A blue and white logo

Description automatically generated

**Software Engineering Department**

**Braude College**

Capstone Project Phase A

**Angel Hands – Sign language Learning Game**



**Project Number 24-1-D-39**

**Advisor**: **PhD. Sulamy Moshe –** [MosheSu@braude.ac.il](mailto:MosheSu@braude.ac.il)

**Students**: Eitan German – [eitan.german@e.braude.ac.il](mailto:eitan.german@e.braude.ac.il)

Dolev Almog – [dolev.almog@e.braude.ac.il](mailto:dolev.almog@e.braude.ac.il)

[Abstract 4](#_Toc167562994)

[1. Introduction 4](#_Toc167562995)

[2. Related Work 5](#_Toc167562996)

[3. Background 6](#_Toc167562997)

[3.1 American Sign Language (ASL) and Current Learning Methods 6](#_Toc167562998)

[3.2 Unity Engine 7](#_Toc167562999)

[3.3 Google Mediapipe Holistic 9](#_Toc167563000)

[3.4 Sign language Recognition using Machine learning (LSTM Model) 10](#_Toc167563001)

[4. Expected Achievements 10](#_Toc167563002)

[4.1. Outcomes 10](#_Toc167563003)

[4.2 Unique Features 11](#_Toc167563004)

[4.2.1 Gamification 11](#_Toc167563005)

[4.2.2 Sign Language Recognition 11](#_Toc167563006)

[4.2.3 Design 12](#_Toc167563007)

[4.3 Criteria for success 12](#_Toc167563008)

[5. Research/Engineering Process 12](#_Toc167563009)

[5.1 Research – ASL (American Sign Language) 12](#_Toc167563010)

[5.1.1 Constraints and Challenges 13](#_Toc167563011)

[5.1.2 Conclusions from Research 13](#_Toc167563012)

[5.2 Research - Game Development with sign language recognition 13](#_Toc167563013)

[5.2.1 Constraints and Challenges 14](#_Toc167563014)

[5.4 Methodology and Development Process 14](#_Toc167563015)

[6. Product 15](#_Toc167563016)

[6.1 Requirements 15](#_Toc167563017)

[6.1.1 Functional Requirements 15](#_Toc167563018)

[6.1.2 Non Functional Requirements 15](#_Toc167563019)

[6.2 Architecture Overview 16](#_Toc167563020)

[6.2.1 Unity-Application 16](#_Toc167563021)

[6.2.2 Scenes 17](#_Toc167563022)

[6.2.3 Interfaces and Game Flow 17](#_Toc167563023)

[6.2.3 Data Management 18](#_Toc167563024)

[6.3 Diagrams 19](#_Toc167563025)

[6.3.1 Package Diagram 19](#_Toc167563026)

[6.3.2 Activity Diagram 20](#_Toc167563027)

[7. Verification And Evaluation 20](#_Toc167563028)

[7.1 Testing plan 21](#_Toc167563029)

[7.2 Usability Testing 22](#_Toc167563030)

[8. References 22](#_Toc167563031)

[9. GitHub Link 22](#_Toc167563032)

# Abstract

Angel Hands Is an educational video game that focuses on sign language learning for children and adults. The game’s goal is to teach its users how to speak in the American sign language (ASL).

It uses computer vision to capture and detect the user’s hands, it uses machine learning to analyze the signs that the user is presenting to the camera of the computer/ mobile phone/ tablet.

To make the game more attractive it incorporates adventure-themed elements and exciting challenges that aim to keep the user motivated and excited to learn.

The game is designed for users of all knowledge levels and experiences in ASL by incorporating multiple difficulty levels for the user to choose from and thus stays relevant even for the more experienced ASL signers.

# 1. Introduction

In today's interconnected world, the importance of effective communication cannot be overstated. Yet, for millions of individuals, traditional methods of communication may not always be accessible. Language barriers can isolate individuals and hinder their ability to express themselves fully, leading to feelings of frustration and exclusion. In such instances, alternative forms of communication, such as sign language, can bridge these gaps and empower individuals to connect with others in meaningful ways.

Recognizing the significance of sign language as a vital means of communication, we introduce "Angel Hands," an innovative educational video game designed to facilitate the learning of American Sign Language (ASL) for both children and adults. This project represents a fusion of technology and education, leveraging the immersive capabilities of modern gaming platforms to make language learning engaging, interactive, and accessible to a broad audience.

At its core, "Angel Hands" seeks to revolutionize the way ASL is taught by harnessing the power of computer vision and machine learning technologies. Through the use of computer vision, the game is able to detect and analyze users' hand movements in real-time, providing immediate feedback and guidance as they navigate through the intricacies of ASL. This dynamic approach not only enhances the learning experience but also accommodates users of varying proficiency levels, from beginners to advanced learners.

In addition to its innovative technological features, "Angel Hands" distinguishes itself through its captivating gameplay mechanics. By incorporating adventure-themed elements and exciting challenges, the game transforms the learning process into an immersive journey filled with discovery and excitement. From solving puzzles to embarking on quests, users are propelled into a world where learning becomes an adventure, fueling their motivation and enthusiasm to master ASL.

