

UNIVERSITY OF TARTU
FACULTY OF SOCIAL SCIENCES

NARVA COLLEGE
INFORMATION TECHNOLOGY SYSTEMS DEVELOPMENT

Ivan Švaiger

DEVELOPMENT OF 3D MODELS, ANIMATIONS, STORYLINE AND DIA-
LOGUE SYSTEM FOR INTERACTIVE ENGLISH LEARNING MOBILE
GAME FOR RUSSIAN SPEAKING USERS

Diploma Thesis

Supervisors: Assistant Andre Säask

Nadežda Furs-Nižnikova

NARVA 2018

Olen koostanud töö iseseisvalt. Kõik töö koostamisel kasutatud teiste autorite tööd, põhimõttelised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

.....

Töö autori allkiri ja kuupäev

CONTENTS

INTRODUCTION	5
The problem.....	5
The solution.....	5
Aim	6
Tasks	7
Outline	7
1. CHOSEN TOOLS AND TECHNOLOGIES.....	8
1.1. Blender	8
1.1.1. Blender description	8
1.1.2. Blender and Adobe Illustrator comparison.....	9
1.1.3. Blender, Maya LT and 3ds Max comparison	9
1.1.4. Why Blender	10
1.2. Tiled.....	10
1.3. JSON	10
1.4. Python.....	11
1.5. Android Studio	11
2. MODELLING OF CHARACTERS AND THEIR IMPLEMENTATION IN THE GAME	
12	
2.1. 3D Modelling and description of the tasks.....	12
2.1.1. Three-dimensional modelling and 3D models	12
2.1.2. Description of the tasks	12
2.2. Characters Modelling	13
2.2.1. Main character	13
2.2.2. Main character modelling process.....	14
2.2.3. Process of modelling	14
2.2.4. Model Painting	16
2.2.5. Adding fur particles.....	18

2.2.6. Main Character Rigging and Animation	18
2.2.7. Other characters.....	21
2.2.8. Modelling maps and tiles	22
2.2.9. Camera settings	24
2.3. Implementation of characters and tiles in the game.....	25
2.3.1. Rendering (Characters/Map).....	26
2.3.2. Atlases	28
2.3.3. Tiled Map	29
3. LANGUAGE ACQUISITION, STORYLINE AND DIALOGUE SYSTEM.....	30
3.1. Language and language acquisition	30
3.2. The characteristics of a «good» language learner	30
3.3. The possibilities of learning the language using modern technologies and games.....	30
3.4. The principle of learning English in our game.....	32
3.5. Teaching materials and dictionaries are referenced	33
3.7. Storyline and how it is built in this game.....	34
3.8. JSON and implementation of dialogues and storyline in the game	37
CONCLUSION.....	39
TERMS AND ABBREVIATIONS	40
REFERENCES	42
APPENDICES	46
RESÜMEE.....	47
LICENCE.....	48

INTRODUCTION

The problem

The modern world is developing rapidly, including the educational system, specifically the learning of foreign languages for primary school children. As classical educational methods are increasingly losing their effectiveness with new generations of students, there are hopes that modern technologies can help us keep the education effectiveness in this area high. In addition, with modern technologies we can not only achieve high results in educational effectiveness, but also keep the learning process fun, which is good in keeping students motivation high, especially in the primary school.

Today many schools still predominantly use the old ways of the education not implementing all the opportunities modern technologies give us. People, especially children, like playing (also video games), but the education system is still not using all the possibilities which playing provides for enhancing motivation. Those few available language-learning games are very similar and mostly only offering quizzes, which does not use the full potential, which modern technology provides.

The solution

The solution is “Learn by playing”, which means that players may observe role model characters or interact directly with them. They may even play the role of the character, make behavioral choices, and virtually experience the rewards and punishments that occur as a result of the character’s actions. (Lieberman 1997)

Thus, a video game can help to ignite interest in the subject (studying a language in this case) and enhance motivation. A language-learning game would be able to increase the vocabulary for a particular subject and show its real usage.

Aim

The aim of this diploma is to develop 3D models, animations, storyline and dialogue system back-end for a language learning game, which would increase a student's interest and motivation through the following features:

- Players may explore game map and interact with game characters;
- Players can complete game levels, which adds competitive and emotional components to the learning process;
- Players can memorize words by using them in game dialogues;
- Players can get confirmations from the game about their progress.

Because of the extensive volume of the described above aim, the work on this project was divided between two authors. The author of this thesis worked on following parts of the game:

- Modelling of main and other game characters;
- Modelling maps and tiles;
- Developing of the storyline;
- Developing of the back-end part of the dialogue system.

The second author's parts were:

- Developing a player's ability to complete levels;
- Developing a player's ability to talk to the game characters;
- Developing an in-game dictionary;
- Developing a player's ability to move freely around the game world map;
- Developing a player's ability to follow the game storyline.

Tasks

To achieve the aim of his parts of the work the author has to complete the following tasks:

1. Find appropriate technologies for development of the game.
2. Design all characters and maps of the game.
3. Create all characters and maps animations of the game.
4. Render all characters and maps of the game.
5. Implement all characters and maps of the game.
6. Create an interactive storyline system of the game.
7. Develop dialogues and the assessment system.

Outline

The work consists from an introduction, two chapters, a conclusion, a list of terms and abbreviations, a list of references and three appendices. The first chapter speaks about modelling process and, in particular, modelling main and other characters, as well as maps and tiles. It covers all stages of 3D models development. The second chapter speaks about language learning, mobile learning games and about the storyline and the dialogue system. In the appendix 1, one can find all Blender files with 3D models and the Python rendering script. In the appendix 2, one can find the JSON format file. In the appendix 3, one can find the result of this thesis, the English learning game.

1. CHOSEN TOOLS AND TECHNOLOGIES

In order to build a full-fledged solution, we need a bundle of technologies and the author chose the following:

1.1. Blender

As Blender is the author's main tool for this thesis, the author goes in great lengths to explain the choice of this tool.

1.1.1. Blender description

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline—modelling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation. (Blender... 2018)

For this work, the author used Blender¹ version 2.79. As the Blender is: being actively developed by hundreds of people from all around the world. These include animators, artists, VFX experts, hobbyists, scientists, and much more. (Blender... n.d.).

This means that some things (Mesh Modelling, Animation, Add-ons etc.) may not work or work correctly on newer or older versions of this software. A more specific change-log version of the Blender 2.79 can be found on their website².

Furthermore for the correct operation of this application, a sufficiently large computing power is needed. The simulation process, characters and animation described below was performed according to «Recommended hardware» of the software producer's website³.

It is worth considering the fact that the scope of 3D technologies is quite extensive and there are many different applications, such as the Adobe Illustrator and other 3D editors (Maya LT, 3ds Max) to perform this work, the comparison with them will be described below.

¹ Blender is available at the following link: www.blender.org/download/

² wiki.blender.org/index.php/Dev:Ref/Release_Notes/2.79

³ www.blender.org/download/requirements

1.1.2. Blender and Adobe Illustrator comparison

Based on more than two thousand reviews and a comparison of the two technologies⁴, we can draw the following conclusions:

- Adobe Illustrator is in the community a more popular game than a Blender.
- Adobe Illustrator has a higher average rating (9.6 vs. 8.6)
- Among the disadvantages of Adobe Illustrator can be identified paid model of distribution and monthly payments.
- Not quite suitable for character animation (which is necessary for this thesis), although 2D animation is still possible.

1.1.3. Blender, Maya LT and 3ds Max comparison

Definitely the choice already from the popular and above-mentioned 3D editors seems quite complex and depends more on the preferences of a particular user or company engaged in 3D modelling.

Based on article «3ds Max, Maya LT or Blender - Which 3D Software Should I Choose for Asset Creation?»⁵, we can draw the following conclusions:

- The game Maya LT and 3ds Max offer a few paid options and only limited versions for students. When Blender offers an open source and complete version with free distribution.
- 3ds Max is not a multi-platform game and only available on the Windows operation system.

In the rest Blender, Maya LT and 3ds Max are quite similar and offer a nearly identical set of services for 3D design.

