# Software Requirements Specification
# ASLingo Application

Team 15, ASLingo

Andrew Kil

Cassidy Baldin

Edward Zhuang

Jeremy Langner

Stanley Chan

October 6, 2023

# Contents



# List of Tables

# List of Figures

Table 1: Revision History

| Date | Developers | Change |
|---|---|---|
| September 25, 2023 | All team members | Initial draft, added some functional requirements |
| September 26, 2023 | Andrew Kil | Added constraints and naming conventions |
| September 26, 2023 | Cassidy Baldin | Added some functional and non-functional requirements |
| September 27, 2023 | Jeremy Langner | Added some points to section 4 Project Issues |
| September 28, 2023 | Cassidy Baldin | Added NFR tables, some rationales for NFRs |
| September 28, 2023 | Andrew Kil | Added abbreviations and assumptions |
| September 28, 2023 | Stanley Chan | Added scope and context of work, some work partitioning events |
| September 28, 2023 | Edward Zhuang | Added to Section 4 and some FR rationales |
| October 2, 2023 | Cassidy Baldin | Edited Section 3 requirements, finished all rationales |
| October 4, 2023 | Stanley Chan | Added individual product use cases, updated work partitioning |
| October 4, 2023 | Edward Zhuang | Edited Section 4 according to meeting with MSLC, and added response to one reflection question |
| October 6, 2023 | Cassidy Baldin | Added fit criterion, fixed formatting, added reflection |
| October 6, 2023 | Stanley Chan | Added individual reflection |
| October 6, 2023 | Jeremy | Added personal reflection response and general reflection question 3. |
| October 6, 2023 | Andrew | Added personal reflection response |

This document describes the requirements for ASLingo. The template for the Software Requirements Specification (SRS) is a subset of the Volere template *Robertson And Robertson (2012)*. Subsections *Clients* and *Customers* were removed due to not having any such dependents.

# 1 Project Drivers

## 1.1 The Purpose of the Project

Learning a new language can be an arduous task that only gets more challenging with age, as individuals may find it difficult to dedicate time and effort to it. American Sign Language (ASL) is particularly hard due to its visual and gestural nature, which is not found in other, verbal languages. The purpose of this project is to ease that challenge by providing an online, easy-to-access web platform for individuals to learn new signs and test their comprehension at their own pace in a fun, interactive manner. Focusing in on consistent effort and continuous feedback, ASLingo provides real-time guidance to ensure users stay on track to achieving their goals of learning ASL.

## 1.2 The Stakeholders

The stakeholders for this project include those who use sign language as their primary mode of communication in daily life as well as those who have an interest in learning ASL. This would naturally expand outward towards educators who wish to promote the learning of ASL to their respective institutions.

## 1.3 Mandated Constraints

The project is constrained by the following:

***The Project Expenses Cannot Exceed $750***

- The project cannot be bought or be a 'ready-made' solution, and any cost incurred must be minimized to ensure cost-efficiency.

***Test Only a Subset of the Most Commonly Used Phrases of ASL***

- The project in the long-term would encompass the entirety of ASL as a whole. But with the given time-constraints, the testable language subset will be limited.

***The Project Must be Finished by the End of the Academic Year***

- This is the hard deadline for the project and sets the time constraint that is used to judge the scope of work the project can encompass.

***Users Should Only Learn from the Provided "On Rails" Approach***

– Users are only able to learn from the preset programs the project has. The education provided should be taught in a manner with little room misinterpretation. While we attempt to make the learning content as broad as possible, users lack true learning freedom to learn what they want.

## 1.4 Naming Conventions and Terminology

Table 2: Naming Conventions and Terminology

| **Term, Abbreviation, or Acronym** | **Description** |
|---|---|
| A | Shorthand for Assumption |
| ASL | Shorthand for American Sign Language. It is a form of sign language primarily used in the US and in parts of Canada |
| ASLingo | The commercial name for the project |
| CV | Refers to Computer Vision, the field of technology that involves processing visual input to achieve various means. |
| CR | Shorthand for 'Cultural Requirements', a subsection of Non-Functional Requirements. |
| HSR | Shorthand for 'Health and Safety Requirements', a subsection of Non-Functional Requirements. |
| FR | Shorthand for Functional Requirements |
| LR | Shorthand for 'Legal Requirements', a subsection of Non-Functional Requirements. |
| LFR | Shorthand for 'Look and Feel Requirements', a subsection of Non-Functional Requirements. |
| MSR | Shorthand for 'Maintainability and Support Requirements', a subsection of Non-Functional Requirements. |

