NAVAID: NAVIGATION ASSISTANT FOR THE VISUALLY IMPAIRED

Prabhav Singh, Sandesh Rangreji, Mukund Iyengar, Anubhav De
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218, USA
{psingh54, srangre1, miyenga2, ade11}@jh.edu

1 INTRODUCTION

Mobility and safe wayfinding are central to independence and quality of life for people with progressive retinal diseases. Retinitis Pigmentosa (RP) is an inherited retinal dystrophy that commonly causes night blindness and progressive constriction of the peripheral visual field; conservative prevalence estimates place RP at roughly 1 in 4,000–5,000 individuals worldwide, corresponding to on the order of 1–1.5M affected people when considered as a discrete diagnostic group (O'Neal, 2024; Cross et al., 2022; Natarajan, 2011). More broadly, at least 2.2 billion people live with some form of near- or distance-vision impairment globally, many of whom experience mobility limitations that assistive technologies aim to mitigate (World Health Organization, 2023; 2019).

Existing navigation aids span low-tech training and the white cane through to smartphone- and wearable-based prototypes that fuse computer vision, GPS, and audio/haptic feedback (Messaoudi et al., 2022; Okolo et al., 2024). Several commercial and research systems (Aira, Be My Eyes, Microsoft Seeing AI) provide human- or cloud-assisted visual interpretation, while open standards like Wayfindr emphasize short, well-timed audio cues for indoor wayfinding (Aira, 2025; Be My Eyes, 2025; Microsoft Seeing AI, 2023; Wayfindr, 2017). Reviews and recent user-centered studies show a proliferation of prototypes and mixed uptake: many systems remain research prototypes or blend human-assisted services rather than delivering low-latency, on-device hazard warnings that are tuned to specific visual phenotypes (Okolo et al., 2024; Soltani et al., 2025).

Our design seeks a middle path: use low-latency, high-recall perception to detect immediate collision-critical hazards and critical affordances (lateral obstacles, curb drops, narrow chokepoints) and translate that compact perceptual summary into short, actionable spoken instructions tuned to RP users' perceptual constraints and cognitive load. We plan to integrate macro-routing via the Google Maps Directions API for route geometry and turn segmentation, while keeping the hazard-detection and instruction loop local or hybrid (edge+ ephemeral cloud) to meet latency and privacy goals (Google Maps Platform, 2025). Optional haptics provide redundant, low-bandwidth cues for situations where audio alone is insufficient.

1.1 PROBLEM STATEMENT

Concrete task. Design and evaluate a mobile navigation assistant that (a) detects and warns about immediate, collision-critical hazards in the user's intended path, (b) issues concise, actionable spoken instructions, and (c) adapts instruction verbosity and modality to the user's residual vision, mobility skill, and context (e.g., crowded sidewalk vs quiet corridor). The target operational domain is short urban and indoor segments[1] where pedestrian hazards (other people, sudden lateral obstacles, curb edges, narrow gaps) and rapid context changes are common.

RP-specific constraints. RP and many MSVI phenotypes primarily reduce peripheral field and contrast sensitivity rather than central acuity. This produces particular mobility failure modes (missed lateral hazards, delayed awareness of approaching obstacles, and increased reliance on auditory scanning and head movement) that differ from central-vision loss or generic low vision (Timmis et al., 2017; Sayed et al., 2019). These constraints motivate the following design priorities:

[1]We might not extend to indoor segments in our version for this class since there are competings apps that do the same. Our focus will be on outdoor scenarios.

- **High recall for lateral hazards:** Prioritize warnings that prevent collisions from the periphery over exhaustive scene descriptions.
- **Conciseness and timing:** Use short phrases, minimal filler, and early warnings that provide sufficient lead time for gait and steering corrections.
- **Low false-alarm rate:** Minimize spurious alerts that erode trust and increase cognitive burden.
- **Audio + haptic primary outputs:** Haptic feedback can serve as alternative to audio feedback to prevent overwhelming the user in cases where the feedback need not be too severe.

1.2 TARGET DEMOGRAPHIC

NAVAID is explicitly targeted at adults with Retinitis Pigmentosa and related progressive peripheral-field losses (moderate-to-severe MSVI) who:

- Rely primarily on audio for spatial guidance.
- Perform independent navigation (e.g., commuting, errands) but face frequent lateral-hazard risk.
- Are willing to use a phone or wearable for assistive prompts.

RP's typical symptom profile—narrowing visual field and reduced contrast sensitivity—creates mobility needs (lateral hazard detection, early warnings, brief actionable prompts) that an audio-first system can address more directly than systems designed for central-vision impairment or for sighted low-vision users who primarily benefit from visual overlays (Timmis et al., 2017; Wayfindr, 2017; Paratore et al., 2023).[2]

Secondary users include people with mixed low vision who opt into multimodal modes; in those cases NAVAID supports a configurable visual overlay, but the baseline interaction and evaluation emphasize audio and optional haptics. Personalization (residual field, walking speed, verbosity preference) is integrated into the instruction policy so that warnings remain timely and minimally intrusive.

2 DESIGN

Figure 1 shows the overall system architecture for NAVAID: sensor input (Camera) feeds a lightweight perception module, which produces a compact contextual representation used by a decision layer to generate concise spoken instructions and optional haptic cues. Macro-routing (turns, route geometry) is obtained from the Google Maps Directions API and is combined with local hazard detections in the Context Creation and Decision layers shown in the figure.

Platform and interaction model. NAVAID is a mobile-first application (phone or wearable companion)[3] whose primary interaction channel is audio with optional haptic redundancy. Users start the app, provide a start and end location, and then begin walking; the app supplies macro-turn cues (from Google Maps Directions API) plus local, high-priority hazard warnings produced by the perception + planner loop (Google Maps Platform, 2025; Wayfindr, 2017). Simplicity is key: the main screen shows only three large actions (Start, Pause, End) and a small settings control (voice/haptic verbosity), following well-established user-centered design principles for assistive mobile apps (Abidi et al., 2024).

[2]Visual overlays and map-based visual augmentations are valuable for low-vision users with usable central vision, but they are ineffective as a primary modality for many RP users because progressive peripheral field loss prevents reliable detection of lateral hazards presented visually; overlays also increase cognitive load when the user must alternate between auditory orientation and visual inspection. Many deployed/experimental solutions emphasize either (a) human-assisted visual interpretation (Aira, Be My Eyes, Microsoft Seeing AI) or (b) indoor audio standards and beacon-based systems (Wayfindr) that require environmental instrumentation or human-in-the-loop services; these approaches demonstrate valuable features but do not substitute for a low-latency, audio-first hazard-warning loop targeted at RP mobility failure modes (Aira, 2025; Be My Eyes, 2025; Microsoft Seeing AI, 2023; Wayfindr, 2017; Okolo et al., 2024).

