CptS 484: Software Requirements

WRS Evolution

Requirements Elicitation

10/12/2025

Revision History

Date	Version	Changes	Editor
09/27/25	0.0.1-21SEP25	Initial Draft	Julian H., Chandler G.
10/12/25	0.0.2-12OCT25	User manual + specifications, editing	Matthew H.

[1] Introduction

1.1.Purpose

What is planned to be built for an application is such that visually impaired users are able to transport themselves from one room inside a building to another with little assistance. The current user base needs assistance from another person, a guide dog, and even a cane for transportation. This may be out of reach to some of that user base, costly, or even time-consuming, causing tardiness. Another key factor is the possibility of getting lost, often causing stress and fear for our main priority user base. Creating an environment in which visually impaired users can traverse buildings in a similar manner to those who are not impaired is the basis of this application.

1.2.Scope

The software will be developed on a 21st-century modern mobile device that has the ability to connect to other IOT devices, contains a good-quality camera, and has the ability to access its GPS, as well as a microphone and a speaker. Each one of these will be used for different features on the device and is very important for the user experience. Since we will have multiple users, including ones who are visually impaired, it will be imperative that we have a simple UI interaction, which includes a way for our users to communicate with the device to activate features. The IoT devices will be used for marking points of each destination, or more technically speaking, starting and ending nodes to a graph, which can include helping with waypoints. We will be using a good-quality camera by using an LLM that can detect objects in real time to prevent accidents. This will also be a good way for us to predict where the user is more accurately with the GPS system for better navigation. We will also be providing owners of buildings to securely upload their building blueprints to provide better data analysis for such predictions.

We have determined that the scope of this project is fairly large, but it is important for many of these features to exist to provide a way for visually impaired people to traverse buildings with little to no help. The main focuses of our features will be the following:

- User Interface interactions will be simplified, including vocal interactions for actions.
- Give close to precise, accurate, and the fastest directions to actual destinations.
- Prevent accidents and provide directions to lost users.
- Manage building data and find users when asking for help
- Learn frequently visited destinations and allow for saving directions/routes.

1.3.Objectives and Success Criteria

The software must contain the following to successfully pass the validation stages:

- Application of user vocal communication commands to action
- Simplistic user interface
- Usable by both impaired and non-impaired users

- Real-time directing
- Real-time object detection
- Uploading of building data
- Different user account types (e.g., caretaker, building owner, main priority user, etc.)
- Emergency options for lost conditions
- Provide all possible paths
- Finds the fastest path from location to destination
- Utilize habitual patterns for saved paths
- Utilize GPS and IoT for better navigation and experience

1.4.Definitions, Acronyms, and Abbreviations

Term	Definition
IOT (Internet Of Things)	Refers to a network of physical devices, vehicles, appliances, and other physical objects that are embedded with sensors, software, and network connectivity, allowing them to collect and share data.
GPS (Global Positioning System)	An accurate worldwide navigational and surveying facility based on the reception of signals from an array of orbiting satellites.

1.5.Overview

Section one is to provide a simple introduction of the application to explain the why and how, including definitions that may stand out as unknown or unclear. Moving forward, the next sections will dive deeper into the applications in more detail. In section 2, this is the start of figuring out the preliminary domain knowledge, functional requirements, and non-functional requirements, where any specifications or further details are elicited. Moving onto section 3, this is a deeper analysis of each of the previous preliminaries to parse out ambiguity, impossibility, or even unobtainability and create different possible options to fix them. Using the new descriptions we can start on the [W]ritten, [R]equirments, and [S]pecifications, which is section 4, where the problems and goals are decided. These are used to create improved versions of the preliminary definitions so that they can be used to identify all of the stakeholders and create the final Functional, Non-Functional, and Specifications for the application.

For sections 5 and 6, these are putting what has been accumulated from the functional, non-functional, and specification to create prototypes and mockups. The prototypes will be descriptions of features with possible graphics to show logical flows to give a low-level look at that feature. This may include a guide to understand the usage as a type of user manual. The mockups will give a first look at the possible designs of the User Interface, including ties in with the prototypes. These will help with identifying anything designed so that it should associate with the current requirements and specification, and does not go outside them.

