Software Requirements Specification for Audio360

Team #6, Six Sense
Omar Alam
Sathurshan Arulmohan
Nirmal Chaudhari
Kalp Shah
Jay Sharma

October 9, 2025

Contents

Revision History

Date	Version	Notes
2025-10-06	1.0	Initial Write-up

1 Goal

1.1 G.1 Context and overall objective

With around 4 million Canadians affected by hearing loss [1], there is a significant need for assistive technologies that can improve situational awareness and safety. Many safety cues and general sound alerts such as the sound of a car approaching, a kettle whistling, or a phone ringing may be missed, leading to increased risk of injury and miscommunication.

Many existing solutions focus on speech transcription, but lack the ability to provide directional information about sound sources or classify non-speech sounds. This project aims to address this gap by developing an assistive device that provides real-time visual indications of sound source locations and classifications.

The objective of this project is to develop an assistive device that aids individuals who are deaf or hard of hearing by providing real-time visual indications of sound source locations and classifications (ex. 'car on your left').

Some of the high-level goals of the project are:

Goal 1. Capture real-time audio data from a microphone array with synchronized sampling to enable accurate situational analysis of sound sources.

Goal 2. Analyze captured audio to determine the direction of arrival (DoA) of sound sources with minimal error and with minimal latency nearing real-time.

Goal 3. Analyze captured audio to classify the sound sources with their English label (ex. 'car', 'phone', 'kettle', 'alarm', 'speech').

Goal 4. Display audio classification and transcription on smart glasses in real-time without obstructing the user's field of view.

Goal 5. Provide a user-friendly interaction with the smart glasses, allowing the user to easily set up, use, and understand visual indicators.

Goal 6. Ensure that the system is comfortable to wear for extended periods of time, with minimal discomfort or fatigue.

1.2 G.2 Current situation

Currently, individuals who are deaf or hard of hearing face significant challenges in maintaining situational awareness due to missed audio cues. Existing assistive technologies address some aspects of this problem, but leave critical gaps:

- **Smart glasses with transcription capabilities:** Some devices can listen to live human audio and transcribe it to text (multilingual) in real-time, displaying the transcript on a smartphone display. However, these solutions focus solely on speech transcription and do not provide directional information about sound sources or classify non-speech sounds.

- **Hearing aids:** Traditional hearing aids amplify ambient sounds to improve awareness of audio sources at various volumes [2]. While this helps individuals with partial hearing loss, it does not assist those who are profoundly deaf, nor does it provide visual cues about sound direction or classification.
- **Notification systems:** Some home automation systems can send visual alerts (e.g., flashing lights) when specific sounds are detected, such as doorbells or smoke alarms. However, these systems are limited to fixed locations and predetermined sound types, lacking portability and real-time directional awareness.

The current solutions fail to address the critical need for real-time, portable, directional awareness of environmental sounds, leaving individuals vulnerable to missing important safety cues such as approaching vehicles, warning beeps from machinery, or emergency alerts.

1.3 G.3 Expected benefits

The proposed system will deliver significant improvements to the daily lives and safety of individuals who are deaf or hard of hearing:

- **Real-time spatial awareness:** Enable identification of sound source locations on a 2D plane in real-time, allowing users to quickly orient themselves toward important sounds such as someone calling their name, an approaching vehicle, or an emergency alarm.
- **Enhanced safety:** Reduce the risk of injury by alerting users to critical safety cues that are typically communicated through sound, such as warning beeps from fork-lifts, tea kettles whistling, car engines or emergency sirens (from emergency vehicles) approaching from behind.
- **Improved situational awareness:** Provide continuous awareness of the acoustic environment without requiring the user to constantly scan their surroundings, reducing cognitive load and enabling more natural interactions with their environment.
- **Sound classification:** Differentiate between various types of sounds (e.g., speech, alarms, vehicles, household appliances) to help users prioritize their attention and responses appropriately.
- **Reduced frustration and miscommunication:** Minimize instances of missed phone calls, doorbell rings, or verbal attempts to gain the user's attention, leading to smoother social interactions and reduced social isolation.
- **Portable and wearable solution:** Unlike fixed home automation systems, the smart glasses form factor provides continuous protection and awareness regardless of location, whether at home, work, or in public spaces.

- **Independence and confidence:** Empower users to navigate their environment more independently without relying on others to alert them to important sounds, fostering greater autonomy in daily activities.