⁴ <https://www.g2crowd.com/compare/adobe-illustrator-vs-blender>

⁵ <https://www.pluralsight.com/blog/film-games/3ds-max-maya-lt-blender-3d-software-choose-asset-creation>

1.1.4. Why Blender

Often, production pipelines require several powerful packages: one for modelling, another for texturing and others for animation and compositing. Fortunately, Blender is kind of a Swiss Army knife program is that it can do many of these tasks. (Flavell 2010:1)

Among the many advantages of Blender, the most important for this work are:

- Open source and multi-platform game.
- The possibility of self-study using a large number of courses and materials available on the Internet.
- Wide spectrum on 3d tools cover: Modelling, Sculpting, Node-base shading, Rigging and animation, Python scripting, Uv Painting and projection, a wide file format compatibility import. (Schiller 2016)

1.2. Tiled

Tiled is a 2D level editor that helps you develop the content of your game. Its primary feature is to edit tile maps of various forms, but it also supports free image placement as well as powerful ways to annotate your level with extra information used by the game. Tiled focuses on general flexibility while trying to stay intuitive. (Tiled... 2017)

Used by the author for further implement maps and tiles into the game.

1.3. JSON

JavaScript Object Notation (JSON) is a text format for the serialization of structured data. It is derived from the object literals of JavaScript. JSON's design goals were for it to be minimal, portable, textual, and a subset of JavaScript. (Crockford 2006)

JSON text format is necessary for further implementation of the dialogue in the LibGDX project.

1.4. Python

Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. It provides constructs that enable clear programming on both small and large scales. (Python... 2018)

Used by the author as a scripting language for rendering in a Blender and also for post rendering optimization.

1.5. Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. (Android... 2018)

Android Studio is not directly used by the author, but all necessary tests are performed using this IDE for the final product implementation (character models, maps, tiles, dialogue system, etc.).

2. MODELLING OF CHARACTERS AND THEIR IMPLEMENTATION IN THE GAME

2.1. 3D Modelling and description of the tasks

2.1.1. Three-dimensional modelling and 3D models

Three-dimensional (3D) modelling of an object can be seen as the complete process that starts from data acquisition and ends with a 3D virtual model visually interactive on a computer. Often 3D modelling is meant only as the process of converting a measured point cloud into a triangulated network (“mesh”) or textured surface, while it should describe a more complete and general process of object construction. (Remondino; El-Hakim 2006)

3D models, in general, represent a 3D object using a collection of points in 3D space, connected by various geometric entities such as triangles, lines, curved surfaces, etc. (Zolotarjov 2009)

Blender, like many other computer-modelling games, allows 3D models to be created either by working with points, called vertices; edges; surfaces or curves. A combination of all of these methods is usually required in order to achieve high-quality images. (Zolotarjov 2009)

In addition to defining the structure of a 3-dimensional model, it may have a texture. A texture is a static (or moving) image wrapped around the model. Blender provides a comprehensive texturing solution as well as different lighting methods. (Zolotarjov 2009)

2.1.2. Description of the tasks

The author used Blender application to perform the following tasks:

- Design of unique characters and map tiles.
- Good topology identification.
- Modelling of unique characters and map tiles.
- Unwrapping of models.

- Using of materials and colours for 3D models painting.
- Animation of the main character.
- Rendering.

2.2. Characters Modelling

2.2.1. Main character

Since the main goal of the final product is provide an opportunity to learn a language in an interactive mobile game, the topic of the first level of the game are animals and for the main character was chosen a dog (puppy). In addition, all the rest 3D model characters (NPC) presented in this chapter are animals, both wild and domestic. All the 3D models presented in this work are the original artwork of the author of the thesis.



Figure 1. The main character (Source: author)

2.2.2. Main character modelling process

The process of 3D modelling the main character can be conditionally divided into 3 stages: process of modelling, model painting and adding fur particles.

Taking into account that the work is based on some practical component and this means that all the theoretical aspects of 3D modelling, an overview of Blender's interface, setup, hotkeys, modifiers, etc. will be omitted and will be shown only directly the development of the 3D models, as the analysis of Blender is not paramount for this thesis work. A short introduction to Blender was given in the chapter 1.1.

2.2.3. Process of modelling

Sculpting with Blender

3D sculpting is a method of manipulating the shape of 3D objects in a direct and intuitive way that is loosely analogous to the way a sculptor works with clay in the real world. It is an excellent way to model detailed organic forms, and many artists find it to be a much more natural way to model than the direct manipulation of **vertices** and **polygons** used in more traditional 3D mesh modelling. For this reason, 3D sculpting is rapidly growing in popularity in the professional computer graphics (CG) modelling world (Mullen 2009:31).

Using the services of Blender and the Sculpting mode as the dominant for modelling the 3D model of the main character, it was decided to divide the modelling of the entire model into 3 important parts: modelling the head, modelling the body and combining the head and body model into one single model of the dog (puppy). Head modelling is a separate modelling of the shape of the head using 3D sculpting method from a common cube in sizes 1x1x1, as well as the modelling of the nose, ears, eyes, and mouth with the tongue. At this point, the eyes are a separate object (independent of the rest of the head) and will be completed later using various techniques. Body modelling is a separate modelling of the dog body, back and front paws and a tail. Combined model of the character is a combined model of head and body, which in fact is practically the final model and will be used in the further stages. For symmetry purposes of the whole model a mirror modifier is used.



Figure 2. *The main character head (Source: author)*

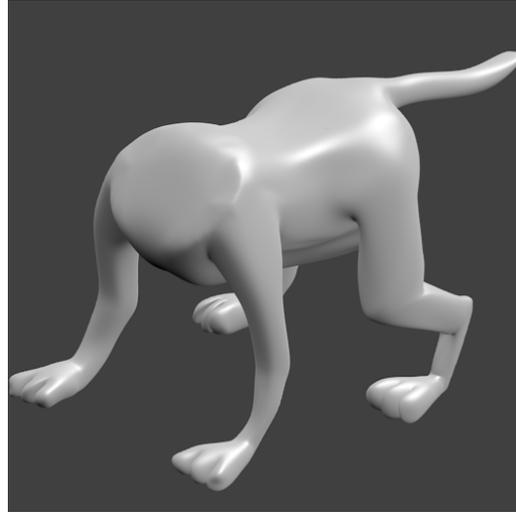


Figure 3. *The main character body (Source: author)*

If the character or object is symmetrical, the modeller can use mirroring tools to create a perfectly symmetrical model. When a **mirror modifier** is active, every change done to one side of the model will be automatically done on both sides of the model's middle line. This greatly speeds up the modelling process. Once the symmetrical version is done, the mirror modifier can be collapsed onto the model so both sides of the model can be edited individually if there are parts that need to be asymmetrical on the model. Any symmetrical parts are though best kept under the influence of the mirror modifier to ensure all future steps like texturing and rigging can be finished quickly and painlessly. (Heikkila 2017)



Figure 4. The main character combined (Source: author)

It should be noted that this 3D model is not the final version and can be slightly changed in the subsequent stages for the convenience of rigging, animation or any other purposes. In addition, the model in the **Figure 4** may differ in shape and size with the original model due to other camera settings, viewing angle, light and rendering parameters, and other parameters that may also vary.

After the end of the modelling stage, we should proceed to the next stage, Model Painting, shown below.

2.2.4. Model Painting

At this stage 3D model is painted using UV/Image Editor, which is the simplest way of painting a complex 3D model. The whole 3D model is divided into separate 2D objects (head, paws, body, tail, etc.), which are painted separately. It is not difficult to see which parts of 2D objects belong to 3D model. In the UV/Image Editor objects painted in brown or white have exactly the same colour on the 3D model.

