

# Reflection and Traceability Report on Software Engineering

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

## 1 Changes in Response to Feedback

The following section indicates changes in response to feedback for all the past submitted documents.

### 1.1 SRS and Hazard Analysis

#### 1.1.1 SRS

- **Feedback:** The Use Case Diagram (Figure 5) seems too complex and focuses too much on implementation. Use case diagrams should be a high level representation of the interactions between the actors and actions. The use case diagram should not have implementation or internal working details of the application.

**Source:** Peer Review by abedmohammed

**Changes Made:** *WONTFIX*: The Use Case Diagram is at a level of detail and clarity that helped the team figure out the necessary use cases for the system, and how each actor engages with it.

As well, from the diagram I do believe the 'extends' arrows are functioning appropriately. Finalize Order extends to Update Order Status as finalizing an order comes before updating order status. This is similar to how we are using Extends, as Authenticating the user happens prior to Accepting the user.

**Issue Link:** [#119](#)

- **Feedback:** In the rationales of some of the requirements under 11.2, there could be more focus on users. For example, UH-PI1 which states "The system shall support multiple languages" rationalizes this by "The assessment must be conducted in two languages for the purpose of the research study." While this is an important and valid justification, it can also be

pointed out that some of the main users; primary children and their parents, would benefit tremendously from multi-language support.

**Source:** Peer Review by jane-klavir

**Changes Made:** Fixed rationale for system supporting multiple languages, to indicate the benefits for parents and children to be able to navigate the system in their preferred language, rather than just navigating the site for the benefits of a research study/clinicians.

**Issue Link:** [#121](#)

- **Feedback:** In section 9.1, authentication is defined by the functional requirements FR-A1 through FR-A5. Although they differentiate between the different types of users, with parent accounts being able to complete assessments and clinician accounts having the permission to view results, there is no requirement discussing how these permissions are enforced, and what happens when edge cases are encountered. Some of the edge cases that can be explored are when parents try to view clinician data or what an admin account's permissions are as a whole (the only data they should be able to view should be encrypted, for privacy reasons)

**Source:** Peer Review by jinalkast

**Changes Made:** *WONTFIX*: Parents don't have access to the clinician dashboard, which can only be accessed through the clinician's login.

Encryption and security are mentioned further in NFRs as security is a quality of the system, not an authentication functional requirement.

**Issue Link:** [#123](#)

- **Feedback:** The diagram under section 6.2 is currently considering too many components. Specifically, I don't think the 'Database', 'Backend Server', 'Language Assessment Website', 'Video Analysis Model', and 'Audio Analysis Model' should be separate components, but rather one component.

**Source:** Peer Review by nguyes44

**Changes Made:** Context of the work diagram has been simplified, reducing complexity and the number of separate components. Now, only 3 main components exist; children, parents, clinicians, all connecting to the system by their means of interaction. Reducing the complexity in the diagram made it simpler to follow, and compacted the system.

**Issue Link:** [#115](#)

- **Feedback:** Most of your constraints are intuitively defined by the client's needs for the platform, but it would help to add a fit criterion and rationale for each, as defined in the template's Constraints section.

**Source:** Peer Review by arvindshastri

**Changes Made:** Added rationales and fit criteria for all constraints to ensure all constraints can be met, and means of achieving all of them are attainable.

**Issue Link:** [#117](#)

- **Feedback:** In section 16.1, CU-CR1 has the fit criterion that mentions that the platform needs to be validated by a cultural "consultant" and CU-CR2 mentions that the assessments must be fully translated to both languages. However, in Section 23, there is no mention of any costs that are related to "translation" or a "consultant". These sections might be reviewed and the associated cost might be mentioned in the document.

**Source:** Peer Review by aysuozdal

**Changes Made:** Removed the requirement for a cultural consultant, as it is clearly out of scope of the project. Instead, replaced the fit criterion to consider a minimum of 5 representatives of the primary user groups to ensure no instances of insensitivity.

**Issue Link:** [#122](#)
- **Feedback:** Some Appearance Requirements go into implementation details and are not completely blackbox

**Source:** Peer Review by jave-klavir

**Changes Made:** Removed implementation details as SRS should not be discussing the specific technologies used. Made wording more black-box to give developers and team the ability to adhere to requirements without focusing on the implementation details (example, changed 'large buttons, simple instructions, and child-friendly imagery' to 'accessible and age-appropriate elements')

**Issue Link:** [#124](#)
- **Feedback:** Minor spelling errors (e.g. Assessmenet on page xxxviii)

**Source:** TA

**Changes Made:** Fixed minor spelling errors throughout the document (Assessmenet appeared numerous times in particular). As well, changed numbering from roman numerals to arabic for ease of understanding (after the table of contents and revision history).

**Issue Link:** [#194](#)
- **Feedback:** Figure 5: "User" should be "Use"

**Source:** TA

**Changes Made:** Fixed simple spelling error for 'Use Case Diagram' figure title

**Issue Link:** [#192](#)
- **Feedback:** Use `\begin{table}` and `\caption{}` inside that so tables have numbers. Then reference them in the text.

**Source:** TA

**Changes Made:** Added table captions to all tables, and referenced/mentioned all tables in text throughout entire document. Further, added a list of tables and a list of figures to the document for ease of access.

**Issue Link:** [#193](#)
- **Feedback:** MS-MR1 is probably a bit hard to test. How will you know until you have to modify something down the line?

**Source:** TA

**Changes Made:** Modified MS-MR1 to consider how modular systems improve maintainability by isolating changes to specific components. Fit criterion is modified such that 'Any update, modification, or replacement of an individual component shall not require any changes to unrelated modules/components; no more than 1 configuration update to dependent components within a module; and have no downtime exceeding 5 minutes during initial deployment'.

**Issue Link:** [#196](#)

- **Feedback:** Doesn't identify template or any changes made to it.

**Source:** TA

**Changes Made:** Added mention that the SRS document 'is based on the Volere Requirements Specification Template. This document adheres to the standards of this template, exactly as given by the course instructor, as modifications indicates a loss of some of the advantage of a standardized template. As a result, the document's template has not been modified. Any non-applicable sections have been stated accordingly, without removal.'

**Issue Link:** [#188](#)

- **Feedback:** The LaTeX formatting of one of the tables is a bit awkward, with the lines not going all the way to the horizontal ones. Look up Booktabs tables in LaTeX, they look much better. Actually, the table on page xxii looks better for the most part.

**Source:** TA

**Changes Made:** Fixed 'Work Partitioning of BUC' table's formatting such that lines fully extended, and table formatting was consistent throughout the document.

**Issue Link:** [#189](#)

- **Feedback:** Good job mentioning HIPAA. Is there a Canadian equivalent that you also need to follow? If so, please explore it and modify it.

**Source:** TA

**Changes Made:** Given the project will be used in both California and Ontario, HIPAA remains in the document. In addition, PHIPA (Ontario – Personal Health Information Protection Act) has been included as a reference and standard, as well as PIPEDA (Personal Information Protection and Electronic Documents Act - federal law for collecting, using, and disclosing personal information).

**Issue Link:** [#186](#)

- **Feedback:** Make use of symbolic constants instead of "magic numbers" in the document.

**Source:** TA

**Changes Made:** Went through the entire document to remove any and all instances of magic numbers in the document. Further, for ease of access, added an additional appendix 'Appendix - Symbolic Parameters' to

conveniently access all variables and their values.

**Issue Link:** [#187](#)

- **Feedback:** 100% of users is probably too ambitious, should have some leeway to that. 80-90% is usually a good idea. Some users just might not be able to do things on the first try (e.g. LF-AR2).

**Source:** TA

**Changes Made:** Fixed all 100% ambitions throughout the document (changed to HIGH\_SUCCESSION\_RATE instead). In addition, when considering testing with real people, we are considering completed assessments, records, sessions, reports.

**Issue Link:** [#195](#)

- **Feedback:** It is best for performance requirements to say "The system shall" rather than saying "The system must": this is so that if the requirement is violated, using "shall" can provide a stronger legal basis for enforcement. One of the capacity requirements mention that "The system must accommodate a minimum of 2000 user accounts at launch.", the fit criterion does not mention a way to verify that, when you reach the verification stage, how do you plan on verifying this requirement is met? will you populate 2000 user accounts to verify that? I think the fit criterion could have some description on how this would be verified. This could also apply to other requirements as well.

**Source:** Peer Review by george536

**Changes Made:** Replaced 'the system must' with 'the system shall' for all performance requirements for consistency and accuracy. Updated the 2000 user requirement to specify the fit criterion for measuring the capacity requirement (The system shall be designed to support MIN\_USER\_ACCOUNTS user accounts at launch, through system design documentation and capacity planning, ensuring the infrastructure and architecture are capable).

**Issue Link:** [#116](#)

- **Feedback:** The goals of the project should be measurable through some metric, or contain a reference to FRs/NFRs to quantify them, as discussed in the Volere template

**Source:** Peer Review by arvindshastri

**Changes Made:** *WONTFIX*: The FRs and NFRs stated throughout the document are how the team plans on managing and quantifying our goals. The goals section is treated more as a general overview and a brief introduction to what the team plans on exploring throughout the document. Given the goals are intended to be clear and easy to understand, the quantification of these goals are explored further throughout the document, and will not be explicitly stated in the Goals section itself.

**Issue Link:** [#118](#)

- **Feedback:** Ensure you have no section headings with a subsection heading right under it and no text introducing it. And don't just leave sections

blank, at least insert Not Applicable, or a small sentence explaining why.

**Source:** TA

**Changes Made:** Added brief description under every section heading, ensuring no section headings with a subsection heading right under it was not missing an introduction. Regarding 'don't just leave sections blank, at least insert Not Applicable, or a small sentence explaining why' No sections were left blank. However, there was a section with a table not in the correct position, giving the illusion there was an empty section previously.

**Issue Link:** [#191](#)

- **Feedback:** Export and insert diagrams as PDFs so they are not pixelated (e.g. all the diagram figures).

**Source:** TA

**Changes Made:** Reuploaded all diagrams to be PDFs to increase readability and improve professionalism.

**Issue Link:** [#190](#)

- **Feedback:** Connect your requirements to your phase-in plan by assigning priorities or phases to them.

**Source:** TA

**Changes Made:** Added 2 new tables, 'Table 16: Development Revision 0 Deadlines'; 'Table 17: Development Revision 1 Deadlines'. Broke down requirements into revision 0 and revision 1 deadlines (separated by corresponding tables). Assigned priorities and deadlines for accomplishing each task, so the team had clear goals in mind while working on the project.

