INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA INFORMÁTICA




Modelo de um jogo de cooperação orientado para a comunidade surda

MARCELO TORRES NORBERTO
Outubro de 2016

POLITÉCNICO
DO PORTO

Instituto Superior de Engenharia do Porto

Cooperative Game Model Aimed at the Deaf Community

Marcelo Torres Norberto

Master degree in
Computer Engineering, Specialization
in Graphic Systems and Multimedia

Supervisor: Paula Maria De Sá Oliveira Escudeiro, PhD

Jury:
President:
[Nome do Presidente, Categoria, Escola]
Vowels:
[Nome do Vogal1, Categoria, Escola]
[Nome do Vogal2, Categoria, Escola] (até 4 vogais)

Porto, October 2016

Dedicatory

This thesis is dedicated to my parents who have shown an enormous amount of support from the beginning. It is also dedicated to my girlfriend Kacey who inspired and supported me.

Finally, this thesis is dedicated to all who have supported and believed in me during this period of my life.

Abstract

The rise of digital content and appearance of new devices capable of a more precise motion recognition creates an opportunity for developers to spread their work to a wider audience.

A cooperation game model for the deaf is presented in this master thesis. It's integrated within the VirtualSign project that consist on a bidirectional translator of sign language. The game consists of a first person puzzle game and requires two players to cooperate in order to get through the game. The game has a chat where users can type but it also makes use of the VirtualSign translator to allow deaf users to chat within the game using sign language. The cooperation is not only necessary but also rewarded as the players performance is represented through a score. The players can compare their performances as team with others through the high score list. The interface is fairly intuitive and allows the player to access all the core features needed for the game and the chat. The game story is also translated through the avatar. The game was designed for the deaf community and players aged above 12 years old.

It was tested by 10 individuals with ages ranging from 23 to 65 and the feedback was very positive allowing the improvement of a few features. The results of the users testing is represented on the QEF where the game scored a 95% quality rate. This game aims to improve the social inclusion of the deaf community and show that it's possible to communicate with them in games using their own language.

Keywords: Sign Language, Cooperation, Multiplayer Game, VirtualSign

Resumo

Com o aumento do conteúdo digital e o surgimento de novas tecnologias capazes de obter uma maior precisão na captura e reconhecimento de movimentos, surge a oportunidade para os criadores de software de expandirem os seus conteúdos para mais pessoas.

Nesta dissertação do mestrado é apresentado um modelo de um jogo de cooperação orientado para a comunidade surda. Este jogo está integrado no projeto VirtualSign que consiste num tradutor bidirecional de língua gestual. O jogo consiste na resolução de puzzles e é jogado em primeira pessoa. Dentro do jogo existe um chat de texto, no entanto permite aos surdos que usem o tradutor VirtualSign para falar em língua gestual. A conversa aparece sempre em texto e traduzida para língua gestual por um Avatar. A cooperação dentro do jogo não só é necessária como é recompensada através da pontuação dos jogadores. Os jogadores podem comparar as suas performances como equipa através da lista de melhores pontuações. O jogo apresenta vários desafios que só podem ser ultrapassados com a cooperação dos jogadores.

A interface é intuitiva e permite aos utilizadores o acesso a todas as principais funcionalidades do jogo e também ao chat. A história do jogo é também traduzida pelo avatar. Isto permite aos utilizadores surdos uma fácil interpretação do jogo, o que não acontece quando só é apresentado texto escrito.

O jogo foi criado com principal foco na comunidade surda, para jogadores com idades superiores a 12 anos. Os testes foram feitos por 10 jogadores com idades contidas entre 23 e 65 anos e o feedback foi bastante positivo, o que permitiu que fossem melhorados alguns aspetos funcionais do jogo. Os testes foram feitos em duas fases, alfa e beta. O resultado dos testes encontram-se representados no QEF. O jogo teve uma pontuação final de 95% utilizando o QEF. Este jogo ambiciona melhorar a inclusão social da comunidade surda e mostrar que é possível comunicar com surdos dentro de um jogo utilizando Língua gestual.

Palavras-chave: Língua gestual, Cooperação, Jogos multijogador, VirtualSign

Acknowledgments

I would like to thank Jorge Lopes for the support integrating the VirtualSign translator into this prototype.
I would also like to thank my supervisor Paula Escudeiro that supported me through all the phases of the project.
Finally, thank you to my friends and family that supported me through this project.

Index

Figure Index

Table Index

Code Index

Acronyms and Nomenclature

Lista de Acrónimos

ISEP	*Instituto Superior de Engenharia do Porto*
GILT	Games Interaction and Learning Technologies
QEF	Quantitative Evaluation Framework
PSL	Portuguese Sign Language
RGB	Red, Green, Blue
SDK	Software Development Kit
5DT	Fifth Dimension Technologies
CLR	Common Language Runtime
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
W3C	World Wide Web Consortium
CPU	Central Processing Unit
IK	Inverse Kinematics
SVM	Support Vector Machine
KNN	K-Nearest Neighbours

1 Introduction

Technology continues to advance and new opportunities arise with it. Access to this technologies is now easier than ever and yet there are some fields hardly explored as they should. One example is the use of technology for the social inclusion of the deaf. In Portugal alone there are around 30 thousand deaf and 120 thousand with some form of hearing disability (Associação Portuguesa de Surdos, 2011).

In order to improve the inclusion of the deaf, a cooperation game model aimed at the deaf community was created and tested. The game is part of the VirtualSign project in which the main component is a bi-directional translator of sign language (VirtualSign, 2013).

1.1 Problem

There is a limited amount of digital content available for the deaf community. However, due to recent year's technological evolution it became possible to apply new ways of using the technology to improve this issue. In this dissertation the focus will be the digital games. Some may assume that deaf can easily play any game that has text and subtitles, however, that is not entirely true. In fact most deaf can read and write, however, their sign language differs from the spoken language, they have different syntax and semantic which makes it harder to read (Padden & Ramsey, 2000). Therefore, the focus of this project is to create a game model based on sign language that allows two players to cooperate through sign language without having to read.

1.2 Dissertation Context

This manuscript was written for the dissertation of the Master's Degree in Computer Engineering, Specialization in Graphic Systems and Multimedia of the Instituto Superior de Engenharia do Porto (ISEP).

This goal is to allow the student to demonstrate the knowledge acquired during the Master's Degree by developing a project and applying good engineering practices during the process while documenting it.

The project is being developed in Games Interaction and Learning Technologies (GILT) and consists in creating a cooperation game aimed at the deaf community using the "VirtualSign" translator (Escudeiro, 2013) which is a bidirectional sign language translator. It allows the translation from gesture to text and text to gesture. Both features are to be used in the game to improve the gameplay and make it accessible and interesting to the target market. The game is being integrated with the current VirtualSign package that already has a serious game that teaches the basics of sign language, which makes it a promising opportunity to allow those who are learning to communicate with others and improve their sign language skills as well.

1.3 Value Analysis Overview

One of the main success factors of this component is the fact that there is no competition of cooperation games aimed at the deaf community, so with its innovative features and concept it has a high chance of obtaining success. Another aspect that may have a big impact is the fact the component is part of the VirtualSign which contains all the requirements for the game and removes the need to focus on distribution and marketing. On the other hand, the translator has a high cost associated with it which may be a handicap to the number of players that will acquire the game. However, with the constant evolution of technology it may be possible to drastically reduce the price in the near future, thus removing the main handicap of this component.

1.4 Approach

The project developed with this dissertation aims to address the issue of the lacking deaf digital content. The project consists of a digital cooperation game where the players can communicate with each other using sign language. The VirtualSign translator is going to allow communication using its 3D avatar and its sign recognition as those features will be integrated in the game.

The game is a first person puzzle genre where each player has to solve puzzles allowing each other to progress. Initially the scenes were designed and all the levels were created. At this stage some objects and textures were adjusted to improve the scenes. The levels had to be carefully designed as game design is crucial for puzzle games as some of the puzzles are built in the scene itself (Rouse III, 2010). As for the gameplay there is a score associated with the time it took to solve all the problems. Each player gets his own score based on how fast each puzzles were solved as well. The double score system grades players as a team and against each other increasing the competition but also creating a balance between competition and cooperation. Not only do the players try to beat each other scores but also the overall scores of other teams as well. This balance aims to aid the social skills of those who play the game as they are induced to communicate with each other to achieve their goals.

In order to control the development of the project a Quantitative Evaluation Framework (QEF) (Escudeiro & Bidarra, 2008) will be used throughout the whole process. The QEF allows access to the development progress as well as the final evaluation of the program. In a final stage there will be tests to analyze the game performance and validate the QEF requirements.

1.5 Contribution

The main goal of this dissertation is to contribute to the social inclusion of the deaf community through games.

This dissertation provides information on state of the art motion recognition software and hardware. It also provides a detailed description of the implementation of the multiplayer feature and the translator into the game as well as the core game mechanics. After the implementation details there are the evaluation results and how they were obtained.

1.6 Overview of the Dissertation

This dissertation has 7 chapters.

The first is the introduction, followed by the second chapter Context that describes the context of the project created for this dissertation.

The third chapter is the state of the art where there are the analysis of the most recent software and hardware that are related to the topic of this dissertation.

Chapter 4 represents the approach used in order to create the game as well as its requirements.

Chapter 5 consists in the design and architecture of the system of the game such as the diagrams, art and concept.

Chapter 6 shows how the game was implemented and the results obtained from the implementation in both Alpha and Beta phases.

Finally, there is the chapter 7 containing the conclusions and future work for the project.

2 Context

The idea to create a new cooperation game aimed at the deaf was inspired by the VirtualSign project. VirtualSign already has a serious game that helps with the learning of sign language. Since there is a lack of multiplayer games for the deaf there was an opportunity to develop this cooperation game model.

2.1 Business concept

The cooperation game model for the deaf community emerged from the opportunity to integrate this model within the VirtualSign project. It is a complement to the product making it so that the users who acquire it have all the means to make a complete use of it. The ideal business model for this product is to be sold as part of a package that is the full VirtualSign set of products. Therefore, the client acquires the VirtualSign translator and the game comes as part of the package. Since VirtualSign already has a serious game teaching the basics of sign language, the client who acquires the package can suggest it not only to the deaf community but also to his non deaf acquaintance, thus, improving communication between each other by playing together. There is a mutual benefit between the translator and the game as the game helps publicize the translator and the translator allows the clients to play the game. Since this model is a component of a package the marketing and distribution will be handled by VirtualSign.

2.2 Process and actors

The game was developed in GILT-Game Interaction and Leaning Technologies by the Master's Degree student Marcelo Norberto. There were tests made by anonymous users to validate the quality of the program. The translator is being developed by Jorge Lopes and Fernando Soares who provided the necessary means to integrate the translator with the game. The project

coordinator is Professor Paula Escudeiro, she guided and validated the full development of the project as well as the corresponding documentation.

2.3 Existing restrictions

This project has a number of restrictions. Since it's a component of VirtualSign it requires the use of the translator. The VirtualSign translator, currently, needs a pair of 5DT sensor gloves and the Kinect. Also the translator only works on the Windows operative system limiting the game to be single platform. As for the code, it is close source as it will be integrating the VirtualSign.

The module gesture to text can only be utilized in windows. The module gesture to text has another restriction, this restriction exists due to the necessity to train the module before using it. That training will improve the percentage of accuracy but takes some time to make.

2.4 Business model

The canvas business model (Joyce, et al., 2015) shows the main aspects of the business such as the revenue streams, partners, costumer segments and what is really provided to the client and what are the costs.

The canvas for this project can be seen in Figure 1.

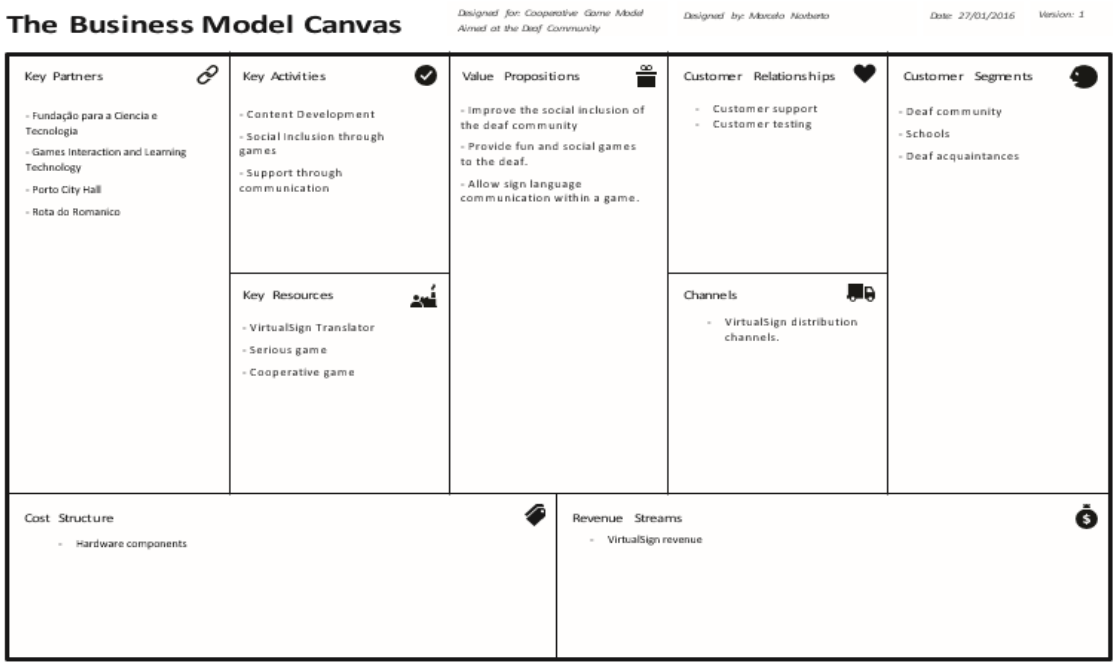


Figure 1 - Canvas Business Model of the project.

3 State of the Art

3.1 Definition of Cooperative gameplay

Cooperative gameplay also referred to as co-op consists of a feature of games where two or more players need to help each other in order to progress in the game. Cooperative games have intrinsic mechanics that causes the players to help each other by using game features such as chats, trades or even the gameplay itself as a user can adjust the way he or she is playing to help another player.

Players can either cooperate locally or through a network depending on the game, as for this dissertation game, it allows both.

This kind of gameplay's popularity has been increasing over time (Kuchera, 2009). Several games have specific scenes and features only unlockable through the cooperation mode such as *"Streets of Rage"* (Steam, 2011). Most of the games nowadays have a co-op mode where the player can go through the game campaign with the help of another.

The early game systems had technical limitations that restrained the game creators to implement co-op in order to keep a good performance. With the evolution of this systems and increase of computer gaming, more and more features could be added and co-op was one of those. Games like "Gauntlet" in 1985 introduced 4 players co-op (The International Arcade Museum, n.d.). Shooters also started using co-op from platform shooters like "Metal Slug" (SNK Playmore, 2008) to first person shooters like "Call of Duty" (Activision Publishing, Inc, 2016). Co-op has been integrated with several kinds of game types other than shooters though, such as Sports games, Role-playing games, Puzzle games and a few more types.

Co-op can be either local or remote, local co-op, also known as couch co-op, means that the players are using the same gaming device and sharing the same screen to play the game. Usually this requires a split screen where the screen is shared in half displaying each player view on each half. An exception to the split screens is if the game is either side view or top view where both players share the exact same area. There are even games that implement both, such as

the “Lego Marvel Super Heroes” (LEGO, 2016) where the game adjusts the screen, if the players are in the exact same area the screens shows just one view over all the game scene but if a player moves away from the other the screen smoothly splits into two showing the views of both players in half the screen. On the other hand, the remote co-op implies that each player is playing from their own devices through a network. There are games that allow both local and co-op where a few users can be sharing one game device and play with another user through a network like *“Little Big Planet”* (SONY, 2016) or *“Broforce”* (Free Lives, n.d.).

3.2 Similar Games and Sign Translator Software

There are some projects related to the topic of this dissertation but none of them implies an automatic bidirectional translation process in a game as this does, thus, making this project very innovative. There is a rising number of serious games projects. Some of the most relevant related works within the sign language scope is described below.

3.2.1 Games Using Sign Language

The game *CopyCat* (Zafrulla, 2011) consists of a game where sign language gestures need to be executed properly in order to proceed. The movement analysis is done through gloves with sensors. However, the researchers from the *CopyCat* project have published a video where they show their intention to use Kinect for movement detection. Their current research platform is a custom system that uses computer vision, colored gloves and wrist-mounted 3-axis accelerometers. These are used to collect data as users sign and machine learning is used to recognize the signs for game play. The system was built on top of Ubuntu Linux and uses the Kinect system as input for the computer vision, which replaces the gloves and sensors.

Most of the sign language games have no recognition of the signs and are just small games that aim to improve learning sign language by using memory games showing gestures and asking questions without any kind of sign recognition (Start ASL, 2016).

3.2.2 Sign Language Translators

ProDeaf is an application that does the translation of Portuguese text or voice to Brazilian sign language (ProDeaf, 2016a). This project is very similar to one of the main components used on the VirtualSign game, which is the text to gesture translation. The objective of *ProDeaf* is to make the communication between mute and deaf people easier by making digital content accessible in Brazilian sign language. The translation is done using a 3D avatar that performs the gestures. *ProDeaf* already has over 130 000 users (ProDeaf, 2016b).

Kinect Sign Language Translator (Microsoft Research, 2013) is another project that is similar to the VirtualSign translator. The project was a result of collaboration facilitated by Microsoft

Research, between the Chinese Academy of Sciences, Beijing Union University, and Microsoft Research Asia. Researchers in China have created a prototype system that understands the gestures of sign language and converts them to spoken and written language and vice versa. The system captures a conversation from both sides as it recognizes the signer and renders a written and spoken translation of the sign, and it also translates spoken words and turns them into sign language. An avatar is used to represent the gestures in sign language.

Showleap is a recent Spanish Sign language translator (Showleap, 2016), it claims to translate sign language to voice and voice into sign language. So far Showleap uses the Leap motion which is a piece of hardware capable of detecting hands through the use of two monochromatic IR cameras and three infrared LEDs and showleap uses also the Myo armband (Thalmic Labs, 2016) . This armband is capable of detecting the arm motion, rotation and some hand gestures through electromyographic sensors that detect electrical signals from the muscles of the arm. So far Showleap has no precise results on the translation and the creators claim that the product is 90% done (Showleap, 2015).

Project Giulia (MAP Technology, 2015) is a similar translator to Showleap but for Libras (Brazilian sign language) instead of Spanish. Project Giulia uses the Myo for gesture recognition and connects it to a smart phone with an application that handles the connection. The application converts the recognized gestures into text and voice and also it has voice recognition.

Motionsavvy Uni is another sign language translator that makes use of the leapmotion (Motionsavvy, 2016). This translator converts gestures into text and voice and voice into text. Text and voice are not converted into sign language with Uni. The translator has been designed to be built into a tablet. Uni claims to have 2000 signs on launch and allows users to create their own signs.

Two university students at Washington University won the Lemelson-MIT Student Prize by creating a prototype of a glove that can translate sign language into speech or text (University of Washington, 2016). The gloves have sensors in both the hands and the wrist from where the information of the hand movement and rotation is retrieved. There is no clear results yet as the project is a recent prototype.

3.2.3 Puzzle Cooperation Games

Portal 2 (Valve, 2011) is the game that inspired the gameplay style used in this game. Portal 2 is a first-person puzzle-platform video game developed and published by Valve Corporation on April 19, 2011. The first Portal game was released in 2007 for Microsoft Windows, OS X, Linux, PlayStation 3, and Xbox 360. In Portal 2 the Player takes the role of Chell in the single-player campaign or as one of two robots, Atlas and P-Body, in the cooperative campaign. The game characters can resist a limited damage but will die at some point. Falling on a solid surface within the game is safe, but falling into pits or toxic pools kills the player immediately. In the single-

player game, if the player dies the game restarts from a recent checkpoint. As for the co-op if the player dies he or she spawns again shortly afterwards without restarting the puzzle. The goal of both campaigns is to explore the Aperture Science Laboratory which is a malleable mechanized maze.

The Trine franchise created by Frozenbyte (Frozenbyte, n.d.) is another example of a successful action puzzle game, with three games released and a smart use of game mechanics that motivate the players to cooperate with each other. It allows up to three players to play simultaneously and each player has to control one of the three different characters, however the game allows the players to switch characters with each other if both agree to do so. This Allows for a player who is struggling to switch with someone else and have them try and get past the puzzle. Each character has a unique set of skills that are needed for specific parts of the level.

Most games have co-op as an extra optional feature but there are those that have only the cooperation mode. One example of a fully co-op game is being developed in Portugal by an indie studio called Arcade Thumb and the game is Crystalverse (Correia, 2016), it is a two players puzzle solving game.

3.3 Portuguese Sign Language

Sign language is the most used mean of communication by the deaf.

The Portuguese Sign Language (PSL) has ten thousand confirmed words in the PSL official dictionary which was edited by Porto Editora and written by Ana Bela Baltazar (Baltazar, 2010). Casa Pia in Lisbon was the first deaf school in Portugal (surdos, 2011). PSL evolved from Sweden sign language as the first PSL teacher in Portugal was from Sweden. PSL then adapted and continued to evolve thanks to the Portuguese deaf community.

In Portugal there are approximately one hundred thousand deaf. Sign language is not only used by deaf but also by those who are close to them. There are different ways the human being uses to communicate but usually the main form of communication is based on the language. For the deaf it is the sign language. Sign language is represented though hand and body movements and facial expressions and it possesses its own vocabulary, grammar and syntax (Correia, 2012).

Spoken languages are naturally evolving based on the region it is used, sign language goes through the exact same process therefor every country has their own and there are even different dialects of sign language within countries.

In order to speak in sign language there are a few elements that should be taken into consideration such as:

- Hands configuration;

- There is a dominant hand and a support hand that depends on the person;
- Hands movement;
- Hands orientation;
- Facial expression;
- Body movement;
- Sign languages have a limited number of hand configurations, in PSL there are 57 different ones.

Those elements combined can then form letters, words or sentences.

Sentences in sign language have a different syntax than its corresponding written language. In written Portuguese the sentences follows a structure where the subject comes first, then the verb and then the object whilst in sign language the structure is subject, object and then verb. In some cases the subject can be omitted if it is the speaker itself, such as “Eu vou para casa” would translate into “casa vou”, the “eu” which means I was omitted and then there is “casa” which is the object and finally there is “vou” which is the verb (Correia, 2012).

3.4 Technologies

3.4.1 Kinect

Kinect (Microsoft, 2016) is a motion capture device created by Microsoft and originally was sold with the Xbox 360 console.

It was released in Europe on November 10th of 2010. (Whitworth, 2010). The device allows the user to interact with the game without any controller.

Kinect has five main features:

- RGB (Red, Green, Blue) Camera that allows facial recognition;
- Depth sensor (Infra-red) that allows a three dimensional analysis;
- Microphone to obtain audio;
- Integrated processor and its own software;
- Recognition of 48 articulation points of the human body;

The RGB Camera and the depth sensor are highlighted in Figure 2.

Figure 2 - Microsoft Kinect with the infrared projector, the RGB Camera and infrared sensor highlighted.[1]

In February 21th of 2011 Microsoft announced the launch of a SDK (Software Development Kit) for windows (Microsoft, 2011), it was released for windows 7 on June 16th (Guthrie, 2011) of the same year in twelve countries.

The SDK included Kinect drivers compatible with the windows 7 and allowed programmers to develop applications for windows with the Kinect.

The main functionalities that the Kinect SDK allows developers to use include:

- Access to the low level streams from the depth sensor, color sensor and the microphone;
- Tracking of the skeleton, meaning the capacity of following the image of the skeleton up to two persons moving within the Kinect field of view;
- Access to the advanced audio capacities that Kinect has;
- Copies of documentation and code;

3.4.2 5DT Gloves

5DT Gloves are data gloves produced by Fifth Dimension Technologies (5DT). The company has two sets of gloves one with 5 sensors per glove and another with 14 (5DT, s.d.). With those gloves it is possible to track the hands configuration. In VirtualSign the gloves used have 14 sensors, 2 sensors per finger plus one in between fingers. The gloves connects to the computer via a USB cable. The material of the gloves is lycra so they can stretch and be comfortable. 5DT provides development kits for Windows, Linux and Mac OSX as well as an optional wireless kit upon purchasing the gloves. Each glove has a cost of 5 495 USD and the wireless kits costs 1495 USD. Figure 3 shows both gloves and the kit.

[1] http://www.codeproject.com/Articles/317974/KinectDepthSmoothing

Figure 3 – 5DT Left and Right gloves with wireless kit.[2]

3.4.3 Autodesk Maya

Autodesk Maya (Autodesk, 2016a) is a 3D modeling software with a vast set of features. Maya has won several awards over the years and so have products like movies and video games that used it (Autodesk, 2016b). This software is available on Windows, OS X and Linux and it was created by Alias Systems Corporation but is currently owned and developed by Autodesk, Inc. Other than 3D modeling Maya allows a specific kind of scripting called Maya Embedded Language.

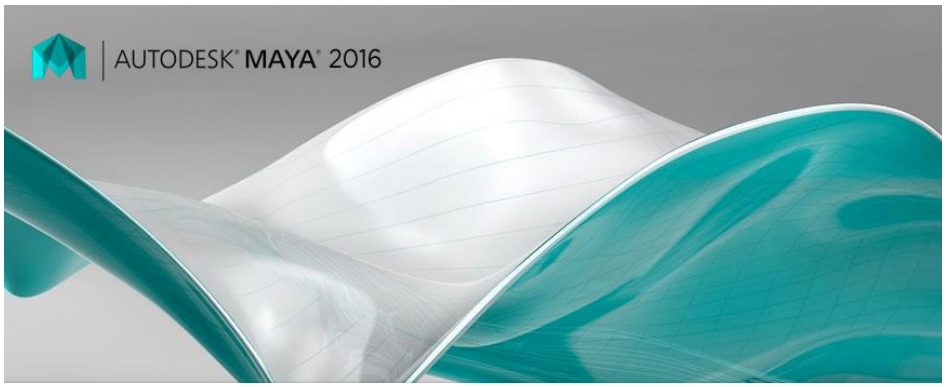


Figure 4 – Autodesk Maya 2016 entry screen and logo.[3]

[2] http://www.5dt.com/?page_id=34
[3] http://blog.animationmentor.com/6-things-to-know-about-maya-2016/

3.4.4 Unity 3D

Unity 3D (Unity Technologies, 2005) is a game engine and IDE created by Unity Technologies, developed in C++ on its core and is available for Windows, Mac OS and Linux (BARD, 2015).

Unity is currently capable of exporting to 21 different platforms (Unity Technologies, 2016a) and it uses a vast set of APIs to be able to do so, for example for rendering it uses OpenGL, OpenGL ES and Direct3D depending on the target platform. On its core unity has a powerful game engine capable of handling complex textures, 3D models, animations, lighting, collisions and scripting.

As for the licensing unity has four different license types which are Personal, Plus, Pro and Enterprise. Each license gives the user access to different features. The Personal license is free but there are some limitations, for instance if the product profits more than 100 000 USD unity requires a license upgrade to plus increasing the limit to 200 000 USD or changing to Pro or Enterprise that have no limit. The biggest differences between licenses are the Pro features that are available in all licenses except the personal one and then the source code access which is only available for Pro and Enterprise (Unity Technologies, 2016b). Unity provides its users with a large set of assets to be used in their games, there is also an asset store where people can share their assets or sell them to others.

Scripting in unity can be done though C#, JavaScript or Boo. An editor named MonoDevelop comes with the unity but the users can chose to use their own for instance you can use and even debug Unity projects using Visual Studio on windows.

The scripts use Mono as the base, which is an open-source implementation of the .NET Framework.

Figure 5 - Unity 3D main Interface and logo.[4]

[4] http://www.rkdesigns.org/2015/04/15/ai-typing-in-unity-3d-ui/

3.4.5 Microsoft Framework.NET

The Microsoft .NET Framework is an attempt from Microsoft to create a unique development platform.

This platform is able to comprehend over 20 different coding languages and have them interact with each other as the Framework is ran in a Common Language Runtime (CLR) (Microsoft, 2016).

The Figure 6 shows the structure of the .Net Framework.

