# UniiLanguage

# IMCAT

# Project Plan (Final)

Alina Kim

Isabel Tuason

Thomas Yung

Christina Vu

Mohammed Imran Shilleh

02.6.2022

# Table of Contents

# Revision History

### Version 1.0

- Date: 1/23/22
- Added the statement of work, stakeholder analysis, technical constraints, WBS, and Gantt chart

### Version 2.0

- Date: 2/1/22
- Updated statement of work to reflect new project direction
- Updated technical constraints to reflect new project direction
- Added additional stakeholders
- Updated Gantt chart to reflect updated deadlines
- Fleshed out WBS development tasks in response to clarified project requirements
- Annotated WBS to reflect which tasks were completed by which team member
- Update WBS point value to reflect sprint report

# Introduction

## Statement of Work

UniiLanguage is a web-based application designed to reinforce language maintenance and acquisition in students who are learning a new language. Geared towards use in a classroom setting between the grades of K-12, the application will serve as an accessory to teachers who wish to incorporate it as part of a language-learning curriculum.

For the scope of this project, this application will consist of one mini-game that involves having students draw the prompt provided in distinct time intervals of 10, 30, and 60

seconds (the 10/30/1 game). This mini-game is designed to be primarily accessible in an independent manner (pre-programmed with an existing prompt list for each available language), but can include a method by which students can input a course code to access a custom-designed prompt list for use in a curriculum setting.

Within the span of twenty-one weeks, this project will have been designed, developed, and deployed, following the specifications laid out within this document, and with an emphasized focus on both usability and modularity. This application will then be utilized to assist language-learning students (including, and perhaps especially, those who are learning English), as well as to collect information on the benefits of the Olson-Gillingham method in the context of a multi-sensory approach to language maintenance and acquisition.

## Stakeholders

| **Stakeholders** | **Interests/Affects** |
|---|---|
| Maria Teresa (Sponsor) | The sponsor's goal is to create an application that can be implemented in schools to assist students in learning new languages by using multi-sensory techniques. |
| Teachers | Teachers provide students with a new tool to learn a new language in an interactive and engaging way. |
| Students | Students learn a new and fun tool for interacting with learning different languages by drawing on a board rather than just reading out of a book. |
| School Districts | School districts will be able to keep supporting the application if they see their students improving on their learning language. |
| Duolingo | Duolingo will be affected negatively since students would possibly not be using their application since they will be learning the language by the UniiLanguage application. |

| | |
|---|---|
| Rosetta Stone | Rosetta Stone will be affected negatively since students would have another way of learning a different language with a different application in school than learning with Rosetta Stone at home. |
| Kahoot | Kahoot will be affected by the new application negatively since teachers might not be hosting Kahoot games and possibly let the students learn from drawing in the new language they are learning. |
| Khan Academy | Khan Academy will be affected negatively by the new application since students might switch applications since they can possibly have more fun and learn more in this new application compared to Khan Academy. |
| Quizlet | Quizlet will be affected negatively since students are able to learn a new language and have fun learning the language by drawing on board compared to Quizlet which do flashcards. |

## Technical Constraints

Due to the fact that this application is being developed without the context of a pre-existing framework within which to be integrated, each of the technical constraints yields a considerable level of flexibility. That being said, these constraints should be adhered to in order to ensure maximum efficacy of the final product.

With a heavy focus on accessibility, this application must make a tradeoff between internet accessibility and platform independency. Due to the fact that this application is being designed primarily for use in classroom settings, UniiLanguage will be developed in a web framework to allow for universal access, regardless of platform. In this same vein, this application must prioritize usability, allowing both teachers and students to intuitively and effortlessly navigate the application. This means that the application

must be flexible, and non-prescriptive-- it must not enforce any particular curriculum, unlike many other existing language-assistance applications.

