Serious Game for Sign Language

Paula Escudeiro, Nuno Escudeiro, Rosa Reis, F.P. Maciel Barbosa, José Bidarra, Ana Bela Baltasar,
Pedro Rodrigues, Jorge Lopes, Marcelo Norberto, and Vaz De Carvalho

Abstract—The work presented in this paper consists in the development of a game to make the process of learning sign language enjoyable and interactive. In order to do this, a game was created in which the player controls a character and interacts with various objects and non-player characters with the aim of collecting several gestures from the Portuguese Sign Language. These gestures can then be represented by the character. This allows the user to visualize and learn or train the various existing gestures. To raise the interactivity and to make the game more interesting and motivating, several checkpoints were placed along the level. This will provide the players a chance to test the knowledge they have acquired so far on the checkpoints by using Kinect. A High Scores system was also created as well as a history to ensure that the game is a continuous motivating process as well as a learning process.

Keywords—Educational game, Gesture language, Sign Language Game development, Portuguese Sign Language.

I. Introduction

THE project consists of a didactic game about Portuguese sign language, where the player can enjoy the game while learning gestures simultaneously[1].

Kinect has also been integrated into the game [2] in order to make it more interactive and appealing.

The main objective of this project is to facilitate the learning of the Portuguese sign language and to improve the dexterity of those who already know it, making learning a pleasant experience.

The deaf community in Portugal is around 100 000 individuals and yet the digital content available for this community is still rather low. With this project, not only we are promoting the knowledge for this restricted community, but we are also encouraging other people to learn and become able to better understand this community [3]..

The game is played in first person view. The player controls a character on the map. Each map represents a level and each level has several scattered objects through the map for the

Paula Escudeiro, Nuno Escudeiro, Rosa Reis, Ana Bela Baltasar, Pedro Rodrigues, Jorge Lopes, Marcelo Norberto, and Vaz De Carvalho are with the Departamento de Engenharia Informática, Instituto Superior de Engenharia do Porto, Porto, Portugal.

F.P. Maciel Barbosa, is with Departamento de Engenharia Electrotécnica, Faculdade de Engenharia da Universidade do Porto, INESC TEC Porto, Portugal.

José Bidarra is with Departamento de Ciências Exactas e Tecnologia, Universidade Aberta, Lisboa Portugal.

(E-mails: pmo@isep.ipp.pt;{ nfe & rmr}@isep.ipp.pt; fmb@fe.up.pt; bidarra@univ.ab.pt; anabela.baltazar@ctilg.pt; pmpro@isep.ipp.pt; 1090603@isep.ipp.pt; 1100633@isep.ipp.pt; cmc@isep.ipp.pt).

player to interact with. All objects collected by the player will be stored in his inventory and can be accessed at any point during the game. Most of these gestures and objects can be used through the inventory; the character will then perform the gesture so that the user can visualize how it is done.

To progress in the game you must collect all the gestures scattered around the level [4]. The faster the player manages to collect all the gestures the higher his score will be.

The player could choose to play using Kinect but previously he must have first obtained all the gestures in the level and perform themselves.

It's possible to find some projects related to this theme/area but none of them implies automatic bidirectional translation process as this game does.

Some of the related work being carried out by institutions and organizations are:

A. CopyCat

The game CopyCat is the most similar project in comparison to this one. It 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.

B. ProDeaf

ProDeaf is a Software that does the translation of Portuguese text or voice to Brazilian gesture language.

The objective of the ProDeaf is to make the communication between mute and deaf people easier, making the digital content of companies accessible in Brazilian gesture language.

C. Beijing University

In Beijing a project which allows the recognition of gestures in gesture language through the Kinect was developed.

D. Faceshift

This software analyzes the facial expressions, the orientation of the user's head and eyes. The information analyzed by the application is used to animate virtual characters that can be used in videos or games.

II. Functional and Non-Functional Requirements

In this session the functional requirements of the application will be described. Functional requirements represent the features available to the user.

The functional requirements were identified in the early process of the development of the project, and include the following:

• Start Menu where the player can choose the type of game he wants (with or without Kinect) consult the options or exit;

• Menu options where you can change the graphics quality, volume, save or load the game and see the table of high scores;

The game requirements within the levels are:

• Handling and control of the character;
• Interact with NPCs (Non-Player Characters);
• Consult the inventory and use the items in it;
• Interaction with map objects;
• Access to the above options menu

Non-Functional Requirements

Regarding non-functional requirements, our work focused on the following:

➢ Usability

Usability is the ease of use of the application and its accessibility to the user [6].