Imagine a paper 3D model of an object, e.g. a sphere, which is to be laid flat on a table. Each of the 3D coordinates of the sphere can be mapped to the 2D coordinate on the flat piece of paper. Blender provides another view of the vertices (coordinates) in the UV/Image Editor. You can select and edit these 2D vertices just like in the 3D Editor window. The purpose of this unwrapping of the coordinates is just to map these coordinates to images/Figures, so that the 3D image can have a realistic looking surface with textures derived from these images. (Blender... 2018)



Figure 5. Painted model (Source: author)

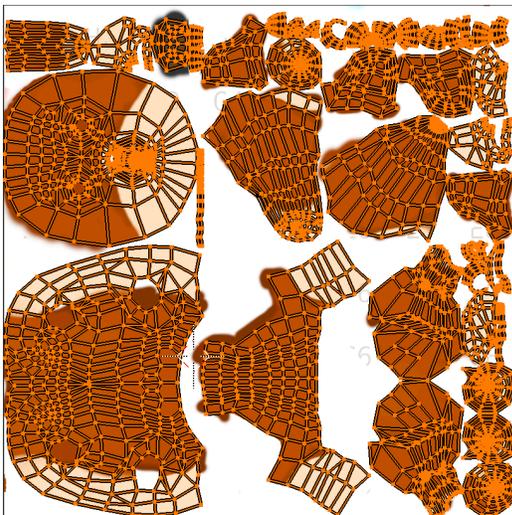


Figure 6. Unwrapped model (Source: author)

2.2.5. Adding fur particles

To create fur on the main character the author used Particle Edit Mode. In general, fur is a set of individual particles (hairs), the adjustment of which allows to achieve the effect of fur. Flexible system of Blender allows you to work with almost individual objects (hairs), this makes it possible to use any system of particles necessary for the user.

Using Particle Edit Mode you can edit the keyed points (keyframes) and paths of Baked Hair, Particle, Cloth, and Soft Body simulations. (Particle... 2018)

One can see how the fur looks like on the main character in **Figure 1**. This image is final in terms of rendering. In the chapter of the implementation of this character in the game will only change the resolution and other minor fixes.

At this stage, the main process of modelling of the main character is finished and starts the stage of rigging and animation, described in the next chapter.

2.2.6. Main Character Rigging and Animation

Process of rigging

To create the walking animation of the main character the first stage should be to use the process of rigging. This is almost the only way for the successful animation of 3D model in Blender.

The process of rigging a character means setting up an armature and its associated constraints. Skinning refers to connecting a mesh to an armature so that manipulating the armature results in the desired deformations of the mesh. The mesh should follow the movement of the armature on the appropriate way. (Mullen 2011)

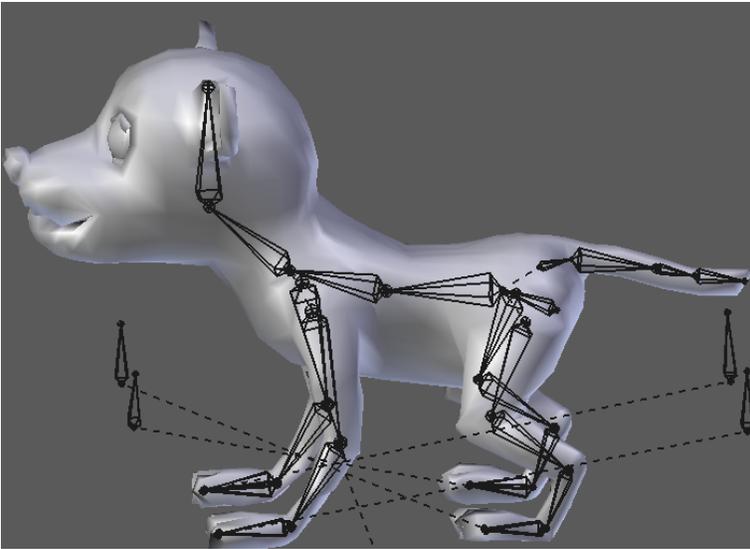


Figure 7. Main character armature system (Source: author)

Figure 7 shows how the armature is positioned as a result of the rigging process within the main character. Given arrangement has much in common with the location of bones inside a real dog. Then animation occurs when each part of 3D model corresponds to the armature, their joint use control the movement.

Animation

Animation is making an object move or change shape over time. Objects can be animated in many ways:

- Moving as a whole object
- Changing their position, orientation or size in time;
- Deforming them
- Animating their vertices or control points;
- Inherited animation

Causing the object to move based on the movement of another object (e.g. its parent, hook, armature, etc.). (Animation... 2018)

Animation is typically achieved with the use of keyframes. (Animation... 2018)

In order to save resources and time, only a few different frames (states) of the 3D model were made, which are shown in the **Figure 8** below. Since the full walking animation takes 40 frames, 10 states are allocated each 5 frames, for example, frame № 1, 6, 11, etc., and frame № 41 repeats the first frame – some kind of a cycle occurs. All other frames Blender generates automatically; automatically determining the only possible state changes between frames. But this does not mean that one cannot do and customize each frame yourself.

The **Figure 8** shows the basic frames, which create the main character walking animation. Each frame is unique and represents a certain pose, corresponding to the movement of the real prototype (real dog).

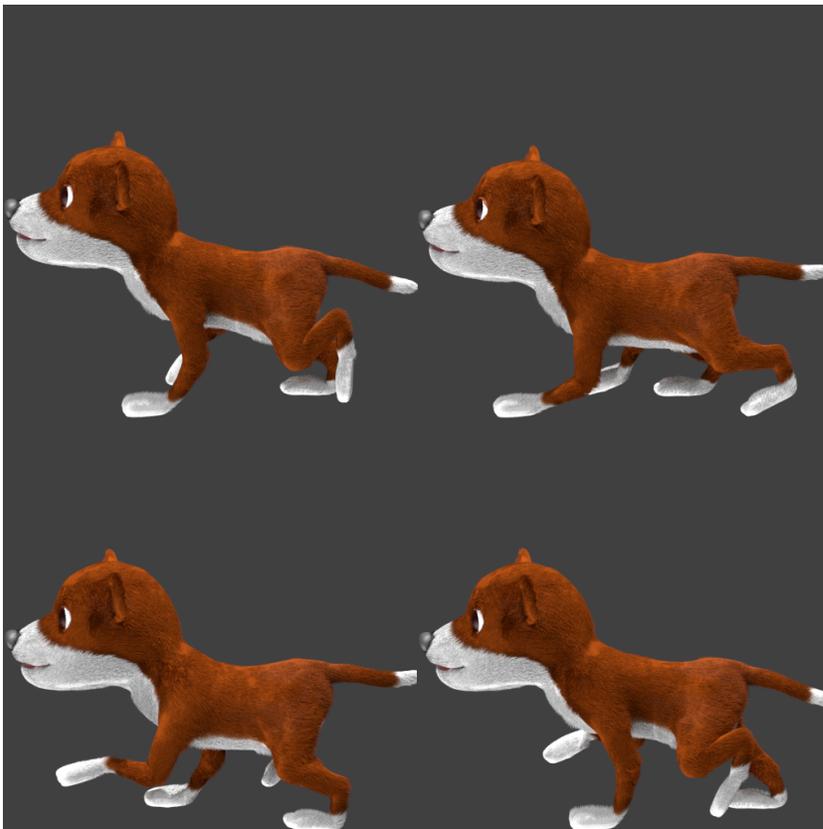


Figure 8. Animation of the main character (Source: author)

In addition to the walking animation, the main character has a front animation, when the character does not move anywhere, but simply stands. This animation reflects a rather popular for real dogs animation, when dog search for something, sniffing the ground. This animation has

61 manually created frames, which is also like walking animation looped around and repeated. More details can be seen as a small video available by the link⁶.

At this stage, the development of the main character is completed and ready to implementation in the game. We can move on to the following chapters that describe other characters and a tile map.

2.2.7. Other characters

The other characters were modelled using the stages described in the chapters 2.2.1 - 2.2.6 of modelling the main character, including using Sculpting Mode and Mirror Modifier, as well as Unwrapping and Painting. It is worth noting that these characters have no System of Particles (fur, hair, etc.), rigging and animation. And in general they are less realistic than the model of the main character.