**Issue Link:** [#198](#)

- **Feedback:** Make sure you put team 22 in all the issues (they were in this one, but I noticed some missing for the HA)

**Source:** TA

**Changes Made:** Communicated with full team to ensure that everyone went back to their peer review issues, and added 'Team 22' in the issue title. Provides easy and convenient tracability.

**Issue Link:** [#199](#)

- **Feedback:** Good references - are there any about speech therapy that could be inserted?

**Source:** TA

**Changes Made:** Added an additional reference regarding speech therapy, which is mentioned in 'The Current Situation' section of the SRS document. The citation describes how to properly execute a variety of activities for effective and engaging at-home speech therapy, which was used as a resource/inspiration for our design.

**Issue Link:** [#208](#)

- **Feedback:** Solution constraint 3.1.7 denotes that the platform must comply with WCAG 2.1 accessibility standards. However, in section 11.5 (accessibility requirements), there is no mention of how the system will

contain these considerations for digital accessibility, or how the adherence will be measured.

**Source:** Peer Review by jinalkast

**Changes Made:** Added on to 11.5, how compliance with with WCAG 2.1 accessibility standards will be verified, through a manual review process, comprising of testing with a variety of technologies such as screen readers, keyboard-only navigation, touch interactions, and evaluation across a variety of screen sizes.

**Issue Link:** [#125](#)

- **Feedback:** Under the Type column of the Data Dictionary, Parent Class, Class and Clinician Class are all use as types. Confusingly, Clinician has the Clinician Class type, but Parent only has the Class type. This is a mixup in terminology. The Content column for User, Parent and Clinician are very vague, only listing "Content". This should be improved to make it more clear what data each type has.

**Source:** Peer Review by thompa39

**Changes Made:** 'Content' has been changed to 'Clinician Content' and 'Parent Content' for clarity in the data dictionary, improving consistency and making it clear what each piece of data is used for.

**Issue Link:** [#120](#)

- **TA Feedback:** I love the Paul Blart persona! I think you need one or two more for other important users, like medical professionals. **Peer Feedback:** Personas do not cover all important users/stakeholders. Without user personas, your document doesn't provide a detailed enough description of what the users want from the system, instead it mainly just covers their role (child takes the assessment, parent delivers it, etc.). Your user persona for the parent was really good and just adding two more, one for the children and one for the SLPs, should provide readers of the SRS all the insight they need to understand your stakeholders.

**Source:** TA and Peer Review by marmanios

**Changes Made:** Added 2 additional important users; a clinician user profile, and a child user profile. Now there are 3 important personas which cover all important users of our system.

**Issue Link:** [#197](#) [#114](#)

### 1.1.2 Hazard Analysis

- **Feedback:** Use `\multirow` to better share/split rows instead of relying on numbers. Reference tables in the body of the document.  
**Source:** TA  
**Changes Made:** Improved the format of the two FMEA tables in the document using `\multirow`. The content of the hazard analysis tables was not directly cited in the narrative, so no in-text references were added. In the future, relevant sections will be updated to include table references where appropriate.  
**Issue Link:** [#233](#)
- **Feedback:** Ensure you are getting to the root cause of the problem in the recommended actions. For instance, HA-D1 : this is a pretty weak way to solve this problem. Yes, backups are important in case something goes wrong, but the better thing is to ensure that injection attacks cannot happen in the first place. No mention of using encryption to protect data.  
**Source:** TA  
**Changes Made:** Rewrote several vague recommended actions in the FMEA table to focus on root causes, such as preventing injection attacks through input validation and parameterized queries. Also added explicit recommendations to implement encryption (e.g., AES-256) to protect data at rest within the database.  
**Issue Link:** [#234](#)
- **Feedback:** Unclear boundaries and Components  
**Source:** Peer Review by marmanios  
**Changes Made:** Renamed "UI" to "User-Facing Application" to better reflect its role in the system. Reorganized the placement of the audio/video recording and analysis components to better align with data flow and reduce overlap between functional boundaries.  
**Issue Link:** [#131](#)
- **Feedback:** Incomplete failure modes/recommended actions for Video Recording component.  
**Source:** Peer Review by jinalkast  
**Changes Made:** Added additional failure modes and recommended actions for Video Recording component.  
**Issue Link:** [#153](#)
- **Feedback:** Critical Assumptions and out of scope hazards are functionally equivalent.  
**Source:** Peer Review by marmanios  
**Changes Made:** Changed the wording of critical assumptions for additional clarity and changed out of scope hazards to talk about additional points that were not addressed in critical assumptions.  
**Issue Link:** [#155](#)



- **Feedback:** "Errors in implementation" for HA-UI2 is ambiguous.  
**Source:** Peer Review by jane-klavir  
**Changes Made:** Changed the wording from "Errors in implementation" to "UI components not rendering correctly due to screen size, device-specific compatibility issues, or missing accessibility focus states."  
**Issue Link:** [#156](#)
- **Feedback:** In the "Failure Mode" column for the Audio Recording it says the audio recording has noise which is a good failure mode however I think that it can be further explained as to what kind of noise it is.  
**Source:** Peer Review by Harshil-P21  
**Changes Made:** Changed the wording from "Audio recording has noise" to "Audio recording has background noise interference".  
**Issue Link:** [#157](#)
- **Feedback:** Unreasonable time limit for MFA completion.  
**Source:** Peer Review by jinalkast  
**Changes Made:** Adjusted "Users accessing sensitive data must pass a multi-factor authentication process within 2 minutes of a code being sent" to "within 5 minutes of a code being sent".  
**Issue Link:** [#158](#)
- **Feedback:** Authentication is listed as a component when it should be more of a feature.  
**Source:** Peer Review by marmanios  
**Changes Made:** The team agreed that authentication was more a feature of our backend instead of it's own component, so it was removed from the hazard analysis and treated as a feature of the backend instead.  
**Issue Link:** [#159](#)

## 1.2 Design and Design Documentation

- **Feedback:** ACs: exact hardware that is used (lots of different devices, microphones, cameras, etc.)  
**Source:** TA  
**Changes Made:** Added an AC3 to account for the exact hardware changes. This was just a change in the design documentation; nothing was changed in the code.  
**Issue Link:** [#498](#)
- **Feedback:** I do not think that input/output devices belong under UCs...some people may use touchscreens, etc.  
**Source:** TA  
**Changes Made:** We removed UC1, taking into account the feedback. This information is actually covered by the AC3 added from the previous feedback.  
**Issue Link:** [#499](#)
- **Feedback:** What do you mean by “Core structure” of the question bank?  
**Source:** TA  
**Changes Made:** We updated the wording throughout the MG document to make it clearer what “core structure” was, and to clearly define exactly what the question bank is by adding the JSON to the appendix of the MIS document.  
**Issue Link:** [#500](#)
- **Feedback:** UC3: Does “bilingual” need to be specified?  
**Source:** TA  
**Changes Made:** For other sections, bilingual does matter as it’s an important part of the capstone; however, in this context, it does not matter, thus it was removed from UNC2 in the MG document.  
**Issue Link:** [GitHub #501](#)
- **Feedback:** App controller is probably more of a behaviour hiding module...  
**Source:** TA  
**Changes Made:** This was great feedback as our team was thinking the same thing. In our final design, we actually didn’t make an app controller and instead decided to make a Landing Page GUI. We also moved this module to be a behaviour hiding module, instead of software hiding like the app controller was before. In our codebase, this also made it easier to connect the clinician GUI and parent GUI, as it was just routing two buttons. This change can also be seen in our module hierarchy diagram. Finally, due to this change, the clinician GUI and parent GUI both independently talk to the API gateway, which we found was easier to code since it didn’t have to be routed through an app controller like in our previous design.  
**Issue Link:** [GitHub #502](#)

- **Feedback:** Media, Audio, and Video processing modules feel more like SW hiding modules.  
**Source:** TA  
**Changes Made:** This was one piece of feedback that we disagreed with. We decided to keep these as behaviour hiding modules, as these modules are more about analyzing the video and audio than recording them. All this module cares about is an input for a pre-recorded video and returning the bias.  
**Issue Link:** [#503](#)
- **Feedback:** Page 4: Can remove “(M??)”. Consider a “logical” module for mic/camera feed.  
**Source:** TA  
**Changes Made:** We removed the “(M??)”. Due to this change, we realized that section 5 and section 7 in the MG document were referencing each other incorrectly. We changed it so that everything points towards section 7 and not section 5. The numbering was also updated to not just be 1 but something more descriptive like 7.2.3. I also added a logical module to section 5 and the module hierarchy diagram in section 8 of the MG document. This was just for design to capture the ability of a browser to capture audio and video in the background. Nothing was changed in our actual codebase.  
**Issue Link:** [#504](#)
- **Feedback:** Page 4: Typo: “Backn”.  
**Source:** TA  
**Changes Made:** Fixed the typo on page 4.  
**Issue Link:** [#505](#)
- **Feedback:** APP is capitalized sometimes but not others (e.g., Table 1).  
**Source:** TA  
**Changes Made:** Everything was changed to “APP” to make sure the document was consistent.  
**Issue Link:** [#506](#)
- **Feedback:** Hyperlink the modules in the list (Section 5) and in the traceability matrices.  
**Source:** TA  
**Changes Made:** The hyperlinks were added to section 5 and the traceability matrix, with the new naming convention 7.x.x. The new hyperlinks point towards the respective sub-part in section 7.  
**Issue Link:** [#507](#)
- **Feedback:** Include diagrams as PDFs to enhance clarity and zoomability.  
**Source:** TA  
**Changes Made:** All diagrams were converted to PDFs and attached.  
**Issue Link:** [#508](#)

- **Feedback:** Not clear what the formats of the JSON objects are.  
**Source:** TA  
**Changes Made:** We added a section in the appendix of the MIS document that shows the JSON structure for modules that need it. This clearly shows what the format of the JSON structure is and the information it contains.  
**Issue Link:** [#509](#)
- **Feedback:** Some things (like contents of the JSON objects) are not clear, might impede implementing.  
**Source:** TA  
**Changes Made:** In the MIS document, instead of saying JSON objects for everything, we made record objects that point toward the appendix, which has the actual JSON object. This way it is clear how the JSON object looks and how the data differ from each other. If we were to redo our capstone, this update would be very helpful in implementing the backend, as you would know the exact parameters each function takes.  
**Issue Link:** [#510](#)
- **Feedback:** Answer should be focused on your project's architectural design and the SE principles.  
**Source:** TA  
**Changes Made:** I agree our answers were a bit generic in terms of the functions we were making and their descriptions. We changed the wording of the question bank service and result service to make it more specific to what our project needed. The authentication service was also updated a bit to take into account our use of a security code, which other platforms might not have, as this is unique to our design.  
**Issue Link:** [#511](#)
- **Feedback:** The output of the Media Processing Module is incomplete about half the time.  
**Source:** Peer Review by jinalkast  
**Changes Made:** Taking this review into account, I updated the wording for the media processing module to make it clear that both video and audio processes will be called and then their outputs would be combined.  
**Issue Link:** [#342](#)
- **Feedback:** There are contradictions in the MIS of the storage module.  
**Source:** Peer Review by marmanios  
**Changes Made:** We added the JSON to the appendix to make it clearer how the data is being stored. I think part of this feedback was the misunderstanding that JSON can't be stored in AWS, which I explained in the comments of the issue.  
**Issue Link:** [#341](#)
- **Feedback:** The App Controller cannot handle scenarios in which no input is provided.