Furthermore, "Angel Hands" prioritizes inclusivity by offering multiple difficulty levels tailored to accommodate users with diverse learning needs and experiences. Whether someone is just beginning their ASL journey or seeking to refine their skills, the game provides a supportive and engaging environment for continuous learning and growth.

In essence, "Angel Hands" represents more than just a video game; it symbolizes a commitment to breaking down barriers and fostering greater inclusivity through the power of technology and education. By empowering individuals to learn and communicate in ASL, we hope to not only enrich their lives but also promote understanding and acceptance in our global community. Join us on this transformative journey as we embark on an adventure of learning, connection, and empowerment with "Angel Hands."

The sign language recognition is the main feature of the project, which enables the game to recognize the signs the user is presenting to the camera.

# 2. Related Work

1. **Educational Video Games for Language Learning**: Educational video games have been widely recognized as effective tools for language learning. Research studies such as those conducted by Gee (2003)[[1]](#footnote-2) have highlighted the benefits of integrating gaming elements into educational environments, citing increased engagement, motivation, and retention of language skills among learners.
2. **Computer Vision in Language Learning Applications**: The integration of computer vision technology in language learning applications has shown promising results in enhancing user experience and learning outcomes. Works such as that of Oviatt (1999)[[2]](#footnote-3) have explored the potential of multimodal interfaces, including hand gesture recognition, in facilitating language acquisition, particularly for sign languages like ASL.
3. **Machine Learning in Gesture Recognition**: Machine learning algorithms have played a pivotal role in advancing gesture recognition systems, enabling more accurate and efficient analysis of hand movements. Studies such as those by Starner et al. (1998)[[3]](#footnote-4) & Wu (2024)[[4]](#footnote-5) have demonstrated the effectiveness of machine learning techniques, such as neural networks and deep learning, in recognizing and interpreting gestures for various applications, including sign language recognition.
4. **Gamification in Language Learning:** The incorporation of gamification principles in language learning has been extensively studied and proven to be beneficial in enhancing motivation and engagement. Research by Nick. (2023)[[5]](#footnote-6)and Fadhlina Izzah Saman et al. (2019)[[6]](#footnote-7) has highlighted the positive impact of gamified elements, such as challenges, rewards, and immersive storytelling, on language acquisition and skill development.

# 3. Background

## 3.1 American Sign Language (ASL) and Current Learning Methods

American Sign Language (ASL)[[7]](#footnote-8) is a complete, natural language that has the same linguistic properties as spoken languages, with grammar that differs from English. ASL is expressed by movements of the hands and face. It is the primary language of many North Americans who are deaf and hard of hearing and is used by some hearing people as well.

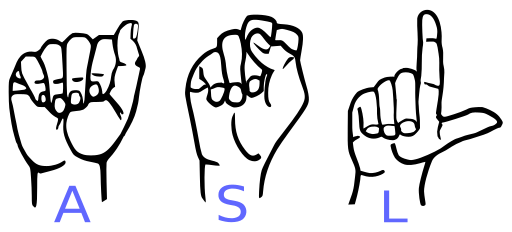
Here's a brief overview of ASL and the current learning methods used to teach this unique language:

**ASL: A Visual Language**: Unlike spoken languages, ASL is not based on sounds but on visual cues. Each sign represents a word or concept, and the language has its own grammar and syntax. ASL users rely on handshapes, movements, and facial expressions to convey meaning, making it a rich and dynamic mode of communication.

**Current Learning Methods**: Traditionally, ASL has been taught through in-person classes, textbooks, and instructional videos[[8]](#footnote-9). These methods often focus on rote memorization of signs and basic grammar rules, with limited opportunities for hands-on practice and interaction. However, advancements in technology have led to the development of innovative learning methods that leverage digital tools and interactive platforms to enhance the learning experience:

1. **Online Courses and Tutorials**: With the rise of e-learning platforms, individuals can now access ASL courses and tutorials online. These courses often feature video lessons, interactive quizzes, and community forums where learners can practice their skills and connect with fellow ASL enthusiasts.
2. **Mobile Apps**: There is a growing number of mobile apps designed to teach ASL in a fun and engaging way. These apps typically offer interactive lessons, games, and quizzes that reinforce vocabulary and grammar concepts while allowing users to practice signing at their own pace.
3. **Virtual Reality (VR) and Augmented Reality (AR)**: Emerging technologies like VR and AR hold promise for immersive ASL learning experiences. By simulating real-world scenarios and providing interactive feedback, VR and AR platforms can help users develop their signing skills in a more realistic and engaging environment.
4. **Community Workshops and Events**: Many communities host ASL workshops, meetups, and events where learners can practice their skills and interact with native signers. These in-person gatherings provide valuable opportunities for immersion and cultural exchange, complementing formal language instruction.

In summary, ASL is a visual language characterized by handshapes, movements, and facial expressions. While traditional learning methods include in-person classes and instructional materials, advancements in technology have led to the development of innovative digital tools and interactive platforms that enhance the ASL learning experience, making it more accessible and engaging for learners of all levels.

[[9]](#footnote-10)

## 3.2 Unity Engine

Unity is a powerful and versatile game engine that has revolutionized the landscape of game development since its inception in 2005. With its user-friendly interface, robust features, and cross-platform capabilities, Unity has become the engine of choice for developers worldwide. Below, we'll explore some of the key capabilities that make Unity stand out in the industry [[10]](#footnote-11).