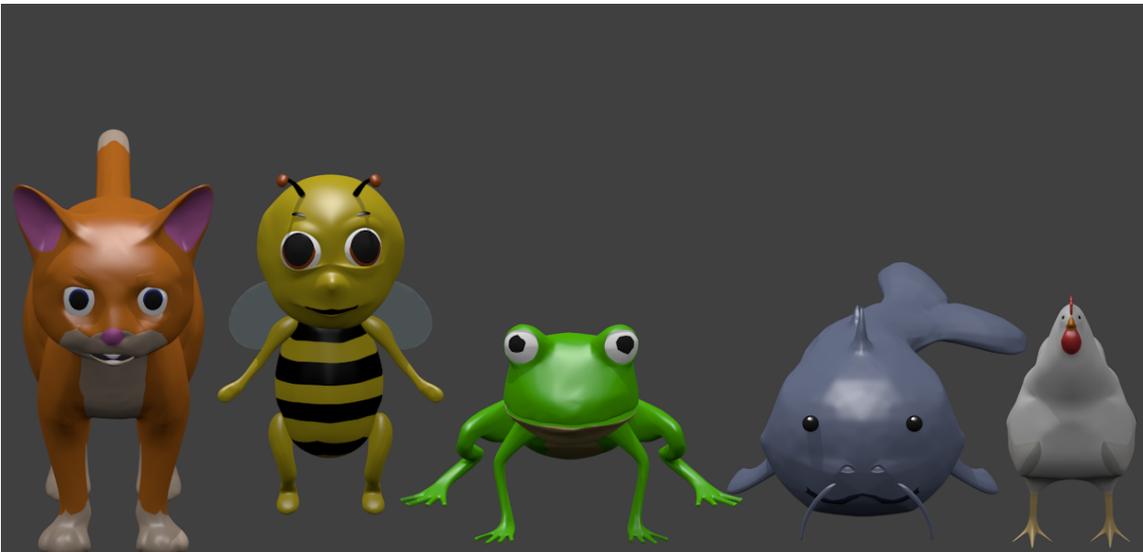


Figure 9. Other characters (Source: author)

3D models of other characters located in this Figure from right to left - Cat, Bee, Frog, Fish and Hen. They are made in the one style and also as the main character (puppy) belong to the animal kingdom, representing the wild or domestic animals. This is due to the topic of the interactive language learning game. Each 3D model of these (other) characters is NPC and can

⁶ <https://www.youtube.com/watch?v=TBxb6XkE1VA>

be located and rendered separately from the others. All the characters, their appearance and color are part of the interactive learning described in the chapter «Language Acquisition, storyline and dialogue system». This means that these parameters can be used to construct various tasks and questions related to the appearance and color of the characters. Therefore, using low-poly style we have the most understandable for the average user both in terms of design and 3D models, and in terms of color.

2.2.8. Modelling maps and tiles

Modelling of maps and tiles is similar to modelling the other 3D models described in this work (the main character and the other characters) with the difference that all tiles are cubes/parallelepipeds deployed in a special way in the isometric projection. The whole map or a separate tile unfolded in the isometric projection looks like a rhombus, if you neglect the Z-axis of coordinates and imagine that this is the usual plane.

How the whole card looks like



Figure 10. The whole map (Source: author)

The whole map has a size of 24x24 tiles. The whole map is one large tile of 4776x8640 pixels, divided into a set (576) of smaller tiles with a size of 198x360 pixels for more convenient im-

plementation in the final application described in the following chapters. In reality, however, the number of tiles is somewhat larger, as necessary to take into account the water animation, each tile animation of water movement which represents the 50 frames.

How the separate tile looks like

In this map there are three types of tiles:

1. The first of them represents only a flat surface. This type of tiles can be seen in **Figure 11** in the middle - this tile is a simple texture of the soil.
2. The next category of tiles is kind of texture, but in addition there is also a whole object placed on its surface. Example - **Figure 11** tile on the left (soil texture, a little bit grass texture and also a whole object - post pointer.)
3. The last of them also represent some kind of surface texture, and only a part of a large 3d object divided into separate tiles. In **Figure 11** on the tile on the right, it is seen that there is a surface, flowers; and also only a part of a large house placed on one tile. The rest of the house is already placed on other tiles, and only by gathering the necessary tiles you can build a whole house.

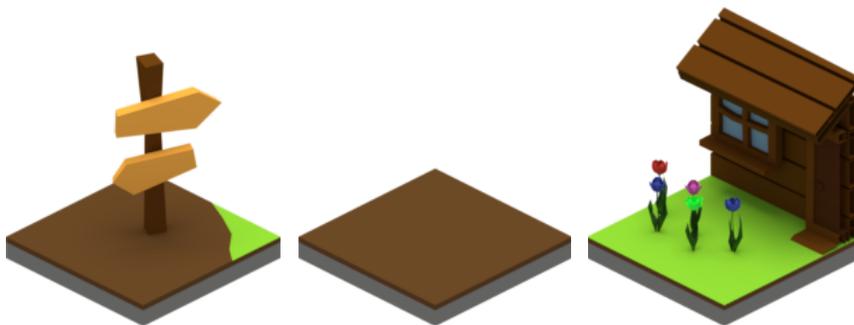


Figure 11. Three types of tiles (Source: author)

Animation of water movement

In this project, there is a completely different type of tile, which is responsible for water movement animation. The creation of animation of the movement of water in general is similar to the creating of animation of the movement of animals or any other. And represents several states of the object (frames of water), their changes in time and some semblance of cycle.



Figure 12. Animation of water (Source: author)

On static graphics difficult to display an animation of the movement of water. Animation can be seen by carefully looking at the edges of objects and on the hillocks of the waves, which are generally changing, and create motion animations. **Figure 12** shows frames of 1,11,21 animations of the whole water surface - all the animation of which takes 252 frames. It is worth noting that the whole water surface moves at the same time, which can be seen in **Figure 10**. The river divides the whole map into 2 parts.

On this stage, all the sufficient graphic part (3D model of the main character, other characters, map and tiles) is done and remains only the correct rendering, camera settings and implementation in the game, which is shown in the following chapters.

2.2.9. Camera settings

First, the camera settings specifically for this learning mobile game assumes that the camera position in the isometric projection. More precise camera settings for working in the isometric projection in Blender can be found at Isometric Tutorial website. (Bellanger 2016)

Isometric projection is a method for visually representing three-dimensional objects in two dimensions in technical and engineering drawings. (Isometric... 2018)

The correct camera settings for isometric rendering of characters (for example, the main character) can be seen in **Figure 6**, which is one of the 5 necessary positions for the render. The correct camera settings for isometric rendering of maps and tiles can be seen in **Figure 10** and **Figure 11**. A more detailed analysis will be made in the chapter 1.6 «Implementation of characters and tiles in the game».

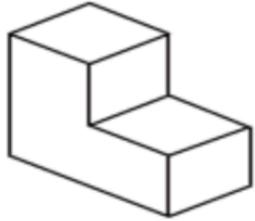
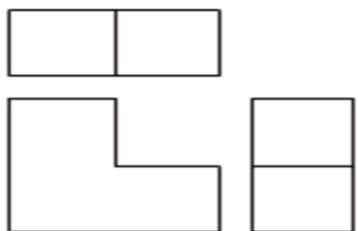
Isometric projection	Orthographic projection
With an isometric projection, the object's angles are distorted in order to show all three dimensions of the object.	The main advantage of using orthographic projection over an isometric is that you are able to see the true shape of an object without any distortion.
Right angles become more or less than 90 degrees, squares become parallelograms, etc.	Right angles will be 90 degrees and squares will be squares.
 <p data-bbox="389 1312 673 1344">Isometric View</p>	 <p data-bbox="925 1312 1266 1344">Orthographic Views</p>

Figure 13. Visual comparison of isometric and orthographic projections⁷

2.3. Implementation of characters and tiles in the game

The implementation of the characters (main, others) and tiles in the final application built on the LibGDX 2D game engine can not be directly transferred from the Blender project. Although the export of 3D models for 3D games on LibGDX is possible in almost any format

⁷ http://www.highereducation.org/HES01/Module_4/module_4_theme_5.html

- .blend, .fbx and many others⁸. But since we are dealing with LibGDX 2D game engine and 2D objects rendered in isometric projection, we should find another way to do the implementation. And the only possible rendering is to render 3D models with special camera settings in the 2D image. The main difficulty of this process is the large amount of the final rendered images, which should be properly grouped.

2.3.1. Rendering (Characters/Map)

Map rendering

The map modeled for this mobile game consists of 576 individual tiles. And in order to not render 576 tiles of the map manually one by one, moving the camera; requires making automation for this process. For these purposes it is best to use a script in the programming language Python.

Blender has a typical Python environment so tutorials on how to write Python scripts will work running the scripts in Blender too. Blender provides the bpy module to the Python interpreter. This module can be imported in a script and gives access to Blender data, classes, and functions. Scripts that deal with Blender data will need to import this module. (Python... n.d.)