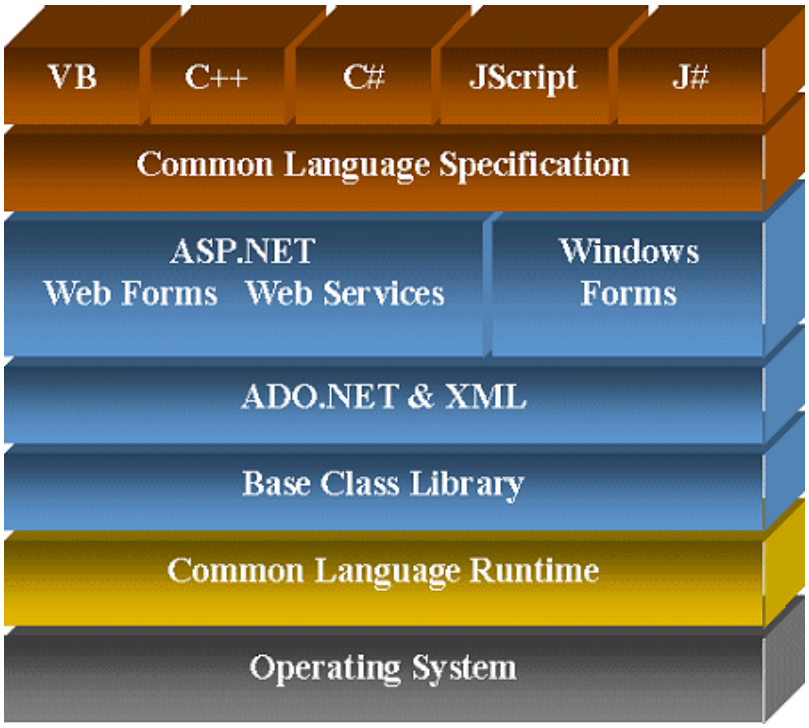


Figure 6 - .NET structure layers[5]

3.4.6 MySQL

MySQL was first released in May 21st 1995. It is an open-source database management system developed in C and C++. The source code is available under a GNU General Public License. MySQL is currently being developed by Oracle Corporation. A large set of platforms support MySQL (MySQL, 2016a) and there are several large companies using it such as Facebook, Twitter and YouTube (MySQL, 2016b). LAMP which stands for Linux, Apache, MySQL, Perl/PHP/Python, and it is one of the most used open-source web applications development kits. The official graphical interfaces provided with the database is the MySQL Workbench which allows easy access to basic database features.

[5] https://msdn.microsoft.com/en-us/library/ms973842.aspx

3.4.7 Sockets

Sockets are used to send messages through a network. A socket must always have a local IP address and the port number to where it will be sent and it also must have a transport protocol for instance Transmission Control Protocol (TCP) or User Datagram Protocol (UDP).

Sockets have a specific flow. For instance in a client-to-server model the sockets from the server wait requests from the client. The socket must be bound to an address that will allow the client to find the server and once the binding is done the server starts waiting for a request form the client. Once the client is connected to the server they can then start exchanging data (IBM, n.d.). The Figure 7 represents the flow mentioned above.

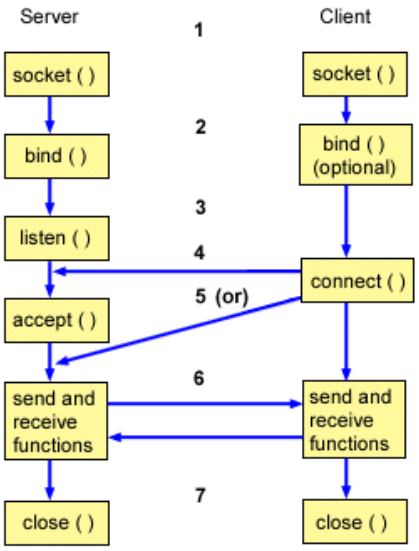


Figure 7 – Standard flow of a connection-oriented socket session.[6]

3.4.8 Web Services

The World Wide Web Consortium (W3C) defines web services as: “A Web service is a software system designed to support interoperable machine-to-machine interaction over a network.” (W3C, 2004).

A web services is hosted on the web as the name suggests and works based on requests and replies to the service. For example an application can use an HTTP request with parameters using POST and the web service will reply accordingly returning a message usually in XML or JSON but it can be any type of string depending on the implementation of the service. The big advantage of this kind of services is that any application in any platform can use it.

The figure 8 shows the basic flow of a Web service implementation.

[6] http://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/rzab6/howdosockets.htm

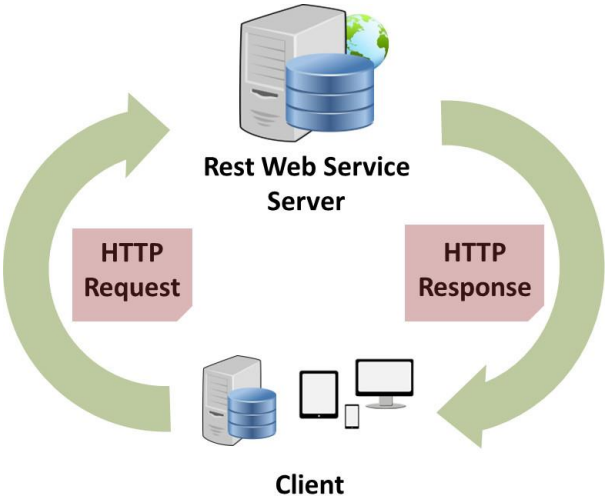


Figure 8 – Web service flow of events.[7]

[7] http://www.codeproject.com/Articles/841330/Creating-a-Web-Service-to-be-consumed-by-connected?msg=4968162

4 Approach and Requirements

4.1 Approach Evaluation

In order to evaluate the cooperation module there are three main aspects that are measured. The user satisfaction is the main focus as the product of this dissertation is a game and as such, it should provide the users with fun and pleasure, thus, filling the user's leisure time and inducing them to return and play again. Another aspect that was measured is the precision, in concrete for this scenario the precision of the translation. The reason the precision is crucial is because communication is the base of a good cooperation game and the precision of the translation will influence that communication between players. Shall the players fail or take too long to send the intended message because of an improper translation, they will gain frustration and eventually lose interest in the game. Finally the other aspect that is measured is time, just as in precision, the time that takes for the users to communicate will affect the quality of the idea that player wants to transmit. Since this cooperation module will use an online connection, several factors must be taken in consideration to assure the flow of the game and chat. The translation itself is nearly instantaneous, however, sending it and receiving in it on the other end may not be. The game must be optimized to not overload the network thus reducing the chances of latency.

The QEF was used in order to evaluate the progress and the project itself. With the QEF a set of functional requirements is established and each requirement has an associated weight corresponding to its importance. The evaluation of each requirement must be very specific there is low to no variation regardless of who evaluates it. The QEF is also supported by user surveys that will validate the subjective parts of the project such as the user satisfaction. These measures are necessary to adjust the project in order to obtain the best possible results according to the three main aspects that will be measured.

4.2 Hypothesis

The creation of a cooperation game for the deaf may improve the social integration of this community by having them communicate with both deaf and non-deaf through a game. One of the most successful cooperation puzzle games of all time is Portal. So this module gameplay is inspired on the same style. Different kinds of environmental puzzles were tested but the main issue was the network. As for the connection there are several possibilities that were tested such as using the Unity's network service or using a third party networking software to sustain the multiplayer. The reason different network services must be tested is based on the fact that latency causes frustration in the players and therefore must be avoided by selecting the best possible networking option.

As for the communication different approaches were used to see where the translation should be displayed in order to avoid having a negative impact on the gameplay. For instance it could be the players own avatar performing the gestures but it would become troublesome if the other player wasn't looking at it, therefore, different approaches were tested to see which one is most efficient.

4.3 Functional requirements

The functional requirements are imperative while developing any kind of software as it influences directly how the application is designed which can have an impact on all the other aspects of the developing process. The functional requirements identified for this application are:

- Start Menu where the player can start the game, consult the options or exit;
- Options Menu where the player can change the graphics quality, volume, translation speed, see the table of high scores and the credits;

The game requirements within the levels are:

- Handling and control of the character;
- Interact with game objects.
- Access to the options menu;
- Chat using the VirtualSign translator;
- Exit to the Start Menu;

4.4 Non-functional requirements

Just like the functional requirements the non-functional have an impact on the design and development of the application so they were identified.

4.4.1 Usability

This project aims to be fairly intuitive, allowing easy adaptation and learning. The interfaces were developed with the care to enable a pleasant interaction. The character controls are taught to present a simple usage. Along the game there are several short explanations of how the player should act to complete the puzzles and surpass the levels.

4.4.2 Performance

The game performance is a factor of the utmost importance, any perceptive delay can affect the gameplay making the game annoying rather than fun. With this in mind the game was developed aiming to be ran at an average of 60 frames per second on the recommended hardware. Besides the code all the factors that constrain the performance of the game must be taken into account, such as textures, bumps, number of vertices of the 3D models among others.

Also, as this game is a cooperation game over the network several precautions must be taken to avoid latency. In order to do so interpolation and extrapolation algorithms will be applied to the project to improve the smoothness of the game and several tests with different networking systems will be made as mentioned on the hypothesis.

4.4.3 Security

All the users' data must be protected at all times, thus making security a must for this application. The client will log in using the VirtualSign server where the web services and database are hosted. The server has its own security restrictions and the data sent from the client to the server is encrypted to ensure there is no leak of information.

4.4.4 Hardware and Software

In order to be able to run the game developed alongside with this dissertation there are some software and hardware restrictions and recommendations for a better use experience. Most of the restrictions are due to the hardware used though the details can be seen in Table 1.

Table 1 – VirtualSign cooperation game specification requirements

Specifications	Minimum	Recommended
Central Processing Unit(CPU)	Processor supporting SSE2 instructions.	Intel Core i7-2630QM equivalent or higher
Graphic Card	Graphic card with DX9 (shader model 3.0) support.	Graphic card with DX11 support (Radeon HD 6770M or higher)
Hard Disk Drive	1 gigabyte free	1 gigabyte free
Random Access Memory	2 gigabyte DDR2	8 gigabytes DDR3 or higher
Operating System	Window XP SP2	Windows 7 or higher
Network	Internet connection	High speed Internet Connections (10 Mb/s or higher)
Other hardware	None	VirtualSign kit (5DT gloves and Kinect)

The game can be played without the VirtualSign kit but the player won't have access to the gesture recognition, leaving typing as the only way to communicate. The recommended specifications hardware are based on the device the application was developed and tested.

5 System design and Architecture

5.1 Game Concept

The game aims to be an educational and social integration tool to improve the user's sign language skills and allow them to communicate with others using those skills.

The game is a first person puzzle game named Puzzatron, where the puzzles are based on simple form objects such as cubes and spheres. The Player will be motivated into solving the puzzles using the surrounding environment. The objects can be moved by the user and there is also intractable objects such as buttons and switches. However, some of those objects are only accessible to one of the players, thus, creating the need to cooperate with each other in order to finish each level.

After the players complete the level, the score will be registered on an in game worldwide ranking table with others players scores to incite competition and increase the players' motivation to perform better in order to beat the other's scores. There is also a personal score that each player achieves so there is competition between the two players as well. The storyboard of the game can be consulted in Annex A.

The communication between the players will be shown as gestures in real time but there will be also a text chat where the previous messages can be accessed.

The game will be available for Windows only due to the restrictions of the VirtualSign Translator and its associated hardware.

5.2 Concept Narrative

Several years into the future a group of scientists were involved in a social experiment project that intended to test human subjects and their ability to communicate and cooperate without sounds. So they designed a set of puzzles that required two people to complete them in a closed

environment where the two subjects are put to the test. The player character enters one of the puzzle rooms and then another player joins.

The players are given instructions initially on how to interact with the environment, each character is told that the they have been implanted with a device that will only allow them to move cubes that matches their color and after that the real challenge begins. The players must complete the puzzles as fast as they can and the only hope they have is to help each other out. Once all the puzzles are solved they pass to another room and so on until all the rooms are beaten and the character is free from the experiment, however, as they exit they will see the names of who got out faster and reach freedom from the experiment. Will their name be on the list? It all relies on their abilities and communication.

5.3 Game Mechanics

The gameplay is based on the interaction between the user, the translator and the keyboard. To progress in the game the player will need to succeed in completing the proposed puzzles. These puzzles must be resolved through the use of the scenes environment and cooperation (Creighton, 2010).

The main interactions are pushing and pulling objects. Also, the user will be able to pick up objects and while they are picked the user may walk with them. To pick up an object the user must click it while it is centered on the screen. Then without letting go of the mouse the player can drag the object along.

Then there is the interaction with buttons and switches that also are locked to the corresponding player. In order to interact with a button or switch the player needs to have it centered on the screen and click it. The switch will give the user both visual and audio feedback whether it has been pressed or not.

Each player has his own special ability that is based on the attribute selected before the match. There are the Light and Dark attributes. Each provides the player with a special ability that needs to be used in specific parts of the scene. The light attribute creates a beam of light that will unlock doors. The Dark attribute creates a dark cloud that covers surveillance cameras on the scene.

All the interactions will be done by pressing the interaction key on the keyboard or mouse.

The players will be able to jump as some of the challenges will require them to reach high spots. This functionality will also be controlled by a keyboard key.

One of the game elements that requires the player to jump is triggered by pressure plates. The pressure plate mechanic is fairly simple. One of the players walks on top of a platform that detects the collision and triggers an event which raises a different platform on the other player side.

The game is designed so that each player stays on one side of the scene separated by a transparent glass. The players can see each other through the glass so they know when to interact with the different game challenges.

A scoring system is present in the game and it implements a timer, so there is a direct connection between these two mechanics as the score is greater as the user takes less time to solve the puzzles.

The game implements also a login system for data storage and for the High score.

The player movement will be controlled by keyboard keys also.

The rotation of the character will be controlled with the mouse as a standard first person controller.

The most important mechanic of this project is however the cooperation. The game implements a full cooperation mode where the two players are able to see each other characters and communicate as their actions influence each other.

Finally, the chat that allows the players communication will be controlled either by typing or with the VirtualSign hardware, the gloves and the Kinect. The message will be displayed on the screen and the avatar will translate it. The avatar translation window can be dragged so it does not negatively interfere with the gameplay.

The mechanics are explained with more detail on Chapter 6 where the implementation is described.

5.4 Class Diagram

In order to understand how the application coding works there are a few aspects that must be taken into consideration. All the classes in Unity are extensions of the class Object, *ScriptableObject*, Component or *MonoBehaviour* (Unity Technologies, 2016c), thus, the reason why the classes are all connected directly or indirectly to *MonoBehaviour* in Figure 9.

5.4.1 Classes Overview

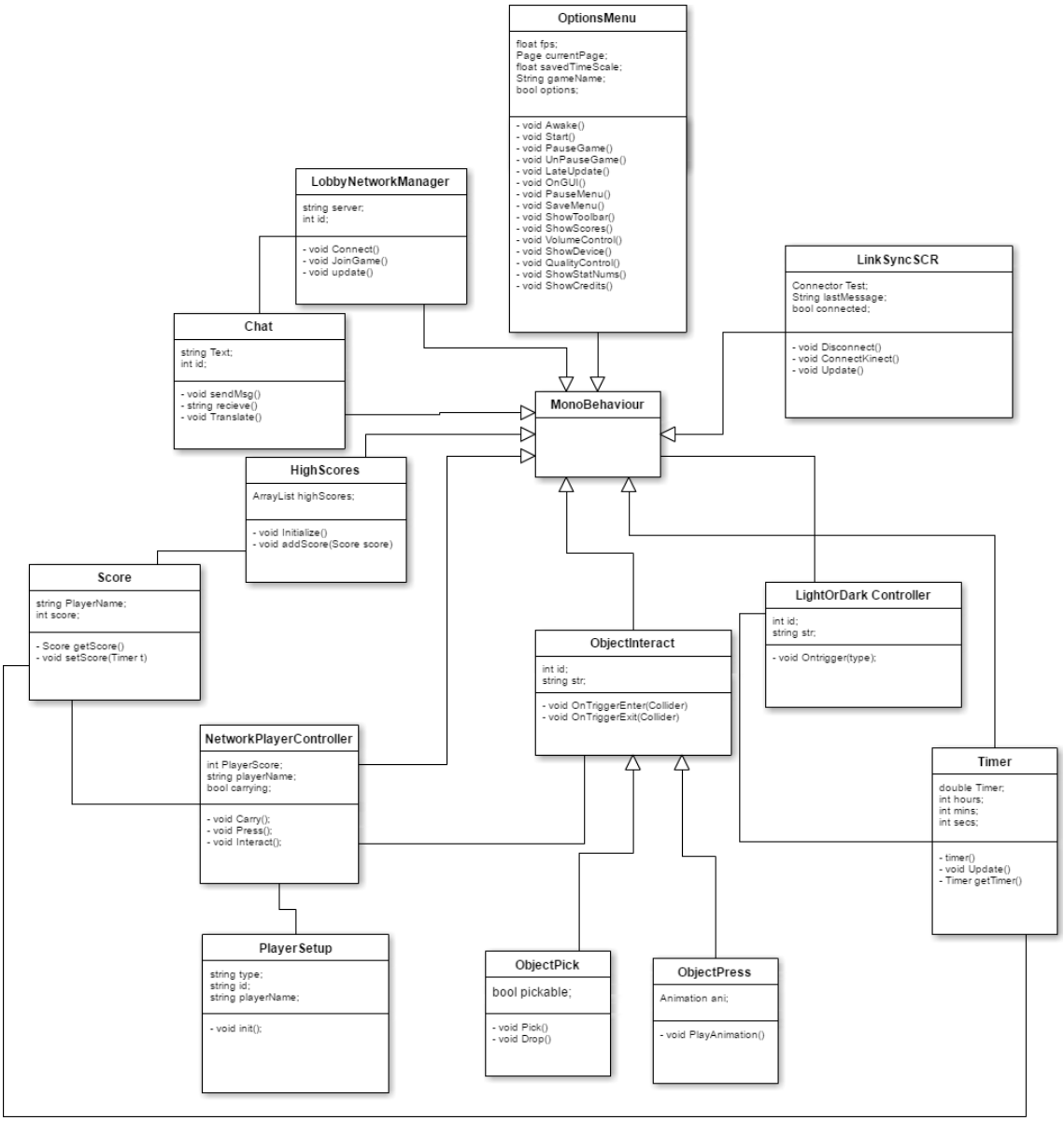


Figure 9 – Class Diagram of the core components of the cooperation module

The *OptionsMenu* stores and keeps information about the game main aspects such as graphics quality, audio level and the translation speed. Functions that are not controlled by the player character, such as loading a level, spawning an object in a specific situation are controlled by the *LobbyNetworkManager*.

The player Controller is where the actions regarding the player character are kept and it has access to information such as the current score, the time during the game, what is the player doing at the moment, for instance, if the player is carrying an object or not. The *LobbyNetworkManager* also handles the connection between the two clients and sends messages over the network both with the chat and the gameplay information.

The Figure 10 shows the simplified unity class diagram of the classes mentioned above to facilitate the understanding of the Unity class system.

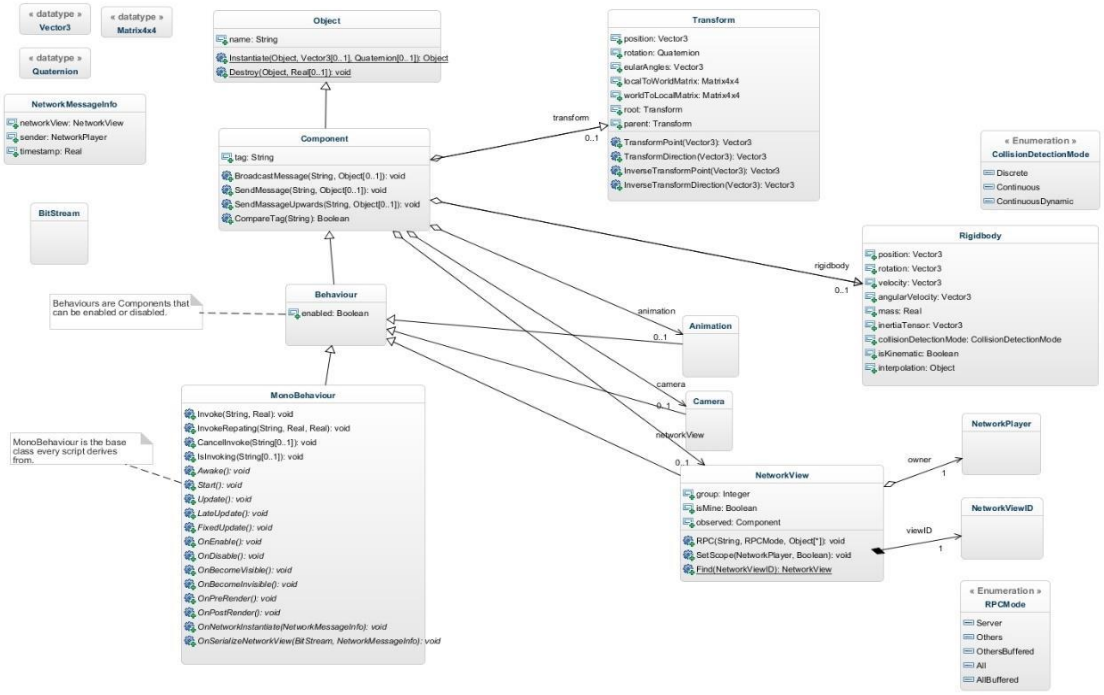


Figure 10 – Unity main class's diagram[8]

5.4.2 Networking Classes

The networking logic differs greatly from the standard single player coding structure.

Using Unity built-in networking there are a few different ways to communicate with clients and the server. So the class diagram in Figure 11 represents the implementation based on that networking logic.

[8] https://repository.genmymodel.com/rajaya/Unity

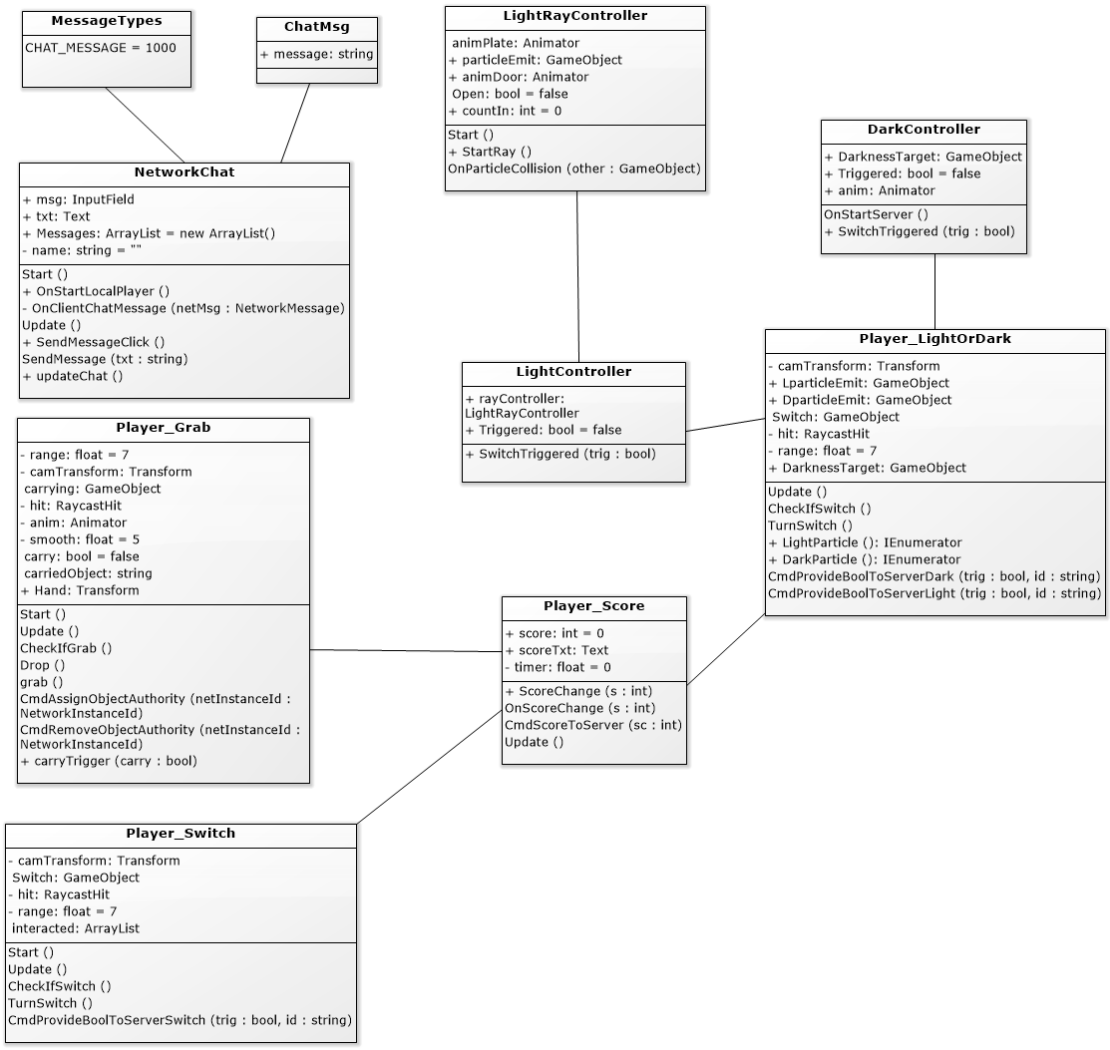


Figure 11 – Network Class Diagram

There is a class for every action of the player that should be sent over the network and there are three classes for the chat. The *NetworkChat* broadcasts the messages to all clients using the *MessageType* has an ID to identify that the specific message is part of the chat and uses the *ChatMsg* that contains the message itself. All the scripts with “player” in the name are attached to the player object itself in Unity. This decision was based on the fact that unity networking commands to the server must be called through the player object. Thus the creation of scripts for every action that are synced across the network and then call triggers on other scripts for example the light and dark controller. The movement and rotation of the avatar is sync through unity’s *NetworkTransform* and *NetworkRotation*. The creation of a script for each of these features were done however the unity native functions proved more efficient.

5.5 Art

The user interface is divided in two sections: the Lobby menu and the in-game interface.

The main menu contains the following options as can be seen in Figure 12:

- *Create*, the create option creates a game session on the server so that all the players can see and join;
- *List Servers*, Lists all the available servers;
- *Play and Host,* Allows the player to create a local match;
- *Join*, Joins the match with the corresponding IP address;
- *Exit*, Exits the game client;

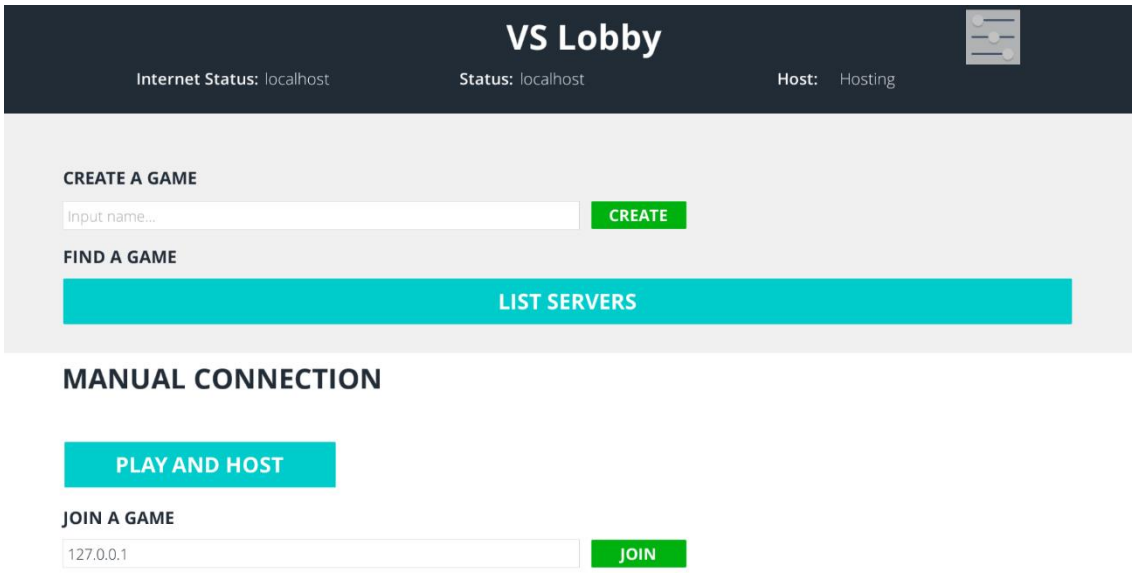


Figure 12 - Interface of the Lobby menu

The icon on the top right of the Figure 12 is the options buttons that opens the options menu represented in Figure 13.

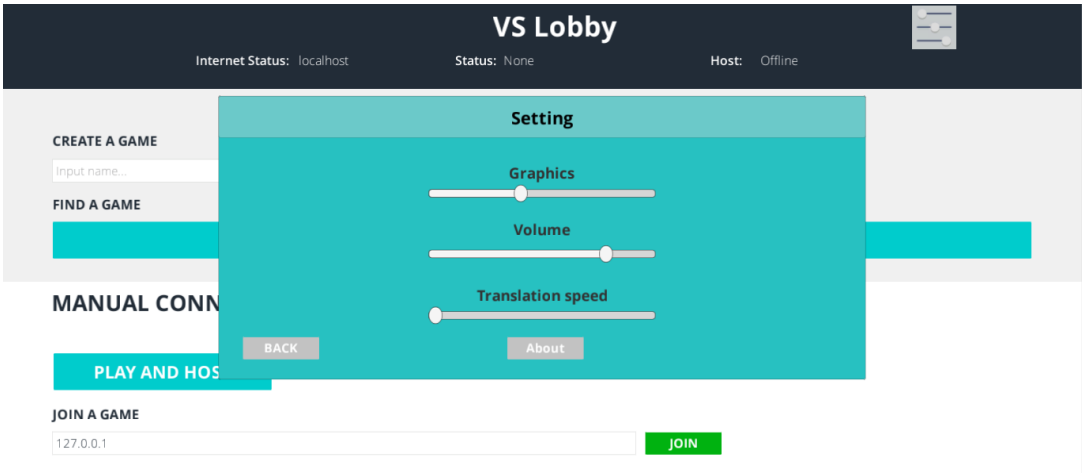


Figure 13 - Settings menu

After pressing *LIST SERVERS* (Figure 12) all the servers will be shown as in Figure 14. The player can see the server name and the available slots. The player can the press *JOIN* to enter one of the game sessions.

