

# Software Requirements Specification

## ASLingo Application

Team 15, ASLingo

Andrew Kil

Cassidy Baldin

Edward Zhuang

Jeremy Langner

Stanley Chan

September 26, 2023

# Contents

<b>1</b>	<b>Project Drivers</b>	<b>1</b>
1.1	The Purpose of the Project . . . . .	1
1.2	The Stakeholders . . . . .	1
1.2.1	Other Stakeholders . . . . .	1
1.3	Mandated Constraints . . . . .	1
1.4	Naming Conventions and Terminology . . . . .	1
1.5	Relevant Facts and Assumptions . . . . .	1
<b>2</b>	<b>Functional Requirements</b>	<b>2</b>
2.1	The Scope of the Work and the Product . . . . .	2
2.1.1	The Context of the Work . . . . .	2
2.1.2	Work Partitioning . . . . .	2
2.1.3	Individual Product Use Cases . . . . .	2
2.2	Functional Requirements . . . . .	2
<b>3</b>	<b>Non-functional Requirements</b>	<b>2</b>
3.1	Look and Feel Requirements . . . . .	2
3.2	Usability and Humanity Requirements . . . . .	2
3.3	Performance Requirements . . . . .	4
3.4	Operational and Environmental Requirements . . . . .	4
3.5	Maintainability and Support Requirements . . . . .	4
3.6	Security Requirements . . . . .	4
3.7	Cultural Requirements . . . . .	5
3.8	Legal Requirements . . . . .	5
3.9	Health and Safety Requirements . . . . .	5
<b>4</b>	<b>Project Issues</b>	<b>6</b>
4.1	Open Issues . . . . .	6
4.2	Off-the-Shelf Solutions . . . . .	6
4.3	New Problems . . . . .	6
4.4	Tasks . . . . .	6
4.5	Migration to the New Product . . . . .	6
4.6	Risks . . . . .	6
4.7	Costs . . . . .	6
4.8	User Documentation and Training . . . . .	6
4.9	Waiting Room . . . . .	6
4.10	Ideas for Solutions . . . . .	6
<b>5</b>	<b>Appendix</b>	<b>7</b>
5.1	Symbolic Parameters . . . . .	7
<b>6</b>	<b>Appendix — Reflection</b>	<b>7</b>

List of Tables

1	Revision History . . . . .	ii
2	Naming Conventions and Terminology . . . . .	2
3	Functional Requirements of ASLingo . . . . .	3

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
Date	Name	Change

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.

### 1.2.1 Other Stakeholders

## 1.3 Mandated Constraints

The project is constrained by the following:

- The Project Expenses Cannot Exceed \$750

## 1.4 Naming Conventions and Terminology

## 1.5 Relevant Facts and Assumptions

User characteristics should go under assumptions.

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 commerical name for the project

## 2 Functional Requirements

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

#### 2.1.1 The Context of the Work

#### 2.1.2 Work Partitioning

#### 2.1.3 Individual Product Use Cases

### 2.2 Functional Requirements

\*\* see Table 3: Functional Requirements of ASLingo, might need to format this differently

## 3 Non-functional Requirements

### 3.1 Look and Feel Requirements

The system should remind users of similar language learning applications (familiarity with the system, fit criterion is a sample of users could rate how familiar/easy to use it is compared to other language learning apps)

The system should show the user how much progress they have made in their learning (level system, progress bar, score after a quiz etc.)

The system should clearly show the user if they have answered the prompt correctly.

### 3.2 Usability and Humanity Requirements

The system should be able to be used by people with little to no training (intro tutorial maybe but should be easy to understand; ease of use)

Table 3: Functional Requirements of ASLingo

Requirement No.	Description	Rationale
FR1	The system should be able to connect with a camera.	
FR2	The system should be able to recognize hand signs.	
FR3	The system should allow users to create an account and sign in.	
FR4	The system should provide a diagnostic quiz for new users.	
FR5	The system should provide a progression based course for ASL.	
FR6	The system should save user progress.	
FR7	The system should allow users to access the program via a web application (functional or a constraint?)	
FR8	The system should be able to communicate to the user if they have answered the prompt correctly (idk how to phrase it but it should give user feedback on if they answered the question they were given)	
FR9	The system should tell the user if they have to adjust the input to the system (error messages to tell the user to move the camera, change lighting, try sign again etc.)	

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 (accessibility)

The system should allow users to personalize their account (avatar, name, progress; personalization)

### **3.3 Performance Requirements**

\*can change the time/percents shown\*

The system should respond to user input within 1? second (speed/latency)

The system should be able to accurately determine the sign shown by the user 90%? of the time (accuracy)

The system should be able to accurately match the user input sign to the prompt given by the system 90%? of the time (accuracy)

The system should be able to host ??? users at one time (capacity)

The system should allow for new signs to be added over the lifespan of the system (scalability/longevity)

The system should show the user if the input needs to be adjusted (fault-tolerance? functional?)

### **3.4 Operational and Environmental Requirements**

The system should be used as a web application on a browser/laptop (expected physical environment)

The system should be able to access a user's camera device

### **3.5 Maintainability and Support Requirements**

The system should be tested regularly to ensure it's functionality and usability (maintenance)

### **3.6 Security Requirements**

The system should allow the user to access their account after creating it (access)

The system should ensure that incorrect input to the system is used (integrity)

The system should store user account info securely? or keep user account info private?  
(privacy)

### **3.7 Cultural Requirements**

The system should be written in Canadian English and teach users using American Sign Language.

### **3.8 Legal Requirements**

The system should adhere to user privacy laws?

The system should not train the model on personal/confidential/illegal data?

### **3.9 Health and Safety Requirements**

This section is not in the original Volere template, but health and safety are issues that should be considered for every engineering project.



## 4 Project Issues

4.1 Open Issues

4.2 Off-the-Shelf Solutions

4.3 New Problems

4.4 Tasks

4.5 Migration to the New Product

4.6 Risks

4.7 Costs

4.8 User Documentation and Training

4.9 Waiting Room

4.10 Ideas for Solutions

References

## 5 Appendix

This section has been added to the Volere template. This is where you can place additional information.

### 5.1 Symbolic Parameters

The definition of the requirements will likely call for SYMBOLIC\_CONSTANTS. Their values are defined in this section for easy maintenance.

## 6 Appendix — Reflection

The information in this section will be used to evaluate the team members on the graduate attribute of Lifelong Learning. Please answer the following questions:

1. Which of the courses you have taken, or are currently taking, will help your team to be successful with your capstone project.
2. What knowledge and skills will the team collectively need to acquire to successfully complete this capstone project? Examples of possible knowledge to acquire include domain specific knowledge from the domain of your application, or software engineering knowledge, mechatronics knowledge or computer science knowledge. Skills may be related to technology, or writing, or presentation, or team management, etc. You should look to identify at least one item for each team member.
3. For each of the knowledge areas and skills identified in the previous question, what are at least two approaches to acquiring the knowledge or mastering the skill? Of the identified approaches, which will each team member pursue, and why did they make this choice?