In addition to these technical constraints, there also exist a number of technical preferences. These prescriptions, while not necessary to create a minimum viable product, will serve to reduce the development costs for further modifications to this application. As such, a focus should be made on independent usage-- allowing students to access the application on their own time, in order to practice these visual associations as language acquisition "drills." In addition, for the sake of accessibility, the application should be very bare-bones in terms of navigation-- users should be able to simply go to the website, and begin practicing immediately. To this end, the website should not feature a login system, but should use client-side cookies in order to ensure that users will not get repeat prompts. In addition, this application should be made as modular as possible-- UniiLanguage ultimately intends to serve as a fully-fledged platform consisting of a system of related language drills, each utilizing some multi-sensory aspect to further reinforce the efficacy of language acquisition.

## Work Breakdown Structure (WBS)

WBS in increasing levels of detail (analogous to a list of sprints): 10 sprints total

1. **Planning** - 1.5 sprints (3 weeks, 25 total story points)
    - 1.1. Administrative Setup
        - 1.1.1. Setting up File Structure (1 point) - Alina
        - 1.1.2. Communication Covenant (1 point) - *Everyone*
        - 1.1.3. Stakeholder Communication Plan (2 points) - *Everyone*
        - 1.1.4. Team Gantt (3 points) - Alina
    - 1.2. Project Ideation and Requirements
        - 1.2.1. Brainstorming Board (2 points) - *Everyone*
        - 1.2.2. UML Class Diagrams (3 points) - *Everyone*
        - 1.2.3. Use Case Sequence Diagramming (3 points) - *Everyone*
        - 1.2.4. Business Plan (5 points) - Alina, Mohammed, Thomas, Christina

1.2.5. Project Plan (5 points) - Alina, Mohammed, Isabel

2. **Design** - 3 sprints (6 weeks, 38 total story points)
    2.1. User Research
        2.1.1. User Flows (3 points) - Isabel, Christina, Mohammed
    2.2. Interface Design
        2.2.1. Sketching (2 points) - Isabel, Christina, Mohammed
        2.2.2. Lo-Fi Wireframes (3 points) - Isabel, Christina, Mohammed
        2.2.3. Mid/High-Fidelity Mockups (10 points) - Isabel, Christina, Mohammed
    2.3. Usability Testing
        2.3.1. User Interviews (5 points) - *Everyone*
        2.3.2. Cognitive Walkthroughs (5 points) - *Everyone*
    2.4. (Figma) Prototyping
        2.4.1. Interactive (Figma) Prototype (10 points) - Isabel, Christina, Mohammed
3. **Development** - 3 sprints (6 weeks, 115 total story points)
    3.1. Front-End
        3.1.1. Prototyping
            3.1.1.1. Independent UI Frame Design - Isabel, Christina, Mohammed (5 points)
            3.1.1.2. Basic Callback Testing (3 points) - Isabel, Christina, Mohammed
        3.1.2. Iterating
            3.1.2.1. Develop MVP (10 points) - Alina, Thomas
            3.1.2.2. Standardize and Organize UI (5 points) - Isabel, Christina, Mohammed
            3.1.2.3. Adjust Based on Backend Constraints (3 points) - Isabel, Christina, Mohammed
        3.1.3. Testing
            3.1.3.1. Test Integration Functionality (5 points) - *Everyone*

