IS-313: Location Intelligence

CDE3301 Interim Report
AY 2024/25

Submitted by
Lim Ji Yong, A0282723A
Lin Hong Yi, A0307436E
Anitej Datta, A0284708U

Supervisors
Dr. Yen Shih Cheng
Graham Zhu Hongjiao

Acknowledgements

The team would like to express our sincere gratitude to B&O for their insights on the importance of location information as well as the ideas they had on potential implementations, which inspired us to decide on developing this middleware.

The team would also like to thank our amazing studio masters Dr Yen and Graham for their support and guidance throughout the past semester, as well as Tse Han for sharing his preliminary results on UWB accuracy.

We would also like to thank our peers from our studio group – Givson, Cheng Yi, Tse Han, Annanya, Kaveri, Siddhi, Tanmay, Rishi, Avaneesh, Teng Wai, Kashfy – for working with us and generously extending help and guidance throughout the semester.

Table of Contents

Chapter 1: **Introduction**

Our Industry Partner - Bang and Olufen

Bang & Olufsen (B&O) is a Danish high-end audio solutions provider. They are renowned for creating iconic audio and home entertainment products - to the highest standards of sound, craft and design.

In recent years, B&O has been keen to develop more “seamless” user experiences. The vision is that users can effortlessly manage their devices, and unite multiple speakers into a cohesive system for an extensive, immersive audio experience.

Currently, B&O is looking into developing a smarter network of audio devices. Many of B&O loud‑speakers already ship with NXP UWB chips, while every modern iPhone (11 and later) carries the Apple U1 chip [1]. This NXP UWB chip enables high-resolution indoor localisation.

Problem Situation

After speaking to B&O’s product designers directly, the team clarified their prospective demands. There are some features that are not currently implemented but B&O would like to make possible. Here, the team has compiled a few scenarios:

1. “Follow‑Me” audio sweet‑spot
   When the listener moves from the sofa to their yoga mat, the stereo image auto-pans so the listener always stays within the acoustic sweet‑spot.

2. One‑Tap speaker set‑up
   When a new speaker is unboxed and plugged in, it can automatically join the integrated home collection.

3. Adaptive multi‑room hands‑free speaker switch
   When the listener crosses the doorway, music crossfades graudually from living room to kitchen exactly with no manual taps nor voice commands.

4. iPhone Gesture control
   To skip a track, mute, or adjust volume, the listener can draw gestures in mid‑air.

5. Dual‑user “audio bubbles”
   With two people sharing the same living room and each wanting different content

(podcast vs. TV show), speakers must render two dynamically-moving audio sweet spots.

6. Localisation-based Playlist Transitions
   This idea builds on the above features. Suppose a listener habitually plays classical music at the dining table and plays rock in the kitchen. This prospective feature automatically observes this pattern. After a few days of this routine, when the listener walks into the dining room after cooking, the speakers will automatically change the next queued song to classical music seamlessly to the dining table.

These scenarios surfaced the recurring requirements of having real-time location knowledge of user and speaker positions, at centimetre level and multi-room scale. Accurate indoor location is the missing data primitive for the next wave of iconic audio experiences.

Introduction

B&O's current solutions to integrate users' collections are the Beolink system [2], [3], and a physical external hub BeoConnect core [4]. However, they still rely on traditional manual control mechanisms of using remote controls, mobile applications, or physically touching the speaker. These systems are designed around static configurations, requiring elaborate setup and on-demand manual intervention. In addition, they are designed with only B&O devices in mind and are a closed system that other brands cannot integrate with.

Hence, the team proposes to develop middleware that provides real-time dynamic positioning for B&O, enabling them to adapt to the user's movement and seamlessly integrate the system into the user's spatial context - without requiring extensive setup or manual intervention. Potentially, other home automations should also be able to tap on this and adapt to user behavior. The team intends to create a highly-accurate indoor grid that enables interactions adapting live to user behaviour, like seamless room transitions, which BeoLink and BeoConnect Core cannot offer.

Given that this location data layer is vital, and the relevant hardware already sits inside most B&O speakers and every modern iPhone, why hasn't the market delivered these experiences? The next section details that gap and clarifies where current approaches fall short.

Chapter 2: **Prior Art & Gap Analysis**

Current Localisation Methods

Given that high-resolution localisation is at the heart of our solution, the team looked into existing technologies with this function.

Background Context on Localisation

Typically, indoor localisation is based on measuring the distance of an object from two or more known points. The devices that emit signals at these known points are referred to as beacons or anchors.

For the anchor to detect the object, a direct path of vision between them is often needed. This straight, unobstructed path is known as a line of sight (LOS). Non-LOS (NLOS) situations are where there is no direct, unobstructed path between a transmitter and receiver, but signals can still relay.

Tech	Accuracy	Pros	Cons
BLE RSSI / AoA	2 m LOS [5]	- BLE hardware is cheap and suitable for widespread use - It is present in most devices and does not need special equipment	- No inherent directional data, so a minimum 3 B&O devices are needed - BLE lacks vertical resolution, making it difficult to track users on different elevations for, say, motion detection
Wi-Fi RTT	≈ 1.5 m LOS / NLOS [6]	- Uses existing access points and does not need special equipment	- No directional data if there are less than three B&O devices in range. But having three Wi-fi routers will be financially and spatially inefficient - Location fingerprinting is usually needed, and is sensitive to future changes in the environment [7]
IR / LiDAR	1-5 cm [8]	- IR is highly-precise and can measure at	- Mapping will not work when blocked by objects such as furniture or walls - IR cannot identify the users

		5 cm accuracy for radius up to 80 cm.	
Camera CV	1 cm	- CV has the potential to achieve precision of a few millimeters - CV does not require users to carry any tracking tag	- Cameras capture detailed visual information, raising privacy issues for use in homes - Multiple cameras have to be set up for full coverage, which can be expensive
UWB TDoA / AoA [9]	0.20 m NLOS	- Direction + range; robust to multipath - UWB can provide 20cm accuracy, and is one of the most precise indoor localization technologies	- Implementations of UWB are currently limited to point-to-point or proprietary RTLS (non-audio market) - UWB hardware is more expensive than traditional localisation technologies

Table 1: Comparison of related indoor localization systems.

Gaps to aforementioned scenarios

The six scenarios mentioned in Chapter 1 are currently not actualised right now. The issues are described below.

Scenario	**Current Solutions Fall Short**	**Why Our Solution Solves It**
"Follow-Me" audio sweet-spot	- BLE and Wi-Fi lack directional data, and multipath interference causes inaccuracies. - Vision-based solutions **fail when users face away** or when light is poor.	- UWB's **direction with range** data gives us exact location tracking with <20 cm accuracy and 50ms latency. - Users can be tracked even when blocked, especially if their phone is in range of at least 1 other speaker, which contributes to the global coordinate grid.

