Character customization: Animated hair and clothing

María Saborit Ribelles

Final Degree Work

Bachelor's Degree in Video Game Design and Development

Supervised by: José Martínez Sotoca

July 1, 2019

Acknowledgments

To Manuel Gavilán Ortiz, for being the best travel companion for the last four years. Without your infinite support I would not have gotten this far.

To my mother, my father and my sister, for their support and believing in me when it seemed it was too late for me.

To my supervisor, José Martínez Sotoca, for trusting me and in this art related project.

To Myriam Farraces García, for being always willing to share her knowledge about art with me.

To Cristian Montoya Martínez, for his disinterested help through these years.

Abstract

This project consists in designing and implementing a 3D female character editor. It is focused in modeling and animating the female character, hairstyle and clothes. This editor will be developed using the Unity 3D Game Engine. It will consist in an interface that allows changing skin and eye color, style and color of hair and, lastly, the clothes the character is to wear among a catalogue of predefined models. With each change, the character will respond with an animation in order to improve the experience of perceiving the final style of the character.

Key words

3D,Customization, character, clothing, hair, animation.

Index

Figure's index

1.Introduction

Contents

Since video games first got access to the Internet, and more specifically since the MMORPG (Massively Multiplayer Online Role-playing Game) genre exists, players have found their need to differentiate themselves from each other in a virtual world where hundreds of players choose the same type of character. At first, it was enough with the capability to change the color of their hair, skin or clothes.

However, as new graphics technology came out and became more realistic, it seems that there is even a competition in the video game industry to have the most impressive, customizable and realistic character editor.

Large companies dedicate millions of euros to improve their character editor, generating new jobs for hundreds of artists and programmers.

More and more players now find more important the creation phase of the character, to the point they even dedicate hours in the process. This is because, once chosen, great part of the character will not be customizable again and it is our avatar inside the digital world.

1.1 Work Motivation

For me, one of the most important parts of an MMORPG has always been the character editor. Since I was a child, I have found fascinating the design and animation of characters, both in animated films and in video games. In particular, I am impressed by the achievements made in the industry regarding hair, fabric and water simulation. For this reason i enrolled 4 years ago this particular degree and with this in mind, I want to finish it.

Based on the knowledge acquired in this degree and having improved my skills in modeling and rigging in parallel formation, this project will be focused on designing, animating and implementing into Unity engine a 3D character that allows changing its appearance and clothes.

Taking into account the little realism achieved in hair and clothing using Unity tools, it is necessary to use external tools to create professional animations for both hair and clothing. After that, these animations are imported and played in Unity Engine.

It is expected for this project to obtain the best results in cloth and hair animation to take a step beyond the limitations when using Unity.

1.2 Influences

As for the influences that have inspired this project, these are the best character editors that are part of the video games as well as similar software and desired artistic styles.

1.2.1 Character editors

As for great character editors, we are taking as a reference those of the following video games (See Fig 1, Fig 2 and Fig 3):

- Black desert
- Tera
- Soul Calibur VI
- The Sims™ 4

Fig 1.Black desert character editor

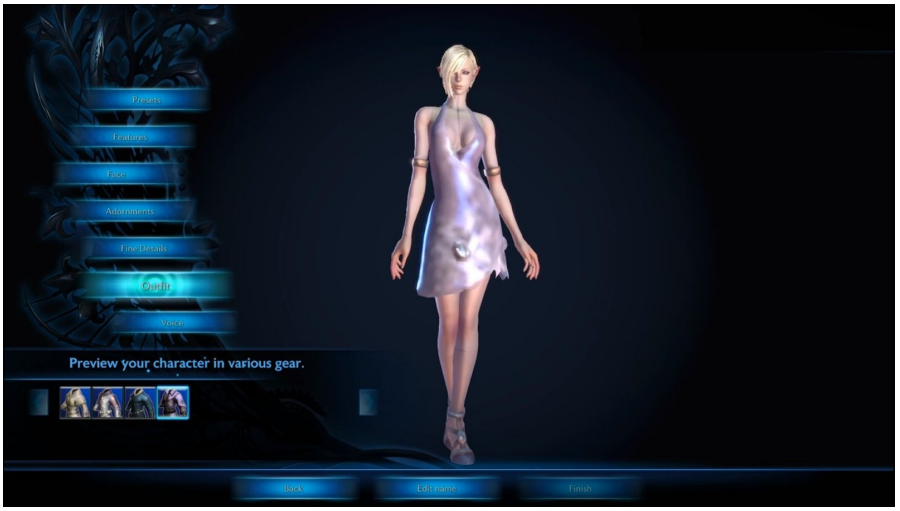


Fig 2.Tera character editor

Fig 3.The Sims™ 4 character editor

1.2.2 Similar software

"Cute Moe Dress Up" (Fig. 4) is a similar software for this project. It performs a simple hair and cloth animation, which is less realistic than the style this project is intended to have. However, it is a faithful representation of the main concept of this work.

Fig 4.Cute Moe Dressup 3D

1.2.3 Art

The artistic style that directly influences the art of this project is called "stylized". It is an intermediate style between cartoon and realistic styles, which retains a somewhat realistic proportions but using simple finishes as in cartoon. This style is very useful in

animation movies and in video games because it does not require high detail to look astonishing.

As examples of “stylized” art style, games like “Kingdom Hearts 3” and movies like “Ralph breaks the Internet” come to mind.

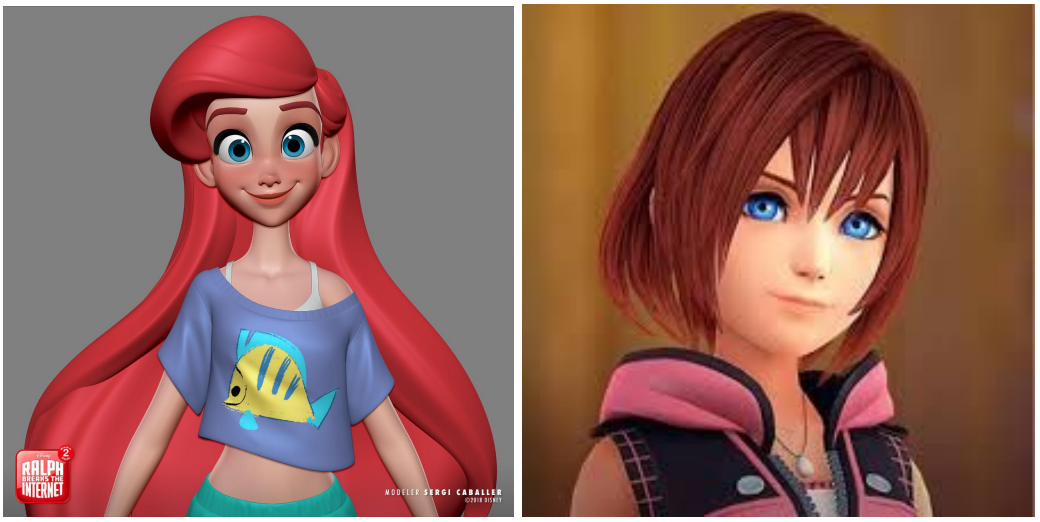


Fig 5. “The little mermaid” from Sergi Caballer and “Kairi” from KH 3

1.3 Related Subjects

This is the list of subjects more related to the knowledge applied in this project:

- **VJ1216 - 3D Design**

  On this subject the necessary knowledge to model 3D objects for video games is found.