[2] Preliminary Definition

2.1. Preliminary Domain

PD_ID	Preliminary Domain Description
PD1	The application will be a smartphone-based indoor navigation system designed for blind and visually impaired users to navigate safely between common indoor destinations (e.g., classrooms, restrooms, offices)

2.2. Preliminary Functional Requirements

P FR_ ID	Preliminary FR Description
PFR1	Provide voice-guided navigation to specific indoor locations.
PFR2	Allow users to input destinations via voice commands.
PFR3	Detect obstacles and alert users through audio feedback.
PFR4	Provide route recalculation if the user deviates from the planned path.
PFR5	Allow users to store frequently used destinations (e.g., dorm room, office) and other path data to identify schedules or habits to provide current and next actions.
PFR6	Support user-friendly onboarding and tutorials for new users.
PFR7	Placing an emergency call after detecting falls or lost connections.

2.3. Preliminary Non-Functional Requirements

PNFR_ ID	Preliminary NFR Description
PNFR1	The system shall provide navigation instructions with minimal latency (< 2 seconds response).
PNFR2	The app shall have a simple, intuitive interface requiring no prior technical knowledge.
PNFR3	The app shall ensure accessibility compliance.

PNFR4	The app shall function offline for core navigation features.
PNFR5	The system shall support Android smartphones with standard sensor sets.
PNFR6	The app shall maintain user privacy and not store unnecessary personal data.
PNFR7	The system shall help the user safely navigate indoors, minimizing risks such as collisions and falls.
PNFR8	The system shall guide the user through the fastest available route to the destination.
PNFR9	The system shall be customizable to each user’s preferences, such as instruction frequency, audio volume, and verbosity of guidance.

[3] Issues with the Preliminary Definition Given

3.1.Domain Issues

Domain Issue ID	Domain Issue Description	
DI1	PD_ID	PD1. The domain includes accessibility technology, app development, and assistive tools.
	Issue: The required level of mapping detail for navigation is unclear.	
	Option 1	Use simplified waypoint-based navigation.
	Option 2	Implement detailed floor maps and obstacle detection.
	Choice	**Option 1**
	Rationale	Waypoint navigation is more feasible with limited resources and given deadlines.
Satisfied by		**FR1** (voice navigation) and **FR3** (obstacle alerts).

3.2.Functional Requirements Issues

FR Issue ID	Description	
FRI1	PFR_ID	PFR1
	Issue: Does not specify the frequency of voice guidance nor the conditions of the voice guidance.	
	Option 1	Use the waypoints of the IoTs as markers for when to announce walk distance and corners.
	Option 2	Use estimated times for a user who has walked previously to predict the number of steps.
	Option 3	Use object detection to detect corners and walls using markers to figure out distances to estimate steps and a PTA.
	Choice	Option 1
	Rationale	Using waypoints will be simpler for the scope and can give accurate updates through the course of their walk to their destination
Satisfied by	FR1 (Navigation Core Module), FR3 (Waypoint Navigation System)	

FR Issue ID	Description	
FRI2	PFR_ID	PFR2
	Issue: Doesn't describe the exact commands for specifying a location.	
	Option 1	This can be done by connecting to the IoTs where each one is given a special identifier with its location name automatically.
	Option 2	Users of the app can add the locations to their building so that they can correctly identify them all. (Including the owner of the building)
	Choice	Option 2
	Rationale	Rooms may end up changing names, and users might only know them by those names. Giving a name to

		them automatically will make it difficult for users to remember different identifications for the room.
Satisfied by	FR5 (Voice Controlled Input System)	

FR Issue ID	**Description**	
FRI3	PFR_ID	PFR3
	Issue: Does not provide the type of audio feedback, nor does it describe how an object will be detected.	
	Option 1	Can detect objects using a camera and an LLM, detecting objects in real time. A sound can be played when you get close to an object, and it will beep faster the closer the object appears.
	Option 2	Can detect objects using sound waves by playing a sound on a frequency allowing for distance detection, and when you get close to an object, it will beep faster the closer the object appears.
	Choice	Option 1
	Rationale	Using the camera won't cause any noise, especially in an environment in which silence is necessary in the building.
Satisfied by	FR1 (Navigation Core Module)	

FR Issue ID	**Description**	
FRI4	PFR_ID	PFR4
	Issue: Where should the recalculation take the user?	
	Option 1	The user re-chooses a new path from their current position.
	Option 2	The software recalculates the path to retrace their steps to their original path.