One-tap speaker set-up	- Currently buying a new BLE/Wi-Fi device requires **manual calibration by experts** to optimize speaker placement. - Current UWB solutions are peer-to-peer, and only work in fixed set-ups.	- Our grid integrates the new device **into the mesh** as soon as they're placed in range, without any manual setup, and can help with **calibrating the new speakers** according to the existing speaker setup.
Adaptive Multi-room Speaker switching; and Localisation based Playlist transitions	- Multi-room mapping will be difficult to conduct with trilateration techniques as the **accuracy is not enough**. - Data from visual methods will be **difficult to piece together** to get multi-room data compiled.	- UWB range is large and can overlap to create a real-time **multi-room** accurate grid. - **Uniformly accurate** user location from the grid enables a well-orchestrated audio crossfade.
iPhone Gesture Control	- Vision-based systems require **LOS** and are **light-sensitive**. - Proximity-based methods do not have the resolution to see **spatial gestures**.	- **UWB's precise location** has the potential to track gesture-based commands. - The system doesn't require line of sight, works irregardless of **lighting conditions**.
Dual-user "Audio bubbles"	- BLE / Wi-Fi **cannot track two users separately** in the same space. - LiDAR or vision systems are limited by field of view.	- Our solution simultaneously tracks **multiple users** with precise location and movement updates, laying the ground to create distinct audio zones (bubbles) for each user.

Table 2: Comparison of related indoor localization systems.

Existing companies

Several companies, such as Pozyx and Kinexon, offer UWB-based RTLS (Real-Time Location Systems), but their implementations are often limited to asset tracking and warehouse management rather than consumer applications. Their systems are typically point-to-point to a central anchor, and do not offer real-time positional data in a dynamic environment, nor do they provide the scalability and adaptability needed for B&O's seamless audio experience.

Apple's U1 chip, used in devices like iPhones (11 and later), AirTags, and some HomePods, have only been used for point-to-point localization (e.g., AirTags, iPhone location tracking) rather than a full networked solution. Google Pixel's UWB applications are similar. These solutions work well for static or small-scale use cases (e.g. locating a lost item, pairing devices), but they don't offer the open, flexible, real-time indoor grid necessary for scenarios like "Follow-me" audio or adaptive speaker systems across multiple rooms.

Chapter 3: **Proposed Concept**

Design Overview

To provide the necessary information to our users, the solution should provide complete location information of devices in the network with minimal setup. We will henceforth refer to this network as the “Grid”. When a new hardware node is added to the system, it establishes its location relative to the other anchors, adding to the “Grid”. The user can also be placed inside the grid by tracking their iPhone. This final complete network is then returned via an API request.

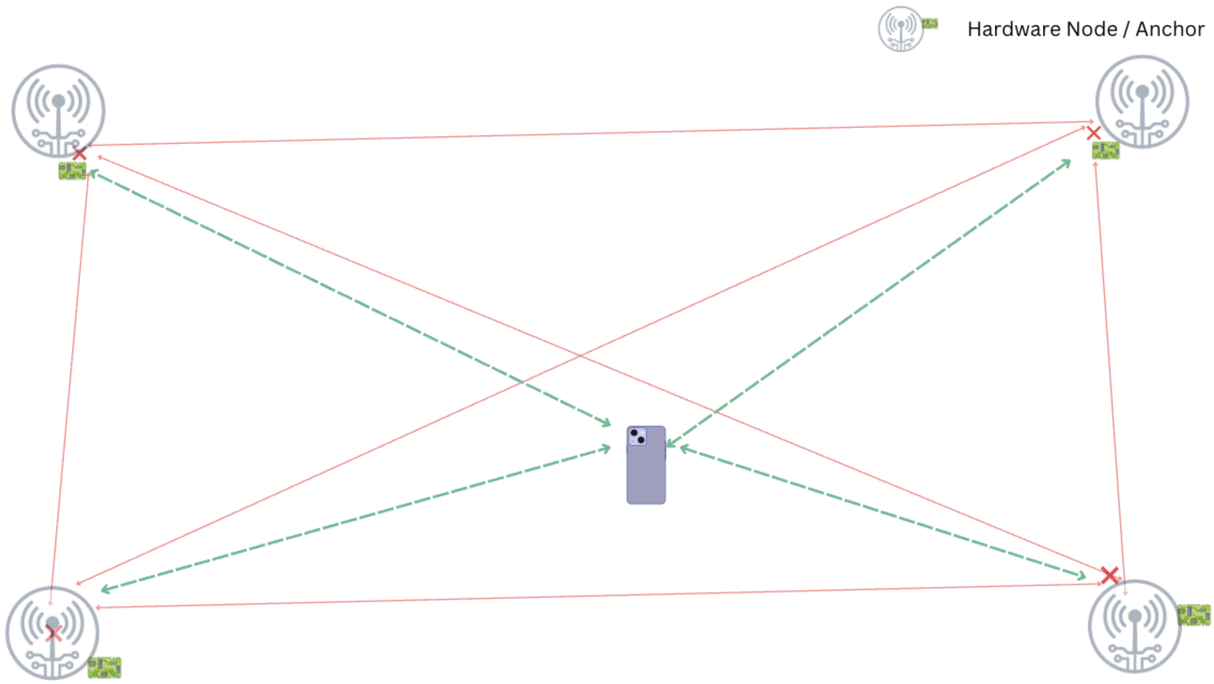


Figure 1: Overall Hardware Sub-System

Requirements of an all encompassing system

1. High accuracy <20cm
2. Communication between different anchors over the local network
3. Handshaking between anchors and iPhones
4. Form a complete grid of different anchors in a physical space
    1. Determine displacement vectors between anchors
    2. Form a global grid of anchor locations based on various displacement vectors
5. iPhone used as a Transmitter

    1. Handshaking with multiple anchors
    2. Determine the phone location (transmitter) on the same grid of anchors
6. API to enable querying of data internally / externally from the network

Hardware and Software Framework Decision Making

They're 3 main systems we will need. UWB, Computation and communication.

UWB

As we are tracking an iPhone, we are fundamentally limited to hardware approved under the Made-For-Apple certification. We also intend to hit the commonly-advertised 20cm accuracy of UWB.

Regarding the UWB chip options, the requirements are 3D positioning capabilities, communication through UART, and a bluetooth antenna. This results in our chosen Murata Type2BP-EVK board. It is also notable that B&O is currently using NXP chips. (See appendix B)

Computation

Our system must be able to solve for the user location given over six nodes and we need to have enough computational headroom to do this alongside communication with other anchors and the UWB board.

The RPi Zero provides us all the hardware requirements for communication over Wi-Fi, NXP board, as well as computational requirements. (more details in appendix)

Communication

The system must be able to adapt to dynamic connections dropping and new anchors being added, working in a decentralized manner so that it does not depend on any particular node.

Hence we decided to adopt a Publisher-Subscriber architecture, this is enabled by frameworks such as MQTT which is highly adaptable allowing us to prototype quickly.

Chapter 4: Technical Implementation, Feasibility, & Limitations

Overall System Diagram

This is the proposed system architecture for our UWB hardware and middleware, showing how they work together - to calculate and pass locational information to an API which can then control the user experience.

