

Cross-Platform 2D and 3D Way-finding Mobile Application for Navigation in Industrial Buildings and Parking Lots

Kristina Cormier
Computer Science
Okanagan College
Kelowna, Canada
0009-0004-3783-9704

Helana Jaraiseh
Computing Science
UFV
Abbotsford, BC, Canada
0009-0000-1239-2820

Justin Drenka
Computer Science
UBCO
Kelowna, Canada
0009-0008-6821-8379

Minami Sato
Computer Science
UBCO
Kelowna, Canada
0000-0000-0000-0000

Youry Khmelevsky
Computer Science
Okanagan College
Kelowna, Canada
0000-0002-6837-3490

Abstract—Indoor navigation within large, multi-level environments such as airports, shopping malls, and university campuses remains challenging, particularly when time constraints and unfamiliar layouts are involved. Global Positioning System (GPS) technologies are unreliable indoors, and many existing solutions require continuous network connectivity or costly infrastructure. This paper presents the design and implementation of a cross-platform mobile way-finding application that supports both indoor and outdoor navigation in complex buildings. The proposed system operates in both online and offline modes, employs graph-based path finding, vector-based multi-floor maps, QR-code-assisted localization, and 2D/3D visualizations. The application is designed for scalability, maintainability, and accessibility, offering a cost-effective alternative to infrastructure-heavy indoor navigation systems.

I. INTRODUCTION

Navigating large indoor environments remains a stressful and error-prone experience for users, especially in multi-level buildings with complex layouts such as airports, shopping malls, and university campuses. Traditional way-finding solutions often rely on stationary kiosks that display a map with the user's current location and destination. While these systems can provide route guidance, they are not portable, require users to memorize routes, and are poorly suited for environments with multiple floors and vertical transitions.

To address these limitations, many organizations have adopted digital indoor navigation tools. For example, the YYC Calgary International Airport provides interactive maps that allow users to locate amenities and services [1]. Similarly, the West Edmonton Mall offers a web-based navigation tool that supports multi-level routing using vector maps [2]. Universities have also begun deploying

campus-wide navigation applications, such as CampusGo, which supports indoor navigation without relying on GPS or WiFi beacons [3].

Despite these advancements, many existing systems depend on proprietary platforms, subscriptions, or continuous connectivity, limiting their accessibility and scalability. Furthermore, offline functionality and cross-platform support are often underdeveloped. This work addresses these gaps by presenting a mobile way-finding system that integrates offline-first design, graph-based routing, and intuitive 2D/3D visualization into a single, extensible architecture.

II. EXISTING WORKS

Indoor navigation systems have been extensively studied due to the inadequacy of GPS in enclosed environments and the increasing complexity of modern buildings. Existing research can be broadly categorized into indoor positioning techniques, routing and path-finding algorithms, visualization methods, mobile system architectures, and accessibility-focused solutions.

Survey papers by Fallah in 2013 provide comprehensive overviews of indoor navigation and positioning systems, highlighting challenges related to accuracy, infrastructure costs, scalability, and usability [4]. More recent survey papers by Kunoth in 2020, and ElSheimy in 2021, confirm these are ongoing challenges [5], [6].

Indoor localization methods are classified into infrastructure-based approaches, such as WiFi fingerprinting, Bluetooth Low Energy (BLE), Radio Frequency Identification (RFID), Ultra-Wideband (UWB), and Long Range (LoRa), as well as infrastructure-free

techniques including pedestrian dead reckoning, vision-based localization, and map-constrained positioning according to Bahl in 2000 [7]. More recently, in 2020, Kuntho's survey uses the same classified approaches [5].

QR-code and marker-based localization methods have been proposed as low-cost alternatives to beacon-based systems, as discussed by Cheraghi in 2018 and Santos in 2019 [8], [9]. These approaches enable users to establish their location by scanning known reference points, making them suitable for offline navigation scenarios. However, most QR-based systems do not support continuous real-time tracking and are often limited to small-scale deployments.

Vishwakarma's 2023 research shows that graph-based representations are commonly used to model indoor spaces for routing, with nodes representing locations or intersections and edges representing navigable paths [10]. In 2015, Martinez explains that classical shortest-path algorithms such as Dijkstra's algorithm and A* are widely adopted due to their deterministic behaviour and reliability, and Fernandez backs up this finding in 2021 [11], [12]. In 2024, Cheng's research suggests that A* can outperform Dijkstra's algorithm when heuristic data are available, Dijkstra's algorithm remains effective for medium-sized indoor graphs and offline computation scenarios [13].