**Cross-Platform Development**: One of Unity's greatest strengths is its ability to deploy games across multiple platforms seamlessly. Whether targeting desktops, consoles, mobile devices, or even emerging platforms like VR and AR, Unity provides developers with the tools they need to reach a wide audience without compromising on quality or performance.

**Visual Scripting with Unity Playmaker**: Unity's visual scripting tool, Playmaker, empowers developers of all skill levels to create complex game logic and behavior without writing a single line of code. With its intuitive interface and extensive library of actions, Playmaker streamlines the development process and allows for rapid prototyping and iteration.

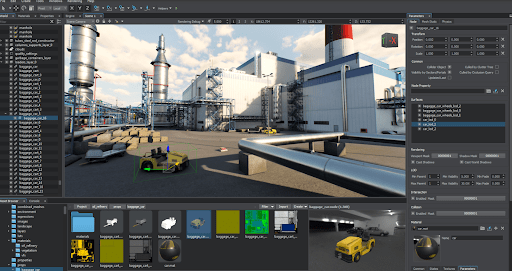
**High-Fidelity Graphics**: Unity's graphics capabilities are second to none, enabling developers to create stunning, high-fidelity visuals that rival those found in AAA titles. From dynamic lighting and real-time reflections to advanced shaders and post-processing effects, Unity provides the tools and resources necessary to bring any creative vision to life.

**Asset Store**: The Unity Asset Store is a treasure trove of resources for developers, offering a vast library of pre-made assets, tools, and plugins to accelerate development workflows. Whether in need of 3D models, sound effects, or code snippets, developers can find everything they need to enhance their projects and save valuable time and resources.

**Multiplayer Networking**: Unity's built-in networking features make it easy to implement multiplayer functionality into games of any scale. Whether creating a small cooperative experience or a massive online multiplayer game, Unity provides the infrastructure and tools necessary to connect players around the world and deliver seamless online experiences.

**Extensive Documentation and Community Support**: Unity boasts an active and vibrant community of developers, artists, and enthusiasts who are always willing to lend a helping hand. In addition to extensive documentation and tutorials provided by Unity Technologies, developers can find support and guidance from fellow community members through forums, user groups, and online communities.

In conclusion, Unity Engine offers a comprehensive suite of tools and capabilities that empower developers to bring their creative visions to life. With its cross-platform deployment, visual scripting, high-fidelity graphics, extensive asset store, multiplayer networking, and robust community support, Unity remains at the forefront of game development, enabling developers to create immersive and engaging experiences for players worldwide.

[[11]](#footnote-12)

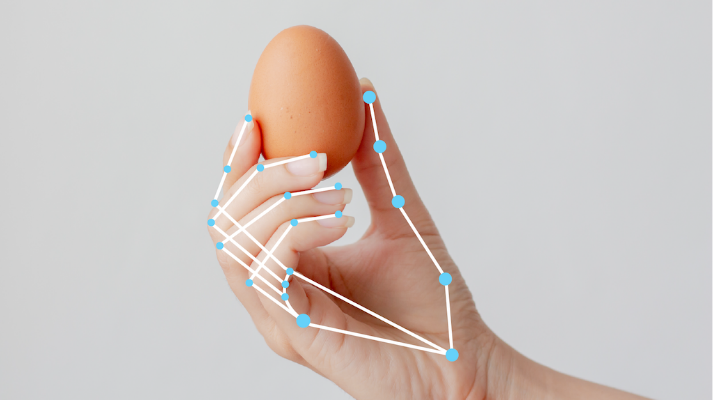
## 3.3 Google Mediapipe Holistic

The Mediapipe Holistic Landmarker task lets you combine components of the [pose](https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker/index), [face](https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker/index), and [hand](https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker/index) landmarkers to create a complete landmarker for the human body. You can use this task to analyze full-body gestures, poses, and actions. This task uses a machine learning (ML) model on a continuous stream of images. The task outputs a total of 543 landmarks (33 pose landmarks, 468 face landmarks, and 21 hand landmarks per hand) in real-time.[[12]](#footnote-13)

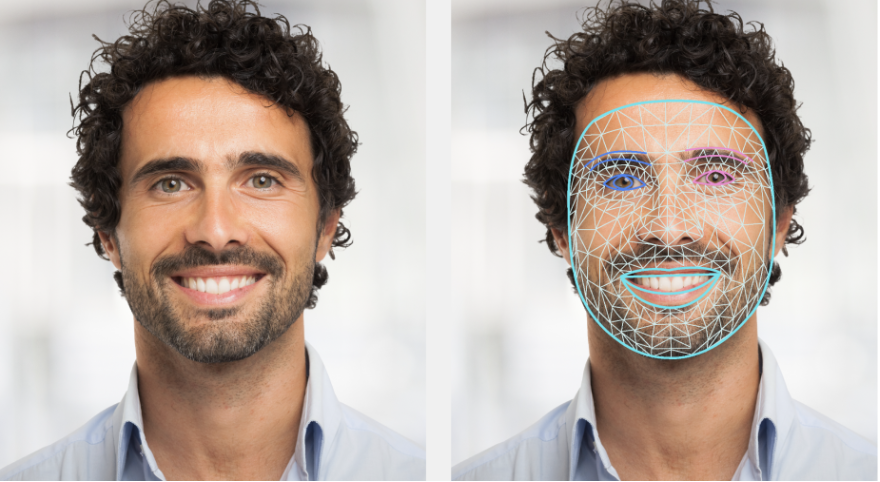
In the past a recognition algorithm of any kind required a very diverse dataset and specifically sign language recognition required many pictures captured by different people wearing different clothes in different environments to get accurate detection.