| | |
|---|---|
| OER | Shorthand for 'Operational and Environmental Requirements', a subsection of Non-Functional Requirements. |
| OpenCV | Refers to the Open Computer Vision Library library available for free to developers in order to develop Computer Vision applications. |
| PR | Shorthand for 'Performance Requirements', a subsection of Non-Functional Requirements. |
| SR | Shorthand for 'Security Requirements', a subsection of Non-Functional Requirements. |
| UHR | Shorthand for 'Usability and Humanity Requirements', a subsection of Non-Functional Requirements. |

### 1.5 Relevant Facts and Assumptions

1. The user will always have the camera their hands when using the application.

   Correct camera angling towards the user's hands is the first condition for the application to work.

2. The environmental lighting will always be sufficient for joint detection.

   Proper lighting is the second condition for the application to work since it cannot properly distinct what the user signs without it.

3. The user's signs will be within reasonable form of the proper sign, enough to be recognized by the system

   If the user signs are correct but with poor form, the system will have a hard time determining if it is correct.

## 2 Functional Requirements

### 2.1 The Scope of the Work and the Product

The scope of ASLingo can be clearly defined by outlining our primary goals for this product.

1. Hand Sign Recognition: Reliably recognize users' hand sign in real-time based on the American Sign Language.

2. Test Users: Quiz users with sign language based questions.

3. User Progression: Track a user's sign language learning progression.

4. Account Management: Store required user information to allow users to create and login to their accounts.

### 2.1.1 The Context of the Work

The following is a context diagram which describes the high-level overview on how the system will be utilized.


Figure 1: Context Diagram

### 2.1.2 Work Partitioning

Table 3: Work Partitioning

| Event ID | Event Name | Input and Output | Description |
|---|---|---|---|
| 1 | User login | User ID (in)<br>User Password (in)<br>Login Status (out) | User logs into the application, the system determines if login is successful. |
| 2 | User logout | User ID (in)<br>Login Status (out) | User logs out of the application, system indicates whether log out is successful or not. |
| 3 | User requests to start a test | User ID (in) | User starts a test. |
| 4 | User signs through webcam | Camera Feed (in)<br>Recognized Sign (out) | User inputs sign language hand signs through webcam, the system responds with the corresponding sign output. |
| 5 | User creates account | User ID (in)<br>User Email (in)<br>User Password (in) | User creates an account |
| 6 | User checks learning progress | User ID (in)<br>User Progress (out) | User views account sign language learning progress. |
| 7 | User completes a test | User ID (in)<br>User Score (in) | User completes a test with a given score. |

### 2.1.3 Individual Product Use Cases

**Use Case Trigger:** User creates an account
**Primary Actors:** user, ASLingo server
**Outcomes:**

1. System creates an account with the designated username, password, and email
2. System registers user as logged in
3. System redirects user to home page

**Use Case Trigger:** User logs into the application
**Primary Actors:** user, ASLingo server
**Outcomes:**

1. System checks if password matches for corresponding username
2. System registers user as logged in
3. System redirects user to home page

**Use Case Trigger:** User logs out of the application
**Primary Actors:** user, ASLingo server
**Outcomes:**

1. System registers user as logged out
2. System redirects user to login page

**Use Case Trigger:** User requests to start a test
**Primary Actors:** user, ASLingo server
**Outcomes:**

1. System redirects user to testing page
2. System requests to turn on user webcam
3. System receives video data from user webcam

**Use Case Trigger:** User signs through webcam
**Primary Actors:** user, ASLingo server, ASLingo Computer Vision Model
**Outcomes:**