Visualization techniques play a critical role in the usability of indoor navigation. Prior work by Fernandez in 2021, and Qiu in 2023, has explored 2D floor plans, stacked multi-floor views, and full 3D visualizations to improve spatial awareness [12], [14]. Although 3D representations enhance understanding of vertical transitions, they introduce challenges related to rendering performance and interaction accuracy. As a result, hybrid systems that allow users to switch between 2D and 3D views are increasingly common.

According to Cheraghi in 2018, and Cheng in 2024, mobile implementations emphasize cross-platform development and offline-first architectures [8], [13]. Santos's 2019 work suggests that local databases such as SQLite are frequently used to cache navigation data, synchronizing periodically with remote servers to ensure consistency while maintaining offline functionality [9]. These designs improve robustness in environments with unreliable connectivity.

Martinez discusses, in 2015, [11] accessibility-focused research on indoor navigation for users with disabilities, particularly individuals with visual impairments. Such systems often integrate audio guidance, obstacle-aware routing, and assistive interfaces, though many require specialized hardware or real-time positioning infrastructure, limiting large-scale deployment.

Recent research extends traditional localization by exploring federated learning and AI-based models to handle privacy and multi-floor scalability in indoor localization [15]–[19]. Additionally, comprehensive reviews of wireless

technologies summarize current techniques and their trade-offs in terms of accuracy, cost, and complexity [20].

In contrast to existing solutions, the system presented in this paper combines QR-assisted localization, graph-based routing, vector-based multi-floor maps, and offline database synchronization within a cross-platform Flutter application. This approach avoids reliance on costly infrastructure while supporting intuitive multi-level navigation and offline operation.

III. SYSTEM ARCHITECTURE AND DESIGN

A. System Architecture

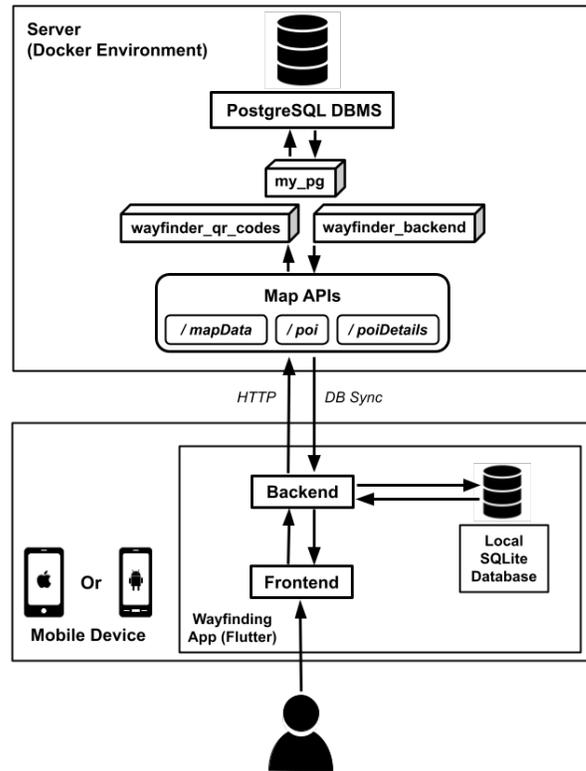


Fig. 1. System Architecture of Mobile Wayfinding App

The illustration shown in Fig. 1 depicts the overall architecture and design of the system. It consists of a mobile client and a containerized back-end. Using an Apple or Android mobile device, the user interacts with the front-end interface (UI), which communicates with the back-end. The back-end is responsible for connecting to the containerized server and for local storage on the user's personal device using a SQLite database. The back-end uses an Application Programming Interface (API) to interact with the PostgreSQL database.

The major subsystems include mobile client, back-end, and external resources. Mobile client includes the UI, map rendering, and path-finding. Data provider and sync service

are vital to the back-end services. External resources consist of map assets and QR codes.

Users input their start location or current location, and their destination, the shortest path is determined using Dijkstra’s algorithm, and a visualization of that path on an appropriate map is displayed. This visualization can be displayed in both 2D and 3D views using a toggle button. This works for both online and offline options, as the maps are stored locally.

