Problem Statement and Goals
Software Engineering

Team #23, Project Proxi
Savinay Chhabra
Amanbeer Singh Minhas
Gourob Podder
Ajay Singh Grewal

Table 1: Revision History

Date	Developer(s)	Change
September 22, 2025	Gourob Podder	Initial Draft
September 22, 2025	Gourob Podder	Revised All Sections; Added Reflection

1 Problem Statement

1.1 Problem

Many individuals face barriers when interacting with computers due to physical disabilities, age-related impairments, or limited familiarity with digital devices. Traditional input methods such as keyboards, mice, and graphical user interfaces can create exclusion, preventing these users from accessing essential digital services, educational resources, and communication tools. This digital divide not only limits independence but also reduces opportunities for social and professional participation.

The problem, therefore, is the lack of an intuitive, accessible, and inclusive interface that enables these users to interact with computers effectively without relying on traditional input devices.

1.2 Inputs and Outputs

Inputs:

- User-provided spoken language, expressed as commands, requests, or questions.

- Contextual information about the user's environment and preferences.

- System status and feedback information.

Outputs:

- System-generated responses delivered in natural language.

- Corresponding system actions/tasklists that reflect the interpreted intent of the input.

1.3   Stakeholders

- Primary Users: Individuals with disabilities, elderly users, or people with limited digital literacy who require accessible interaction with computers.

- Secondary Users: Caregivers, educators, and family members who assist primary users.

- Organizations: Institutions and workplaces seeking to provide inclusive technology solutions.

- Developers/Researchers: Teams advancing assistive technology and inclusive design.

1.4   Environment

Hardware: A standard personal computer equipped with a microphone and speakers.

Software: Runs on a general-purpose operating system like Windows or MacOS with support for speech input and output. The system is intended to be platform-agnostic and should not rely on specialized hardware beyond audio input/output.

2   Goals

1. Achieve at least 90% speech recognition accuracy for supported languages in quiet environments.

2. Provide spoken responses with a latency of less than 2 seconds for common commands.

3. Enable users to complete at least 80% of core computer tasks (e.g., opening files, browsing the web, writing documents) entirely through voice commands.

4. Improve task completion rates for primary users by 20% compared to traditional input methods (measured through usability studies).

5. Attain a user satisfaction rating of 4 out of 5 or higher in accessibility-focused usability tests.

3   Stretch Goals

1. Support multiple languages and dialects, achieving at least 85% recognition accuracy in each.

2. Implement adaptive personalization to reduce recognition errors by 15% for frequent users by learning their speech patterns.

3. Enable multimodal interaction, such as combining voice with simple gestures or touch, to increase task flexibility by at least 10%.

4. Provide offline functionality capable of executing at least 50% of core commands without internet access.

4   Extras

1. **Usability Testing** : Conduct structured usability studies with target user groups (e.g., individuals with disabilities, elderly users, and those with limited digital literacy). Testing will measure ease of use, task completion rates, and overall satisfaction to ensure Proxi effectively addresses accessibility needs.

2. **User Documentation** : Provide clear, accessibility-focused documentation and onboarding materials, including written guides and simple audio-based tutorials, to support users in adopting and effectively using Proxi.

Appendix — Reflection

What went well while writing this deliverable

Writing the deliverable went well in terms of clearly defining the problem and the scope as we all have had experiences with people in our lives that would benefit from a solution to this problem. The abstraction of inputs and outputs helped us focus on the high-level functionality of Proxi without getting bogged down in implementation details. We were able to identify stakeholders, environments, and goals in a structured manner, ensuring that the documentation aligned closely with the Capstone checklist.

Pain points experienced and how they were resolved

One of our pain points we encountered was ensuring that the inputs, outputs, and goals were sufficiently abstract while still being precise and measurable. Initially, our examples were too concrete, mentioning specific applications or tasks, which could have limited the perceived generality of the system. We solved this by rephrasing inputs as "spoken language commands" and outputs as "system responses and actions reflecting user intent." Another difficulty was translating high-level goals into measurable success metrics. We overcame this by defining clear metrics such as speech recognition accuracy, and response latency.

Adjusting the scope of goals for Capstone suitability

We adjusted the goals to ensure the project remained feasible within the Capstone timeline. While Proxi's long-term vision could include multi-language support, offline functionality, and multimodal input, we focused the core goals on voice-based interaction, accessibility, and measurable improvements in task completion and user satisfaction. This allows us to ensure the project is ambitious enough to demonstrate meaningful innovation in assistive technology, while remaining manageable in scope for a capstone. Finally, stretch goals were included to outline potential extensions without inflating the baseline expectations.