⁸ A more detailed overview of this rendering can be found at: <http://www.gamefromscratch.com/post/2014/01/19/3D-models-and-animation-from-Blender-to-LibGDX.aspx>

```

import bpy

objs = []
for obj in bpy.data.objects:
    if obj.name.startswith("Landscape"):
        objs.append(obj)

# loop true all the meshes, hide all but active, and copy active to 0,0,0 and render
for curObj in objs:
    prevLoc = curObj.location.copy()
    curObj.hide_render = False

    for otherObj in objs:
        if curObj != otherObj:
            otherObj.hide_render = True

    curObj.location = (0,0,0)
    bpy.data.scenes['Scene'].render.filepath = "/Localization on disk/Landscape" curObj.name ".png"
    bpy.ops.render.render(write_still = True)

    curObj.hide_render = True
    curObj.location = prevLoc

```

Figure 14. An example of Python rendering script (Source: author)

What does this script do

This script renders each tile in turn, having the word «Landscape» in its name and creates files with the same name in the form of separate .png Figures with a resolution of 198x360 pixels. Since all tiles are initially of the same size, then placing camera correctly on the first tile (or any other), you can render the image simply by moving the camera to a previously known distance (the width of the tile).

The original version of this script and a more detailed description can be found by the link⁹. But it slightly differs from the version of the script offered by the author of the thesis.

Characters rendering

Is the same rendering method as the rendering of tiles, only need to change the camera settings, while this rendering assumes rendering the whole character's model at once, rather than splitting it into parts. All the characters are rendered at a resolution of 256x256 pixels. And the main condition is to render the main character in all five directions at once, as the main character in final application can move forward, backward, left, right and this is necessary for the

⁹ <http://www.gamefromscratch.com/category/Design.aspx?page=3>

animation of the movement, and there is still a static position of the main character. All other characters can be rendered in any one of 5 positions, because they represent a static object.

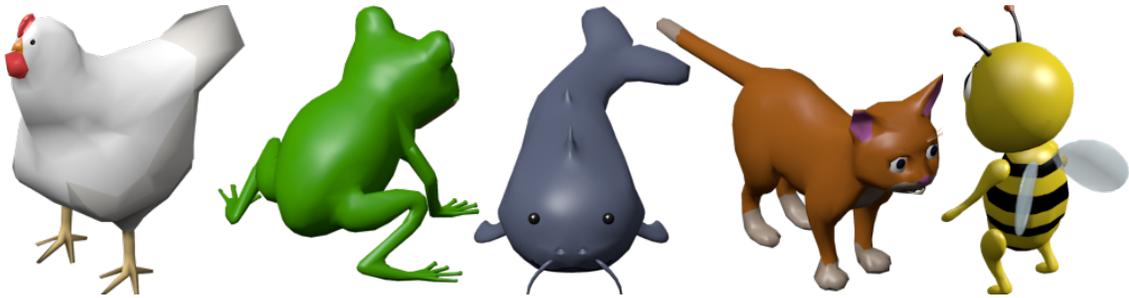


Figure 15. Five possible isometric positions for the characters rendering (Source: author)

2.3.2. Atlases

In real-time computer graphics, a texture atlas (also called a sprite sheet or an image sprite) is an image containing a collection of smaller images, usually packed together to reduce the atlas size. In an game where many small textures are used frequently, it is often more efficient to store the textures in a texture atlas which is treated as a single unit by the graphics hardware. (Texture... 2018)

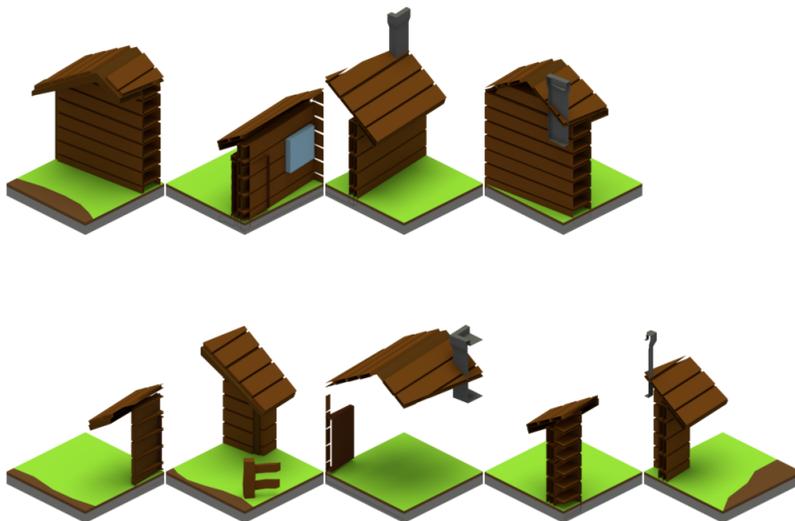


Figure 16. An example of an atlas of house textures. (Source: author)

2.3.3. Tiled Map

This atlas of the house shown in Figure 14 can be used to create a tile map in Tiled. Specifically for this atlas, you need to create a new empty tile map with a size of 3x3 and a single tile of 198x99 pixels. Then adding an atlas of the house to this map in Tiled and placing individual tiles in the correct order, you can get the image shown in **Figure 15**. Further this card from Tiled can already be directly integrated into the LibGDX gaming engine, because Tiled supports it natively.

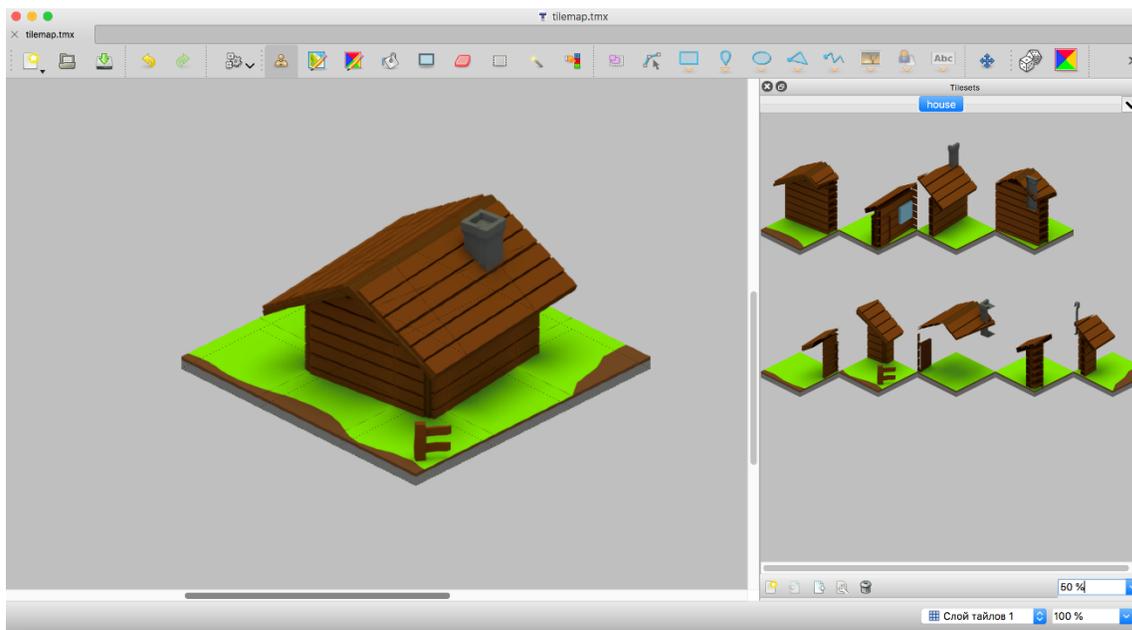


Figure 17. Map example in Tiled (Source: author)

At this point, the main part of the modelling and implementation is successfully completed and you can move on to the next chapter of the thesis, and specifically to the language acquisition, storyline and dialogue system.

3. LANGUAGE ACQUISITION, STORYLINE AND DIALOGUE SYSTEM

3.1. Language and language acquisition

Language is a complex, specialized skill, which develops in the child spontaneously, without conscious effort or formal instruction, is deployed without awareness of its underlying logic, is qualitatively the same in every individual, and is distinct from more general abilities to process information or behave intelligently. (Pinker 1994:18)

Language acquisition is very similar to the process children use in acquiring first and second languages. It requires meaningful interaction in the target language-- natural communication-- in which speakers are concerned not with the form of their utterances but with the messages, they are conveying and understanding. Error correction and explicit teaching of rules are not relevant to language acquisition. (Brown 1970)

3.2. The characteristics of a «good» language learner

In my opinion the most accurate characteristic of a good learner is described in the Matthew J. Davies work under the title «Are the characteristics of «good» language learner identifiable?: An examination of «good» language learner assumption in a young learner EFL classroom»¹⁰ On pages 4-6 presents the most complete information provided at different angles and points of view. Using this information about the «good» learner characteristics, was developed interactive language learning mobile game presented in this work.