Using Google Mediapipe Holistic would allow us to assess only the key landmarks of the users’ video feed. This means that we no longer need a diverse data set to get accurate predictions. We are going to use only the landmarks to train and test the model which means faster and easier training and way faster recognition in real time.

Pose Landmarks[[13]](#footnote-14) Hand landmarks[[14]](#footnote-15)

Face Landmarks[[15]](#footnote-16)



## 3.4 Sign language Recognition using Machine learning (LSTM Model)

Deep learning algorithms, particularly Long Short-Term Memory (LSTM) networks, offer significant potential for interpreting sign language. LSTMs, a type of recurrent neural network (RNN), are well-suited for sequential data processing and have shown promise in various natural language processing tasks, including sign language recognition. Here's a summary of how LSTM networks can be utilized for interpreting sign language [[16]](#footnote-17) [[17]](#footnote-18):

1. **Sequential Data Processing**: LSTM networks excel in capturing temporal dependencies and long-range dependencies within sequential data, making them ideal for modeling the sequential nature of sign language gestures. By processing sequences of hand movements and gestures over time, LSTM networks can learn complex patterns and dynamics inherent in sign language expressions.
2. **Feature Extraction and Representation Learning**: In sign language recognition, LSTM networks can be employed to extract meaningful features from raw input data, such as video sequences of hand gestures. Through training, LSTM layers learn to automatically discover and represent relevant features, eliminating the need for manual feature engineering and improving model performance.
3. **Gesture Classification and Recognition**: Trained LSTM models can classify and recognize sign language gestures with high accuracy. By leveraging learned representations of sign language sequences, LSTM networks can effectively distinguish between different signs and interpret their meanings, enabling real-time translation of sign language into text or speech.
4. **Real-Time Interpretation**: LSTM networks can facilitate real-time interpretation of sign language by processing continuous streams of input data and providing immediate feedback. This capability is crucial for applications such as sign language translation systems and assistive devices for Deaf individuals.

# 4. Expected Achievements

## 4.1. Outcomes

The goals we aim to accomplish with this project is to create a game that can teach any person sign language in a fun and interactive way, we strive to create a game that will be able to teach any person sign language regardless of their knowledge level with sign language. The game will include problem-solving, critical thinking, and decision-making elements which can further help make an even more engaging learning experience. The game will provide various difficulty levels to create a learning environment suitable for a person with varying levels of experience and thus providing a path for the user to become great at sign language. With its sign recognition feature, anyone with a computer and a webcam will be able to learn sign language from the comfort of their home. With the versatile design of this feature, if provided with a good dataset we will be able to teach multiple sign languages, our primary focus as of now will be ASL which is the American sign language. By utilizing cloud technology, the game can be easily accessed from anywhere with an internet connection.

## 4.2 Unique Features

### 4.2.1 Gamification

Gamification is a strategy that integrates entertaining and immersive gaming elements into nongame contexts to enhance engagement and motivate certain behaviors[[18]](#footnote-19).

In our educational game, gamification is reflected through various features and elements designed to make learning more engaging, fun, and rewarding. These features include:

* Challenges and levels: The game incorporates challenges and levels that gradually increase in difficulty, providing a sense of accomplishment and progression as the user Improves his sign language.
* Interactive gameplay: The game utilizes interactive gameplay elements that keep the player engaged and motivated.
* Fun and engaging visuals: The game uses fun and engaging visuals, and sound effects that make learning more enjoyable and immersive.
* Game points and rewards: The game offers in-game points and rewards for completing levels, learning new signs, and achieving specific goals. These points and rewards serve as motivators to continue playing and learning.

### 4.2.2 Sign Language Recognition

The development of the sign recognition feature for our game provides an interactive and engaging experience for the user, this feature allows the user to interact with the game in an intuitive way, resembling the actual way that a person would communicate using sign language in real life. this feature allows the user to practice sign language by showing signs with their hands to the camera, which are then processed by a sign language recognition algorithm that detects and predicts which sign the user is presenting to the camera, it could be further used to measure the accuracy of the user’s "pronunciation" using sign language and giving recommendations on how to improve his signs.

### 4.2.3 Design

The use of fun and engaging visuals, sound effects and levels in the game creates an immersive environment for learning that will encourage the user to learn more and improve his sign language as he plays the game. This approach makes learning more enjoyable and helps people stay motivated as they interact with the game. we strive to create such an environment that will be fun to stay in for prolonged periods of time, it will encourage the user to explore the game world and encounter various challenges and tasks.