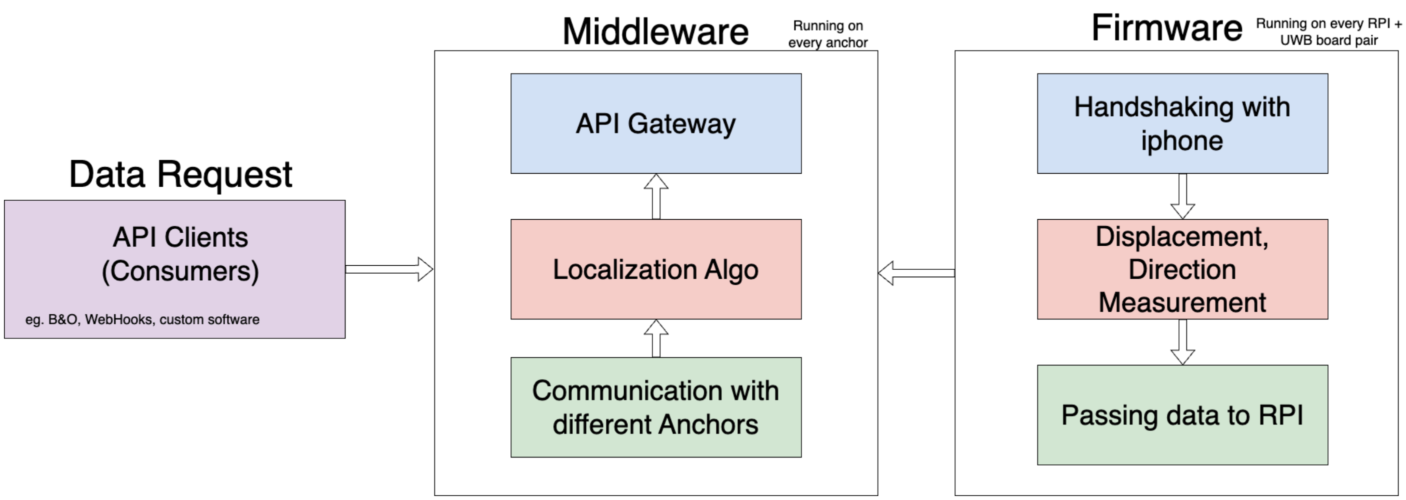


Figure 2: System architecture of UWB firmware and hardware

We plan to run this Middleware layer on every node (RPI + NXP UWB hardware). Although this introduces slight computational redundancies in the system, it aims to increase reliability. This is in line with our requirements, B&O's, as well as that of the larger IoT industry.

Firmware

A handshake is formed between the UWB hardware and the iPhone. Then, displacement data is generated by the firmware. Lastly, this data is passed to the RPi to compute the location on the global grid.

1. **Handshaking with iPhone**

To start a UWB session with an iPhone, Apple first requires an exchange of tokens and configuration data. This is done over any non-UWB data channel. The team will be looking into using MQTT and Bluetooth for this exchange, and deciding on one based on reliability.

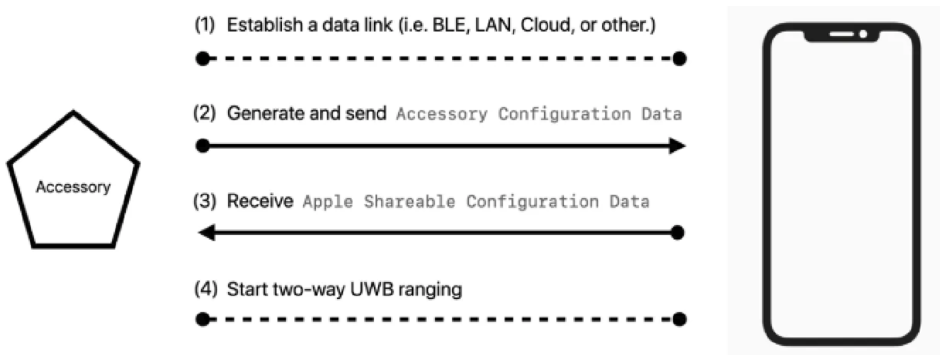


Figure 3: Establishing secure handshake between NXP UWB chip and iPhone

One significant challenge with running the middleware is holding multiple simultaneous sessions. This entails opening, handling and closing multiple sessions. We solve this problem by creating multiple instances of accessory objects with their own logic.

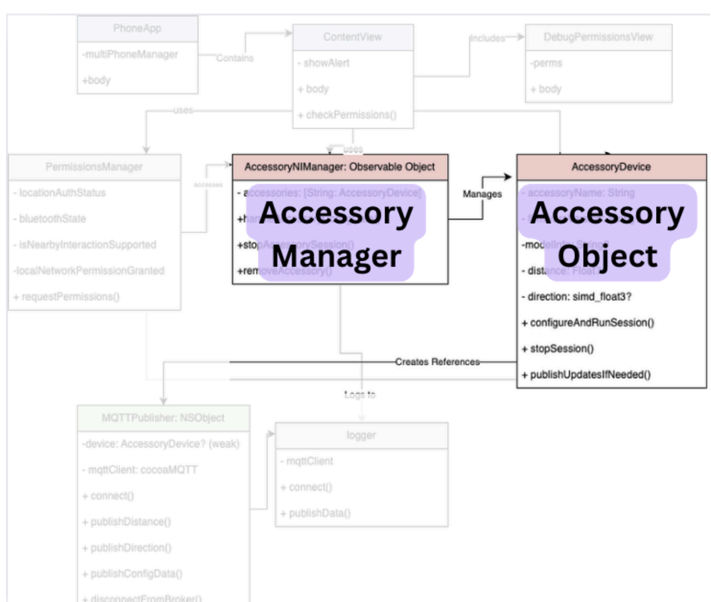


Figure 4: UML of the iPhone Application in Development
(Full UML in Appendix C)

2. Displacement, Direction Measurement

In order to obtain the measurements, the phone must be set up as a transmitter and the board a receiver. The displacement of the phone is obtained by the board using time of flight (TOF) and Angle of arrival (AOA) measurements. The logic for the above is included in NXP's Software Development Kit.

3. Passing data to RPI

The UWB board will be connected to the RPi through UART. This will serve as the channel to pass location data to the RPi for more heavy computation. This will also serve as the data channel for "1. Handshaking with iPhone".

Middleware

The Middleware is responsible for taking in locational data from multiple nodes, then transforming it into the complete graph which we refer to as the Global Grid.

1. Communication with other Anchor devices

To establish a way to distinguish between different anchors and data streams, the communication between different Anchors will be done through MQTT. This allows us to communicate with all devices on the network, overcoming the limitation of bluetooth having a small range.

2. Localization Algo

To meet our requirement of providing complete data with the "context" of a complete grid and all other anchors on it, having a complete localization system is critical. Two main problems surface. First, all data read is not exact. Second, we need to create a "grid" then place the user inside.

We will first solve for displacement vectors between each node pair then put all nodes together to form a complete graph.

a. Forming displacement pairs

When taking measurements between 2 different nodes, there is bound to be deviations in the distance and direction data generated.

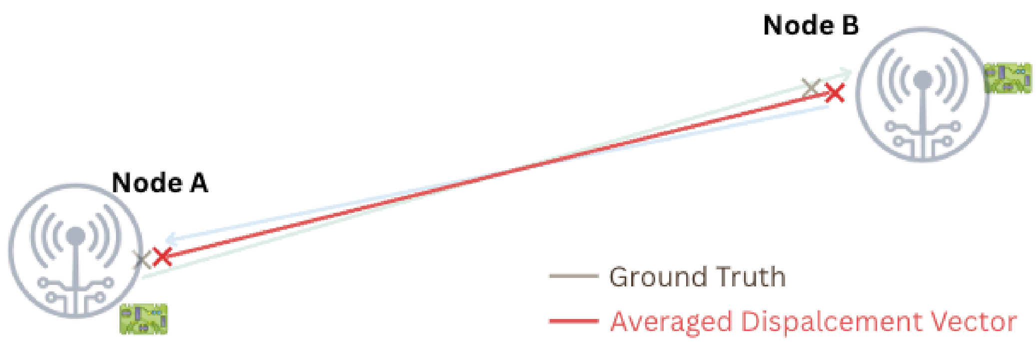


Figure 5: Two Anchors Detecting Relative Vectors from Each Other

The ground truth vector likely lies somewhere between the 2 vectors. We can "tune" which speaker to trust more and take a weighted average or take a simple average first for a proof of concept. This process gives us the "edges" of the graph.

b. Forming a complete graph

Similarly, we do not expect the edges to be perfect. It can be expected that A + B ≠ C. The loops do not close perfectly. However, a closed graph is needed to establish the complete grid.