[3]Ideally we could do this in smart-glasses. Hanging mobiles serve as simpler way to prove the working.

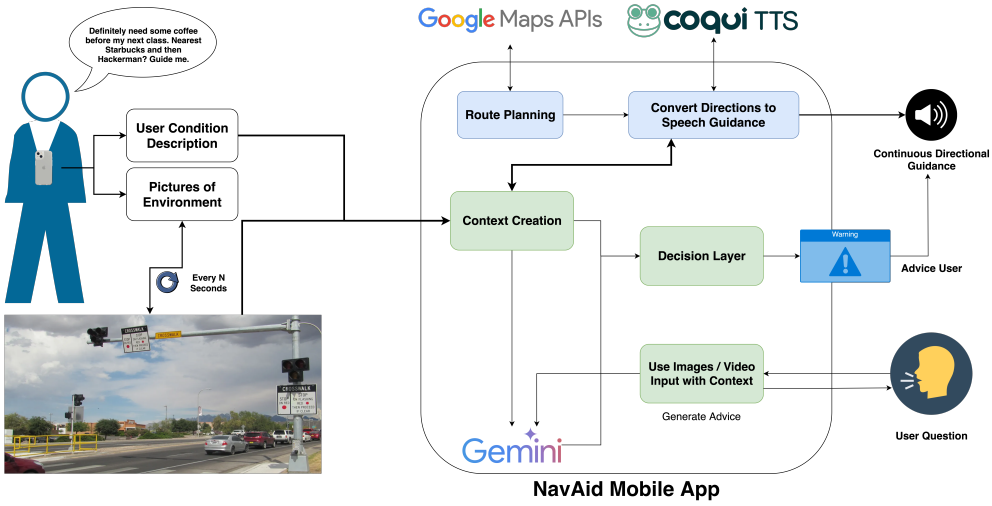


Figure 1: NAVAID architecture (camera → lightweight perception → context → decision/policy → TTS/haptics). The pipeline emphasizes a low-bandwidth perceptual summary (hazard type, relative bearing, time-to-contact) rather than full scene descriptions.

Why audio-first and simple UI? People with moderate-to-severe MSVI, especially RP, depend on auditory cues and have limited value for visual overlays; User-centered work emphasize short, well-timed audio segments and route segmentation to reduce cognitive load (Wayfindr, 2017). Reviews of assistive navigation prototypes show adoption and usability are strongly coupled to simplicity and short audio messages rather than verbose descriptions (**?**Soltani et al., 2025).

What the app collects? To personalize warnings and instruction timing we collect a small, privacy-conscious profile tailored for mobility performance.

- **Residual-field self-report:** Structured questionnaire that approximates peripheral field extent (e.g., can see objects at center only, limited peripheral to X degrees). This helps set lead-time thresholds and lateral-warning sensitivity (Timmis et al., 2017).
- **Walking speed and stride calibration:** Brief calibration walk (10–30s) to estimate comfortable walking speed and step length; used to translate time-to-contact into meters and choose warning lead times.[4]
- **Haptic preference and dominant-side setting:** Which side to place vibratory cues (left/right) and whether the user prefers vibrate-first, audio-only, or both.
- **Device mounting and carriage habit:** Where the phone will be placed (chest mount, belt, handheld). This affects camera pose priors and occlusion handling.
- **Route and avoided features:** User-specified preferences such as avoid stairs, prefer curb ramps, avoid crowded pedestrian plazas — used to tune routing and warning aggressiveness.

Collecting these signals is inexpensive for the user but materially improves safety, latency choices, and personalization compared with generic profiles; prior work on personalized inertial/PDR and mobile assistive tools motivates these specific signals (Moisan et al., 2025; Project, 2021).

Perception and compute tradeoffs. NAVAID delegates scene perception and semantic understanding to multimodal model pipelines (edge- or cloud-hosted CV + MLLM services) that perform affordance recognition, ambiguity resolution, and hazard classification (Liang, 2024; Yin et al., 2024; Huang et al., 2024; Abidi et al., 2024). These remote models supply compact hazard summaries that feed the decision layer; text-to-speech rendering is performed locally (Coqui TTS or equivalent) so audio output remains immediate and private even when external services are used for perception (Coqui TTS, 2024). We explicitly separate the *safety-critical* rendering path (local TTS + deterministic

[4]For example if the person walks fast, we need more photos per minute.

canned fallbacks) from the *enrichment* path (MLLM-derived clarifications or richer instructions) so that optional remote enrichment never blocks or degrades immediate safety cues; exact timing and confidence thresholds will be defined in the Methods and evaluation protocol.

TTS and audio stack. We use a local TTS stacknfor low-latency, private voice rendering and allow cloud TTS as an opt-in for higher-quality voices when connectivity and privacy preferences permit (Coqui TTS, 2024). Instruction phrasing is constrained to short templates (2–5 words for instant alerts; slightly longer for turn/route instructions) following Wayfindr guidance about segmented route cues and pilot-able chunks of information (Wayfindr, 2017).

Haptics - when and why? Haptics act as a primary or redundant cue in high-risk or audio-unfriendly moments. We define concrete haptic trigger conditions:

1. **Imminent lateral collision:** If time-to-contact falls below a safety threshold, trigger an immediate strong vibrate alert on the side corresponding to the obstacle bearing. This provides sub-second redundancy when audio may be masked. Evidence shows haptic patterns improve route following and obstacle awareness in BLV users (Khusro et al., 2022).
2. **Silent mode or public spaces:** Of the user has enabled silent mode or is in user-specified noisy contexts, default to haptics with a minimal audio fallback.
3. **User preference:** Users with hearing limitations can use haptics as primary output.

Haptic patterns are short, distinct, and mapped consistently to affordances (left/right/urgent) to avoid user confusion (Khusro et al., 2022).

2.1 POSSIBLE EXTENSIONS

Dual-camera (Front & Rear) configuration. Adding a second camera (rear or a second front-facing camera with offset baseline) enables near-360 lateral coverage or short-baseline stereo for depth estimates and more robust lateral hazard detection, especially for overhanging obstacles and approaching cyclists (Project, 2021; Eye & Ear, 2021).

Smart-glasses integration. A glasses-mounted camera + open-ear (bone-conduction) audio allows more natural, hands-free operation and reduces occlusion from pockets; modern smart-glasses (commercial examples and research prototypes) demonstrate feasibility for object recognition and scene description, though battery life and cost remain constraints (Verge, 2025; OrCam, 2024). Glasses integration is a natural extension for users who prefer wearable form-factors; NAVAID's pipeline can run on a paired smartphone (edge compute) with the glasses acting as sensors and audio output.