B. Database Architecture and Design

The database architecture was designed to support offline-first navigation while remaining scalable for future deployment. A relational schema models buildings, floors, nodes, edges, and points of interest, enabling efficient graph-based routing across multiple levels. The back-end uses PostgreSQL as the authoritative data store, while SQLite is employed on the mobile device to cache navigation data locally. At application startup, a synchronization service retrieves the latest data from the back-end and initializes the local database, enabling the system to operate fully offline. The back-end is containerized using Docker to ensure consistent deployment and facilitate scalability. This design allows the system to scale across multiple buildings and campuses without disrupting existing functionality.

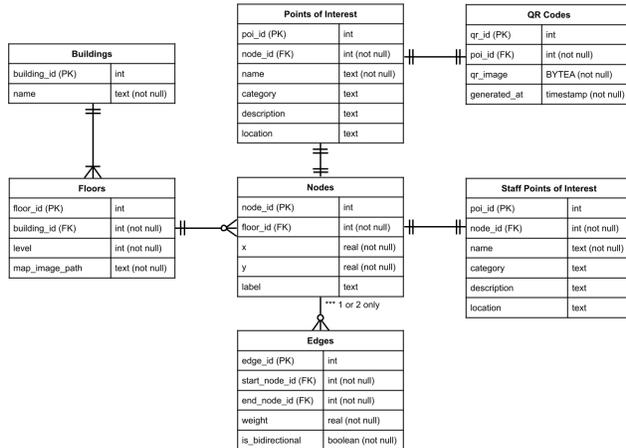


Fig. 2. Database Schema of Mobile Wayfinding App

Fig. 2 shows the database schema. It is containerized, shareable, and flexible for future growth. It is intended to scale with ease, allowing future growth without disrupting existing functionality. Furthermore, it is containerized using Docker. This allows the same database to be used for both native and cross-platform approaches.

The project uses PostgreSQL for online services and SQLite for offline solutions. SQLite enables offline service by storing data on the device. Once the device is connected to the network, it synchronizes with the remote database, updates its data, and receives the latest changes from it.

Although the database is flexible, scalable, and portable, future work is planned to improve it.

C. Systems Design

The system design emphasizes maintainability, scalability, and usability by leveraging a modular, cross-platform architecture and an offline-first service model. The selected technology stack enables efficient deployment across mobile platforms, supports reliable operation in environments with limited connectivity, and allows the system to scale to additional buildings and datasets without requiring significant architectural changes.

1) *Cross-platform*: The application is implemented using Flutter to support cross-platform development. All features were tested on simulators, emulators, and physical Android and iOS devices to ensure consistent behaviour and performance across platforms.

2) *Offline Service*: The application provides full offline functionality. At startup, a synchronization service (SyncService) retrieves the latest map and routing data from the back-end, re-initializes the local database, and stores the updated data in SQLite on the mobile device. This process creates a local offline mirror that MapService can access instantly. Once the data has been loaded, the application operates fully offline without requiring network connectivity.

3) *Search and List-based Navigation*: The system provides both a searchable list of locations and an auto-suggestion search bar to allow users to select their current location and destination. The selected locations are used as inputs to the path-finding service. These complementary selection methods support users who may not know the exact name of a location while still enabling quick and precise navigation.

4) *Interactive Map*: The interactive map allows users to explore and navigate the environment visually. Users can switch between floors using a drop-down menu, enabling route inspection and exploration beyond the currently active level. Destinations and current locations can be selected through the search interface, the list of points of interest, or by scanning a QR code. After selecting both locations, the user initiates route generation, and the computed path is rendered on the vector-based map. Users may update their destination or current location at any time by selecting a map node and choosing the appropriate action.

Users choose their destination from the list of points of interest, or type the location name in the search bar and select the location that appears under the search box. After that, they chose their location as they did

for their destination. They can also use the QR code to set their location. After they click the button “Find a path”, the vector image of the map takes over the screen, and it shows a path from the user’s current location to their destination. Once they choose their current location, they can press the button “Find a path.” This uses the destination and the user’s current location to generate the path, which will appear on the vector image map. If the user wants to change their destination or current location, they can do so by clicking ”set as destination” or ”set as current location” after selecting the location node. Outdoor navigation uses pedestrian-aware routing features such as crosswalk detection and sidewalk-based path planning.