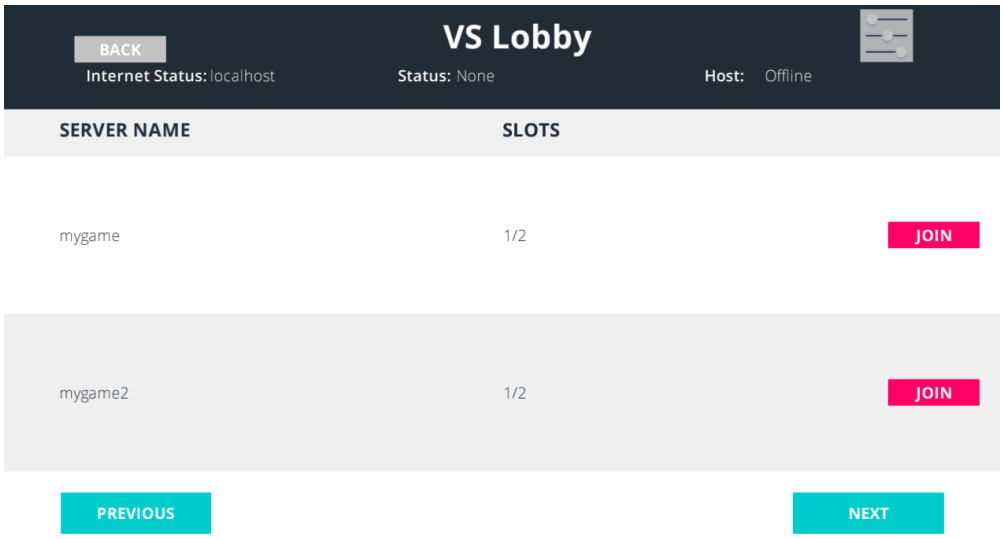


Figure 14 - Available servers listing

In game the interface will contain the score, the timer and the chat as can be seen in Figure 15.

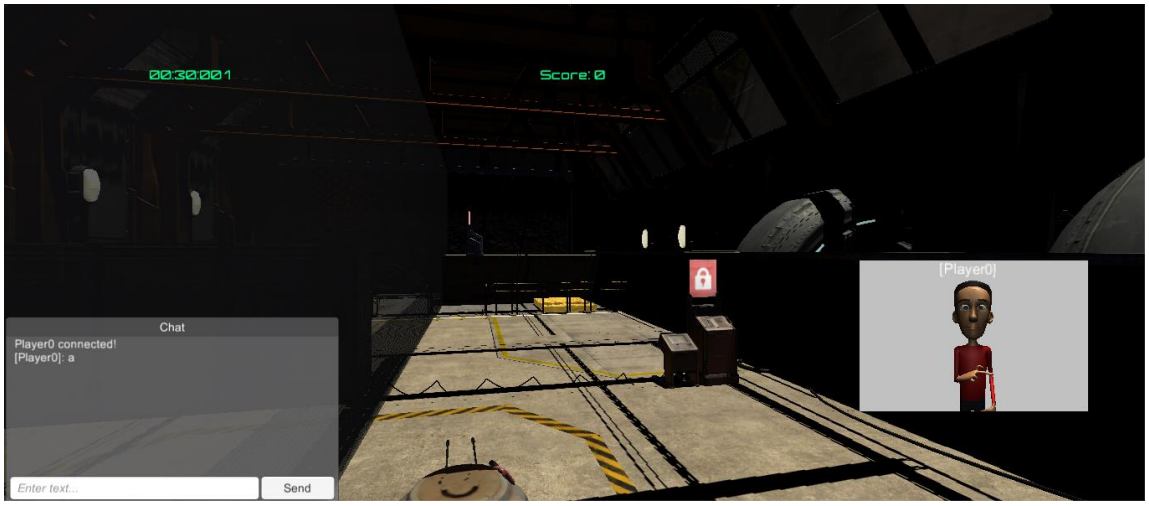


Figure 15 - In-game interface

The score is located at the top of the image, the timer on the top left corner of the screen and the avatar that translates can be placed anywhere so it has the lowest impact on the gameplay. The chat is in the lower left corner.

5.6 Scenes

The scenes were created using both unity assets and assets created by the developer, same applies to the textures.

There are four scenes in which the first is the Lobby and the other 3 are levels.

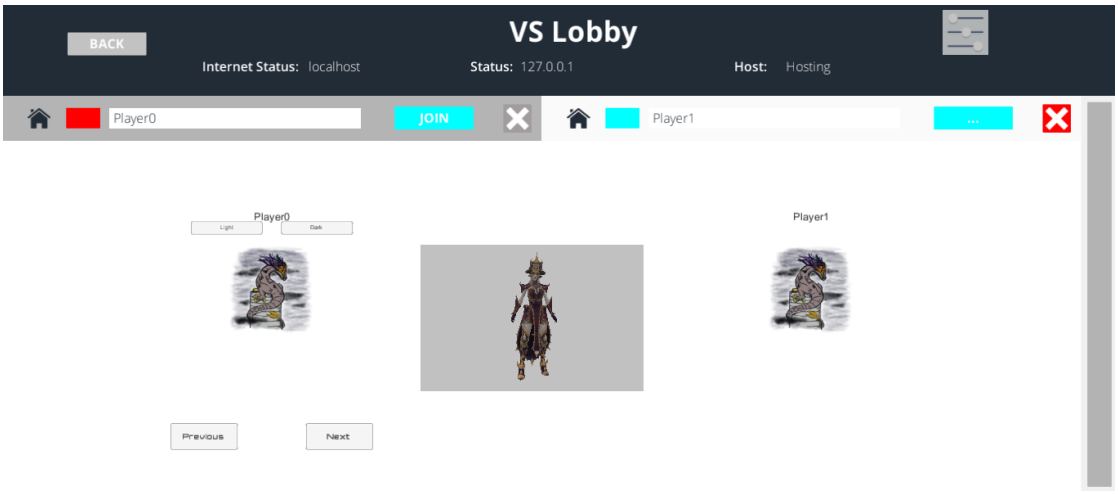


Figure 16 - Game session lobby

The lobby has two main screens one is the initial screen seen in Figure 12 and the other is the one in Figure 16 where the player choses the name, avatar and type for the match.

The first game scene can be seen in Figure 17 and it is a fully closed factory environment with 4 different challenges that are presented to the players. One of those challenges requires the dark type ability to proceed.

Figure 17 - Overview of the first game scene

On the second game scene, Figure 18, there are three challenges and this time is the light player that has to use the light ability.

Figure 18 - Overview of the second game scene.

The second scene follows a similar pattern to the first one but this time with an open sky showing that the player has moved through the game story and is closer to freedom.

The third game scene shares a similar environment with the first two but is larger and incorporates all the puzzle types of the game. This final scene can be seen in Figure 19 and it

shows that the end of the level is an open field with a wall containing the best scores from all the players meaning the final goal of the story has been reached.

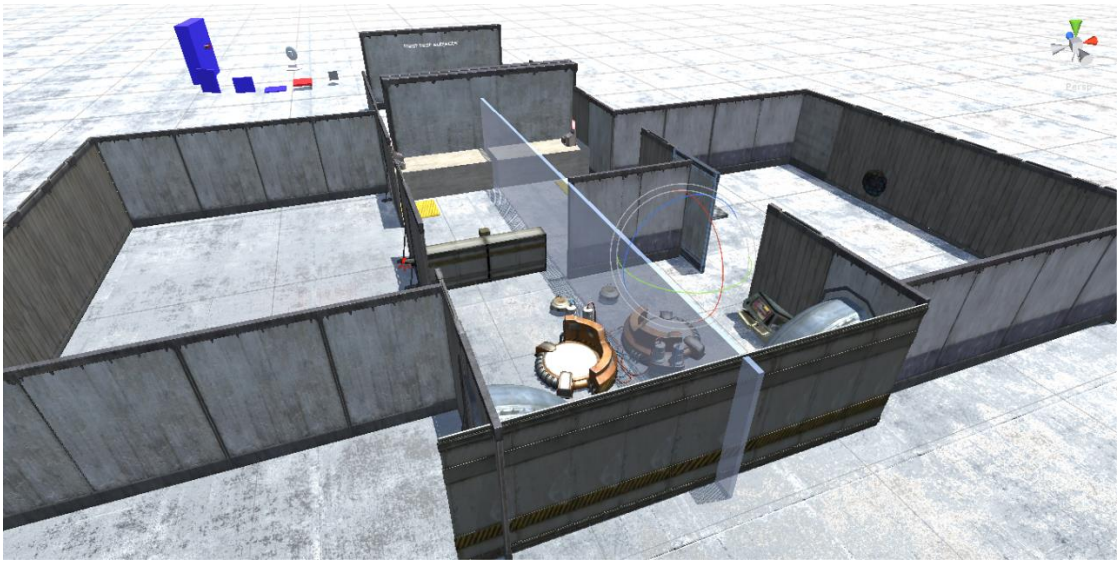


Figure 19 - Overview of the third game scene

All the game scenes share a specific aspect which is a central division of the level. Every level has two sides in which each of the players are and in between those sides there is a transparent glass that allows the players to see each other. The only way for the players to be together in the same area is by reaching the end of the game.

5.7 Translation and Avatar

The translation is done by the VirtualSign avatar that can be seen in Figure 20. The avatar was created, rigged and animated on Maya. It uses animations for the hand configurations, the movement is controlled through Inverse Kinematics (IK) handlers (Autodesk Maya, 2016). Even though the IK handlers are established within Maya, Unity allows the use of them as long as they are marked as so on the Animator tool. The animator tool, also known as Mecanim (Unity Technologies, 2014b) also provides an interface to set avatar as humanoid, meaning if any avatar has a set of bones in the rig that matches the humanoid standard it will be classified as humanoid and all humanoid animations will be usable with it. Finally the rotation of the hands are done by code and that is only possible thanks to the humanoid features or else the hand would deform unnaturally when rotated.

This avatar was integrated in the game and after doing tests trying to identify the best place on the screen for the translation, the decision was to let the player adjust and place the avatar on his will.

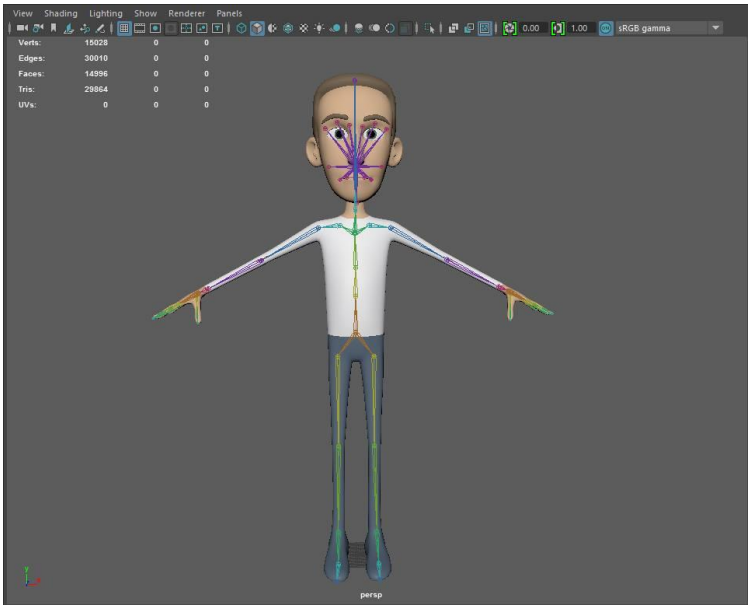


Figure 20 - – VirtualSign Avatar and its skeleton

On the other hand, the translation from the sign to text is received in the game via socket and it uses the VirtualSign recognition application seen in Figure 21.

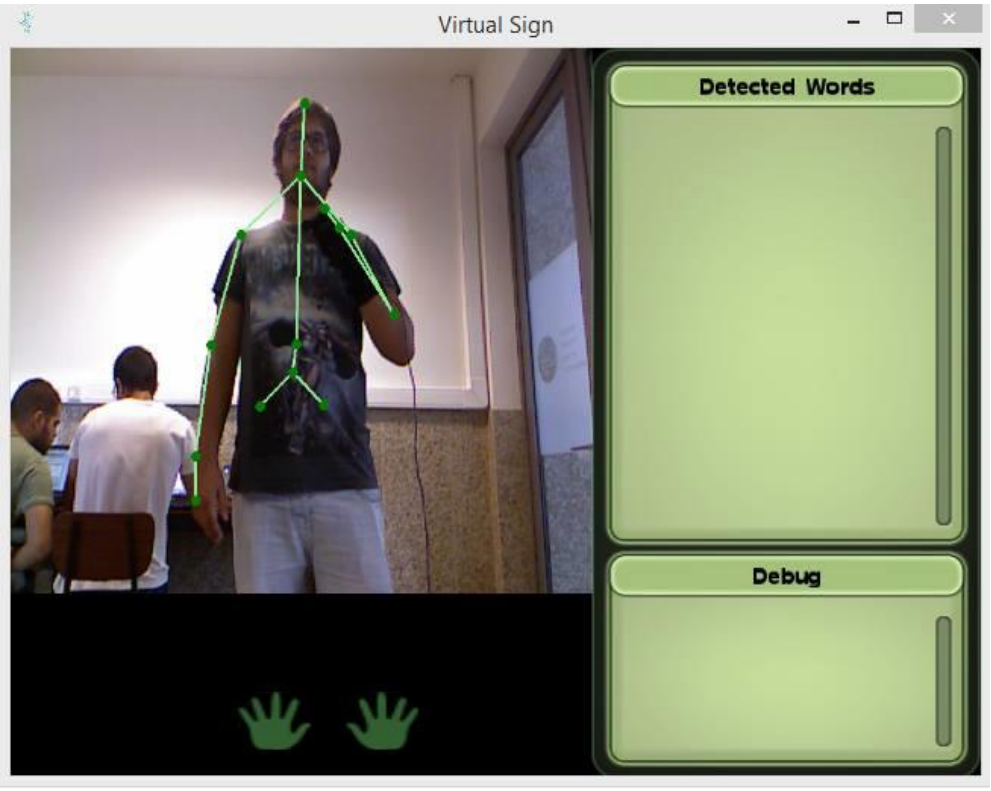


Figure 21 - VirtualSign Gesture to Text Translator.

The recognition is done through two main components, a gesture classifier and a motion classifier. The gesture classifier uses the 5DT gloves and the motion classifier uses the Kinect

Skeletal Tracking (Microsoft, 2016b). Together those two components are able to identify previously trained signs using Support Vector Machine (SVM) (Steinwart & Andreas, 2008) and K-Nearest Neighbors (KNN) (Peterson, 2009).

5.8 Gameplay Elements

Gameplay elements are the core of the gaming experience. Several puzzles that require forms, switches and buttons scattered around the scenes.

The user will see the other player on the other side allowing him to know with which objects the user should interact with. All the objects are scattered through the map and to access them the player has to solve multiple puzzles using all his surroundings. When all the puzzles are completed the player is able to reach the level exit which will allow them to go on to the next level.

The switches will always trigger an animation that will, for instance, open a new path or unlock a needed object.

The boxes are movable so the user will need to place them in the correct spot in order to solve the puzzle. The players will need cooperation to solve puzzles as they may need the other in order to reach a specific place. For example, there can be a button on platform higher than the player's reach with jumping, but the other player can interact with something that will move that platform thus boosting the players jump just enough to reach the button.

The scenes itself will have built-in puzzles, for instance a wall will have just enough space to fit a Cone that is accessible to the player, and when the player inserts the cone in it an event will be triggered.

Every time a puzzle is solved, based on the time that took to solve it, the score of the player who solved it will be increased as well as their global score as a team.

Another element that is always present is the chat, the chat will always be displayed during the game. The user will be able to communicate with the other using the translator.

There will be tips through the scene to improve the user experience, note that those tips do not give away how to solve the puzzle but what actions can be done in order to do so.

The flow of the game can be seen In the Figure 22, it shows generically how the game will work from starting the game until finishing a level.

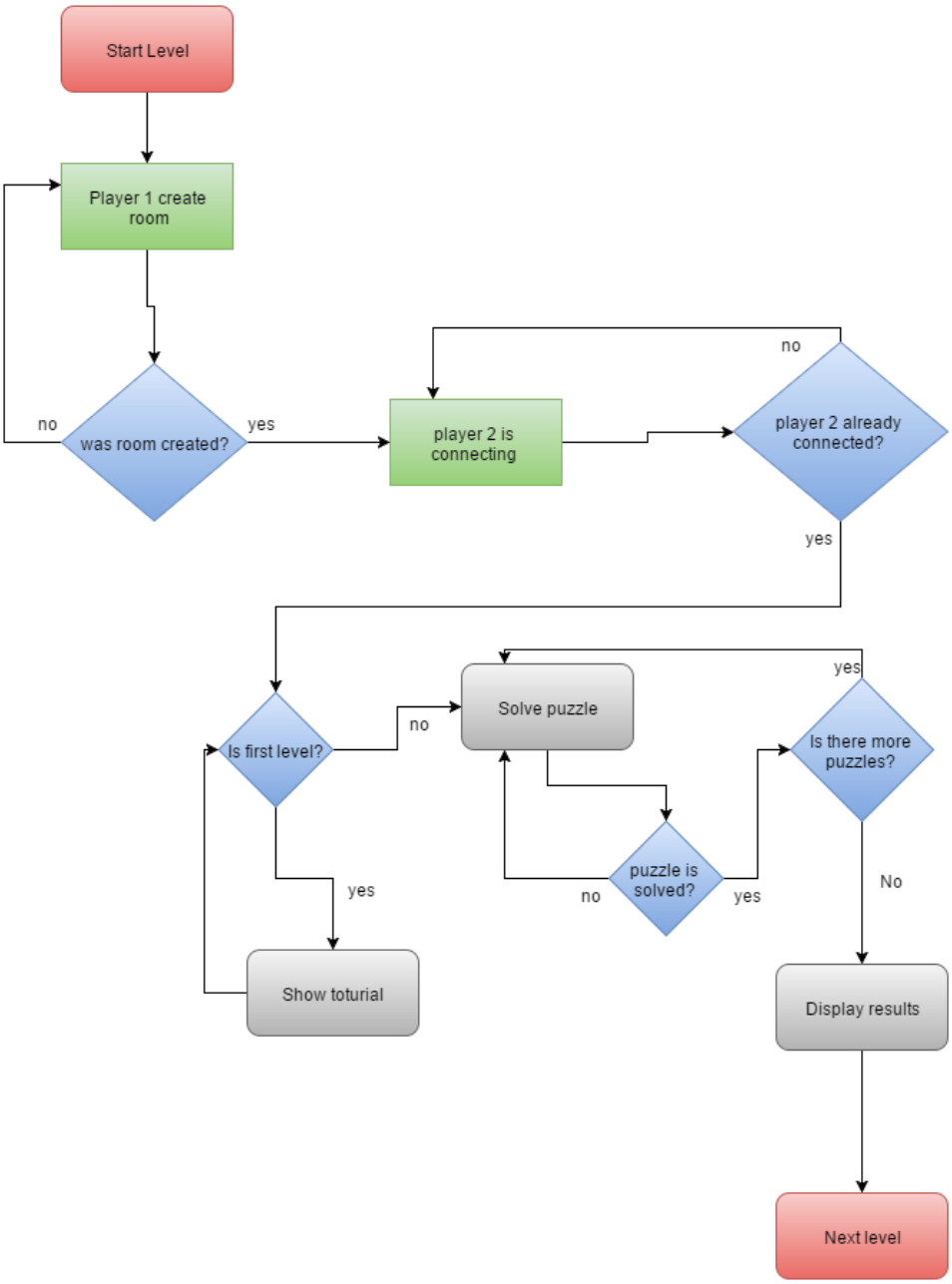


Figure 22 - Generic flowchart of the game

5.9 Sound and Music

Despite being a game aimed at the deaf community audio is still be used as anyone can play Puzzatron.

Open source sounds and music are used in Puzzatron. It focuses on subtle and calm sounds that relax the player while trying to solve a puzzle, therefore balancing the game flow (Collins, 2008). It is desired that the player anxiety doesn't increase which could result in the quitting of solving the puzzles when the player encounters difficulties. As Sylvain Moreno said: "*Evidence has*

suggested that music can improve behavioural performance in several domains, including intelligence." (Moreno, 2009). Thus using music may have a positive impact on the player performance solving the puzzles.

5.10 Architecture

The system architecture as a whole has two main components. The basic overview is represented in Figure 23. The main component is the game client that includes the game module and the Virtualsign translator. Then there is the Web Server component that hosts all the web services needed for the game. Those web services have access to the server database where the players' information is kept. The game clients communicate with each other using Unity network commands and with the web server through HTTP requests.

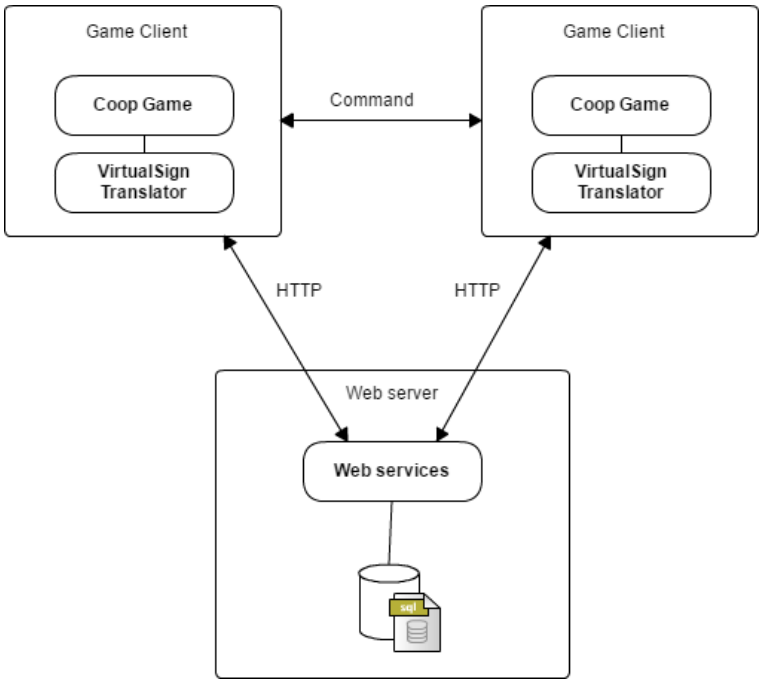


Figure 23 - Full system architecture.

5.10.1 Game client

The game client is divided into layers, given its high degree of complexity as shown on the Figure 24. At the top layer there is the interface. All the functionalities of the project can be accessed through this layer by the user. This layer is responsible for the transmission of the actions of the user for the following layers. On the lower level there are three layers. The sockets layer which is responsible for the connection between the game and the VirtualSign translator, the game engine layer that is responsible for the execution of the game itself, representing the

functions of unity, and finally, the network layer which is layer that controls the multiplayer as it is responsible for the transmission of information between the two game clients (Pantel & Lars, 2002).

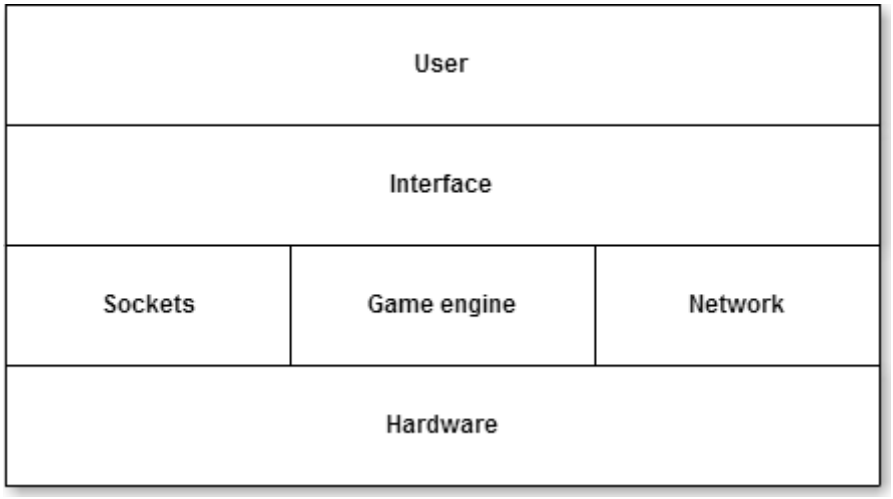


Figure 24 - Application architecture

5.10.2 Networking

The game clients are able to interact with each other using the native Unity Networking asset also described as UNET (Unity Technologies, n.d.).

The main reason for implementation UNET and not Photon was based on the customizability (Unity Technologies, 2014). Both assets provided similar performances with two players. Photon has more example assets for instance a chat making it easier to use initially. UNET, however, has some Engine integrated features such as the *NetworkTransform* that syncs the player character position. Overall they are two great assets but the fact UNET allows access to a high level scripting API and more customization of the code was the differentiating factor for this project.

The UNET provides easy access to the multiplayer game session's information and allows the developer to change the max players per room. In order to use UNET the developer must have a Unity account and create a new multiplayer project. Once the project has been created a UNET ID will created for that project. The overview of the networking UNET API can be seen in Figure 25.

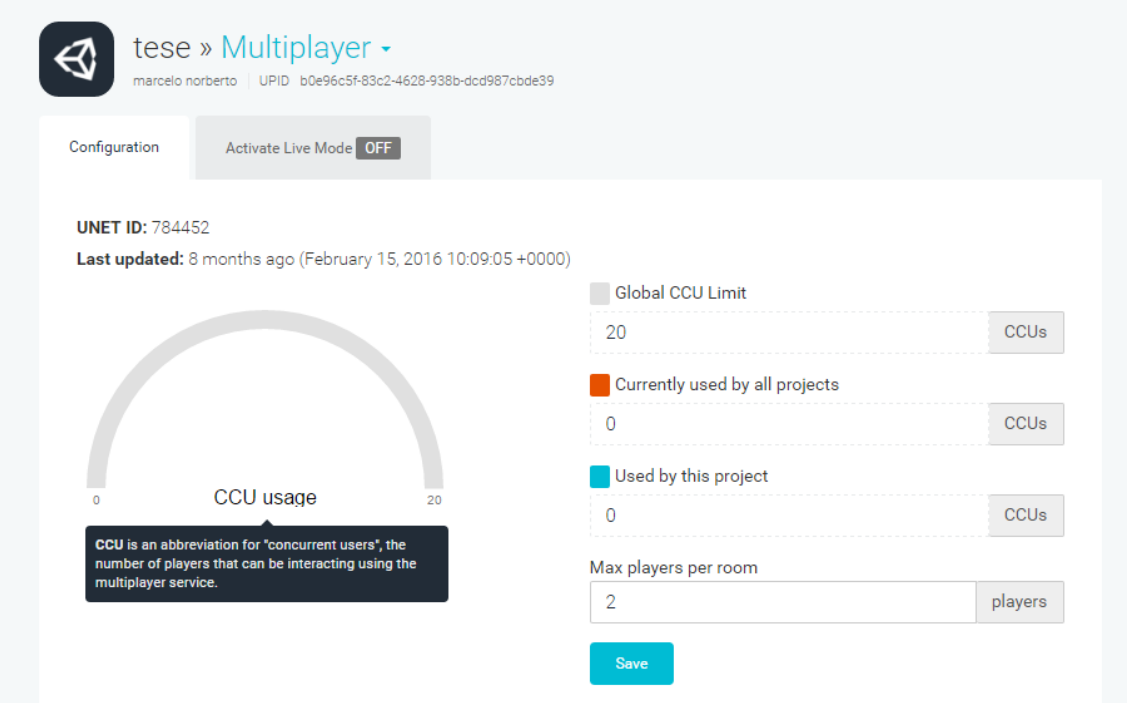


Figure 25 - UNET API overview

5.10.3 Web server

As for the Web server component it uses PHP and a MySQL database. Different HTTP requests are supported such as POST and GET and those requests are then processed and the necessary queries to the database are executed. The replies are in JSON so they are easily parsed. Two of the main features of the Web server for this application is the player account and log in information and the words database that allows the game translator to be continuously updated.

The database is stored on the VirtualSign server and uses phpMyAdmin (phpMyAdmin, 2016) which is a free tool written in PHP used to administrate MySQL databases over the web.

Figure 24 presents the database Data Model.

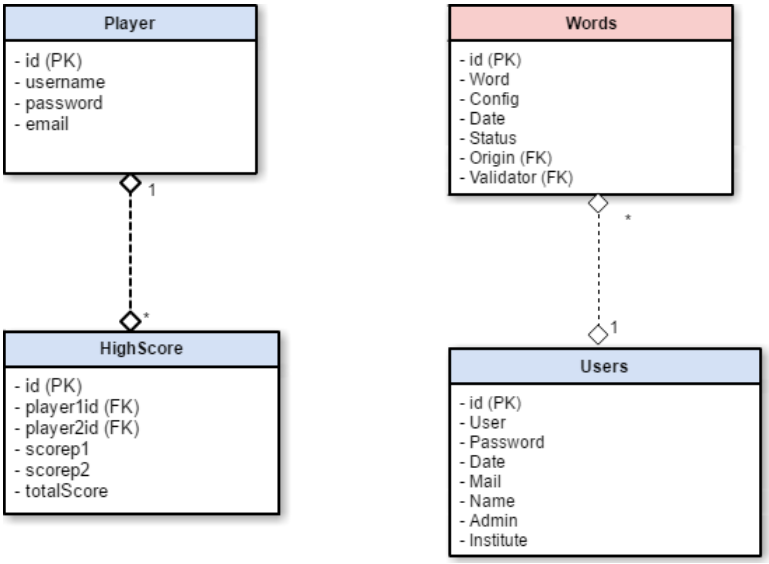


Figure 26 - Server Database Data Model

On the left side of Figure 26 there are the player and high score tables. The player table stores the personal information of the player and its login information. The high score table is created in every match of the game and it stores both players' IDs as well as their individual and collective scores.