3 MACHINE LEARNING BACKEND

This section summarizes the modules that power NAVAID's routing, audio rendering, perception, and decision logic: (1) a Maps API to provide macro-navigation and step geometry, (2) a local text-to-speech (TTS) engine with cloud fallback, (3) a multimodal Core AI model for image-to-hazard summaries, and (4) a decision module that fuses route context and periodic image captures to produce short spoken or haptic instructions.

3.1 MAPS API

We use the Google Maps Directions API for macro-routing and step segmentation (walking mode) so the app has access to route geometry and turn segmentation for the user's requested path (Google Maps Platform, 2025). The app fetches the route at session start and uses the step list to align local hazard warnings with imminent route decisions; this keeps the backend straightforward and avoids on-device route computation (Google Maps Platform, 2025).

3.2 TEXT2SPEECH MODEL

Text rendering is local-first: we deploy a compact, locally runnable TTS engine for immediate, private audio output (examples: Coqui TTS / OpenTTS family), and we evaluate higher-quality cloud voices (Google Cloud Text-to-Speech WaveNet) (Coqui TTS, 2024; Google Cloud, 2025) as

opt-in fallbacks where users prefer them. Local TTS handles short, low-latency prompts and canned safety messages; cloud TTS is reserved for user-selected voice quality preferences.

3.3 CORE AI MODEL

The Core AI layer is a multimodal model that consumes recent camera frames plus compact route context and returns a concise hazard summary. Our initial candidate is Google Gemini (multimodal API), which we will evaluate alongside other state-of-the-art MLLMs to balance capability and latency (Liang, 2024; Google AI, 2025). The model's expected structured output (one-per-detection) is intentionally compact; each detection contains at minimum:

- `item_type`: single-word label (e.g., *road-crossing*, *construction-zone*, *person*),
- `description`: one short sentence describing the observed item or hazard,
- `general_instructions`: one short sentence describing how to avoid it (actionable phrasing),
- `confidence`: scalar in $[0, 1]$,
- `urgency`: coarse bucket (low/medium/high).

This compact schema avoids verbose scene descriptions while providing immediately actionable information (Liang, 2024; Abidi et al., 2024).

3.4 DECISION MAKING SYSTEM

The decision module continuously fuses Google Maps step context, recent model detections, and the user profile to decide when to render audio or haptic cues. The app captures images every N seconds (adaptive N chosen from the user profile and walking speed) and sends those frames to the Core AI for hazard summaries; when a returned detection meets personalization thresholds, the decision module issues a short local TTS phrase (or haptic pattern) built from the model's `general_instruction` field. This loop keeps the runtime behavior simple and interpretable: periodic image capture $\rightarrow$ Core AI hazard summary $\rightarrow$ personalization filter $\rightarrow$ local TTS / haptic output.

4 DATA

We use a small combination of public corpora and a minimal in-situ collection to evaluate TTS, perception, and end-to-end behavior.

4.1 PUBLIC DATASETS

TTS / instruction-text corpora. For TTS evaluation we use human-written navigation instruction corpora drawn from Touchdown and Room-to-Room (R2R). Touchdown provides realistic outdoor, street-style instruction sentences useful for outdoor TTS tests (Chen et al., 2020). R2R provides additional turn-by-turn phrasing (indoor) to increase linguistic variety for TTS evaluation (Anderson & et al., 2018). These will be used just to evaluate and choose our TTS systems.

Scene and Hazard benchmark. For outdoor hazard identification we use Mapillary Vistas as a single, high-coverage urban dataset whose street-level annotations map well to pedestrian hazards (construction, crosswalks, bikes, vehicles, curb features) (Neuhold et al., 2017). We can use a subset of this dataset to get images with hazards and measure recall of MLLMs to choose the one we will use.

4.2 MANUAL JHU MINI-DATASET

For the class project pilot we will collect a very small, local dataset around JHU to sanity-check end-to-end behavior:

- **Size:** $\approx$100 images total.

- **Capture:** Chest/front-facing phone mount; capture images every $\sim$1–2s while walking short routes that include common campus hazards (intersections, curb ramps, construction zones, crowded sidewalks, bicyclists/scooters).[5]
- **Labels:** Per-image binary label (Hazard: Yes/No) plus a short 1–6 word note when Hazard=Yes (e.g., `construction_blocking_sidewalk`, `person_in_path`). No bounding boxes; only team self-labeling required.
- **Use:** Quick end-to-end sanity checks (does Core AI flag hazards present in team images; are generated instructions plausible for short pilot walks).

5 EVALUATION

This section describes the offline and in-field evaluation protocol for NAVAID. We tie each evaluation stage to the datasets described earlier (Touchdown / R2R for TTS text, Mapillary Vistas for outdoor hazard benchmarks, and the JHU mini-dataset for end-to-end sanity checks).

5.1 OFFLINE MODULE EVALUATION

TTS evaluation (text-only). Use a held-out set of $\sim$100–200 instruction sentences sampled from Touchdown and R2R (Chen et al., 2020; Anderson & et al., 2018). Metrics:

- **Latency:** time from instruction text to start-of-audio (ms). Measure median and 95th percentile.
- **Intelligibility / Naturalness:** subjective Mean Opinion Score (MOS) on a 1–5 scale collected from at 4 raters for a subset ($\sim$30) of synthesized sentences; report mean $\pm$ 95% CI.
- **Instruction confidence calibration:** measure whether the system's internal confidence (if any) correlates with human acceptability; report Expected Calibration Error (ECE) or a reliability diagram for the confidence scores.

These metrics assess whether local TTS output is fast, intelligible, and calibrated for deployment.

Hazard detection (offline, dataset-based). Evaluate detection performance using Mapillary Vistas (mapped to binary hazard/non-hazard for our target classes) and the JHU mini-dataset for realistic user-shot images (Neuhold et al., 2017). Metrics and definitions:

- **True Positives (TP):** hazards correctly detected.
- **False Negatives (FN):** hazards present but missed (critical).
- **False Positives (FP):** non-hazard incorrectly flagged.
- **Recall / Sensitivity:** $\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}$. **Goal: maximize recall; false negatives (misses) are most critical.**
- **Precision:** $\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$. Useful but secondary; high FP rate is tolerable compared to FN for safety-critical hazards.
- **F1-score:** harmonic mean of precision and recall for reporting.
- **Critical Hazard Miss Rate (CHMR):** proportion of hazards that were not warned within an acceptable lead-time window (see field evaluation for lead-time definition).
- **Detection Lead Time:** measured time between detection/warning issuance and the moment the hazard would be encountered under observed walking speed; report median and interquartile range.
- **Per-class recall:** recall measured separately for key hazard classes (construction, person, bicycle, curb/drop).

[5] Construction is currently common on the JHU campus; include construction scenarios in the manual collection.