Fig. 3. Floor 1

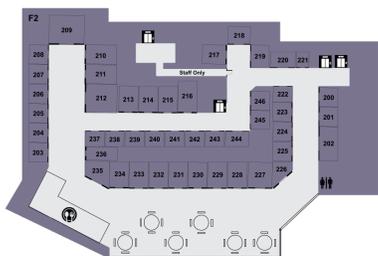


Fig. 4. Floor 2

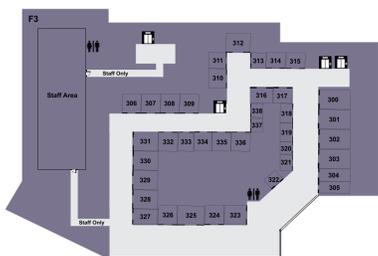


Fig. 5. Floor 3

IV. THE PROTOTYPE IMPLEMENTATION AND TESTING

The prototype was deployed using a containerized back-end environment and tested across multiple mobile platforms. Secure remote access mechanisms, including VPN

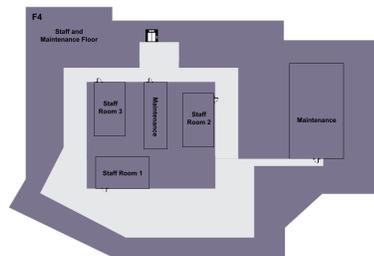


Fig. 6. Floor 4

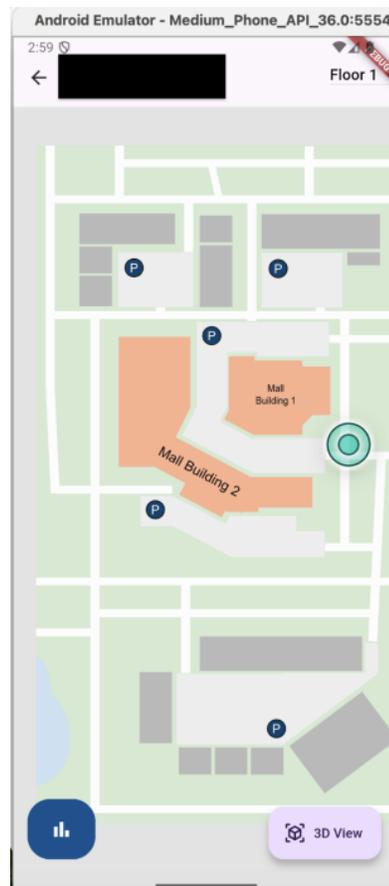


Fig. 7. Android UI with map view of exterior, including parking

and SSH, were used to deploy and maintain the back-end services. Performance evaluation focused on path-finding latency, map rendering responsiveness, and offline functionality. The system consistently generated routes within two seconds for medium-sized indoor graphs, both online and offline. Testing scenarios included multi-floor navigation, QR-based location initialization, and transitions between indoor and outdoor environments.

V. LIMITATION

The current prototype exhibits limitations in 3D interaction accuracy, particularly in detecting tap events due to stacked-floor coordinate transformations. Additionally,