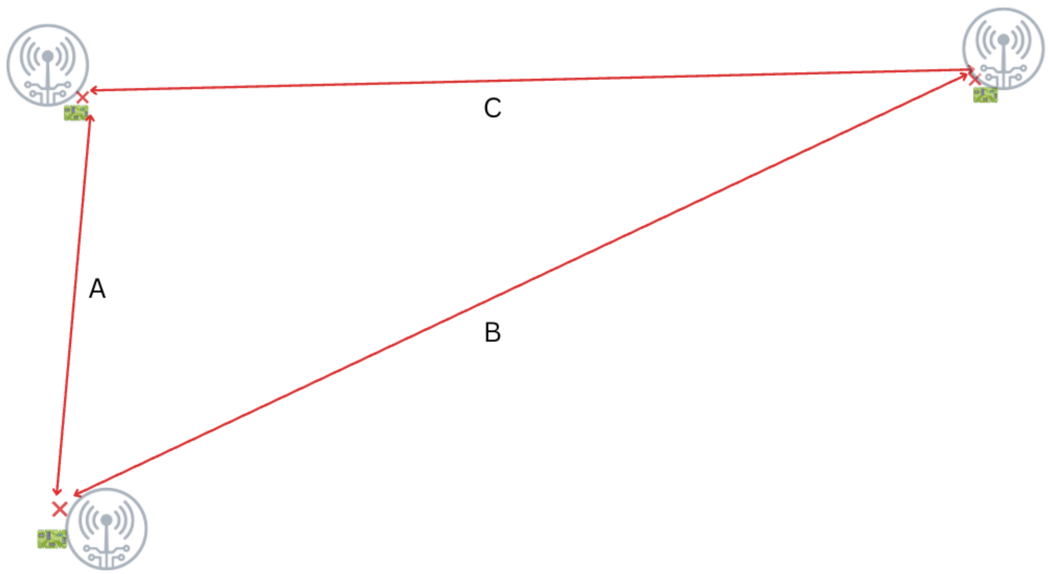


Figure 6: Three Anchors Detecting Relative Vectors from Each Other

Graph Pose Optimization (PGO) can then be employed to distribute the error across the whole graph. This noise function accounts for both directional and distance inaccuracies.

The formula is as follows:

$$\min_{\vec{x}_1, \vec{x}_2, ..., \vec{x}_n} \sum_{(i,j) \in E} \left\| (\vec{x}_j - \vec{x}_i) - \vec{d}_{ij} \right\|^2$$

whereby d_{ij} is our measured (imperfect) relative vector, computed as distance × direction. X_i *and* X_j solved represent the estimated positions for each node pair [10].
(See appendix D for a more rigorous breakdown)

3. API gateway

Lastly, we need to be able to deliver the data from the PGO effectively to the user. This could be querying the data from within the network, or externally.

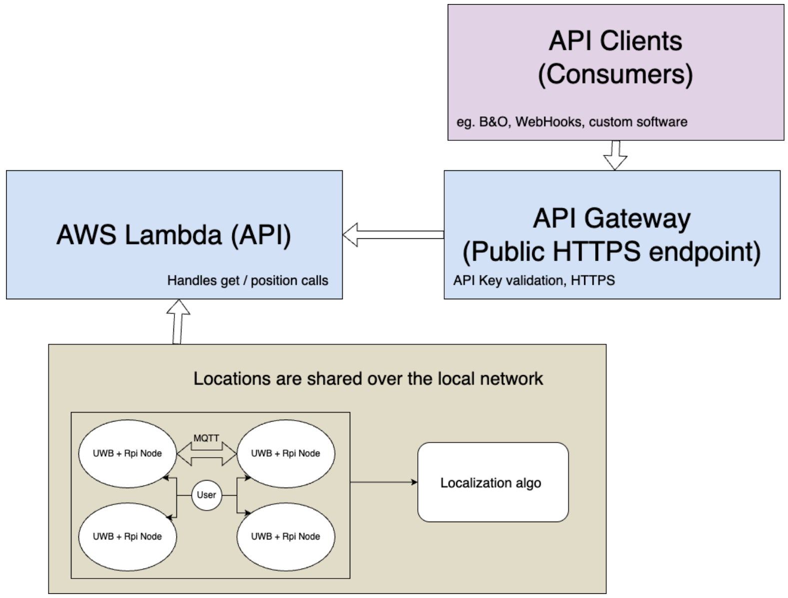


Figure 7: UWB Data Flow through an API Gateway

Any device on the local network will be able to query the system through MQTT. For external queries, eg. IFTT, B&O web based services or training of AI, we can manage the query using AWS Lambda. This allows our system to be as accessible by the user locally while not completely exposing our system to the internet.

Chapter 5: Development Plan (Next Steps & Future Roadmap)

Sem 1 Development:

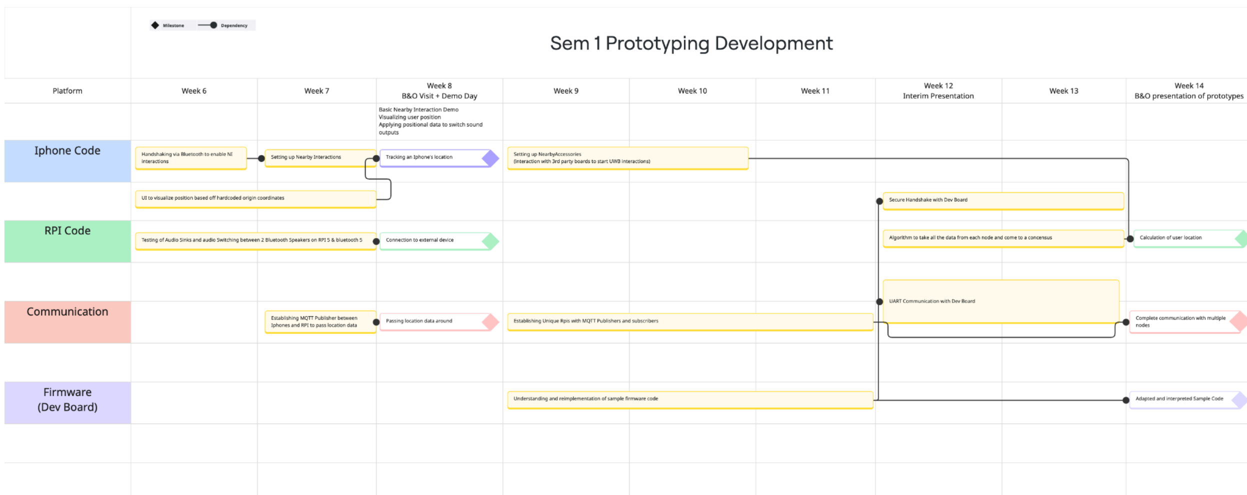


Figure 8: Gantt Chart for the First Semester Development

Over the Semester, our team set out to overcome two of the main challenges for the final implementation, namely establishing handshaking between multiple iPhones with UWB test boards, as well as holding multiple simultaneous UWB sessions. This mostly involved adapting NXP's firmware as well as learning and writing Swift code using Apple's NI Accessory Framework.