On the right side of the Figure 24 there are the translator related tables. Those are required to update the set of words available for translation within the game. In order to have access to an online repository of words a new component was added to the VirtualSign translator called VirtualSign Studio. It allows users to login into a configuration interface and create words in sign language. Those words are then stored in the Words table and stay pending for validation.

There are different kinds of users, the user type is defined by the admin attribute on the Users table. The two main types are the default user and the validator. The default user is the one that creates gestures and they are stored online for validation. The validator has all the default user privileges plus an interface to validate currently store gestures on the database. In this interface the validator can accept, refuse or review a word. Only the validated words are sent through the web service to be used in the game.

6 Implementation and Evaluation

The Cooperation game Puzzatron was developed with Unity3D using C#. The networking was created with Unet. The 3d models are mostly from unity assets and the VirtualSign avatar was created in Maya. However the player avatars were retrieved from Mixamo (Adobe Systems Incorporated, 2016). The sound is mostly mp3 or wav commercial free audio files. The web server uses PHP to handle the communications between the client and the server and access to the database. The game is available on windows only due to the restrictions mentioned on the non-functional requirements.

6.1 Alpha Version

In the alpha version the main goal was the implementation of all the core features of the game. Initially the creation of the multiplayer component as the rest of the scripting would to be written based on that component. Thus several tests were done to the check which platform provided the best multiplayer for unity and as mentioned on the networking architecture the best option for this scenario was Unet. So the following features were implemented: multiplayer lobby, movement, rotation and animation synchronization, picking up the boxes, interact with switches, use the special ability, timer, score, chat and the translation of the chat.

6.1.1 Multiplayer Lobby

Unet provides a lobby for the developers to adjust to their needs. By default it accepts an online and an offline scene. The offline scene is either the lobby or an entry interface where the players chose which game to join or create. The online scene is the level itself. It also provides a simple system to spawn a player object. The asset as is could work for a vast set of scenarios but not for a cooperation game with customizable characters and attributes as well as multiple levels. Thus the reason why Unet was chosen instead of photon since UNET allows for a high customization of the original scripting.

The core lobby scripts were changed in order to allow for the intended features. Unet has a lobby player and a player object. When a room (game session) is created a lobby player is instantiated and it has a customizable name and color on the lobby but those references were not sent into the game as when the level starts it's the player object that is instantiated. So an objective was to have information about the player name, player type and player character and at the same time sync this information to the other client in the game room. Even though the Unet documentation is lacking details on how to change the player by code, after some research and numerous different approaches a solution was found as is presented on Code 1.

```
  public override GameObject
OnLobbyServerCreateGamePlayer(NetworkConnection conn, short
playerControllerId)
        {
            LobbyPlayer temp_ = currentPlayers[conn.connectionId];
            int index = 0;
            int posID=0;
            string type;
            string name;

            index = temp_.playerPrefabId;
            type = temp_.playerType;
            name = temp_.playerName;
            gamePlayerPrefab = spawnPrefabs[index];
                if (type == "Light" && lightTypeCount==0)
                {
                    lightTypeCount++;
                    posID = 0;

                }
                else if (type == "Dark" && darkTypeCount == 0)
                {

                    darkTypeCount++;
                    posID = 1;
                }
                else
                {
                if(darkTypeCount==0)
                {
                    posID = 1;
                    type = "Dark";
                }
                else{
                posID=0;
                    type="Light";
                }
                }
            GameObject _temp =
(GameObject)GameObject.Instantiate(spawnPrefabs[index],
            startPositions[posID].position,
            Quaternion.identity);
            _temp.GetComponent<PlayerID>().playerType = type;
            _temp.GetComponent<PlayerID>().playerName = name;
            return _temp;
```

```
        }
```

Code 1 - Instantiation of the player object with all the player information form the lobby

The function on Code 1 is in the *LobbyManage*r script and it overrides the game player object. Initially in the function a temporary lobby player is created (*temp_*) based on the *connectionId*. Since every player has a different connection ID it's possible to identify which player the script is trying to access with it. The reason a lobby player is needed is because the player name, type and character are stored in it before the game starts.

The variables *type*, *name* and *index* are the ones where this information is kept and then they are used to pass those values into the game player (*_temp*) as they are stored in the *PlayerID* script except the character model that has to be defined as a *spawnable prefab* and then is accessed in the *spawnable prefabs* list using its *index*. One of the reasons it was challenging to achieve this method, besides the lack of documentation, is synchronizing this information across clients.

On a few of the failed initial attempts the player would spawn either just on the host or the same character on both clients. Unet has a different flow for the host and for a client. The host, even though it is a client, it acts as server, thus anything that needs to be synchronized has to go through the host fist. So when a change is done on the host, the clients will receive it through *SyncVars*. *SyncVar* is an attribute that syncs variables from the server to clients (Unity Technologies, 2016d). So in order to a client to send information to the host it must be through a Command attribute of unity (Unity Techonologies, 2016e).

Taken into consideration the flow of information in UNET, several methods and variables were created to sync the information of the lobby player.

```
[SyncVar(hook = "OnMyName")]
public string playerName = "";
[SyncVar(hook = "OnMyPrefab")]
public int playerPrefabId = 0;
[SyncVar(hook = "OnMyPrefabType")]
public string playerType = "Dark";
```

Code 2 - Initialization of the SyncVars

In order to define a variable as a *SyncVar* it is necessary to write *[SyncVar]* before the variable is declared. On Code 2 however, there is also a hook function, what that means is that every time that variable is changed not only the variable will be synced from the server to all the clients but also a function will be called on both server and clients. The function name must be passed as a string to the hook.

With the synchronization of those variables and hook functions the server can then synchronize all the clients but that is not enough as the client still wouldn't be able to synchronize with the server, thus the need of the commands that can be seen on Code 3.

```
 [Command]
public void CmdNameChanged(string name)
{
    playerName = name;
}

[Command]
public void CmdPrefabChange(int id)
{
    playerPrefabId = id;
}

[Command]
public void CmdPrefabChangeType(string type)
{
    playerType = type;
}
```

Code 3 - Commands in the LobbyPlayer script to sync the name, character and player type

Similar to the *SyncVar* attribute a *Command* must be declared before the method writing *[Command].* Unity then knows that that function is a network command that can be sent from a client to the server but it must still be called when needed as in Code 4.

```
public void OnNameChanged(string str)
        {
            CmdNameChanged(str);
        }
public void clickTypeLight()
        {
            CmdPrefabChangeType("Light");
        }
public void clickTypeDark()
        {
            CmdPrefabChangeType("Dark");
        }
```

Code 4 - Methods calling the "Commands"

The *Commands* must begin with the prefix "Cmd". By calling those commands when a variable that should be synced is changed its possible to assure that both server and clients have synchronized the required information with each other.

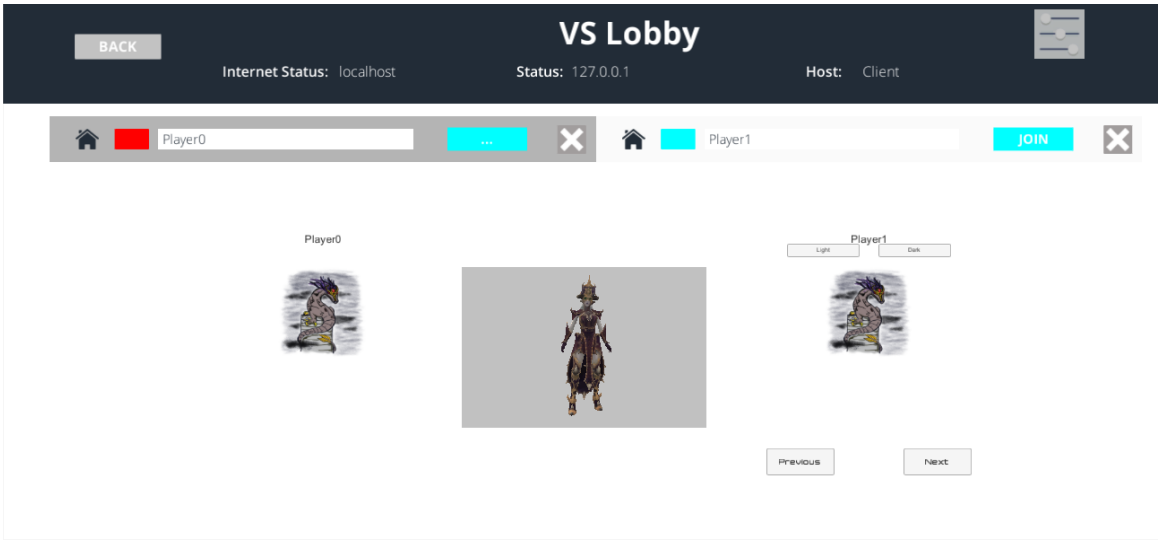


Figure 27 - Lobby default state with two players

Figure 27 both players have the default attributes. The players may then change the name, type and character. The players can chose the same name and the same character however, the types must be different so if two players pick the same type, the priority will be of the host who created the game session. Figure 28 shows the interface after the players have chosen their details.

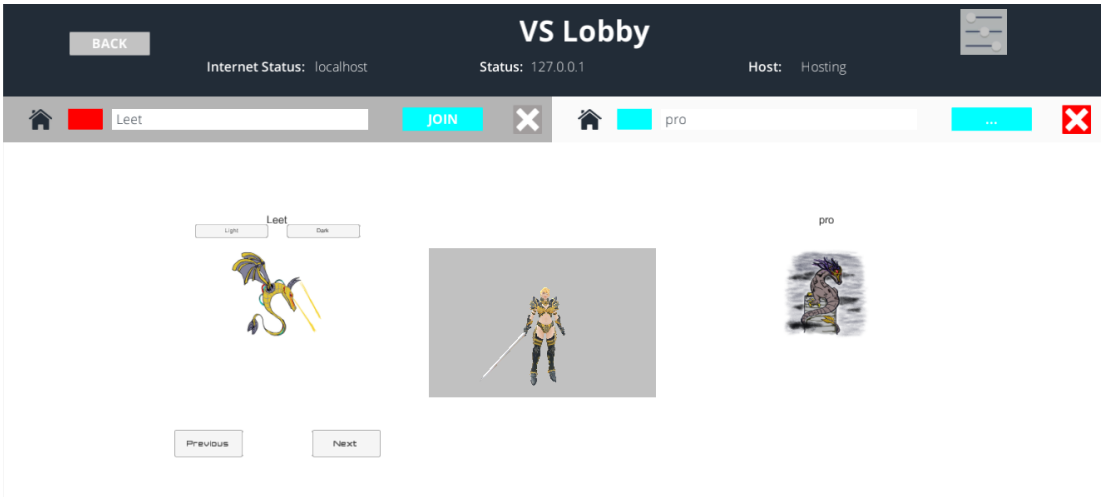


Figure 28 - Lobby room after the players have picked their preferences

The player on the left side of the screen is always the host of the game. Figure 25 shows the game lobby view from the client perspective and the Figure 26 from the host perspective. One of the noticeable differences other than the side of the player is that the host has the ability to remove the player out of the game session, thus why the “x” icon on top right corner is highlighted in red in Figure 26 and not in Figure 25.

6.1.2 Movement Rotation and Animation Synchronization

The player character interaction is done though a default script named *FirstPersonController* which provides basic movement controls. However, this applies only for the client where the application is being ran. UNET provides some built in components for such synchronization. All the UNET components need a *NetworkIdentity* (Unity Technolgies, 2016f) attached to object which works an ID for every network object. The *NetworkTransform* can sync the transform (position and rotation) of an object or its *rigidbody* (Unity Technologies, 2016g) or it *FirstPersonController* if the item has one. Figure 29 illustrates the available sync modes.

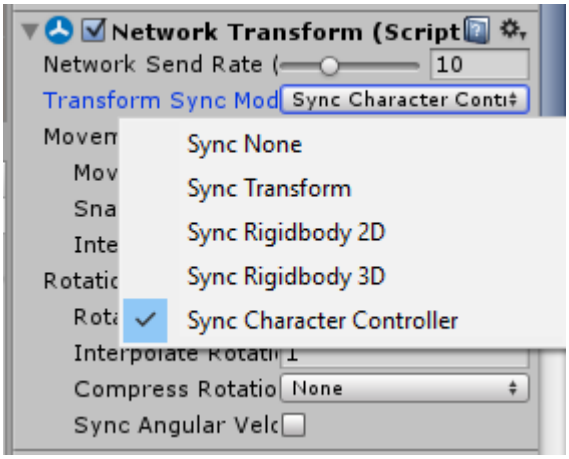


Figure 29 - Network Transform component sync modes

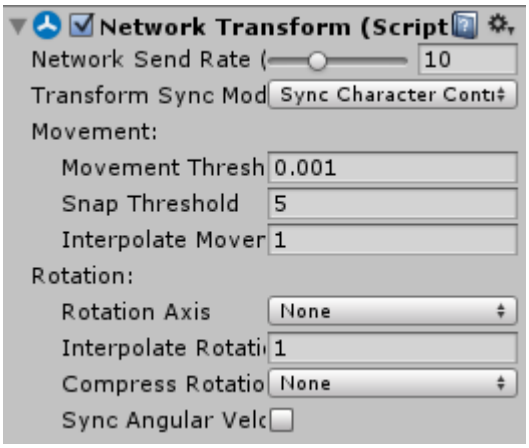


Figure 30 - Network Transform component options overview

Figure 30 all the options of the *NetworkTransform* can be seen. The options on the image are the actual final configuration used on the game. Initially the transform sync was used but despite what adjustments were done it always had some degree of delay in the movement. In parallel a script to sync the movement was developed and to some point it showed better

results than syncing the transform. After several tries the network transform proved to have an outstanding performance when synced with the character control.

```
void LerpPosition()
    {
        if (!isLocalPlayer)
        {
            if (useHistoricalLerp)
            {
                HistoricalLerp();
            }
            else
            {
                myTransform.position = Vector3.Lerp(myTransform.position,
SyncPos, Time.deltaTime * lerpRate);
                SyncPos = myTransform.position;
            }
        }
    }

    void HistoricalLerp()
    {
        if (SyncPosList.Count > 0)
        {
            myTransform.position = Vector3.Lerp(myTransform.position,
SyncPosList[0], Time.deltaTime * lerpRate);
            if (Vector3.Distance(myTransform.position, SyncPosList[0]) <
gettingClose)
            {
                SyncPosList.RemoveAt(0);
            }

            if (SyncPosList.Count > 10)
            { lerpRate = fasterLerpRate; }
            else { lerpRate = normalLerpRate; }
        }
    }
```

Code 5 - PlayerSyncPosition script core functions Lerp position and Historical Lerp

Code 5 represents the position sync functions developed to compare the performance with the UNET native feature. When comparing the *NetworkTransform* with the linear interpolation (lerp) the results were similar with some stuttering but when the historical lerp was applied then the player sync script performance was better with no delay what so ever but it had some noticeable delay before the movement started on the receiving end. So at this point the performance was acceptable but not the desired.

After a few experiments the decision was to change the network transform and sync it to the *CharacterController* which basically uses the input from the *FirstPersonController* and sends it over the network to sync between clients. This method removed any delay as it was as if the other character was being moved by the player and had no delay unlike the historical lerp. As for the rotation it could also be handled by the *NetworkTransform* however in this case the performance of the script developed for that purpose proved slightly smoother. With the use

of the historical lerp seen in Code 6, the rotation was smooth and the delay barely noticeable especially if the player is moving as well. A direct lerp was also tested but the results were delayed as expected.

```
    void HistoricalLerp()
    {
        if (syncPlayerRotList.Count > 0)
        {
            LerpPlayerRotation(syncPlayerRotList[0]);
            if (Mathf.Abs(playerTransform.localEulerAngles.y -
syncPlayerRotList[0]) < gettingClose)
            {
                syncPlayerRotList.RemoveAt(0);
            }
        }
        if (syncCamRotList.Count > 0)
        {
            LerpCamRot(syncCamRotList[0]);
            if (Mathf.Abs(CamTransform.localEulerAngles.x -
syncCamRotList[0]) < gettingClose)
            {
                syncCamRotList.RemoveAt(0);
            }
        }
    }

     void LerpPlayerRotation(float rotAngle)
     {
        Vector3 playerNewRot= new Vector3(0, rotAngle,0);

        playerTransform.rotation=Quaternion.Lerp(playerTransform.rotation,Q
        uaternion.Euler(playerNewRot),lerpRate * Time.deltaTime);
     }

     void LerpCamRot(float rotAngle)
     {
        Vector3 camNewRot = new Vector3(rotAngle,0,0);
        CamTransform.localRotation=Quaternion.Lerp(CamTransform.localRotati
 on,Quaternion.Euler(camNewRot), lerpRate*Time.deltaTime);
     }
```

Code 6 - PlayerSyncRotation script core functions Lerp Player Rotation, Lerp Camera rotation and historical lerp.

Animations are one of the best ways to provide the player with some feedback from his actions. To provide such feedback to the players several animations were used in the game especially on the character. One of the tools unity provides is called the animator. It consists on a simple interface where the user can drag animations to and create layers and parameters that trigger those animations. This tool is shown in Figure 31. The animations are displayed as boxes and the user can create connections between them and define the parameter that triggers that connection.

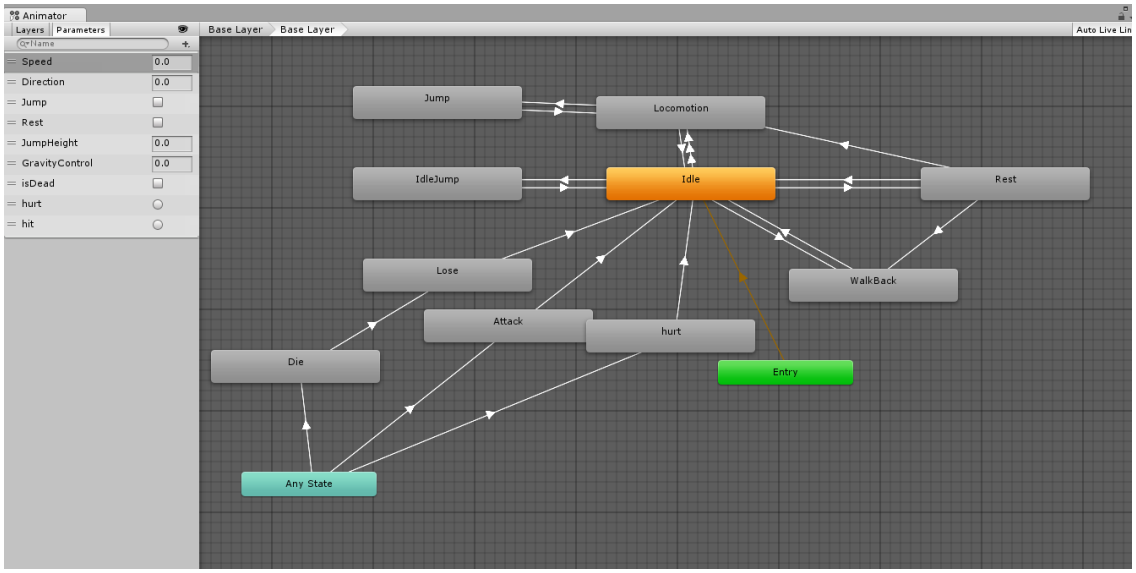


Figure 31 - Unity Animator interface of the player

The player character has an animator associated with it and the parameters are called from the *FirstPersonController* script which was adjusted for that purpose. This logic, however, applies only for the local client. Which leads to the one other reasons why UNET was chosen which is native support to sync animation through network using a *NetworkAnimator* component. Figure 30 shows the component on the Unity inspector.

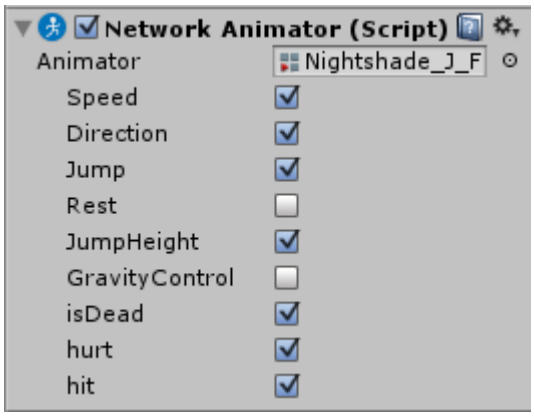


Figure 32 - Unity Network Animator Component with the character animation parameters

The *NetworkAnimator* uses the reference to the character animator and then displays all parameters of the animator. In Figure 32 the checked boxes are the variables that will be synced through the network and triggered on all the clients animators that correspond to the one assigned to the *NetworkAnimator*. The performance of this component is very satisfactory as all the animations are played with close to no delay across clients.

6.1.3 Picking up the boxes

One of the core mechanics implemented in the game has to do with the interaction of the player with boxes and the ability to pick them up and place them somewhere else. A position was defined in order to move the box according to the player. That position is a child of the player object and is where the box will be place while the player is carrying it. However there was a challenge where the player was carrying the box but only the local client could see it even though the box had a *NetworkTransform* syncing the transform. The reason why that happened was that for a game object to be synchronize using UNET components it needs to be defined as a player prefab on the network manager. The problem then was that the box was not a player thus would not be wise to mark it as so. The solution was to pass the client authority temporarily to the cube as demonstrated on Code 7.

```
  void grab()
    {
        if (Physics.Raycast(camTransform.transform.position,
camTransform.forward, out hit, range))
        {
            Debug.Log(hit.transform.name);
               Debug.Log(hit.transform.tag);
            if (hit.transform.tag == "Grab")
            {

               CmdAssignObjectAuthority(hit.transform.gameObject.GetCompone
               nt<NetworkIdentity>().netId);
                carrying = hit.collider.gameObject;
                carrying.GetComponent<Rigidbody>().useGravity = false;
                hit.transform.position = Hand.position;
                hit.transform.parent = transform;
         }
            }
        }


    [Command]
    void CmdAssignObjectAuthority(NetworkInstanceId netInstanceId)
    {

        NetworkServer.objects[netInstanceId].AssignClientAuthority(connecti
        onToClient);
        carriedObject = netInstanceId.ToString();
        carry = true;
    }
```

Code 7 - Grab a cube function.

On Code 7 the first function *grab* casts a ray from the player center of the screen and checks if there is a collision with an intractable object. If there is then that object is assigned with the client authority which makes UNET recognize it as a network object. Then its location is set to the player predefined carry position until the player drops it and the client authority is removed. Figure 33 shows a player carrying a cube.

Figure 33 - Cube being carried by a player.

One of the features implemented that also uses this mechanic is the pressure plates. Initially pressure plates were going to be used by the player only but one of the advantages of having several different mechanics in a game is that sometimes it is possible to combine them. Pressure plate's mechanic implemented in the game is pretty simple. It consists on a plane on the floor with a collider that detects when any object enters it and calls two animations. One on the plane itself and another on a platform that will rise or a door that will open. And if the object leaves the plane collider then the animations play backwards lowering the platform or closing the door.

As for the network since objects that go on top of the plane are already being synced there could be no network script for it but to prevent a situation of lag for instance a network identity was added to the plane anyway. As well as all the objects animated by this interaction having a Network Animator which ensures that if the event happens on one side its synced to all the clients. The result of this behavior can be seen in Figure 34 where the yellow platforms, which the original position is the one in Figure 33, were raised because the box was placed on the pressure plate.

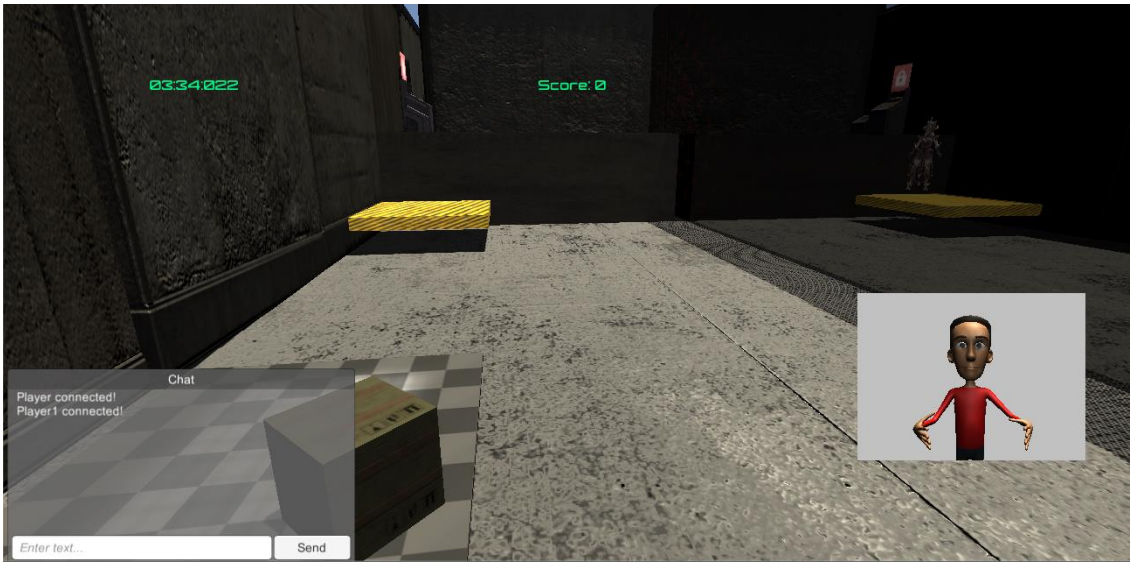


Figure 34 - Pressure plate pressed by a box placed on top of it

6.1.4 Interact with switches

Switches have a simple interaction and yet so much can be created using them. The switches in the game are very visible as part of the game design itself. The goal of having switches in the scene is for the players to feel urged to press them, which is what is intended. However, those switches are not always within reach and in order to get access them the player needs to overcome a few challenges. Once reaching the switch the player can then interact with it. Each switch is connected to a door and the player can open or close that door by clicking the switch. Figure 35 shows one of the switches at the end of level two on both states, opened and closed. On the left side of each half of the figure it is possible to see a giant wall when it is closed and it disappears when it is opened.

Figure 35 - On and off switch

The networking logic for the switches follows the same flow of the lobby where the player needs to send a *Command* to the server that then syncs to the players. However there is the same problem that the cubes had as the switches are not player objects. The solutions was to send a *Command* from the player when there is an interaction with a switch and on that command the

network ID of the switch is sent to the server. The server can then update the switch on all clients. The command implementation is shown on Code 8.

```
  [Command]
    void CmdProvideBoolToServerSwitch(bool trig, string id)
    {
        if (GameObject.FindGameObjectsWithTag("Switch1") != null)
        foreach (GameObject obj in
GameObject.FindGameObjectsWithTag("Switch1"))
        {
            if (obj.GetComponent<NetworkIdentity>().netId.ToString() == id)
                obj.GetComponent<SwitchController>().Triggered = trig;
        }
        if (GameObject.FindGameObjectsWithTag("Switch2")!=null)
        foreach (GameObject obj in
GameObject.FindGameObjectsWithTag("Switch2"))
        {
            if (obj.GetComponent<NetworkIdentity>().netId.ToString() == id)
                obj.GetComponent<SwitchController>().Triggered = trig;
        }
    }
```

Code 8 - Switch function command

6.1.5 Special ability

The special ability was the last mechanic thought for the game. It is not a new concept but is still very interesting to have different abilities perform different actions. In the game there are two different kinds of ability. One is the dark which gives the player the power to create a dark cloud that can be used to cover surveillance cameras such as in Figure 36.

Figure 36 - Dark ability used on a camera

The other ability is the light which allows the player to interact with light emitting objects that will concentrate the light into a beam that will trigger events such as opening a door.

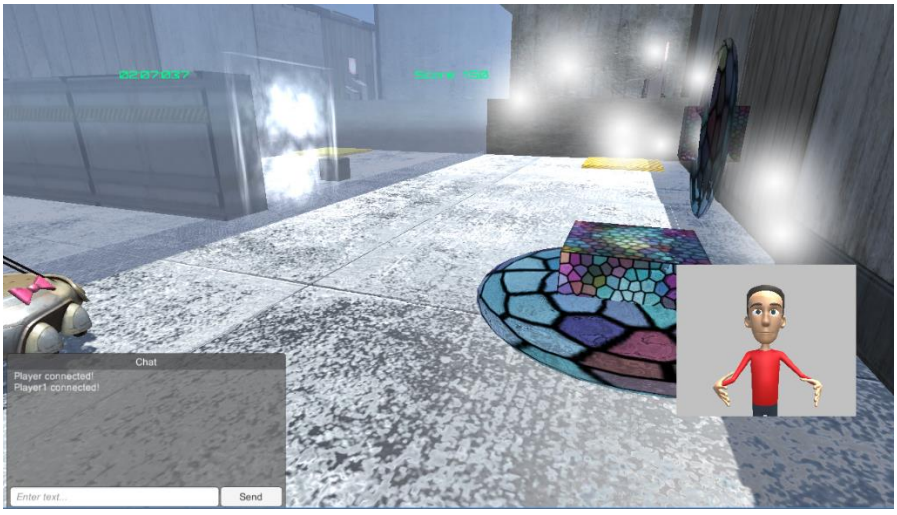


Figure 37 - Light emitting intractable objects

Figure 37 there are light particles around the objects where the light ability can be used.