3.1.3.2. Usability Feedback (5 points) - Isabel, Christina, Mohammed

3.1.4. Finalizing

3.1.4.1. Polish UI (5 points) - Isabel, Christina, Mohammed

3.1.4.2. Add Details and Multisensory Elements (5 points) - Alina, Thomas

3.2. Back-End

3.2.1. Prototyping

3.2.1.1. Experiment With Random Prompt Generation (3 points) - *Everyone*

3.2.1.2. Establish Basic Software Framework (5 points) - Alina, Thomas

3.2.1.3. Develop Independent Microservice Modules (5 points) - Alina, Thomas

3.2.2. Iterating

3.2.2.1. Develop MVP (10 points) - *Everyone*

3.2.2.2. Integrate Modules Into Framework (5 points) - Alina, Thomas

3.2.2.3. Adjust Based on Frontend Constraints (3 points) - Alina, Thomas

3.2.3. Testing

3.2.3.1. Write Edge Cases and Debug Tests (5 points) - Alina, Thomas

3.2.3.2. Reiterate with Test Cases (3 points) - Alina, Thomas

3.2.4. Finalizing

3.2.4.1. Polish Functionality (5 points) - Alina, Thomas

3.2.4.2. Optimize Space-Time (5 points) - Alina, Thomas

3.2.4.3. Refactor for Modularity (5 points) - Alina, Thomas

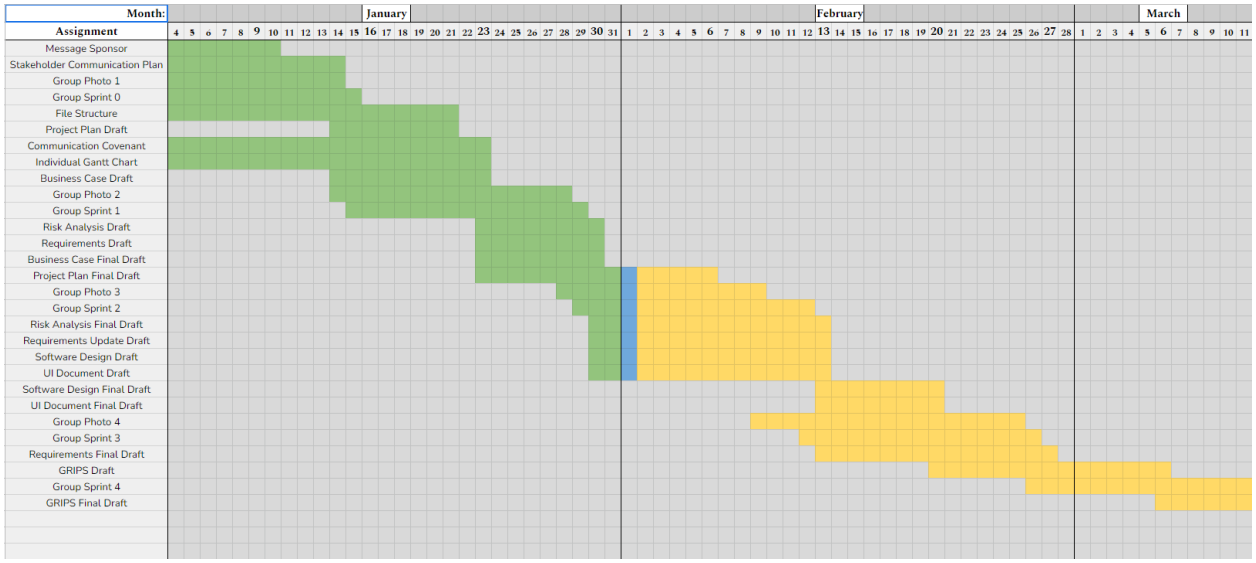
3.3. Integration

3.3.1. Test Integration Functionality (5 points) - Alina, Thomas

3.3.2. Iterate As Necessary (10 points) - *Everyone*

4. **Execution** - 2 sprints (4 weeks, 30 story points)
    4.1. Deployment (15 points) - *Everyone*
    4.2. Sprint Review (5 points) - *Everyone*
    4.3. Testing (10 points) - *Everyone*
5. **Maintenance** - 0.5 sprints (1 week, 10 story points)
    5.1. Testing (10 points) - *Everyone*

## Gantt Chart*




**Any task not parallelizable is dependent on the task which come before it*

**Project managers:**

Pre-Sprints (Weeks 1-2): Thomas

Sprint 1 (Weeks 3-4) : Alina

Sprint 2 (Weeks 5-6) : Mohammed

Sprint 3 (Weeks 7-8) : Isabel

Sprint 4 (Weeks 9-10) : Christina