1.4 G.4 Functionality overview

The system will provide the following principal functions:

- **Real-time audio capture:** Continuously capture audio signals from a synchronized microphone array mounted on smart glasses, ensuring precise temporal alignment for accurate spatial analysis.

- **Direction of arrival (DoA) estimation:** Process captured audio to determine the angular direction of sound sources on a 2D plane relative to the user's position, with a target accuracy of $\pm 45°$ for single sound sources.

- **Sound source classification:** Analyze audio characteristics to classify detected sounds into meaningful categories (e.g., speech, vehicle sounds, alarms, household appliances) using audio fingerprinting techniques, with a target accuracy of at least 90%.

- **Visual feedback generation:** Generate intuitive visual representations of detected sound sources, including their direction and classification, displayed on the smart glasses interface with minimal latency (≤ 1 second).

- **Multi-source handling:** Detect and track multiple simultaneous sound sources when feasible, prioritizing the most relevant or critical sounds based on classification and proximity.

- **Real-time processing:** Execute all signal processing, direction estimation and classification algorithms real-time, with consistent performance and low latency.

- **Noise cancellation or audio filtering:** The system will modify or filter the actual sounds in the environment based on direction of arrival in order to improve directional hearing. This would help improve the quality of the transcriptions provided by the system.

1.5 G.5 High-level usage scenarios

The following scenarios illustrate fundamental usage paths through the system:

1.5.1 Scenario 1: Pedestrian crossing detection

A user is walking in an urban environment and approaches a street intersection. As they prepare to cross, a car approaches from their left side. The system detects the engine sound, estimates its direction (e.g., 90° to the left), classifies it as a vehicle, and displays a visual indicator on the smart glasses showing the direction and classification. The user recognizes the alert and waits for the vehicle to pass before crossing safely.

1.5.2 Scenario 2: Kitchen safety alert

A user is cooking in their kitchen when a tea kettle on the stove begins to whistle. The system captures the high-pitched sound through the microphone array, determines that it is coming from behind and to the right (e.g., 135°), classifies it as a kettle or alarm sound, and displays a directional indicator. The user turns toward the alert and removes the kettle from heat, preventing a potential hazard.

1.5.3 Scenario 3: Social interaction

A user is in a crowded room when someone calls their name from across the space. The system detects the speech sound, estimates the direction (e.g., 30° to the right), classifies it as speech or a human voice, and displays the information on the glasses. The user turns in the indicated direction to make eye contact and engage in conversation, reducing social friction and missed interactions.

1.5.4 Scenario 4: Workplace awareness

A user is working in an industrial setting when a forklift begins reversing nearby, emitting a warning beep. The system detects the beeping pattern, determines its direction (e.g., directly behind at 180°), classifies it as a warning signal, and alerts the user with a prominent visual indicator. The user steps aside to maintain a safe distance from the moving equipment.

1.6 G.6 Limitations and exclusions

The following aspects are explicitly outside the scope of this project:

- **Autonomous danger assessment:** The system will not independently evaluate whether a detected sound represents an immediate danger or automatically alert the user of hazardous situations. It will present directional and classification information, leaving interpretation and response decisions to the user.

- **Augmented reality overlay:** The system will not provide full augmented reality capabilities with spatial overlays showing sound locations directly mapped onto the user's field of view. Visual feedback will be presented through a simpler display interface on the smart glasses.

- **User response monitoring:** The system will not track whether the user has noticed, acknowledged, or responded to presented alerts. There is no feedback loop to ensure user reaction or to escalate notifications.

- **Multilingual speech transcription:** Audio transcription functionality, if implemented, will be limited to English only. Support for other languages is not included in the current scope.
- **3D spatial localization:** Direction estimation will be constrained to a 2D horizontal plane around the user. Elevation angle determination (above or below the user's head level) is excluded from the core functionality.
- **Sound source distance estimation:** While direction will be provided, the system will not attempt to estimate the absolute distance to sound sources.
- **Continuous recording or data storage:** The system will not record or store audio data beyond what is necessary for real-time processing. No historical logs of detected sounds will be maintained.
- **Network connectivity:** All processing will occur locally on the embedded hardware. The system will not require internet connectivity or cloud-based services for core functionality.

1.7 G.7 Stakeholders and requirements sources

1.7.1 Primary stakeholders

- **Individuals who are deaf or hard of hearing:** The primary end-users of the system, who will directly benefit from improved situational awareness and safety. This group is quite large in population, with approximately 4 million people who experience hearing loss in Canada alone (1 in 10).