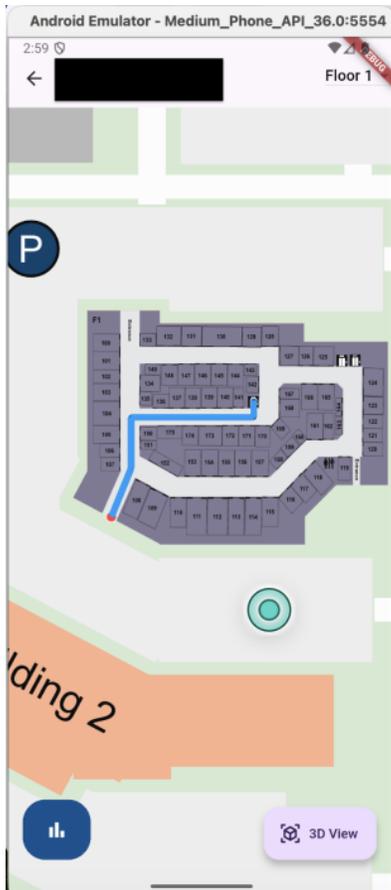


Fig. 8. Android UI with map view of Floor 1, with a path

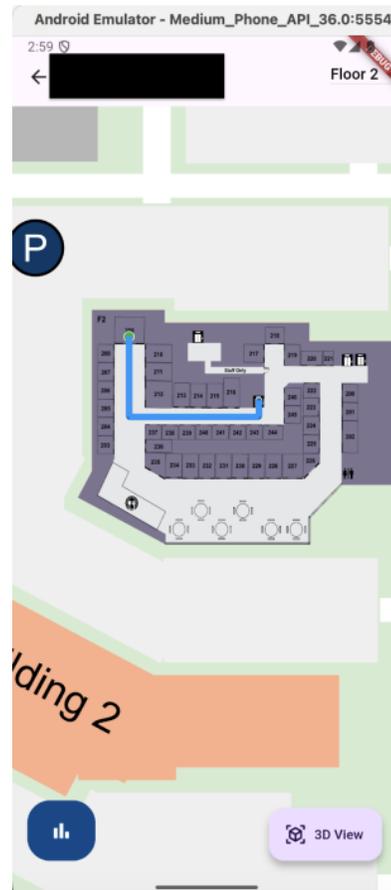


Fig. 9. Android UI with map view of Floor 2, with path

rapid zooming may result in rendering glitches. Database re-initialization issues were observed in specific offline synchronization scenarios.

VI. FUTURE WORKS

Although the proposed system demonstrates the feasibility of a cross-platform, offline-capable indoor way-finding application, several enhancements remain for future investigation and development.

A. Real-Time Indoor and Outdoor Location Tracking

One of the most significant areas for future work is real-time user location tracking. The current system relies on user-selected locations and QR-code-assisted positioning, which provides a low-cost and reliable solution but does not support continuous tracking. Future implementations may integrate Bluetooth Low Energy (BLE), Ultra-Wideband (UWB), WiFi fingerprinting, or hybrid sensor-based approaches to enable real-time positioning while maintaining acceptable accuracy and privacy.

B. Path-finding Optimization and Intelligent Routing

The current prototype employs Dijkstra's shortest-path algorithm, which performs well on small-to-medium indoor graphs and supports both online and offline operation.

Future work may explore the use of A* search with heuristics, as well as AI- and machine-learning-assisted routing techniques, to optimize performance further and adapt routes to user preferences, accessibility requirements, or historical navigation patterns.

C. Accessibility Enhancements

While the system currently prioritizes elevators over stairs, additional accessibility features are planned. These include turn-by-turn audio guidance, screen reader support, haptic feedback, and improved support for users with visual or mobility impairments. QR-code-based localization presents challenges for visually impaired users; therefore, future solutions may incorporate alternative location-estimation techniques and voice-driven interactions to improve inclusivity.

D. Database Design and Data Management Improvements

The existing database schema is designed to be flexible and scalable; however, future revisions will refine data normalization. In particular, parking areas should be represented as a dedicated entity rather than embedded within the floor structure, as this will improve routing accuracy for paths that involve both indoor and outdoor navigation.

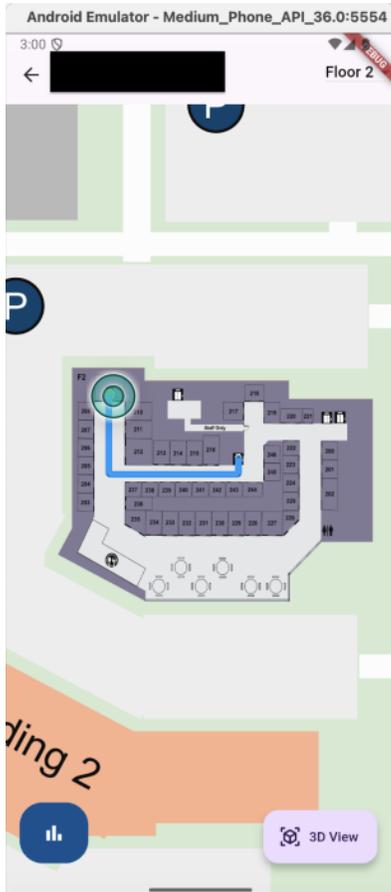


Fig. 10. Android UI with map view of Floor 2, with path and location

Additional validation mechanisms and synchronization safeguards will also be implemented to ensure reliable database re-initialization and consistency of updates.

E. Administrative and Analytics Features

Future development will include administrative tools and analytics dashboards to support large-scale deployment and system maintenance. An administrative interface would enable authorized personnel to manage buildings, floors, nodes, edges, and points of interest without directly manipulating the database. Analytics features could provide anonymized insights into navigation usage patterns, such as frequently visited destinations, common routes, and peak usage times. These insights could be used to optimize map layouts, improve routing graphs, enhance accessibility, and support facility planning decisions, while adhering to user privacy and data protection principles.

