limited resources and ensure focused development, we plan to adopt a staged evaluation approach. Rather than testing all feedback modes to the full extent from the get-go, we will begin with a preliminary screening phase. In this initial phase, each feedback mode we end up developing will be evaluated through small-scale, low-cost tests with a limited number of participants. These tests will help us identify any feedback modes that are fundamentally unintuitive, poorly received, or difficult to implement reliably.

Feedback methods that perform poorly in this early phase will be discarded or bumped down to a lower priority or future consideration status. For the remaining promising methods, we will proceed with the more detailed evaluations and iterations. This staged approach allows us to focus effort where it has the highest potential payoff, and refine those feedback methods that show the greatest promise.

9.2.2 | Advanced Feedback Modalities

Although phone vibrations are a great starting point in terms of feedback methods, they may lack the precision needed for complex environments. We therefore plan to explore more advanced methods, including:

- **Spatial Audio:** Expanding on the binaural approach to include cues like distance or directionality with environmental reverberation.
- **Multimodal Systems:** Combining audio and haptics for richer communication.

We may also try to revisit legacy concepts from previous work on this project, particularly custom wearable devices that could provide the user with haptic or audio feedback. Building and testing these devices will now be more within our reach as compared to previous years. As we are now working with a digital environment instead of real world data, their functionality needs to be focused solely on providing feedback to the user. Of course, they should also be evaluated for comfort, accessibility, and user autonomy. Wearability, discretion and user comfort will remain core design criteria, as was emphasized by work done on this project in previous years.

9.3 | Cooperation with Other Companies

On a more distant horizon for our project, we imagine close cooperation with companies such as NAEXT. As discussed previously, our digital twin environment is more of an intermediate step that we need to develop and test our end goal of a feedback mechanism.

Once we have refined our feedback methods, we will focus on integrating them into systems like NAEXT's. Since we do not aim to provide the full localization and navigation pipeline ourselves, our solution will function as an output layer within such systems, translating the positional and directional data they generate into real-time, intuitive feedback for the user. This means our system will rely largely on external localization and navigation services for environmental awareness, while taking on the responsibility of communicating navigational cues through vibration, audio, or other methods. In addition, maintaining a separation of concerns in this manner allows our solution to remain flexible and compatible with a variety of navigation technologies, enabling it to adapt easily across different use cases and infrastructures if necessary.

Eventually, we must also evaluate the commercial viability of our solution. An initial step we envision in this direction will be conducting a structured user study with participants who have visual impairments. This study, which we plan to carry out under the supervision of the TU/e ethics board, will help us gather feedback on the perceived usefulness, intuitiveness, and comfort of our feedback methods. Unlike the earlier in-depth evaluations (Section X), which are primarily focused on iterative testing and comparison of feedback modalities, this user study will aim to assess non-technical feasibility questions. It will help us explore how users perceive the system as a whole, their confidence in using it, trust in its guidance, and willingness to adopt it in everyday life.

Following this study, we intend to organize smaller pilot deployments in real-world environments. We would like to collaborate with high-traffic indoor spaces such as public transport hubs or hospitals. These settings have been identified in our prior research as particularly challenging for individuals with visual impairments, making them ideal candidates for evaluating the usefulness of our system. These pilots will help assess how well the system as a whole performs when embedded into a full navigation pipeline like NAEXT's, with our feedback methods acting as the user-facing output layer.

In addition, to ensure the involvement of the main user, we plan to collaborate closely with individuals who have visual impairments throughout the pilot phase. Their input will guide the design and adjustment of feedback methods, helping us make adjustments based on very direct experience. This approach will not only help improve the quality of our product, but also strengthen trust in the system.

While ethical considerations have not been a major factor in our current small-scale setup — where testing is limited to our team and controlled environments — this will change as we move toward broader deployment. Once our system interacts with real users, particularly in public settings and in collaboration with companies like NAEXT, we will need to address a range of ethical, privacy, and legal concerns.

One major consideration is data privacy. Our system may eventually process sensitive information such as indoor locations of users, their movement patterns, and behavior over time. Even if we do not directly store this data, working within a navigation pipeline could involve receiving or transmitting user-related information. This means we must take into consideration questions regarding user consent, data handling, storage, and sharing, especially when users belong to a vulnerable group such as individuals with visual impairments.

To address this, we will need to define and embed within the product clear policies on data collection (only collecting what is necessary), user consent, and anonymization. Collaborations with third-party localization providers will also require agreements to enforce these ethical standards across the entire product stack.

Finally, from a legal standpoint, scaling our product may involve compliance with certain accessibility or consumer protection laws. We must make sure that the system is usable for people with a wide range of visual impairments, and that it does not introduce any additional barriers.

