National College of Information Technology (NACIT)
(Blantyre Campus)
Advanced Diploma in Computing (Level5DC)
Computing Project Proposal Paper

Mobile Application for Visually Impaired Indoor Navigation at LMJ Hospital

Name of Student: Chrispine Sibale
Student ID Number: 00186137
Email Address: raf33arn@gmail.com
Phone Number: 0981167823 / 0887833995

Name of Project Coordinator: Mrs Mwausegha
Name of Project Supervisor: Mrs Singini

ABSTRACT

This project proposes the development of a mobile application designed to assist visually impaired individuals in navigating LMJ Hospital independently. The nature of the project involves integrating smartphone-based technologies—such as computer vision, natural language processing, and machine learning—to support real-time obstacle detection, voice-activated interaction, landmark recognition, and audio-based indoor navigation. The system will use on-device processing, operate offline using SQLite-stored maps, and offer a user-friendly voice interface. The intended outcome is to empower visually impaired patients by enhancing mobility, privacy, and access to healthcare services without needing physical changes to hospital infrastructure. The project is expected to contribute a scalable, affordable, and practical solution to digital accessibility within local medical environments.

CONTENTS

1. LITERATURE REVIEW

a. Introduction

LMJ Hospital in Blantyre, Malawi, serves a wide variety of patients with different health needs. However, visually impaired individuals often struggle to access these services independently due to the hospital's complex structure. The proposed project tackles the problem of **indoor navigation for visually impaired individuals**, which is vital in enhancing autonomy, confidence, and access to health services.

Indoor navigation refers to a system that helps users orient themselves and move through indoor spaces using digital tools. Unlike outdoor GPS, indoor navigation requires local mapping and alternative detection systems such as computer vision. According to the National Library of Medicine (NLM, 2020), indoor navigation systems significantly improve the safety and autonomy of visually impaired users by providing orientation cues and avoiding reliance on memory or others.

This project aims to solve LMJ Hospital's accessibility gap by creating an indoor navigation app that offers real-time, context-aware assistance through visual recognition and voice interaction.

b. Similar Systems

1. **Navigine**

1.1. System Overview and Core Functionality

Navigine is recognized as a global provider of integrated positioning mobile technologies, specializing in advanced indoor navigation and proximity solutions that serve a diverse range of industries. The company offers a comprehensive platform that includes development tools for mobile developers and system integrators, enabling them to create a wide array of indoor navigation and tracking services tailored to specific client needs (Microsoft, 2022).

The core functionality supported by the Navigine platform encompasses the ability to download and work with custom indoor maps, integrate with iBeacons for precise localization, generate local push notifications based on proximity, accurately determine a user's real-time position on a map, construct optimal routes to desired destinations, and collect valuable foot traffic analytics for operational insights. For end-users, applications built upon the Navigine platform are designed to provide their current position, deliver useful contextual wayfinding information about the indoor environment, and offer detailed directions to a specified destination, thereby enhancing indoor mobility and orientation (Apple, 2022).

1.2. Technological Implementation and Positioning Approach

Navigine's primary technological approach for achieving high-accuracy indoor positioning relies heavily on the deployment of external physical infrastructure. This predominantly includes Bluetooth Low Energy (BLE) iBeacons and Wi-Fi signals strategically placed throughout the indoor environment. Academic research associated with similar systems also indicates the potential use of Ultra-Wideband (UWB) technology, which is highly regarded for its ability to support real-time location and navigation. UWB technology offers distinct advantages, including strong resistance to narrowband interference and robust performance in complex indoor multipath environments, without requiring a direct line-of-sight communication path (Riehle, 2008).

To further enhance positioning accuracy and reliability, Navigine integrates data from internal sensors on smartphones. These include accelerometers, gyroscopes, barometers, and compasses, which are combined with detailed map information and sophisticated motion models to provide a more precise and stable location estimate. The system employs advanced pathfinding algorithms, such as Dijkstra's algorithm, to efficiently determine the shortest and most optimal route between a user's current location and their desired destination. Additionally, some applications built on the Navigine platform may leverage characteristic magnetic field anomalies within buildings as supplementary landmarks to aid in navigation and orientation (Apple, 2022).

The reliance on infrastructure investment presents a trade-off with positional accuracy for broader scalability. Navigine's fundamental dependence on pre-installed physical infrastructure like UWB and BLE beacons for its "high accuracy" represents a common, yet often costly and complex, approach to indoor positioning. While UWB offers robustness, it necessitates specific hardware (tags and sensors), and BLE signals are known to be prone to "noise" and "attenuation" by building materials, potentially leading to "jumping locations". This implies that achieving and maintaining high accuracy with these technologies often demands substantial initial infrastructure deployment and ongoing calibration. This architectural philosophy fundamentally diverges from MVI-LMJ's explicit goal of "no physical changes to hospital infrastructure" and its emphasis on affordability. Navigine prioritizes infrastructure-dependent accuracy, while MVI-LMJ prioritizes minimal environmental modification and cost-effectiveness, accepting the challenges inherent in on-device sensor fusion (Guild, 2018).