1.7.2 Secondary stakeholders

- **Family members and caregivers:** Individuals who support people with hearing loss and will benefit from improved communication and reduced safety concerns.
- **Employers and workplace safety officers:** Organizations that employ individuals with hearing loss and are responsible for maintaining safe working environments.
- **Accessibility advocates and organizations:** Groups focused on improving quality of life and independence for individuals with disabilities.
- **Healthcare providers and audiologists:** Professionals who may recommend or integrate such assistive technologies into patient care plans.
- **Future developers and researchers:** The broader engineering and scientific community who may build upon this work or apply similar techniques to related problems.

1.7.3 Requirements sources

- **Academic literature:** Research on hearing loss impact, assistive technologies, direction of arrival algorithms, and audio classification techniques.
- **Domain experts:** Consultation with domain experts, such as Dr. Mohrenschildt, for technical feasibility and requirements validation.
- **Existing assistive technologies:** Analysis of current solutions such as hearing aids, transcription glasses, and home alert systems to identify gaps and opportunities.
- **Hardware and software documentation:** Technical specifications for microcontroller, source code libraries, smart glasses hardware, and microphone array components.
- **Standards and best practices:** IEEE standards for embedded systems, accessibility guidelines, and real-time system design principles.
- **Proof of concept testing:** Empirical results from prototyping and laboratory testing to validate technical approaches and refine requirements.

2 Environment

2.1 E.1 Glossary

- **Microphone Array:** A collection of microphones that are synchronized to capture audio from the environment to create a single multi-channel audio signal.
- **Microcontroller:** Compact embedded hardware system with a CPU, memory, input/output peripherals, designed to process real-time data directly on the device.
- **Microphone:** Input sensor to a device that converts audio soundwaves to digital bytestream.
- **Normal Operating Condition:** The state in which the system, including hardware and softwre, operate within expected environmental and usage parameters. This includes audio input sampled at the correct sample rate, input audio signals are within the human audible range (20 Hz – 20 kHz) [3], and audio input levels below 90 dB.
- **Spectral Leakage:** The spreading of a signal's energy across multiple frequency bins in a frequency spectrum due to finite time windowing. Essentially, noise that is caused by sharp signal cutting when processing audio signals as windows.

2.2 E.2 Components

[List of elements of the environment that may affect or be affected by the system and project. Includes other systems to which the system must be interfaced. —SS]

2.3 E.3 Constraints

[Obligations and limits imposed on the project and system by the environment. —SS]

2.4 E.4 Assumptions

[Properties of the environment that may be assumed, with the goal of facilitating the project and simplifying the system. —SS]

2.5 E.5 Effects

[Elements and properties of the environment that the system will affect. —SS]

2.6 E.6 Invariants

[Properties of the environment that the systemâ€™s operation must preserve. —SS]