With this in mind this project should be fairly intuitive, allowing easy adaptation and learning.

The user interfaces must be simple so anyone can use them easily.

The character controls must be simple, and throughout the game there should be explanations of what needs to be done.

The accessibility is guaranteed because it is only necessary to run an executable file in order to play while Kinect is not in use.

While playing using the Kinect two applications are required as well as the necessary drivers and assuring that the Kinect is properly plugged in.

➢ Performance

The gaming performance is always a factor of the utmost importance, because the response time from the game to the user is always immediate, any delay or decrease of the number of frames per second can affect the gameplay making the game frustrating rather than fun.

To maintain the performance this game was tested to never run less than 60 frames per second on an optimal computer. The ideal frame rate for a game must be around 40 frames per second. The essential functions must be constantly executed and the code must be optimized to avoid the waste of resources.

Besides the code all the factors that constrain the performance of the game should be taken into account, such as textures, bumps, number of vertices of the 3D models among others.

Based on all these factors the performance of the game is assured.

III. GAME ARCHITECTURE

For this project there were two applications developed, the game in Unity 3D and the interface that connects the VirtualSign to Unity. The interface was developed in Microsoft Visual Studio.

The project is divided into layers, given its high degree of complexity.

At the top level there is the interface. All the functionalities of the project can be accessed through this layer by the user. This layer is responsible for forwarding the actions of the user to the next layers.

On the lower level there are three layers. The sockets layer is responsible for linking the Unity game application to the Kinect that is why this layer is below the interface in order to provide the layer above with the player input. Another layer of the lower level is the game engine; this layer is responsible for the execution of the game itself, representing the functions of Unity. Finally, there is the business layer, which is where the game functions are available to the player.
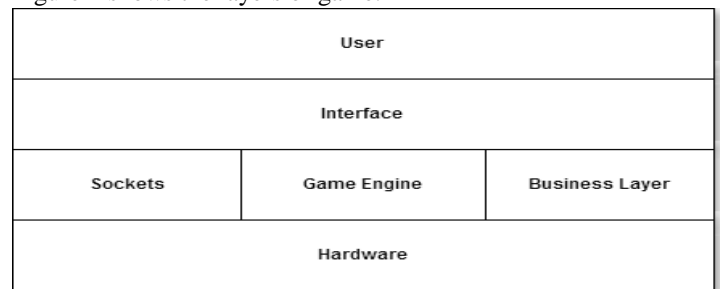
Figure 1 shows the layers of game.

User		
Interface		
Sockets	Game Engine	Business Layer
Hardware		

Fig. 1 Layers of game

➢ Development

For the development of this project it was decided to start with the implementation of the basic functionality and then proceed to the animation of the avatar, which was later replaced by an animated avatar now available at GILT.

In the development of this project a draft was first developed implementing the basic functionalities, as shown in figure 2. After that the player avatar was animated, however it was then replaced by the one provided by GILT.

After having a basic scenario created the development phase of the scripts started. The character motion scripts used were provided by Unity.
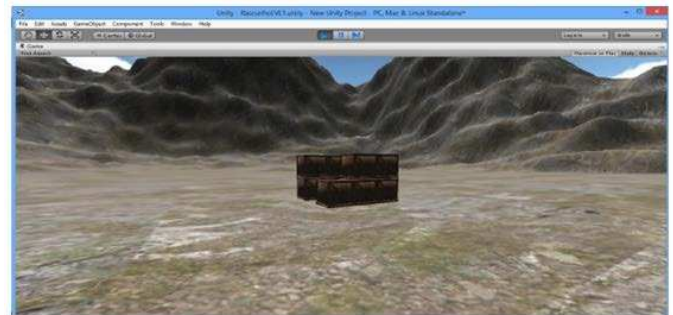

Fig. 2 The basic functionalities of the game

IV. GAMEPLAY

The first script developed was the inventory script. The inventory stores the items acquired by the player and provides access to them at any time.

Then scripts to interact with the objects were created. These scripts were optimized to be reused for multiple objects without having to change the code [6].