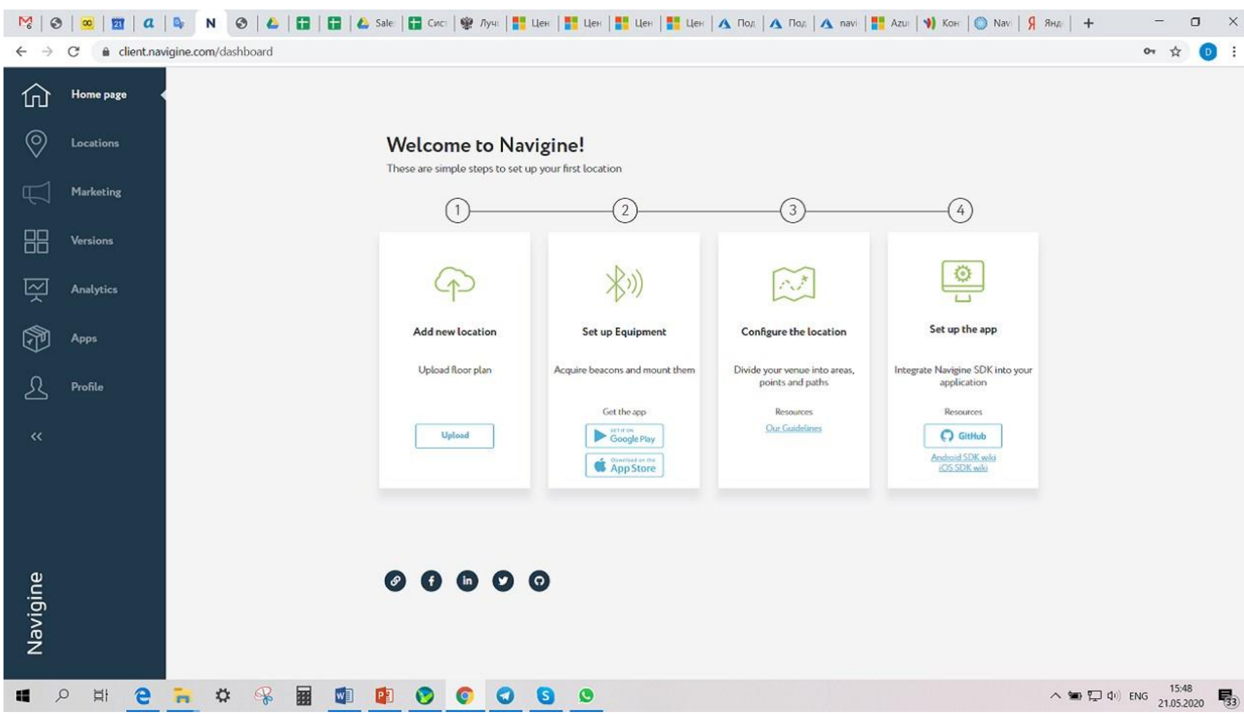


Figure 1: Navigine Admin Dashboard. Image from the App Store (Apple, 2022).

1.3. Assessment: Accessibility, Navigation, and Appearance

As a platform provider, Navigine's direct accessibility features for visually impaired end-users are not extensively detailed in the provided information. However, applications developed using the Navigine platform can be designed with accessibility in mind. Such applications are capable of providing directions primarily through spoken communication, aiming to create an audible experience for navigation. The design philosophy often includes minimizing the amount of user input required, allowing the user to concentrate more on safety during navigation. Instructions can also be supplemented with haptic feedback delivered via smartwatches, further enhancing the accessibility for visually impaired users by providing an additional sensory cue.

In terms of navigation, the system provides turn-by-turn guidance. Directions are typically announced a few meters before an anticipated turn, for example, "at the upcoming junction, turn left," given five meters in advance, to allow users sufficient time to prepare. To further reduce cognitive load, distance information can be provided in approximate, rounded numerical measures, such as "ten paces to your next turn". A backtracking feature is also available, allowing users to invert a previously traversed route for independent return. However, a significant aspect of its navigation is its fundamental reliance on pre-installed physical infrastructure (beacons, Wi-Fi). This means that deployment is highly site-specific and its ubiquitous availability is contingent upon prior investment and setup in each location. Furthermore, while widely used, BLE signals can be noisy and susceptible to attenuation by various building materials, potentially leading to location "jumping" and reduced accuracy for precise, fine-grained localization.

Regarding appearance, as Navigine functions as a platform offering development tools, the visual "appearance" of end-user applications built upon it can vary significantly depending on the developer's implementation. Screenshots typically depict map-based interfaces with visual representations of user location and routes. For visually impaired users, the "appearance" of the system would be primarily perceived through the clarity, naturalness, and intuitiveness of its audio instructions and the effectiveness of any supplementary haptic feedback.

The platform-centric approach of Navigine has an indirect impact on direct user accessibility. Navigine is consistently described as providing "development tools" and being a "platform" for integrators. This indicates a business-to-business (B2B) model, where Navigine supplies the underlying technology, but the ultimate end-user accessibility, navigation, and appearance for visually impaired individuals are contingent upon the specific application built *on top* of their platform. This is a crucial distinction from MVI-LMJ, which is being developed as a complete, accessible end-user solution from its inception. While Navigine's technology *can* support accessible features, it does not inherently guarantee them. This implies that MVI-LMJ, by controlling the entire application stack, possesses more direct and comprehensive control over ensuring its accessibility features meet the precise and nuanced needs of its visually impaired users.