3.3. The possibilities of learning the language using modern technologies and games

Another important direction in research that requires more attention is use and effect of computer technology on foreign language learning. As classroom tasks become more focused on real-world issues, texts, or events, and problem-solving- based tasks, technology introduces a new dimension to the teaching and learning process that incorporates the use of social media

¹⁰ http://seas3.elte.hu/coursematerial/RyanChristopher/Davies_goodlearner.pdf.

such as Facebook, Twitter, Skype, Voice Thread, and others. Digital media allows students to manipulate learning materials and language at their own pace and according to individual needs. (Kramer; Moeller; Catalano 2015:327-332)

A remarkable number of people are turning to their mobile devices to learn a foreign language. The global market for digital English language learning products, for example, reached \$1.8 billion in 2013. Revenues are projected to surge to over \$3.1 billion by 2018. (Adkins, 2008)

Today the use of mobile technologies for language learning purposes is a rapidly evolving topic and therefore will be considered in my opinion three most popular and best mobile language learning applications to date (2018 year) are:

- Duolingo. The app divides languages into different topics, such as clothing or business, but also into adverbs, pronouns, and other grammatical subjects. Duolingo is a great introduction, but it can feel like you are learning a random mix of information, often through repetition. The offline mode is limited, too, so you really need a Wi-Fi or data connection to make the most of Duolingo. (Hill 2018)
- HelloTalk. An app aimed to facilitate speaking practice and eliminate the potential stress of real time conversation. Learners can find native speakers and converse with them using a whatsapp-like chat with voice and text messages. Users can correct each other's messages with an in-built correction tool, which transforms the language exchanges into tiny tutoring sessions. (Krzeminska 2018)
- Busuu. The app takes you through learning individual words to simple dialogues and questions about the dialogues all of which include audio where you can listen to native pronunciation. The lessons are organized in topical themes where we learn skills and expressions connected to tasks. (Krzeminska 2018)

Advantages of software-as-teacher

One of the greatest advantages of software-as-teacher, as compared to human-as-teacher, is that software possesses the potential to record complex user input in a precise, reliable manner. While a teacher may not remember every error that a student generates, software, if developed

properly, could provide invaluable formative information that would otherwise be too substantial for a human to plausibly record. This ability for software to update automatically its functionality based on input received or data processed is known as adaptive learning. While growing in popularity, it is still a largely unexplored arena in mobile language learning games. (Heil; Wu; Lee 2016)

3.4. The principle of learning English in our game

Our language learning interactive mobile game offers a slightly different concept of learning a foreign language. Focusing on English for mainly Russian-speaking children of primary school age (grades 1-5) we propose instead of simple boring tests and answers to questions entertaining-interactive component.

Our game is a RPG (role-playing game) with the open world and quests, where the user can freely move around the map, communicate with the characters and perform various tasks. In the future, more in-depth study of the environment in the game can be available, some interactive functions - inventory, quest items, etc., but to date only a map, characters and quests connecting them are available.

The main principle is that the game can be divided into thematic levels and the characters of these levels can be the studied objects and also like their actions. The first level of game is a quest on the topic of animals. The main characters is the dog (puppy), also there are other animals in the game (Frog, Cat, Bee, Fish, Hen), which act as NPC (Non-Player Character) - they all described in the chapter 2.2.7. The main character can move around the map described in chapter 2.2.8. and participate in dialogues with other characters (NPCs) that have a static position on map. Moving steadily from one character to another, the learner uses and learns new words in an interactive-entertaining form. In this way, the systematic language acquisition takes place.

The reason of this decision

The reason of this decision is primarily concerned with the importance of visuals in Second Language Acquisition. The visual component is one of the most important for the study of a foreign language for schoolchildren and children.

Most of the language teachers seem to agree that the use of visuals can enhance language teaching. As they help teachers to bring the real world into the classroom, they make learning more meaningful and more exciting. (Brinton 2000)

Another important thing to note is that visuals may help in order to build mental models, and communicate relationships among content objects in a more efficient way than can words alone. (Clark; Lyons 2004)

3.5. Teaching materials and dictionaries are referenced

Teachers and specialists in the field of education especially for Russian-speaking children of primary school age, which will help students to study the material in informative and entertaining form, develop the materials on which we base.

Among them, author was chosen the dictionary – «Англо-русский словарик в картинках» (English-Russian dictionary in pictures)¹¹. It is a dictionary in pictures, which has about 1000 words and expressions grouped according to topics studied in grades 1-5.

The dictionary in our game is made up only from words on materials English-Russian dictionary in pictures. Therefore, the learner must have some preliminary knowledge in the language being studied, since the game is aimed at studying specific topics, rather than learning the language as a whole.

Also for the compilation of the storyline and dialogues, several useful English-language lessons for Russian-speaking children of primary school age on animals topic have been used:

¹¹ Available at link: <https://yadi.sk/i/M3XNwogTsxUyd>.

- «План – конспект открытого урока по английскому языку во 2 классе по теме ЖИВОТНЫЕ». ¹² This is a lesson plan on the topic of animals. Useful in terms of how to use new words in the study of language and some new topic; as well as what topics can be used. For example, the author of this lesson suggests to use such a characteristic of animals - the name of animal, what it is, its colour, what it does etc. This approach is almost completely borrowed in our game.
- Разработка урока английского языка для начальных классов «Животные»¹³. This work represents a summary of the training lesson on animals for Russian-speaking students in primary classes. Is useful in terms of implementation from simple words to meaningful suggestion and method to gradually complicate them.

3.7. Storyline and how it is built in this game

The storyline in the English learning mobile game can be divided into 3 parts: the author's words, the words of the main character and dialogue system.

The author's words are used to introduce and familiarize the learner with the games, as well as an explanation of all the rules of the game and the assessment system. The author's words, unlike all the others (the words of the main character, the dialogue system) are made purely in Russian.

The words of the main character are adapted, simple sentences in English, which introduce the learner to the main character.

Dialogue system is a **Branching Dialogue System**, each branch of the dialogue which initially consists of 4 answers. Each branch of the dialogue initially consists of 4 answer options. Choosing the correct answer on the first attempt the player moves on to the next question and so on until the very end. If the wrong answer option is selected, this wrong answer removed from the list and a new dialog box appears with 3 answer options. If, again, the wrong answer

¹² Available at link: <http://www.school36-tambov.ru/125.pdf>

¹³ <https://videouroki.net/razrabotki/razrabotka-uroka-angliyskogo-yazyka-dlya-nachalnykh-klassov-zhivotnye.html>

is selected from the three options, the remaining two remain. The last attempt to correctly answer the question in the dialogue comes from 2 options, and if the player chooses the wrong variant here, then the player will be shown the correct option.

In games where branching dialogue is the primary gameplay focus, the player's choices often affect the NPC's attitudes toward the player one way or another, with the player attempting to guess the "best" response in order to maximize NPC disposition. Game dialogue becomes more interactive when conversations can take different paths. The player reads dialogue and chooses their response from a limited set of choices available to them. Conversation typically moves forward such that the player cannot go back to previous topics or responses. (Ellison 2008)



Figure 18. Example of branching dialogue system in our game (Source: author)

Figure 18 shows a branch of a Branching Dialogue System in our game in the case if the player chose the wrong answer initially (Answer1) to the question (Question). The red answers in the red box are wrong (Answer1, Answer2, Answer4). Answer in the green box is correct (Answer3). Next we get a dialogue where the player selects between Answer2, Answer3 and Answer4. Answer1 is no more among possible answers. The Answer3 in any case leads the player to proceed to the next question. If was selected Answer2 or Answer4 (incorrect answers), the new branches are created in which the given answer (Answer2 or Answer4) is absent and there are already two choices. A choice of two possible options, one of which is correct can lead to two outcomes: the player's choice of the correct answer (Answer3) or the choice of the wrong answer. Answer 2 or Answer4. Choosing the right answer in this case leads to the player getting the minimum possible number of points (250) and moving on to the next question. Choosing the wrong answer leads to showing the correct answer to the player, getting 0 points for the answer to the question and moving on to the next question.