## 4.3 Criteria for success

The primary goal of our project is to develop an interactive and engaging game for everyone to learn sign language. Our game incorporates a sign language recognition feature that enables the user to learn, practice and improve their signing skills in a natural and interactive environment. The success relies on several criteria:

* The sign recognition feature must accurately recognize and analyze signs displayed by the user, providing real time feedback.
* The game must provide age-appropriate content and cover a wide range of sign language levels and experience varying from letters, words, sentences.
* The game's performance must be flawless, ensuring a seamless learning experience.
* The game's interface must be intuitive and user-friendly to allow an easy navigation experience.
* The game must incorporate fun and engaging gameplay elements that keep the user motivated and interested in learning, ensuring that they remain engaged throughout their learning journey.

These criteria are essential for the success of the project, as they ensure that the game is effective in helping players to learn sign language and it is user friendly, engaging and enjoyable.

# 5. Research/Engineering Process

## 5.1 Research – ASL (American Sign Language)

The initial research challenge is to transfer the learning process of ASL into a game.  
In order to do so we need to understand the deep meaning of learning a new language.  
A theoretical background would gain us a deeper understanding of those challenges and will help us to develop motivations for the players to have a unique and fun experience of learning ASL.  
We searched for gamification for other learnings languages games to be inspired on how to encourage players to learn ASL. while doing so we would aspire to make ASL accessible and more familiar.

In our research the conventional way of learning sign language is through videos, textbooks or in person meetings, there are very little games that aim to teach sign language and they are very limited.  
Our goal is to understand how to make sign language learning accessible and fun, we researched many articles and watched many videos to get the basic understanding of the flow of the language so that we would be able to develop a game around it with a great learning experience.

### 5.1.1 Constraints and Challenges

* The need to adhere to safety and privacy regulations when developing an application that actively uses the camera, this can limit data collection and personalization features.
* Ensuring that the application strikes the right balance between entertainment and educational value.
* Designing effective feedback mechanisms to motivate players and provide them with a sense of progress.

### 5.1.2 Conclusions from Research

* A combination of fun, challenging, and practical learning activities can help to sustain a player's interest and maximize their learning outcomes.
* Personalization features, such as adapting the application to the players sign language level and progress, can improve the effectiveness and engagement of the learning experience.

## 5.2 Research - Game Development with sign language recognition

Asl is the American sign language and will be the main language we will be teaching in our game, keeping in mind that there is a variety of sign languages, we strive that our game will be able to load sign language model file to support other languages in the future.

We researched many models for SLR (Sign Language Recognition), varying from computer vision algorithms and deep learning algorithms, while the computer vision algorithms worked well, we found it hard to create and train a model withing a reasonable time. When going to deep learning algorithms we found that the data we feed it greatly impacts the accuracy and responsiveness of the sign language recognition and while some of the models worked well, they were almost impossible to expand easily since we needed thousands of pictures to retrain the model which also took a very long time to train.

Eventually we found a middle ground between deep learning and computer vision to solve the task of detecting signs in an easy and effective way. We found and adapted a model that uses computer vision to draw landmarks on the feed from the users camera, these landmarks are then processed through a LSTM model, this process is quite quick and provides us with a simple way to teach the model new signs with very little effort, the code captures short videos of each sign numerus times to create a dataset for the LSTM model.

### 5.2.1 Constraints and Challenges

* The primary challenge lies in ensuring that the game precisely recognizes and reacts to the player's hand movements. Achieving a high degree of accuracy in recognition technology can be challenging.
* Sign language recognition demands substantial processing power and memory, which might restrict some devices. This limitation can affect the game's complexity and the variety of features that can be implemented.
* Integrating sign language into game mechanics presents a challenge. The game needs to be designed so that sign language recognition technology is smoothly incorporated into the gameplay, ensuring it doesn't feel like a mere add-on.
* Sign language recognition technology can vary in performance based on the player's hand characteristics.
* Sign language recognition technology involves collecting and storing user data, which can raise concerns about privacy.

## 5.4 Methodology and Development Process

For development we chose to go with the Agile methodology which we find to be an iterative and flexible approach to project management and software development. It emphasizes collaboration, adaptability, and continuous improvement throughout the development process.

We will be splitting out the development into:

1. Building the various scenes and basic gameplay mechanics using Unity engine.
2. Developing the sign language recognition model.
3. Training and testing the sign language recognition model.
4. Integrating the sign language model into the game and optimizing it for accuracy and responsiveness.
5. Incorporating engaging gameplay elements and challenges to keep players motivated and excited to learn.
6. Implementing cloud technology to allow for data storage and sharing, as well as for remote updates and bug fixes.
7. Continuously iterating and improving the game.
8. Conducting user acceptance testing to ensure that the game meets all requirements and functions properly.

Throughout the process, on the last day of each iteration an evaluation will be made, a working version of the game will be compiled and tested, this will assure us that we always have a working version of the game. if any changes or corrections needed, they will be made before the next iteration starts (in case the changes require a lot of work will be added as a user story for the next iteration). At the last day the user stories for the next iteration will be evaluated and changed accordingly. We will focus on delivering a working software which considers feedback from users that will be testing and evaluating our product.

