Software Project Management Plan: Everywhere Locator

Version: 1.1

Date: September 2, 2025

1. Introduction

1.1 Project Overview

Everywhere Locator is a mobile application designed to assist visually impaired individuals with indoor navigation. The project's primary goal is to create a functional application that empowers users to identify and locate common indoor objects using their smartphone's camera. The user will state their desired object via voice command, the application will use the camera and a Large Language Model (LLM) for object recognition, and it will provide audio feedback to guide the user.

1.2 Project Deliverables

Upon completion, the project team will deliver the following:

- A cross-platform mobile application for both iOS and Android, built with React Native.
- The complete source code, hosted in a GitHub repository.
- User documentation detailing how to build, configure, and deploy the application.
- A final presentation summarizing the project's goals, process, and outcomes.

1.3 Evolution of this Document

This SPMP is a living document. It will be updated throughout the project lifecycle to reflect changes in scope, schedule, or processes. The Project Manager will be responsible for maintaining the document.

1.4 References

- React Native Documentation: https://reactnative.dev/docs/getting-started
- Shadcn UI (or similar design system) for UI components and accessibility standards.

1.5 Definitions, Acronyms, and Abbreviations

- CV: Computer Vision
- LLM: Large Language Model (e.g., Google Gemini)
- MVP: Minimum Viable Product
- PM: Project Manager
- PO: Product Owner
- QA: Quality Assurance
- SPMP: Software Project Management Plan
- UI/UX: User Interface / User Experience

2. Project Organization

2.1 Process Model

This project will follow a two-phase model aligned with the academic deadlines. Each phase will be treated as a one-week sprint within an Agile framework. This allows for a focused effort on requirements and design in the first week, followed by a focused development sprint in the second week. The team will hold a weekly progress meeting to review progress and plan for the next phase.

2.2 Organizational Structure

The 10-person team will be divided into two functional sub-teams to streamline development:

- Core Experience Team (5 members): Focuses on the user interface, voice command processing (voice-to-text), and audio feedback (text-to-voice).
- Vision & Navigation Team (5 members): Focuses on camera integration, LLM API calls, and the core object recognition and guidance logic.

2.3 Organizational Boundaries and Interfaces

The Project Manager / Product Owner (PM/PO) will serve as the primary interface between the two teams, ensuring alignment on goals and priorities. However, direct communication between developers on both teams is encouraged to resolve technical dependencies quickly.

2.4 Project Responsibilities

Role	Name (NetID)	Responsibilities
PM/PO	Duy Pham (ldp210003)	Defines project vision, manages backlog, leads meetings, removes impediments.
Vision & Nav Lead	Victor Sim (vas230001)	Leads camera and LLM integration, oversees navigation algorithm.
Core Experience Lead	Chase Uherek (cxu230001)	Leads UI/UX development, ensures accessibility standards are met.
Developer	Nathan Boyle (nmb210005)	Core Experience Team - UI/UX implementation.
Developer	Viet-Long Nguyen (vhn200002)	Core Experience Team - Voice processing.
Developer	Alberto Escobar (axe220010)	Core Experience Team - Audio feedback system.
Developer	Kim Chau (kxc220015)	Vision & Nav Team - Camera API integration.
Developer	Shreyaa Arun (sxa220019)	Vision & Nav Team - LLM API integration.
Developer	Onkar Sandhu (oss210000)	Vision & Nav Team - Navigation logic.
QA / Testing	Kutsal Aksu (koa220001)	Responsible for testing across both teams, manages bug reports.

3. Managerial Process

3.1 Management Objectives and Priorities

The project's priorities are phased:

- Phase 1 Priority: To produce a clear, well-defined set of requirements and a visual design (mock-up) that accurately captures the project's goals and user scenarios.
- Phase 2 Priority: To develop a functional running prototype (MVP) that successfully implements the core features defined in Phase 1.

3.2 Assumptions, Dependencies, and Constraints

- Assumptions:
  - The user's smartphone has a functional camera.
  - The user's smartphone has a stable internet connection for LLM API calls.
- Dependencies:
  - The project is dependent on a third-party LLM (e.g., Google Gemini) for its object recognition capabilities. API availability and performance are external factors.
- Constraints:
  - The application must be developed using React Native to support both iOS and Android.
  - The project has a strict 4-week deadline, ending on October 2, 2025.