**Source:** Peer Review by abedmohammed

**Changes Made:** This feedback was ignored as we didn't have an app controller anymore.

**Issue Link:** [#340](#)

- **Feedback:** The Real-Time Feedback Module classification or naming might be unclear.

**Source:** Peer Review by jane-klavir

**Changes Made:** This feedback was ignored as we didn't have a real-time feedback module anymore, as we felt that real-time feedback could be distracting to the child during an assessment.

**Issue Link:** [#339](#)

- **Feedback:** The MG "secret" is not actually secret or otherwise is mis-named.

**Source:** Peer Review by marmanios

**Changes Made:** We updated the "secret" to be better worded, as I agree it sounded more like a service before updating it.

**Issue Link:** [#338](#)

- **Feedback:** The Module Guide references vague Use Cases (UCs).

**Source:** Peer Review by marmanios

**Changes Made:** This was a similar feedback to our TA feedbacks which was corrected by making the wording of our UCs clearer.

**Issue Link:** [#337](#)

## 1.3 VnV Plan and Report

### 1.3.1 VnV Plan

- **Feedback:** Should remove the ? from usability survey questions section. Some text overflows boxes. Some sections have subsections inside them with no introductory text.  
**Source:** TA  
**Changes Made:** Removed ? from usability survey questions, fixed the text overflow in test case boxes, and added subsection introductory text to any subsections that were missing them.  
**Issue Link:** [#292](#)
- **Feedback:** You mention UI industry standards, etc., but what about adherence to things like HIPAA?  
**Source:** TA  
**Changes Made:** Added checking adherence to HIPAA to relevant test cases: FR-ST-DSC3: Prevent PII Storage in Database, SR-ST-P1: Privacy and Data Protection Law Compliance Test, and CR-ST-D1: HIPAA-Compliant Secure Data Storage Test.  
**Issue Link:** [#293](#)
- **Feedback:** Test cases have no clear name, description, or overall goal for each one, which makes it confusing to understand what each test is actually trying to accomplish on a bigger-picture level.  
**Source:** Peer Review by abedmohammed  
**Changes Made:** Added names for each test case followed by a concise summary. For example, the first test went from 'FR-ST-A1' to "FR-ST-A1: Select Parent Role for Login", followed by "Purpose: Verifies that selecting the Parent role from the login screen correctly redirects the user to the Parent login page."  
**Issue Link:** [#204](#)
- **Feedback:** The Verification and Validation Team table in Section 3.1 is a bit vague on what some of the roles actually involve.  
**Source:** Peer Review by abedmohammed  
**Changes Made:** Role descriptions were added under the team table for clarification. For example, "SRS Verification: Ensure all functional and non-functional requirements are clear, testable, and traceable in the VnV plan."  
**Issue Link:** [#206](#)
- **Feedback:** Some of the procedures such as PR-ST-SL1 in the test cases could be a bit more specific on how they're to be conducted.  
**Source:** Peer Review by jane-klavir  
**Changes Made:** To make the procedures of test cases more clear, specific details were added to the procedures. For example, "Test and measure feedback response times for various interactions." became "Identify at least

five types of interactive elements (e.g., buttons, form submissions) and measure response feedback time for each using a stopwatch or browser dev tools. Confirm response occurs within SHORT\_PROCESSING\_TIME.”

**Issue Link:** [#221](#)

- **Feedback:** FR-ST-DSC5 aims to verify long-term storage of assessment reports, and long-term storage is at least MAX STORAGE TIME = 7 years. Perhaps for testing purposes, make some small limit for storage time and verify that reports are accessible for all boundaries of that smaller time.

**Source:** Peer Review by jane-klavir

**Changes Made:** Edited the test case FR-ST-DSC5: Retain Reports Over Time by Patient ID to verify long-term storage by simulating a shortened retention period and confirming report accessibility at various points throughout the time period instead of testing with the max storage time of 7 years, which is an unrealistic testing timeframe.

**Issue Link:** [#222](#)

- **Feedback:** The objectives are vague. What does it mean to ”accurately” and ”correctly” record the assessment? What does it mean for the UI to be ”user-friendly”?

**Source:** Peer Review by marmanios

**Changes Made:** Edited the objectives section to describe objectives in more specific detail. For example, an old objective ”verify the system’s ability to accurately record and store video and audio of clients during speech therapy assessments” becomes ”The system should capture audio and video that meet minimum quality thresholds, such as resolution, clarity, and synchronization between video and audio streams, as defined in performance requirements (Section 4.2.3).”

**Issue Link:** [#223](#)

- **Feedback:** In the NFR section of the test cases there is no ”Test Case Derivation” section but the FR had this section in the test case.

**Source:** Peer Review by Harshil-P21

**Changes Made:** Test case derivations were added for each non-functional requirement test case, based on the functional requirement ones. For example, ”Test Case Derivation: This test ensures that the user interface design meets key look-and-feel requirements by validating navigation simplicity, ease of use, and child accessibility. It covers constraints on UI depth and option count (LF-AR1), intuitive navigation and task flow (LF-AR2, LF-AR4), and confirms that the platform is usable without adult guidance by children aged 6–12.” was added under LF-ST-LFR1: Navigation Simplicity and Child Usability.

**Issue Link:** [#224](#)

### 1.3.2 VnV Report

- **Feedback:** Look and Feel (and this is a general note too): In Actual result, point to the actual data instead of just stating that it passes.

**Source:** TA

**Changes Made:** Actual results for the Look and Feel test cases have been edited to point to specific data supporting the expected and actual output match. For example, "VERY\_HIGH\_SUCCESS\_RATE of users can complete all core tasks independently" was changed to "5/5 of tested users can complete all core tasks defined during usability testing independently".

**Issue Link:** [#617](#)

## 2 Challenge Level and Extras

### 2.1 Challenge Level

The challenge level for this project was classified as **general**. The scope was limited in terms of the research required, and the domain knowledge needed was primarily focused on basic full-stack web development using a technology stack of our choice. Open-source tools and libraries, such as MediaPipe and Deepgram, were leveraged for audio and video processing. The project involved designing and building a system from scratch, with specific emphasis on user experience, accessibility, and secure data handling in a telehealth setting.

### 2.2 Extras

Two approved extras were tackled and completed as part of this project:

- **Usability Testing Report:** Usability testing sessions were conducted with representative users of the system, including parents, children and clinicians. Feedback was collected and analyzed to improve both the visual design and the functional flow of the application. Iterative design improvements were made based on this feedback to enhance intuitiveness, clarity, and accessibility.
- **Developer Documentation Guide:** A comprehensive developer guide was created to support future maintainers and contributors. It included setup instructions, system architecture explanations, technology stack breakdowns, service responsibilities, and customization notes for key components like the bias detection feature.



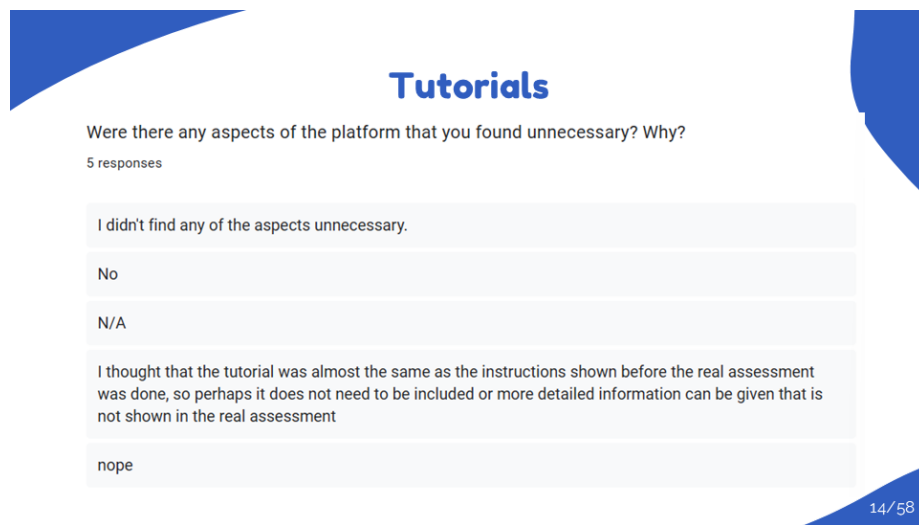
### 3 Design Iteration (LO11 (PrototypeIterate))

The system went through 2 rounds of usability testing between revision 0 and revision 1. Usability testing was completed with both children and parent participants, as well as clinicians. This allowed the team to get real feedback from its potential users, providing feedback to enhance the system overall, and make improvements to make their lives easier.

Following each usability testing scenario, they filled out Google Forms, to provide additional feedback.

#### 3.1 Parents Interface

Figure 1 depicts the feedback regarding tutorials. Figure 2 depicts revision 0's tutorials pages. These tutorial pages contained screenshots of the original system, which many users found confusing. As a result, the team removed the screenshots and decided to focus more on assessment specific tutorials, which is emphasized in revision 1's tutorial system.



The screenshot shows a Google Form titled "Tutorials" with the question "Were there any aspects of the platform that you found unnecessary? Why?". It indicates that there are 5 responses. The responses are listed in a table-like format:

I didn't find any of the aspects unnecessary.
No
N/A
I thought that the tutorial was almost the same as the instructions shown before the real assessment was done, so perhaps it does not need to be included or more detailed information can be given that is not shown in the real assessment
nope

In the bottom right corner of the form, it says "14/58".

Figure 1: Tutorials Feedback

# Tutorials

## Test Selection Tutorial

To begin the assessment, you will be prompted to select your language and test type, which was provided by your clinician via email.

Test Selection

Select Test Language

Select a language

Select Test Type

Select a test type

Next

Available Options

Language Type: English, Mandarin

Test Type: Matching, Repetition, Quantifiers

Next →

## Media Testing Tutorial

### Consent Form

During the assessment, you will be required to review and complete a consent form. This ensures you understand how your data will be used. Please read the provided information carefully when prompted.