- **VJ1226 - Character design and animation**

  In this subject the knowledge is provided to start in the rigging and character animation.

- **VJ1204 - Artistic Expression**

  A subject in which the basics of drawing, composition, proportions and techniques are learned.

- **VJ1227 - Game Engines**

  A subject in which the student learns to work with a video game engine (Unity).

- **VJ1223 - Video Game Art**

  Subject needed to develop environment and characters.

- **VJ1224 - Software Engineering**

  A subject in which to learn the agile methodologies necessary to participate and manage a project.

1.4 Objectives and expected results

The objectives for this project include the complete development of a character starting with its model to its implementation in a character editor.

Main objectives:

Objective 1. To make a complete character artist workflow (sculpted, retopology, uv, Hand painting textures, hair, cloth...)

Objective 2. To create a stylized design for body, hair and clothes.

Objective 3. Animations of clothes and hair.

Objective 4. Introducing those animations in a Unity UI that allows the user to see them applied in clothes and hair.

Objective 5. To allow the user to change the hair and clothes of the character through buttons.

Secondary Objectives:

Objective 6. Illumination and favorable environment for the cartoon immersion in the character editor.

Objective 7. Improve the character editor to allow changing the color of the eyes, hair and skin.

Objective 8. Secondary screen called "Photo Call" in which the user is able to capture a screenshot of the character.

Expected results:

1. Clean polygonal mesh for the character.
2. Fluid animations that perform no clipping in clothes and hair.
3. Attractive HUD that allows changing clothes and hair style.
4. Alternate animations each time a change in the model is made.
5. Rotating camera that performs zoom in order to improve the main view.
6. Best texture selection of the clothes.
7. Capture into an image file the final result of the customization.

1.5 Development Tools

The software used to carry out this project is the following:

1.5.1 2D Art

- Adobe Photoshop CS6[2]: To design 2D elements such as buttons, backgrounds and textures
- Game-icons[3]: online asset which allows button customization.

1.5.2 3D Art and Animation

- ZBrush 4R8[4]: used as modeling/sculpting tool to sculpt the body of the character.

- Maya 2019[5]: 3D main modeling tool used in this project for retopology, UVs creation, rigging, character animation and as an export manager to Unity.
- Vroid Studio[6]: Anime character editor. Its hair builder is used to streamline the process of creating tufts
- Blender 2.8[7]: 3D modeling tool used to export Vroid format to Maya.
- Substance Painter 2019[8]: The reference texturing app for 3D professionals and enthusiasts.
- Marvelous Designer 2018 personal[9]: One of the most advanced cloth design tools in the industry used by professionals to achieve realistic clothes while saving time.
- Adobe Mixamo[10]: Online auto-rig tool with a bank of animations oriented to video games.

1.5.3 HUD implementation

- Unity 3D 2019.1.2f1[11]: Multi-platform Game Engine used to program the character editor UI.
- Visual Studio Code[12]: development IDE.

1.5.4 Documentation and final presentation

- Google Docs[13]: tool for documents.
- Google Drive[14]: tool for storage.

1.6 Justification

In order to meet the deadlines of a 300 hours project and taking into account that there is only one person to develop the entire project, it has been decided to streamline the work process by relying on all the software available for a professional performance of work.

1.6.1 2D Software justification

Photoshop CS6 has been chosen because it is the best software for 2D edition and also the one which most people is used to. *Game-icon.net* website has been used to develop the icons and optimize the process.

1.6.2 3D Software justification

In order to obtain a professional finish, the best software for excellence in sculpting has been used: ZBrush 4R8, taking into account the basic knowledge acquired in parallel to the formation during this degree.

Maya 2019 has also been used as the main modeling program, as it is intuitive for developing UVs and retopology and is the most complete and optimized in the rigging. There was no previous experience with this software. However, its use has been combined with *Blender 2.8* , needed to read Vroid archives and export them to Maya.

On the other hand, in terms of body texturing, *Marvelous Designer 2018* has been chosen since it is a professional hand painting software that allows the user to obtain the best finish in the shortest time.

Regarding the design of hairstyles, *Vroid Studio* has been used as support since the creation of hairstyles with professional results is as tedious as repetitive. This tool

generates strands through user defined parameters and performs great finishes for the design and creation of hairstyles.

In order to focus on the specific animation of hair and clothing, and with the intention of streamlining the workflow, the *Adobe Mixamo* tool has been used to generate the basic body rig. This choice was also made taking into account the entire bank of animations available on its website which would allow the character to be implemented in any kind of game.

1.6.3 HUD implementation justification

In order to visualize all the work mentioned above, *Unity 3D 2019* has been chosen since it is the engine most commonly used through this degree. This saves a great amount of time and is easier to use than other engines.

Visual Studio is a tool that performs nicely as they have already been used through this degree and are also free to use.

1.6.4 Documentation and presentation justification

Finally, *Google Docs* has been chosen as the tool for documentation and *Google Drive* as the backup folder. This decision was made taking into consideration that it is simple to use and allows sharing al the information with the supervisor at any moment while allowing direct feedback on the documents.

"This page has been left intentionally in blank".

2. Planning and resources Evaluation

Contents

This chapter focuses on the initial planning and the main aspects that were taken into account for it.

2.1 Initial planning

The following table displays the main tasks that were initially planned and their estimated cost in hours:

Task	Hours
Investigation and references	20h
Model the female character basic model	10h
Basic model topology	25h
Model hair style	25h
Model cloth elements	25h
Rigging	40h
UVs	25h
Texturization	25h
Animation	25h
Program and Implementation into Unity	50h
Writing documentation	40h
Presentation	10h

Total hours:

320h

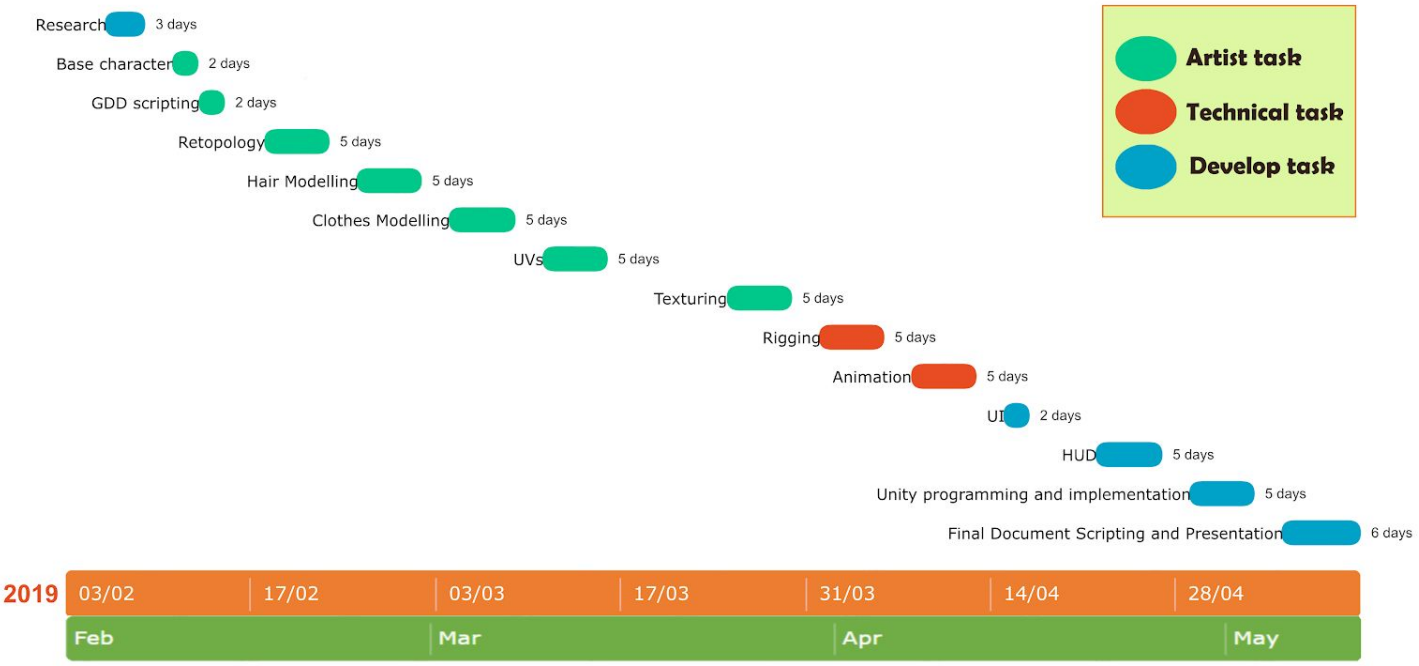


Fig 6. Gantt Diagram

2.2 Resources evaluation

Personal budget

The estimated cost of this project divided by blocks of tasks does not include the learning time for each of these tasks. The following table displays an estimate of these costs:

Role	Field	€/h	Hours	Cost
3D artist	Art	10	110	1100€
Technical artist	Art	15	110	1500€
Programmer	Program	15	50	750€
Documentation	Other	10	50	500€

Total Cost:

3850€

The salary information is based on an average estimation of the job offers found on the *Stratos*[15] website.

Software licenses budget

The price of licenses can be very high for 3D art programs, but there are useful student licenses that are cheaper or even free to use. The cost for each license can be found on its respective website.

Licenses cost:

- ➔ Maya 2019 (free for students)
- ➔ Zbrush 4R8 (157,13€ / 6 Months)[4]
- ➔ Marvelous Designer 2019 (44,14€/ month)[9]
- ➔ Substance painter 2019 (€ 129/perpetual license)[8]
- ➔ Photoshop CSS6 (19,66€/month for students)[2]
- ➔ Unity 3D (free)
- ➔ Blender (Open source)
- ➔ Visual Studio Code (free)

Total Cost for 5 months:

427,57€

Final Cost of the project	4.278,57€

"This page has been left intentionally in blank".

3. Design

Contents

3.1 Introduction

This project consists of implementing a female character edition. In this context, the design, animation and implementation of the character is needed.

The character will be modeled following the standards of the professional video game industry, taking into account the recommended workflow.

The editor and its HUD will be implemented in Unity 3D.

The interface has been designed so that it can interact with changing clothes and hairstyles.

Special attention will be paid to cloth and hair animation and their interaction with other animations as they loop.

3.1.1 Workflow

The final implementation of this application is divided into the following work blocks (As seen on Fig 7):

1. Character design: Sculpting, modeling, retopology, UV, texturing and hand painting.
2. Animation Design: Rigging and animation.
3. Environment Design: Importing elements into Unity and developing the HUD.

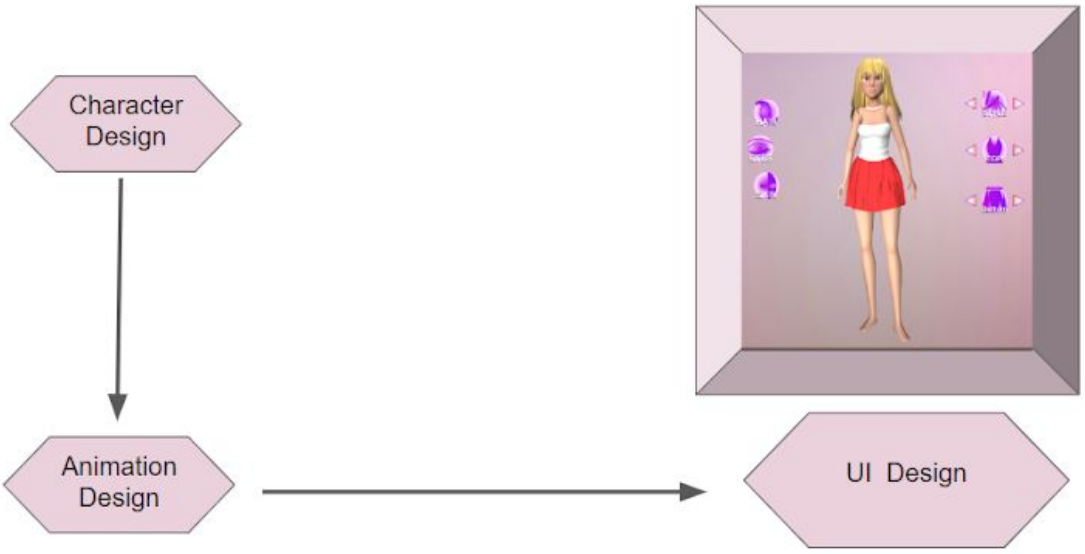


Fig 7. Workflow diagram

3.2 Character Design

The character is the central pillar of the application since everything is related to interacting with it.For this reason, there has been special interest in its design, to which was devoted many hours of searching for references in order to find the perfect balance for the level of detail. This was necessary so that it does not affect the overall performance of the rest of the tasks.

3.2.1 Style

As it was mentioned before, the character style will be “stylized”(As seen on Fig.8). It is widely used in animation and video games as it allows the user to give all the needed realism through the use of textures while being close to realism.

The difference between the “stylized” and the cartoon style is clear to see. It is then more relevant to see the difference with the realistic style.

The realistic style has much more detail in the painting of the skin and in the finishing of the hair. However, it is allowed to keep a great detail of it in clothes and texture (As seen on Fig.9).

Thus, the character has been designed as “stylized”, to save time and speed up the workflow. This is because the more details a model has, the more complicated its development becomes.