F. Security and Access Control

Security remains an important area for future improvement. Enhanced authentication, role-based access control for administrative features, and secure data synchronization mechanisms will be investigated to ensure that both user data and system infrastructure are protected as the platform scales.

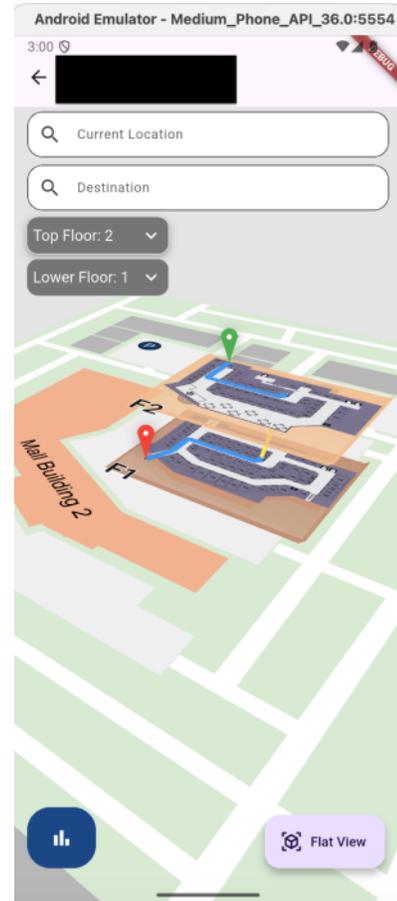


Fig. 11. Android UI with map view of 3D map with floor transition path

G. System Expansion and Feature Extensions

This project represents a small-scale prototype. Future work includes expanding the system to support additional buildings and campuses, improving the completeness of visitor information, and exploring complementary features, such as conversational assistants or chatbots, for information retrieval. Further development and evaluation in real-world environments will be necessary to assess scalability, usability, and long-term performance.

VII. CONCLUSION

Indoor wayfinding systems continue to evolve as demand for intuitive, mobile-friendly navigation solutions increases. While many existing platforms rely on subscriptions or proprietary infrastructure, this work demonstrates a scalable, offline-capable, and cross-platform alternative. By integrating graph-based routing, QR-assisted localization, and multi-floor visualization, the proposed system offers a practical solution for complex indoor and outdoor navigation environments.

ACKNOWLEDGMENTS

Thank you to the Computer Science Department and Grants in Aid fund [21] at Okanagan College (OC), as