The logic behind the special abilities is the same as the switches except that instead of just using animations it triggers also triggers particles. Each player has a particle system attached and so

do the objects where the special abilities are used on. With Unity particle system (Unity Technologies, 2016h) it is possible to detect collision on the particles emitted which opens an array of possibilities to make use of this feature. In this case it was used on the light beam which will only trigger the animation to open the door when it collides against the light wall on the left side of Figure 36 for instance.

6.1.6 Timer and score

In a puzzle based game where the puzzles have only one solution the best way to evaluate the player performance is by measuring the time required to solve the puzzle. A timer was implemented into the game and that timer is used to calculate the score. The faster the player solves the puzzle the higher the score the player will obtain.

```
public void ScoreChange(int s)
 {
     if (timer > 60)
         s = s / 2;
     timer = 0;
    CmdScoreToServer(score+s);
    Debug.Log("score is " + score);
}
```

Code 9 - Function that changes the score

Code 9 shows a simple function that is called whenever a player solves a challenge. The time it took is checked and if it is greater than 60 seconds the score obtained is halved. The timer is then restarted and a command to the server is called so the score is synced to all the clients.

6.1.7 Chat and Text Translation

The implementation of the chat was a complex process as it required to identify the player name whose message was originating from. Send this message to the server and then having the server send it to the client. So an event Handler was registered on the server and a method *Send* (Unity Technologies, 2016i) was called from the client containing the message. The method sends a network message to the server with a specific message ID that was previously defined for the chat messages. The chat interface is illustrated in Figure 38.

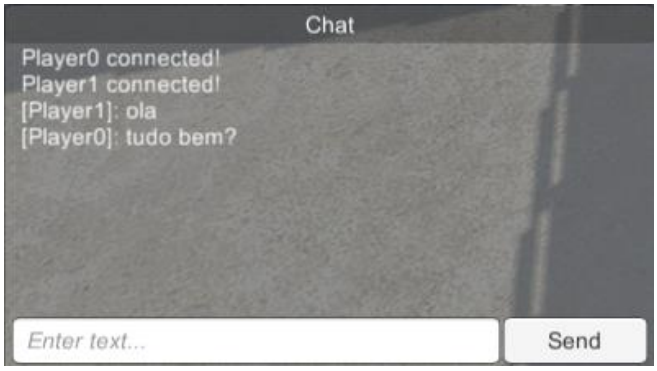


Figure 38 - In-game chat interface

The translation was implemented right after the chat. A new camera was added to the scene so that the player can see the translator avatar. The camera size was adjusted to take only a small portion of the screen. However, to prevent it from interfering with gameplay the user can move the translation area on the screen as shown in Figure 39.

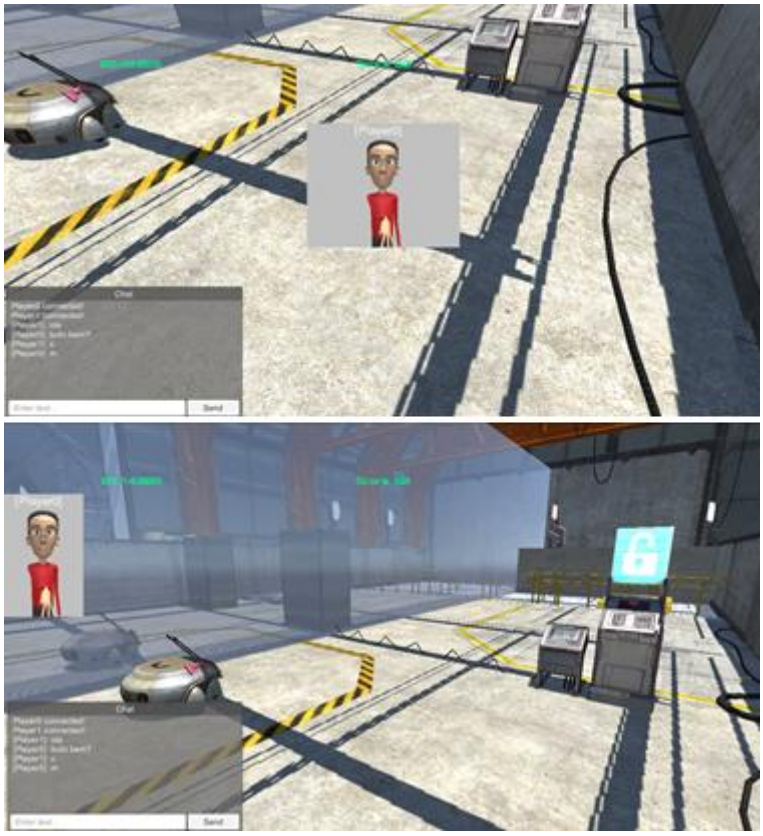


Figure 39 - Translator avatar in different areas of the screen

In order to move the avatar the player must click inside the rectangular area where the avatar is inserted and then drag it. On the bottom image of Figure 39 the avatar was placed to the furthest left position and the rectangular area adjusts accordingly. This freedom of placement was achieved using the mouse input positions on the screen and centering the camera viewport

to those positions. The camera viewport is the area of the screen where the camera view is displayed. The avatar was placed in front of the camera and the needed game objects for the translation were added to the scene such as the IK handles and the rotation handlers. To translate text into gesture a script was created to handle the translation and receive the text from the chat.

The translation of text to gesture uses a parameterization system that is parsed and converted into the animation and movements of the avatar. Every expression in sign language can be turned into a parameter using the VirtualSign Studio application. However initially this tool only saved the information locally on the machine. Together with Puzzatron an online repository for the VirtualSign Studio was developed. Through this repository it was possible to gain access to all configured and validated content. It uses a simple database with a table containing information about the users that added the gestures and a table for the gestures. The game uses HTTP POST request to communicate with the web server that is implemented in PHP that queries the database and sends back the information related to the request parsed as JSON.

The game automatically updates on start up every time it is ran. Code 10 shows the retrieving of the data for the updating process.

```
  IEnumerator updateDB()
    {
        PlayerPrefs.SetString("DBPT", "");
        PlayerPrefs.SetString("Lang", "PT");
        Lang = "PT";
        AsyncExample.StartAsync("http://193.136.60.223/virtualsign/pt/API.p
 hp?method=getAllWordsByCountry&jsoncallback=?");
        yield return null;
    }
public string DoWork(string url)
        {
            NameValueCollection param = new NameValueCollection();
            param.Add("Lang", Lang);
            WebClient c = new WebClient();
            c.Proxy = null;
            try
            {
                byte[] responseArray = c.UploadValues(url, "POST", param);
                c.Dispose();
                return Encoding.ASCII.GetString(responseArray);
            }
            catch
            {
                Debug.Log("Failed to connect to database or internet please
check your connection");
                return "";
            }
        }
     private static void onServerResponse(IAsyncResult result)
        {
            AsyncResult res = (AsyncResult)result;
            myCallBackDelegate deleg =
(myCallBackDelegate)res.AsyncDelegate;
```

```
        serverResponse = deleg.EndInvoke(result);
        if(serverResponse!="")readyToLoad = true;
    }
```

Code 10 - Gestures updater function

The update is performed asynchronously for a better performance and so it doesn't stop the main thread of the game causing it to temporarily freeze. After the *onServerResponse* function is called the response is then parsed and added to the game database of gestures.

The translation is done regressively, meaning that when a player sends a message, it checks the full content (string) of that message and looks for a match on the game database. If no match is found then the process is repeated without the last word of the string, until eventually there is a match or in case there is none then when the first word is reached that word is translated letter by letter. Figure 40 shows the avatar after translating the letter "m".

Figure 40 - Avatar after a translation

It can also be seen in Figure 40 that on top the avatar there is the name of the player from who the message originated written in between square brackets.

6.2 Alpha Testing

The evaluation of Puzzatron had two phases. The alpha test was the first phase where the features planned for the alpha version of the game were tested.

The tests were evaluated using QEF and through an online survey with Google Forms (Annex B).

A total of 10 testers (8 male and 2 female) aged between 23 and 65 tested the game and answered the surveys. The link with the form and instructions was given to the testers. The forms were answered anonymously.

The alpha test survey focused on the game basic features and on the QEF. Most of the features implemented on the alpha phase were related to the usability and game play of Puzzatron thus the questions focus for the most part on those two factors.

6.2.1 Results

The survey of the alpha test phase had 15 questions of which the last one was an optional written answer about possible improvements to the game. All the testers replied to all the questions except the last one that was answered by 4 of the testers.

The first question goal was to try and understand how hard the learning curve was during the alpha phase and the results were positive overall considering most at this point the game had three challenges available and most users were able to do two or more Figure 41 even though the game had no instructions yet.

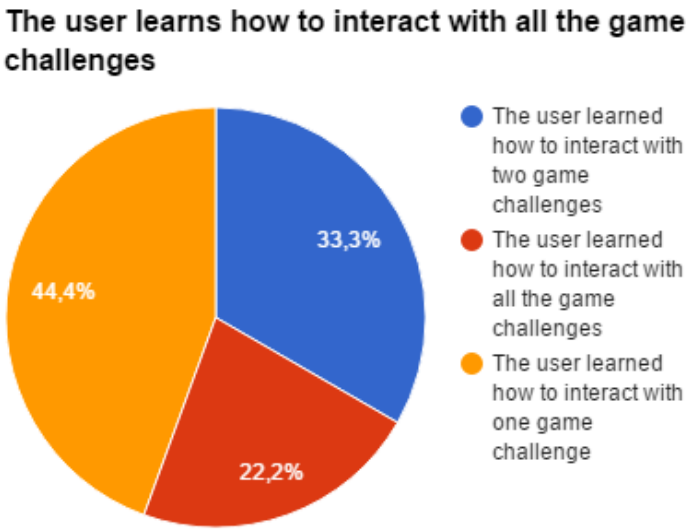


Figure 41 – Alpha Test first question result

The remaining questions graded the game between 1 and 5 with 1 being negative and 5 being positive. Those results are presented in Figure 42.

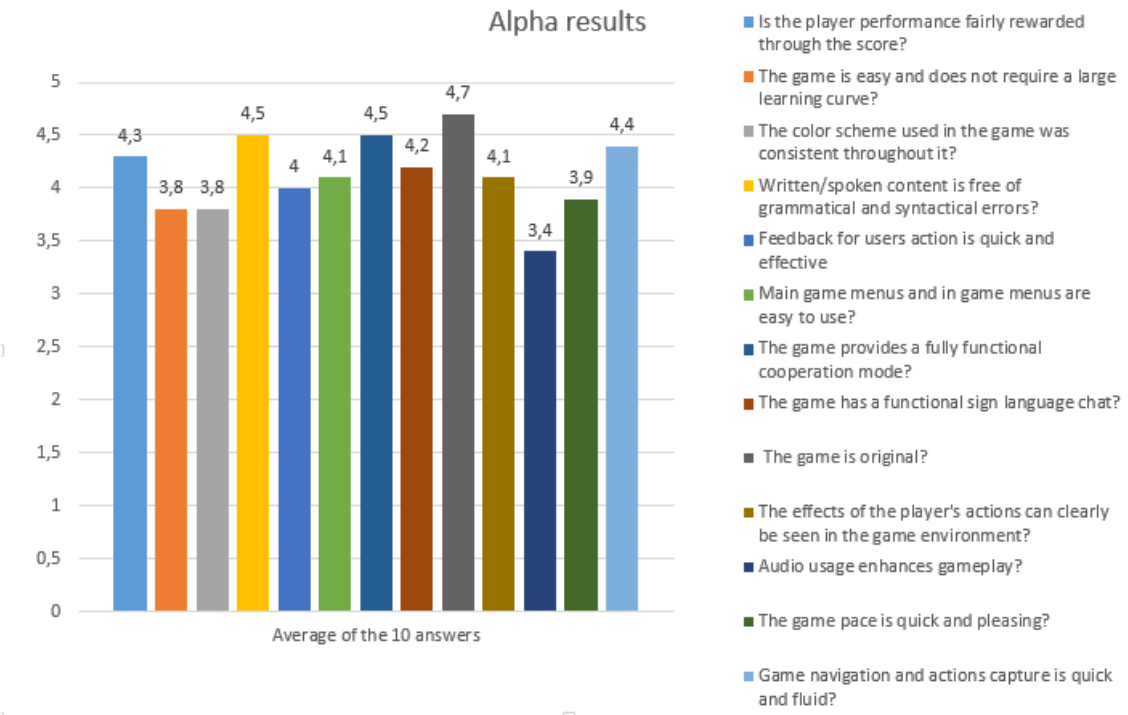


Figure 42 – Alpha test results average of the 13 grading questions based on the 10 testers answers

The answers with that scores the highest rates were the ones referring to the functionalities that were the focus of the alpha version. As for the lowest scoring question about the audio, there was only audio on a few interactions and no music yet at this point thus the bad result.

Overall most questions scored above 4 which was a great feedback for the alpha phase and some suggestions were given by the players that can also be seen on.

Table 2 – Alpha tests suggestions from 4 of the testers.

Alpha test Suggestions
“Sound when collecting the chip at the end of the level”
“I think it was funny if the player changed from the lightside to the dark side in the beggining of the next map and if it was in the dark side change to the ligth side.”
“More ways of interaction.”
“Achievements would be fun.”

The QEF result at the end of the Alpha phase had a quality rate of 72% (Figure 43).

q	D	q_i	Dimension	Q_j	p_{ij}	Factor	pr_{jk}	Requirements	pc_k
72%	0,65	78,13	Pedagogical	91,67	0,75	Learning	10	PL01 - The story line promotes interactive learning curve	75
							10	PL02 - The checkpoints use clear language for the target group	100
							10	PL03 - In the end of the checkpoints the player is awarded by it's performance	100
				37,5	0,25	Assessment	10	PA01 - The game promotes self-assessment	75
							10	PA02 - The game provides a screen to check out the top player results	0
		65,61	Ergonomic	62,5	0,55	Usability	10	EU01 - The game is easy and does not require a large learning curve	25
							10	EU02 - The user should be able to retry a level to try to achieve a better result	50
							10	EU03 - A help button is provided	0
							10	EU04 - The game difficulty its increased by level	100
							10	EU05 - The common element's colors used in the game were consistent throughout it	75
							10	EU06 - Written/spoken content is free of grammatical and syntactical errors	75
							10	EU07 - Feedback for users action is quick and effective	100
							10	EU08 - Main game menus and in game menus are easy to use	75
							10	EU09 - Options in game menu correctly do what their supposed to	50
							10	EU10 - A user can leave and restart the level anytime during the gameplay	50
							10	EU11 - The game should provide a cooperative mode for multiplayer	100
							10	EU12 - The game has a sign language chat	50
				69,44	0,45	Game Play	10	EG01 - The game story is well represented in the game sets	0
							10	EG02 - The game is original	100
							10	EG03 - The effects of the player's actions can clearly be seen in the game environment	100
							10	EG04 - Audio usage enhances gameplay	50
							10	EG05 - The game pace is quick and pleasing	100
							10	EG06 - The game is challenging and defies the user to beat his own best results	75
							10	EG07 - Game content enhances the game play	100
							10	EG08 - Game navigation and actions capture is quick and fluid	100
							10	EG09 - The game story creates an educational emersive context for the players	0
		50	Technical	50	1,00	Support	10	TS01 - The game is easily updatable	100
							10	TS02 - Usage statistics and user history is saved	0
							10	TS03 - There is a unique entry point to the game	100
							10	TS04 - The connection with the VS transalator is stable and fast	0

Figure 43 - QEF of the game on the alpha phase

This rate was achieved through the metrics on the Annex C. The measured dimension that scored the highest was the pedagogical with 78.13%. Followed by the ergonomic with a rate of 65.61% and finally the technical with 50%. The Technical was the by far the lowest as half the features were not yet implemented in the alpha phase. Also neither the story nor the instructions were added in the alpha phase which explains the values obtained for the usability and gameplay. As for the assessment the biggest drawback was the fact the high score table wasn't yet implemented. Overall the 72% quality rate was very positive considering the available functionalities at this stage.

6.2.2 Results Conclusions

The tests of the alpha phase showed very positive results. The goal for this phase was focused on the basic game play features and those were the ones that scored the highest on both the surveys and the QEF.
The lowest scoring features were the ones yet to be implemented or incomplete as they were planned for the beta phase and needed other features to be implemented first.
Several suggestions were made during the testing phase and they were considered for the Beta development phase.

6.3 Beta Version

In the Beta version the main goal was the implementation of the remaining features that required or were better tested after the implementation of the Alpha. So the following features were implemented: account system, game story, sign recognition though the VS translator, high score system and interface graphic improvements.

6.3.1 Account System

An account system provides an efficient way of implementing game analytics as the players' information is sent to the webserver and stored in the database where the player information is kept and associated with his performance. The game analytics consists on the process of analyzing patterns in a set of data (El-Nasr, et al., 2013). The result of that analysis can be used for several reasons one being obtaining automatic feedback about the game. For instance if a majority of the players is finishing the game with the lowest score possible that tells the developer that the game is probably too hard and needs adjusting. Another important feature of the account system is that upon registering the player is asked to write down his email as it is show in Figure 44 providing the game creator with a way to contact the players. It is also important because it shows how many users have played the game.

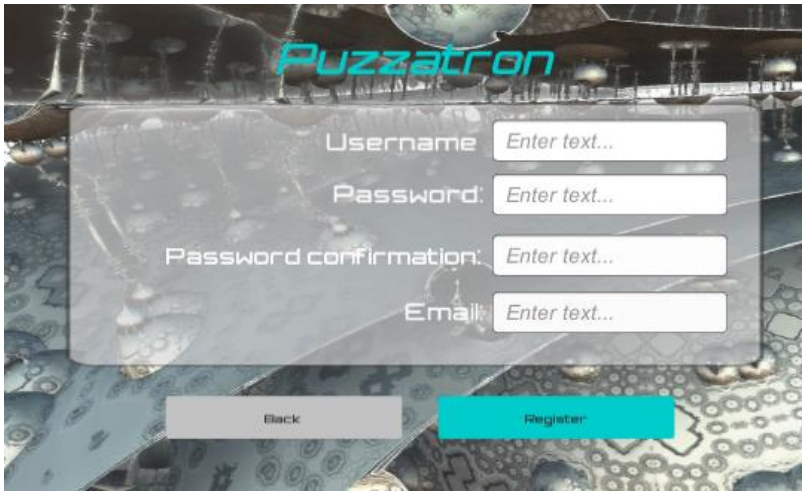


Figure 44 - Puzzaton register interface

However the account system wasn't created to obtain the game statistics alone. With the use of an account system the player score is stored and associated with his account and it allows the future implementation of interesting features such as a friend system.

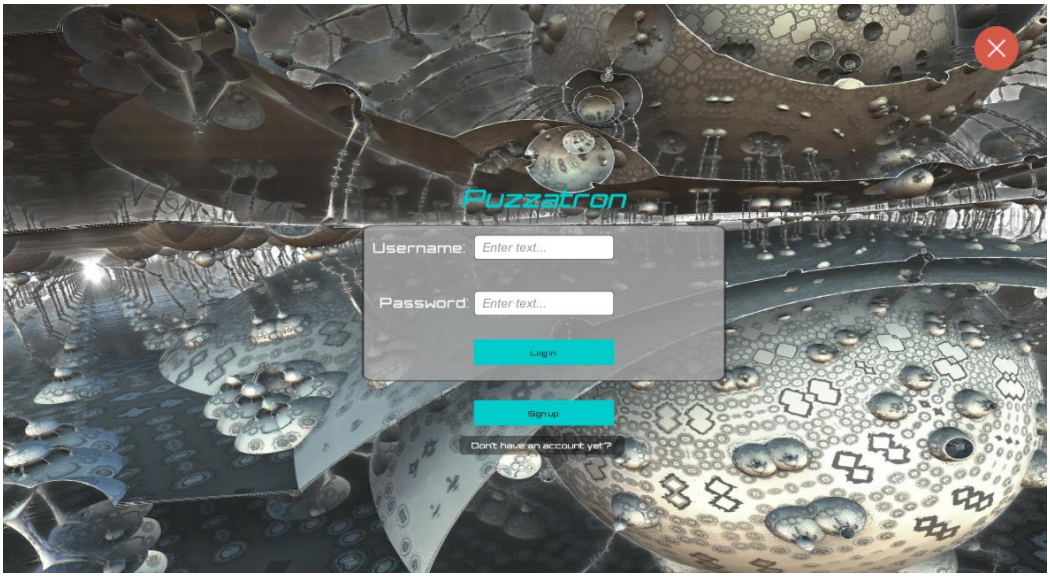


Figure 45 - Puzzatron starting interface

The login interface is illustrated in Figure 45 and it is also the initial interface that the player will see upon starting Puzzatron. It provides a simple login interface with fields for username and password input. A sign up button is also provided in case the user does not yet have an account. Both sign up and log in require the user to be connected to the internet as it performs a request to the webserver that then replies whether the operation was successful or not. The player data is encrypted with MD5 to ensure no leakage of personal information.

6.3.2 Game Story

The game story was based on the concept narrative mentioned on this dissertation. Several pieces of the story are prompted to the player in strategic positions. When a piece of the game story is displayed to the user a window will pop up containing the text. That text will be also translated though the avatar as represented in Figure 46.

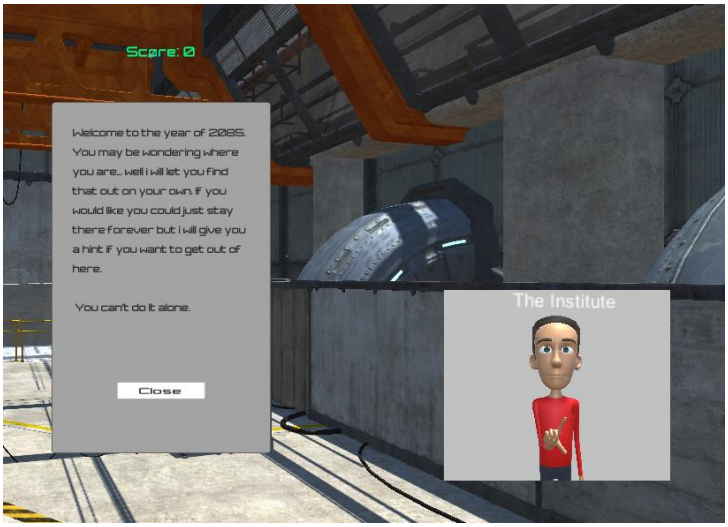


Figure 46 - Window with the first part of the game story and the avatar translating it

The avatar also displays the name "*The Institute*" while translating the game lore as it's the unknown entity keeping the players captive. The window that displays the text is reusable as it contains a lore manager that stores all the game story by parts and can then be called just by a template script that provides the index of the story part until when the text is supposed to be displayed.

```
public void openLore(int end)
{
    StartCoroutine(toggleMouse());
    lorePanel.SetActive(true);
    closeBtn.SetActive(false);
    nextBtn.SetActive(true);
    endLore = end;
    clickNext();
}
```

Code 11 - Function that creates the lore window and sets its limit

Code 1 shows the function that is called by the trigger that sends the index of the end text of the lore. That value is stored the windows is displayed and the mouse becomes visible. The closing or changing of the pages if there are more is handled by the function *clickNext* that uses the variable *endLore* to switch the window button event to act accordingly.

6.3.3 Sign Recognition

The VS translator from gesture to text was implemented only on the Beta version as it requires external hardware components (Kinect and 5DT gloves) thus making it harder to test and debug. In order for this communication between the game and the translator to be possible both sides implemented a socket connection. This connection allows the VS translator to send the result of the translation immediately after the classification of the gestures performed by the user (Figure 47). This results is sent in the form of a string that is then received in the game and added to chat as text and is also translated by the avatar.

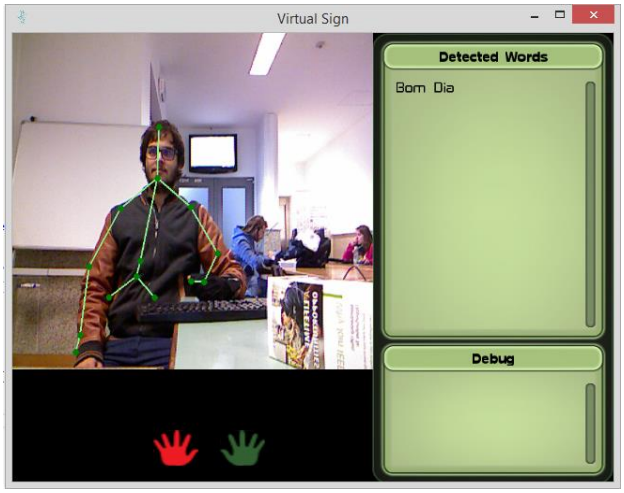


Figure 47 - VS sign to gesture translator after performing the sign “Bom dia”.

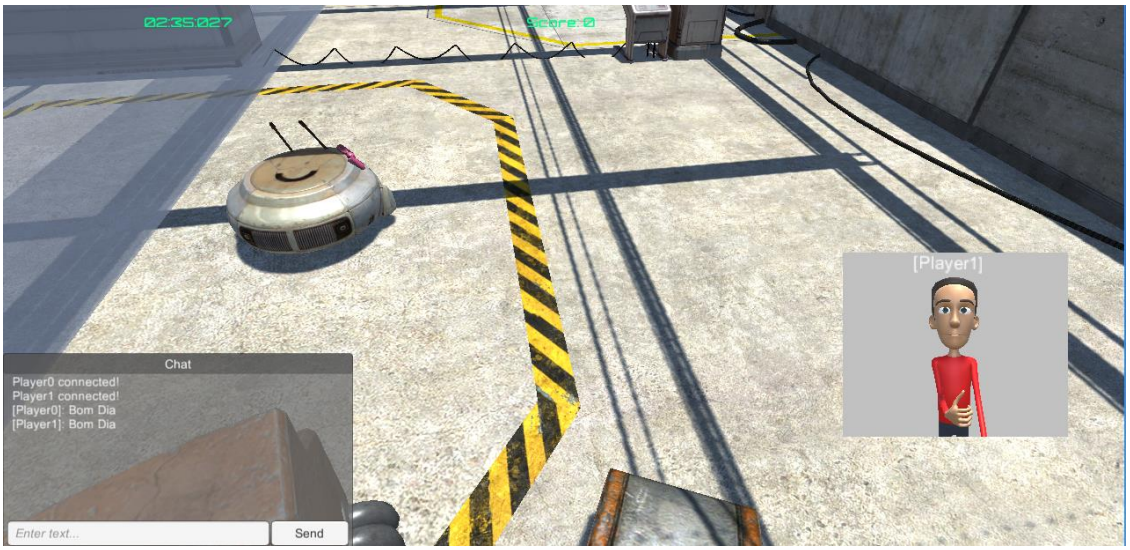


Figure 48 - Game chat after receiving the expression “Bom dia” from the VS translator

Figure 48 shows the avatar translating the expression “Bom Dia” that was received from the VS gesture to text translator and added to the chat as text. The connection between the two applications is handled in a thread to lower the impact on the game performance. The thread waits for an input from the VS translator socket. The socket will trigger an event to add the received text to the chat field that automatically sends it to the avatar for translation.

6.3.4 High score

High scores can motivate the players to play more and perform better on the game (Madigan, 2014). When a game challenge is beaten the user is rewarded though a score according to the user performance. So the better the performance the higher the score of a player, thus the

player with the highest score can be seen as better than the other ones and that alone creates competition.

The high scores are displayed in two different places on the game. One is a list of all the top 20 scores in the game ordered from the highest to the lowest and can be accessed through the settings menu (Figure 49).

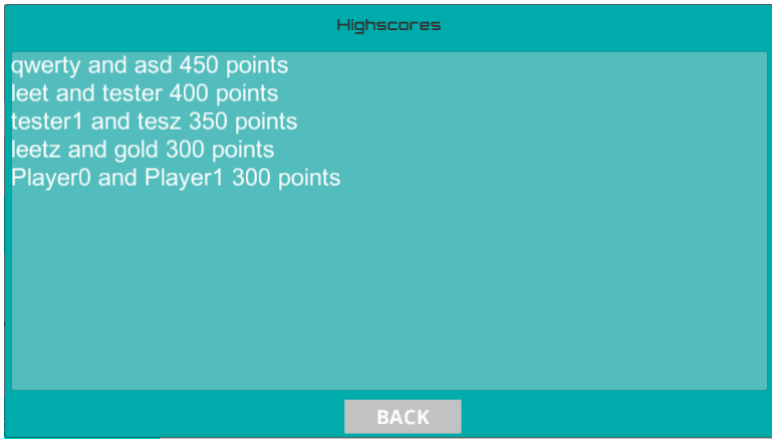


Figure 49 - High score table from the settings menu

The other high score display is at the end of the game and displays only the top three teams of players. The fact that the game ends with a giant wall with the best players names on it motivates the players to try again and reach a score high enough to have their names on that final wall because every player that beats Puzzatron will see it (Figure 50).