Additionally, we have also experimented and developed a prototype of what an application of a seamless experience might look like for B&O, doing a demo using the NInteraction framework on two iphones, showing the switching on audio. One notable limitation during the demo was that the iphone was in the receiver configuration, reducing the angle in which it can receive from.

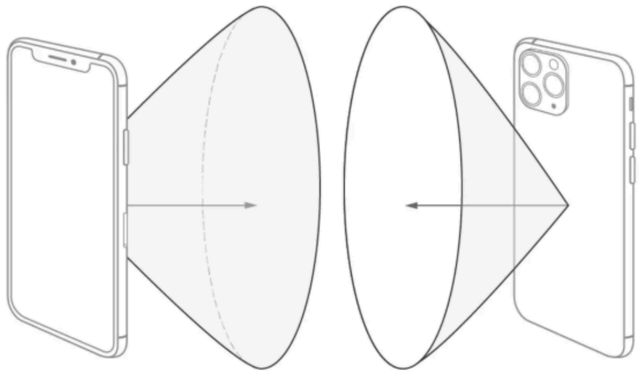
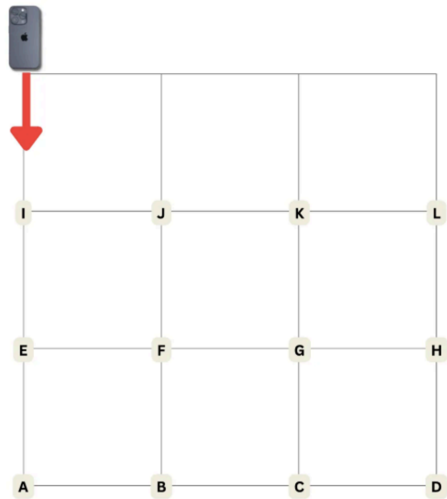


Figure 9: iPhone Spatial Range as a UWB Receiver [11]

Lastly, we also collaborated with a teammate to obtain preliminary data for the accuracy of the current NXP chip with a single anchor. The accuracy of the NXP chip with one anchor, stated below, can be used as a benchmark for the accuracy of the system. However, testing with larger distances has yet to be conducted.



	Actual Distance	Tracked Distance	Deviation (%)
A	135	136	0.740740741
B	142.3	143	0.491918482
C	162.25	159	2.003081664
D	190.92	187	2.053216007
E	90	90	0
F	100.62	95	5.585370702
G	127.28	127	0.219987429
H	162.25	161	0.770416025
I	45	47	4.444444444
J	63.64	65	2.137020742
K	100.62	111	10.31604055
L	142.3	147	3.302881237

$$\left|\frac{(Tracked - Actual)}{Actual}\right| \times 100\%$$

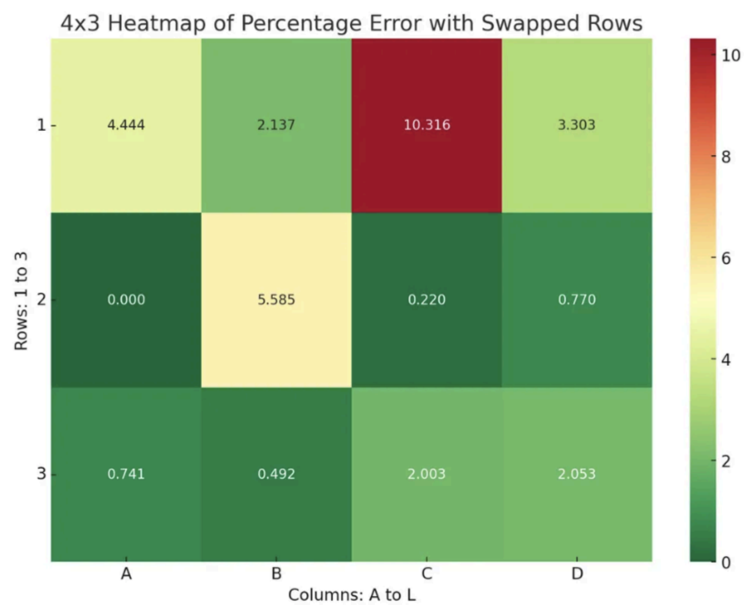


Figure 10: Preliminary Heat Map Testing showing Deviation of Readings from Ground Truth Location, with an iPhone as the Transmitter and a NXP 2BP Muratta Board as the Receiver

Sem 2 Development

In semester 2 we have broken the task down into 2 main parts: developing the middleware, then the scenario implementation.

1. **Developing the middleware**

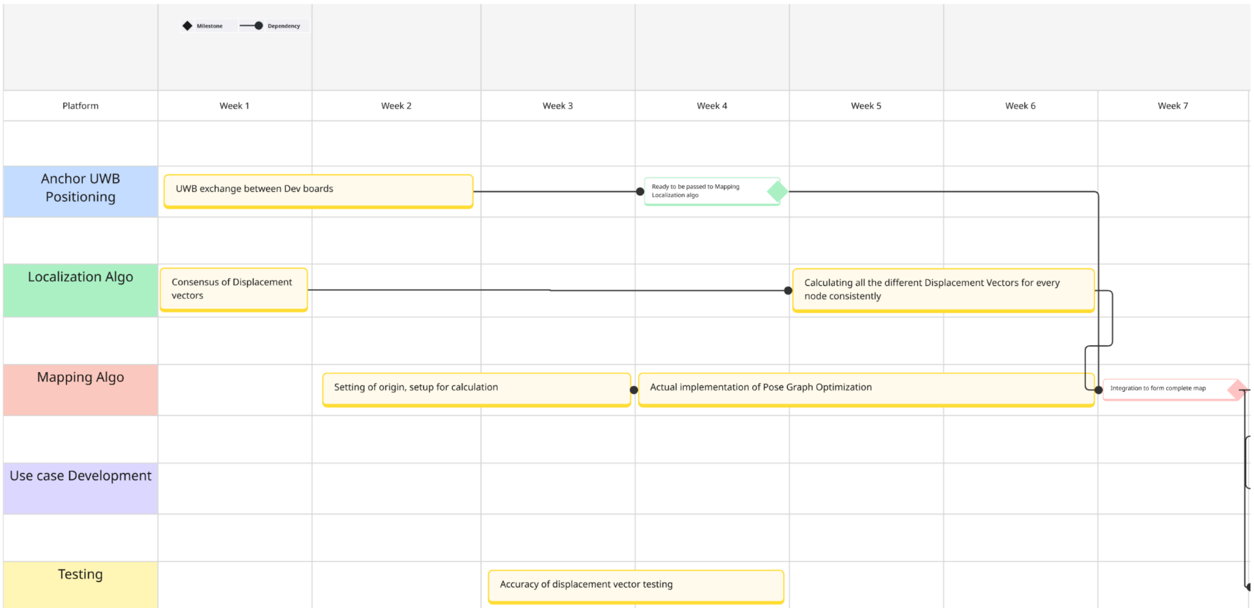
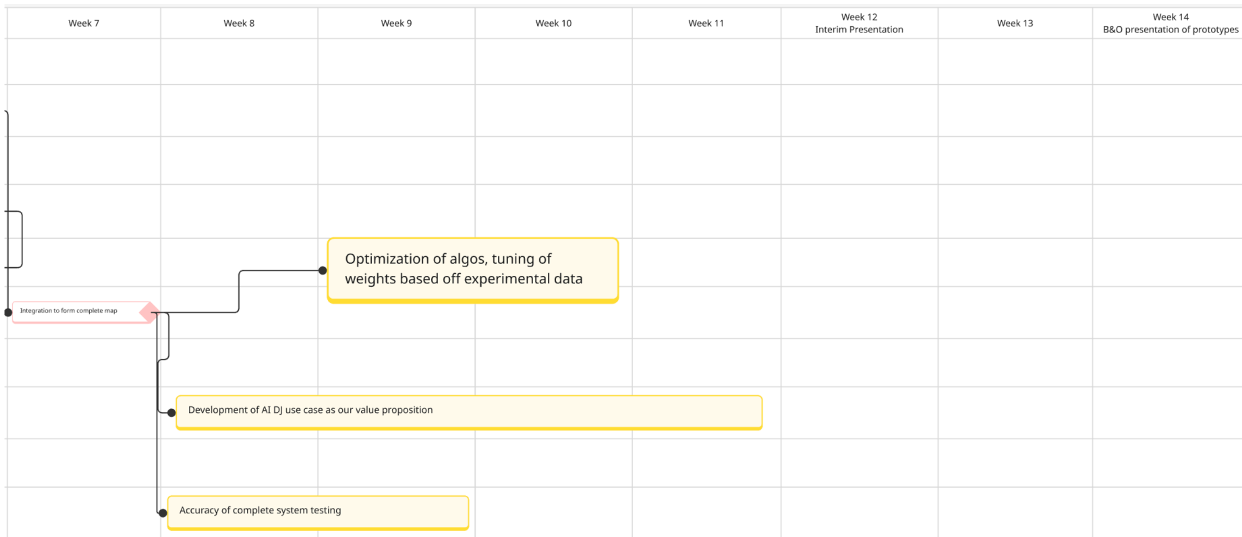