1. System interprets user signs with computer vision model
2. System responds informing user of correctness of sign

**Use Case Trigger:** User checks learning progress
**Primary Actors:** user, ASLingo server
**Outcomes:**

1. System receives request from user to view learning progress
2. System responds with user progress

**Use Case Trigger:** User completes a test
**Primary Actors**: user, ASLingo server
**Outcomes:**

1. System registers that user completed test, and saves user score

### 2.1.4 Traceability Matrix

| Use Case No. / Requirement No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| FR1 | | | | X | | | |
| FR2 | | | | X | | | |
| FR3 | | | | | X | | |
| FR4 | X | | | | | | |
| FR5 | | X | | | | | |
| FR6 | | | | | X | | |
| FR7 | | | | | | X | |
| FR8 | X | | | | | X | |
| FR9 | | | | | | X | X |
| FR10 | | | | X | | | |
| FR11 | | | | X | | | |
| FR12 | | | X | | | | |

Figure 2: Traceability Matrix for Functional Requirements

## 2.2 Functional Requirements

Table 4: Functional Requirements of ASLingo

| Requirement No. | Description | Rationale |
|---|---|---|
| FR1 | The system should be able to connect with a camera. | Connecting with a camera is a requirement for providing input to the system for hand sign recognition. |

| | | |
|---|---|---|
| FR2 | The system should be able to recognize American Sign Language hand signs. | Hand sign recognition is a requirement for users to practice what they have been learning. |
| FR3 | The system should allow users to create an account. | Account creation is a requirement for users to save progression. |
| FR4 | The system should allow users to sign into their account if it exists. | Account sign-in is a requirement for users to save progression. |
| FR5 | The system should allow users to sign out of their account if they are signed into it. | Account sign-out is necessary for good account security practices. |
| FR6 | The system should provide a diagnostic quiz for new users. | A diagnostic quiz is a requirement for determining the current skill level of the user. |
| FR7 | The system should provide a progression-based course for ASL. | A progression-based course is a requirement for ensuring users are taught ASL in a comprehensive manner. |
| FR8 | The system should save user progress. | Saving user progress is a requirement for ensuring the user follows the progression-based course. |
| FR9 | The system should allow users to access the program via a web application | The system is being built for a web application, so the user should be able to access it in this way. |
| FR10 | The system should be able to communicate to the user if they have answered the prompt correctly. | The user should know if they have answered the prompt correctly to learn the language correctly. |

| | | |
|---|---|---|
| FR11 | The system should notify the user of any potential errors that may arise during camera recognition. | This will let the user know if they need to adjust their input setting (camera angle, lighting etc.) so the system can accurately access their hand signs. |
| FR12 | The system should allow the user to their desired testing track from a selection of testing categories. | This is to provide variation in the manner of testing as well as in the subject matter that the user can test themselves on. |

# 3 Non-Functional Requirements

## 3.1 Look and Feel Requirements

Table 5: Look and Feel Non-Functional Requirements

| **Requirement No.** | **Description** | **Rationale** | **Fit Criterion** |
|---|---|---|---|
| LFR1 | The system should remind users of similar language learning applications | This will allow users to use the system intuitively if they have knowledge of other learning apps. | Ask a sample of users to rate how familiar/easy to use it is compared to other language learning apps. |
| LFR2 | The system should show the user how much progress they have made in their learning schedule. | The user should be able to gain feedback from the system about how much they learned. | The developers shall certify the product complies with these requirements. |
| LFR3 | The system should clearly show the user if they have answered the prompt correctly. | The user should gain feedback about if they are correct with the hand sign they have shown. | The developers shall certify the product shows the user the correct response based on their input. |

## 3.2 Usability and Humanity Requirements

Table 6: Usability and Humanity Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| UHR1 | The system should be able to be used by people with little to no training. | The system should be able to be used without the need for formal training to make it easier for the average user. | After their first encounter with the product, 75% of users shall report if they are comfortable using the application. |
| UHR2 | The system should be able to be used by people who are hard of hearing or deaf, as well as those who are able to hear. | The system should be accessible for all people wanting to learn ASL. | Users will be asked to rate their experience using the application on a scale of 1-10, with the average reporting a score above 7. |
| UHR3 | The system should allow users to personalize their account. | The user should be able to input their name, see their progress etc. | The developers shall certify the product complies with these requirements. |

## 3.3 Performance Requirements

Table 7: Performance Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| PR1 | The system should respond to user input quickly. | If a user has to wait too long after an input they may be less engaged. | 95% of tests should repond to user input within 1 second. |
| PR2 | The system should be able to accurately determine the sign shown by the user. | The system must be able to understand what hand signs the user is inputing to ensure they are learning effectively. | The system should accurately determine the correct hand sign from a user in 95% of tests. |
| PR3 | The system should show the user if the input needs to be adjusted. | The user should know if they need to change their camera angle, lighting etc. for the system to accurately give them proper feedback. | The developers shall certify the product complies with these requirements. |