	Option 3	The software recalculates the path to another path from the current position, which may or may not be the same.
	Choice	Option 2
	Rationale	Re-choosing a path during a walk could cause commotion, and re-tracing allows the user to keep their originally selected path.
Satisfied by	FR3 (Waypoint Navigation System)	

FR Issue ID	**Description**	
FRI5	PFR_ID	PFR5
	Issue: What constitutes a habit or a schedule, and is this enforced, as well as possible permanent or temporary blockage?	
	Option 1	Uses path, locations data, time, date, and visiting data to find the most used paths and accident detection counts to figure out the best path to automatically take the user to those destinations after prompting the user.
	Option 2	Use current locations and paths to create a schedule themselves and have the software notify them of such after setup.
	Choice	Option 1
	Rationale	This stops the need for the user to formulate this on their own. The software can auto-detect this and find possible blockage and use the best next action. Otherwise, the user would have to update or know of current problems to update the schedule.
Satisfied by	FR2 (History/Frequent Locations Interface)	

FR Issue ID	**Description**	
FRI6	PFR_ID	PFR6

	Issue: User-friendly onboarding means for all types of users and different types of tutorials for those types as well.	
	Option 1	Make one tutorial that can be used by all of the user base, including the user-friendly interaction on the user interface, to be done like this as well.
	Option 2	Create separate views where some can be used by all users and some only by specific users, including only the tutorial portions that users will use.
	Option 3	Create completely different views for each user and create different tutorials for each one.
	Choice	Option 2
	Rationale	Combining both and adding user interfaces for specific users can allow for easier teachability not only to the user themselves but across each user. Tutorials can be created simply in partitions instead.
Satisfied by	FR4 (Tutorial Module)	

FR Issue ID	**Description**	
FRI7	PFR_ID	PFR7
	Issue: How are emergencies handled?	
	Option 1	Send out a 911 alert to be helped by them immediately.
	Option 2	Sounds out a very loud noise from the device to get the attention of anyone nearby to assist the user.
	Option 3	Send an alert to the closest caretaker or emergency contact of their last known or current location.
	Choice	Option 3
	Rationale	This will get them an exact person they feel comfortable with, and if there are any more serious problems, the one assisting them can call the police themselves.
Satisfied by	FR6 (Fall Detection & Sound Alert System)	

3.3.Non-Functional Requirements(NFR) Issues

NFR Issues ID	Description	
NFRI1	PNFR_ID	PNFR1
	Issue: The system shall provide navigation instructions with minimal latency (< 2 seconds response).	
	Option1	Use faster and lighter processing methods and models.
	Option2	Get GPS and sensor data directly from the phone's built-in services.
	Option3	Save route data ahead of time to reduce waiting.
	Choice	Option1 + Option3.
	Rationale	Lightweight models process faster, and pre-saved routes speed up processing.
Satisfied by	PFR1 (voice-guided navigation), PFR4 (route recalculation), PNFR4 (offline maps).	

NFR Issues ID	Description	
NFRI2	PNFR_ID	PNFR2
	Issue: The app shall have a simple, intuitive interface requiring no prior technical knowledge.	
	Option1	Use voice commands as the main way to control the app.
	Option2	Use short, clear audio instructions that are easy to understand.
	Option3	Design a clean interface with fewer buttons and less clutter.
	Choice	Option1

	Rationale	Voice commands are the easiest and most natural option for blind users.
Satisfied by	PFR2 (voice command input), PFR6 (onboarding/tutorial), PNFR3 (accessibility compliance), PNFR9 (customization).	

NFR Issues ID	**Description**	
NFRI3	PNFR_ID	PNFR3
	Issue: The app shall ensure accessibility compliance.	
	Option1	Follow official accessibility rules (like WCAG, ADA).
	Option2	Connect with built-in phone accessibility tools (TalkBack, VoiceOver).
	Option3	Build a custom accessibility framework.
	Choice	Option 1
	Rationale	Following existing standards is the easiest and ensures compliance.
Satisfied by	PFR1 (voice guidance), PFR2 (voice input), PFR3 (obstacle alerts), PNFR2 (simple UI).	

NFR Issues ID	**Description**	
NFRI4	PNFR_ID	PNFR4
	Issue: The app shall function offline for core navigation features.	
	Option1	Store maps locally on the phone.
	Option2	Let users choose which maps/buildings to download.
	Option3	Sync maps only through a constant internet connection.

	Choice	Option1.
	Rationale	Storing maps locally is simple for guaranteeing offline use.
Satisfied by	PFR1 (navigation), PFR4 (route recalculation), PNFR1 (low-latency), PNFR5 (support on standard smartphones).	