Overall, while ethical concerns are not yet a large barrier in the development of our product, we must make sure that we keep them in mind. As stated previously, our intended user is a group of people that is rather vulnerable, and as such ethics is baked into the problem at hand. We must learn how to approach and navigate ethical questions as soon as possible, and make sure we do not take our product down a path that is disconnected from the needs of the user.

10 | References

[1] Are wi-fi positioning systems still worth implementing in 2025? https://mapsted.com/blog/wifi-positioning-system-explained. Accessed: 2025-05-23.

[2] Ydlidar g4 360° laser scanner. https://www.ydlidar.com/products/view/21.html. Accessed: 2025-05-23.

[3] gi0rg0spapamichail. Ydlidar library. https://registry.platformio.org/libraries/gi0rg0spapamichail/YDLidar. Accessed: 2025-05-24.

[4] grayson arendt. Rplidar a3 library. https://github.com/grayson-arendt/RPLidar-A3-Arduino. Accessed: 2025-05-24.

[5] gRPC Authors. gRPC. https://grpc.io/, 2025. Accessed: 2025-05-25.

[6] Max Holmberg, Oskar Karlsson, and Michael Tulldahl. Lidar positioning for indoor precision navigation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 358–367, 2022.

[7] Glenn Jocher, Jing Qiu, and Ayush Chaurasia. Ultralytics YOLO, January 2023.

[8] kaiaaai. Laser distance scan controller library. https://registry.platformio.org/libraries/kaiaai/LDS. Accessed: 2025-05-24.

[9] Maria Kozhevnikov, Jodie Royan, Olesya Blazhenkova, and Andrey Gorbunov. *The Role of Immersivity in Three-Dimensional Mental Rotation*, pages 143–157. 01 2008.

[10] Bing Li, Juan Pablo Muñoz, Xuejian Rong, Qingtian Chen, Jizhong Xiao, Yingli Tian, Aries Arditi, and Mohammed Yousuf. Vision-based mobile indoor assistive navigation aid for blind people. *IEEE Transactions on Mobile Computing*, 18(3):702–714, 2019.

[11] Mozilla Developer Network. WebSockets API. https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API, 2025. Accessed: 2025-05-25.

[12] Naext. Technology that brings spaces to life. naturally. https://www.naext.eu/en.

[13] Jon Postel. User Datagram Protocol. Request for Comments 768, Internet Engineering Task Force (IETF), 1980. Accessed: 2025-05-25.

[14] Thomas Potter, Zoran Cvetković, and Enzo De Sena. On the relative importance of visual and spatial audio rendering on vr immersion. *Frontiers in Signal Processing*, Volume 2 - 2022, 2022.

[15] Microsoft Research. Microsoft soundscape. https://www.microsoft.com/en-us/research/project/microsoft-soundscape/, 2022. Accessed: 2025-05-23.

[16] Leah A. Wasser. The basics of lidar. https://www.neonscience.org/resources/learning-hub/tutorials/lidar-basics. Accessed: 2025-05-23.

[17] WebRTC. WebRTC. https://webrtc.org/, 2025. Accessed: 2025-05-25.

[18] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024.

A | Individual Contributions

A.1 | Lucas Iovdij

At the beginning of the project, I asked everyone whether I could take up the responsibility of being the project lead. This was in line with my PDP, and ever since then, I have been working to make sure the project keeps running and that everyone has the best environment to focus on their work in.

Throughout the course of the project, I was responsible for setting up our meetings and keeping track of our current goals and agenda. This was usually done in my personal docs, as the team eventually drifted away from using a shared platform. The information I kept track of was minutes, our goals, the progress towards those goals, and our possible next steps. As a part of this, I tried to make sure that every design decision we made was in line with everyone's goals and that it involved everyone's input. Though subtle, I often worked to nudge our discussions towards the agendas we planned by means of posing particular questions or shelving discussion points for later. I did this while taking into consideration the perspective of our *client*, so that we had forward momentum without impeding on anyone's stances and desires during the development. Additionally, all these decisions needed to be made in the context of our defined problem statement, so I also made sure to make the reasoning for our decisions be tested against this problem; think: "would making our solution be a handheld device be something our target user would be okay with occupying their hands with?"

Aside from that, I also took part in the team-wide research at the beginning of the project, where every one of us conducted research on different technologies we could use to create an indoor positioning solution. In particular, I researched the use of Wi-Fi - the methods of implementing it, its specifications, and collected some examples that were also trying to implement this positioning system utilizing this technology. Although we decided not to use it in the end, the information I collected gave us clear reasons as to why to avoid it, as it was not suited for this project at all.