3 System

3.1 S.1 Components

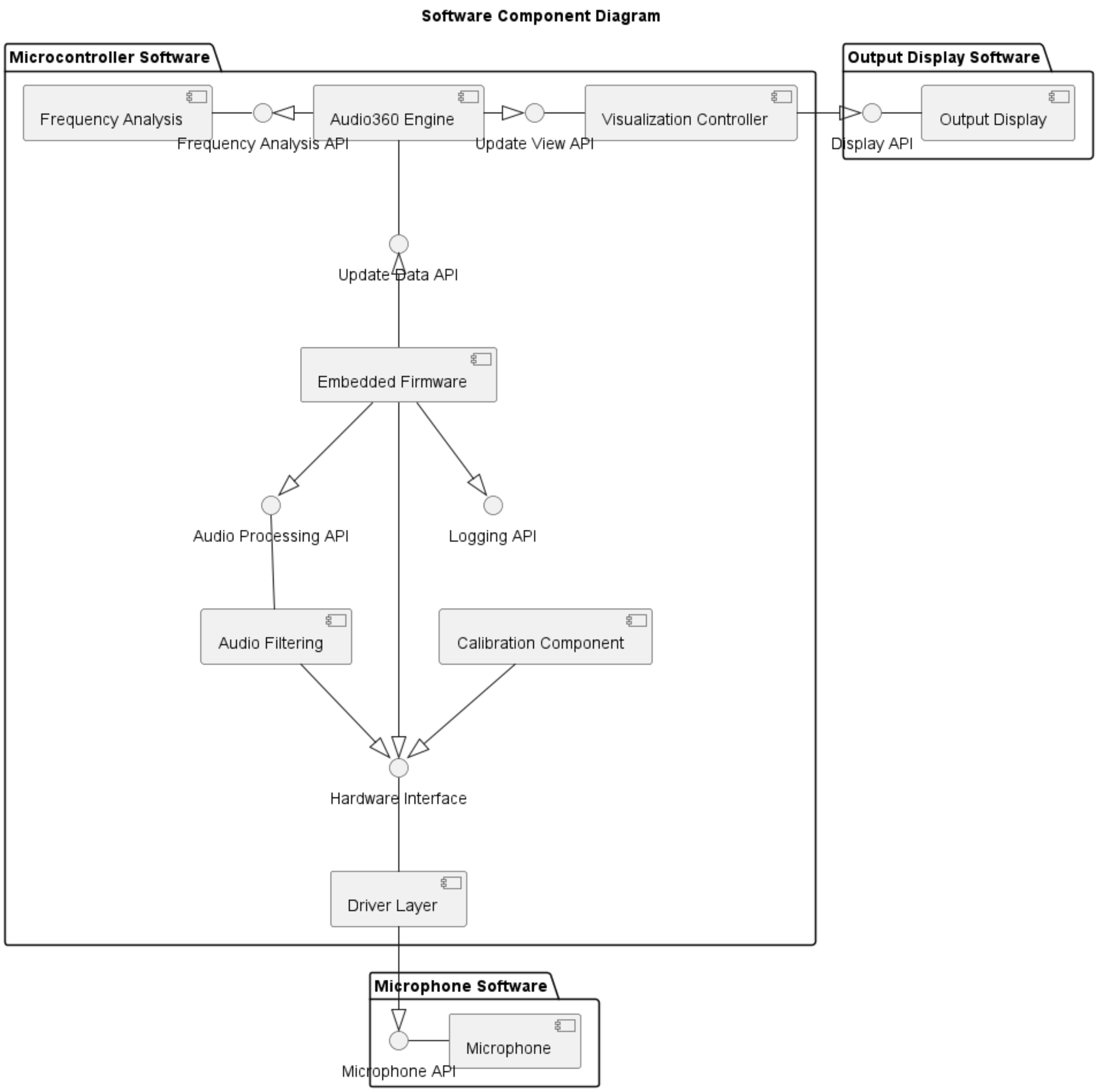

Figure 1: Software component diagram.

3.1.1 Software Components

- **Embedded Firmware:** Main component responsible for managing all embedded software on the Microcontroller.
- **Driver Layer:** Component responsible for providing drivers on the microcontroller. It provides interfaces for high level applications to interact with hardware components

external to the microcontroller

- **Audio Filtering:** Component responsible for processing raw audio signals in real-time sent by the microphones.
- **Calibration Component:** Component for calibrating hardware including microphones and output display.
- **Audio360 Engine:** Main component responsible for running Audio360 features. It serves as the primary interface and controller for the Audio360 system. The component is designed to be hardware agnostic, allowing seamless integration with any firmware running on other hardware.
- **Frequency Analysis:** Component responsible for analyzing properties of processed frequency signals. Sub components of this includes audio classification and direction of arrival estimation.
- **Visualization Controller:** Component reposonsible for creating and sending visualization output to the output display.

3.1.2 Hardware Components

- **Microphone:** Component responsible for collecting audio signals from the environment.
- **Output display:** Component responsible for displaying visuals to the user.
- **Microcontroller:** Component responsible for executing the main software and interfacing with input and output peripheral devices.

3.2 S.2 Functionality

3.2.1 Embedded Firmware

Functional Requirements

- **FR1.1:** The firmware shall schedule tasks based on its priority.
- **FR1.2:** The firmware shall process and synchronize audio signals from all microphones connected to the microcontroller.
- **FR1.3:** The firmware shall handle memory errors to prevent the micrcontroller from crashing.
- **FR1.4:** The system shall perform continuous diagnostics on all hardware components to monitor hardware errors in real-time. This ensures the system can react to failures as soon as they occur.

Non-Functional Requirements

- **NFR1.1:** The firmware shall process audio signals in real-time with monotonic frame sequences. Earliest frames shall have higher priority.
- **NFR1.2:** The firmware shall operate faster than the connected microphone sample rate 44100 Hz, refering to NFR7.1.