As for the age of the character, she will be about 18-20 years old, with semi realistic proportions, focusing the detail on the face, eyes and mouth.

Fig 8. Stylized female character examples

Fig 9. Comparison between realistic style (left) and stylized (right)

3.2.2 Hair

The style of the hairstyles will also suit that of the body, so it has been designed by tufts instead of individual hairs.

This level of detail would be the most desirable for a video game that does not require hyper realistic graphics. Figure 8 shows two examples of “stylized” hairs.

At least 3 different hairstyles will be created to be displayed:

- Mane
- High pigtail
- Two braids

3.2.3 Clothing

As previously mentioned, stylized clothing can perform a medium realism supported by realistic textures. This is achieved because a high quality texture does not mean extra work for an artist since they are already generated. In fact, it would involve extra work to hand paint those textures to display a more cartoonish look.

Three levels have been designed for clothes:

- Top
- Bottom
- Shoes

Each of these sections will contain at least three garments. The player can combine skirts, pants, shirts, tops, etc.

3.2.4 Color Palette

There are three elements that enable color changing:

- Hair
- Eyes
- Skin

3.3 Animation Design

There is an idle animation for when the character is not receiving any change and five additional animations that will trigger and change whenever an element is changed at the editor.

These are its characteristics to achieve complex animations —considering complex animations, those which have many factors to take care of as hair, body and clothes combined.

- Body: for the generation of the basic skeleton Mixamo will be used, thus allowing the animation of the body of the character with any mockup of its bank
- Hair: A system of dynamic joints has been used for the hair animation.
- Clothes: A cloth simulator has been used.
- The skinning of the body, clothes and hair will be hand made and detailed in order to achieve maximum quality.
- The animation will be fluid and will not allow clipping for both clothes and hair.

3.4 Environment Design

This part of the project will be developed using Unity3D and consists of 3 main screens that enable navigation(As seen on Fig.10).

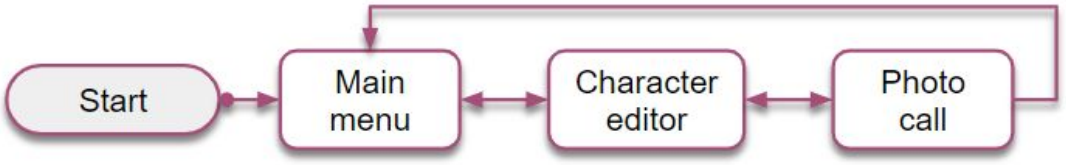


Fig 10.Workflow diagram of the different phases of the character editor.

3.4.1 Main menu

It shows up as soon as the application is loaded. The user is able to start customizing the character from here and also to exit the application (As seen on Fig.11).

There are 2 buttons:

- Play button: Leads to character customization.
- Exit button (X): Exists the application.

Fig.11 Main Screen first sketches

3.4.2 Character editor

This screen contains the options to customize the character. It also allows the user to export the model, to go back to the main menu and to access the “photo call”.

The aesthetics will be transparent with a soft background to focus on the character. By clicking on the character and dragging the mouse to the left or to the right the camera rotates.

The options to modify the character by using arrow buttons are:

- HairStyle
- Top cloth
- Bottom cloth
- Shoes

- Skin color
- Eyes color
- Hair color

On the other hand, these buttons support other functionalities:

- Back button: Leads back to the main screen.
- Photocall button: Leads to photo call screen.
- Zoom In button: approaches the character.
- Rotation button: rotate the character.

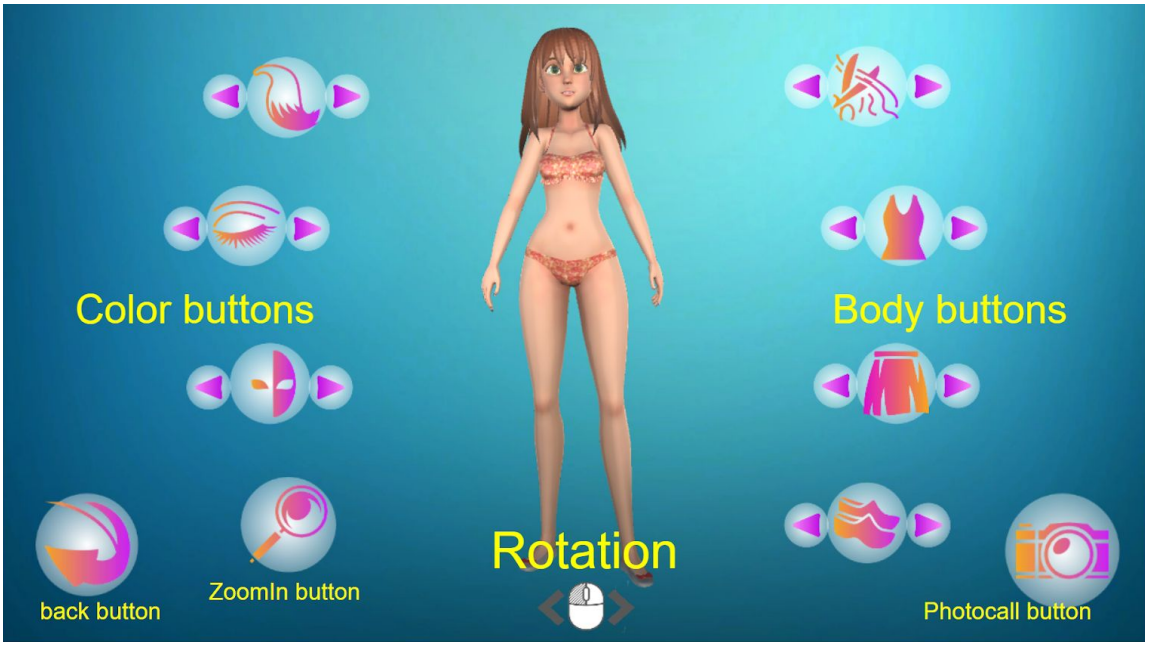


Fig.12 Character editor first sketches

3.4.3 Photocall

On this screen the user is able to make a photo session to the character. The user can save these photos so they can be used later on. It is also allowed to go back to the character editor or to the main menu.

- Back button: leads back to character editor.
- Action button: play an animation.
- Snapshot button: Saves a screenshot in PNG format.
- Background button: Changes the background.
- ZoomIn button: approaches the character.

Fig.13 Photo Call screen first sketches.

"This page has been left intentionally in blank".

4. Work development and results

Contents

4.1 Work development

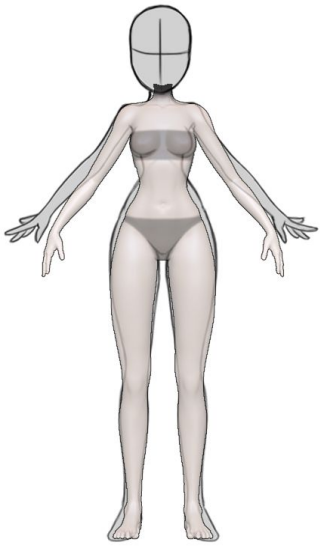
In this section, the entire process of developing the application is divided into three large blocks. To do this, the chronological order of the planning will be used.