# 6. Product

## 6.1 Requirements

### 6.1.1 Functional Requirements

|  |  |
| --- | --- |
| 1 | The game should have a sign language recognition feature |
| 2 | The game should provide real time feedback on the signs the user is presenting |
| 3 | The game should have themed gameplay |
| 4 | The game should have engaging gameplay elements and challenges |
| 5 | The game should display guidance dialogs |
| 6 | The game should store the progress of the user in a Database/ Game Save file |
| 7 | The game should support multiple screen resolutions |
| 8 | The game should allow the player to navigate around the map |
| 9 | The system has adjustable settings |
| 10 | The system offers 3D game environment |
| 11 | The system support camera live feedback |
| 12 | The system should support connection to google cloud services |

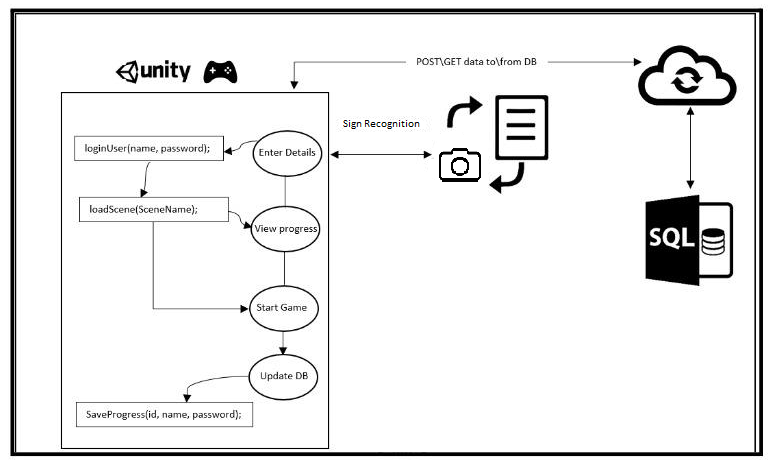
### 6.1.2 Non Functional Requirements

|  |  |
| --- | --- |
| 1 | Gameplay themes will include Learning, Arcade modes for gameplay |
| 2 | Guidance dialog will provide the user with the way a sign should be presented (image/animation) |
| 3 | Sign language recognition should be quick and responsive. |
| 4 | Sign language recognition should be accurate |
| 5 | Supported resolutions will be 1080p, 2k, 4k |
| 6 | The game should run smoothly and not crash |
| 7 | In the occasion of a game crash there should be a proper log for it for debugging and future improvement |
| 8 | The game should have Attractive music and environment |
| 9 | The game must stand on its educational purpose |
| 10 | Google cloud services will be used to store player gameplay progress |
| 11 | Google cloud services will be used to get user information and progress |

## 6.2 Architecture Overview

Our architecture consists of several components working together to provide a seamless and interactive experience for the user:

* Unity engine for game development.
* Sign Language recognition feature to allow the game to recognize and analyze signs displayed by the user to the camera.
* Cloud technology to store user’s data.
* SQL will be used as the database management system to maintain and organize user data and progress.



### 6.2.1 Unity-Application

Our Unity application will be using the Model View Controller (MVC) architecture. This architecture separates the game's components into three main parts:

**Model**: contains the data and logic of the game, such as the game state, player progress, and speech recognition algorithm. It is responsible for updating and maintaining the game state.  
**View**: contains the user interface and presentation layer of the game, including the graphics, animations, and sound effects. It displays the game state to the user and handles user input.  
**Controller**: The mediator between the Model and View, receiving input from the user and updating the Model and View accordingly. It is responsible for game flow and logic, such as transitioning between game stages, managing player progress, and handling user input validation.

### 6.2.2 Scenes

Our Scenes - represent different social scenarios. We plan on creating interesting and exciting scenes to make an attractive video game. We are going to use a 3D world with various visual elements that would be intractable to create a more fluid experience, our goal is to create a simulation of what it would feel like to be hard of hearing, this would greatly improve the users understanding of the difficulties a person could experience when they are hard of hearing. The scenes will include minigames that would require the user to display signs in sign language, in early levels the signs that would be presented will be single words or a few letters and in more advanced levels a player might have to say entire sentences. We would like to create an open world environment that would allow players to explore and go to classes to learn new words, maybe even encounter NPCs (Non-Player Characters) and will have to communicate with them to get directions or complete tasks.

### 6.2.3 Interfaces and Game Flow

Main Menu:



Game Mode Selection:



Difficulty Selection:

A screenshot of a video game

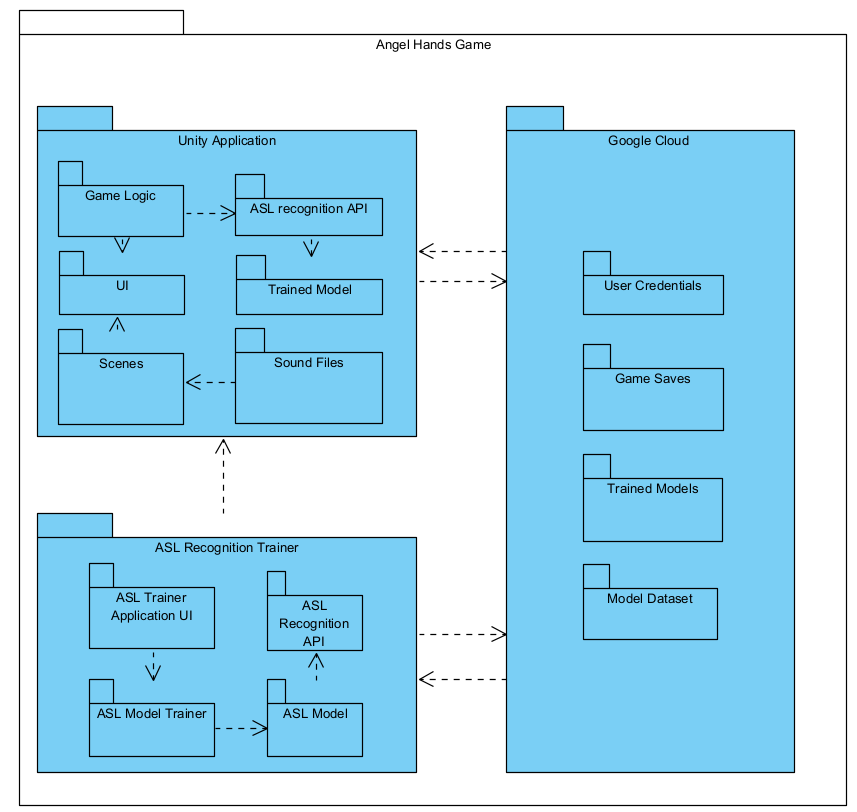
Description automatically generated

### 6.2.3 Data Management

Once A level is completed or a save request is issued, the progress will be automatically saved to the cloud. The stored information will include the user’s information, in game avatar, current level, points earned, challenges completed, current location in the open world. This would allow a user to continue with his progress from any place on any computer or mobile device.

## 6.3 Diagrams

### 6.3.1 Package Diagram



### 6.3.2 Activity Diagram

# 7. Verification And Evaluation

Verification and evaluation of a game are vital aspects of game development, ensuring that the game functions correctly, provides a satisfying user experience, and meets the desired objectives. Through evaluation, developers assess the game's overall design, gameplay mechanics, balance, and player engagement, while verification is a technical process that confirms the absence of bugs, errors, and performance issues.

In order to ensure that our educational game is effective in teaching sign language and that the sign recognition feature is working as intended we will implement the following verification and evaluation plan:

## 7.1 Testing plan

The testing plan is designed to ensure the quality and effectiveness of the application and the sign recognition feature. By thoroughly testing each of these elements, we can identify any potential issues or areas for improvement.

|  |  |  |  |
| --- | --- | --- | --- |
| Test No. | Module | Tested Function | Expected Result |
| 1 | Unity Application | Scene Loading | The scene should load correctly and without any errors |
| 2 | Unity Application | Scene Transitions | The transition between scenes should be quick and seamless |
| 3 | Unity Application | 3D Objects Positioning | The 3D objects should be positioned in the correct locations |
| 4 | Unity Application | 3D Objects Proportions | The 3D objects should have the correct proportions |
| 5 | Unity Application | 3D Object interactions | The user should be able to interact with different elements in the game. |
| 6 | Unity Application | 3D Object movement | The user should be able to move within the 3D world, there should be bounds that the user cannot cross, make sure the user doesn’t go beyond areas that are defined in the map |
| 7 | Sign Recognition | Correct Recognition | The system should recognize the signs correctly |
| 8 | Sign Recognition | Fast Recognition | The system should determine the sign that the user is presenting in a timely manner, it should be quick and responsive. |
| 9 | Sign Recognition | Handling incorrect signs | Testing to see that incorrect sign \ untrained signs are not detected and not labeled incorrectly, if a sign is incorrect, it shouldn’t display anything other than incorrect sign. |
| 10 | System | Camera Permissions | The game should ask for permissions from the user to use the camera the first time he uses the game. |
| 11 | System | Camera Availability upon startup | If the camera was not detected the user should be prompted to check it when the game starts |
| 12 | System | Camera Availability mid game | If while playing the game the camara connection is lost the user should be prompted that the camera is not working and that he is required to fix the issue before he can continue playing, he should be able to save the game and exit the game while being prompted to fix the camera. |

## 7.2 Usability Testing

The overall usability of the game will be tested, the test will include multiple people using different hardware (Computers, Cameras), the test will be performed in various lighting conditions as this might greatly affect the ASL recognition Accuracy. We would like to test on People of different age, color skin, size, wearing different clothing, in different environments.

Also, the user interface will be tested and reviewed to assure its simplicity.