3.2.2 Driver Layer

Functional Requirements

- **FR2.1:** The driver layer shall provide an interface for higher level software to interact with the hardware on the microcontroller.
- **FR2.2:** The driver layer shall process hardware interface requests based on system permissions of the requester.
- **FR2.3:** The driver layer shall return error codes upon return to higher-level software to support error handling.
- **FR2.4:** The driver shall maintain data integrity on memory slots that is actively being used.

Non-Functional Requirements

- **NFR2.1:** The driver shall immediately propagate any errors to the firmware layer for proper error handling.

3.2.3 Audio Filtering

Functional Requirements

- **FR3.1:** The audio filtering component shall convert digital audio waveform to frequency domain.
- **FR3.2:** The audio filtering component shall normalize the amplitude of incoming and outgoing signals.
- **FR3.3:** The audio filtering component shall filter incoming audio signals to reduce frequency spectral leakage.
- **FR3.4:** The audio filtering component shall use available hardware acceleration method for efficient computations.
- **FR3.5:** The audio filtering component shall detect and flag audio anomalies such as clipping, lost signal and silence. This enables identification of microphone or signal path faults.

Non-Functional Requirements

- **NFR3.1:** The audio filtering component shall represent audio signals in the frequency domain with less than 10% error from the true value.
- **NFR3.2:** The audio filtering component shall be scalable to handle different input signal sizes without missing timing constraints defined in NFR1.2 for input signal size up to 4096 frames.
- **NFR3.3:** The audio filtering component shall have an audio processing success rate end to end of atleast 90% over 60 seconds.

3.2.4 Audio360 Engine

Functional Requirements

- **FR4.1:** The Audio360 Engine shall retrieve data, such as frequency domain and errors, from the microcontroller via the driver layer for further processing.
- **FR4.2:** The Audio360 Engine shall notify dependent components when new frequency data is available. The latest data shall always be used.
- **FR4.3:** The Audio360 Engine shall control the flow of processed audio data from audio frequency analysis to visualization.
- **FR4.4:** The Audio360 Engine shall disable audio classification and directional analysis features until microphone faults addressed in FR3.5 are resolved. This prevents the generation of unreliable or unsafe outputs.

Non-Functional Requirements

- **NFR4.1:** The Audio360 Engine shall poll data without conflicting with ongoing microcontroller memory writes, ensuring no data loss.
- **NFR4.2:** The Audio360 Engine shall retrieve frequency data within the timing constraints of the embedded firmware defined in NFR1.2.
- **NFR4.3:** The Audio360 Engine shall permanently discard microphone audio data immediately after completion of audio analysis.

3.2.5 Frequency Analysis

Functional Requirements

- **FR5.1:** The frequency analysis component shall classify sound sources based on features extracted from the frequency domain representation of the audio input signal.

- **FR5.2:** The frequency analysis component shall estimate the direction of arrival of the audio source using frequency domain representation of the audio input signal.
- **FR5.3:** The frequency analysis component shall represent the direction of arrival as an angle (θ) in radians. θ is measured relative to the forward axis of the glasses frame to the sound source, Figure 2.
- **FR5.4:** The frequency analysis component shall notify users when a sound classification result has low confidence or is unrecognized to prevent misleading contextual feedback.