Figure 50 - End of the game and the wall with the best scores

All the scores are stored on the web server database and are updated in real time. The high score list is displayed making a HTTP POST requests from the game client to a web service that returns the highest scores.

6.3.5 Interface and Audio Improvements

Some improvements were made to the interface from the Alpha phase to the Beta. The biggest change was the adjustment of the font and the buttons so that all have the same style.

There were some color changes too on the windows and buttons as well (Figure 51).

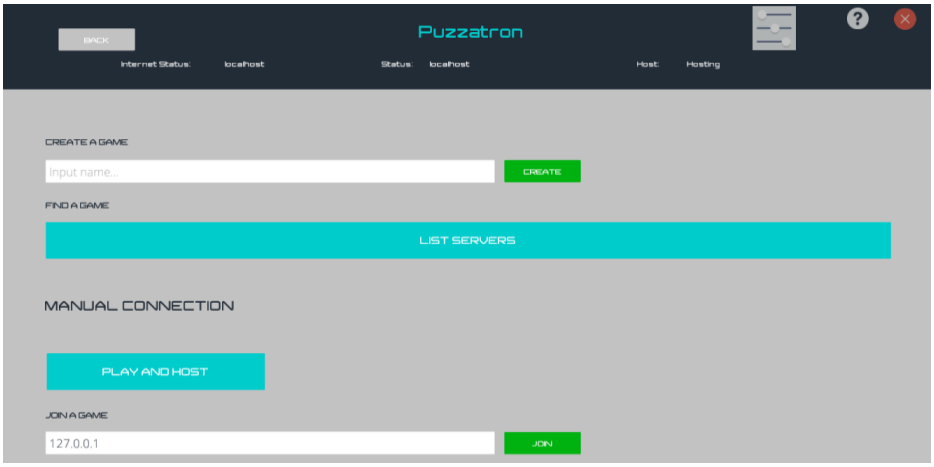


Figure 51 - Updated lobby interface.

Another improvement that was implemented was the insertion of the help button on the top menu that can be access on the lobby or even inside the game. That option opens the instructions window that contains the information on how to play the game and some tips.

As for the audio, some more feedback sounds were added as initially only switches had them. Background music was added too for each level. This was achieved by adding several audio sources with different audio clips throughout the game.

6.4 Beta Testing

Similar to the Alpha tests, Beta tests were evaluated using QEF and through an online survey with Google Forms (Annex D).

The same 10 testers from the Alpha phase (8 male and 2 female) aged between 23 and 65 tested Puzzatron and answered the surveys. However, due to the need of using the VS translator only 5 of those testers were able to try that feature and answer one of the questions. The link with the form and instructions was given to the testers. The forms were answered anonymously.

The Beta test survey focused on the game Technical features, the connection to the VS translator and on the QEF requirements. All the features were implemented on the beta phase and others improved. Thus, the questions focus on all the game important aspects.

6.4.1 Results

The survey of the beta test phase had 20 questions of which the last one was an optional written answer about possible improvements to the game.

The first 15 questions were the same as the alpha test to evaluate the improvement between phases and then an additional 5 questions were added to evaluate the new features.

The first question results show a significant improvement with 70% of the testers learning how to interact with all the game challenges (Figure 52).

Figure 52 – Number of challenges the players learn result chart

The remaining questions graded the game between 1 and 5 with 1 being the worst and 5 being the best. Those results are presented in Figure 53 and there is an improvement from an average of 4 to 4.5 out of 5 from the alpha to the beta phase. One of the most important points which is the cooperation system scored the highest with a value of 4.9 out of 5.

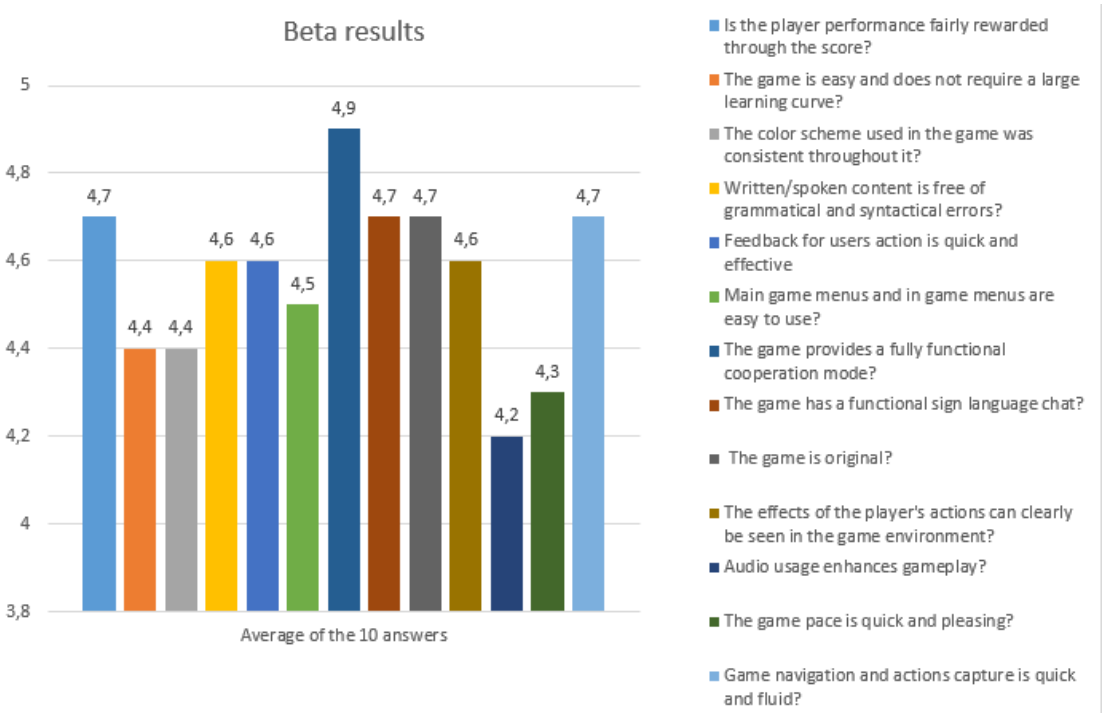


Figure 53 – Beta results of the first 13 questions

The fact that not a single test scored below 4 also shows a great improvement from the alpha phase. The testers gave the audio the lowest score but there is still an increase of 0.8 from the alpha phase.

The beta phase had several new features that were not considered on the alpha tests. The results of the tests for those new features are shown in Figure 54.

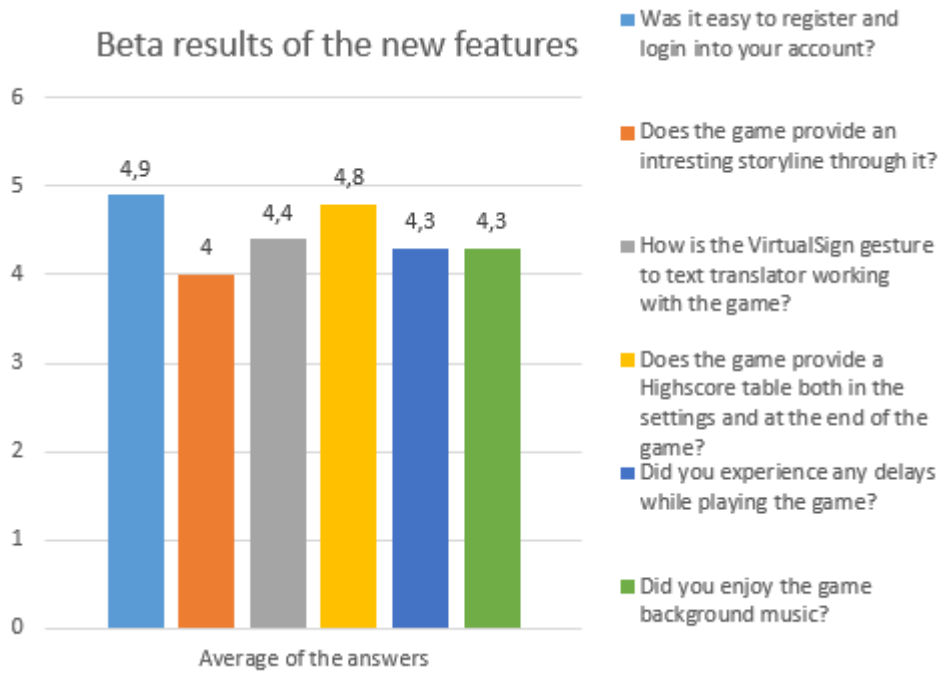


Figure 54 – Chart of the beta results of the new features

The results for the new features were impressive scoring an average of 4.45 out of 5. All the testers found the account system very easy to use, thus scoring the highest value on the evaluation. Once again none of the tests had a value lower than 4. The storyline scored the lowest value which was 4.

Four testers have inputted suggestions. Those can be seen on Table 3.

Table 3 – Beta tests suggestions from 4 of the testers

Beta test Suggestions
“The game should allow the use of a gamepad to play it.”
“Allow to pick the level where to start.”
“I think that it would be nice to have any single games options”
“More maps”

The QEF result at the end of the Beta was 95% quality rate (Figure 55).

This rate was achieved using the same metrics as in the alpha test. The most important dimension of the Beta phase was the Technical where most features were implemented. This dimension obtained a quality rate of 100% with all the requirements being fully achieved through the implementation of the account and high score systems to store the data and the connection to the VS translator. On the Pedagogical dimension there was a slight increase also

because of the high score that influenced the requirements related to the assessment factor making the Pedagogical dimension reach a quality rate of 90.63%.

Finally, the Ergonomic dimension had a significant increase from the alpha phase due to several improvements done for the usability and game play. The interfaces were improved, the help button was added, the colors and sounds were improved as well. All those improvements affected several of the requirements achieving an 88.41% quality rate for the Ergonomic dimension.

q	D	qi	Dimension	Q_j	p_{ij}	Factor	pr_{jk}	Requirements	pc_{jk}
95%	0,15	90,63	Pedagogical	91,67	0,75	Learning	10	PL01 - The story line promotes interactive learning curve	75
							10	PL02 - The checkpoints use clear language for the target group	100
							10	PL03 - In the end of the checkpoints the player is awarded by it's performance	100
				87,5	0,25	Assessment	10	PA01 - The game promotes self-assessment	75
							10	PA02 - The game provides a screen to check out the top player results	100
		88,41	Ergonomic	81,25	0,55	Usability	10	EU01 - The game is easy and does not require a large learning curve	25
							10	EU02 - The user should be able to retry a level to try to achieve a better result	50
							10	EU03 - A help button is provided	100
							10	EU04 - The game difficulty its increased by level	100
							10	EU05 - The common element's colors used in the game were consistent throughout it	100
							10	EU06 - Written/spoken content is free of grammatical and syntactical errors.	75
							10	EU07 - Feedback for users action is quick and effective	100
							10	EU08 - Main game menus and in game menus are easy to use	75
							10	EU09 - Options in game menu correctly do what their supposed to	50
							10	EU10 - A user can leave and restart the level anytime during the gameplay	100
							10	EU11 - The game should provide a cooperative mode for multiplayer	100
							10	EU12 - The game has a sign language chat	100
				97,22	0,45	Game Play	10	EG01 - The game story is well represented in the game sets	100
							10	EG02 - The game is original	100
							10	EG03 - The effects of the player's actions can clearly be seen in the game environment	100
							10	EG04 - Audio usage enhances gameplay	100
							10	EG05 - The game pace is quick and pleasing	100
							10	EG06 - The game is challenging and defies the user to beat his own best results	75
							10	EG07 - Game content enhances the game play	100
							10	EG08 - Game navigation and actions capture is quick and fluid	100
							10	EG09 - The game story creates an educational emersive context for the players	100
		100	Technical	100	1,00	Support	10	TS01 - The game is easily updatable	100
							10	TS02 - Usage statistics and user history is saved	100
							10	TS03 - There is a unique entry point to the game	100
							10	TS04 - The connection with the VS transalator is stable and fast	100

Figure 55 - QEF of the game on the Beta phase

6.4.2 Results Conclusions

There is a noticeable improvement in both the surveys and the QEF evaluation from the alpha phase to the beta.

As for the surveys, all the core features were well accepted by the testers with an increase of 0.5 over the alpha phase to the beta. The only drawback was the audio and the story but even those scored above 4 out of 5. One of the main concerns was also the latency while playing the game. Even though some testers claimed to experience it, it happened very rarely and could just be caused by the internet connection where the application was being used. As for the new features, the account system had a great acceptance and the testers were pleased with the chat translation. The fact that the cooperation and the chat had such high scores on the evaluation was a great accomplishment as those requirements are the very core of the project.

As for the QEF the alpha phase quality rate was 72% and on the end of the beta phase it reached 95% which is impressive. Most of the requirements acquired a perfect score. The biggest issue was the learning curve. Despite the fact the game instructions are always available to the player, there are some aspects that could be improved increasing the ease of

use, such as, a dynamic difficulty adjustment that tips the player. Nonetheless, the usability factor, which was the lowest, still acquired a quality rate of 81.25%.

According to the testers, for the most part, all the features were improved, however there were several suggestions for future work. Overall the final results prove to be better than anticipated and all the features were implemented successfully.

7 Conclusion

The main goal of this dissertation was to improve the social inclusion of the deaf community through the creation of a cooperation game. The game Puzzatron was fully developed and tested, together with its components, by several individuals. One of the differentiating factors that makes this game so unique is the fact there is no record of a cooperation game that integrates a bi-directional sign language translator like this does.

The main target of this project was the deaf community and those surrounding them such as family and friends. The sign language has its own syntax and semantic which makes it harder for the deaf to be able to fluently read written text. However, Puzzatron provides a fun way to spend time together even if the two players are unable to speak each other language as the game will handle the translations for them in real time, thus removing the need of the deaf to read and write even though they still can if they choose to.

Puzzatron was evaluated by 10 testers with ages ranging from 23 to 65 years old some experienced players and some inexperienced. The VS translator component was also tested and validated by sign language experts and deaf users. Those results can be consulted on the Annex E. The VirtualSign project is being successfully disseminated throughout the world. The list of the current dissemination can be seen on Annex F.
Although there were a limited amount of testers, mostly due to the restrictions the project itself was subjected to for instance with the use of the VS translator, the results are still positive as they were consistent with all the testers in both alpha e beta phases.

Overall the game has a great potential according to the results and the insertion of a component for the deaf was a success. Puzzatron proves that it is possible to have any game accessible to the deaf community using their own language.

7.1 Future work

As for future work there are several suggestions that were given by the testers in both of the testing phases. Some of the suggestions of the alpha were implemented into the beta but some were not such as implementing achievements and allowing the players to switch types.

During the Beta tests there were also some interesting suggestions such as allowing the player to play with a game pad instead of the keyboard and mouse, which could be useful especially for the deaf player since the player has to stay in front of the Kinect at a fixed distance and it's not always easy to find a place for the keyboard and mouse at that distance.

As for the translation, in future versions, the user should be able to choose different avatars for the translation as well. Another important feature is switching languages. The VS translator currently supports Libras and PSL but it is expanding and the system is already implemented to able to adjust to that expansion. So for future development there should be an option to let the player pick which sign language the avatar will be translating to.

Also, further improvements on the translator are being researched and alternative hardware's that may remove the need of the gloves or the Kinect thus making the project more accessible. Also alternative hardware could provide tools to implement the game in different platforms that just Windows.

Finally, the VS translator from gestures into text could be fully incorporated within the game allowing for the creation of an asset that could then be easily applied to other games opening a vast set of possibilities for the deaf social inclusion.

References

5DT, n.d. *Data Gloves.* [Online]
Available at: http://www.5dt.com/?page_id=34
[Accessed July 2016].

Activision Publishing, Inc, 2016. *Call of Duty.* [Online]
Available at: https://www.callofduty.com/
[Accessed September 2016].

Adobe Systems Incorporated, 2016. *Mixamo.* [Online]
Available at: https://www.mixamo.com/
[Accessed July 2016].

Associação Portuguesa de Surdos, 2011. *Apsurdos.* [Online]
Available at:
http://www.apsurdos.org.pt/index.php?option=com_content&view=article&id=43&Itemid=57
[Accessed June 2016].

Autodesk Maya, 2016. *Inverse Kinematics.* [Online]
Available at: https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2016/ENU/Maya/files/GUID-07C3BA47-32BB-477B-B6C5-1090E5C9B81C-htm.html
[Accessed July 2016].

Autodesk, 2016a. *Autodesk Maya.* [Online]
Available at: http://www.autodesk.com/products/maya/overview
[Accessed June 2016].

Autodesk, 2016b. *Autodesk at the Academy Awards.* [Online]
Available at: http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=15300670
[Accessed June 2016].

Baltazar, A. B., 2010. *Dicionário de Língua Gestual Portuguesa.* [Online]
Available at: https://www.portoeditora.pt/produtos/ficha/dicionario-de-lingua-gestual-portuguesa/3501376

BARD, N., 2015. *UNITY COMES TO LINUX: EXPERIMENTAL BUILD NOW AVAILABLE.* [Online]
Available at: https://blogs.unity3d.com/pt/2015/08/26/unity-comes-to-linux-experimental-build-now-available/
[Accessed June 2016].

Collins, K., 2008. *Game sound: an introduction to the history, theory, and practice of video game music and sound design.* s.l.:Mit Press.

Correia, G., 2016. *Observador.* [Online]
Available at: http://observador.pt/2016/07/03/o-videojogo-portugues-que-nasce-de-uma-historia-de-amor-com-9-110-quilometros-de-distancia/
[Accessed August 2016].

Correia, I., 2012. Entre Línguas se (des) constrói o texto: interferência linguística da Língua Gestual Portuguesa no Português. *Exedra: Revista Científica 6,* pp. 57-66.

Creighton, R. H., 2010. *Unity 3D game development by example: A seat-of-your-pants manual for building fun, groovy little games quickly.* s.l.:Packt Publishing Ltd.

El-Nasr, M. S., Drachen, A. & Alessandro, C., 2013. *Game analytics: Maximizing the value of player data..* s.l.:Springer Science & Business Media.

Escudeiro, P., 2013. Virtual sign translator. *International Conference on Computer, Networks and Communication Engineering (ICCNCE 2013).,* pp. 290-292.

Escudeiro, P. & Bidarra, J., 2008. Quantitative Evaluation Framework (QEF).. *RISTI: Revista Ibérica de Sistemas e Tecnologias de Informação,* pp. 16-27.

Fitnect, 2013. *fitnect.* [Online]
Available at: http://www.fitnect.hu/

Free Lives, n.d. *Broforce.* [Online]
Available at: http://www.broforcegame.com/
[Accessed August 2016].

Frozenbyte, n.d. *Frozenbyte games.* [Online]
Available at: http://www.frozenbyte.com/games/
[Accessed August 2016].

GestSure, n.d. *gestsure.* [Online]
Available at: http://www.gestsure.com/

Guthrie, R., 2011. *XBox Kinect Software Development Kit SDK Beta Released to Public.* [Online]
Available at: https://blogs.msdn.microsoft.com/mis_laboratory/2011/06/27/xbox-kinect-software-development-kit-sdk-beta-released-to-public/
[Accessed June 2016].

IBM, n.d. *How Sockets Work.* [Online]
Available at: http://www.ibm.com/support/knowledgecenter/ssw_ibm_i_73/rzab6/howdosockets.htm
[Accessed August 2016].

Joyce, A., Paquin, R. & Pigneur., Y., 2015. *The triple layered business model canvas: A tool to design more sustainable business models.* s.l., ARTEM Organizational Creativity International Conference..

Kinect, 2010. *Kinect for windows.* [Online]
Available at: https://developer.microsoft.com/pt-pt/windows/kinect

Kuchera, B., 2009. *Co-op gaming is here to stay: Ars helps you find players.* [Online]
Available at: http://arstechnica.com/gaming/2009/01/co-op-gaming-is-here-to-stay-ars-helps-you-find-players/
[Accessed July 2016].

LEGO, 2016. *Lego Marvel Super Heros.* [Online]
Available at: https://www.lego.com/en-us/marvelsuperheroes
[Accessed June 2016].

Madigan, J., 2014. *Psychology of High Scores, Leaderboards, & Competition.* [Online]
Available at: http://www.psychologyofgames.com/2014/11/psychology-of-high-scores-leaderboards-competition/
[Accessed July 2016].

MAP Technology, 2015. *Projeto Giulia "A mão que fala"- Formulário de Aquisição.* [Online]
Available at: http://maptechnology.com.br/pt-br/novidades/projeto-giulia-m%C3%A3o-que-fala-formul%C3%A1rio-de-aquisi%C3%A7%C3%A3o
[Accessed June 2016].

Microsoft Research, 2013. *Kinect Sign Language Translator – part 1.* [Online]
Available at: https://www.microsoft.com/en-us/research/kinect-sign-language-translator-part-1/
[Accessed July 2016].

Microsoft, 1975. *microsoft.* [Online]
Available at: https://www.microsoft.com/pt-pt/

Microsoft, 2011. *Kinect for Windows SDK to Arrive Spring 2011.* [Online]
Available at: https://blogs.technet.microsoft.com/microsoft_blog/2011/02/21/kinect-for-windows-sdk-to-arrive-spring-2011/
[Accessed June 2016].

Microsoft, 2016. *.NET.* [Online]
Available at: https://www.microsoft.com/net/intro
[Accessed July 2016].

Microsoft, 2016b. *Skeletal Tracking.* [Online]
Available at: https://msdn.microsoft.com/en-us/library/hh973074.aspx
[Accessed August 2016].

Microsoft, 2016. *Kinect for Xbox 360.* [Online]
Available at: http://www.xbox.com/en-US/xbox-360/accessories/kinect
[Accessed July 2016].

Moreno, S., 2009. Can music influence language and cognition?. *Contemporary Music Review,* pp. 329-345.

Motionsavvy, 2016. *Motionsavvy Uni.* [Online]
Available at: http://www.motionsavvy.com/
[Accessed June 2016].

MySQL, 2016a. *Supported Platforms: MySQL Database.* [Online]
Available at: http://www.mysql.com/support/supportedplatforms/database.html
[Accessed August 2016].

MySQL, 2016b. *MySQL Customers.* [Online]
Available at: http://www.mysql.com/customers/
[Accessed August 2016].

Padden, C. & Ramsey, C., 2000. American Sign Language and reading ability in deaf children. *Language acquisition by eye,* pp. 65-89.

Pantel, L. & Lars, C. W., 2002. On the impact of delay on real-time multiplayer games.. *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video,* pp. 23-29.

Peterson, L. E., 2009. *K-nearest neighbor..* [Online]
Available at: http://scholarpedia.org/article/K-nearest_neighbor
[Accessed August 2016].

phpMyAdmin, 2016. *phpMyAdmin.* [Online]
Available at: https://www.phpmyadmin.net/
[Accessed September 2016].

ProDeaf, 2016a. *Solutions.* [Online]
Available at: http://www.prodeaf.net/en-us/Solucoes
[Accessed February 2016].

ProDeaf, 2016b. *Prodeaf.* [Online]
Available at: http://www.prodeaf.net/en-us/OQueE
[Accessed June 2016].

Rouse III, R., 2010. *Game design: Theory and practice.* s.l.:Jones & Bartlett Learning.

Sacks, O. W., 1989. *Seeing Voices: A Journey Into the World of the Deaf.* s.l.:University of California Press.

Showleap, 2015. *Showleap blog.* [Online]
Available at: http://blog.showleap.com/2015/12/2016-nuestro-ano/
[Accessed July 2016].

Showleap, 2016. *Showleap.* [Online]
Available at: http://www.showleap.com/
[Accessed July 2016].

SNK Playmore, 2008. *SNK PLAYMORE PRODUCTS.* [Online]
Available at: https://www.snkplaymore.co.jp/us/game/products/?prd_kw=metal+slug
[Accessed July 2016].

SONY, 2016. *Little Big Planet.* [Online]
Available at: http://littlebigplanet.playstation.com/
[Accessed June 2016].

Start ASL, 2016. *Fun American Sign Language Game List.* [Online]
Available at: https://www.start-american-sign-language.com/sign-language-game_html
[Accessed June 2016].

Steam, 2011. *Streets of Rage.* [Online]
Available at: http://store.steampowered.com/app/71164/?
[Accessed August 2016].

Steinwart, I. & Andreas, C., 2008. *Support vector machines.* s.l.:Springer Science & Business Media.

surdos, A. p. d., 2011. *Associação portuguesa de surdos.* [Online]
Available at:
http://www.apsurdos.org.pt/index.php%3Foption%3Dcom_content%26view%3Darticle%26id%3D41%26Itemid%3D8

Thalmic Labs, 2016. *Myo Gesture Control Armband.* [Online]
Available at: https://www.myo.com/
[Accessed June 2016].

The International Arcade Museum, n.d. *Gauntlet.* [Online]
Available at: http://www.arcade-museum.com/game_detail.php?game_id=7922
[Accessed August 2016].

translator, s. l., 2013. *Kinect sign language translator.* [Online]
Available at: https://www.microsoft.com/en-us/research/kinect-sign-language-translator-part-1/

Unity Technolgies, 2016f. *NetworkIdentity.* [Online]
Available at: https://docs.unity3d.com/Manual/class-NetworkIdentity.html
[Accessed August 2016].

Unity Technologies, 2005. *Unity 3d.* [Online]
Available at: https://unity3d.com/
[Accessed September 2016].

Unity Technologies, 2014. *ANNOUNCING UNET – NEW UNITY MULTIPLAYER TECHNOLOGY.* [Online]
Available at: https://blogs.unity3d.com/2014/05/12/announcing-unet-new-unity-multiplayer-technology/
[Accessed July 2016].

Unity Technologies, 2014b. *Mecanim.* [Online]
Available at: https://unity3d.com/pt/learn/tutorials/topics/animation/animate-anything-mecanim
[Accessed July 2016].

Unity Technologies, 2016a. [Online]
Available at: http://unity3d.com/unity/multiplatform/
[Accessed August 2016].

Unity Technologies, 2016b. [Online]
Available at: https://store.unity.com/
[Accessed August 2016].

Unity Technologies, 2016c. *MonoBehaviour.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/MonoBehaviour.html
[Accessed August 2016].

Unity Technologies, 2016d. *SyncVarAttribute.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/Networking.SyncVarAttribute.html
[Accessed September 2016].

Unity Technologies, 2016g. *Unity - Scripting API: Rigidbody.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/Rigidbody.html
[Accessed September 2016].

Unity Technologies, 2016h. *Unity - Scripting API: ParticleSystem.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/ParticleSystem.html
[Accessed September 2016].

Unity Technologies, 2016i. *Unity - Scripting API: Networking.NetworkClient.Send.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/Networking.NetworkClient.Send.html
[Accessed September 2016].

Unity Technologies, n.d. *Multiplayer and Networking.* [Online]
Available at: https://docs.unity3d.com/Manual/UNet.html
[Accessed July 2016].

Unity Techonologies, 2016e. *CommandAttribute.* [Online]
Available at: https://docs.unity3d.com/ScriptReference/Networking.CommandAttribute.html
[Accessed July 2016].

University of Washington, 2016. *UW undergraduate team wins $10,000 Lemelson-MIT Student Prize for gloves that translate sign language.* [Online]
Available at: http://www.washington.edu/news/2016/04/12/uw-undergraduate-team-wins-10000-lemelson-mit-student-prize-for-gloves-that-translate-sign-language/
[Accessed June 2016].

Valve, 2011. *Portal 2.* [Online]
Available at: http://www.thinkwithportals.com/
[Accessed July 2016].

VirtualSign, 2013. *VirtualSign.* [Online]
Available at: http://193.136.60.223/virtualsign/en/index.php#about
[Accessed 2016].

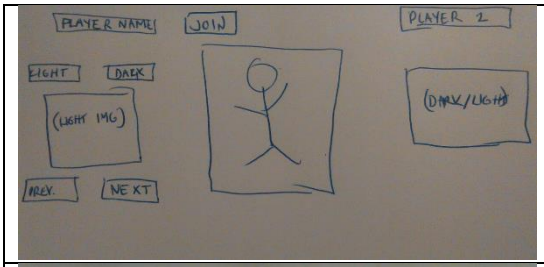
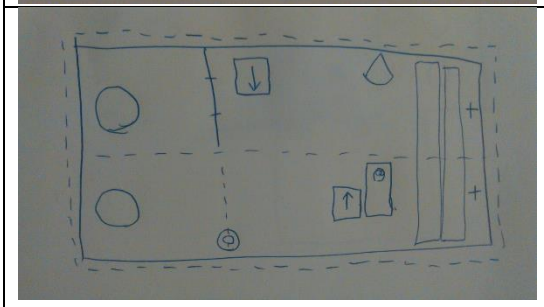
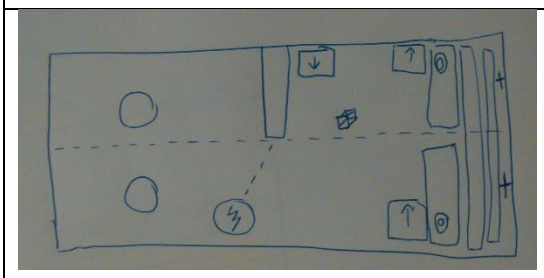
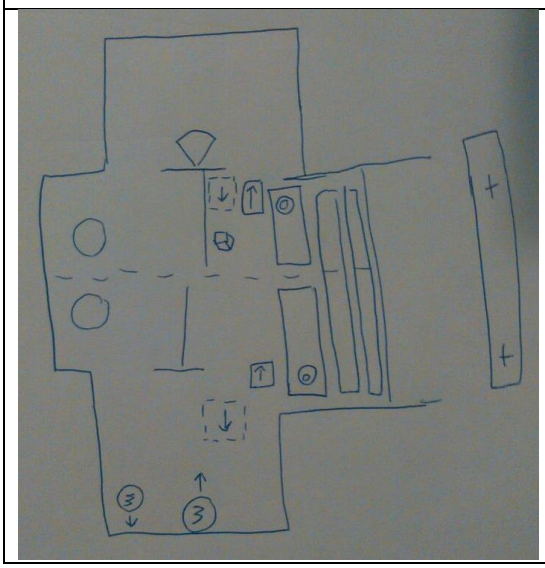
W3C, 2004. *Web Services Glossary.* [Online]
Available at: https://www.w3.org/TR/2004/NOTE-ws-gloss-20040211/#webservice
[Accessed August 2016].