3.3 Risk Management

Risk ID	Risk Description	Probability	Impact	Mitigation Strategy
R-01	Technical: The LLM's object recognition is inaccurate or too slow for real-time navigation.	Medium	High	Focus the MVP on a limited set of 5-10 common, easily distinguishable objects (e.g., "door", "chair", "table"). Implement clear audio cues for when an object cannot be identified.
R-02	User Experience: Voice commands are not interpreted correctly or the audio feedback is confusing.	Medium	High	Design a simple, structured command syntax (e.g., "Find the"). Conduct early and frequent testing with team members to refine the audio cues.
R-03	Safety: The app provides incorrect guidance, potentially	Low	Critical	The app will include a mandatory disclaimer and onboarding message stating it is an assistive aid and not a replacement for a cane or other primary navigation tools. Guidance will be suggestive (e.g., "The door

Risk ID	Risk Description	Probability	Impact	Mitigation Strategy
	leading a user into an obstacle.			appears to be to your left") rather than prescriptive.
R-04	Project: The 4-week timeline is too aggressive, leading to burnout or an incomplete product.	High	Medium	The PM/PO will strictly enforce the MVP scope. Any non-essential features will be moved to a post-launch backlog.

3.4 Monitoring and Controlling Mechanisms

- Task Tracking: All work will be tracked as tickets in Linear. The board will be reviewed during the weekly progress meetings.
- Change Control: Any requested changes to the scope of an in-progress ticket will be evaluated by the PM/PO. If not critical, the change will be added to the backlog for consideration in a future iteration.

4. Technical Process

4.1 Methods, Tools, and Techniques

- Platform: React Native
- Version Control: Git, with the central repository hosted on GitHub.
- Object Recognition: Google Gemini API (or a similar LLM with vision capabilities).
- Project Management: Linear.app

4.2 Software Documentation

- Code Comments: All functions, components, and complex logic blocks must be documented using JSDoc-style comments to explain their purpose, parameters, and return values.
- Architecture: A README.md file in the project root will provide a high-level overview and setup instructions. A docs/ folder will contain a simple diagram of the application architecture.

4.3 Project Support Functions

- Code Repository: GitHub will be used for version control, code reviews (via Pull Requests), and issue tracking.
- Communication: A dedicated Discord group chat will be used for daily team communication.

5. Work Elements, Schedule, and Budget

Work Elements and Schedule

The project is divided into two distinct one-week phases, each with a major submission deadline.

Phase 1: Requirements & Design (Interim Project I)

- Timeline: Sep 2, 2025 – October 2, 2025
- Goal: To define the project's scope, requirements, and user experience through detailed analysis, scenario modeling, and design mockups.
- Key Activities:
  - Analyze "AS-IS" scenarios for visually impaired individuals navigating indoors.
  - Develop "TO-BE" scenarios illustrating how the Everywhere Locator app will improve this experience.
  - Create UI/UX mockups and wireframes for the application flow (the "mock-up prototype").
  - Draft the preliminary Workshop Requirements Specification (WRS) document.
  - Develop Questionnaire I to validate assumptions and gather feedback on the proposed solution.
  - Prepare the Interim Project I presentation slides.
- Deliverables (Due Thursday, Oct. 2 at 11:59 p.m.):
  - Interim Project I Presentation (PPT)
  - Preliminary Workshop Requirements Specification (WRS) Document
  - Questionnaire I
  - Evolving Project Plan (this document)

Phase 2: Prototyping & Validation (Final Project I)

- Timeline: Friday, October 3 – Thursday, October 9, 2025
- Goal: To develop a functional, running prototype based on the requirements from Phase 1 and to establish formal traceability.

- Key Activities:
    - Implement the core UI in React Native based on the Phase 1 mockups.
    - Develop the "running prototype" by integrating the camera, voice input, and a basic LLM connection for object recognition.
    - Finalize the WRS document with any learnings from the prototyping process.
    - Create the Compact and Full Forward Traceability matrices, linking requirements from the WRS to specific components in the prototype.
    - Calculate the requirements creeping rate.
    - Update the Project Plan and presentation with final progress.
- Deliverables (Due Thursday, Oct. 9 at 11:59 p.m.):
    - Final WRS Document
    - Compact-Forward-Traceability Matrix
    - Full-Forward-Traceability Matrix
    - Final Questionnaire
    - Updated Presentation (PPT), if any changes were made
    - Final evolving Project Plan

Budget

The project has a minimal budget allocated for essential third-party services.

- LLM API Usage Fees: $5.00