5.2 CLOSED COURSE TRIAL

Setup. A short, instrumented course with known, repeatable hazards (e.g., placed obstruction, simulated construction barrier, marked narrow passage). Recruit a small number of participants (e.g., 4–6) to perform multiple runs each. Use the JHU mini-dataset images to seed scenario selection.

Metrics. Evaluate system behavior in a controlled, repeatable environment:

- **Critical Hazard Miss Rate (CHMR):** fraction of intentional hazards that produced no timely warning.
- **Warning Lead Time:** time between warning and reach-time (median per hazard type).
- **False Alarm Rate (FAR):** number of spurious warnings per minute or per run.
- **Success Rate:** proportion of runs completed without intervention.

5.3 SUPERVISED REAL-WORLD ROUTES

Setup. Real-world, supervised point-to-point routes (example: campus route from a common origin to a destination such as "Starbucks to Malone Hall") that include natural hazards and dynamic context. Participants complete routes while wearing the system; a trained safety monitor walks nearby to intervene if necessary.[6]

Metrics. Measure end-to-end system effectiveness and user experience in natural conditions:

- **Detected hazards per route:** number of hazards the system flagged during the route.
- **Proportion of hazards detected:** (system-detected hazards that correspond to ground-truth hazards labeled by the safety monitor/post-hoc review) divided by total hazards observed.
- **Critical Hazard Miss Rate (CHMR):** as above, but evaluated across full routes.
- **Intervention count and latency:** number of safety interventions per route and average time between hazard onset and intervention.
- **Route success / completion without incident:** binary per-run outcome.
- **Subjective measures:** short post-route questionnaires (usability, trust, perceived workload). Suggested instruments: short SUS-like scale and single-item trust rating.
- **System telemetry:** aggregate logs of model round-trip latency, TTS latency, and number of MLLM calls per route (for cost/latency analysis).

5.4 MAPPING TO DATA AND MINIMAL SAMPLE SIZES

- **Offline TTS:** evaluate on $\sim$100–200 sentences from Touchdown/R2R for MOS and WER proxies (Chen et al., 2020; Anderson & et al., 2018).
- **Offline hazard detection:** evaluate on Mapillary subsets (held-out scenes mapped to hazard/non-hazard) and on the JHU mini-dataset ($\sim$100 images) for realistic user-shot performance (Neuhold et al., 2017).
- **Closed course:** 4 participants $\times$ 3–5 runs each gives a small but repeatable set of trials for initial tuning.
- **Supervised routes:** 2–3 supervised route runs suffice for a class-project pilot; focus on qualitative and safety-critical numeric signals rather than narrow statistical significance.

6 IMPLICATIONS & ETHICS

NAVAID raises both clear benefits and concrete challenges. Below we state the positive benefits and challenges to ethics in our solution:

[6] All blindfolded or simulated-vision trials must follow institutional safety protocols. A safety monitor must be present; participants are not put at risk. Note that this is not exactly our target demographic but its close.

Benefits

- **Privacy-sensitive:** data are processed in the cloud in real time with no storage of raw video or audio by default.
- **Inclusive:** designed to work across diverse environments and user needs, reducing reliance on costly infrastructure or human assistance.
- **Stakeholder impact:** empowers blind/low-vision users while influencing caregivers, accessibility advocates, and urban design considerations.
- **Scalability:** baseline phone deployment with an explicit path toward wearable/smart-glasses form factors.
- **Anonymized disability data:** any collected information about a user's vision loss and related disability context is de-identified and anonymized prior to storage or analysis.

Challenges

- **Connectivity dependence:** requires stable network access; performance drops in low-network areas.
- **Shared-space impact:** frequent audio cues may affect nearby pedestrians in crowded environments (social acceptability concerns).
- **Over-trust in AI:** users may overly depend on automated guidance even when the system is uncertain.
- **Other privacy concerns:** the system tracks user location for routing and safety — we will not reveal exact location to external services (e.g., MLLM providers) without explicit safeguards and user consent.

7 MILESTONES AND TIMELINES

Milestone	Duration	Days (approx)	Key deliverables
Milestone 1: Individual Model Evaluation	8 days	Day 0–7	TTS latency/MOS/WER tables; Image-to-guidance recall/precision/CHMR; 10 qualitative examples.
Milestone 2: End-to-End Implementation	14 days	Day 8–21	Runnable prototype (phone + backend); integration docs; automated evaluation scripts.
Milestone 3: Closed Course Trials	7 days	Day 22–28	Closed-course report: CHMR, warning lead time, FAR, intervention counts; prioritized bug list.
Milestone 4: App Dev & Real-World Routes	14 days	Day 29–42	App build; supervised-route metrics and telemetry; user feedback report.

Table 1: Milestones, durations, and deliverables.

7.1 MILESTONE 1: INDIVIDUAL MODEL EVALUATION (8 DAYS)

Part 1 — TTS System Evaluation (4 days): implement local TTS baseline; synthesize $\sim$100 sentences from Touchdown/R2R; measure latency, MOS proxy and WER proxy; produce latency/MOS/WER table and brief findings.

Part 2 — Offline Image-to-Guidance Evaluation (4 days): run Core AI on Mapillary subset + JHU mini-dataset; compute recall/precision and CHMR proxies; deliver performance table and 10 qualitative examples.

7.2 MILESTONE 2: END-TO-END IMPLEMENTATION (2 WEEKS)

Integrate Maps API, Core AI, Decision Module, and local TTS into a runnable prototype with logging and basic automated evaluation scripts. Deliver prototype, integration notes, and evaluation hooks.

7.3 MILESTONE 3: CLOSED COURSE TRIALS (1 WEEK)

Run controlled trials (4 participants / multiple runs) on a short instrumented course. Deliver closed-course report with CHMR, lead time, FAR, intervention counts, and prioritized fixes.

7.4 MILESTONE 4: APP DEVELOPMENT & SUPERVISED REAL-WORLD ROUTES (2 WEEKS)

Week 1: finalize audio-first app UI and personalization settings. Week 2: supervised point-to-point route tests (safety monitor present); deliver supervised-route metrics, telemetry, and user feedback.

REFERENCES

M. H. Abidi et al. A comprehensive review of navigation systems for visually impaired users. *arXiv / ScienceDirect*, 2024. URL https://www.sciencedirect.com/science/article/pii/S2405844024078563.

Aira. Aira — visual interpretation service. https://aira.io/, 2025. Accessed 2025-10-05.

Peter Anderson and et al. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, 2018. URL https://github.com/peteanderson80/Matterport3DSimulator.

Be My Eyes. Be my eyes — about / community statistics. https://www.bemyeyes.com/, 2025. Accessed 2025.

Daniel L. Chen, Meng Gao, and et al. Touchdown: Natural language navigation and spatial reasoning in visual street environments. In *EMNLP / ACL Workshops*, 2020. URL https://github.com/salesforce/touchdown.