NFR Issues ID	**Description**	
NFRI5	PNFR_ID	PNFR5
	Issue: The system shall support Android smartphones with standard sensor sets.	
	Option1	Support only the most essential sensors, such as GPS and accelerometer.
	Option2	Support a broader set of sensors, including gyroscope.
	Option3	Support all common sensors available on most Android phones.
	Choice	Option1.
	Rationale	Supporting only basic sensors makes the app usable on most devices.
Satisfied by	PFR1 (navigation), PFR3 (obstacle detection), PFR4 (recalculation), PNFR7 (safety), PNFR8 (fastest route).	

NFR Issues ID	**Description**	
NFRI6	PNFR_ID	PNFR6
	Issue: The app shall maintain user privacy and not store unnecessary personal data.	

	Option1	Only keep anonymous usage data.
	Option2	Encrypt anything that must be saved locally.
	Option3	Collect detailed user data for better analysis.
	Choice	Option1.
	Rationale	Anonymous data collection is the simplest and safest way to protect privacy.
Satisfied by	PFR5 (saved destinations), PFR6 (onboarding/tutorial data), PNFR2 (simple UI), PNFR9 (customizable preferences).	

NFR Issues ID	**Description**	
NFRI7	PNFR_ID	PNFR7
	Issue: The system shall help the user safely navigate indoors, minimizing risks such as collisions and falls.	
	Option1	Use the phone's camera alongside LLMs for simple object detection and audio narration
	Option2	Add vibration alerts to warn of nearby obstacles.
	Option3	Install extra hardware sensors in buildings.
	Choice	Option 1 + Option 2
	Rationale	Vibration alerts work more reliably in noisy or crowded environments.
Satisfied by	PFR3 (obstacle detection with alerts), PFR1 (navigation instructions), PNFR5 (sensor support).	

NFR Issues ID	**Description**	
NFRI8	PNFR_ID	PNFR8

	Issue: The system shall guide the user through the fastest available route to the destination.	
	Option1	Use simple pathfinding algorithms.
	Option2	Recalculate routes when the user goes off track.
	Option3	Depend on external servers to calculate routes.
	Choice	Option1.
	Rationale	A simple pathfinding algorithm is easy to add and does not depend on the internet.
Satisfied by	PFR1 (voice navigation), PFR4 (recalculation), PNFR1 (low latency), PNFR5 (sensor support).	

NFR Issues ID	**Description**	
NFRI9	PNFR_ID	PNFR9
	Issue: The system shall be customizable to each user's preferences, such as instruction frequency, audio volume, and verbosity of guidance.	
	Option1	Add a simple settings panel where users can manually adjust preferences
	Option2	Use adaptive AI that automatically adjusts settings based on user behavior.
	Option3	Provide preset modes such as "short," "detailed," or "verbose" for quick setup.
	Choice	Option1.
	Rationale	A basic settings menu is the easiest way to give users control without complexity.
Satisfied by	PFR2 (voice input), PFR5 (saved destinations), PFR6 (onboarding/tutorial), PNFR2 (UI simplicity).	

[4] WRS

4.1.[W]ritten

4.1.1. Problem

Problem ID	Problem Description	Corresponding Goals
P1	Visually impaired users currently struggle to navigate indoor spaces independently, relying mainly on others or navigation tools such as canes and guide dogs.	G1, G2
P2	Indoor environments often lack consistent mapping data, making it unclear how the application will generate accurate routes.	G3
P3	Smartphone sensors may provide inconsistent or inaccurate data indoors.	G3, G4
P4	Users need clear and accessible input methods, but voice command accuracy can vary depending on background noise or accents.	G5
P5	Obstacle detection must provide timely alerts to avoid collisions, but latency or weak detection models could put users at risk.	G6
P6	Users may become disoriented if they deviate from the planned route, requiring fast recalculation for an updated route.	G7
P7	Emergency situations such as falls, disconnections, or becoming lost must be handled quickly, but the best option for this is unclear.	G8
P8	The app must be simple enough for new users to learn without difficulty.	G9

P9	Different stakeholders have different expectations; the system must balance these varying needs.	G10
P10	User privacy and data security must be preserved when handling sensitive information like building maps and user locations.	G10