Whitworth, D., 2010. *Kinect gets UK release date.* [Online]
Available at: http://www.bbc.co.uk/newsbeat/article/10996389/kinect-gets-uk-release-date
[Accessed June 2016].

Zafrulla, Z., 2011. American sign language recognition with the kinect.. *Proceedings of the 13th international conference on multimodal interfaces. ,* pp. 279-286.

Annexes

Annex A – Game Storyboard

	Both player enters the lobby and choose the character name, character look and type.
	On the first level player 1(on the lower half) has to press the switch (circle with smaller circle inside) so player 2(on the top half) can get through the door. Player 2 then steps on the lowering platform (square with down arrow) and the rising platform goes up(square with up arrow)(so player 1 can jump on it and press the last switch. Player 2 has to use the dark ability on the camera and then the level is finished when they collect the final items (plus icon). Audio: There is sounds for the switches and background music
	On the second level the player 1 uses his light ability unlocking the door for player 2. Player 2 grabs the box and places it on the lowering platform causing both raising platforms to go up. Both players climb the platforms and press the switches on each side to finish the level by collecting the final items. Audio: There is sounds for the switches and background music
	On the third level Both players move away from each other into further apart rooms where both have to use their special ability. Player 1 has to step on the descending platform which will cause it to disappear and appear on the player 2 area. Player 2 has then to grab the box and place it over the lowering platform causing both raising platforms to go up. Both players climb the platforms and press the switches on each side then they reach an area where they are finally together and there is a huge wall with the high scores (rectangle with the plus signs). Audio: There is sounds for the switches and background music

Annex B – Alpha Test Questionnaire

VirtualSign Coop Game Alpha Test

In this document we ask you to answer the following questions based on your experience playing the game Puzzatron.
Select the answer on the written choices or pick a number between 1 and 5 where 1 means you disagree entirely and 5 means you completely agree.

*Obrigatório

How old are you? *

A sua resposta

Gender *

- ○ Male
- ○ Female

1. The game promotes an interactive learning curve? *

- ○ The user learned nothing about the game interactions
- ○ The user learned how to interact with one game challenge
- ○ The user learned how to interact with two game challenges
- ○ The user learned how to interact with three game challenges
- ○ The user learned how to interact with all the game challenges

2. Is the player performance fairly rewarded through the score? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

3. The game is easy and does not require a large learning curve? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

4. The color scheme used in the game was consistent throughout it? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

5. Written/spoken content is free of grammatical and syntactical errors? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

6. Feedback for users action is quick and effective *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

7. Main game menus and in game menus are easy to use? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

8. The game provides a fully functional cooperation mode? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

9. The game has a functional sign language chat? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

10. The game is original? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

11. The effects of the player's actions can clearly be seen in the game environment? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

12. Audio usage enhances gameplay? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

13. The game pace is quick and pleasing? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

14. Game navigation and actions capture is quick and fluid? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

15. What would you suggest that should be changed or added to the game at this stage?

A sua resposta

SUBMETER

Annex C – Quantitative Evaluation Framework Metrics

Dimension	Pedagogical					
Factor	Learning					
		Wfk - Fullfilment				
Requirement	Metric Evaluation	0	25	50	75	100
PL01 - The story line promotes interactive learning curve	As the user moves forward, he will learn the basics of the game	The user learn nothing about the game interactions	The user learns how to interact with one game challenge	The user learns how to interact with two game challenges	The user learns how to interact with three game challenges	The user learns how to interact with all game challenges
PL02 -The checkpoints use clear language for the target group	The tasks requested are very direct, use few words and use common words	The tasks are illegible	The language used in the tasks is complex, uses less common words and they have grammatical errors	The language used in the tasks is complex and uses less common words	The language used in some of the tasks is complex and uses less common words	Tasks are very direct, use few words and use common words
PL03 - In the end of the checkpoint the player is awarded by it's performance	In the end of the checkpoints the player is awarded by it's performance	There is no feedback	The checkpoint is a static object	The Checkpoint result it's not dynamic	The checkopoint result is dynamic but it have grammatical errors	Dynamic Checkpoint result with no grammatical errors

Dimension	Pedagogical					
Factor	Assessment					
		Wfk- Fullfilment				
Requirement	Metric Evaluation	0	25	50	75	100
PA01 - The game promotes self-assessment	On each level results screen, the user is presented with it's current position in the game and story enabling him to understand is performance based on top results and game	No results presented on level finish	Only current result is presented.	Current result presented and lattest three best scores.	Available graph with 5 previous scores	Results, top scores, graphs and retry button are available.
PA02 - The game provides a screen to check out the top player results	The user can check the top results for the game accessing the option available on the main menu for the game	The option is not available in game menu	No access to functionality	The screen is presented but the results are not available	The screen is presented on option select but only one result is presented	The user can access the functionality, check the results and go back to main menu

Dimension	Ergonomic					
Factor	Usability					
		Wfk - Fullfilement				
Requirement	Metric Evaluation	0	25	50	75	100
EU01 - The game is easy and does not require a large learning curve	For each game screen (menus, in game menus, game levels) the user is presented with a simple tutorial to learn the screen mechanics and its usage. This screen is always poped out on the first time the user sees the screen	Tutorial functionality not available	Tutorial available for initial game menu	Tutorial available for all menus	Tutorial available for all game menus and some levels	Tutorial available for all game menus and game levels
EU02 - The user should be able to retry a level to try to archieve a better result	For each level the game is presented with the final game result and last available results. It should prompt the player to restart the level to retry a best result.	No final results screen presented.	Result screen presented without values or options enabled.	Result screen is presented with results but no options buttons presented.	Result screen is presented with results and continue button presented.	Result screen complete results presented with fully functional retry or continue options available
EU03 - A help button is provided	At all points in the game a help button is presented in the option menu of the game providing the user with help tips.	Not available	Only available in level one	Available in the two first levels	Available in all levels	Available in all levels and menus
EU04 - The game difficulty its increased by level	The game rules and constraints have to reflect in more challenging levels for higher level number. (1- easy, 2 - easy-medium, 3 - medium, 4 - hard)	All levels present same difficulty	Level one is easy and other are impossible	Level one and two are balanced	All levels are balanced	Levels dificulty are ballanced and difficulty is incremental
EU05 - The common element's colors used in the game were consistent throughout it	The game schema is consistent and lean. No color tweaks in menus or game sets.	No visible color schema applied	Weird colors in initial menu and confusing in game sets and other menus.	Some colors don't match with same actions and in game effects.	Colors schema is clear but some weird color effects are presented in animations in game.	Colors are used correctly in game according to initial schema definition
EU06 - Written/spoken content is free of grammatical and syntactical errors.	No grammatical errors are presented to the player in any screen or phase in the game	Errors are visible in all screens	Errors can be easy to find	Errors found in game story and tutorials descriptions	Minor typos found in game menus	No errors presented in any point of the game
EU07 - Feedback for users action is quick and effective	When a player executes a movement that represents an interaction with in game objects (menu options, in game objetcts) a feedback event is presented.	No feedback presented to the players inptu movements and actions	Only available in game menus	Available in game menus and levels (no audio)	Available in game menus and levels (animations and no audio)	All game objects and player actions present audio and visual feedback
EU08 - Main game menus and in game menus are easy to use	The game buttons must present icons that allow players to understand what's the button do and action takes less than 3 seconds to be activated on input cursor on top	No icons or actions available.	Some icons present in game options	Game menu options icons presented but not the in game menu.	Game menu and in game menus presented with icons	Game menu and in game menus presented with icons and activate options under 3 secs cursor on top available.
EU09 - Options in game menu correctly do what their suposed to	The actions provided by the menu and in game menu do what they suposed to do.	No actions happen to any of the buttons presented.	Main game menu start game and leave are enabled	50% of the options buttons act as their suposed to	75% of the options buttons act as their suposed to	All game options are available
EU10 - A user can leave and restart the level anytime during the gameplay	In each level during gameplay the player is presented with a position pattern that enables the game to pause.	No action pattern enables the game to pause.	Game pause pattern enable but no action occours	Game can be paused and leave option is presented	Game is paused and leave and retry option is presented with no actions enable	Player can pause and leave or restart current level
EU11 - The game should provide a cooperative mode for multiplayer	The game must allow the player to create a multiplayer lobby and play with someone else flawlessly.	The user can't play the game	The user can enter the game but noone can join	The user can enter the game and others can join but there is no coordination between clients	The user can create a game and others can join but the connection has unusual high	The player can smoothly play with others
EU12 - The game has a sign language chat	The players can communicate with each other through the sign language chat.	There is no communication.	The messages can only be seen by the sender	The message is sent, the text is displayed but the avatar doesn't translate	The message is sent and translated by the avatar but incorrectly	The messages are sent and translated correctly

Dimension	Ergonomic					
Factor	Game Play					
		Wfk - Fullfilement				
Requirement	Metric Evaluation	0	25	50	75	100
EG01 - The game story is well represented in the game sets	Each level must present the correct story scope boundaries defined in content and game rules.	No content available.	Game level design does not represent correctly game story	Game objectives and times not accordingly to game rules	Some levels do not have game rules applied	All game levels represent correct behaviour accordingly
EG02 - The game is original	The game content is original and no copyrighted content is used in any part of the game	No content available.	No content available.	No content available.	Used some copyrighted content	All game content (audio and images) created by the game team or commercial free
EG03 - The effects of the player's actions can clearly be seen in the game	For each player movement and action an adequated sound, animation or both is presented	No action performed by the system	Animations feedback in game menus	Some animations in levels but not all objects have correct	Audio and animations presented in more than 80% game	Audio and animations feedback presented in 100%
EG04 - Audio usage enhances gameplay	Audio tracks and audio sound effects are pleasent and enhance the player to enjoy the game	No audio present	Audio effects in25% of the game objects interactions with the user are presented	Audio effects in 50% of the game objects interactions with the user are presented	Audio effects in all objects interactions with the user are presented	Audio effects in all objects interactions with the user are presented and background music is available
EG05 - The game pace is quick and pleasing	Transitions, animations and game levels objectives respect timestamps to avoid the player from getting bored and leaving the game.	No animations or transitions	Only game menu visual feedback is provided	At least all in game objects interactions have immediate feedback	All transitions take max of 3 secs and levels lji 2 in game objects interactions have immediate feedback	All transitions take max of 3 secs and in game objects interactions have immediate feedback
EG06 - The game is challenging and defies the user to beat is own best results	The game rules must be well defined avoiding the user to get high results without excellent performance and avoid the user to perform very badly providing in game hints.	No game rules applied	25% game rules applied	50% game rules applied	75% game rules applied	All game rules applied
EG07 - Game content enhances the game play	The game content represents clearly the game story elements	No game elements present	No	Not all	Not all but at least half	Yes
EG08 - Game navigation and actions capture is quick and fluid	When in game menus the players actions, movement and positions are well read by the kinetic device and respond within acceptable time	No game response to device input	The players movement is captured but does not represent the correct input in the game screen	The players movement is well captured and represented on game screen with some major tweaks	The players movement is well captured and represented on game screen but some tweaks are present	The players movement is well captured and represented on game screen
EG09 - The game story creates an educational emersive contest for the players	The game story has a continued objective with an educational purpose, making the player want to advance in the game in order to fulfill the next task	Game not available	Game pace is boring and slow	Game pace is interesting at times but mostly boring	Game pace is boring at times but mostly interesting and motivational	The game is fun and enables the user to keep plaging to complete all game levels

Dimension	Technical					
Factor	Support					
		Wfk - Fullfilment				
Requirement	Metric Evaluation	0	25	50	75	100
TS01 - The game is easily updatable	New updates will be notified in the application, and the user just have to aprove it	No update will be made		The game will be manually fully updated		The game will be automatically fully updated
TS02 - Usage statistics and user history is saved	The usage statistics and game history are saved, and each user have access to them	The game will not save any data	Just the player name will be saved	Just the scores will be saved	The best 3 scores will be saved	All the statistics and the top 3 scores will be saved
TS03 - There is a unique entry point to the game	The game can only be started form the main screen	There is no entry point		The game can be started from anywhere		The game can only be started on the main screen
TS04 - The connection with the VS transalator is stable and fast	The connection with the VS translator doesn't fail and is fast.	Can't establish the connection with the translator	Can connect to the translator but loses connection easily	Connect to the translator but the reply is very slow >5 sec	Connect to the translator but the reply is slow >2 sec	Connect to the translator and has instant reply

Annex D – Beta Test Questionnaire

VirtualSign Coop Game Beta Test

In this document we ask you to answer the following questions based on your experience playing the game.
Select the answer on the written choices or pick a number between 1 and 5 where 1 means you disagree entirely and 5 means you completely agree.

*Obrigatório

How old are you? *

A sua resposta

Editar este formulário

Gender *

- ○ Male
- ○ Female

1. The game promotes an interactive learning curve? *

- ○ The user learned nothing about the game interactions
- ○ The user learned how to interact with one game challenge
- ○ The user learned how to interact with two game challenges
- ○ The user learned how to interact with three game challenges
- ○ The user learned how to interact with all the game challenges

2. Is the player performance fairly rewarded through the score? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

3. The game is easy and does not require a large learning curve? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

4. The color scheme used in the game was consistent throughout it? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

5. Written/spoken content is free of grammatical and syntactical errors? *

Editar este

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

6. Feedback for users action is quick and effective *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

7. Main game menus and in game menus are easy to use? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

8. The game provides a fully functional cooperation mode? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

9. The game has a functional sign language chat? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

10. The game is original? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

11. The effects of the player's actions can clearly be seen in the game environment? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

12. Audio usage enhances gameplay? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

13. The game pace is quick and pleasing? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

14. Game navigation and actions capture is quick and fluid? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

15. Was it easy to register and login into your account? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

16. Does the game provide an intresting storyline through it? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

16. How is the VirtualSign gesture to text translator working with the game?

	1	2	3	4	5	
Not working	○	○	○	○	○	Working perfectly

17. Does the game provide a Highscore table both in the settings and at the end of the game? *

	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

18. Did you experience any delays while playing the game? *

	1	2	3	4	5	
Definitely yes	○	○	○	○	○	Definitely no

19. Did you enjoy the game background music? *

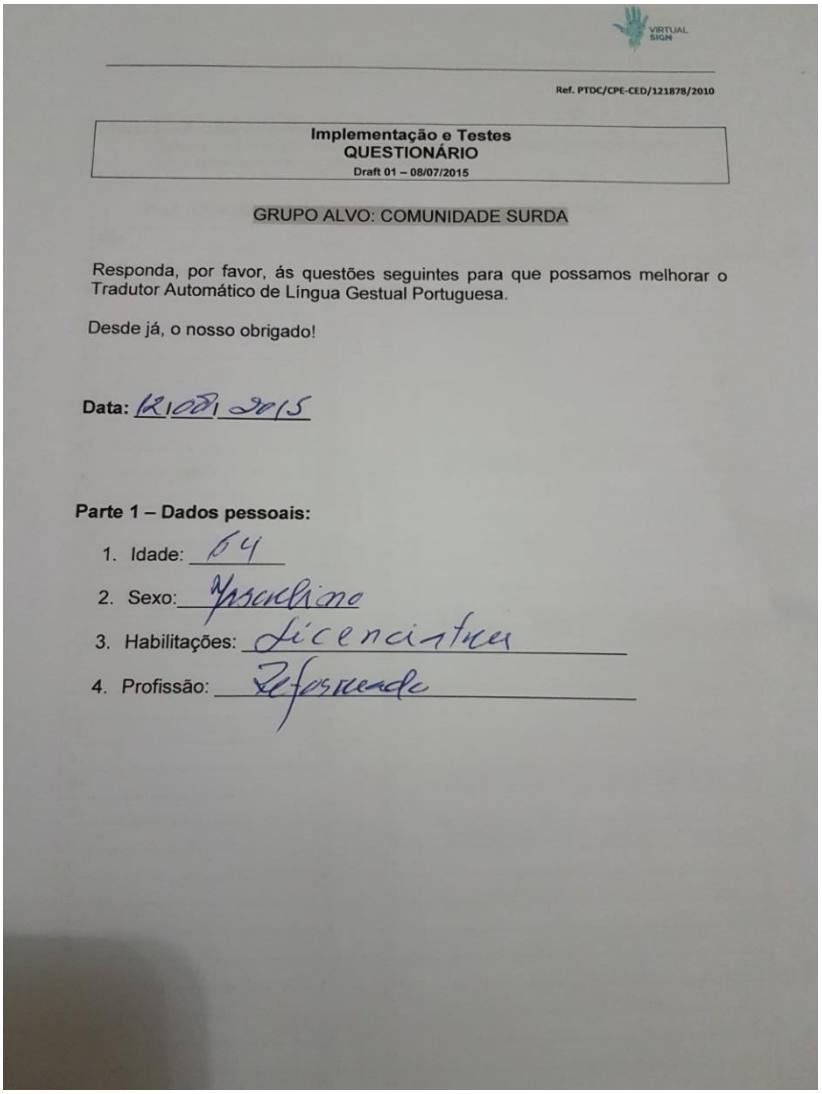
	1	2	3	4	5	
Definitely no	○	○	○	○	○	Definitely yes

20. What would you suggest that should be changed or added to the game at this stage?

A sua resposta

SUBMETER

Annex E – VirtualSign Avatar Translation Evaluation



Ref. PTDC/CPE-CED/121878/2010

Implementação e Testes
QUESTIONÁRIO
Draft 01 – 08/07/2015

GRUPO ALVO: COMUNIDADE SURDA

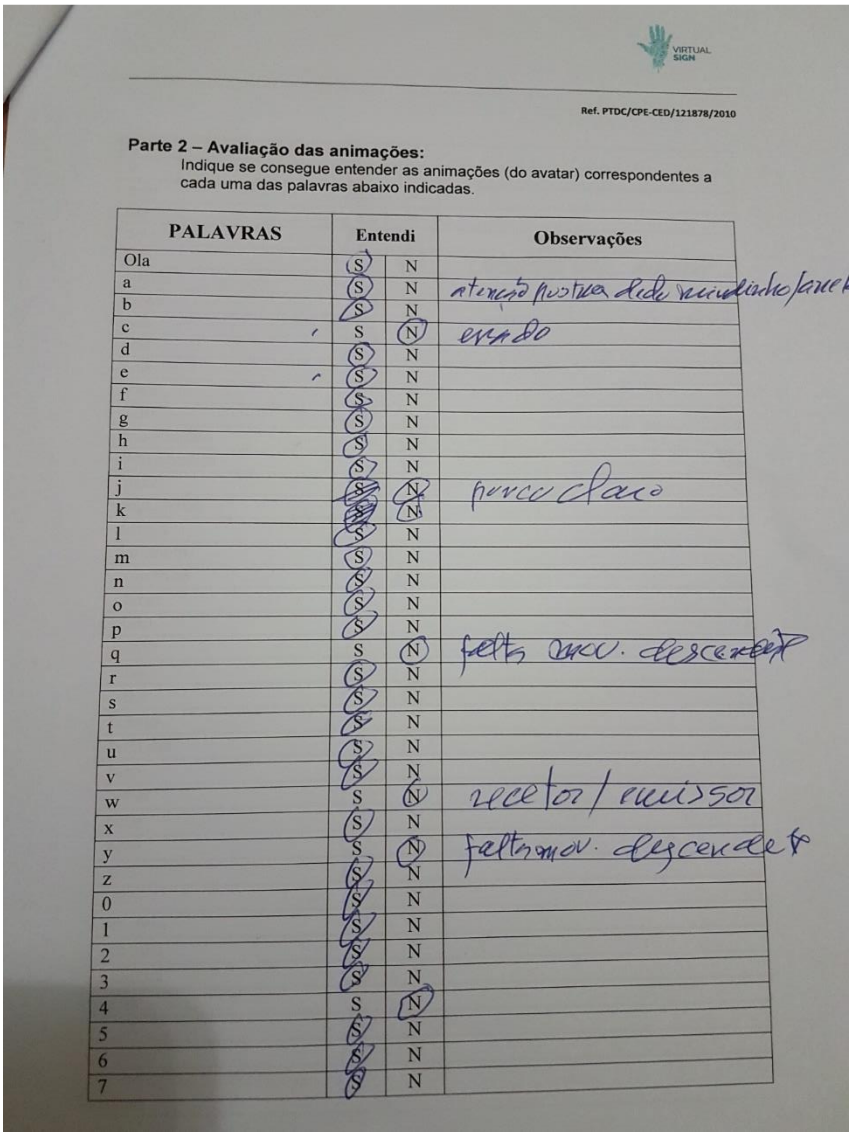
Responda, por favor, ás questões seguintes para que possamos melhorar o Tradutor Automático de Língua Gestual Portuguesa.

Desde já, o nosso obrigado!

Data: 12/08/2015

Parte 1 – Dados pessoais:

1. Idade: 64
2. Sexo: Masculino
3. Habilitações: Licenciatura
4. Profissão: Reformado



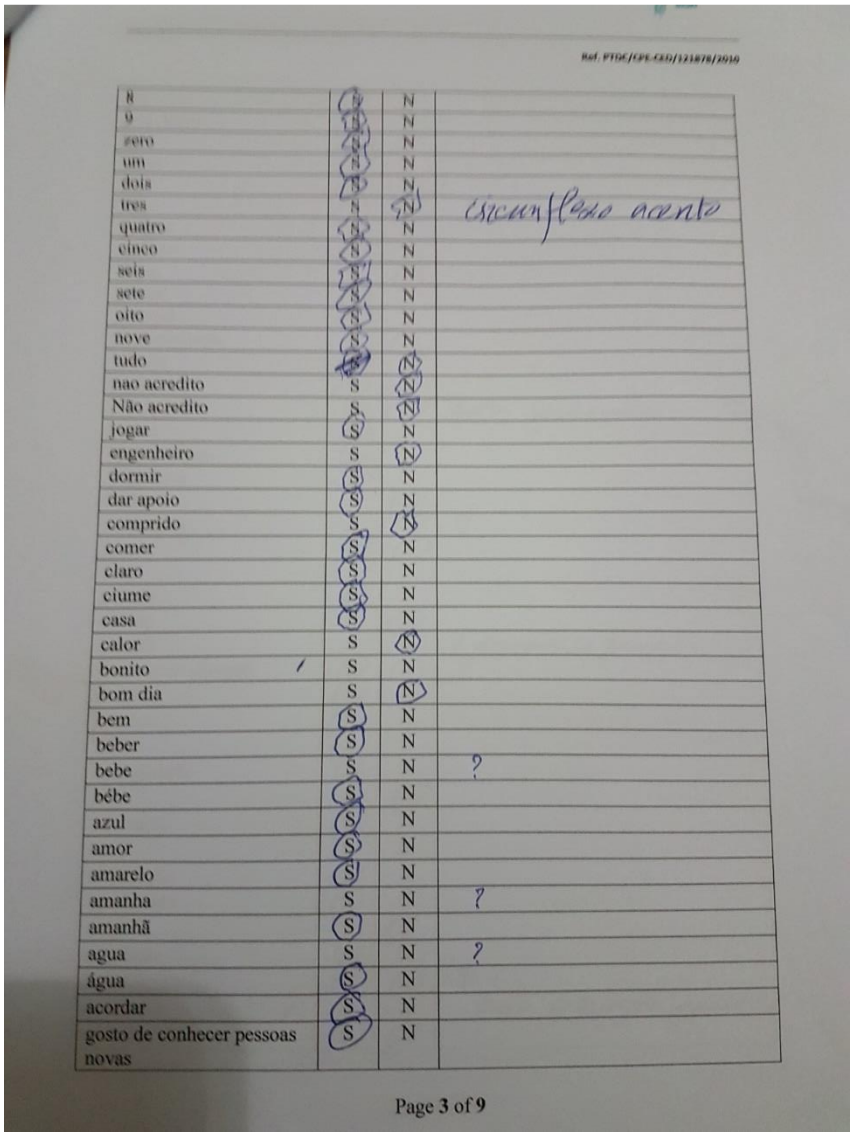
VIRTUAL SIGN

Ref. PTDC/CPE-CED/121878/2010

Parte 2 – Avaliação das animações:

Indique se consegue entender as animações (do avatar) correspondentes a cada uma das palavras abaixo indicadas.