Having been established some objects on the map with the script to be added the inventory proceeded to the creation of this graphical interface.

The map has objects and those objects contain scripts on them, the scripts the objects contain allow them to be added to the inventory of the player. Creation of the graphical interface was then started.

The inventory consists of forty-two spaces that are empty upon initialization.

With the inventory set up and ready to receive the objects that the user can acquire, the handling of collisions with these objects was created in order to detect when the user is within a reasonable distance to perform the interaction.

A. Score

Players' score are incremented during the game as they acquire new gestures.

The shorter the time it takes between the acquisition of two objects, the greater the score. If the delay is less than one minute, one hundred points are acquired, if it is between one and two minutes, fifty points are acquired, if the delay exceeds two minutes, twenty-five points will be obtained regardless of time spent. Figure 3 shows the score of game.
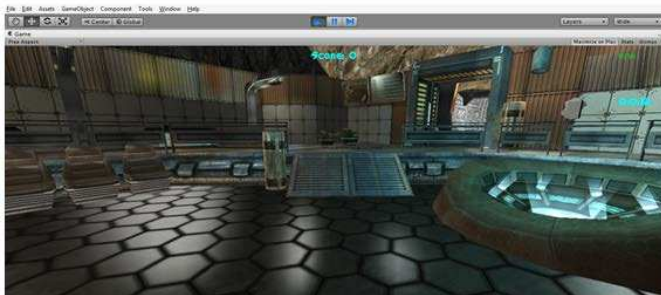

Fig. 3 Score of the game

B. Connect to Kinect

After finishing the structure of the game, the application to connect to Kinect was started.

Since this application would be an adaptation of the VirtualSign project, it was necessary to be developed in C # on the Microsoft Visual Studio environment due to compatibility reasons.

The VirtualSign application detects and translates the gestures that the user makes, saving the one with higher probability of success. With this in mind we needed access to this part application code in order to be able to send the necessary content to Unity. The connection method used was the socket as Unity supports .NET Framework. For this connection the use of an API that makes the connection in Unity and Development was needed.

An application where relevant text would be sent was also developed.

For this application the official Unity Sockets example was

adapted, and a window was created where the user could see the connection status. This window runs on the thread so it does not directly affect the performance of the recognition application.

To make the connection, the IP address and port are needed. This information is sent to the API and this will make the connection which will return a message acknowledging the success or failure of the operation. This API is entitled SharpConnect.dll. This DLL file was slightly modified in order to work properly based on the functionalities that were developed.

After the connection has been established, the information from Kinect is received and analyzed. This then confirms if it is the information we were expecting. If it is not, then the user will be notified of what gesture he did and which gesture was expected.

It will only be possible to proceed when the user manages to perform the requested gesture correctly.

V. CONCLUSION

The development of a game is always a complex task to do and many adversities were faced along the way. A lot of effort and time were needed to get outdo this challenges, and a fair amount of knowledge was acquired during this process.

The selection of this target population arises due to the growing number of students with special needs who complete the elementary and high school and come to higher education. This situation demands for new means that allow these individuals to have easy access to educational digital content. In order to motivate them towards the learning process we have created a game that combines the sign language learning process with the pleasant feeling of playing a digital game.

REFERENCES

[1] Reyes, I. 2005 "Comunicar a través del silencio :las posibilidades de la lengua de signos
[2] D. Catue, Programming with the Kinect for Windows: Software Development Kit, Microsoft, 2012
[3] Herrero-Blanco, A., Salazar-Garcia, V. 2005 "Non-verbal predicability and copula support rule in Spanish Sign Language". In: de Groot, Casper / Hengeveld, Kees (eds): Morphosyntactic.
[4] Digital Game-Based Learning (McGraw-Hill, 2001) by Marc Prensky. Chapter 1.
[5] Holm, Ivar (2006). Ideas and Beliefs in Architecture and Industrial design: How attitudes, orientations, and underlying assumptions shape the built environment. Oslo School of Architecture and Design. ISBN 82-547-0174-1.
[6] Salen, Katie; Zimmerman, Eric (2004). Rules of Play: Game Design Fundamentals. Cambridge, Massachusetts: The MIT Press. ISBN 978-0-262-24045-1. "Game play is the formalized interaction that occurs when players follow the rules of a game and experience its system though play.