Coqui TTS. Coqui tts documentation. https://coqui.ai/, 2024.

N. Cross et al. Retinitis pigmentosa: Burden of disease and current management. *Eye (London)*, 2022. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9232096/. PMCID: PMC9232096.

Mass Eye and Ear. Wearable devices can reduce collision risk in blind and visually impaired people (press release). https://masseyeandear.org/news/press-releases/2021/07/wearable-devices-can-reduce-collision-risk-in-blind-and-visually-impaired-people, 2021.

Google AI. Gemini api documentation. https://ai.google.dev/gemini-api/docs, 2025. Accessed 2025-10-05.

Google Cloud. Text-to-speech documentation (wavenet). https://cloud.google.com/text-to-speech/docs, 2025. Accessed 2025-10-05.

Google Maps Platform. Directions api — google maps platform documentation. https://developers.google.com/maps/documentation/directions, 2025. Accessed 2025.

J. Huang et al. A survey on evaluation of multimodal large language models. arXiv preprint arXiv:2408.15769, 2024. URL https://arxiv.org/abs/2408.15769.

S. Khusro et al. Haptic feedback to assist blind people in indoor navigation. *Sensors (Basel) / PMC*, 2022. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8749676/.

C. X. Liang. A comprehensive survey and guide to multimodal large language models. arXiv preprint arXiv:2411.06284, 2024. URL https://arxiv.org/abs/2411.06284.

M. D. Messaoudi et al. Review of navigation assistive tools and technologies for the visually impaired. *Sensors (Basel)*, 2022. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9606951/.

Microsoft Seeing AI. Seeing ai — a visual assistant for the blind (microsoft garage). https://www.seeingai.com/, 2023. Accessed 2025.

T. Moisan, H. Fu, V. Renaudin, and M. I. Sayyaf. Personalized inertial navigation for the visually impaired. 2025.

S. Natarajan. Retinitis pigmentosa: A brief overview. *Indian Journal of Ophthalmology*, 59(5): 343–346, 2011. doi: 10.4103/0301-4738.83608. PMID: 21836337; Accessed 2025-10-05.

Georg Neuhold, Tobias Ollmann, Simon Rota Bulo, and Peter Kontschieder. Mapillary vistas dataset: Street-level image data for semantic understanding. https://www.mapillary.com/dataset/vistas, 2017. Accessed 2025-10-05.

Gabriel I. Okolo, Turke Althobaiti, and Naeem Ramzan. Assistive systems for visually impaired persons: Challenges and opportunities for navigation assistance. *Sensors*, 24(11):3572, 2024. doi: 10.3390/s24113572. URL https://doi.org/10.3390/s24113572.

TB O'Neal. Retinitis pigmentosa. StatPearls [Internet], 2024. URL https://www.ncbi.nlm.nih.gov/books/NBK519518/.

OrCam. Orcam device description. https://www.orcam.com/, 2024.

M. T. Paratore et al. Exploiting the haptic and audio channels to improve cognitive mapping for visually impaired users. *Frontiers / PMC*, 2023. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC9942617/.

SENSATION Project. Sensation: A light wearable, portable and cost-effective system to support assistive navigation of blind or visually impaired pedestrians. https://www.researchgate.net/publication/367371947_SENSATION_A_light_wearable_portable_and_cost-effective_system_to_support_assistive_navigation_of_blind_or_visually_impaired_pedestrians, 2021.

A. M. Sayed et al. Towards improving the mobility of patients with peripheral visual field loss using digital spectacles (dspecs). *Artificial Organs / Ophthalmology (PMC)*, 2019. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7002240/.

Iman Soltani et al. User-centered insights into assistive navigation technologies for individuals with visual impairment. https://arxiv.org/pdf/2504.06379.pdf, 2025. preprint; Accessed 2025.

M. A. Timmis et al. Visual search behavior in individuals with retinitis pigmentosa. *Investigative Ophthalmology & Visual Science*, 2017. URL https://iovs.arvojournals.org/article.aspx?articleid=2655023.

The Verge. These smart glasses use ai to help low-vision users. https://www.theverge.com/, 2025.

Wayfindr. Wayfindr open standard for audio wayfinding, 2017. URL https://www.wayfindr.net/wp-content/uploads/2017/12/Wayfindr-Open-Standard-Rec-1.1.pdf.

World Health Organization. World report on vision. https://www.who.int/publications-detail-redirect/world-report-on-vision, 2019. Accessed 2025-10-05.

World Health Organization. Blindness and vision impairment — fact sheet, 2023. URL https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment.

S. Yin et al. A survey on multimodal large language models. *National Science Review*, 2024. URL https://academic.oup.com/nsr/article/11/12/nwae403/7896414.

APPENDIX

A. INSIGHT STATEMENTS WORKSHEET

This worksheet summarizes insights gained during the empathy phase (Task 1). Insights were gathered through secondary research including scientific literature on RP mobility Timmis et al. (2017); Sayed et al. (2019), online forums and discussions from the RP community (Foundation Fighting Blindness forums, Reddit `r/blind`), YouTube first-person navigation videos from RP users, and articles from accessibility advocacy organizations.

Theme 1: Peripheral Vision Loss & Lateral Hazard Detection

Insights:

1. RP users commonly miss lateral hazards—including pedestrians, cyclists, and obstacles approaching from the sides—because progressive peripheral field loss eliminates awareness of objects outside the narrowing central vision "tunnel." This creates frequent near-collision events during everyday navigation.
2. Users compensate for peripheral loss through increased head-turning and auditory scanning, but these strategies are cognitively demanding and cannot fully replace lost peripheral awareness, especially in dynamic environments with multiple moving hazards.
3. Ground-level hazards such as curb edges, sudden drop-offs, and uneven pavement are particularly dangerous because they fall outside the remaining central vision field and are often detected too late to prevent tripping or falls.

Theme 2: Cognitive Load & Information Overload

Insights:

1. Existing navigation systems that provide verbose, continuous descriptions significantly increase cognitive burden when users are already managing limited vision, orientation, and obstacle avoidance simultaneously. Users report "tuning out" systems that talk too much.
2. Effective navigation requires warnings with sufficient lead time (2-4 seconds before hazard encounter) to allow gait adjustment and steering corrections, but many prototype systems issue warnings too late or at inconsistent intervals.
3. High false-alarm rates erode user trust and cause alert fatigue, leading users to ignore or disable warning systems even when genuine hazards are present. A single missed critical hazard is more dangerous than several false positives, but trust requires balance.