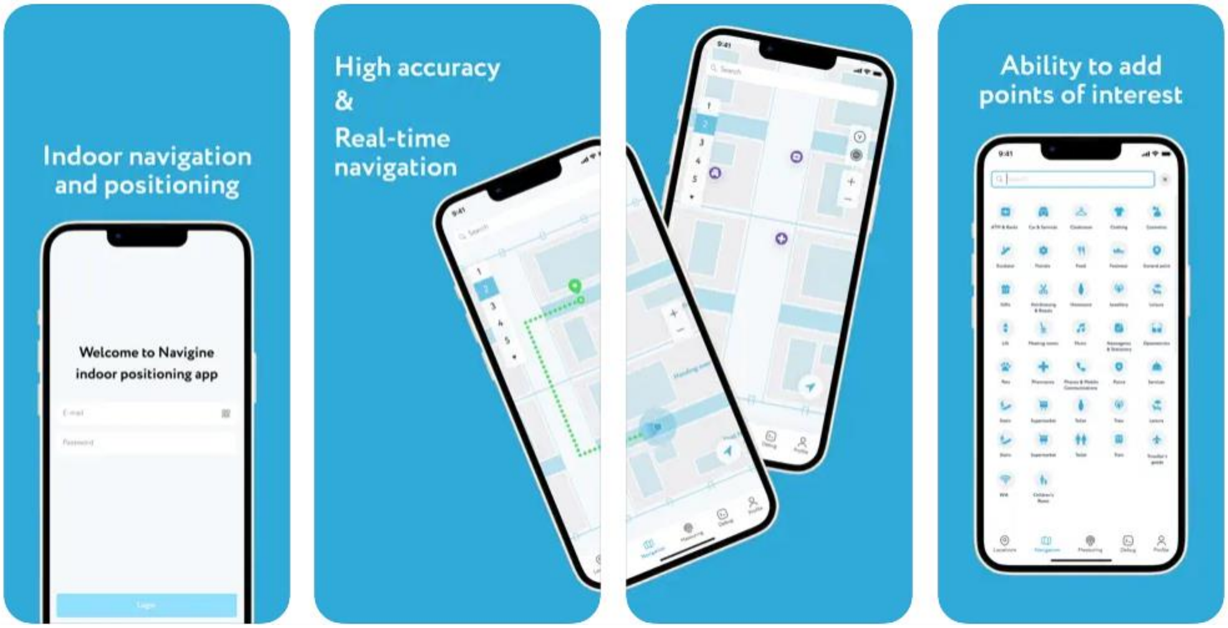


Figure 2: iPhone Screenshots of Navigine Mobile App on the App Store (Apple, 2022)

2. **Seeing AI**

2.1. **System Overview and Core Functionality**

Seeing AI is a free mobile application developed by Microsoft Research, initially launched for iOS and subsequently expanded to Android devices. Its primary purpose is to narrate the visual world for individuals who are blind or have low vision, leveraging the power of Artificial Intelligence (AI) to interpret and describe their surroundings (SeeingAI, 2025).

The application offers a diverse and comprehensive set of functionalities designed to assist with various daily tasks. These include instant reading of short texts, providing audio guidance to capture and recognize text from printed documents, scanning barcodes for product identification with audio cues, recognizing and describing people (including their facial expressions), identifying currency notes, and a "Find My Things" feature to help users locate personal objects. A particularly notable feature is the "World" channel, which provides an Audio Augmented Reality (AR) experience. This channel announces objects in the user's surroundings with spatial audio, enabling a more immersive exploration of unfamiliar environments, especially when utilizing devices equipped with LiDAR technology (Microsoft, 2016).

2.2. **Technological Implementation and AI-driven Capabilities**

Seeing AI harnesses advanced AI, combining both cloud-based and on-device processing to deliver its intelligent features. It specifically utilizes AI running locally on the device to describe the immediate environment, ensuring responsiveness and a degree of privacy (Lukman, et al., 2025). The underlying technology incorporates key AI disciplines, including sophisticated computer vision for image understanding, machine learning for pattern recognition and object identification, and natural language processing for generating descriptive and intuitive audio narratives (Taylor&Francis, 2020).

The development of Seeing AI was built upon significant breakthroughs in vision-to-language technology and the achievement of highly accurate image classification systems, which demonstrated error rates comparable to or even surpassing human accuracy. For indoor navigation, Seeing AI has recently ventured into an innovative approach, using AI to map indoor locations and create routes with "virtual beacons (Perkins, 2022)." A key design goal for this feature was to enable navigation "with only a camera without GPS, Wi-Fi, or beacons," signifying a strong emphasis on infrastructure-independent, camera-based solutions for indoor environments. Academic research further supports the system's capabilities, highlighting its use of neural networks for advanced object detection and tracking, along with depth information for precise object localization (NLM, 2020).

Seeing AI's pioneering role in camera-first, AI-driven environmental understanding serves as a significant blueprint for MVI-JMJ. Seeing AI's core strength lies in its "intelligent camera app" that uses AI to "narrate the world". This represents a conceptual leap from traditional navigation systems, which often focus primarily on positional data. Seeing AI's ability to "describe scenes," "recognize objects," and even "map indoor worlds in 3-D and navigating it with only a camera without GPS, Wi-Fi, or beacons" powerfully demonstrates the potential of computer vision and AI for comprehensive environmental understanding. This directly validates and provides a strong precedent for MVI-LMJ's camera-based obstacle detection and landmark recognition approach. It suggests that MVI-LMJ can aim for a richer, more context-aware form of navigation, moving beyond simple turn-by-turn directions to providing a dynamic "narrative" of the hospital environment, including real-time obstacles and points of interest (Microsoft, 2016).

2.3. **Assessment: Accessibility, Navigation and Appearance**

Seeing AI is explicitly designed with the blind and low vision community in mind, aiming to transform the visual world into an accessible audible experience. The user interface incorporates high contrast and enlarged icons, which benefit users with residual vision. Crucially, a voice narrates feature names, facilitating effortless navigation within the app for all visually impaired users. Users are provided with options to adjust the voice and speaking speed, though some user feedback indicates a desire for more granular control over speech rate (Liao, 2024).

In terms of navigation, while not its primary initial focus, Seeing AI's indoor navigation feature is actively evolving, utilizing AI to map locations and create routes with "virtual beacons". The "World" channel provides an Audio Augmented Reality experience, leveraging spatial audio to announce objects in the user's surroundings. This capability significantly aids in exploring unfamiliar environments and understanding spatial relationships. This navigation feature is continuously being refined based on user feedback, indicating an iterative development approach (Perkins, 2022).