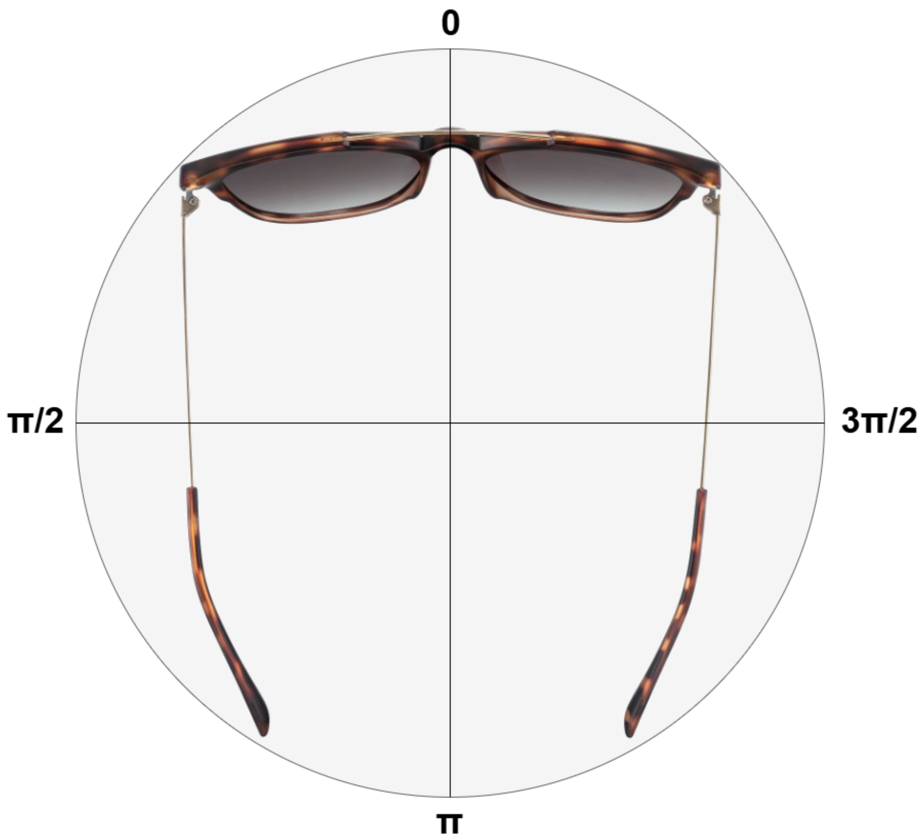

Figure 2: Coordinate System.

Non-Functional Requirements

- **NFR5.1:** The frequency analysis shall classify atleast 3 distinct sound sources relevant to people who are hard of hearing.
- **NFR5.2:** The frequency analysis component shall achieve a minimum classification accuracy of 90% under normal operating conditions.

- **NFR5.3:** The component shall estimate the direction of arrival of an audio source with a maximum error of 0.78 rad. This ensures reliable directional awareness for the user.

3.2.6 Visualization Controller

Functional Requirements

- **FR6.1:** The visualization controller component shall notify users of the direction of an audio source whenever it is detected by the system.

- **FR6.2:** The visualization controller component shall alert users when the core safety features such as direction determination or classification fail. This ensures that users are aware of degraded safety functions.

Non-Functional Requirements

- **NFR6.1:** The visualization controller component shall prioritize to display the most safety-critical information at the time because display space is limited.

- **NFR6.2:** The visualization controller component shall present information in a non-intrusive manner, minimizing visual obstruction so users can safely perform external activities.

3.2.7 Microphone

Functional Requirements

- **FR7.1:** The microphone shall collect soundwaves from its environment and translate to digital representation.

- **FR7.2:** The microphone shall collect soundwaves with the human audiable range (20 Hz - 20 kHz) [3].

Non-Functional Requirements

- **NFR7.1:** The microphone shall collect soundwaves at 44100 Hz. According to Nyquist's theorem, audio sampling should be greater than two times the highest frequency to avoid aliasing [4]. The maximum frequecy audible to humans is approximately 20 kHz. [3].

3.2.8 Output Display

Functional Requirements

- **FR8.1:** The output display shall notify users of the classified sound source.

- **FR8.2:** The output display shall notify users of the direction of the sound source.

Non-Functional Requirements

- **NFR8.1:** The output display shall operate at a minimum of 30 display updates per second to ensure low latency and provide information to the user without perceptible delay.

3.2.9 Microcontroller

Functional Requirements

- **FR9.1:** The microcontroller shall run the main Audio360 software in a closed environment. No external devices other than the microphones and output display shall be able to connect to it. This ensures protection against unauthorized access.

Non-Functional Requirements

- **NFR9.1:** The microcontroller shall have a clock speed of atleast 100 MHz to support audio sampling at 44100 Hz, derived from NFR7.1.

- **NFR9.2:** The microcontroller shall provide atleast four microphone inputs to support directional analysis of audio.

3.3 S.3 Interfaces

Audio360 is designed as a closed embedded system due to its safety-critical application domain, refering to functional requirement FR9.1. External access is intentionally restricted to ensure reliability, security, and user safety. The only user interface is the integrated visual display. The display provides real-time information by notifying users of detected sounds and their corresponding direction. No external programmatic interfaces (APIs) will be provided. This design decision ensures that system integrity is maintained and that no external software can interfere with the safety-critical operation of the device.

3.4 S.4 Detailed usage scenarios

[Examples of interaction between the environment (or human users) and the system: use cases, user stories. —SS]
This section provides detailed versions of the high-level scenarios from **??**, with specific technical details and interaction flows.