Later on, I aimed to give others as much technical assistance where it was needed. As per my developmental goals, I strived to take up soft skills first and foremost, but I also provided some of my coding skills where needed. Specifically, I essentially made our initial visualisation method of the data collected by the LiDAR sensor with the help of Olaf. I made it so that it stored the read data, angles and distances, into a csv document that was then used to compile 2D points out of the data, making it into a map. I made it in an effort to make the data reusable, i.e. the collected csv documents were accessible off premises allowing others to work on the positioning model even without having access to the LiDAR. I then made sure to also note down the way this method works, what libraries it required, and how one could set up the LiDAR to work with their computer so that everyone could always just follow along with this guide and easily develop upon it further. Moreover, I shared my knowledge with Teodor and Alicja during the development of the Godot digital twin, as I use this engine in my free time, hopefully helping them achieve our goals more quickly.

In this regard, I also tried taking care of all of the official correspondence. This includes tasks like filling out questionnaires for the Honors academy, writing mails to the staff at TU/e for support, and taking care of reimbursements. As a part of this, I also took care of most of our communications with the product owner, Juan. I kept him up to date with the progress of our team and our upcoming milestones and listened to any feedback he would give.

A.2 | Yash Rana

During the project, my role was a technical development role. As such, most of my responsibilities were focused on developing software/devices. Additionally, I also took on the role of being the presenter during demonstrations and group presentations.

In this report, sections 6 and 7 were written by me.

In the initial part of the project, my responsibility was to research and use the LiDAR scanner and extracting information into the program. This involved exploring an embedded device with a LiDAR scanner together with Teodor. However we identified that this was not a viable solution.

As part of the team process, during the brainstorming and design phase of the product I offered several solution for incorporating the LiDAR scanner into a physical device. These and the designs of the team helped us refine possible solutions. As a team we wanted to develop a solution on mobile devices due to how common they are in day-to-day life.

After this I was responsible for the setup of the digital twin environment, discussed in Section 6. Specifically I developed the functionality of the 3D engine (the core functionality), the UDP server and the Android app. I also implemented the collection, processing and communication of the gyroscope orientation to the server. Additionally, in order to help the rest of the team to contribute in the use and development of this environment, I helped them with understanding the environment during team development sessions.

In order to provide an alternative to LiDAR scanning, while also being able to use mobile devices for navigation I developed the computer vision aid described in Section 7. The goal of this was to explore, and possibly create an approach that could be expanded upon in the future. In this sense, I believe that this is quite a promising technology that would not only set us apart from competitors in the same space, but would greatly provide value to users. At the same time a concrete, but not fully fleshed out, program like the one developed can be a stepping stone for future work, and can make this technology seem more approachable.

As it was a goal in my PDP, I was responsible for delivering presentations during multiple events, such as the mid term demo. This meant that I worked together with the team in developing materials such as posters and slides, and then presented them. The focus of these presentations was on explaining our vision and product development to audiences of different knowledge levels.

I believe that as a part of the team I was able to make several significant contributions to the developments that we made during the year. However I could not have made these contributions without the ideas and designs that were born out of discussion with my team members. These opportunities allowed me to further explore my interests in the technology involved, while building towards a product that would be able to fully navigate a visually impaired user.

A.3 | Alicja Napieralska

This year, since I was the only team member who worked on the project last year too, a lot of my contribution revolved around bridging the knowledge we had gained from the previous year. I worked a lot on refining the idea and providing the team with insights based on our past research, navigating the focus of the project, indicating which paths might be worth exploring and making sure we were heading in an effective direction.

Additionally, for the first part of this year, I was responsible for researching the technology we would be using moving forward. This included investigating different options like beacons, Wi-Fi, and Bluetooth-based solutions, as we were still figuring out which tech would be the most reliable. I also handled a lot of the logistical side of things, such as finding facilities and lab spaces on campus where we could work and test our ideas. I ultimately focused on beacon technology, as it proved the most precise and although expensive, I found out our university has a lab with such system.

As the year progressed and we started to have a more tangible idea for the project progression, my focus shifted towards researching similar projects and identifying where we could take ours next, especially similar university projects, such as one made in Delft University that also explores indoor navigation systems (although with reliance on visual cues). I was responsible for looking into feedback and output methods figuring out the best ways to relay navigational information to users. How to use vibration and force to indicate direction, and what would be effective as the output method for our design.

Another part of my role was communication with our client Juan, alongside Lucas. I handled planning and arrangements with him, as well as keeping him updated by providing briefs and updates on our progress.