Figure 11: Gantt Chart for the First Semester Development

For the middleware to be completed, the different steps are dependent on each other. Given our small team size, we will work on the tasks sequentially then put it all together to form the complete middleware in Week 7.

At the same time, testing will be done in parallel to ensure the robustness of each subsystem

2. Applying the middleware



Our ideal application we want to develop is the location based music selection as it showcases the advantage of the granularity and context of the information, directly tying into our value proposition.

While the application is being developed, we also hope to dedicate effort to fine tuning the middleware to achieve the best possible results. Extensive testing will also be performed, this is detailed in the testing plan.

Testing Approach

1. UWB accuracy and reliability will be tested using a grid. (Potential testbed in appendix.) At every junction on the grid, a calculation will be made between the actual and detected values of distance, and direction
    a. To test **reliability**, each reading will be conducted 20x, so as to compare the standard deviation.
    b. To compare the **different chip** manufacturers' consistency, conduct the above experiments using the following transmitter-receiver combinations:
        - NXP - NXP ranging
        - Iphone - NXP ranging
    c. To compare the consistency when the iPhone **transmitter is moving**, vary velocity to simulate a user walking around at different speeds
2. **Effect of obstructions** on connection strength and accuracy

    - Across obstacles: walls, pillars, humans.
    - For collinear obstructions, measure deviation in |actual - detected| distance and direction
    - If collinear measurement deviation is significant, find the angle at which it is within acceptable range of deviation

3. To decide on using MQTT or Bluetooth **channel for establishing handshake** (exchange of tokens and configuration data), we will be establishing both connection types 20 times each, to compare:
    - Effective range
    - Time for handshake

Chapter 6: Reflection

Ji Yong

This project has been a fun challenge so far, especially given the small team size. The additional requirement of presenting demo prototypes in weeks 7 and 12 of this first semester posed some extra pressure too. The scoping of the problem was particularly difficult because B&O initially pushed for using UWB technology straight away, rather than framing a specific problem we needed to solve. We spent a lot of time refining our approach to develop a problem-solution pair that leveraged UWB effectively. While Hong Yi focused on the iPhone app and learned Swift along the way, I concentrated on the Raspberry Pi setup, tackling issues like audio sink switching. Our Week 7 prototype, which enabled audio transitions from speaker to speaker based on proximity, was well-received by B&O's product designer. We see it as steps taken in a positive direction. I'm excited about our next prototype in Week 12, where we'll expand our UWB setup to offer 360-degree coverage, rather than a conical receiving range like in Week 7. This experience has deepened my understanding and interest in real-time positioning systems; in Semester 2, we'll enhance the system by eliminating hardcoded speaker positions for a dynamic live grid. Despite the challenges, this project has been a great learning experience, and I'm optimistic about the direction we're heading in.

Hong Yi

Across the past semester, this project has been a significant challenge. After developing the idea of the applications of UWB and our true value being middleware, we were able to progress much faster, focusing much more on the value proposition instead of audio application (which we feel B&O have an upper hand anyways). One thing that struck us at the start was how software heavy our project was going to be. Especially given our background this was a good learning experience. Coding a whole implementation for multisession UWB interactions taught me lots and we are currently in the process of refining it for our next demo with B&O. This will be a significant milestone for us. All in all, I am very excited to see what is to come out of our system and explore as many applications of our middleware as possible, creating our killer features to further justify the value of our middleware.

Anitej Datta

From the start of the course, we were involved in exploring a variety of ideas on how we might solve the problem of creating iconic audio experiences for a partner like Bang and Olufsen. Interacting and engaging with peers' ideas helped me appreciate what types of potential innovations could be added to an already high quality product catalog. Eventually, our team decided to settle on the UWB based software ecosystem, which can not only serve B&O's needs but also appear as a potential market competitor in the IoT space. We were able to establish the unique value of UWB over other competing technologies through hands-on experimentation during a demo in Week 8.

I am quite excited to contribute to the development of this software solution which will require thorough and sound understanding of the underlying hardware capabilities to implement the planned features. Through this project, I believe I can learn a lot of useful skills as an aspiring Electrical Engineer looking forward to a career in areas such as embedded systems and robotics, among others.

References

[1] Apple Inc., "Nearby Interaction with UWB," Apple Developer. Accessed: Apr. 21, 2025. [Online]. Available: https://developer.apple.com/nearby-interaction/

[2] "What is Beolink Multiroom?," Bang & Olufsen Support. Accessed: Apr. 21, 2025. [Online]. Available: https://support.bang-olufsen.com/hc/en-us/articles/4411572883089-What-is-Beolink-Multiroom

[3] "The History of Beolink (B&O Multiroom)." Accessed: Apr. 21, 2025. [Online]. Available: https://www.bang-olufsen.com/en/sg/story/beolink-history

[4] "Beoconnect Core, Turntable and TV Speaker Hub - Bang & Olufsen." Accessed: Apr. 21, 2025. [Online]. Available: https://www.bang-olufsen.com/en/sg/accessories/beoconnect-core

[5] F. Milano *et al.*, "BLE-Based Indoor Localization: Analysis of Some Solutions for Performance Improvement," *Sensors*, vol. 24, no. 2, Art. no. 2, Jan. 2024, doi: 10.3390/s24020376.