Theme 3: Independence, Autonomy & Technology Acceptance

Insights:

1. RP users highly value independent navigation for daily activities (commuting to work, running errands, social visits) and express frustration with solutions that require human assistance or special infrastructure that limits where and when they can travel.
2. Users strongly prefer audio-first interfaces over visual overlays because their remaining vision is unreliable for peripheral information and visual attention diverts from critical orientation cues. Haptic feedback is welcomed as a complementary channel, especially in noisy environments.
3. Current commercial solutions (Aira, Be My Eyes) provide valuable services but depend on human operators or cloud connectivity that may not always be available, creating gaps in coverage during critical navigation moments. Users want on-device, immediate hazard warnings.

B. "HOW MIGHT WE...?" WORKSHEET

This worksheet translates insights from the empathy phase into actionable design questions (Task 2). Each question is framed to allow multiple solution approaches while remaining focused on specific user needs.

Insight: RP users commonly miss lateral hazards (people, cyclists, obstacles) approaching from their peripheral blind zones.

How might we... provide early, directional warnings about obstacles approaching from users' peripheral blind zones with sufficient lead time (2-4 seconds) to enable collision avoidance through gait or steering adjustments?

Insight: Ground-level hazards (curb edges, drop-offs, uneven pavement) fall outside remaining central vision and cause trips/falls.

How might we... detect and warn about ground-level hazards—including curb edges, steps, and surface discontinuities—early enough for users to safely navigate around or prepare for elevation changes?

Insight: Verbose navigation instructions increase cognitive load when users are managing multiple tasks (orientation, obstacle avoidance, auditory scanning).

How might we... deliver concise, actionable navigation cues (2-5 words) that provide critical information without overwhelming users who are simultaneously managing orientation, balance, and environmental awareness?

Insight: High false-alarm rates erode trust and cause users to ignore or disable warning systems.

How might we... maintain high recall for critical collision hazards (minimize false negatives) while keeping false-alarm rates low enough to preserve user trust and prevent alert fatigue over extended use?

Insight: Users prefer audio-first interfaces because visual overlays require visual attention that is unreliable and diverts from orientation cues.

How might we... design an audio-first warning system that provides spatial awareness and directional guidance without requiring visual attention or creating confusion in acoustically complex environments?

Insight: Existing solutions require human assistance or infrastructure, limiting independence and creating coverage gaps.

How might we... create an on-device, real-time hazard detection and warning system that operates independently without requiring external infrastructure, human operators, or continuous cloud connectivity?

Insight: Effective warnings require 2-4 seconds lead time, but many systems issue warnings too late or inconsistently.

How might we... ensure that hazard warnings are issued with consistent, sufficient lead time based on the user's walking speed and the hazard's urgency level, enabling reliable corrective action?

C. BRAINSTORMING SESSION DOCUMENTATION

Meeting 1: Initial Ideation — September 26, 2025

Location: Malone Grad Lounge — Duration: 90 minutes — Attendees: All team members

Session Goals:

- Generate ideas addressing our "How Might We?" questions
- No filtering — wild ideas encouraged
- Aim for 50+ ideas

Brainstorming Output (Post-it Notes):

Smart glasses with camera Detect obstacles Audio warnings	Phone on chest mount Hands free Front-facing camera	Haptic vest Vibrate patterns for direction Left/right warnings	White cane with sensors Ultrasonic Beeping sounds
AI describes everything it sees Full scene description	Only warn about URGENT stuff No clutter Short phrases	Google Maps integration Turn-by-turn Add hazard layer	Bone conduction audio Keep ears free Ambient awareness
Train custom model on sidewalk images RP-specific?	Use GPT-4 Vision Or Gemini Cloud-based	Depth camera LIDAR? Too expensive?	Crowdsource hazard reports Community map Real-time updates
Personalize warnings Fast/slow walkers Field of view	Indoor + Outdoor Start with outdoor Sidewalks easier	Privacy concerns Local processing No video storage	Testing with actual RP users JHU hospital? Wilmer Eye?

Key Insights from Meeting 1:

- Audio-first is critical — RP users can't rely on visual displays
- Phone mount (chest/belt) more practical than glasses for MVP
- Need to balance detail vs. speed — can't describe everything
- Privacy: local TTS, ephemeral cloud processing for hazard detection
- Google Maps for macro routing + our system for local hazards

Concepts Selected for Prototyping:

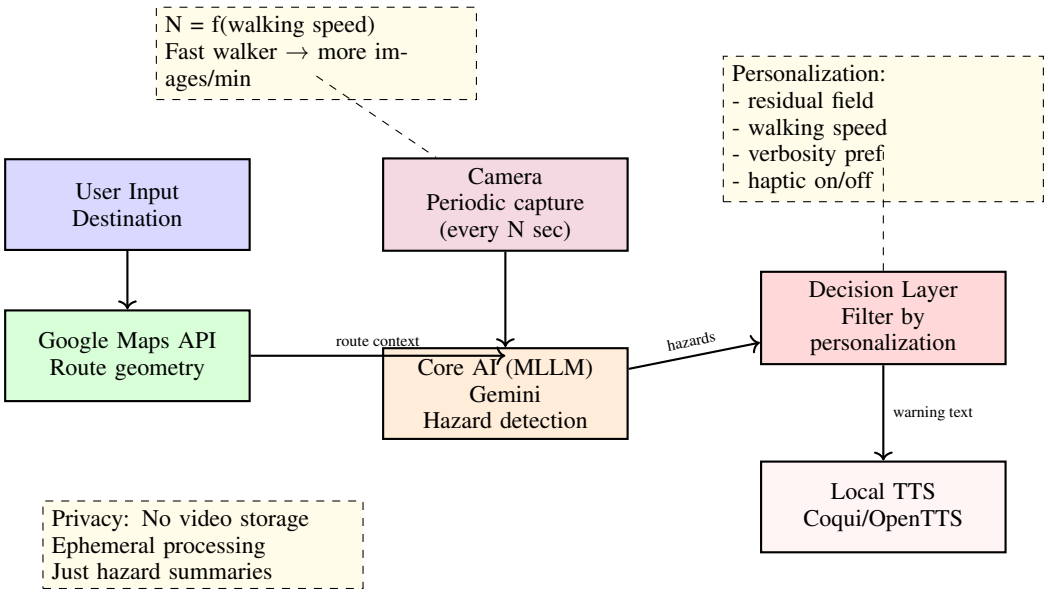
1. Phone-based camera system with chest/belt mount
2. Cloud MLLM for hazard detection (high recall priority)
3. Local TTS for immediate audio warnings
4. Google Maps integration for turn-by-turn
5. Personalization based on walking speed and residual vision

Meeting 2: Architecture Refinement — September 28, 2025

Location: Brody Learning Commons — Duration: 60 minutes — Attendees: All team members

Session Goals: Refine system architecture from Meeting 1, Define specific data flows, Identify technical challenges, Plan evaluation approach

Detailed Architecture Discussion:



Technical Challenges Identified:

Challenge 1: Latency
MLLM processing can take 2-5 seconds. Need hazard warnings in <1 sec.
Solution idea: Hybrid approach — cache common scenarios, use edge processing where possible

Challenge 2: False Positives vs False Negatives
Missing a hazard = dangerous. Too many false alarms = user ignores system.
Solution idea: High recall threshold, personalized sensitivity

Challenge 3: Depth/Distance Estimation
How far away is obstacle? Important for timing warnings.
Solution idea: Ask MLLM to estimate distance? Check if accurate enough.