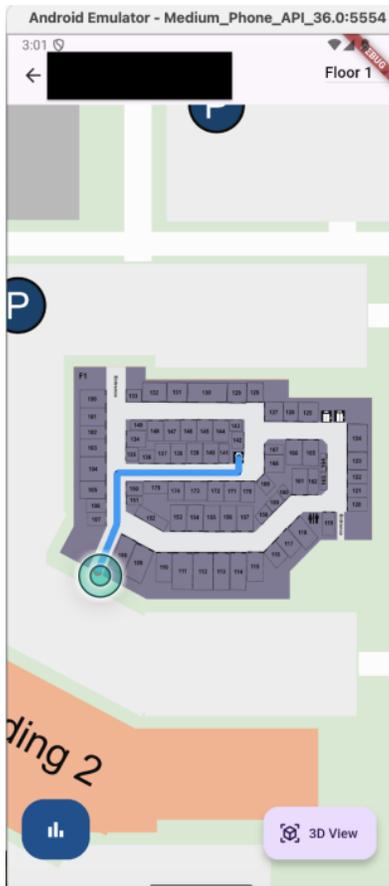


Fig. 13. Android UI with map view of Floor 1, with a path

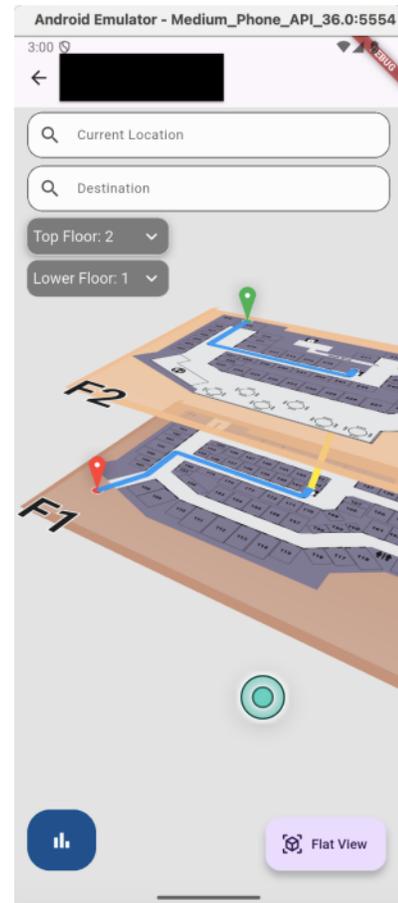


Fig. 12. Android UI with map view of 3D map with floor transition path

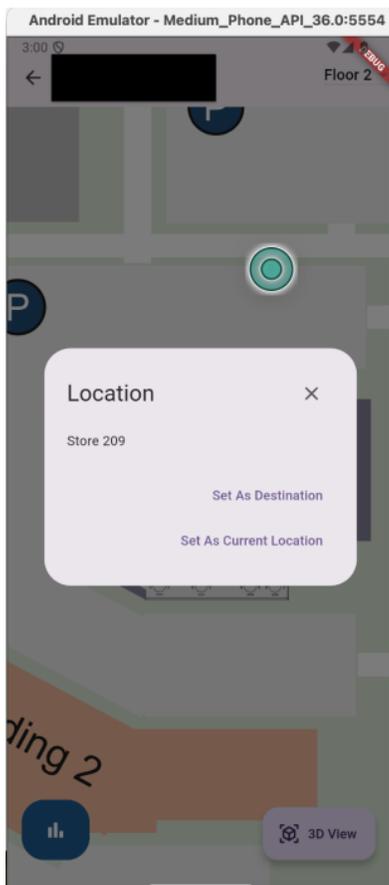


Fig. 14. Android UI with option to set a location as a Destination or Current Location on Floor 2

well as UBC Okanagan (UBCO) and the University of the Fraser Valley (UFV) for supporting this research.

Several students have contributed to a previous part of this project, which led to this study. These students include Dolcy Sareen (OC), Dylan Huitema (OC), Clara Elliott (OC), Ahab Masud Siddiqui (UBCO), Samyak Jain (UBCO), and Aliasgar Sakarwala (UBCO).

Portions of the documentation were reviewed and refined using built-in AI-assisted writing tools to improve clarity and readability. The authors conducted all system design, implementation, experimentation, and analysis.

REFERENCES

- [1] "YYC - At the airport - Terminal maps." [Online]. Available: <https://www.yyc.com/en-us/at-the-airport/terminal-maps>
- [2] "West edmonton mall interactive map," Online, 2024. [Online]. Available: <https://www.wem.ca/>
- [3] "Campusgo - indoor navigation app, university of windsor," Online, 2024. [Online]. Available: <https://www.uwindsor.ca/campusgo>