Regarding appearance, the application maintains a clean and functional interface, with a strong emphasis on audio output as the primary mode of information delivery. Its visual design aims to align with mainstream Augmented Reality (AR) and AI applications, which contributes to its broad appeal and helps to reduce the perception of it being solely an "assistive" tool, thereby enhancing its desirability and integration into everyday technology use (Liao, 2024).

The evolving definition of "navigation" within Seeing AI includes comprehensive environmental awareness. Seeing AI's "World" channel and its ability to "describe scenes" and "recognize objects" fundamentally broadens the concept of "navigation" beyond simple wayfinding . For visually impaired users, true independence in an indoor environment involves not just knowing *where* to go, but also understanding *what is around them* – identifying dynamic obstacles, recognizing specific landmarks, and comprehending the general layout. This implies that MVI-JMJ should aspire to provide a rich, real-time contextual understanding of the hospital environment, integrating obstacle alerts and landmark announcements seamlessly into the navigation experience, rather than treating them as separate functionalities (Microsoft, 2016).

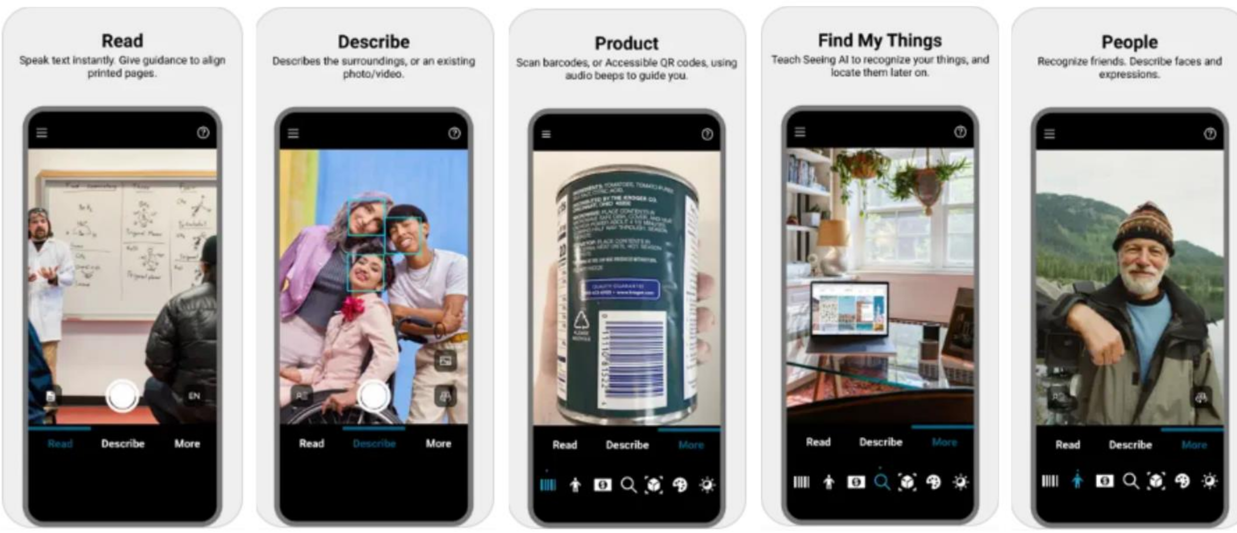


Figure 3: Image showing Seeing AI's capabilities to Read Text, Describe Surroundings, Scan Products, Locate Items, and Recognize People (Apple, 2025).

3. **LowViz Guide**

3.1. **System Overview and Core Functionality**

LowViz Guide is a specialized indoor navigation app created for blind and visually impaired users through a partnership between Indoo.rs (indoor positioning specialists) and MD Support (macular degeneration advocacy). The app excels at providing audio-guided navigation, particularly for temporary setups like conferences and seminars that serve many visually impaired attendees, though the underlying technology also works in permanent locations such as transit hubs, medical facilities, and retail centers. The app allows users to search for specific locations and receive audio guidance to their desired destination (Kendrick, 2015).

3.2. **Technological Implementation and Beacon-Based Guidance**

The system relies on small Bluetooth beacons (iBeacons) roughly the size of D batteries, placed strategically throughout indoor spaces. Users' iPhones detect these beacons and use signal strength to determine location and proximity. Before deployment, venues are mapped and important locations are marked, with custom audio messages recorded for each beacon point. As users move through the space, their device triggers relevant audio descriptions and directions. A clever feature uses changing audio tones that drop in pitch as users get closer to their target, creating an intuitive distance indicator (Kendrick, 2015).

This beacon-based approach works exceptionally well in controlled, pre-mapped settings where temporary installations are needed quickly. However, it has inherent limitations: it cannot detect dynamic obstacles like moving people or equipment, and requires extensive manual setup including physical beacon placement and custom audio recording for every location. This creates a fundamental contrast with camera-based systems that can understand environments in real-time without extensive pre-configuration (Corada, 2025).

3.3. Assessment: Accessibility, Navigation, and Appearance

LowViz Guide is specifically designed to be accessible for blind and visually impaired users, leveraging the native accessibility features of iOS devices. It supports voice activation for searching destinations and routing-by-voice, allowing for hands-free operation. The app's use of a changing pitch tone as the user approaches a destination provides an intuitive, non-visual proximity cue, enhancing spatial awareness (Kendrick, 2015).

In terms of navigation, the application provides clear, step-by-step spoken directions to guide users along their chosen path. Users can search for specific locations within the venue using either typing or dictation, and the system will provide the necessary audio guidance. The effectiveness of this navigation is highly dependent on the accuracy of the iBeacon placement and the quality of the pre-recorded messages. User feedback on similar AI assistants suggests that clear, structured instructions are paramount for usability, indicating that the simplicity of the interface and clarity of audio cues are key for effective navigation (Tara, 2025).