The difference between our approach and the classical one is that, in fact, the attitude of the NPC to the character does not change during the dialogues, but the player's final score changes-the number of points scored for the lesson directly depends on correct or incorrect answers. A more accurate description of the assessment system will be given below.

Possible ways to make assessment in games

In a typical digital game, as players interact with the environment, the values of different game-specific variables change. For instance, getting injured in a battle reduces health and finding a treasure or another object increases your inventory of goods. But now consider monitoring educationally-relevant variables at different levels of granularity in games. In addition to checking health status, players could check their current levels of systems thinking skill, creativity, and teamwork, where each of these competencies is further broken down into constituent knowledge and skill elements. If the estimated values of those competencies got too low, the player would likely feel compelled to take action to boost them. (Shute; Ke 2012)

Assessment system in our game

The assessment system given in this paper has no theoretical basis, but is done simply for an example. It can be replaced at any time with any other.

- For the answer to the question at the first attempt the learner receives a reward of 1000 points.
- For the answer to the question in the second attempt the learner receives a reward of 500 points.
- For the answer to the question from the third attempt, the learner receives a reward of 250 points.
- If all the answers were wrong then the result is 0 points.

Then all the results for all the dialogs are summed up and the learner is given the corresponding score. In fact, the result is such an assessment system depends only on the number of correct/incorrect answers.

3.8. JSON and implementation of dialogues and storyline in the game

Libgdx JSON implementation does not strictly follow the standard: it is a bit more forgiving. Firstly, value names do not need to be quoted. Officially, it should be "name": "David" rather than name: "David". Secondly, it allows C-style comments in case developers want to document complicated formats inline. (Marquez; Sanchez 2014)

There are a lot of editors for JSON, which simplify the work with the text and make the functionality quickly accessible. But you can also open the JSON as an ordinary text file, which is shown in Figure 18.

```

"2" : {
  "text" : "There are R F O G letters in my name. Please, do my name from letters.",
  "answers" : [
    {
      "score" : 0,
      "destinationID" : 3,
      "text" : "Your name is FORG."
    },
    {
      "score" : 0,
      "destinationID" : 4,
      "text" : "Your name is GOFR. "
    },
    {
      "score" : 1000,
      "destinationID" : 9,
      "text" : "Your name is FROG."
    },
    {
      "score" : 0,
      "destinationID" : 5,
      "text" : "Your name is R0FG."
    }
  ]
}

```

Figure 19. How the dialogue in JSON looks like (Source: author)

It is worth describing some of the main parameters:

1. The digit “2” in the first line indicates that this is the second dialog in the queue. The first dialogue (“1”) was devoted to acquaintance with the character (frog) and does not contain all the significant points from the point of view of using the JSON, so it can be missed.
2. “Text” in this case means either a question given to the main character or a response (answer) from the main character.
3. “destinationID” denotes which dialog in the queue the dialogue is leading to.
4. “score” is a the result for choosing a particular answer. In this case “score” 1000 means that the answer is correct. And “score” 0 means that the learner made a mistake.

Thus making other dialogs, changing or deleting some parameters; you can implement the dialogue in the JSON text format and directly in the LibGDX game engine and specifically in the interactive English learning mobile game.

CONCLUSION

The result of this thesis is the development of 3D models, animations, storyline and dialogue system back-end for the interactive English language learning mobile game based on the Android operating system. Another author developed the other parts of the game. The result is a working game with all the planned features implemented. The authors hope that this game will help to increase students' interest and motivation in studying English.

The author of this thesis completed the following parts of the game development:

- Describing all the main stages of the development of 3D character models – modelling, painting, adding particles and animation.
- Rendering and implementation of all 3D models in the game.
- Modelling of maps and dividing the map into tiles.
- Analysing the field of language learning, as well as giving a brief overview of currently available language-learning games.
- Based on this data, developing a storyline and dialogue system for the game.

As all 3D models presented in this work are the original artwork of the author of the thesis, only the main character – the puppy – has all the animations and visual effects available. The secondary characters currently only have static positioning and no particle system.

Prospects for further development are the development of new thematic levels and the possible expansion of the functionality as a whole, including adding animations and particle systems to all secondary game characters, more in-depth study of the environment in the game can be made available, some interactive functions – inventory, quest items, etc. The task of further development is to make the game more interactive and interesting for the learner, and learning the language more diverse and motivating.

TERMS AND ABBREVIATIONS

Edge - the outer or furthest point of something

Vertices - the point of two or more curves, lines or edges meet.

Rendering - is the automatic process of generating a photorealistic or non-photorealistic image from a 2D or 3D model. (Rendering...2018)

(Good) topology - is the layout of a model, how the vertices and edges are placed to create the mesh surface. Good topology is essential if you want fast framerates and good deformation. (Topology...2017)

Polygon - a flat shape with three or more straight sides.

3D coordinates - Three-dimensional space with x,y,z coordinates.

Keyframe - frame, which defines some position in which there are certain qualitative changes.

Cycle - a group of events that happen in a particular order, one following the other.

Rhombus - a flat shape that has four sides that are all of equal length.

LibGDX - is a free and open-source game-development game framework.

.png format - Portable Network Graphics is a raster graphics file format.

Resolution - the number of distinct pixels in each dimension.

Concept - a principle or idea.

Quest - Adventure game in which the player assumes the role of a protagonist in a interactive story. (Adventure...2018)

Assessment - the process of considering all the information about the situation or person and making a judgement.

Queue - to arrange tasks in order.

Gamification - the game of game-design elements and game principles in non-game contexts.
(Gamification...2018)

Tile - a flat, thin, usually square piece of baked clay, plastic, or other material used for covering floors, walls, or roofs.

Pixel - the smallest unit of an image on a television or computer screen.

REFERENCES

Zolotarjov, Anton 2009. Open source 3D Modelling suite Blender, in comparison to proprietary games. University of Tallinn. Available at http://www.cs.tlu.ee/instituut/opilaste_tood/seminari_ja_proseminari_tood/2009_kevad/anton_zolotarjov_seminaritoo.pdf, accessed April 28, 2018.

Flavell, Lance 2010. Open source 3D Modelling, Animation and Game Design. Apress. Available at https://books.google.ee/books?hl=ru&lr=&id=aHmlWV50mIAC&oi=fnd&pg=PP2&dq=3d+modelling+blender&ots=AwU2JQ0SIV&sig=WCsY-8lqkJCmehWywuz7DTU_jhg&redir_esc=y#v=onepage&q=3d%20modelling%20blender&f=false, accessed April 28, 2018.

Schiller, Pierre 2016. Blender 3D - A fresh perspective on 2016. LinkedIn. Available at <https://www.linkedin.com/pulse/blender-3d-fresh-perspective-2016-pierre-schiller>, accessed April 28, 2018.

Pinker, Steven 1994. Language Instinct. The New Science of Language and Mind. Penguin Books. Available at <http://cirpstudents.com/Research%20Library/assets/the-language-instinct.pdf>, accessed April 29, 2018.

Rodger, Brown 1970. Derivational complexity and order of acquisition in child speech, accessed April 29, 2018.

Kramer Moeller, Aleidine; Catalano, Theresa 2015:327-322. Foreign Language Teaching and Learning. University of Nebraska-Lincoln. Available at <https://digitalcommons.unl.edu/cgi/viewcontent.cgi?article=1195&context=teachlearnfacpub>, accessed April 29, 2018.

Mullen, Tony 2009:31. Mastering Blender. Wiley Publishing, Inc. Available at https://books.google.ee/books?hl=ru&lr=&id=Iighl2bcGS0C&oi=fnd&pg=PR17&dq=blender+sculpting&ots=VJv0J-uI3U&sig=h_nhqprNpZSvtUQD3Y_qYOSI_X4&redir_esc=y#v=onepage&q=blender%20sculpting&f=false, accessed April 30, 2018.

Heikkila, Elisabet 2017:5. A Guide to Building a 3D Game Character. Kajaanin Ammattikorkeakoulu University of Applied Sciences. Available at https://www.theseus.fi/bitstream/handle/10024/133905/Heikkila_Elisabet.pdf?sequence=1&isAllowed=y, accessed April 30, 2018.