Recording Consent

We kindly ask for your consent to record video and audio. This recording may be used for analyzing responses and ensuring the quality of the assessment. Your data will be handled securely, only viewable by your clinician and used solely for the purpose outlined above.

Do you consent to the recording of video and audio for this assessment?

I consent

I do not consent

After completing the consent form during the assessment, you will be guided to configure your video, audio, and microphone devices. This step ensures that all devices are working correctly before you proceed.

15/58

Figure 2: Tutorials Changes

Figure 3 depicts the feedback parents provided with regard to bugs, which mainly focused on the submission process. Figure 4 depicts on the left side the original submission interface, and the right side depicting the new submission interface, which contains an animated loading wheel. This provides active information to users, reducing confusion, and showing that there is a process going on.

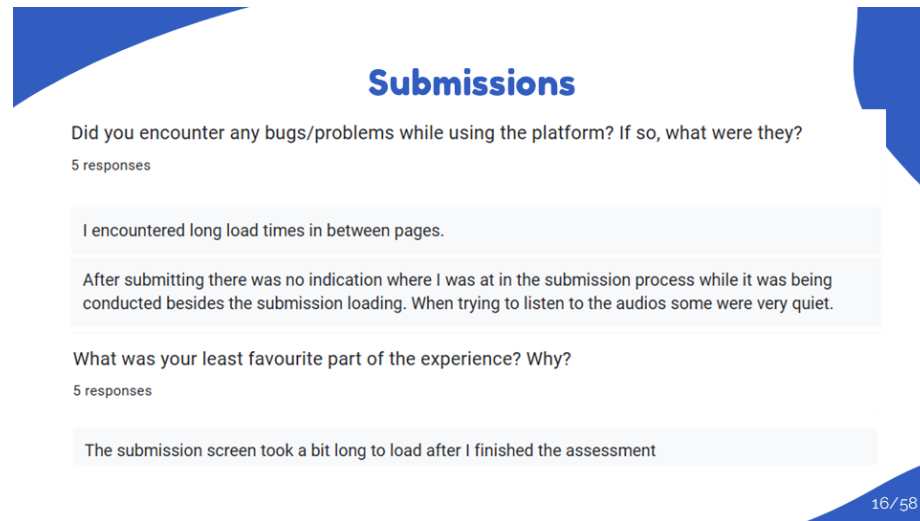


Figure 3: Submissions Feedback

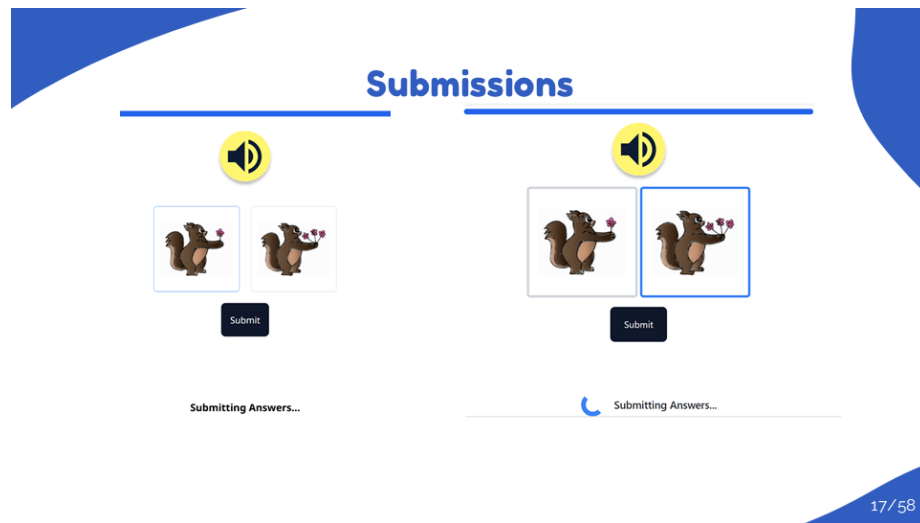


Figure 4: Submissions Changes

Figure 5 depicts the feedback parents provided with regard to the homepage and having a difficult time finding the tutorial button. Figure 6 depicts the original home page, which, if the screensize didn't scale accordingly, the tutorial button required scrolling to access. Figure 7 depicts the updated home page, which now has the tutorial button moved to the right side of the screen, which is much more apparent for users.

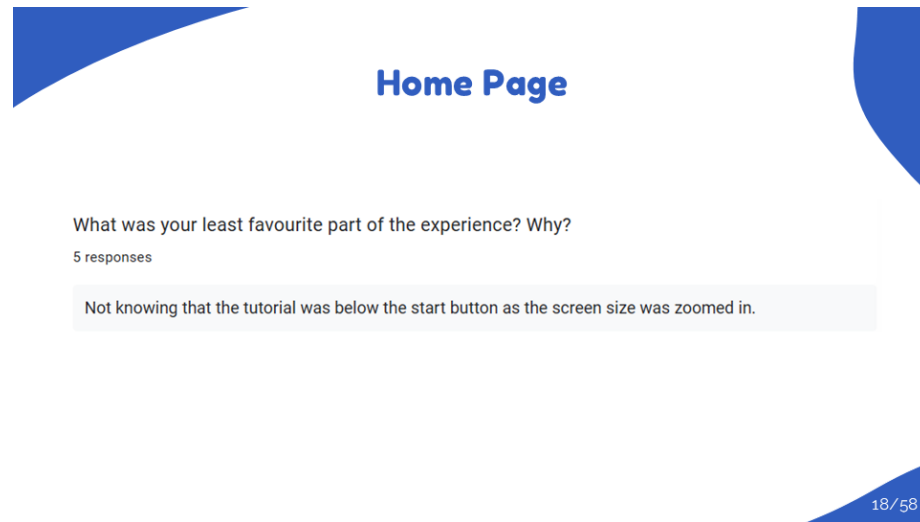


Figure 5: Home Page Feedback

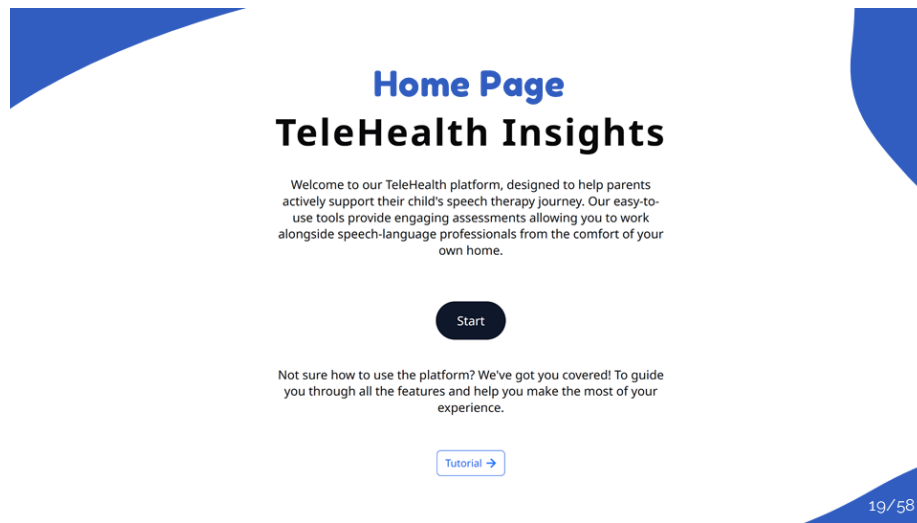


Figure 6: Original Home Page

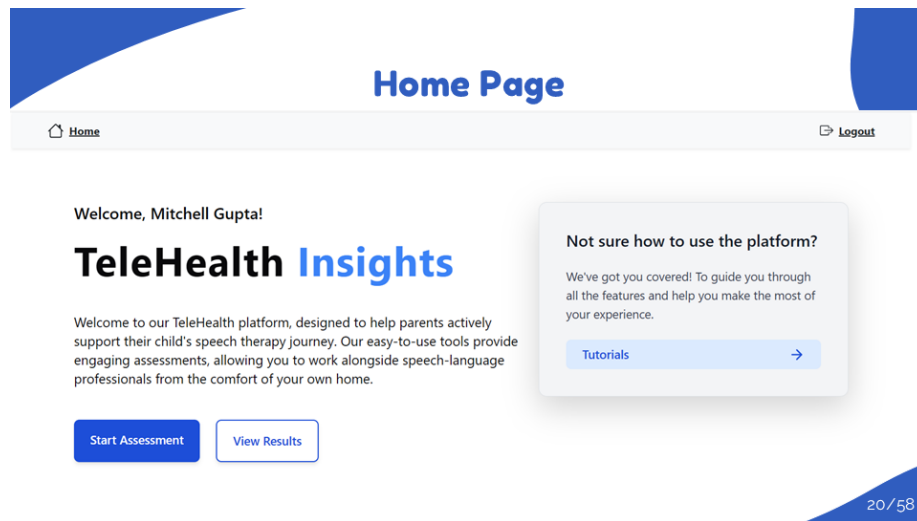


Figure 7: Updated Home Page

Figure 8 depicts the feedback parents provided with regard to the system not having clear instructions on how to approach next steps. Throughout testing, the team recognized this repeatedly came up during repetition assessments, in particular. Figure 9 depicts the new repetition assessments modifications. The buttons are greyed out to prevent unintentional paths from being explored, and to ensure that users go through the correct sequence of buttons to complete an assessment.



Figure 8: Repetition Assessments Feedback



Figure 9: Repetition Assessments Changes



### 3.2 Usability Test Results

Figure 10 depicts the overall feedback with regard to 7 prompts, which are listed in the figure. Most of the feedback was positive, with the exception of 'All the button interactions reacted and responded how I thought they should', which was addressed in Figure 9.