For the first half of the year, I also worked on the requirements documents that did not make the final cut as we have pivoted a lot from the initial ideas we had for this year and that I was researching. A lot of my work was into usage of camera-based systems, which we abandoned mid way through the project and only came back at the very end of the project, when there was to little time to redefine it.

In terms of the deliverables and the tangible product, I created the 3D virtual environments used to test our devices, rooms and corridors. I learned how to work with Blender, a 3D design tool used often in the industry, from scratch to design these environments, and then applied them in the Godot game engine, which I also learned from grounds up this year, where we would be able to connect it to the code of the device and feedback. They were then used to test various edge cases we identified during our research and feedback methods in a more realistic setting that resembled more actual public spaces. Through these virtual environments, we were able to simulate how the technology would function in real-world scenarios and thus refine our approach, such as improving the input to the system from the environment or better repositioning of the device in relation to its surroundings.

A.4 | Olaf Hermans

Initially, my role within the team was to work on the LiDAR just like the others. Specifically, I wanted to contribute by getting the LiDAR to work, as I had no prior experience in this. That being said, this also caused me to not fully grasp what was going on when Yash and Teodor were working on it, as it required some coding experience in C++, which I had none. To still be able to make a contribution to the LiDAR part of the project, I worked together with Lucas to make sensible data of the LiDAR readings and visualize it in a 2D map.

After concluding that LiDAR would not be a feasible option, and continuing to build a digital twin, I decided to start and work on the LiDAR game mentioned in 4. I had chosen to do a somewhat deviated part from the main project, as it allowed me to contribute to the project while completing my PDP, which focused on improving my programming skills.

I tried to contribute to the team process by staying involved in meetings and contributing to discussions. Although I have to admit that this was sometimes limited due to my lack of experience and knowledge in the discussed matter, compared to other team members.

My main contributions to the team deliverables were the creation of the LiDAR maps together with Lucas and the creation of the LiDAR Game mentioned in Section 4. Consequently, I also wrote the corresponding section in this report, together with our visit to Naext mentioned in Section 5.

A.5 | Teodor Krajč

For this project year, my role was that of a technical developer. Therefore, my responsibilities and contributions can be summarized as the technical implementation of the product. In addition to this, I contributed to discussions during group meetings and often tried to make a connection between technical implementation and higher level design decisions. From my position as a technical developer, I tried to translate technical challenges that we were encountering into terms that were accessible to teammates who were less familiar with the tools and technologies involved at the current stage.

Towards the beginning of the project, I took part in the initial research phase (mentioned earlier in the report). In particular, I focused on investigating the functionality and feasibility of BLE (Bluetooth Low Energy) technology as a potential component of our solution. This involved researching the theoretical background and understanding it's technical constraints: range, precision, and cost. Eventually, this contributed to the group's decision to discard BLE in favor of approaches better suited to our project.

Once we decided to opt for LiDAR SLAM as our cornerstone technology, I was responsible for the initial embedded work together with Yash, as we had the most experience and interest in this topic. Early on, we met up and soldered all the components together. I spent time experimenting with a number of libraries, trying to understand how to configure the LiDAR. It took multiple attempts and sessions before we managed to get a usable output from the device. Getting the LiDAR to produce usable data was a difficult process that involved driver issues, unclear documentation, and trial-and-error configuration.

Ultimately, we chose to move away from LiDAR due to the numerous challenges we encountered during its implementation, as well as its limited viability as part of a working end product. At this point, I advocated for a shift in direction, as I felt the team was stuck and would benefit from reassessing our approach. This led to a group-wide brainstorming phase aimed at identify a more feasible solution. During this phase, I contributed to the discussions, helping come up with and evaluate alternative ideas. Due to personal reasons, I was absent for part of the project, during which the NAEXT visit happened.

Due to personal reasons, I was absent for part of the project, during which the NAEXT visit happened. After this, thanks to the initiative taken by Yash, the main direction of our project shifted toward the development of a digital twin environment. In line with my PDP goals of improving technical capabilities and exploring my interests, I took on the role of shaping and iterating on this system. I began working on lateral translation, an important feature required to simulate real movement. This part of the implementation remains a work in progress at the time of writing, due to erratic behavior of the IMU data. I also helped ensure our codebase ran smoothly on multiple devices, resolving compatibility issues on my own device. Additionally, I implemented basic manual movement controls in the digital twin to help with internal testing of the environment.

When Alicja began developing the collision environment, I helped integrate her contribution into the digital twin. This required merging our work so that user collisions could be accurately represented within the simulation. Lastly, in this report I helped write Section 10. Future Steps.

Lastly, in this report I helped write Section 10. Future Steps.