Regarding appearance, the LowViz Guide app features a functional layout with "Map View, Categories, and Search" tabs. While explicit visual user interface screenshots for LowViz Guide itself are not provided in the snippets, the design philosophy clearly prioritizes audio-based interaction and a straightforward, easy-to-navigate interface, minimizing visual clutter. The focus is on the auditory experience and the intuitiveness of the voice-activated controls (Kendrick, 2015).

The importance of clear, structured audio cues and the challenge of dynamic information are key considerations. LowViz Guide's reliance on pre-recorded messages and the innovative use of a changing pitch tone for proximity underscore the critical importance of clear, consistent, and intuitive audio feedback for visually impaired users. General user feedback on similar AI assistants further reinforces that "clear, numbered instructions" are highly effective, while complex, multi-task commands can lead to confusion. This implies that while MVI-JMJ will generate dynamic voice instructions based on real-time computer vision data, the clarity, conciseness, and structured nature of these instructions will be paramount for user comprehension, trust, and effective navigation, especially when integrating real-time obstacle detection which introduces dynamic and potentially unpredictable environmental information (Tara, 2025).

c. Technological Review

The project employs the following technologies:

- **Flutter & Dart:** Chosen for cross-platform compatibility on Android and iOS.
- **OpenCV & MediaPipe:** For image analysis, object detection, and obstacle recognition.
- **TensorFlow Lite:** Enables efficient on-device machine learning model inference.
- **Google TTS/STT APIs:** For converting text to voice instructions and recognizing spoken commands.
- **SQLite:** Used for local storage of maps, routes, and landmark data.
- **Rasa or similar NLP engine:** For offline voice command interpretation.

These tools are ideal due to their mobile efficiency, open-source availability, and flexibility for integration.

2. REQUIREMENTS

a. Introduction and Information Gathering

Requirement analysis is essential to ensure that the system accurately meets user needs. This project used the following methods:

1. **Interviews:** Conducted with visually impaired individuals and LMJ staff to capture real-life challenges.

2. **Observation:** On-site analysis of common navigation paths and departmental layouts.

3. **Document Review:** Existing hospital maps and related materials were examined to model the environment.

b. Functional Requirements

Functional requirements define the core actions the system must perform. The MoSCoW prioritization method was used:

Obstacle Detection

The system **must** detect physical obstacles such as walls and furniture in real time using the phone's camera.

Audio Navigation

The system **must** guide users using voice prompts based on predefined map paths.

Voice Commands

The system **must** allow users to request destinations or actions using natural voice commands.

Landmark Recognition

The system **must** identify and announce key landmarks like departments, restrooms, and exits.

Offline Map Access

The system **must** operate without the internet using preloaded hospital maps in SQLite.

Audio Settings

The system **should** allow users to adjust language, voice speed, and volume settings.

Route Recalculation

The system **could** adapt and suggest new routes when a user deviates from the original path.

Usage Feedback

The system **could** track anonymized usage patterns for future improvement but **must not** store any health-related data.

c. Non-Functional Requirements

Performance

The system **must** provide obstacle alerts and audio navigation within two seconds of detection.

Usability

The system **must** support complete voice-based interaction and accessible audio output.

Portability

The app **must** work on Android and iOS smartphones.

Security & Privacy

The system **must not** transmit or save personal health information; all processing remains on-device.

Maintainability

The system **should** have modular components to allow updates without reworking the entire system.

Reliability

The system **must** function in varying light and noise environments typical of hospital settings.

3. USE CASE

The MVI-LMJ Indoor Navigation System provides voice-controlled navigation assistance for visually impaired patients at LMJ Hospital. The system operates offline using pre-loaded hospital maps and real-time computer vision for obstacle detection.

3.1. Actors

Patient (Primary Actor)

The visually impaired individual who uses the system to navigate the hospital independently through voice commands and audio feedback.

Administrator (Secondary Actor)

IT personnel responsible for system maintenance, data updates, and performance monitoring to ensure reliable operation.

3.2. Use Cases

System-Level Use Cases

1. **Recognize Landmarks**: Automatically identifies and announces hospital landmarks such as departments, facilities, and key locations to provide spatial awareness.

2. **Detect Obstacles & Objects**: Uses camera and computer vision to identify obstacles, furniture, and people in real-time, providing audio warnings to prevent collisions.

Patient Use Cases

1. **Access Emergency Help**: Allows patients to request immediate assistance during medical emergencies or when lost, alerting hospital staff to their location.

2. **Navigate to Department**: Provides step-by-step voice guidance to specific hospital departments based on offline map data.

3. **Process Voice Commands**: Interprets natural language voice inputs and executes corresponding system functions.

4. **Find Nearby Facilities**: Identifies and announces nearby amenities such as restrooms, ATMs, seating areas, and services within detection range.

5. **Adjust Settings**: Enables patients to customize voice speed, language preferences, volume levels, and accessibility options.

Administrator Use Cases

1. **System Backup and Recovery**: Maintains data integrity and system availability through regular backups and recovery procedures.

2. **Load Offline Maps**: Updates hospital layout data in SQLite database when facility changes occur.

3. **Update Landmark Database**: Maintains current information about hospital departments, facilities, and points of interest.

4. **Monitor System Performance**: Tracks system metrics including response times, navigation accuracy, and user satisfaction.

5. **Manage User Accounts**: Handles patient authentication, account settings, and access permissions.

6. **Generate Usage Reports**: Produces anonymized analytics on system usage patterns for continuous improvement.