Blender 3D: Noob to Pro/UV Map Basics, 2018. Available at https://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/UV_Map_Basics, accessed May 01, 2018.

Blender 2.79 Manual, 2018. Available at <https://docs.blender.org/manual/en/dev/physics/particles/mode.html>, accessed May 02, 2018.

Mullen, Tony 2011:121. Introducing Character Animation with Blender. Wiley Publishing, Inc. Available at https://books.google.ee/books?hl=ru&lr=&id=8TTOBgAAQBAJ&oi=fnd&pg=PA274&dq=rigging+blender&ots=RseWzeVnmJ&sig=eLmEnmfNlNdod8SNR4hPOBdGbSw&redir_esc=y#v=onepage&q&f=false, accessed May 02, 2018.

Texture atlas, 2018. *Wikipedia*. Available at https://en.wikipedia.org/wiki/Texture_atlas, accessed May 05, 2018.

Hill, Simon 2018. The best language-learning apps for Android and iOS. Digital Trends. Available at <https://www.digitaltrends.com/mobile/best-language-learning-apps/>, accessed May 05, 2018.

Adkins, Sam S. 2008:5. The US Market of Mobile Learning Products and Services: 2008-2013 Forecast and Analysis. Ambient Insight. Available at http://ambientinsight.com/Resources/Documents/AmbientInsight_2008-2013_US_MobileLearning_Forecast_Executive-Overview.pdf, accessed May 05, 2018.

Hell, Regina Catherine; Wu, Jason S.; Lee, Joey J., 2016. A review of mobile language learning games: trends, challenges and opportunities. Teacher College, Columbia University, USA. Available at <https://polipapers.upv.es/index.php/eurocall/article/view/6402/7213>, accessed May 05, 2018.

Crockford, Douglas 2006. The game/json Media Type for JavaScript Object Notation (JSON). Network Working Group. Available at <https://tools.ietf.org/html/rfc4627#ref-ECMA>, accessed May 06, 2018.

Brinton, Donna M. 2001. The use of Media in Language Teaching. Heinle and Heinle. Available at <https://docslide.us/documents/use-of-media-in-language-teachingbrinton2001.html>, accessed May 06, 2018.

Clark, Ruth C.; Lyons, Chopeta 2007. Graphics for learning: Proven guidelines for planning, designing, and evaluating visuals in training materials. Available at <https://onlinelibrary.wiley.com/doi/abs/10.1002/pfi.4140431011>, accessed May 06, 2018.

Ellison, Brent 2008. Defining Dialogue Systems. Gamasutra. Available at https://www.gamasutra.com/view/feature/132116/defining_dialogue_systems.php?page=1, accessed May 07, 2018.

Topology. *Polycount wiki*. Available at <http://wiki.polycount.com/wiki/Topology>, accessed May 07, 2018.

Rendering (computer graphics). *Wikipedia*. Available at [https://en.wikipedia.org/wiki/Rendering_\(computer_graphics\)](https://en.wikipedia.org/wiki/Rendering_(computer_graphics)), accessed May 07, 2018.

Adventure game. *Wikipedia*. Available at https://en.wikipedia.org/wiki/Adventure_game, accessed May 07, 2018.

Lieberman, Debra A. 1997. What Can We Learn From Playing Interactive Games? University of California. Available at http://ocw.metu.edu.tr/pluginfile.php/2381/mod_resource/content/0/ceit706/week7/Lieberman_What%20-makesplaying.pdf, accessed May 08, 2018.

Bellanger, Clint 2016. Isometric Tiles in Blender. Available at http://clintbellanger.net/articles/isometric_tiles/, accessed May 08, 2018.

Remondino, Fabio; El-Hakim, Sabry 2006. Image-based 3D Modelling: A Review. The Photogrammetric Record. Available at http://www.close-range.com/docs/Image-based_3D_Modeling_A_Review_2006_remondino_elhakim_phorec06%5B1%5D.pdf, accessed May 18, 2018.

Gamification. *Wikipedia*. Available at <https://en.wikipedia.org/wiki/Gamification>, accessed May 20, 2018.

Marquez, David Saltares; Sanchez, Alberto Cejas 2014. *Libgdx Cross-platform Game Development Cookbook*. Packt Publishing. Available at https://books.google.ee/books?id=opQe-BQAAQBAJ&printsec=frontcover&hl=ru&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false, accessed May 20, 2018.

Shute, Valerie J.; Ke, Fengfeng 2012. *Games, Learning, and Assessment*. Available at <https://pdfs.semanticscholar.org/0591/77829bc6c427054af4c28fa77e2d882b09c5.pdf>, accessed May 22, 2018.

Krzeminska, Marta 2018. 10 best language learning apps. *Lingualift*. Available at <https://www.lingualift.com/blog/best-language-learning-apps/>, accessed May 05, 2018.

APPENDICES

Appendix 1. Blender files and Python rendering script:

<https://drive.google.com/open?id=1R7XofhfHU-a9qrfGzrxf5PVXXY1L-Nnx>

Appendix 2. .JSON storyline and dialogue system file:

<https://drive.google.com/open?id=1iAqQlqQlW1oLzowZiXQ1iRjwwKpC5Y0e>

Appendix 3. Game

<https://github.com/vitalibassov>

RESÜMEE

Seoses sellega, et on põhjust arvata, et traditsiooniliste õpetamismeetodite efektiivsus ei ole sama kõrge uute põlvkondade puhul, nagu see oli varem, muutub aktuaalseks uute õpetamis- ja õppimismeetodite otsimine ja katsetamine. Üheks potentsiaalseks lootustandvaks selle probleemi lahendamise teeks on mänguelementide sissetoomine haridusse, eriti algkooli.

Käesoleva lõputöö eesmärk on anda oma panuse sellesse tähtsasse protsessi algkooli võõrkeeleõppe valdkonnas inglise keele õppimiseks mõeldud interaktiivse õppemängu loomise teel, mida saaks kasutada õpilaste huvi ja motivatsiooni suurendamiseks õppeprotsessis. Õppemäng on mõeldud peamiselt vene keelt kõnelevate lastele. Kasutades lähenemist «Learn by playing» ja kõige kaasaegsemad tehnoloogiad (Blender, Python, JSON jms), suutsime saavutada positiivseid tulemusi ja valmistada mobiilse rakenduse, milles õpilane saab mängulises interaktiivses vormis omandada vajalikke teadmisi. Kuna sellise rakenduse loomine oli väga mahukas töö, jagati see töö kahe autori vahel.

Käesolevas töös on toodud lühike ülevaade kõigist vajalikest töös kasutatud tehnoloogiatest ja väike sissejuhatus 3D-modelleerimisse. Autor kirjeldab kõikide mängus kasutatud mudelite loomist ja mängu peategelase animeerimist. Lisaks oli vaja autoril valmistada mängu jaoks keeleõppe materjale mängu dialoogide ja põhiloo jaoks ja ning teha ülevaade juba olemasolevate sarnaste keeleõppe mobiilsete rakenduste kohta. Kõik käesolevas lõputöös kasutatud 3D mudelid on autori enda loominguiline töö.

Selle lõputöö jaoks on kasutatud palju erinevaid allikad, sealhulgas ka akadeemilisi. 3D modelleerimiseks on kasutatud erinevaid Blender'i kasutusjuhendeid ning õpikuid, kaasaarvatud Elisabet Heikkilä «A Guide to Building a 3D Game Character», Fabio Remondino ja Sabry El-Hakim «Image-based 3D Modelling: A Review». Mängu dialoogide ja põhiloo väljatöötamise jaoks on kasutatud: «Англо-русский словарик в картинках» (inglise-vene sõnastik piltides) ning mitmed inglise keele õppetunnid, mis on spetsiaalselt välja töötatud venekeelsetele algkoolide lastele.

Lõputöö tulemuseks on töötav mäng, millel on kõik töös kavandatud funktsioonid.

LICENCE

Non-exclusive licence to reproduce thesis

I, Ivan Švaiger (date of birth: 12.06.1992),

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for addition to the DSpace digital archives until expiry of the term of validity of the copyright,

“Development of 3D models, animations, storyline and dialogue system for interactive English learning mobile game for Russian speaking users” supervised by Assistant Andre Sääsk and Nadežda Furs-Nižnikova.

2. I am aware of the fact that the author retains the right referred to in point 1.
3. This is to certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Narva, **22.05.2018**