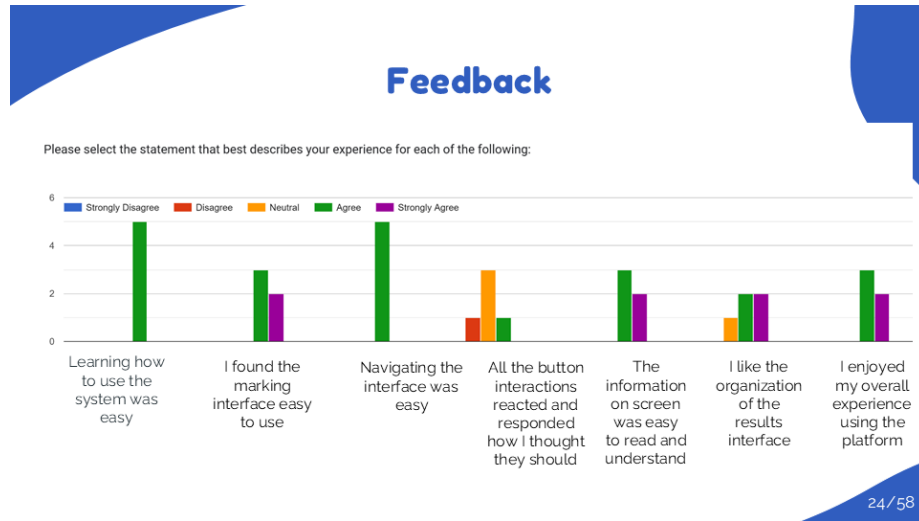


Figure 10: Parents Usability Tests Results

Figure 11 depicts the feedback parents provided with regard to the system's ability to scale the visuals appropriately. Given 40% of parents indicated the screen's visuals did not scale appropriately, the system's parent side was redeveloped to better adapt to screensizes, so that the system can be used regardless of screensize or device. The team's supervisor indicated that their testing with children demonstrated how many children preferred to take the assessments on tablets, rather than computers. The updated interface, depicted in Figure 12 shows how the interface can resize for both tablets (left) and computers (right).

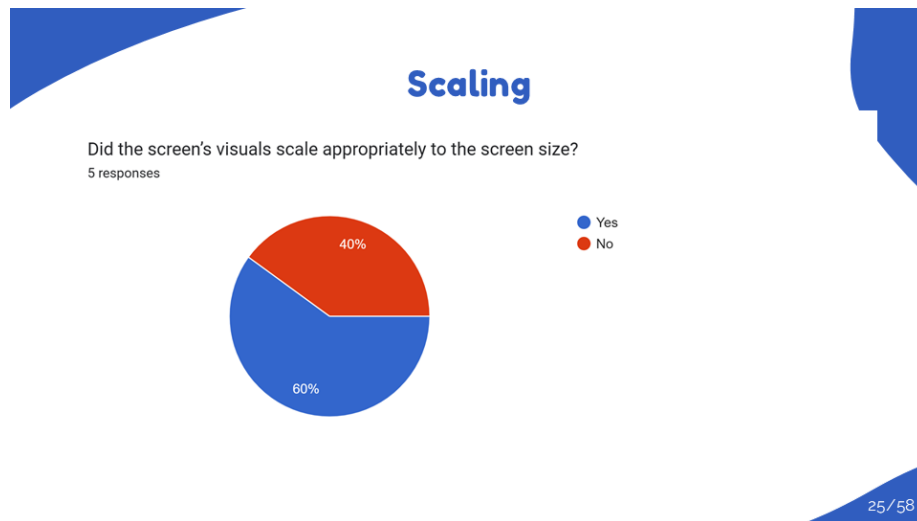


Figure 11: Scaling Feedback

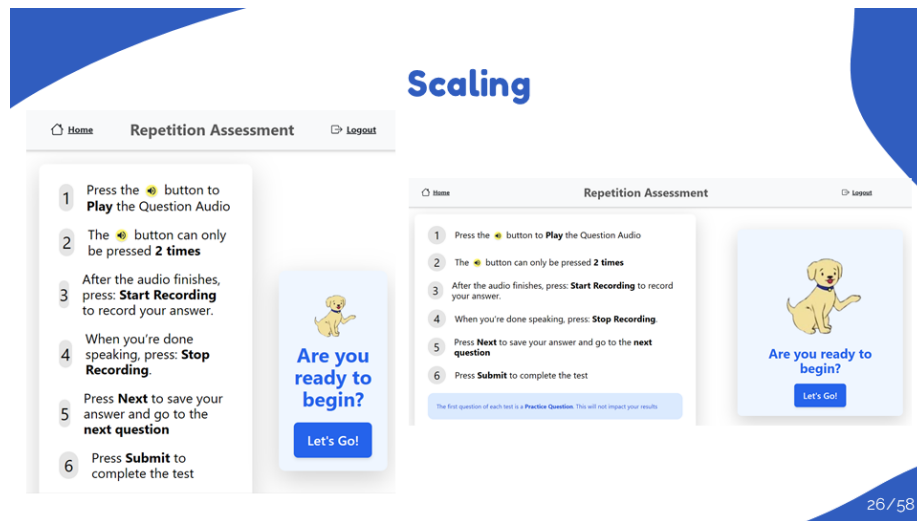


Figure 12: Scaling Changes

Figure 13 depicts how the matching assessments were a favourite amongst parents and children. Upon speaking with the team's supervisor, the team was instructed to make the visuals even larger so that children were more engaged with the system, and it would be even easier to select the options.

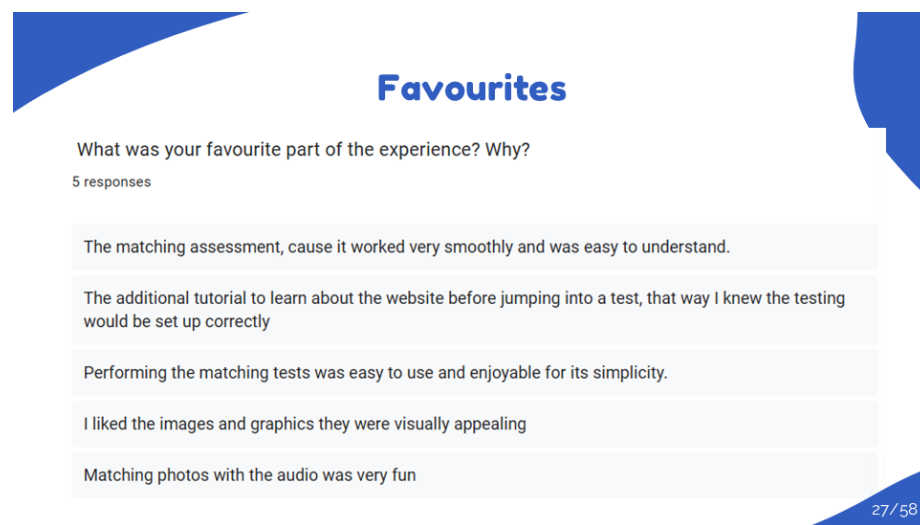


Figure 13: Parents Favourites

### 3.3 Clinicians Interface

Figure 14 depicts the feedback clinicians provided with regard to the system lacking a reminder to give the client ID to parents when they create an account. This feedback was common amongst clinicians during the usability testing sessions, with many unsure about what to do with the security code (referred to in the figure by client ID). Figure 15 depicts the modified 'Add New Client' feature, which contains a reminder to note down the code and provide it to the client so they can create their account. This change improved the systems ease of use by directly communicating with the clinician, so they don't have to make the mental note themselves to note down the security code.

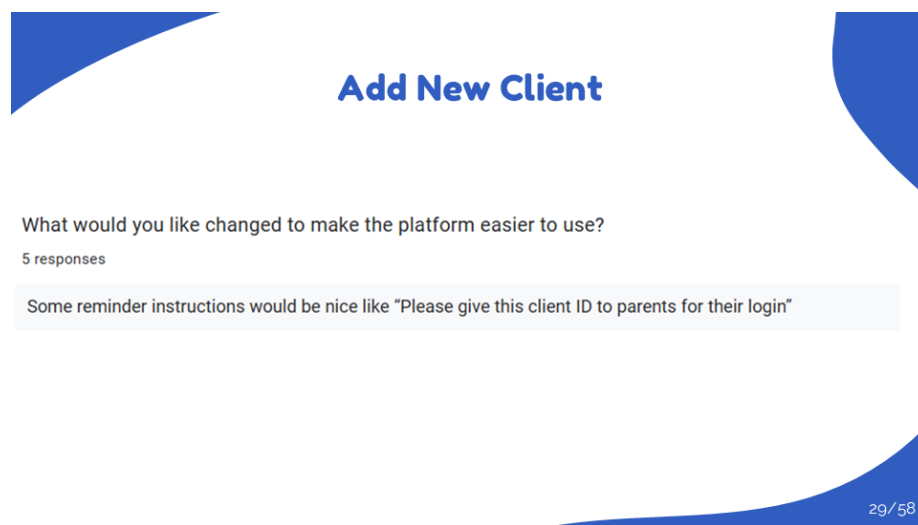


Figure 14: Add New Client Feedback



Figure 15: Add New Clients Changes

Figure 16 depicts the feedback clinicians provided with regard to the dashboard. The dashboard was a favourite amongst the majority of clinicians, particularly the graph and trends. The team decided to review the dashboard to see how we could improve it even more. The original dashboard, depicted in Figure 17 had repetitive labels on the x-axis (restating Test number for each data point). As well, many clinicians missed the legend because it was below the graph. In response to this feedback, the team decided to move the legend into a box above the graph and include an x-axis label for 'Attempt', depicted in Figure 18. In addition, the team wanted to give clinicians the ability to filter assessments so they could access the information that mattered the most to them upon reviewing the assessment information. Therefore, the filters above the graph were added, which also affects the assessment results displayed to the right of the graph.