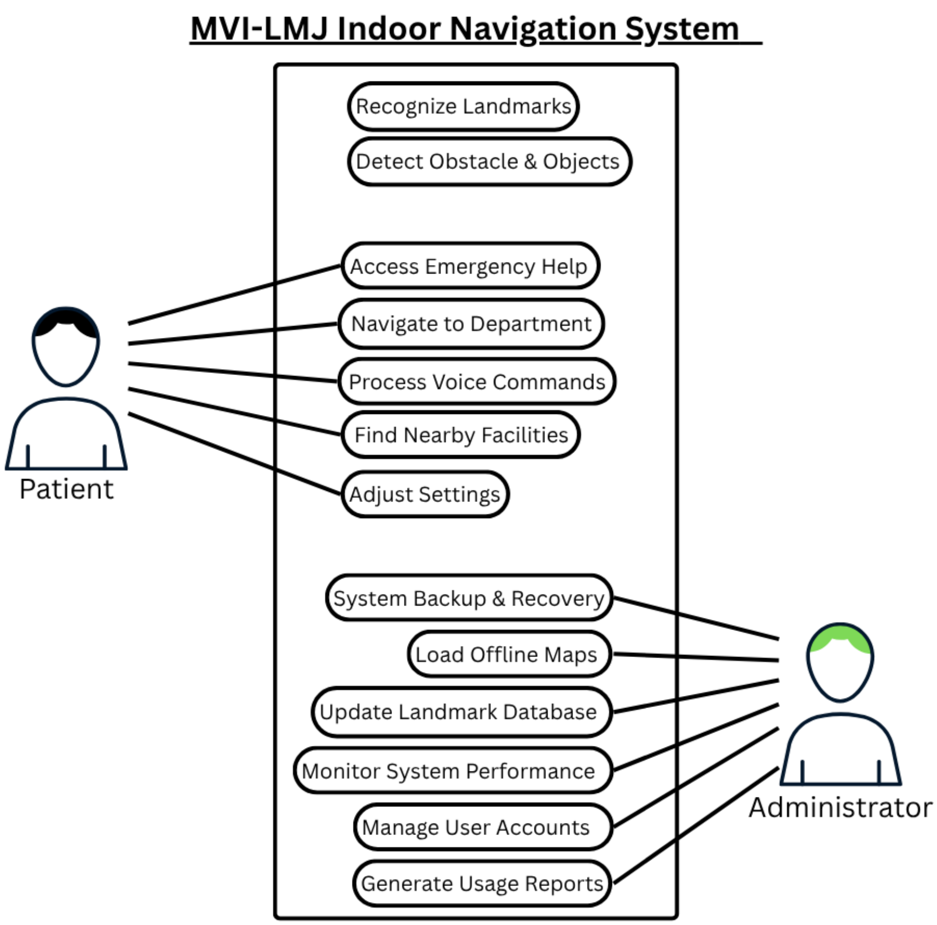


Figure 4: MVI-LMJ System Use Case Diagram.

4. ARCHITECTURE

a. System Architecture

Interfaces with Other Systems

- **Human:** Visually impaired users and staff during setup
- **Automated:** Smartphone camera, mic, sensor APIs, and on-device TTS/STT

Technical Architecture Overview

- **Presentation Layer:** Flutter-based mobile front end
- **Processing Layer:** Computer vision, NLP, and navigation logic
- **Data Layer:** SQLite maps and landmark database
- **Audio Layer:** Text-to-speech and speech-to-text engine

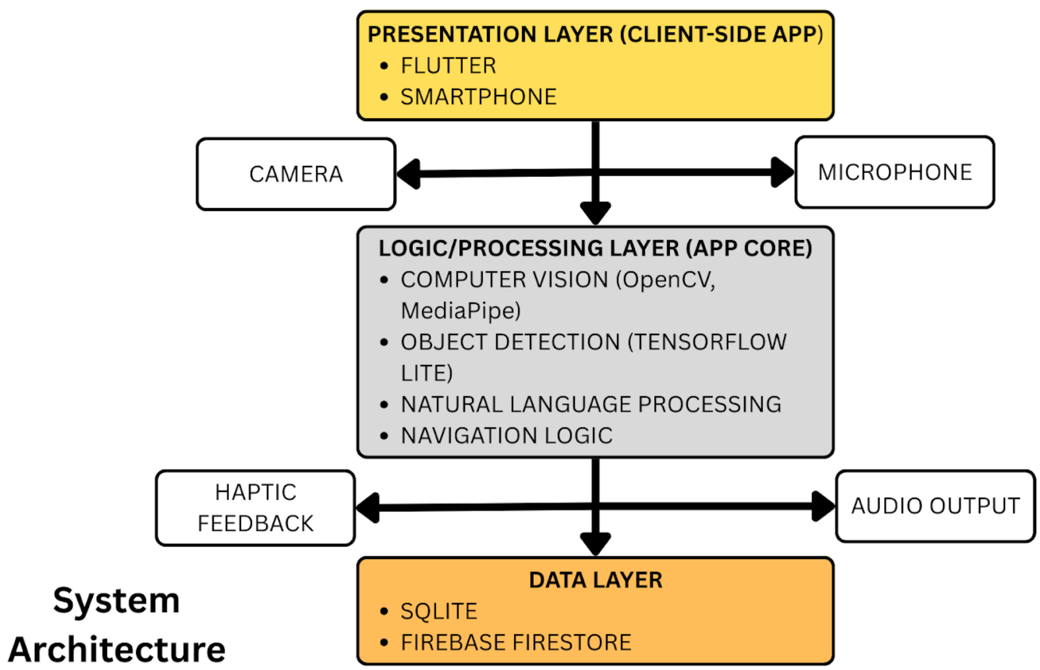


Figure 6: System Architectural Diagram

b. Class Diagram

Figure 5 presents the UML class diagram for the proposed navigation assistance system, illustrating the structural relationships between five core components. The system employs a coordinator pattern with the **Navigator** class serving as the central orchestrator that manages navigation logic and coordinates interactions between specialized subsystems.