3.4.1 S.4.1 Pedestrian Crossing Detection

User Story: As a deaf individual, I want to be alerted when vehicles are approaching so I can cross streets safely.
Scenario: A user is walking in an urban environment and approaches a street intersection. As they prepare to cross, a car approaches from their left side.
Interaction Flow:

1. System detects engine sound pattern characteristic of a vehicle
2. DoA algorithm estimates direction
3. Audio classification identifies the sound as a vehicle
4. Smart glasses display visual indicator showing direction and notifies the user of a vehicle approaching
5. User recognizes the alert and waits for the vehicle to pass before crossing safely

3.4.2 S.4.2 Kitchen Safety Alert

User Story: As a deaf individual, I want to be notified of household safety sounds so I can respond to potential hazards.
Scenario: A user is cooking in their kitchen when a tea kettle on the stove begins to whistle.
Interaction Flow:

1. Microphone array captures the high-pitched whistle sound
2. System processes the audio and identifies the characteristic pattern
3. DoA algorithm estimates the direction
4. Audio classification identifies it as a kettle or alarm sound (90% accuracy target)
5. Smart glasses display directional indicator notifying the user of a kettle whistling ($\leq$ 1s latency)
6. User turns toward the alert and removes the kettle from heat, preventing potential hazard

3.4.3 S.4.3 Social Interaction

User Story: As a deaf individual, I want to know when someone is trying to get my attention so I don't miss social interactions.
Scenario: A user is in a crowded room when someone calls their name from across the space.
Interaction Flow:

1. System captures audio from all directions through the microphone array
2. Voice activity detection identifies human speech patterns
3. DoA algorithm estimates the direction
4. Audio classification identifies the sound as speech or human voice
5. Smart glasses display the information with directional arrow
6. User turns in the indicated direction to make eye contact and engage in conversation

3.4.4 S.4.4 Workplace Awareness

User Story: As a deaf worker, I want to be aware of industrial safety warnings so I can maintain workplace safety.
Scenario: A user is working in an industrial setting when a forklift begins reversing nearby, emitting a warning beep.
Interaction Flow:

1. System detects the characteristic beeping pattern through microphone array
2. DoA algorithm estimates the direction
3. Audio classification identifies it as a warning signal or industrial alarm
4. Smart glasses display prominent visual indicator notifying the user of a forklift backing up and directional arrow
5. User steps aside to maintain a safe distance from the moving equipment

3.5 S.5 Prioritization

[Classification of the behaviors, interfaces and scenarios (S.2, S.3 and S.4) by their degree of criticality. —SS]

3.6 S.6 Verification and acceptance criteria

[Specification of the conditions under which an implementation will be deemed satisfactory. —SS]

4 Project

4.1 P.1 Roles and personnel

[Main responsibilities in the project; required project staff and their needed qualifications. —SS]

4.2 P.2 Imposed technical choices

[Any a priori choices binding the project to specific tools, hardware, languages or other technical parameters. —SS]

4.3 P.3 Schedule and milestones

[List of tasks to be carried out and their scheduling. —SS]

4.4 P.4 Tasks and deliverables

[Details of individual tasks listed under P.3 and their expected outcomes. —SS]

4.5 P.5 Required technology elements

[External systems, hardware and software, expected to be necessary for building the system. —SS]

4.6 P.6 Risks and mitigation analysis

[Potential obstacles to meeting the schedule of P.4, and measures for adapting the plan if they do arise. —SS]

4.7 P.7 Requirements process and report

The development of this project requires iterating over various stages of the requirements process. These stages include:

1. **Elicitation**
   (a) Conduct stakeholder interviews and surveys to gather users' needs.
   (b) Review background documents and research articles describing how core features have been implemented in the past.
   (c) Consult with the technical supervisor of this application (MVM) to gain insight into what approaches can be used.
2. **Analysis**
   (a) Based on information retrieved from the elicitation process, derive a list of soft and hard goals for the application.
   (b) Using the goals defined previously, derive a list of key requirements. This will be influenced by the goals of the stakeholders, but also input on potential limitations based on discussion with the supervisor.
   (c) Group requirements into functional vs. non-functional categories.
   (d) Prioritize requirements using the MoSCoW framework (Must, Should, Could, Won't).
   (e) Iteratively evaluate defined requirements based on new constraints that arise during implementation.
3. **Documentation**
   (a) Write requirements in a structured format that is clear, testable, and unambiguous.

(b) Use labels for different requirements and goals for traceability.

4. **Specification**

   (a) Using the `docs/` folder in the main repository for this project, update the various reports with the latest information based on what was discussed or finalized in other stages of the elicitation process.