Data and Evaluation Planning:

Rough notes:

- **TTS eval:** Touchdown + R2R datasets, measure latency + MOS
- **Hazard detection:** Mapillary Vistas (urban street scenes), measure recall
- **JHU mini-dataset:** Walk around campus, capture 100 images with hazards
- **Evaluation metrics:**
  - Critical Hazard Miss Rate (CHMR) — most important!
  - Warning lead time (seconds before encounter)
  - False alarm rate (per minute)
  - User trust (subjective, post-route survey)
- **Real-world test:** Supervised walks (safety monitor present)

Decisions Made:

1. System architecture finalized (see diagram above)
2. Gemini as primary MLLM candidate (will benchmark against others)
3. Local Coqui TTS for audio rendering
4. Prioritize outdoor navigation for MVP (indoor is already done)
5. High recall for hazard detection — better safe than sorry

Meeting 3: TA Feedback Integration — October 4, 2025

Location: Virtual (Zoom) — Duration: 60 minutes — Attendees: Team + Project Mentors (CA/TA)

Feedback Received from TAs/CAs:

1. Distance Measurement Discussion
"How are you getting actual distance to obstacles? This is critical for warning timing. Can your MLLM provide depth information? What if you use multiple cameras for stereo vision?"

2. Add Distance Prediction Metric
"You need a metric: Predicted Distance vs Actual Distance. How will you measure ground truth? Consider using depth camera for validation during testing."

3. More In-Depth RP Research
"Do more research on exact issues RP patients face. Can you consult with someone who actually has RP? Reach out to Wilmer Eye Institute or RP advocacy groups for user interviews."

4. Scene Description Feature
"What if user wants to ask about something specific? Like 'what's that black spot ahead?' Consider adding query-based scene description, not just hazard warnings."

Team Response and Action Items:

Re: Distance Measurement
Action: Research MLLM depth estimation capabilities. Gemini's spatial reasoning can provide approximate distances from monocular images. For MVP, rely on MLLM distance estimates. Future: explore dual-camera setup for better depth.
Added to proposal: Mention distance estimation as MLLM output, note dual-camera as future extension.

Re: Distance Metric
Action: Add "Detection Lead Time" metric to evaluation section — time between warning issuance and hazard encounter. Also add "Distance Estimation Error" as secondary metric.
Added to proposal: Section 5.1 — new metric definitions.

Re: RP Research
Action: Review additional literature on RP mobility challenges (Timmis et al., 2017; Sayed et al., 2019). Reach out to Foundation Fighting Blindness forums for user perspectives. Note: May not get direct RP user for class timeline, but will document research sources.
Added to proposal: Enhanced Section 1.2 with RP-specific constraints.

Re: Scene Description
Action: Good idea for future work! For MVP, focus on autonomous hazard warnings. User-initiated queries require different interaction model (button to trigger query). Note as possible extension in Section 2.1.
Added to proposal: Brief mention in "Possible Extensions" section.

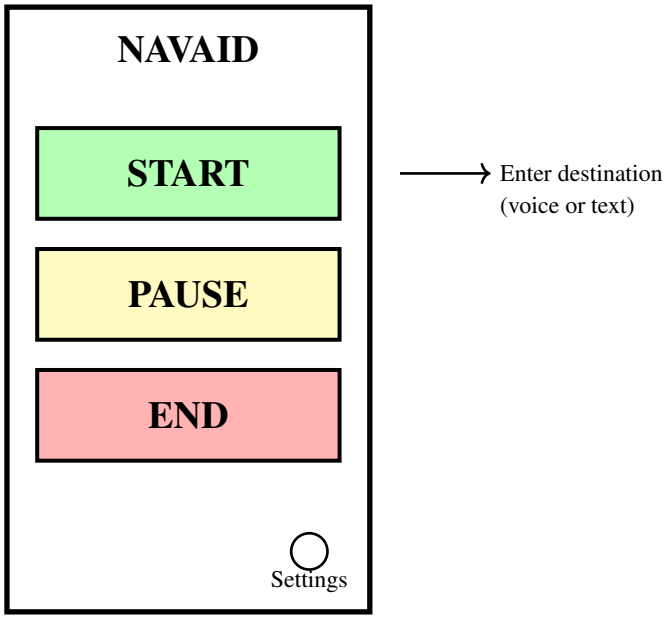
Updated Evaluation Metrics (Added):

- **Detection Lead Time:** Time (seconds) between warning issuance and when user would encounter hazard at current walking speed. Target: 2-4 seconds.
- **Distance Estimation Error:** Absolute error between MLLM-estimated distance and ground-truth distance (measured with depth camera or manual measurement during controlled tests). Target: < 0.5m error for obstacles within 5m.

D. LOW-FIDELITY PROTOTYPE & USER EXPERIENCE DESIGN

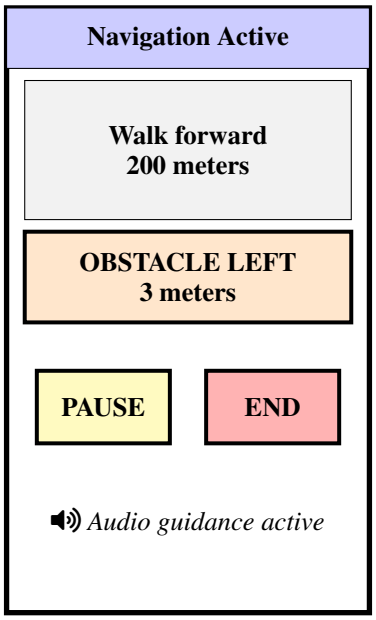
The NAVAID interface is designed around simplicity and audio-first interaction. The main screen features a simple, three-button interface (Start, Pause, End) with large touch targets suitable for users with limited visual acuity. All interactions prioritize audio feedback with minimal visual dependence. The system delivers short, directional hazard warnings via text-to-speech (e.g., "obstacle left, 3 meters") rather than verbose descriptions. Optional haptic patterns provide redundant alerts for lateral collision risks, particularly useful in noisy environments or as a user preference. Personalization settings allow users to configure warning verbosity, lead time thresholds, haptic preferences, and device carriage position to match their individual mobility profile.