The **Navigator** class maintains navigation state through private attributes (currentLocation, destination, activeRoute) and exposes public methods for route calculation and position updates. It establishes usage relationships with three key components: **ObstacleDetector** for real-time hazard detection through camera processing, **LandmarkRecognizer** for identifying and announcing predefined locations, and **MapManager** for accessing route data from the SQLite database.

The **VoiceInterface** class is integrated through an aggregation relationship, indicating that Navigator contains a voice interface component for processing speech commands and providing audio feedback. The **MapManager** serves as a shared data repository, utilized by both Navigator and LandmarkRecognizer to access map tiles, route information, and landmark databases.

This architecture demonstrates clear separation of concerns, with each class maintaining single responsibility while supporting the overall navigation functionality. The relationships show minimal coupling between components, facilitating maintainability and extensibility of the system.

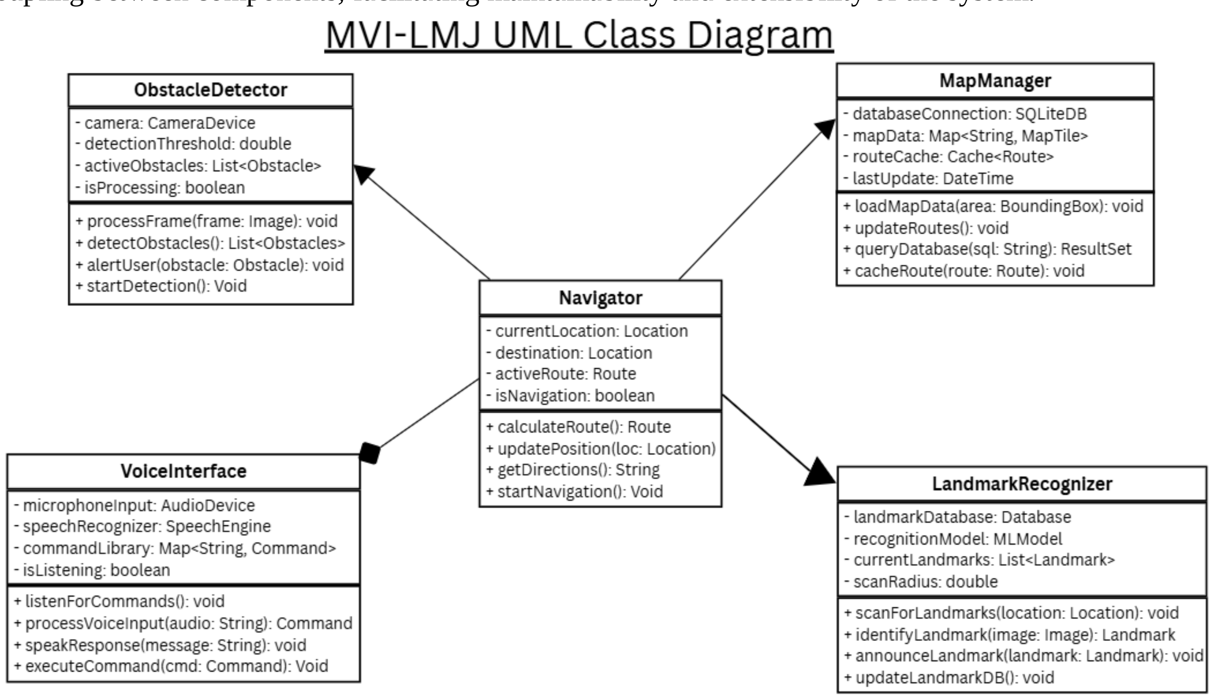


Figure 6: Shows the UML Class Diagram for MVI-LMJ

5. REFERENCING

References

Apple, 2022. *App Store Preview: Navigine.* [Online] Available at: https://apps.apple.com/jo/app/navigine/id972099798 [Accessed 1 July 2025].

Guild, L., 2018. *A Hybrid Indoor Positioning System for the Blind and Visually Impaired Using Bluetooth and Google Tango.* [Online] Available at: https://lighthouseguild.org/a-hybrid-indoor-positioning-system-for-the-blind-and-visually-impaired-using-bluetooth-and-google-tango/ [Accessed 1 July 2025].

Lukman, Angreani, D., Muh & Bakti, R. Y., 2025. *The Use of Artificial Intelligence in The Development of Vision Aids for The Visually Impaired.* [Online] Available at: https://www.researchgate.net/publication/389070867_The_Use_of_Artificial_Intelligence_in_The_Development_of_Vision_Aids_for_The_Visually_Impaired [Accessed 1 July 2025].

Medicine, N. L. o., 2020. *Indoor Navigation Systems for Visually Impaired Persons: Mapping the Features of Existing Technologies to User Needs.* [Online] Available at: https://pubmed.ncbi.nlm.nih.gov/31979246/ [Accessed 1 July 2025].

Microsoft, 2016. *An app for visually impaired people that narrates the world around you.* [Online] Available at: https://www.microsoft.com/en-us/garage/wall-of-fame/seeing-ai/ [Accessed 1 July 2025].

Microsoft, 2022. *Navigine Platform.* [Online] Available at: https://appsource.microsoft.com/en-us/product/web-apps/navigine1585640136850.navigine_platform?tab=overview [Accessed 1 July 2025].

Riehle, T., 2008. *An Indoor Navigation System to Support the Visually Impaired.* [Online] Available at: https://umaine.edu/vemi/wp-content/uploads/sites/220/2016/08/Riehle-etalEMBC-2008-An-Indoor-Navigation-System-to-Support-the-Visually-Impaired.pdf [Accessed 1 July 2025].