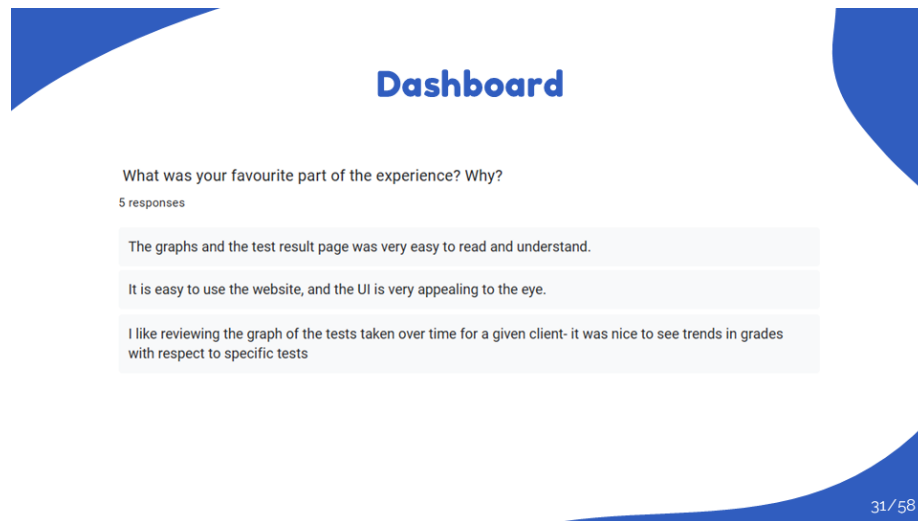


Figure 16: Dashboard Feedback



Figure 17: Dashboard Original

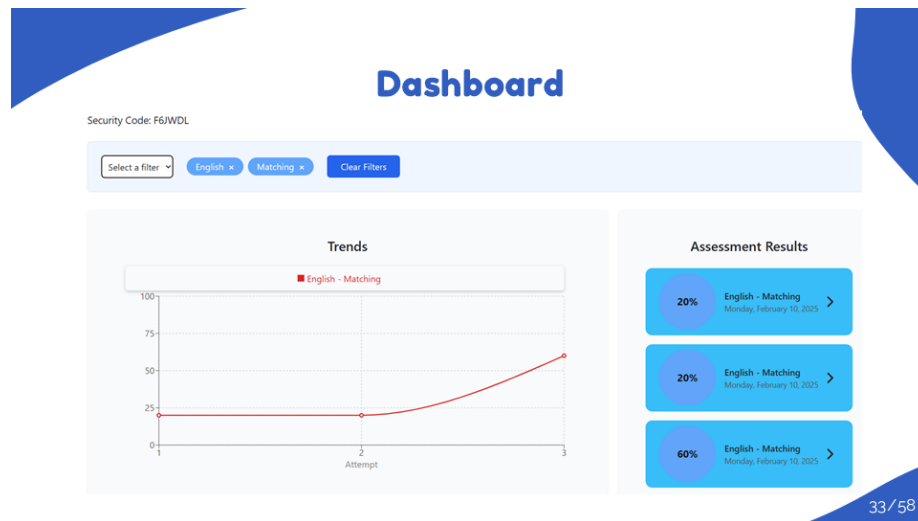


Figure 18: Dashboard Updated

Figure 19 depicts the feedback clinicians provided with regard to the bias review pages. One clinician requested the addition of a 'bias note' feature. Another clinician requested a more convenient way to navigate between pages in an assessment. Figure 20 depicts the original Bias Review page, which also featured a 'save changes' button. Many clinicians noted, during testing, that clicking an additional save changes button was inconvenient, and they preferred to see changes applied immediately. All these changes led to the implementation of the updated Bias Review page, depicted in 21. The updated page removed the 'save changes' button and saves changes immediately. As well, page navigation is made conveniently through the page to page navigation buttons in the top right. Further, clinician notes were added so that clinicians can add their own notes and store them with the questions. A major design change as well was moving all the bias-related information into its own panel on the right side of the question information, so that similar information was grouped together, improving usability.

## Bias Review

What would you like changed to make the platform easier to use?

5 responses

Adding a bias note so that the clinician is able to add their own bias that they discover.

Which part of the platform needs the most improvement? Why?

5 responses

If i had to choose maybe the page to page navigation, using the home button sometimes is inconvenient when you don't want to head back to the home page.

34/58

Figure 19: Bias Review Feedback

## Bias Review

**Bias Detected**

Question timestamp: 00:00

Question ID: 1

User Answer

A B

Correct Answer: B

Hide Bias List

**Bias Detected:**

- 3.65s - choose (Faces Detected: 2)
- 5.22s - one (Faces Detected: 2)
- 8.52s - this (Faces Detected: 2)
- 10.09s - one (Faces Detected: 2)

Save Changes

35/58

Figure 20: Bias Review Original



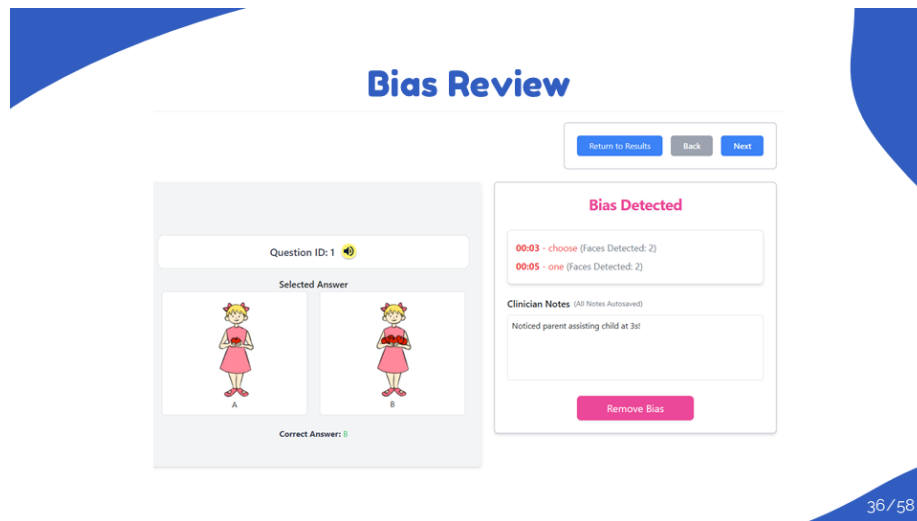


Figure 21: Bias Review Updated

Figure 22 depicts the feedback clinicians provided with regard to the grading system for repetition assessments. As depicted in Figure 23, the original Repetition Assessments Bias Review page also contained a 'save changes' button. The team wanted to improve clinicians' abilities to grade questions efficiently and effectively. This was done by including a 'Submitted Answer' audio player, so that clinicians don't have to search through the video to find a user's submission, as depicted in Figure 24.

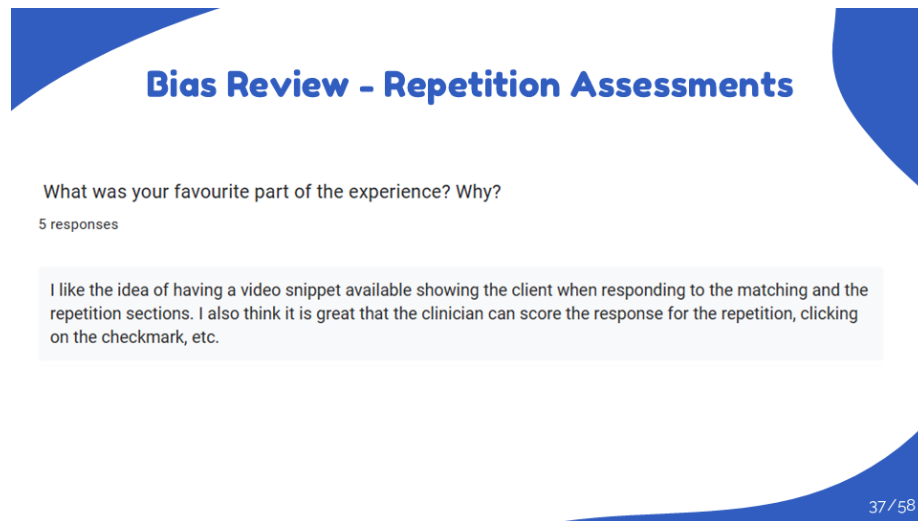


Figure 22: Bias Review - Repetition Assessments Feedback

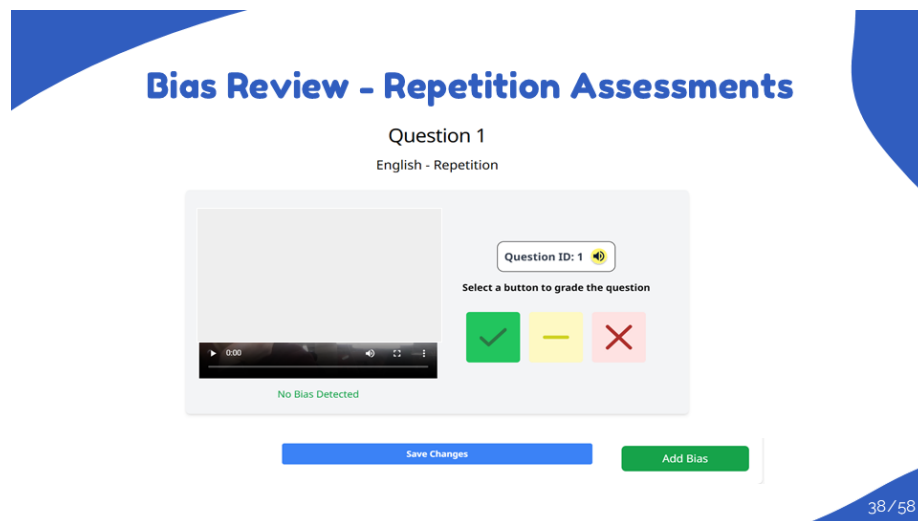


Figure 23: Bias Review - Repetition Assessments Original

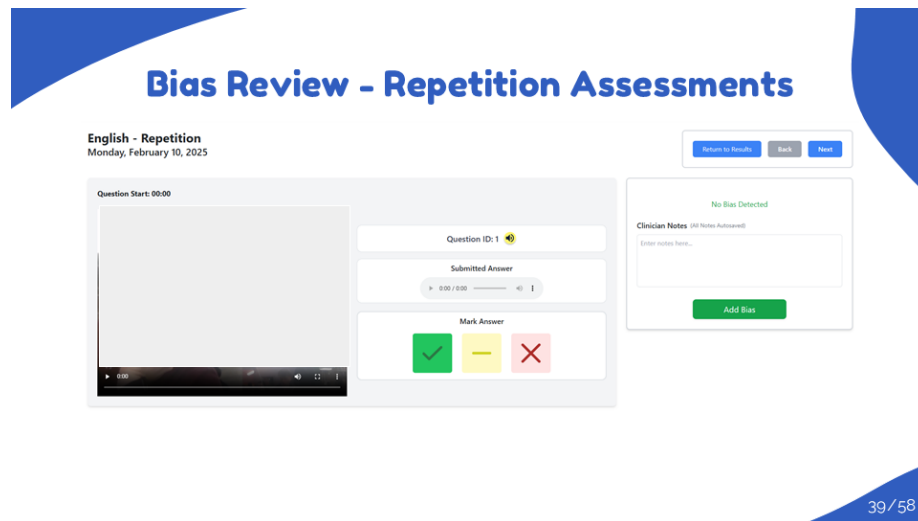


Figure 24: Bias Review - Repetition Assessments Updated

### 3.4 Usability Test Results

Figure 25 depicts the overall feedback with regard to 7 prompts, which are listed in the figure. Majority of the feedback was positive, with the exception of 1 neutral response for 'I like the organization of the results interface'.