4.1.1 Artist development

All the process that does not need a technical part, will be detailed below.

The necessary tasks in order to design a game-ready character are:

4.1.1.1 References



An intensive search for references was made. Among all of the images that could serve as a reference, a few were selected as blueprints. Pinterest was used as it is a huge art repository.

Since no reference image suited the desired style, Photoshop was used to superimpose two blueprints and generate one of their own (As seen on Fig.14).

Later it was imported in Zbrush as a blueprint.

Fig.14 Reference blueprint

4.1.1.2 Character Blocking

Once the reference blueprint was made, the modelling started by blocks. This technique is called “blocking” or “blockout” and it consists of inserting base geometries to outline the proportions and modify them with ease (As seen on Fig.15).

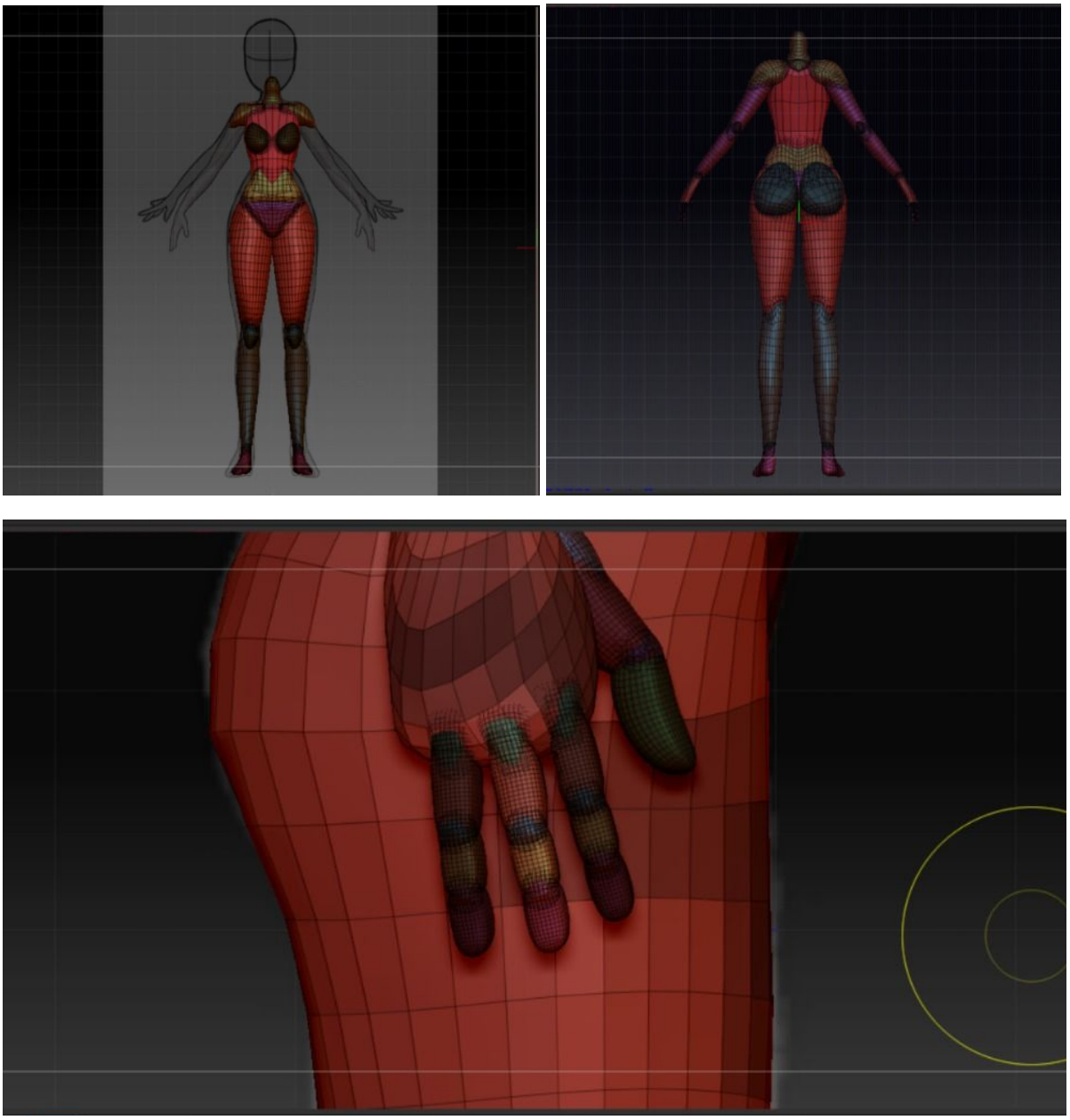
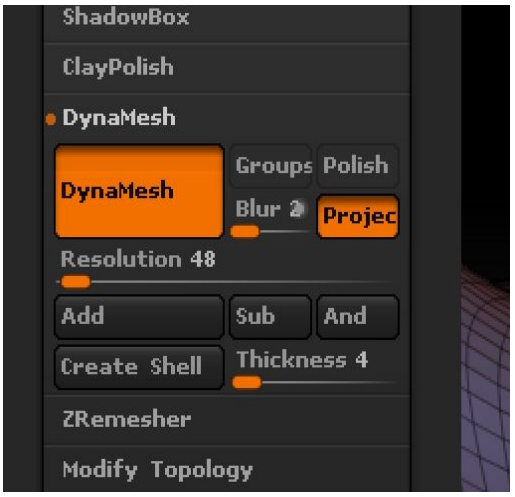


Fig.15 Character Blocking views

Once the proportions were adjusted in all the views, a Dynamesh was applied to join the blocks in the same mesh.



Dynamesh also has the effect of merging two sections of your SubTool (SubTools are separate polygon objects) together if they overlap.

The resolution was set to a reasonable value in order to keep sculpting (Fig.16).

Then the smooth tool was applied to all newly merged joints to get a uniform mesh(As seen on Fig.17).

Fig 16. Dynamesh Options

Fig 17. Character view after smoothing

4.1.1.3 Character detailed design

Once the unified complete model was achieved, the details were polished by using the zbrush brushes to sculpt.

Fig 18. Unified model with polished details.

Once the unified and detailed complete model was achieved (As seen on Fig.18), there was found that the mesh had too much resolution and the mesh was very dirty. It was necessary to use ZRemesher.

Fig 19. ZRemesher options

ZRemesher is the tool of Zbrush that allows obtaining a mesh in low resolution, optimal to be subdivided (As seen on Fig.19). Therefore, it is a tool of automatic retopologizing.

Before applying ZRemesher, a duplicate of the body mesh was made to store the high details in one and apply ZRemesher in the other (As seen on Fig.21).

Then guides were drawn surrounding the areas where more detail is required, so the algorithm of retopologizing knows how to redistribute the quads (As seen on Fig.20).