## 3.4 Operational and Environmental Requirements

Table 8: Operational and Environmental Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| OER1 | The system should be used as a web application on a browser/laptop. | The system is being built for a web application, so the user should be able to access it in this way. | 95% of users are able to successfully use the product as a web application on a broswer source. |
| OER2 | The system should be able to access a user's camera device. | The user's camera will be used as the input device to see the user's hand signs. | 90% of users are able to successfully use their camera on their input device. |

## 3.5 Maintainability and Support Requirements

Table 9: Maintainability and Support Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| MSR1 | The system should be tested regularly to ensure it's functionality and usability. | This will ensure that the system does not experience any bugs or other errors when being updated over time. | The product must pass 95% of its developer tests at least once a week. |

| | | | |
|---|---|---|---|
| MSR2 | The code implemented in the system should adhere to specified coding standards to ensure it's readibilty for future updates. | This will ensure that the system code can be understood over time, so others can update it if they wish. | The developers shall certify the product complies with these requirements. |
| MSR3 | The code implemented in the system should be tested using code coverage methods to test all functions of the system. | This will ensure that all aspects of the system can be tested for errors, and fixed if errors are found during testing. | 80% of code coverage shall be achieved when running test cases on the system. |
| MSR4 | The system should allow for new signs to be added over the lifespan of the system. | This will allow the system to expand over time, as well as be able to add in new modern signs. The modules of the product will be written according to coding standards to allow for new additions to the system. | The developers shall certify the product complies with these requirements. |

## 3.6 Security Requirements

Table 10: Security Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| SR1 | The system should allow the user to access their account after creating it. | The user should be able to create an account and be able to log in and out of it as needed. | The developers shall certify the product complies with these requirements. |
| SR2 | The system should ensure that incorrect input to the system is used. | The system should not be trained using incorrect usage of ASL as this would contradict the goal of learning how to use the language properly. | The developers shall certify the product complies with these requirements. |
| SR3 | The system should store user account information using encryption. | This will ensure that all user information will be kept private and secure. | The developers shall certify the product complies with these requirements. |

## 3.7 Cultural Requirements

Table 11: Cultural Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| CR1 | The system should be written in Canadian English and teach users using American Sign Language. | The primary users of this application at this stage will be Canadian English speakers who want to learn American Sign Language. | The developers shall certify the product complies with these requirements. |

## 3.8 Legal Requirements

Table 12: Legal Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| LR1 | The system should adhere to user privacy laws. | The system should never share user information or break any other user privacy law. | N/A |
| LR2 | The system should not train the model on personal/confidential/illegal data | The system should only be trained on data that is safe to use, and is in the public domain. | The developers shall certify the product complies with these requirements. |

## 3.9 Health and Safety Requirements

Table 13: Health and Safety Non-Functional Requirements

| Requirement No. | Description | Rationale | Fit Criterion |
|---|---|---|---|
| HSR1 | The system should warn users to ensure they have enough space to practice ASL in their environment. | This will ensure that the user will not use this application if there is not enough space to comfortably do so. | The developers shall certify the product complies with these requirements. |

# 4 Project Issues

## 4.1 Open Issues

There are currently no open issues with the project at the moment.

## 4.2 Off-the-Shelf Solutions

1. The ASL App is a mobile exclusive platform with 2,500+ signs and phrases to teach ASL via short video clips. This app offers 4 packs for free with the basics like the

alphabet, numbers, and universal gestures. Paid packs are also available ex. compliments, moods, and social gestures for $0.99. The app ultimately serves as a mobile hub for common expressions to study and learn wherever you are.

2. Canadian Hearing Services offer both in-person and virtual educational ASL courses for a variety of experience levels. These courses educate via teacher instruction, role play, videos, and work books. A variety of other public, private, and educational institutes offer similar courses and instructional content.

3. Duolingo is a popular mobile application that provides language courses for many languages from across the world, with 50+ million monthly active users. The app utilizes gamification to encourage consistent user progress and interest. Majority of the courses are based on testing users on vocabulary and grammar via reading, listening, and speaking problems which increase in difficulty as users progress. Its courses are well developed through research and accord with global language standards, such as the Common European Framework of Reference for Languages (CEFR).