4.1.2. Goals

Goal ID	Goal Description	Backward Traceability	Forward Traceability
G1	Provide reliable indoor navigation so visually impaired users can safely move between destinations independently.	P1	IFRO1
G2	Reduce reliance on external aids such as canes by delivering precise navigation support.	P1	IFRO1
G3	Generate accurate indoor routes by leveraging building data, IoT devices, and stored waypoints.	P2, P3	IFRO3
G4	Improve sensor reliability by combining data inputs to achieve consistent indoor navigation.	P3	IFRO1
G5	Provide strong voice input and output sensors to handle external noise and accents.	P4	IFRO5
G6	Implement real-time obstacle detection and fast alerts through audio or vibration feedback to keep users safe.	P5	IFRO1
G7	Provide responsive route recalculation and safe rerouting when users deviate from the planned route.	P6	IFRO3

G8	Integrate emergency handling features such as caretaker alerts, fall detection, or location sharing.	P7	IFRO6
G9	Create a simple onboarding process and interface that is easily handled by users with limited technical experience.	P8	IFRO4
G10	Balance the needs of multiple stakeholders while ensuring strong data security and user privacy.	P9, P10	NA

4.1.3. Improved Understanding of Domain, Stakeholders, Functional, and Non-Functional Objectives

4.1.3.1. Improved Domain

Improved Domain ID	Improved Domain Description
ID1	Theia, available only on smartphones, is used to navigate users safely with the help of user-chosen assistants within a closed building to a destination. All necessary tools and data have been set up and are managed by the Owner or Organization that owns the building.

4.1.3.2. Stakeholders

There are many possible stakeholders that are direct users of the application, even those who may not directly utilize the application:

- Navigation User
  - Uses the app to navigate through the building to reach destination
    - Visually impaired
    - Any person requested use of the application
- Assistants
  - Helping those who are using the application when an emergency happens
    - Advocator
    - Family Members
- Administration
  - Manages the application by using a portal to see all non-privileged data of buildings, users, etc.
    - Client

- Owner of the application
    - Employees of the application

- ❖ Building Owner
  - ➢ Managing the building data for the application usage of all users
    - Organization
    - Personal Home
- ❖ Onlooker
  - ➢ Interaction through seeing a user use the application
    - Building Staff
    - Building Visitors
- ❖ Medical Professional
  - ➢ Helping a user who has been injured or viewing application emergency records
    - Nurse
    - Doctor
    - Paramedic

4.1.3.3. Improved Functional Objectives

Based on the above information and our goals, the functional objectives of Theia are:

Improved FR Objective ID	Objective Description	Alleviates Problems	Achieves Goals
IFRO1	Implement audio navigation using real-time obstacle detection.	P1	G1, G4
IFRO2	Implement a history/frequent locations interface.	P1, P6	G1, G7
IFRO3	Create a waypoint navigation system using IoT beacons	P2, P3	G3, G4
IFRO4	Create a tutorial for visually impaired people so they can understand more of the accessibility features provided.	P8	G9
IFRO5	Implement voice-controlled input to allow for hands-free.	P4	G5
IFRO6	Detect when the device has fallen and make a loud sound to allow visually impaired users to find their device again easily.	P7	G8

4.1.3.4. Improved Non-Functional Objectives

Improved NFR Objective ID	Objective Description	Alleviates Problem	Achieves Goal
INFRO1	Ensure navigation instructions are delivered quickly with low latency	P5	G6
INFRO2	Provide a simple, intuitive interface usable without technical knowledge.	P8	G9
INFRO3	Guarantee accessibility compliance.	P1	G1
INFRO4	Enable offline functionality for core navigation by storing maps locally.	P6	G7
INFRO5	Support Android smartphones with basic sensors (GPS, accelerometer).	P3	G4
INFRO6	Protect user privacy by limiting data collection to anonymous usage.	P10	G10
INFRO7	Enhance user safety with multimodal feedback (audio + vibration alerts).	P5	G6
INFRO8	Always calculate and guide via the fastest safe route.	P2	G3
INFRO9	Allow customizable preferences (instruction frequency, verbosity, audio volume).	P9	G10

4.2.[R]equirements [S]pecification

4.2.1. Functional Requirements

FR ID	Satisfies Functional Requirement Issue	Satisfies Objectives	Satisfied by prototype feature

FR1	FRI1, FRI3	IFRO1	Navigation Core Module
FR2	FRI5	IFRO2	History/Frequent Locations Interface
FR3	FRI1, FRI4	IFRO3	Waypoint Navigation System
FR4	FRI6	IFRO4	Tutorial Module
FR5	FRI2	IFRO5	Voice Controlled Input System
FR6	FRI7	IFRO6	Fall Detection & Sound Alert System