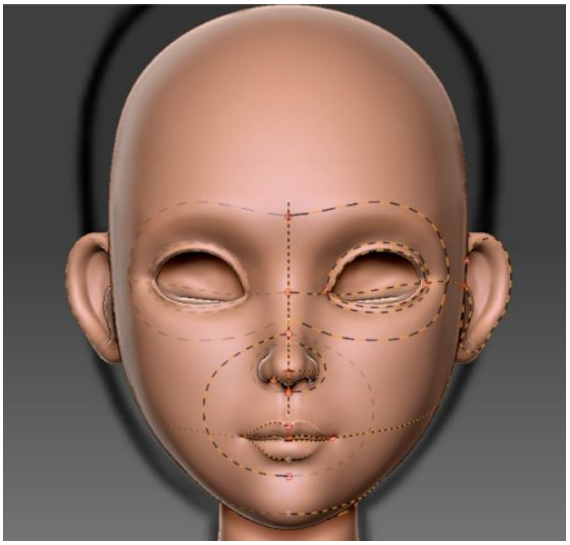


Fig 20. Before Zremesher

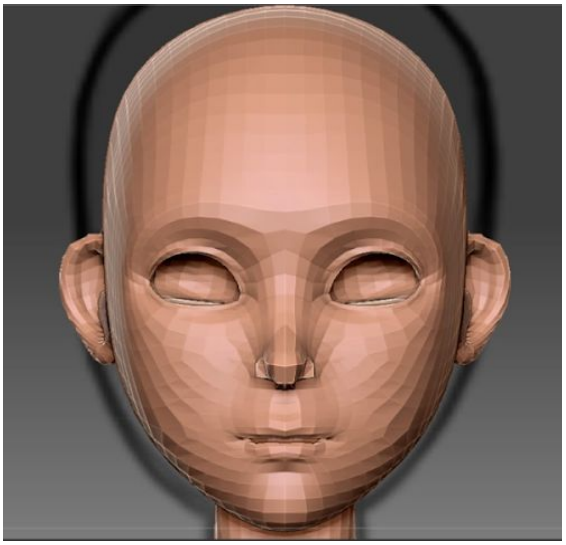


Fig 21. After ZRemesher

Next, the details of the old mesh with dynamesh (high) were projected onto the new ZRemesher mesh (low). Detail was polished alternating between adding subdivisions to the mesh and projecting the details again to get the finish of the figure —about 4 subdivisions (As seen on Fig.22).

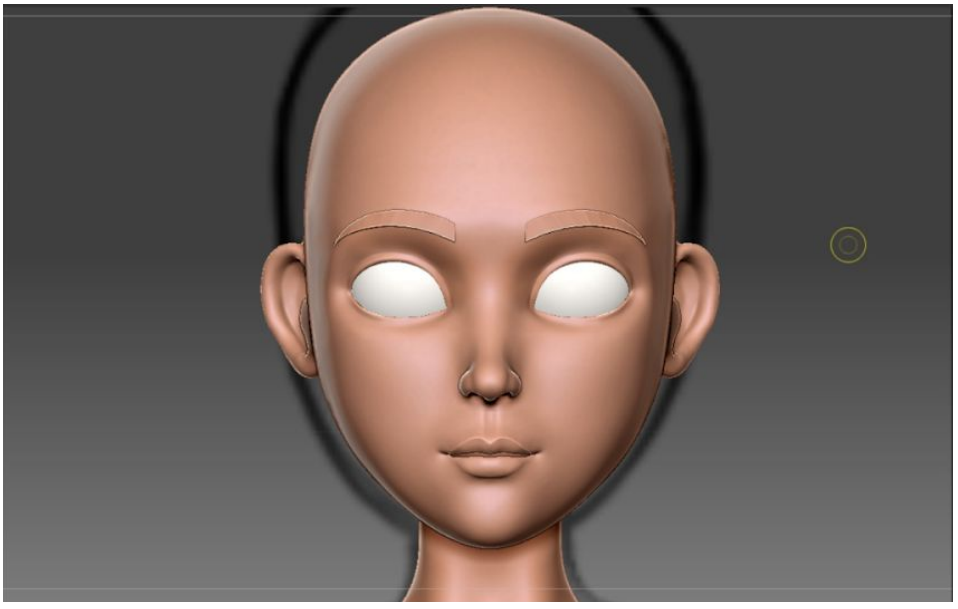


Fig 22. Finishing details after 4 subdivisions

The difference that is seen between a version with Dynamesh(As seen on Fig.23) and the other using Zremesher + subdivisions and projection of details in the figures(As seen on Fig.24) is noticeable. Work continued with the cleanest mesh keeping the initial dynamesh.