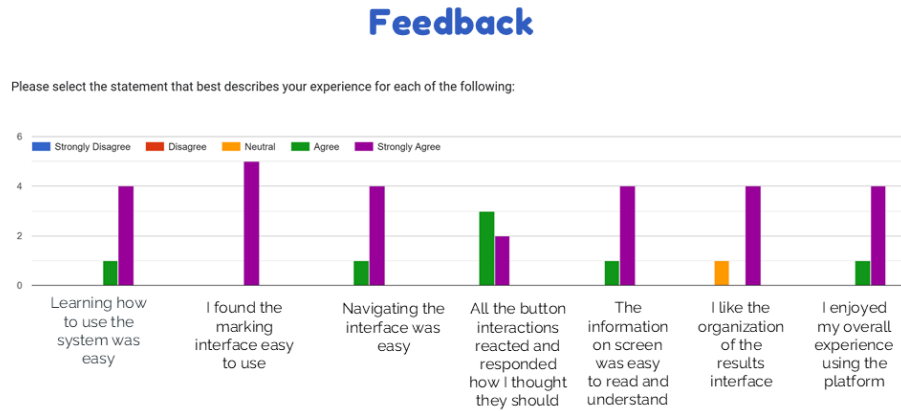


Figure 25: Clinicians Usability Tests Results

Figure 26 depicts how difficulties were minimal. Figure 27 depicts how there was no least favourite experience with the clinician interface. Figure 28 depicts how there were no aspects of the platform clinicians found unnecessary. Figure 29 depicts how the clinician interface scaled appropriately 100% of the time. Figure 30 depicts how clinicians were excited to use the platform and thinks it will be a beneficial tool.

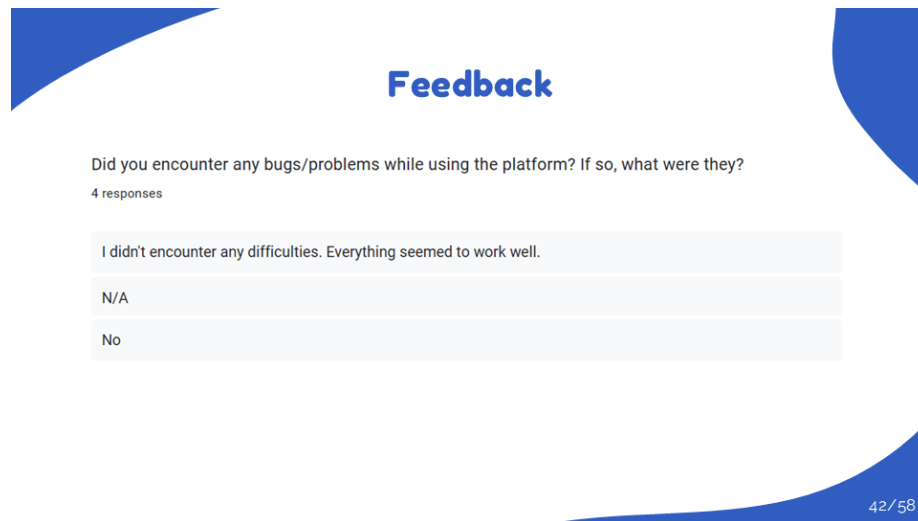


Figure 26: Clinician Feedback - 1

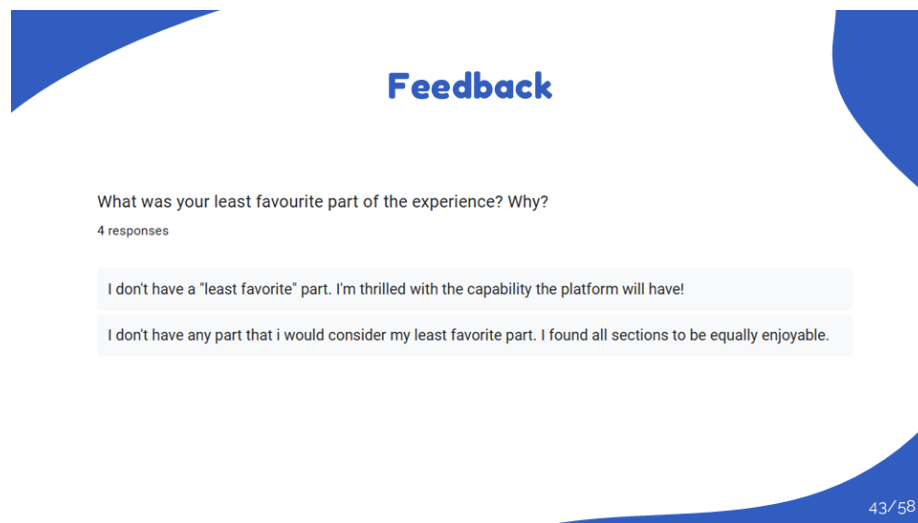


Figure 27: Clinician Feedback - 2



Figure 28: Clinician Feedback - 3

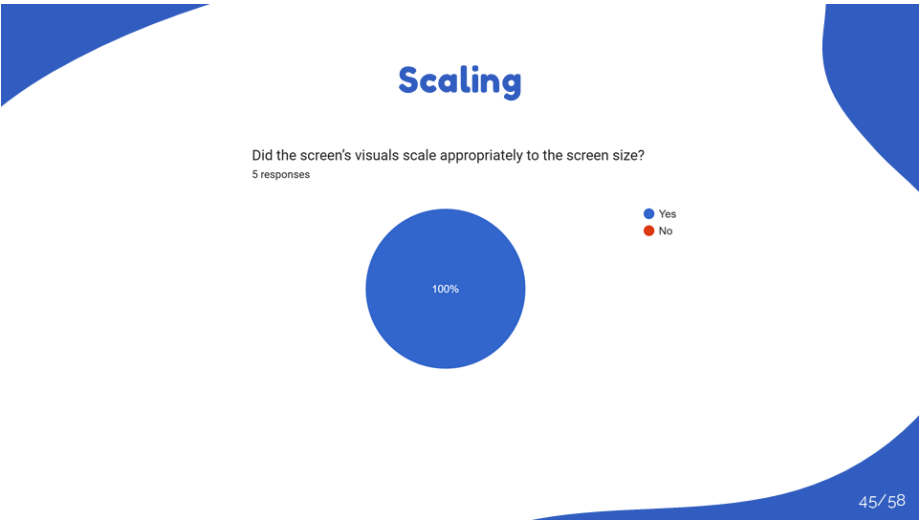


Figure 29: Scaling

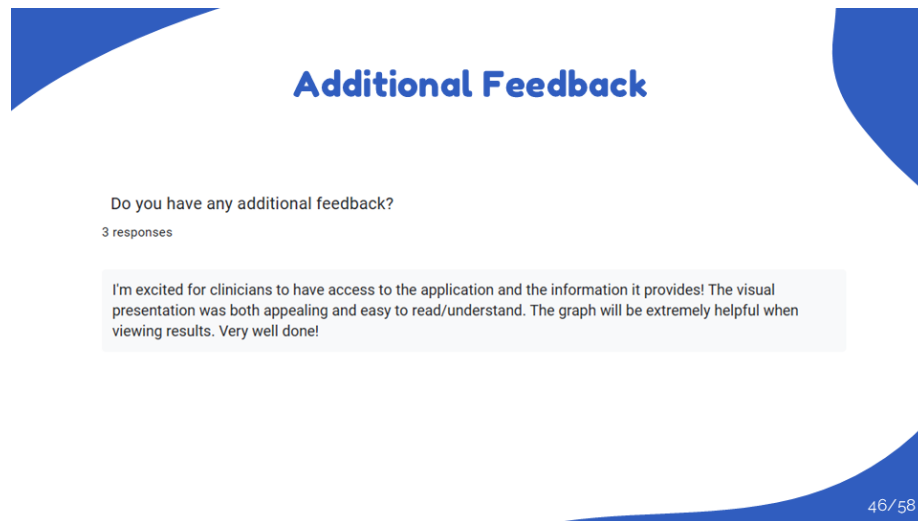


Figure 30: Additional Feedback

## 4 Design Decisions (LO12)

### 4.1 Limitations

We had a tight timeline, so we chose an architecture we could build quickly in terms of, all of us could work on it without blocking each other, which is why we decided to go with the microservice architecture design for our backend. Our backend had to be deployed on a server with limited speed (due to financial limitations), which pushed us to use asynchronous processing so children don't have to wait for slow backend processing. These time and server limitations guided us to focus on only the essential features first, rather than building a large or overly complex system. It also greatly helped our scope of the project, as due to the time limitations we had to be mindful of not being too ambitious.

### 4.2 Assumptions

We assumed that users have reliable internet connections, can record audio and video, and that assessments would not exceed a certain amount of time. We also assumed the platform would grow over time, so we designed it to be scalable by using a microservice architecture. On the front end, we made it dynamic with an abstract factory pattern to generate our assessments, so adding or modifying quizzes would be simpler as the assumption is that the platform will grow.

### 4.3 Constraints

We built the system as a web platform to meet the requirement of easy access without extra software. We made sure to consider HIPAA, PHIPA, and PIPEDA rules, which meant enforcing secure logins and data storage which was done through role base access and bycrypt encryptions. The platform also had to handle enough users at once without slowing down, so made sure our deployment would have options to speed up the servers if the future developers choose so. We included support for different video and audio settings based on bandwidth, and we followed WCAG 2.1 standards for accessibility. Lastly, legal rules about keeping patient records for a set number of years influenced how we designed our data storage and archiving processes, so that removing data would be very easy and our system wouldn't crash if an assessment is not found, as it calls the API every time when refreshing the page.



## 5 Economic Considerations (LO23)

There is a growing market for TeleHealth Insights, especially as the demand for remote speech-language services continues to rise. According to the U.S. Bureau of Labor Statistics, employment opportunities for speech-language pathologists are projected to grow by 18% from 2023 to 2033 [1]. Additionally, the cultural and educational impact of the COVID-19 pandemic has accelerated the demand for at-home, tech-enabled solutions, which offer both convenience and accessibility for families and clinicians.

Currently, our platform is open-source and publicly available on GitHub, with no immediate plans for monetization. If we were to transition to a commercial version in the future, we estimate development and compliance costs would total approximately \$30,000–\$50,000 CAD. A possible pricing model could involve charging clinics a subscription of \$99–\$149 per month. At \$120/month, we would need around 21–35 clinician users for one year to break even. While we currently focus on open access, these numbers suggest commercial viability is achievable with a small professional user base.

To attract users, we could pursue collaborations with universities and leverage our relationship with our client to connect with a wider network of speech-language pathologists for use in clinical studies and research. Other ways to attract users include creating demo videos and case studies that showcase the platform in real-world settings. Additionally, we will maintain comprehensive documentation and tutorials on GitHub to support developers and researchers who wish to adapt or extend our solution.

There is a significant potential user base for this platform. If we were to focus solely on speech-language pathologists and therapists, there were approximately 18,300 SLPs in Canada [2] and 172,100 in the United States as of 2023 [3]. The need for this type of tool is increasing, as research estimates that approximately 7.5% of school-aged children have some form of language disorder [4].

