

Problem Statement and Goals

Software Engineering

Team #22, TeleHealth Insights
Mitchell Weingust
Parisha Nizam
Promish Kandel
Jasmine Sun-Hu

Table 1: Revision History

Date	Developer(s)	Change
Sept 20	Promish Kandel	Added Initial problem statement
Sept 22	Promish Kandel	Modified Problem, Added Goals, Inputs/Outputs, Environment
Sept 23	Promish Kandel	Modified Goals, Added Challenge Level and Extras
Sept 23	Mitchell Weingust	Proofread and Modified: Problem Statement, Goals, Stretch Goals, Challenge Level and Extras

1 Problem Statement

[You should check your problem statement with the problem statement checklist. —SS]

[You can change the section headings, as long as you include the required information. —SS]

1.1 Problem

Children with speech difficulties often require regular assessments to track their progress during speech therapy. These assessments are vital in monitoring development, identifying issues, and adjusting treatment plans accordingly. For bilingual children, these assessments can be challenging because of the shortage of bilingual speech language pathologists (SLPs). Parents, who are the key informants with understanding of multiple languages spoken in the home,

have the potential to become helpers in bilingual assessments, especially in telehealth settings. However, existing bilingual language assessment tools are often designed and built based on SLPs' in-person practices, and lack designs specifically addressing parents' needs and behaviors when they act as at-home test administrators. The lack of such design considerations could also affect the outcome from these assessments, when SLPs need to assess and understand children's results from remote assessments facilitated by parents. Hence, there are opportunities to build a system that better supports bilingual language assessments at home settings for children and parents, through providing better guidance and instructions for parents, capturing more contextual data to complement the results for SLPs, and engaging children in these assessments.

1.2 Inputs and Outputs

1.2.1 Input

- Video Stream of the individual taking the assessment
- Selecting answers via mouse clicks
- Microphone Audio

1.2.2 Output

- Audio analysis for background noise
- Video analysis for keyboard and face movement
- Analysis of the selected answer for each question
- Summary of results and analysis details for clinicians

[Characterize the problem in terms of “high level” inputs and outputs. Use abstraction so that you can avoid details. —SS]

1.3 Stakeholders

- Parents with children that have speech difficulties
- Children with speech difficulties (who need to take language assessments)
- Clinicians (SLPs) who work with children that have speech difficulties
- Project Researcher, Dr. Yao Du, Clinician Assistant, Professor at the University of Southern California
- Project Supervisor, Dr. Irene Ye Yuan, Assistant Professor in the Department of Computing and Software at McMaster University

1.4 Environment

1.4.1 Software

Software should be cross-compatible amongst Linux, MacOS and Windows operating system

1.4.2 Hardware

Hardware required includes any personal computer with:

- Sounds Output (Speaker, Headphones)
- Microphone
- Webcam (Internal/External)

[\[Hardware and software environment —SS\]](#)

2 Goals

- Intuitive and helpful interface that can guide parents to effectively administer language tests.
 - The application should be easy to navigate with clear and meaningful symbols. It should also provide feedback so that end users are aware of their interactions being processed throughout the assessment.
- Engaging interface and interaction for children when taking the assessment.
 - The webpage should have a simple but visually appealing design to keep children engaged throughout the assessment.
 - The webpage should have colours and images to attract attention to the assessment's questions and selections.
- Provide reliable assessment results for SLPs by capturing additional contextual data and preliminary analysis
 - The application should provide additional information to SLPs to identify background interference and signs of bias or test complications.
 - The application should filter out noise and be able to identify multiple users to detect additional guidance from others.
- Data security to ensure health/sensitive records are stored and accessed securely
 - The software should use a security protocol to store and retrieve sensitive data from a secure database

- Provide cross-platform integration for different screen sizes
 - The webpage should be accessible for parents and children, regardless of the chosen device, by rendering correctly on all screen sizes and formats.

3 Stretch Goals

- Provide a detailed interface for clinicians to analyze stored data
 - The software should provide a web interface for clinicians to upload assessment results, for storing and analyzing data more efficiently.

4 Challenge Level and Extras

Our challenge level is general as the project scope is limited in terms of how much research is required. The required domain knowledge is basic web-design in a stack of our choice. We are also planning on using open-source large language models for audio and video processing.

The extras for this project includes:

- User Documentation: Providing users with a guide on how to use and better understand the system.
- Usability Testing: Provide the team information on how usable the design is, along with improvements on how the system looks and feels

[State your expected challenge level (advanced, general or basic). The challenge can come through the required domain knowledge, the implementation or something else. Usually the greater the novelty of a project the greater its challenge level. You should include your rationale for the selected level. Approval of the level will be part of the discussion with the instructor for approving the project. The challenge level, with the approval (or request) of the instructor, can be modified over the course of the term. —SS]

[Teams may wish to include extras as either potential bonus grades, or to make up for a less advanced challenge level. Potential extras include usability testing, code walkthroughs, user documentation, formal proof, GenderMag personas, Design Thinking, etc. Normally the maximum number of extras will be two. Approval of the extras will be part of the discussion with the instructor for approving the project. The extras, with the approval (or request) of the instructor, can be modified over the course of the term. —SS]

Appendix — Reflection

[Not required for CAS 741 —SS]

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 did you and your team adjust the scope of your goals to ensure they are suitable for a Capstone project (not overly ambitious but also of appropriate complexity for a senior design project)?