## 4.3 New Problems

There are currently no new problems.

## 4.4 Tasks

- Development should follow the agile methodology, with emphasis on different aspects of the project: front-end, back-end, and computer vision.

## 4.5 Migration to the New Product

Not applicable for our project.

## 4.6 Risks

1. The primary risk of this product is the potential for error when trying to analyze and recognize a user's sign to give feedback or determine if their form is correct. This could cause users to improperly learn signs and hinder their learning.

## 4.7 Costs

1. Website domain - TBA
2. App hosting platform - TBA
3. Database - TBA

## 4.8 User Documentation and Training

The system and its interface should be intuitive enough to learn how to use the app. User documentation to be supplied with the product should include a general setup guide and can include a glossary for basic signs, such as for the alphabet.

## 4.9 Waiting Room

1. The system should incorporate full body signs (eg. the sign for "shorts") and facial expression/body language recognition, as they are a crucial aspect of sign language.
2. The system should provide different courses for learners of varying skill level.
3. The system should teach other common sign languages.

## 4.10 Ideas for Solutions

N/A

# 5 Appendix — Reflection

1. Many software design and principles courses contribute to the team's success on the capstone project. Our project is primarily software-based, involving a combination of everything taught throughout the program, including software design, development, and documentation. Here is a list of specific courses that were particularly helpful for us:

    - SFWRENG 2AA4
    - SFWRENG 2C03
    - SFWRENG 2XA3
    - SFWRENG 2XB3
    - SFWRENG 3A04
    - SFWRENG 3RA3
    - SFWRENG 3S03
    - SFWRENG 3XA3
    - SFWRENG 4HC3

2. Some overall knowledge and skills that will be required in the entire team include domain knowledge (the fundamentals of ASL and basic signing), skills in technical tools for development (React, OpenCV, Python, etc.), documentation writing skills, and communication skills for connecting with stakeholders.

    - Edward – On the technical side, I will need to familiarize myself with OpenCV and computer vision in general. As well, I will need to learn how to apply machine learning to recognizing specific hand motions. On the non-technical side, I will need to familiarize myself with ASL and at minimum learn basic hand signs. I aim to learn more than just the fundamentals, so I can help to design the course that our project provides.
    - Cassidy – For technical skills, I am mostly familiar with back end development, so through this project I hope to learn more about front end web development, as well as learning some fundamentals of machine learning for the hand motions. I also hope to be able to learn some ASL through this project, as well as further develop project development and time management skills.
    - Stanley – As I would like to familiarize myself with computer vision, I will also need to focus on some back-end fundamentals as well. Although I'll be focusing mainly on the front-end, since our computer vision component will work hand and hand with the back-end of ASLingo, I'll naturally need to be able to know both at a basic level. For non-technical skills, I'd like to gain more experience in communicating with stakeholders on a more project management level.
    - Jeremy – With a focus on the full stack development role within the team I will have to become comfortable with modern full stack development technologies and

practises. Such practises will include creating a responsive front end that allows for an intuitive user experience, creating a back end ecosystem to store our data, and integrating the OpenCV and ML aspects. I also look forward to using github CI/CD and ensuring good code coverage testing and smooth deployments for future changes.

- Andrew – I aim to bring my industry experience in the field of system verification and translate that over to this project. I'll mainly be focusing on ensuring correctness within the python-OPenCV back-end by writing units tests to ensure complete code coverage. Naturally this will require me to understand OpenCV enough to test for edge cases and more niche scenarios in which we do not intend for the project to be used while working in lock-step with the rest of the team. I hope to additionally polish up on my communications and UI design skills to bring forward a design that will satisfy the stakeholders and make the use of our final product as smooth as possible.

3. All team members are making it a personal goal to attend as many McMaster Sign Language Club meetings as possible to learn basic signs and gestures, as well as interact with some of our stakeholders. To learn full stack development we will be following tutorials and best practises from the industry standards such as Mozilla MDN Web Docs, documentation from our framework and language of choice,and articles posted on FreeCodeCamp, GeeksForGeeks, Medium etc. Learning OpenCV and AI/ML topics will also be learned via a similar method.