[6] M. Orfanos *et al.*, "Testing and Evaluation of Wi-Fi RTT Ranging Technology for Personal Mobility Applications," *Sensors*, vol. 23, no. 5, Art. no. 5, Jan. 2023, doi: 10.3390/s23052829.

[7] G. Retscher, "Indoor Navigation," in *Encyclopedia of Geodesy*, Springer, Cham, 2016, pp. 1–7. doi: 10.1007/978-3-319-02370-0_9-1.

[8] M. Özcan, F. Aliew, and H. Görgün, "Accurate and precise distance estimation for noisy IR sensor readings contaminated by outliers," *Measurement*, vol. 156, p.

107633, May 2020, doi: 10.1016/j.measurement.2020.107633.
[9] H. Guo, S. Song, H. Yin, D. Ren, and X. Zhu, “Optimization of UWB indoor positioning based on hardware accelerated Fuzzy ISODATA,” *Sci Rep*, vol. 14, no. 1, p. 17985, Aug. 2024, doi: 10.1038/s41598-024-68998-0.
[10] X. Li and H. Ling, “PoGO-Net: Pose Graph Optimization with Graph Neural Networks,” in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada: IEEE, Oct. 2021, pp. 5875–5885. doi: 10.1109/ICCV48922.2021.00584.
[11] “Initiating and maintaining a session,” Apple Developer Documentation. Accessed: Apr. 21, 2025. [Online]. Available: https://developer.apple.com/documentation/nearbyinteraction/initiating-and-maintaining-a-session

Appendix:

A) Summarised Survey Results

The team conducted a survey, consisting of 17 subjects, for a preliminary gauge.

53% (9/17) own at least one IoT device, but all interactions are phone- or voice-triggered.
None of the participants have exploited spatial context in their smart home systems right now.

But when introduced to the idea of location-based automations down to 20cm accuracy, **100%** of respondents were inclined to leverage on the feature if it becomes accessible.

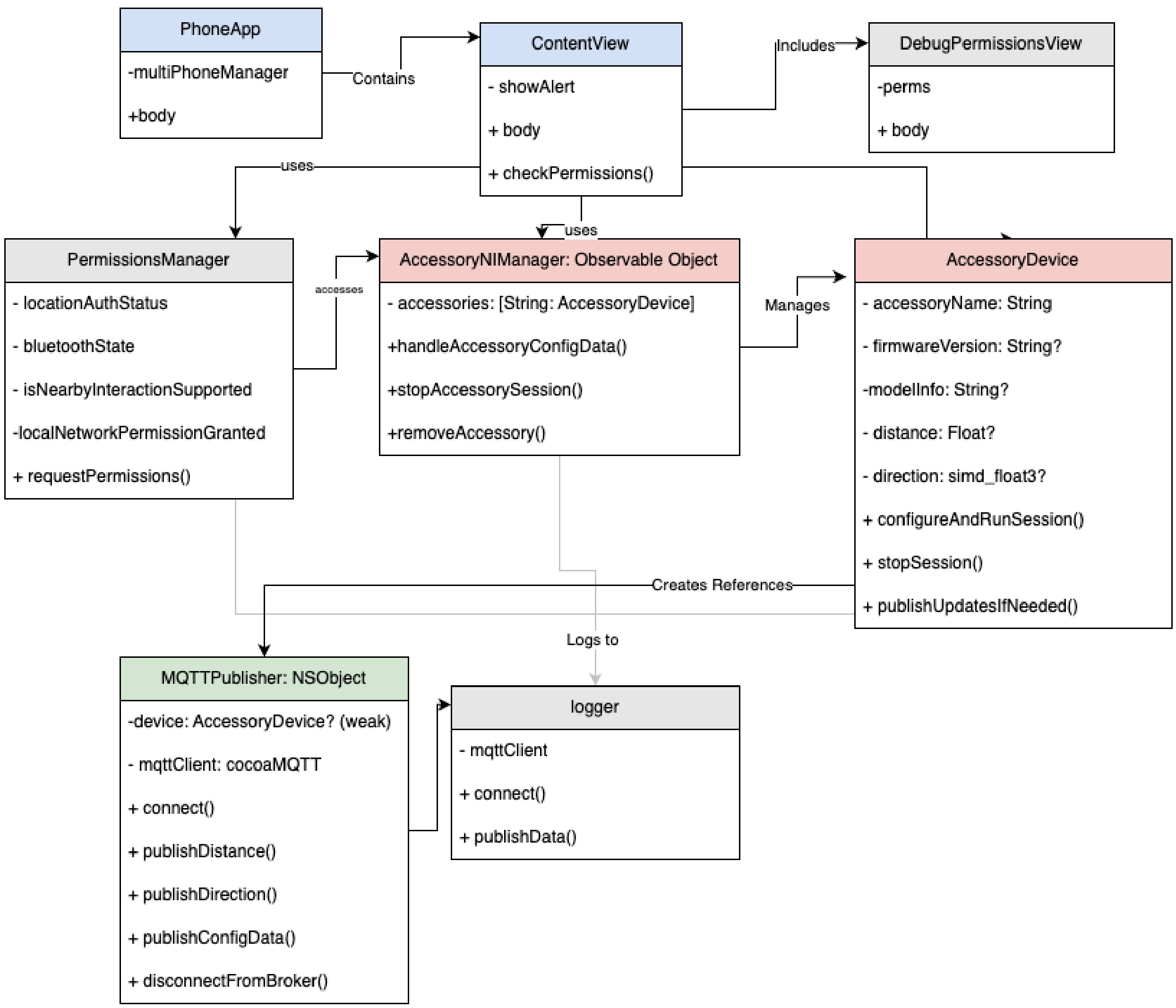
When introduced to the feature of a “A Spotify PLaylist DJ that knows which room I’m in” (Chapter 1, Problem Situation 6), on a likert scale of 5, **76%** (13/17) rated the idea with a “4” or “5”.

B) UWB Board Comparison Chart

Development Kit	UWB Chipset	Advantages of UWB Chipset	Bluetooth Support	Interfaces	Antenna Configuration	Power Supply Options	Notable Features
Murata Type2BP EVK	SR150	Supports 3D Angle-of-Arrival (AoA) measurements; suitable for anchor devices	QN9090 BLE MCU	USB, UART, SWD	Onboard UWB antenna, External BT antenna	USB, COM port	Pre-flashed with Apple Nearby Interaction example code; compact design
Murata Type2DK EVK	SR040	Optimized for low-power tag applications; suitable for battery-operated devices	QN9090 BLE MCU	UART, SWD	Integrated UWB/BLE onboard antenna	Coin-cell battery, USB	Designed for UWB tag/tracker development; low power consumption
MobileKnowledge MK UWB Kit	SR150/SR040	Combines advantages of SR150 and SR040; supports both anchor and tag configurations	QN9090 BLE MCU	USB, UART, Arduino headers	3D AoA Antenna Board	USB, Battery, External power	Includes both anchor and tag reference designs; comprehensive SDK for iOS/Android
Qorvo DWM3001CDK	DW3110	Fully aligned with FiRa™ PHY and MAC specifications; supports TWR and TDoA	nRF52833 BLE SoC	USB (dual ports), UART, SWD, Raspberry Pi header	PCB UWB antenna	USB, Battery, External power	On-board J-Link debugger; suitable for evaluating UWB technology
Qorvo DWM3000EVB	DW3000	Designed for flexibility with various host processors; suitable for evaluating UWB technology with Nordic platforms	External (e.g., nRF52832DK)	Arduino Shield, SPI	PCB UWB antenna	Via connected development kit	Flexible host support; suitable for integrating with Nordic Semiconductor platforms
SynchronicIT SFM10-KIT	OL23D0	Fully integrated and programmable; allows for host-free UWB solutions	None	SPI, UART, I²C	Not specified	Battery optimized	Easy-to-use hardware and software; suitable for creating host-free UWB solutions

C) App UML Diagram

This UML diagram details the attributes and the methods in each object, this ensures that it conforms to the Nearby interaction framework and is easy to build upon and debug.
This UML diagram does not yet include the handshake for the nearby accessory key, which has yet to be implemented using MQTT.