4.2.2. Non-Functional Requirements

NFR ID	**Nonfunctional Requirement 1**
NFR1	The system shall provide navigation instructions with minimal latency (<2 seconds response).
Operationalized Functional Requirements	FR1, FR3
Satisfies Nonfunctional Requirement Issue	NFRI1
Satisfies Non-functional Objective	INFRO1
Constrains	Processing models must remain lightweight and pre-save route data for fast guidance.
Satisfied by prototype feature	Real-time voice guidance, IoT waypoint navigation

NFR ID	**Nonfunctional Requirement 2**
NFR2	The app shall have a simple, intuitive interface requiring no prior technical knowledge.
Operationalized Functional Requirements	FR4, FR5
Satisfies Nonfunctional Requirement Issue	NFRI2
Satisfies Non-functional Objective	INFRO2
Constrains	Limits the number of on-screen buttons and relies on voice

	input for accessibility.
Satisfied by prototype feature	Onboarding tutorial, minimal UI layout

NFR ID	**Nonfunctional Requirement 3**
NFR3	The app shall ensure accessibility compliance.
Operationalized Functional Requirements	FR1, FR5
Satisfies Nonfunctional Requirement Issue	NFRI3
Satisfies Non-functional Objective	INFRO3
Constrains	Must follow WCAG and ADA guidelines for text, audio, and interaction design.
Satisfied by prototype feature	Voice input, screen-reader compatible navigation

NFR ID	**Nonfunctional Requirement 4**
NFR4	The app shall function offline for core navigation features.
Operationalized Functional Requirements	FR1, FR3
Satisfies Nonfunctional Requirement Issue	NFRI4
Satisfies Non-functional Objective	INFRO4
Constrains	Requires local map storage and optional download management.
Satisfied by prototype feature	Cached indoor map for offline use.

NFR ID	**Nonfunctional Requirement 5**
NFR5	The app shall support Android smartphones with standard sensor sets.
Operationalized Functional Requirements	FR1, FR3
Satisfies Nonfunctional Requirement Issue	NFRI5
Satisfies Non-functional Objective	INFRO5
Constrains	Limits hardware dependency to common Android sensors for broad compatibility.
Satisfied by prototype feature	Mobile device sensor integration

NFR ID	**Nonfunctional Requirement 6**
NFR6	The app shall maintain user privacy and not store unnecessary personal data.
Operationalized Functional Requirements	FR5, FR6
Satisfies Nonfunctional Requirement Issue	NFRI6
Satisfies Non-functional Objective	INFRO6
Constrains	No PII stored; encrypted logs only.
Satisfied by prototype feature	Anonymous usage session logging

NFR ID	**Nonfunctional Requirement 7**
NFR7	The system shall help the user safely navigate indoors, minimizing risks such as collisions and falls.
Operationalized Functional Requirements	FR1, FR3
Satisfies Nonfunctional Requirement Issue	NFRI7
Satisfies Non-functional Objective	INFRO7
Constrains	Requires continuous sensor input and multimodal alerts.
Satisfied by prototype feature	Audio and vibration obstacle alert system

NFR ID	**Nonfunctional Requirement 8**
NFR8	The system shall guide the user through the fastest available route to the destination.
Operationalized Functional Requirements	FR1, FR3
Satisfies Nonfunctional Requirement Issue	NFRI8
Satisfies Non-functional Objective	INFRO8
Constrains	Routing must prioritize speed and safety using lightweight algorithms.
Satisfied by prototype feature	Optimized pathfinding and rerouting module

NFR ID	**Nonfunctional Requirement 9**
NFR9	The system shall be customizable to each user’s preferences, such as instruction frequency, audio volume, and verbosity of

	guidance.
Operationalized Functional Requirements	FR4, FR5
Satisfies Nonfunctional Requirement Issue	NFRI9
Satisfies Non-functional Objective	INFRO9
Constrains	Preferences must persist locally and update immediately on change.
Satisfied by prototype feature	User settings panel for audio and feedback customization

4.2.3. Specifications

Functional Specification ID	Functional Requirement
FS1	Provide quick, real-time, audio-guided navigation with offline functionality by using IoT beacons as route waypoints. This ensures that visually impaired users will receive accurate directions without the need for an Internet connection.
Satisfies Functional Requirement	FR1, FR3
Satisfies Objectives	IFRO1, IFRO3, INFRO1, INFRO4, INFRO8
Satisfied by prototype feature	Real-time voice guidance, IoT waypoint navigation, audio and vibration alert system