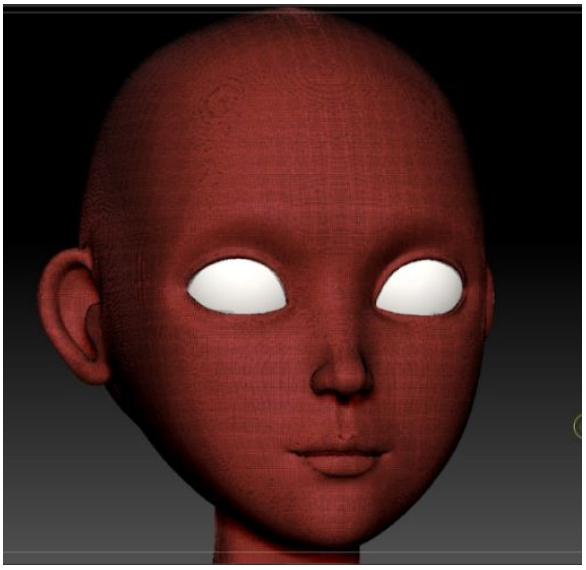


Fig 23.Dynamesh version

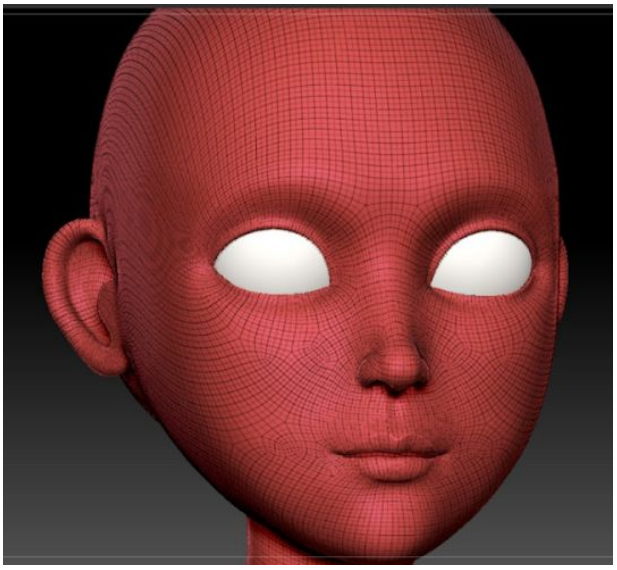
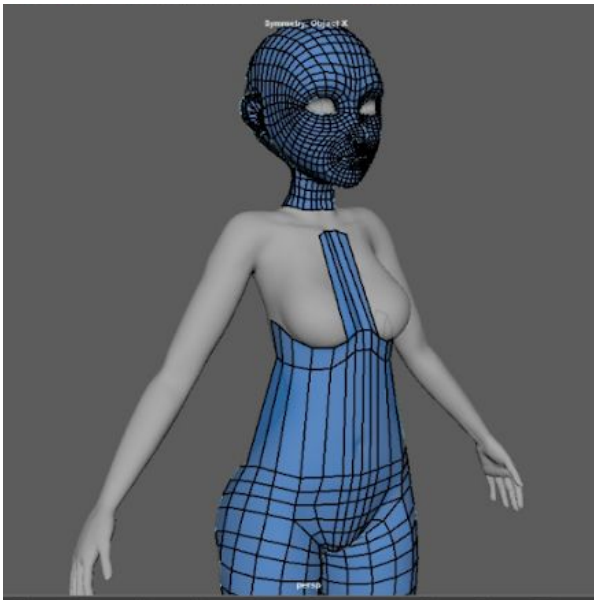


Fig 24. ZRemesher subdivisions version

After adding the details of eyes, mouth and nails, the next step was to decimate, reducing the number of polygons in the mesh automatically without losing the initial shape. This was necessary in order to export the object to another program such as Maya. If this was not done, Maya would not be able to manage this vast number of polygons.

4.1.1.4 Retopology



Retopology is the act of recreating an existing surface with more optimal geometry. A common use-case is creating a clean, quad-based mesh for animation(As seen on Fig.25). Retopology is completely necessary to be able to handle a low poly mesh. If the mesh was left with all the details in high poly, it would slow down the program and the animations would be much more difficult to make.

To carry out the retopology, the model done in Zbrush was imported into Maya.

Fig 25. Retopology in process

After that, the "make live" command was applied. This command works like a magnet for any new geometry created nearby.

The next step was to apply the tool of retopology in Maya called “Quad drag”. This tool creates, modifies and deletes quads to cover the original mesh completely.

In order to obtain a great topology ready to be animated, the joints were taken into account, increasing the number of loops near them (As seen on Fig.26(left)), thus allowing for better animation.

It was necessary to pay special attention to the facial loop always taking into consideration the “mask” and the “muzzle” that surround the eyes and mouth respectively for better animation.

Once the process was finished, a new low poly mesh was obtained ready to be worked on (As seen on Fig.26(right)).

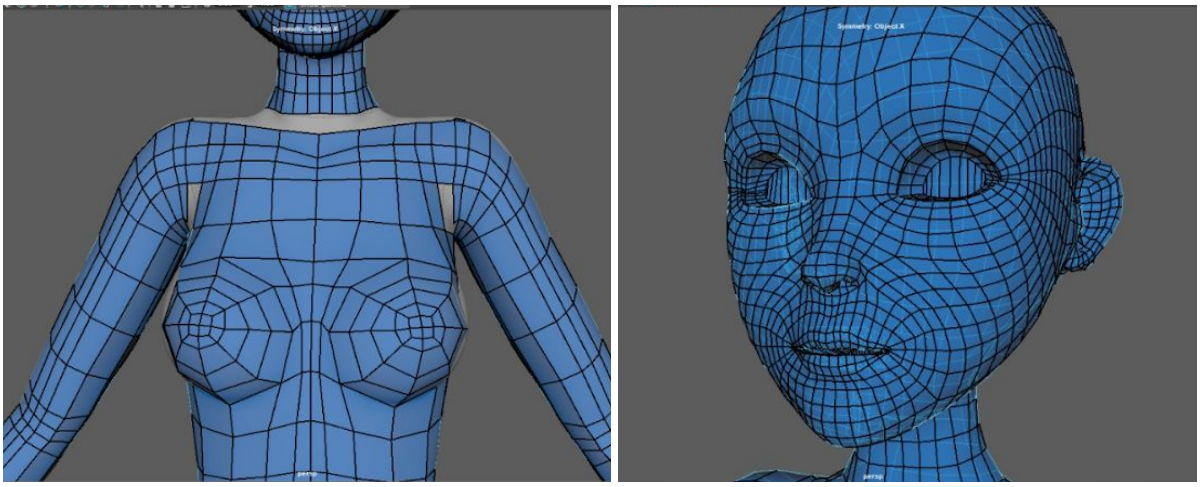


Fig 26. Retopology in progress (left) and face finished Retopology (right)

4.1.1.5 Hair design

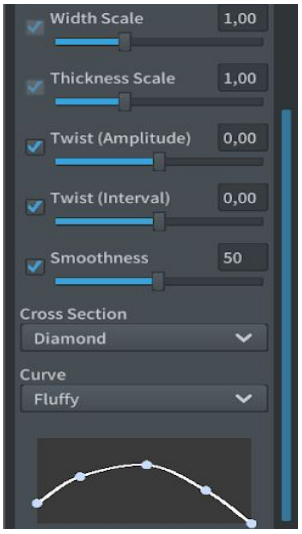


Fig. 27. Hair Parameters

For the design and creation of the hairstyles, the hair builder of the tool Vroid Studio was used.

This tool allows adjusting a vast number of parameters for the generation of tufts.

A “Cross Section” has been used, assigned to the “Diamond” shape (Fig.27), as it offers a higher resolution and a better finish.

Using the Twist parameters (Amplitude and Interval) on two hair tufts and adjusting their position to match, the “braid” effect was created (as seen on Fig.28).

Once the desired strand had been obtained, the next step was to create hair models by drawing curves on the head. For this task a 3D grid (As seen on Fig.29) has been used that serves to trace the strands over it and is fully editable through control points.

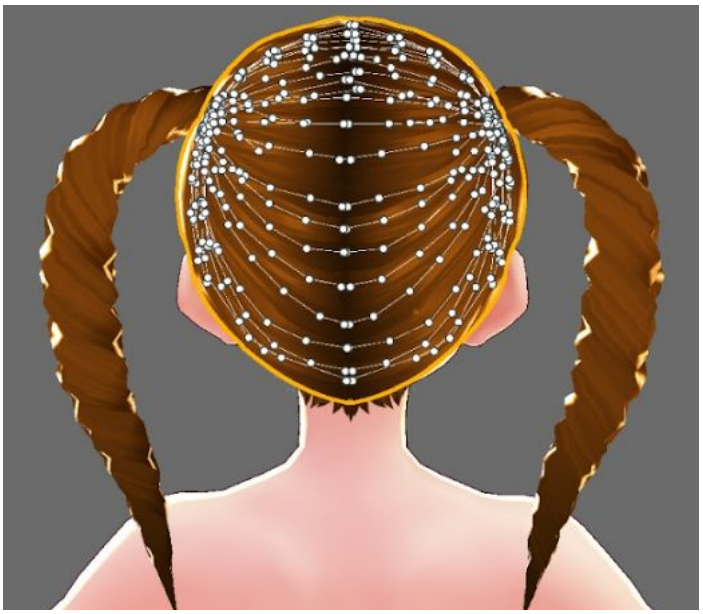


Fig 28. Example hair 1

The curves and the vertices of each strand can be seen on the tool window (as seen on Fig.30).