# 8. References

1. [Gee, J.P. (2003). What video games have to teach us about learning and literacy.](https://blog.ufes.br/kyriafinardi/files/2017/10/What-Video-Games-Have-to-Teach-us-About-Learning-and-Literacy-2003.-ilovepdf-compressed.pdf)

2.[Oviatt, S. (1999). Ten Myths of Multimodal Interaction.](https://dl.acm.org/doi/10.1145/319382.319398)

3.[Starner, T., et al. (1998). Real-time American Sign Language recognition using desk and wearable computer based video.](https://www.researchgate.net/publication/3192935_Real-time_American_sign_language_recognition_using_desk_and_wearable_computer_based_video)

4.[Meng Wu (2024). Gesture Recognition Based on Deep Learning: A Review.](https://www.researchgate.net/publication/378817255_Gesture_Recognition_Based_on_Deep_Learning_A_Review)

5.[Nick Barney (2023). Gamification.](https://www.techtarget.com/searchhrsoftware/definition/gamification)

6.[Fadhlina Izzah Saman, et al. (2019). Sign Language Learning Application Via Gamification.](https://eric.ed.gov/?id=EJ1238764)

7.[What Is American Sign Language (ASL)?](https://www.nidcd.nih.gov/health/american-sign-language)

8.[How to learn sign language the best way – HandSpeak](https://www.handspeak.com/learn/7/)  
9.American Sign Language - [pic link](https://en.wikipedia.org/wiki/American_Sign_Language)

10.[Unity (game engine) , Wikipedia.](https://en.wikipedia.org/wiki/Unity_(game_engine))

11.Whats is Unity - <https://twitter.com/AaronChurchgcc/status/1591048109077168130>

12. [Holistic landmarks detection task guide](https://ai.google.dev/edge/mediapipe/solutions/vision/holistic_landmarker)

3. [Pose landmark](https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker)

4. [Hand landmarks](https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker/index)

15. [Face Landmark](https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker)

16.[Sign Language Recognition Application Using LSTM and GRU (RNN)](https://www.researchgate.net/publication/369857149_Sign_Language_Recognition_Application_Using_LSTM_and_GRU_RNN)

17.[American Sign Language Recognition Using Machine Learning and Computer Vision](https://digitalcommons.kennesaw.edu/cgi/viewcontent.cgi?article=1024&context=cs_etd)  
18.[Gamification By Nick Barney](https://www.techtarget.com/searchhrsoftware/definition/gamification)

# 9. GitHub Link

[Angel-Hands-Sign-Language-Learning-Game](https://github.com/EitanGerman/Angel-Hands-Sign-Language-Learning-Game)

1. [Gee, J.P. (2003). What video games have to teach us about learning and literacy.](https://blog.ufes.br/kyriafinardi/files/2017/10/What-Video-Games-Have-to-Teach-us-About-Learning-and-Literacy-2003.-ilovepdf-compressed.pdf) [↑](#footnote-ref-2)
2. [Oviatt, S. (1999). Ten Myths of Multimodal Interaction.](https://dl.acm.org/doi/10.1145/319382.319398) [↑](#footnote-ref-3)
3. [Starner, T., et al. (1998). Real-time American Sign Language recognition using desk and wearable computer based video.](https://www.researchgate.net/publication/3192935_Real-time_American_sign_language_recognition_using_desk_and_wearable_computer_based_video) [↑](#footnote-ref-4)
4. [Meng Wu (2024). Gesture Recognition Based on Deep Learning: A Review.](https://www.researchgate.net/publication/378817255_Gesture_Recognition_Based_on_Deep_Learning_A_Review) [↑](#footnote-ref-5)
5. [Nick Barney (2023). Gamification.](https://www.techtarget.com/searchhrsoftware/definition/gamification) [↑](#footnote-ref-6)
6. [Fadhlina Izzah Saman, et al. (2019). Sign Language Learning Application Via Gamification.](https://eric.ed.gov/?id=EJ1238764) [↑](#footnote-ref-7)
7. [What Is American Sign Language (ASL)?](https://www.nidcd.nih.gov/health/american-sign-language) [↑](#footnote-ref-8)
8. [How to learn sign language the best way – HandSpeak](https://www.handspeak.com/learn/7/) [↑](#footnote-ref-9)
9. [pic link](https://en.wikipedia.org/wiki/American_Sign_Language) [↑](#footnote-ref-10)
10. [Unity (game engine) , Wikipedia.](https://en.wikipedia.org/wiki/Unity_(game_engine)) [↑](#footnote-ref-11)
11. <https://twitter.com/AaronChurchgcc/status/1591048109077168130> [↑](#footnote-ref-12)
12. [Holistic landmarks detection task guide](https://ai.google.dev/edge/mediapipe/solutions/vision/holistic_landmarker) [↑](#footnote-ref-13)
13. [Pose landmark](https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker) [↑](#footnote-ref-14)
14. [Hand landmarks](https://ai.google.dev/edge/mediapipe/solutions/vision/hand_landmarker/index) [↑](#footnote-ref-15)
15. [Face Landmark](https://ai.google.dev/edge/mediapipe/solutions/vision/face_landmarker) [↑](#footnote-ref-16)
16. # [Sign Language Recognition Application Using LSTM and GRU (RNN)](https://www.researchgate.net/publication/369857149_Sign_Language_Recognition_Application_Using_LSTM_and_GRU_RNN)

    [↑](#footnote-ref-17)
17. [American Sign Language Recognition Using Machine Learning and Computer Vision](https://digitalcommons.kennesaw.edu/cgi/viewcontent.cgi?article=1024&context=cs_etd) [↑](#footnote-ref-18)
18. [Gamification By Nick Barney](https://www.techtarget.com/searchhrsoftware/definition/gamification) [↑](#footnote-ref-19)