## 6 Reflection on Project Management (LO24)

The processes and tools used for project management are emphasized in the questions below.

### 6.1 How Does Your Project Management Compare to Your Development Plan

#### 6.1.1 Team Meeting Plan

As outlined in the Development Plan, the team met in person every Monday from 3:30-4:30 PM throughout both the Fall and Winter semesters. Additional meetings were held as needed, either in person or virtually through the team's Capstone Discord server—particularly as major deliverable deadlines approached.

In-person team meetings were typically held in Thode Library and followed a structured format:

- Progress check-in (5-10 minutes)
- Agenda debrief (5 minutes)
- Work session or discussion (40 minutes)
- Planning next steps (5-10 minutes)

Supervisor meetings with Dr.Irene Yuan were held in person at ABB every Tuesday from 9:45-10:15 AM during the Fall term, and every Wednesday from 12:45-1:30 PM during the Winter term. Exceptions to the meeting schedule were made when the university was closed due to holidays or by group consensus.

### 6.1.2 Team Communication Plan

Communication was a key factor in the success of our project, and our team used a combination of platforms to stay organized and connected throughout the development process.

**Discord** was our primary platform for day-to-day communication. The server was organized into structured text channels to keep discussions focused and easy to navigate:

- **#general** - Used for general updates, announcements, and quick questions.
- **#documentation-and-resources, #important-documents, #initial-ideas** - Shared references, documentation drafts, and brainstorming content.
- **#meetings, #supervisor-meetings, #schedule** - For coordinating both internal team meetings and meetings with our supervisor.
- **#tasks, #accounts, #adminscript** - For task assignment, account tracking, and scripts used for admin purposes.
- **#frontend, #backend** - Discussions and debugging related to each code-base.
- **#stand-up-updates** - Asynchronous team updates and progress logs.
- **#help** - A collaborative support space for asking technical questions or offering help.
- **#usability-tests, #wont-fix** - Specific discussion threads around user testing feedback and deprioritized issues.

**GitHub** was used for:

- Code versioning through pull requests and feature branches.

- Project tracking using the GitHub Projects board (Kanban-style) with columns for *Backlog*, *To Do*, *In Progress*, *In Review*, and *Done*.
- Issue tracking and task assignment, using milestones, labels, and assignees for structured coordination.

**Email** was used for formal communication with individuals outside the development team, including our supervisor, course instructors, and external professionals.

### 6.1.3 Team Member Roles

The main roles from development plan stayed the same however assignments did change. While all members contributed to documentation, development, testing, and issue tracking, the following specialized responsibilities were maintained throughout the project:

- **Mitchell Weingust – Project Manager** Chaired team meetings, managed the overall project timeline, ensured deadlines were met, and kept the team aligned, organized and on track for all deliverables. Coordinated communication amongst the team’s TA and the course instructor. Responsible for leading the design decisions of the clinician interface.
- **Parisha Nizam – Team Liason & Frontend Lead:** Chaired external meetings and managed communication with the capstone supervisor, collaborators and any other external stakeholders. Also led the frontend development.
- **Promish Kandel – Lead Developer & Machine Learning Video/Audio Lead:** Led the technical design and coordination of the system. Oversaw backend architecture and implementation while also managing the video analysis component of the bias detection system.
- **Jasmine Sun-Hu – UI/UX Design Lead:** Responsible for leading the design decisions of the parents’ and children’s interface, along with overall user experience.

### 6.1.4 Workflow Plan

Our team used GitHub Issues as the core of our workflow. All tasks—including features, bugs, documentation, and meetings; were tracked as individual issues. Each issue included labels, milestones, assignees, and detailed checklists for sub-tasks. Instead of using GitHub Projects, we managed progress directly through labeled issues and milestone tracking. Pull requests were tied to issues and required review before merging. GitHub Actions handled CI workflows, enforcing linting and running tests. A rollback strategy was included to ensure safe and stable deployments.

### 6.1.5 Expected Technologies

The following tools and technologies were used throughout the project, as originally planned:

- **JavaScript** - Main programming language for both frontend and backend development. Widely supported and efficient for full-stack development.
- **Python** - Used for machine learning and data analysis components. Ideal for prototyping with strong library support.
- **React & Tailwind** - Used for building a responsive frontend interface with fast development and styling via utility-first CSS.
- **Node.js & Express** - Powered the backend server with non-blocking I/O, ideal for real-time operations and REST API integration.
- **MediaPipe** - Employed for facial detection and video processing in the bias detection module. Provided pre-trained models and reliable computer vision utilities.
- **Figma** - Used for collaborative UI/UX design and wireframing before implementation.
- **VSCode** - Primary code editor, offering powerful extensions and built-in Git integration.
- **GitHub & GitHub Actions** - Used for version control, task tracking, and automated CI/CD workflows including linting and testing.
- **ESLint & Prettier** - Enforced consistent code style and formatting across the team.
- **Jest & Pytest** - Testing frameworks for JavaScript and Python codebases respectively, enabling robust unit and integration tests.
- **AWS S3** - Provided secure cloud storage for media files, questions, and results.
- **Netlify** - Used to deploy the frontend with Git integration and automatic builds.
- **Render** - Hosted the backend services with automated deployment and SSL support.
- **Deepgram** - Used for real-time speech-to-text transcription and keyword detection in the bias detection system.

## 6.2 What Went Well?

Our team’s project management went well in several areas, particularly in coordination, organization and documentation. The team consistently attended weekly in-person meetings with the capstone supervisor as outlined in the development plan, and supplemented those with additional ad hoc meetings both virtual and in person as necessary when deadlines were approaching. This momentum helped ensure consistent progress throughout the project. The team also made effective use of Discord for daily communication; the structured channel layout made it easy to keep track of and to have discussions on development, documentation, testing, usability and other topics simultaneously. Likewise, GitHub was used often, especially for issue tracking and task management. The use of labeled issues, milestones, assignees, and pull requests enabled smooth coordination, accountability, and fairly distributed workloads within the team. In addition, our team adhered closely to our chosen technologies from the development plan listed in the previous section.

## 6.3 What Went Wrong?

Our team’s communication could have been a bit more consistent. There were times the team was unsure about which issues were already being worked on by other team members, and times when it would look like team members made minimal progress, when in reality, they just had larger commits and would complete issues all at once. This led to the team being unsure about the progress of different checkpoints and milestones. However, upon meeting with the team, we discussed strategies to try to resolve this issue, such as posting daily updates, and showing progress screenshots of work. This took place later in the development process.

Another thing that went wrong along the way was the modification and changes of requirements throughout the development process. There were times in meetings with the team’s supervisor and the external stakeholder (Dr. Du) when requirements would change, and the team would need to quickly adapt. However, the fast-paced development cycle, along with balancing the team’s schedules made it difficult to complete everything requested. As a result, the team had to prioritize which features were necessary, and which features were ‘nice to haves’. Most items discussed were implemented, but given more time, the team would’ve wanted to add more features to further enhance the system.

## 6.4 What Would you Do Differently Next Time?

If we had to do this project again, the main thing we would do differently is connecting our design to code and doing a bit of research before making our final design. We found that there was a disconnect between our design and the correct way of making modules into code and how to connect our frontend to backend. A good example of this is the app controller, as we first thought we

could route all API calls from the clinician and parent GUI through one module. However, when coding, we found that this would be difficult and serve very little purpose. We also think we spent too much time thinking about all the modules we could have versus the modules we needed. You can see this between our rev 0 module hierarchy in the MG document to rev 1; half of the backend modules are gone as we didn't need them. Finally, we would spend more time solidifying the requirements, and we found that there were times that new requirements were being added, which made coding that much harder. Overall, these are the two main areas that we would focus on for future projects and into the workforce.

## 7 Reflection on Capstone

The following sections focus on what we learned during the course of the capstone project.

### 7.1 Which Courses Were Relevant

The following Software Engineering courses were relevant for our capstone project:

- **SFWR ENG 2AA4 - Software Design I:** Provided foundational knowledge in modular design and object-oriented programming, which helped with structuring the frontend and backend codebases.
- **SFWR ENG 3A04 - Software Design III:** Built upon design principles from 2AA4 and introduced patterns and architecture used in larger systems, which helped inform our microservice design and API structure.
- **SFWR ENG 3RA3 - Software Requirements:** Guided our approach to defining stakeholder needs, use cases, and system requirements, all of which were critical in shaping our initial documentation and goals.
- **SFWR ENG 3S03 - Software Testing:** Provided valuable knowledge on creating meaningful and maintainable test cases. This was directly applied to our testing strategy using Jest and PyTest.
- **SFWR ENG 4HC3 - Human Computer Interfaces:** Strongly influenced the design and usability of our parent, clinician, and child interfaces. Concepts like accessibility, minimal cognitive load, and user engagement shaped many UI/UX decisions.

### 7.2 Knowledge/Skills Outside of Courses

To build and deploy a full-stack system, we had to learn several skills and technologies not covered in our course curriculum:

- **Web Development:** We learned how to develop modern web applications using tools like React, Tailwind CSS, Vite, Node.js, and Express.js.
- **Cloud Infrastructure and Deployment:** We gained experience deploying applications using Netlify (frontend), Render (backend), and AWS S3 (cloud storage). This included managing environments, handling build pipelines, and ensuring secure data access.
- **Media and Machine Learning Integration:** Integrating real-time audio and video processing with Deepgram and MediaPipe required understanding how to interface with external APIs and handle client media streams effectively.
- **CI/CD Pipelines:** Although testing was covered in class, setting up automated pipelines using GitHub Actions was a new skill we had to learn.

## References

- [1] U.S. Bureau of Labor Statistics, “Speech-Language Pathologists: Occupational Outlook Handbook,” *U.S. BLS*, <https://www.bls.gov/ooh/healthcare/speech-language-pathologists.htm> (accessed Apr. 1, 2025).
- [2] Government of Canada Job Bank, “Speech-language pathologist in Canada,” *Job Bank*, <https://www.jobbank.gc.ca/marketreport/outlook-occupation/22734/ca> (accessed Apr. 2, 2025).
- [3] U.S. Bureau of Labor Statistics, “Occupational Employment and Wages, May 2023: 29-1127 Speech-Language Pathologists,” *U.S. BLS*, <https://www.bls.gov/oes/2023/may/oes291127.htm> (accessed Apr. 2, 2025).
- [4] A. M. Tomblin, “Epidemiology of Specific Language Impairment,” in *Neurobiology of Language*, G. Hickok and S. L. Small, Eds., Academic Press, 2016, pp. 111–122. <https://www.sciencedirect.com/science/article/pii/B9780444641489000028> (accessed Apr. 2, 2025).