5. **Validation**

   (a) Share draft requirements with stakeholders for confirmation.

   (b) Share requirements with the project supervisor to get expert opinion on feasibility of requirements.

   (c) Within the team, ensure the requirements are feasible, measurable, and aligned with the project scope.

As this project follows the V-model methodology, Figure 3, the team will aim to make the various stages of the requirements process mentioned above linear. Since although this model allows the team to re-visit previous parts, it will cost a lot to change in later stages of the project. This is why this team is heavily prioritizing the initial stages of the requirements process.

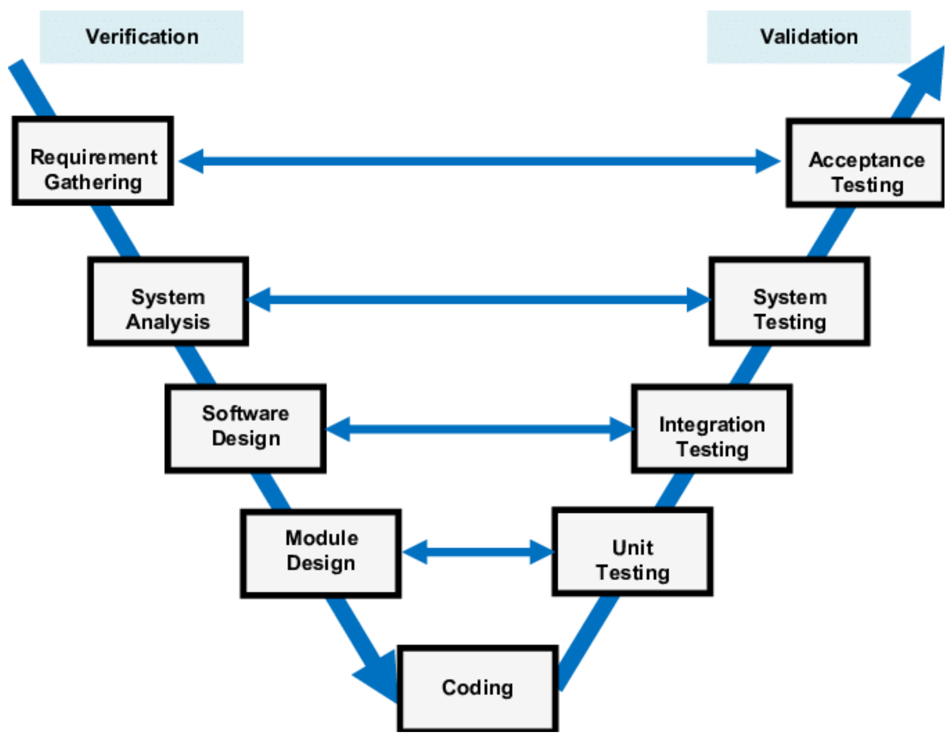

Figure 3: Illustration of the v-model process.

References

[1] Healthing.ca, "Hearing loss in canada: Stats, impact and resources," *Healthing.ca*, 2025.

[2] N. I. on Deafness and O. C. Disorders, "Hearing aids," *National Institute Deafness and Other Communication Disorders*, 2022, nIH Publication No. 99-4340.

[3] D. Purves, G. J. Augustine, D. Fitzpatrick *et al.*, *The Audible Spectrum*. Neuroscience, 2001. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK10924/

[4] DataForth, "Sampling laws," *DataForth*, 2025. [Online]. Available: https://www.dataforth.com/data-acquisition-and-control-sampling-law

Appendix — Reflection

The purpose of reflection questions is to give you a chance to assess your own learning and that of your group as a whole, and to find ways to improve in the future. Reflection is an important part of the learning process. Reflection is also an essential component of a successful software development process.

Reflections are most interesting and useful when they're honest, even if the stories they tell are imperfect. You will be marked based on your depth of thought and analysis, and not based on the content of the reflections themselves. Thus, for full marks we encourage you to answer openly and honestly and to avoid simply writing "what you think the evaluator wants to hear."

Please answer the following questions. Some questions can be answered on the team level, but where appropriate, each team member should write their own response:

1. What went well while writing this deliverable?
2. What pain points did you experience during this deliverable, and how did you resolve them?
3. How many of your requirements were inspired by speaking to your client(s) or their proxies (e.g. your peers, stakeholders, potential users)?
4. Which of the courses you have taken, or are currently taking, will help your team to be successful with your capstone project.
5. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.
6. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?