Functional Specification ID	Functional Requirement
FS2	Enables audio interaction with the app using voice commands, which reduces the need for physical screen interaction for visually impaired users.
Satisfies Functional Requirement	FR5
Satisfies Objectives	IFRO5, INFRO2, INFRO3, INFRO5, INFRO9
Satisfied by prototype feature	Voice input

Functional Specification ID	Functional Requirement

FS3	Theia can suggest routes faster by storing a user’s frequent destinations and viewing navigation history. This will allow the app to learn from the user’s habitual routes to certain destinations at certain times.
Satisfies Functional Requirement	FR2
Satisfies Objectives	IFRO2, INFRO1, INFRO8, INFRO9
Satisfied by prototype feature	Navigation history, IoT waypoint navigation

Functional Specification ID	Functional Requirement
FS4	Provides users a tutorial to teach users interactions with the app’s interface and voice commands for different kinds of users (e.g., caretakers, visually impaired users, etc.) for full functionality of the app.
Satisfies Functional Requirement	FR4
Satisfies Objectives	IFRO4, IFRO5, INFRO2, INFRO3, INFRO5, INFRO7
Satisfied by prototype feature	Onboarding tutorial, minimal UI layout

Functional Specification ID	Functional Requirement
FS5	Detects when the device has fallen, indicating that the user may need assistance (e.g., dropped device or fell). Triggers sound alerts if the user needs to find their device again, and contacts the appropriate form of assistance (e.g., caretaker or emergency services) for the situation.
Satisfies Functional Requirement	FR6
Satisfies Objectives	IFRO5, IFRO6, INFRO3, INFRO5, INFRO6, INFRO7
Satisfied by prototype feature	Mobile device sensor integration, fall detection and alert system

[5] Preliminary Prototype & User Manual

We decided to interpret our preliminary prototype using diagrams and a detailed description of the diagram so that the interpretation does not get lost. We also decided to use a User Manual to provide a more in-depth use case or process they can take for each user of the Theia system.