D) More on Pose Graph Optimization

Because UWB is subject to:

- Distance error (±10–30 cm due to multipath or signal bounce)
- Direction error (from jitter or occlusion in Apple's direction vector)

Hence if we try to build a map just by adding vectors, we'll end up with inconsistencies (like loops that don't close).

PGO corrects this by finding a globally consistent layout of all node positions that **best matches the noisy measurements**.

Optimization formula

We define a cost function to minimize the total squared error between the expected and observed displacements:

$$\min_{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n} \sum_{(i,j) \in E} \left\| (\vec{x}_j - \vec{x}_i) - \vec{d}_{ij} \right\|^2$$

Where:

- $\vec{x}_i$ and $\vec{x}_j$ are the true positions of nodes i and j
- $\vec{d}_{ij}$ is the **measured** displacement (from UWB)
- E is the set of all connected node pairs

Graph Structure

X_i represents a node, an unknown global position of the device

Each edge, given by d_{ij} is the measured displacement.

Subtracting the estimated relative vector with the measured relative vector gives us the error vector. (with both scalar and angle error), squaring the magnitude gives us the error we try to minimize.

$$\left\| (\vec{x}_j - \vec{x}_i) - \vec{d}_{ij} \right\|^2$$

Solving method

We can use a **nonlinear least squares solver** (like Levenberg-Marquardt)

This is suitable for running on the **Raspberry Pi Zero**, due to:

- The small number of nodes (typically < 20)
- The sparse structure of the graph

Since all positions are **relative**, we fix one node (e.g. the first accessory) at the origin:
This prevents the whole solution from "floating" in space.

Further optimization using a weighted cost function:

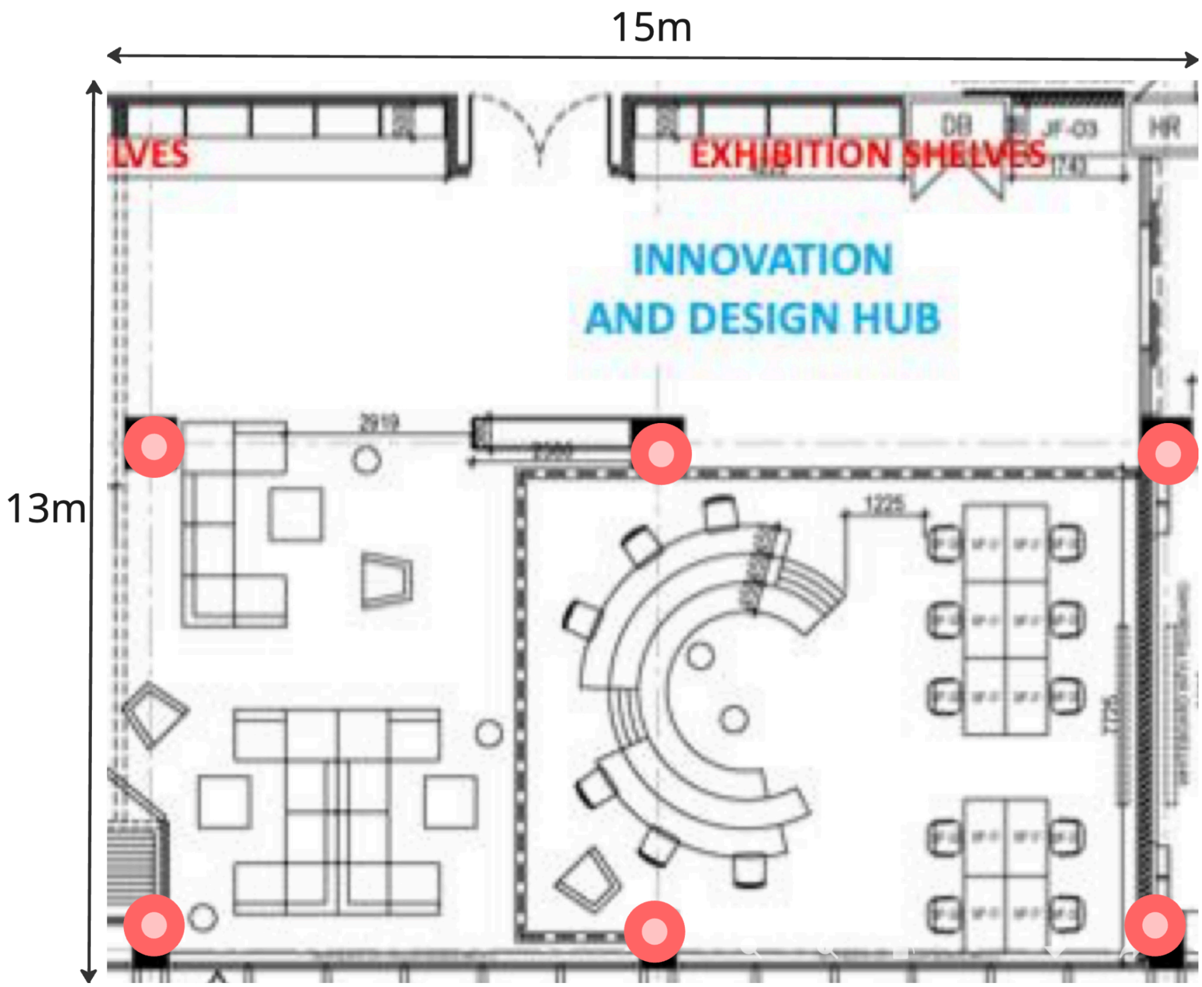
$$\min_{\vec{x}_1,\ldots,\vec{x}_n} \sum_{(i,j)\in E} w_{ij} \cdot \left\| (\vec{x}_j - \vec{x}_i) - \vec{d}_{ij} \right\|^2$$

Where:

- $w_{ij} = \frac{1}{\sigma_{ij}^2}$
- σ_{ij} is the **estimated standard deviation** (error) of the measurement
- Lower σ_{ij} → higher confidence → higher w_{ij}

This extra term allows us to account for potentially “less trustworthy” data sources, eg. we can weigh it to take the closer measurement more than the further one This has to be experimentally tuned.

E) Building a grid testbed:



We can set up the anchors in the IDP Hub as follows. The layout of the hub allows us to be in line of sight of all nodes or to block certain ones out base off where we stand.
For measurements, we can use the charging ports on the ground as landmarks, placing the phone along these points and seeing their % deviation to test for reliability.