SeeingAI, 2025. *Google Play.* [Online] Available at: https://play.google.com/store/apps/details?id=com.microsoft.seeingai [Accessed 1 July 2025].

Taylor&Francis, 2020. *Seeing AI.* [Online] Available at: https://taylorandfrancis.com/knowledge/Engineering_and_technology/Computer_science/Seeing_AI/ [Accessed 1 July 2025].

6. APPENDIX

Use Case Descriptions

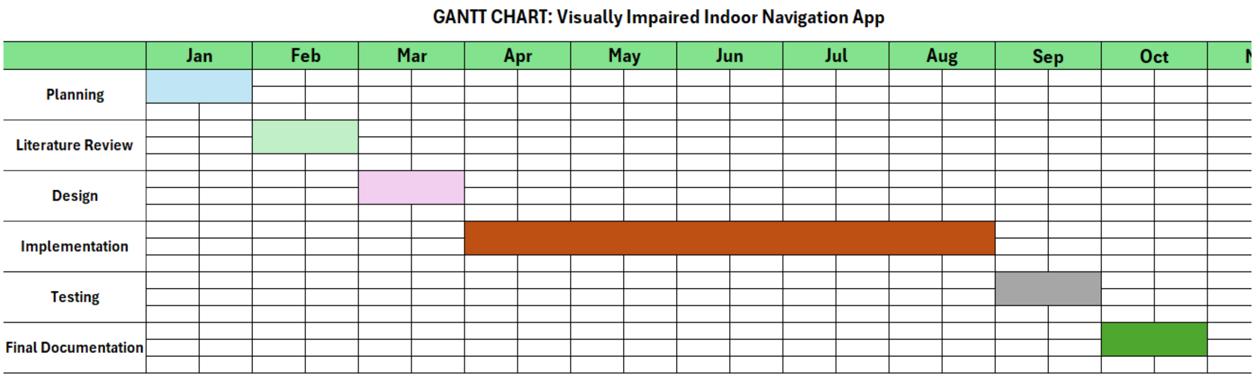
Table A.1: Patient Use Cases

Use Case Name	Primary Actor	Description
Recognize Landmarks	Patient	The system identifies and announces predefined landmarks such as hospital departments, waiting areas, elevators, and other significant locations to help patients orient themselves within the facility.
Detect Obstacle & Objects	Patient	The system uses camera-based detection to identify obstacles (wheelchairs, medical equipment, furniture) and objects in the patient's path, providing audio alerts to ensure safe navigation.
Access Emergency Help	Patient	Patients can quickly access emergency assistance through voice commands or gesture controls, immediately connecting them to hospital staff or security personnel when urgent help is needed.
Navigate to Department	Patient	The system provides step-by-step audio navigation instructions to guide patients from their current location to specific hospital departments, clinics, or service areas.
Process Voice Commands	Patient	The system interprets and responds to spoken instructions from patients, including navigation requests, information queries, and system control commands through natural language processing.
Find Nearby Facilities	Patient	Patients can locate nearby amenities such as restrooms, cafeterias, pharmacies, gift shops, ATMs, and other hospital facilities within their vicinity.
Adjust Settings	Patient	Users can customize system preferences including voice speed, volume levels, language selection, navigation preferences, and accessibility options to suit their individual needs.

Table A.2: Administrator Use Cases

Use Case Name	Primary Actor	Description
System Backup & Recovery	Administrator	Administrators perform regular system backups, data recovery operations, and ensure system continuity through comprehensive backup and restoration procedures.
Load Offline Maps	Administrator	The system administrator uploads and maintains offline map data to ensure navigation functionality continues even during network connectivity issues or system maintenance periods.
Update Landmark Database	Administrator	Regular maintenance of the landmark database including adding new locations, updating existing landmark information, removing obsolete entries, and ensuring accuracy of location data.
Monitor System Performance	Administrator	Continuous monitoring of system performance metrics, user activity logs, error rates, response times, and overall system health to ensure optimal operation.
Manage User Accounts	Administrator	Creation, modification, and deletion of user accounts, assignment of access privileges, password resets, and management of user permissions within the system.
Generate Usage Reports	Administrator	Production of comprehensive reports detailing system usage statistics, user behaviour patterns, popular destinations, system performance metrics, and other analytical data for decision-making.

Gantt Chart


GANTT CHART: Visually Impaired Indoor Navigation App
Jan
Feb
Mar
Apr
May
Jun
Jul
Aug
Sep
Oct
Planning
Literature Review
Design
Implementation
Testing
Final Documentation

Work Breakdown Structure

Phase	**Description**	**Duration**	**Milestones**
Planning	• Creating and documenting the problem statement, system scope and objectives including resources required • Creating a timeline and milestones	January (4 week)	Finalized documentation of problem statement, project scope and objectives
Literature review	• Research other navigation aids and mobile apps for the visually impaired	February (4 weeks)	Literature review document finalization
Design	• Designing the User interface with no dependence on visual elements like buttons or icons • Design the voice interaction and computer vision flow diagram • Designing the database schema to store maps and landmark data structure	March (4 weeks)	System, UI, and database schema completion and documentation
Implementation	• Installing development tools • Coding the front-end and user interface using Flutter • Developing the real-time obstacle development module • Developing the voice control and interaction module • Developing the indoor navigation logic (pathfinding) and dynamic generation of audio directions	April - August (16 weeks)	Developing a functional prototype and implementation of all core modules
Testing	• Conduct Unit testing by testing each individual module e.g obstacle detection, voice command recognition works correctly on its own • Gather feedback from test users	September (4 weeks)	Finalization of testing documentation and incorporation of feedback
Final Documentation	• Prepare full project documentation • Include screenshots, code explanations, and user manual	October (4 weeks)	Submission of final documentation and presentation of the project