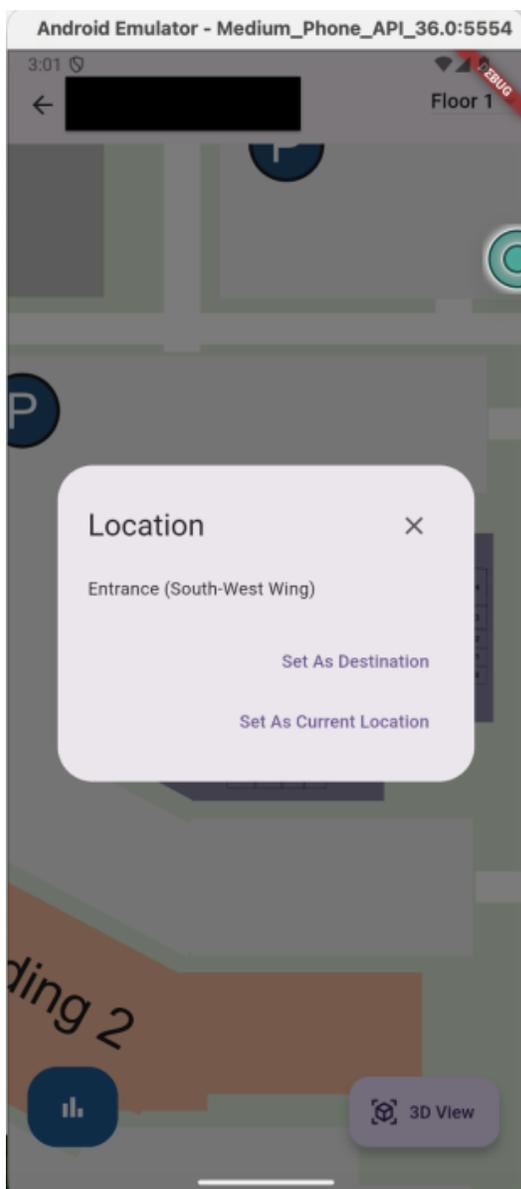


Fig. 15. Android UI with option to set a location as a Destination or Current Location on Floor 1

[4] N. Fallah, I. Apostolopoulos, K. Bekris, and E. Folmer, "Indoor human navigation systems: A survey," *Interacting with Computers*, vol. 25, no. 1, pp. 21–33, 2013.

[5] J. Kunthoth, A. Karkar, S. Al-Maadeed, and A. Al-Ali, "Indoor positioning and wayfinding systems: a survey," *Human-centric Computing and Information Sciences*, vol. 10, no. 18, 2020.

[6] N. El-Sheimy and L. You, "Indoor navigation: state of the art and future trends," *Satellite Navigation*, vol. 2, no. 7, 2021.

[7] V. Bahl and V. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *Proceedings of IEEE INFOCOM*, 2000, pp. 775–784.

[8] S. A. Cheraghi, V. Namboodiri, and K. Sinha, "Ibeaconmap: Automated indoor space representation for beacon-based wayfinding," *arXiv preprint arXiv:1802.05735*, 2018.

[9] J. Santos *et al.*, "A generic approach toward indoor navigation and pathfinding with markers," *Remote Sensing*, vol. 11, no. 24, p. 3052, 2019.

[10] R. Vishwakarma, R. B. Joshi, and S. Mishra, "Indoorgnn: A graph neural network based approach for indoor localization using wifi rssi," *arXiv preprint arXiv:2312.07609*, 2023.

[11] A. S. Martínez-Sala, F. Losilla, J. C. Sánchez-Aarnoutse, and J. García-Haro, "Design, implementation and evaluation of an indoor navigation system for visually impaired people," *Sensors*, vol. 15, no. 12, pp. 32 168–32 187, 2015.

[12] T. M. Fernández-Caramés *et al.*, "An indoor navigation methodology for mobile devices by integrating augmented reality," *IEEE Access*, vol. 9, pp. 120 455–120 470, 2021.

[13] Y. Cheng and R. Manduchi, "Palms: Plane-based accessible indoor localization using mobile smartphones," *arXiv preprint arXiv:2410.15694*, 2024.

[14] Z. Qiu, M. Ashour, X. Zhou, and S. Kalantari, "Navmarkar: A landmark-based augmented reality wayfinding system for older adults," *arXiv preprint arXiv:2311.12220*, 2023.

[15] S. Subedi and J.-Y. Pyun, "A survey of smartphone-based indoor positioning system using rf-based wireless technologies," *Sensors*, vol. 20, no. 24, p. 7230, 2020.

[16] Y. Etiabi, W. Njima, and E. M. Amhoud, "Federated learning based hierarchical 3d indoor localization," *arXiv preprint arXiv:2303.00450*, 2023.

[17] D. Gufran and S. Pasricha, "Fedhil: Heterogeneity resilient federated learning for robust indoor localization with mobile devices," *arXiv preprint arXiv:2307.01780*, 2023.

[18] Y. Etiabi, W. Njima, and E. M. Amhoud, "Femloc: Federated meta-learning for adaptive wireless indoor localization tasks in iot networks," *arXiv preprint arXiv:2405.11079*, 2024.

[19] A. Jaheen, S. Elsamanody, H. Rizk, and M. Youssef, "Simdeep: Federated 3d indoor localization via similarity-aware aggregation," *arXiv preprint arXiv:2508.01515*, 2025.

[20] Anonymous, "A review of indoor localization techniques and wireless technologies," *Wireless Personal Communications*, vol. 119, p. 289–327, 2021.

[21] "GRANTS-IN-AID FUND GUIDELINES." [Online]. Available: <https://acm-assets.okanagan.bc.ca/digitalassetdirect/asset/did/5008/GIA%20Guidelines.pdf>