5.1. Data Flow Diagram

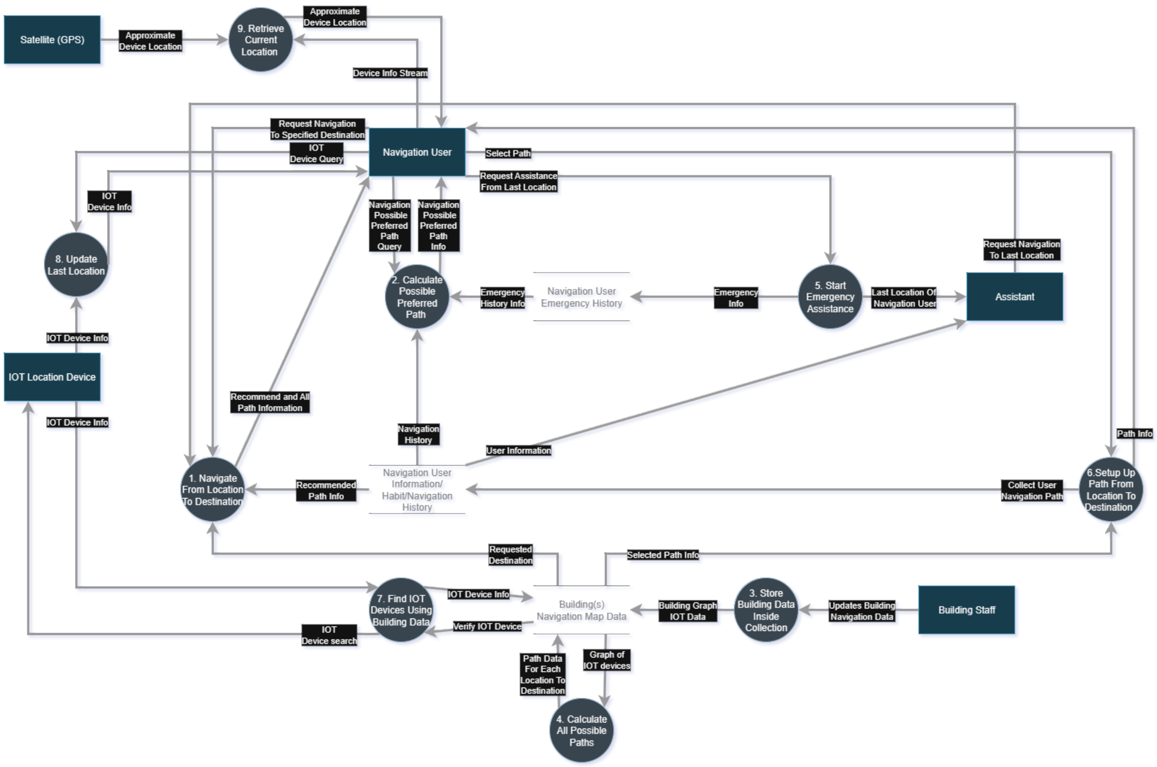
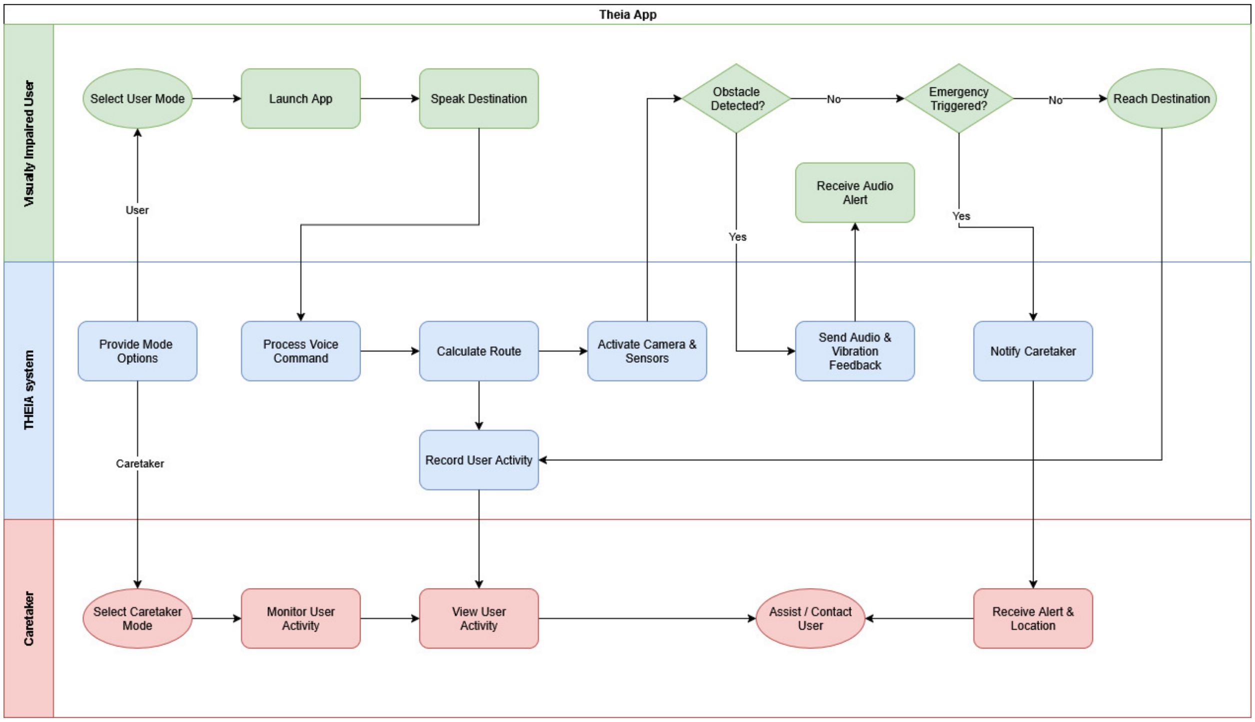


Figure 5.1-1
Data Flow Diagram Theia Navigation System

To further explain what is going on in Figure 5.1-1, it details the navigation system within the Theia app at the data storage level and how that data is transferred throughout the system from one entity to another through processes and accessing the data storage with them.

Most of the data that is used within the system that is stored is the users' Emergency Information and their Habits/History of their paths they have taken. The previous data stored along with the Building Data are used to develop processes that can use this data. Overall, every process is used to send data from one entity to another, providing data for navigating a building regardless of the user interface experience.

5.2. Swimlane Diagram



5.3. User Manual

Please see [this document.]

[6] Prototype Interface Mock-ups

We have completed the initial Theia app mock-ups of the home page, user mode, and caretaker mode. The mock-ups were created using Figma. To view, click the link to the [Figma Design File].

[7] References

TBD; we currently have no references for this document.