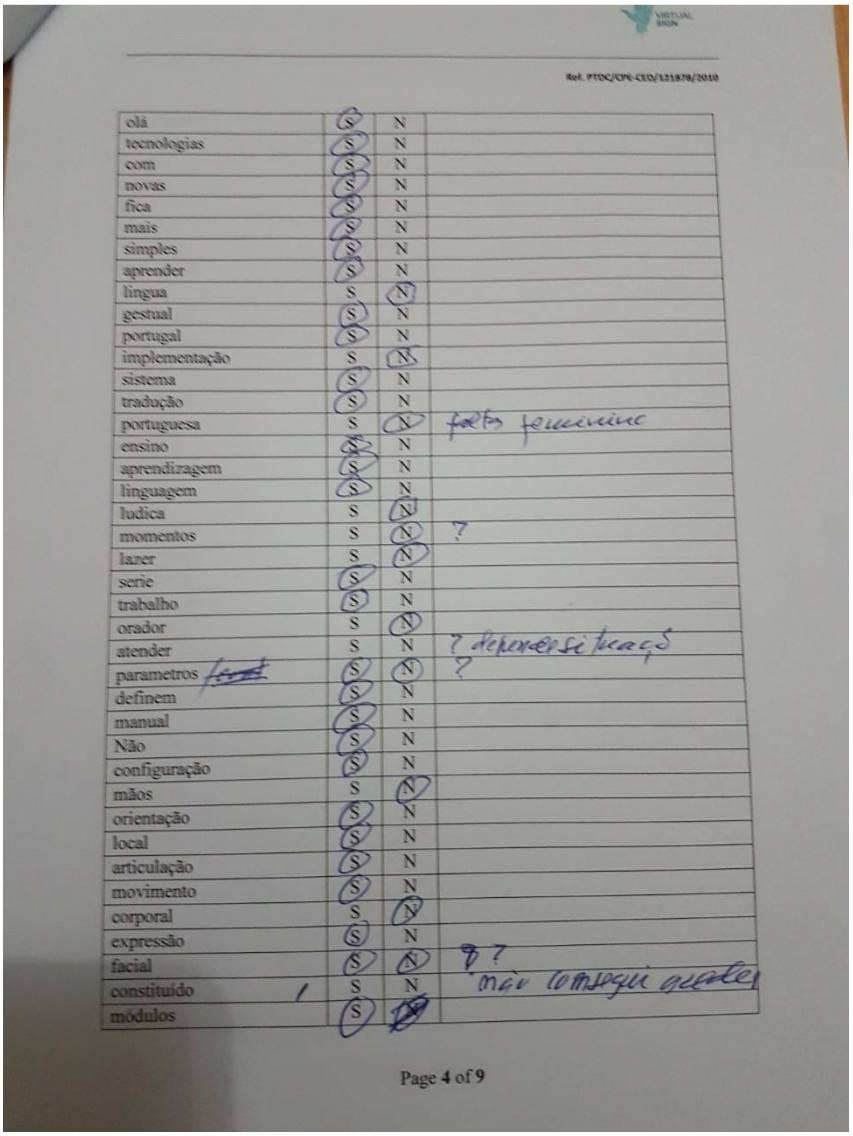
PALAVRAS	Entendi		Observações
Ola	(S)	N	
a	(S)	N	atenção postura lado mão/face
b	(S)	N	
c	S	(N)	errado
d	(S)	N	
e	(S)	N	
f	(S)	N	
g	(S)	N	
h	(S)	N	
i	(S)	N	
j	(S)	(N)	pouco claro
k	(S)	(N)	
l	(S)	N	
m	(S)	N	
n	(S)	N	
o	(S)	N	
p	(S)	N	
q	S	(N)	falta mov. descendente
r	(S)	N	
s	(S)	N	
t	(S)	N	
u	(S)	N	
v	(S)	N	
w	S	(N)	recetor / emissor
x	(S)	N	
y	S	(N)	falta mov. descendente
z	(S)	N	
0	(S)	N	
1	(S)	N	
2	(S)	N	
3	(S)	N	
4	S	(N)	
5	(S)	N	
6	(S)	N	
7	(S)	N	



Ref. PTDC/CPE-CED/121878/2010

8	S	N	
9	S	N	
zero	S	N	
um	S	N	
dois	S	N	
tres	S	N	Escorrega acento
quatro	S	N	
cinco	S	N	
seis	S	N	
sete	S	N	
oito	S	N	
nove	S	N	
tudo	S	N	
nao acredito	S	N	
Não acredito	S	N	
jogar	S	N	
engenheiro	S	N	
dormir	S	N	
dar apoio	S	N	
comprido	S	N	
comer	S	N	
claro	S	N	
ciume	S	N	
casa	S	N	
calor	S	N	
bonito	S	N	
bom dia	S	N	
bem	S	N	
beber	S	N	
bebe	S	N	?
bébe	S	N	
azul	S	N	
amor	S	N	
amarelo	S	N	
amanha	S	N	?
amanhã	S	N	
agua	S	N	?
água	S	N	
acordar	S	N	
gosto de conhecer pessoas novas	S	N	

Page 3 of 9



Ref. PTDC/CPE-CED/121878/2010

olá	S	N	
tecnologias	S	N	
com	S	N	
novas	S	N	
fica	S	N	
mais	S	N	
simples	S	N	
aprender	S	N	
lingua	S	N	
gestual	S	N	
portugal	S	N	
implementação	S	N	
sistema	S	N	
tradução	S	N	
portuguesa	S	N	faltou feminino
ensino	S	N	
aprendizagem	S	N	
linguagem	S	N	
ludica	S	N	
momentos	S	N	?
lazer	S	N	
serie	S	N	
trabalho	S	N	
orador	S	N	
atender	S	N	? depende situação
parametros	S	N	?
definem	S	N	
manual	S	N	
Não	S	N	
configuração	S	N	
mãos	S	N	
orientação	S	N	
local	S	N	
articulação	S	N	
movimento	S	N	
corporal	S	N	
expressão	S	N	
facial	S	N	??
constituido	S	N	não consegui ouvir
módulos	S	N	

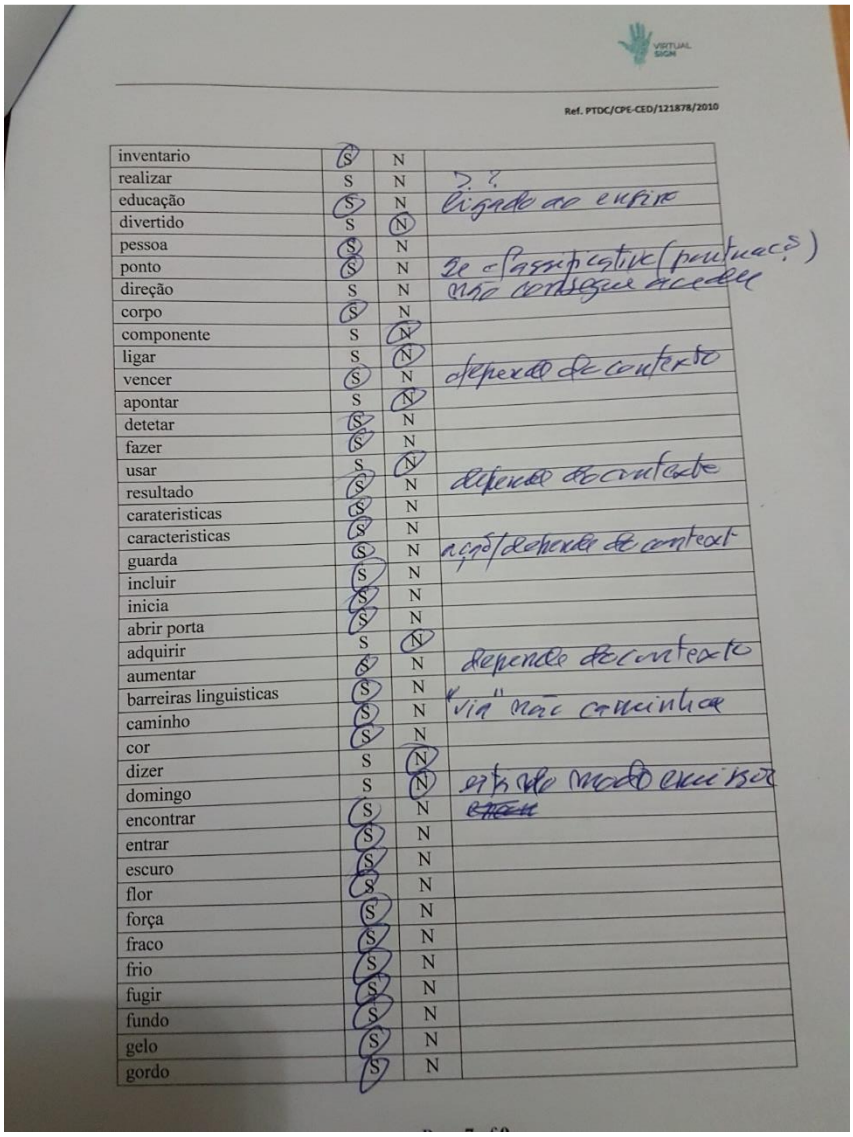
Page 4 of 9

Ref. PTDC/CPS-CED/121878/2010

interligados	S	N	
tradutor	S	N	
gestos	S	N	?
texto	S	N	
jogo	S	N	
didático	S	N	
fornecer	S	N	
informação	S	N	
dividir	S	N	
conquistar	S	N	
inicio	S	N	
definido	S	N	?
adoção	S	N	
dominante	S	N	
término	S	N	
marcado	S	N	
retorno	S	N	
posição	S	N	
neutra	S	N	
mudança	S	N	
abecedário	S	N	
fornecem	S	N	
permitindo	S	N	
estimar	S	N	
fornece	S	N	
comparação	S	N	
séries	S	N	
pontos	S	N	se classificat. vo
efetuada	S	N	
aplicando	S	N	
algoritmo	S	N	
sequencias	S	N	
sequencia	S	N	
resultante	S	N	
constituir	S	N	
palavra	S	N	
usando	S	N	
classificamos	S	N	
análise	S	N	
aspetos	S	N	

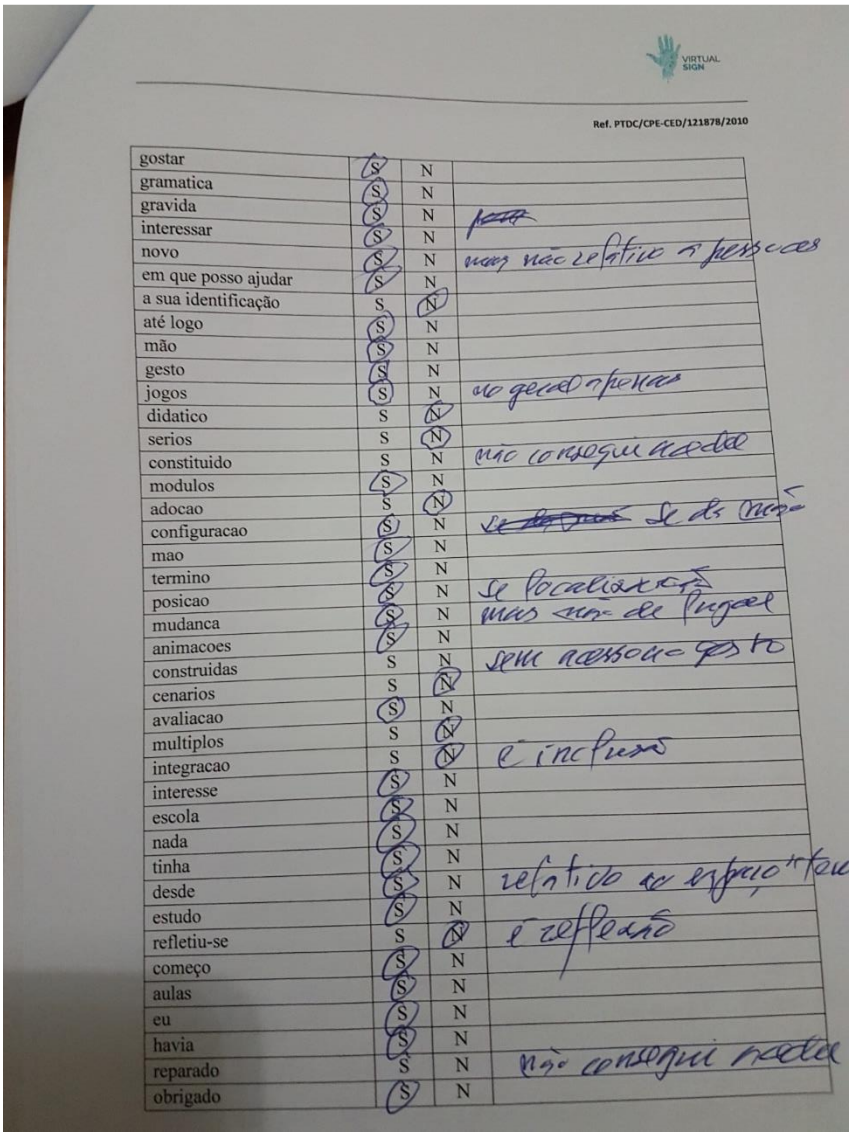
Page 5 of 9

dados	S	N	
recolhidos	S	N	
classificador	S	N	
animações	S	N	
construídas	S	N	
podem	S	N	
utilizadas	S	N	
outros	S	N	
efetua	S	N	
correção	S	N	depende contexto
integração	S	N	isto é incluir
comunicação	S	N	
distância	S	N	
palavras	S	N	
cenários	S	N	
avaliação	S	N	
conhecimentos	S	N	
múltiplos	S	N	
interação	S	N	
cenário	S	N	
alfabeto	S	N	
interativos	S	N	
navegação	S	N	
livre	S	N	
jogador	S	N	
escolhe	S	N	
ordem	S	N	
eventos	S	N	
futurístico	S	N	
frases	S	N	
específicas	S	N	
ativam	S	N	
didatica	S	N	
bomdia	S	N	
mapas	S	N	
larga	S	N	
dimensão	S	N	
ambiente	S	N	
interativo	S	N	
jogabilidade	S	N	



Ref. PTDC/CPE-CED/121878/2010

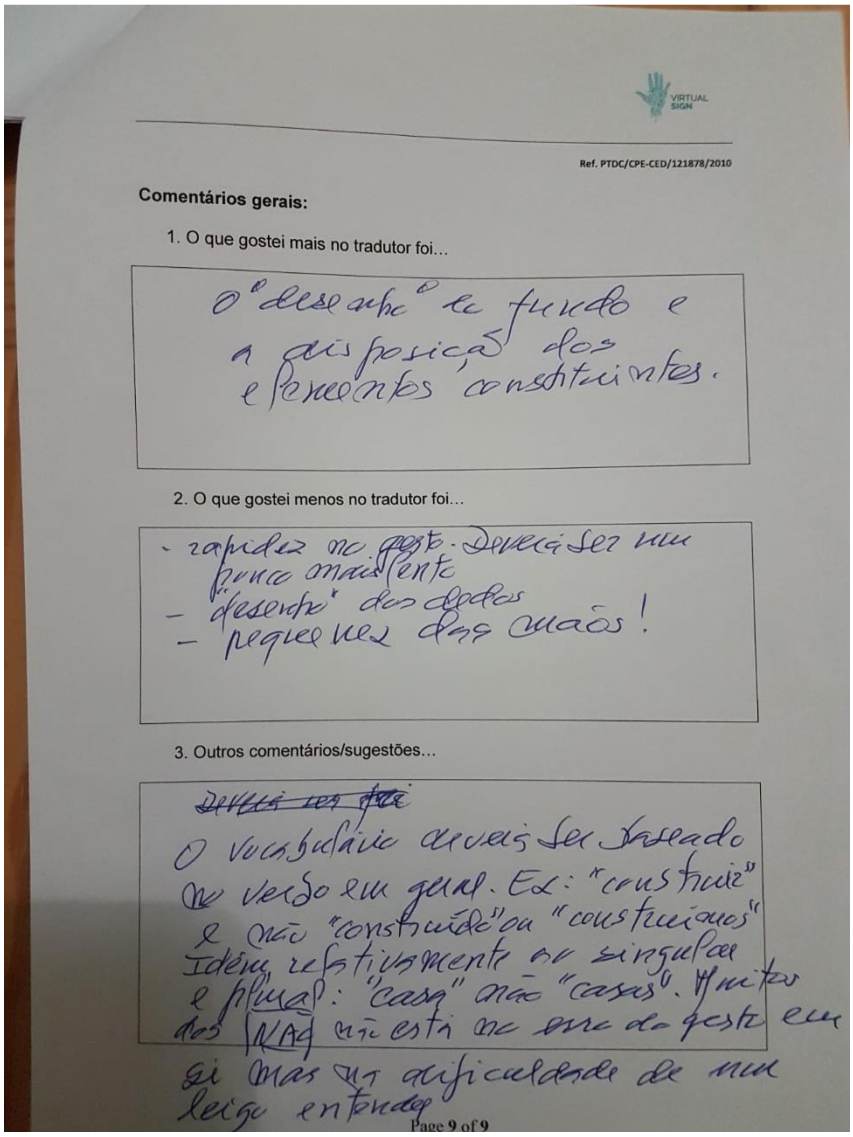
inventario	S	N	
realizar	S	N	? ?
educação	S	N	ligado ao ensino
divertido	S	N	
pessoa	S	N	
ponto	S	N	se a pessoa estiver (pontuação)
direção	S	N	não consegue aceder
corpo	S	N	
componente	S	N	
ligar	S	N	
vencer	S	N	depende do contexto
apontar	S	N	
detetar	S	N	
fazer	S	N	
usar	S	N	
resultado	S	N	depende do contexto
carateristicas	S	N	
caracteristicas	S	N	
guarda	S	N	acção/depende do context
incluir	S	N	
inicia	S	N	
abrir porta	S	N	
adquirir	S	N	
aumentar	S	N	depende do contexto
barreiras linguisticas	S	N	
caminho	S	N	"via" mais comunitária
cor	S	N	
dizer	S	N	
domingo	S	N	está de modo errado
encontrar	S	N	
entrar	S	N	
escuro	S	N	
flor	S	N	
força	S	N	
fraco	S	N	
frio	S	N	
fugir	S	N	
fundo	S	N	
gelo	S	N	
gordo	S	N	



VIRTUAL SIGN

Ref. PTDC/CPE-CED/121878/2010

gostar	S	N	
gramatica	S	N	
gravida	S	N	
interessar	S	N	
novo	S	N	mas não relativo a pessoas
em que posso ajudar	S	N	
a sua identificação	S	N	
até logo	S	N	
mão	S	N	
gesto	S	N	
jogos	S	N	no geral a pessoas
didatico	S	N	
serios	S	N	
constituido	S	N	não consegui nada
modulos	S	N	
adocao	S	N	
configuracao	S	N	se das coisas
mao	S	N	
termino	S	N	
posicao	S	N	se localização
mudanca	S	N	mas não de lugar
animacoes	S	N	
construidas	S	N	sem acessorios
cenarios	S	N	
avaliacao	S	N	
multiplos	S	N	
integracao	S	N	é incluir
interesse	S	N	
escola	S	N	
nada	S	N	
tinha	S	N	
desde	S	N	relativo a espaço + tempo
estudo	S	N	
refletiu-se	S	N	é reflexão
começo	S	N	
aulas	S	N	
eu	S	N	
havia	S	N	
reparado	S	N	não consegui nada
obrigado	S	N	



Ref. PTDC/CPE-CED/121878/2010

Comentários gerais:

1. O que gostei mais no tradutor foi...

O "desenho" de fundo e a disposição dos elementos constituintes.

2. O que gostei menos no tradutor foi...

- rapidez no gesto. Deveria ser um pouco mais lento
- "desenho" dos dedos
- pequenez das mãos!

3. Outros comentários/sugestões...

~~Difícil ver tudo~~

O vocabulário deveria ser baseado na versão em geral. Ex: "construir" e não "construído" ou "construíamos" Idem, referindo-se ao singular e plural: "casa" não "casas". Muitos dos NAQ não estão na área do gesto em si mas na dificuldade de um leigo entender

Page 9 of 9

Annex F – VirtualSign Project Dissemination Table

Ano	Publicação	Url
05/2013	Paula Escudeiro et al. Virtual Sign Translator", Proceedings of the International Conference on Computer, Networks and Communication Engineering (ICCNCE 2013), Beijing, China	http://www.atlantis-press.com/publications/aisr/iccnce-13/
08/2013	Paula Escudeiro et al. "Automatic Sign Language Translator Model", Proceedings of the 2013: Annual International Conference on Advances Technology in Telecommunication, Broadcasting, and Satellite (TelSaTech), Jakarta (Indonesia)	http://telsatech.org/?page_id=90 http://www.aspbs.com/science.htm#202
03/2014	Nuno Escudeiro et al. "Aplicação de um Tradutor Virtual de Língua Gestual a Jogos Sérios", IV Jornadas de Informática da Universidade Aberta, Porto, Portugal	http://repositorioaberto.uab.pt/handle/10400.2/3206
04/2014	Paula Escudeiro et al. "Virtual Sign Supporting Inclusive Learning and Equity", Computers and Technology in Modern Education, 5th International Conference on Education and Educational Technologies (EET '14), Kuala Lumpur, Malaysia'	http://www.wseas.us/e-library/conferences/2014/Malaysia/EET/EET-02.pdf http://www.wseas.us/e-library/conferences/2014/Malaysia/EET/EET-00.pdf
04/2014	Paula Escudeiro et al. "VirtualSign Game - Learning Sign Language", 5th International Conference on Education and Educational Technologies (EET '14), Kuala Lumpur, Malaysia'	http://www.wseas.us/e-library/conferences/2014/Malaysia/EET/EET-04.pdf http://www.wseas.us/e-library/conferences/2014/Malaysia/EET/EET-00.pdf
04/2014	Paula Escudeiro et al. "Serious Game for Sign Language", International conference on Intelligent Systems, Data Mining and Information Technology (ICIDIT '2014), Bangkok, Thailand	http://iieng.org/siteadmin/upload/1400E0414080.pdf http://www.iieng.org/proceedingspdf.php?id=17
07/2014	"ISEP cria ferramenta para traduzir língua gestual em aulas", Article published in the Viva! with the purpose of disclose the VirtualSign project.	http://www.viva-porto.pt/Em-Destaque/isep-cria-ferramenta-para-traduzir-lingua-gestual-em-aulas.html
07/2014	"Criada ferramenta para traduzir língua gestual em aulas", Article published in the TVI 24 Online with the purpose of disclose the VirtualSign project.	http://www.tvi24.iol.pt/503/tecnologia/lingua-gestual-tradutor-investigadores-porto-ferramenta-tvi24/1565585-4069.html
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas - SAPO Saúde", Article published in the Sapo Online with the purpose of disclose the VirtualSign project.	http://saude.sapo.pt/noticias/saude-em-familia/investigadores-do-porto-criam-ferramenta-para-traduzir-lingua-gestual-em-aulas.html

07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the RCM Pharma with the purpose of disclose the VirtualSign project.	http://www.rcmpharma.com/actualidade/id/23-07-14/investigadores-do-porto-criam-ferramenta-para-traduzir-lingua-gestual-em-aul
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Rádio Nova Online with the purpose of disclose the VirtualSign project.	http://www.radionova.fm/noticias/ler/19291
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Público Online with the purpose of disclose the VirtualSign project.	http://www.publico.pt/sociedade/noticia/investigadores-do-porto-criam-ferramenta-para-traduzir-lingua-gestual-em-aulas-1663921
07/2014	"Tradução bilateral de língua gestual", Article published in the Primeiro de Janeiro Online with the purpose of disclose the VirtualSign project.	http://www.oprimeirodejaneiro.pt/opj/diarias.asp?idioma=item_lingua1&cfg=0&item=14803
07/2014	"Investigadores do ISEP criam ferramenta para traduzir língua gestual em aulas", Article published in the Porto24 Online with the purpose of disclose the VirtualSign project.	http://www.porto24.pt/inteligencia/investigadores-do-isep-criam-ferramenta-para-traduzir-lingua-gestual-em-aulas/
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Porto Canal Online with the purpose of disclose the VirtualSign project.	http://portocanal.sapo.pt/noticia/32732/
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Pontos de Vista Online with the purpose of disclose the VirtualSign project.	http://www.pontosdevista.com.pt/index.php?option=com_content&view=article&id=12126:investigadores-do-porto-criam-ferramenta-para-traduzir-lingua-gestual-em-aulas&catid=37:ciencias&Itemid=91
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Médico Jornal Online with the purpose of disclose the VirtualSign project.	http://www.jornalmedico.pt/?p=9284
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Diário Digital Online with the purpose of disclose the VirtualSign project.	http://diariodigital.sapo.pt/news.asp?id_news=720105
07/2014	"Portugal cria tecnologia que traduz língua gestual", Article published in the Boas Notícias Online with the purpose of disclose the VirtualSign project.	http://boasnoticias.pt/noticias_Portugal-cria-tecnologia-que-traduz-l%C3%ADngua-gestual_20383.html?page=0
07/2014	"Investigadores do Porto criam ferramenta para traduzir língua gestual em aulas", Article published in the Amarense Online with the purpose of disclose the VirtualSign project.	http://www.oamarense.com/noticia.php?id=4455
07/2014	"Virtual Sign permite tradução bilateral da língua gestual", Article published in the Jornal de Notícias with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=03264c91-ec1e-417a-b7f0-05a40437dd43&analises=1

07/2014	"Porto cria tradutor de língua gestual", Article published in the Diário de Notícias with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=57900a53-c8e3-4875-bd48-ee498d7dd815&analises=1
07/2014	"Assim comunicamos melhor.", Article published in the Blog Atenta Inquietude with the purpose of disclose the VirtualSign project.	http://atentainquietude.blogspot.pt/2014/07/assim-comunicamos-melhor-e-bom.html
07/2014	"ISEP cria tradutor de língua gestual em aulas", Article published in the Metro Portugal with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=f9fd53d7-dff2-4e64-9de6-546fb8cc2d34&analises=1
07/2014	"Tradução nas salas de aula", Article published in the Destak with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=4cc225be-32af-447f-80be-821455983134&analises=1
07/2014	"Projeto do Porto traduz linguagem gestual para texto", Interview granted to the newspaper Diário de Noticias with the purpose of disclose the VirtualSign project.	http://193.136.60.223/virtualsign/docs/Newspaper_dn_27-07-2014.pdf
07/2014	"Projeto que pretende revolucionar a comunicação com os surdos", Interview granted to the radio Antena 1 with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=3ae05d69-a140-46c9-a8f8-bea9b1137c27&userId=18440d84-ada1-4565-97ee-ad56316c8df0
08/2014	“ISEP criou uma ferramenta para apoiar a comunicação com os deficientes auditivos”. Article published in the Noenigma with the purpose of disclose the VirtualSign project.	http://www.noenigma.com/2014/08/isep-criou-uma-ferramenta-para-apoiar.html
08/2014	"Tradutor para surdos traduz palavras e textos para língua gestual", Interview granted to the public TV channel RTP (Portuguese Radio and Television) with the purpose of disclose the VirtualSign project.	http://www.rtp.pt/noticias/index.php?article=759240&tm=8&layout=122&visual=61
09/2014	Paula Escudeiro et al. "Serious Game on Sign Language", XV International Conference on Human Computer Interaction, INTERACCION 2014	http://delivery.acm.org/10.1145/2670000/2662350/a97_Escudeiro.pdf?ip=193.136.60.71&id=2662350&acc=ACTIVE%20SERVICE&key=2E5699D25B4FE09E%2E6A9D8E1D9CC021F6%2E4D4702B0C3E38B35%2E4D4702B0C3E38B35&CFID=695688266&CFTOKEN=19082378&__acm__=1437661601_8376ebc44f3c47a48731d5dfd4cde6f3 http://interaccion2014.ull.es/images/Program.pdf
11/2014	"VIRTUALSIGN - O tradutor de língua gestual", Article of disclosure of the project, integrated in the number 19 of ISEP.BI (Newsletter of the Institute of Engineering of Porto)	http://193.136.60.223/virtualsign/docs/ISEP.BI.pdf
11/2014	Marcelo Norberto et al. "VirtualSign Translator in Serious Games", Third International Conference on Virtual and Networked Organizations Emergent Technologies and Tools, Póvoa de Varzim, Portugal	http://2100projects.org/conferences/vinorg14/Documents/Programme_BS14_ViNOrg14.pdf http://2100projects.org/conferences/vinorg14/proceedings.htm

12/2014	Paula Escudeiro et al. "Educational Contents for the Hearing Impaired Community: Using Games to Promote Equity and Inclusion", 16th International Conference on Information Integration and Web-based Applications & Services (iiWAS2014), Hanoi, Vietnam.	http://dl.acm.org/citation.cfm?id=2684200&picked=prox
12/2014	Paula Escudeiro et al. "Educational Contents for the Hearing Impaired Community: Using Games to Promote Equity and Inclusion", 16th International Conference on Information Integration and Web-based Applications & Services (iiWAS2014), Hanoi, Vietnam.	http://repositorioaberto.uab.pt/bitstream/10400.2/3653/1/Preprint_TALE2014.pdf http://www.tale-conference.org/tale2014/proceedings.php
02/2015	José Bidarra et al. "Tradutor para surdos traduz palavras e textos para língua gestual", Interview granted to the public TV channel RTP2 (Portuguese Radio and Television) with the purpose of disclose the VirtualSign project.	http://www.rtp.pt/play/p676/universidade-aberta
02/2015	Paula Escudeiro et al. "Award cerimony for the Inclusion and Digital Literacy prize", Demonstration of VIRTUALSIGN project to the ceremony participants.	http://www.ticsociedade.pt/premiovencedores
03/2015	"Tradutor para surdos traduz palavras e textos para língua gestual", Interview granted to the public TV channel SIC (Sociedade independente de comunicação) with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=f702d4fa-fefb-42fd-8057-0f948985cc72&userId=18440d84-ada1-4565-97ee-ad56316c8df0
03/2015	"Tradutor para surdos traduz palavras e textos para língua gestual", Interview granted to the public TV channel SIC Noticias (Sociedade independente de comunicação) with the purpose of disclose the VirtualSign project.	http://sicnoticias.sapo.pt/pais/2015-03-14-Ferramenta-criada-no-Porto-ganhou-premio-com-tradutor-de-lingia-gestual
03/2015	"Tradutor para surdos traduz palavras e textos para língua gestual", Presentation of the text module in CAMUL master's degree class.	
03/2015	"ISEP cria ferramenta para traduzir língua gestual em aulas", Article published in the Diário dos Açores with the purpose of disclose the VirtualSign project.	http://www.pt.cision.com/cp2013/ClippingDetails.aspx?id=29bb1706-cdc9-449b-afd4-bdcbc8f0b2c6&analises=1
03/2015	José Bidarra et al. " A Kinect Game in the VirtualSign Project: Training and Learning with Gestures", 7th International Conference on Digital Arts Creating Digital e-Motions. Óbidos, Portugal.	http://www.researchgate.net/publication/273773186_A_Kinect_Game_in_the_VirtualSign_Project_Training_and_Learning_with_Gestures https://dl.dropboxusercontent.com/u/4360015/Proceedings_ARTECH2015.pdf
04/2015	Paula Escudeiro et al. "VirtualSign",Presentation of the VirtualSign project in the Acesso.02 Educação e Inclusão conference.	https://www.ipp.pt/personnel/News.aspx?id=7772&back=%2Fpersonnel%2FNews_Tag.aspx%3Ftag%3Dacesso.02
05/2015	Marcelo Norberto et al. "Real Time Bidirectional Translator of Portuguese Sign Language", 11th	http://www.webist.org/Program/2015/WEBIST_paperList.htm

	International Conference on Web Information Systems and Technologies.	
05/2015	Paula Escudeiro et al. "VirtualSign", Presentation of the project VirtualSign in the Convenção Multidisciplinar de Educação	http://www.cme-gondomar.pt/detalhenoticias.php?nID=2#.Va0CI_lVhBc
06/2015	Marcelo Norberto and Jorge Lopes et al. "VirtualSign", Presentation of the project VirtualSign to a group of teachers from Kazakhstan	
06/2015	Marcelo Norberto and Jorge Lopes et al. "VirtualSign", Demonstration of the project VirtualSign in 3ª Jornadas da Engenharia do ISEP	http://jornadasinternacionais.aeisep.pt/#section-tecno
06/2015	Marcelo Norberto et al. "Virtual Sign – A Real Time Bidirectional Translator of Portuguese Sign Language", The 6th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-exclusion	http://www.dsai.ws/2015/?page_id=381#st22 http://www.journals.elsevier.com/procedia-computer-science/
06/2015	Marcelo Norberto and Jorge Lopes et al. "VirtualSign", Presentation of the project VirtualSign to another group of teachers from Kazakhstan	
07/2015	José Bidarra et al. "VirtualSign Translator", 7th International Conference on Education and New Learning Technologies, Barcelona, Spain	https://repositorioaberto.uab.pt/bitstream/10400.2/2893/1/Virtual%20Sign%20Translator.pdf http://library.iated.org/publications/EDULEARN15
	CAAI	
07/2015	ICMET	http://www.icmet.ac.cn/2015%20Bali%20conference%20program.pdf
09/2015	Paula Escudeiro et al. "VirtualSign in Serious Games", 5th EAI International Conference on Serious Games, Interaction and Simulation, SGAMES 2015	http://sgamesconf.org/2015/show/home