Figure 1: Home Screen



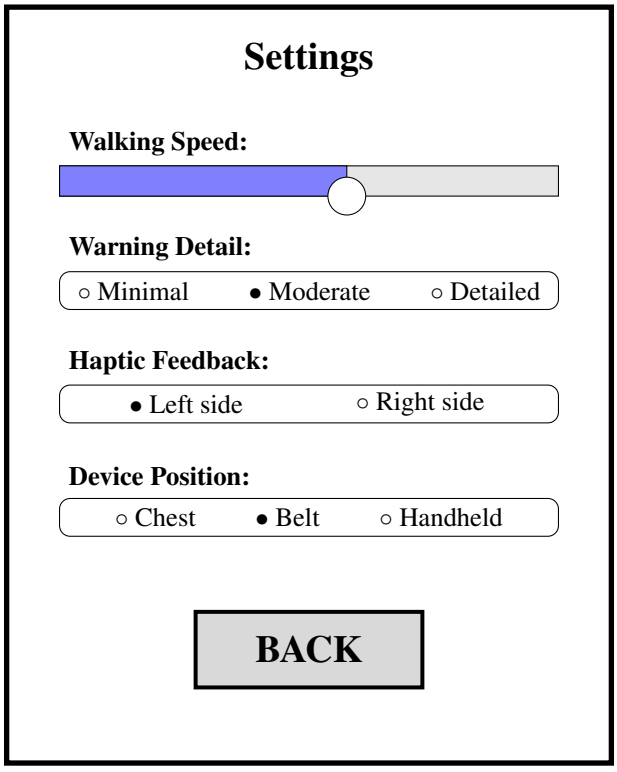
Main screen features three large, high-contrast buttons for primary actions. Pressing START prompts for destination input via voice or text, then begins audio-guided navigation. All actions confirmed via audio feedback. Settings accessible but not prominent.

Figure 2: Active Navigation Screen



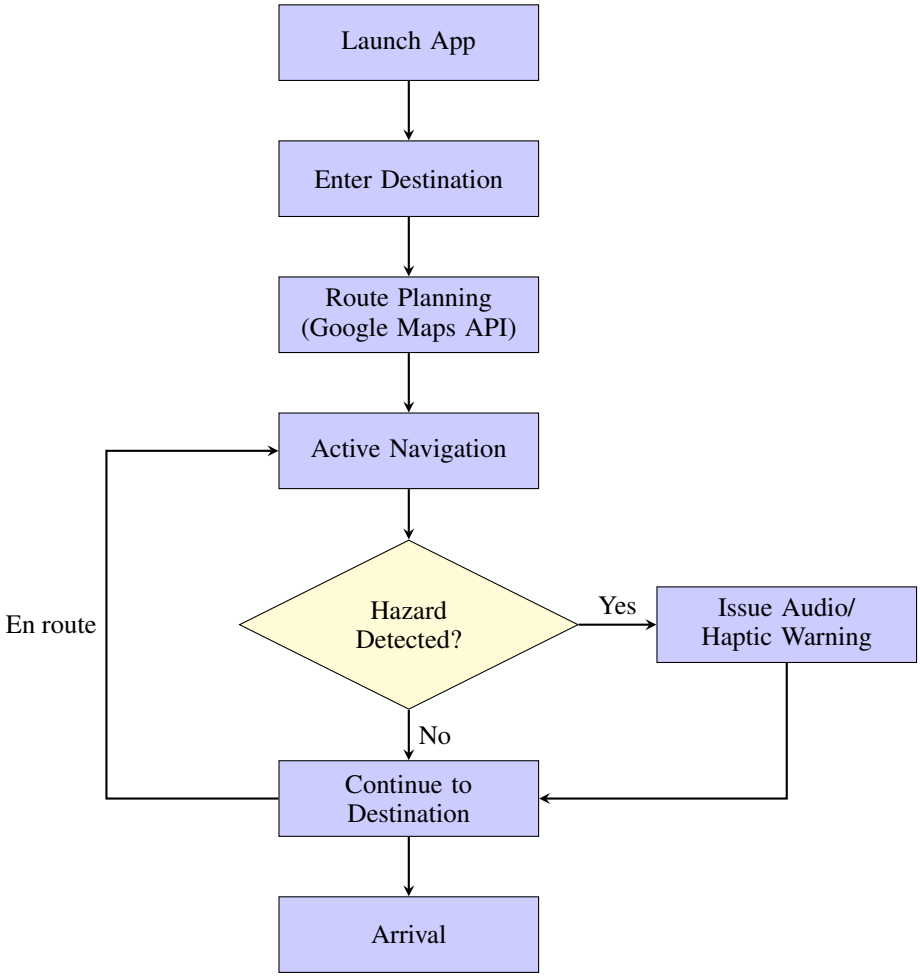
During navigation, screen shows current macro-instruction (from Google Maps API) and hazard warnings when detected. Orange warning box appears only when hazards are present. Primary interaction remains audio; visual display is supplementary.

Figure 3: Settings / Personalization



Settings screen allows personalization of warning timing, verbosity, haptic preferences, and device carriage. All controls use large touch targets and provide audio confirmation of changes.

Figure 4: User Flow Diagram



User flow shows main navigation loop. System periodically captures images (every N seconds based on walking speed), sends to Core AI for hazard detection, and issues warnings when hazards meet personalization thresholds. Navigation continues until destination reached.

Human-AI Interaction Model:

Our system implements a *complementary partnership* where AI handles rapid perceptual processing while humans provide route decisions and contextual judgment:

1. **AI: Perceive & Alert** — System continuously scans for collision-critical hazards (lateral obstacles, curb drops, narrow gaps) and issues concise warnings

2. **Human: Decide & Act** — User interprets warning in context and chooses corrective action (slow, steer, stop)

3. **AI: Adapt** — System adjusts sensitivity based on user profile (residual field, walking speed, environment)

Example Interaction Sequence:

User: Starts navigation to destination

System: ”Route ready. 800 meters, 3 turns.”

User: Begins walking

System: [Detects cyclist approaching from left]

System: ”Bicycle left, 4 seconds” + haptic pulse

User: Slows pace, shifts right

System: ”Continue 150 meters, turn right”

User: Continues walking

This interaction pattern repeats throughout navigation, with the system providing macro-turn instructions from Google Maps and local hazard warnings from the Core AI perception module.

Key Design Decisions:

- **Conciseness over completeness:** Warnings use 2-5 words following Wayfindr principles for audio navigation
- **High recall for safety:** Prioritize detection of collision-critical hazards; tolerate higher false-positive rate over missed hazards
- **Redundant modalities:** Haptic provides backup when audio is masked by ambient noise or user preference
- **Personalized timing:** Lead time calculated from user’s walking speed and hazard urgency (2-4 second window)