Later it was exported to Blender (since it is the only program available that accepts this Vroid format) and from Blender to Maya for its adaptation to body modeling.

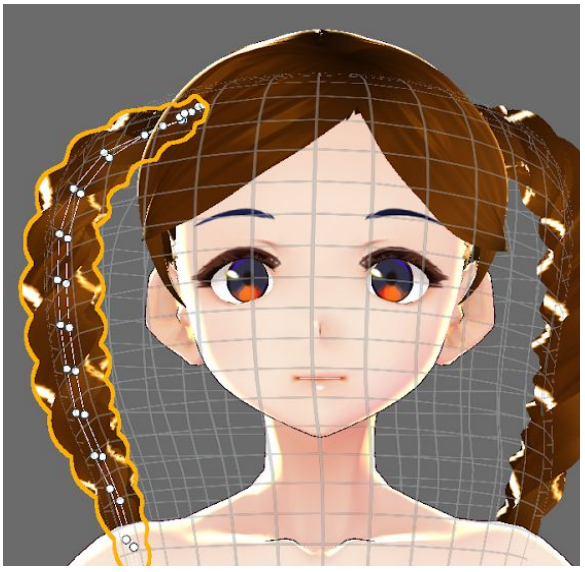


Fig 29. Braid example

Fig 30. Bang example

In Maya the mesh was edited adding subdivisions and scaling the necessary proportions.

After that, the “unlock normals” and “soft edges” buttons were used to soften the mesh. This way, a better finish is achieved without increasing the resolution.

4.1.1.6 Clothes design

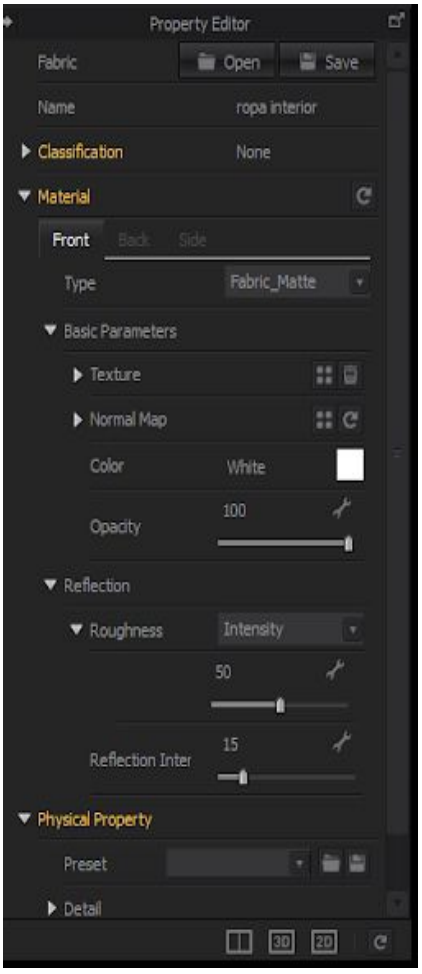


Fig 31. Marvelous Designer

For this task, Marvelous Designer has been used. It is very useful for modelling dresses, shirts and skirts. It creates a patronage of clothing and sews it onto the model.

First, the pattern was drawn piece by piece by adjusting the vertices to the proportions of the body in the "2D Pattern Window" (The right panel is shown in figure 32), and the patterns generated in 3D were then repositioned in the "3D Pattern Window" (left panel figure 32).

Then the "Simulate" button was pressed to start the simulation and see the clothes fit the body. To achieve satisfactory results, the parameters of the "Property Editor" (As seen on Fig.31) were tested and modified until the desired result was achieved.

As a final step, the "Remeshing" command was applied in the "3D Pattern Window", which generates a quite acceptable automatic retopology for animation. Fig.32 shows a clean topology generated by this command.

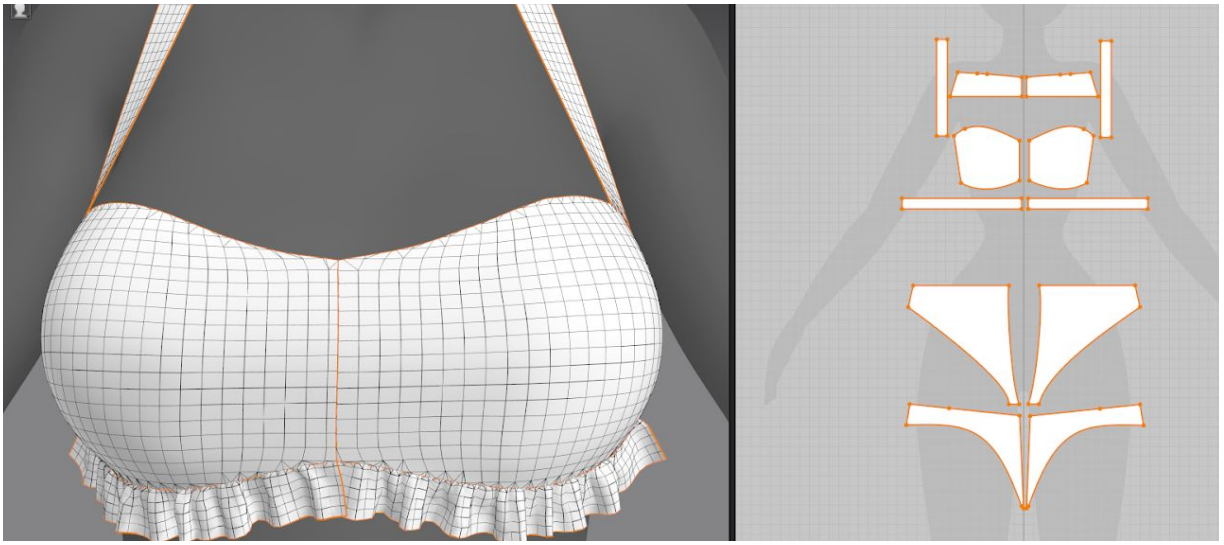


Fig 32. Automatic Retopology

4.1.1.7 UV mapping

UV mapping is the 3D modelling process of projecting a 2D image to a 3D model's surface for texture mapping. A UV map can either be generated automatically by the software application, made manually by the artist, or some combination of both. Often a UV map will be generated, and then the artist will adjust and optimize it to minimize seams and overlaps.

For this model, UV was cut by hand and repositioned taking into account that parts of the body needed more resolution at the time of being painted (As seen on Fig.33). In this case, it was left more space for the face, as it will be more detailed than the rest of the body. For the eyes, mouth, tongue and nails, another UV was generated in order to supply more space for the face.

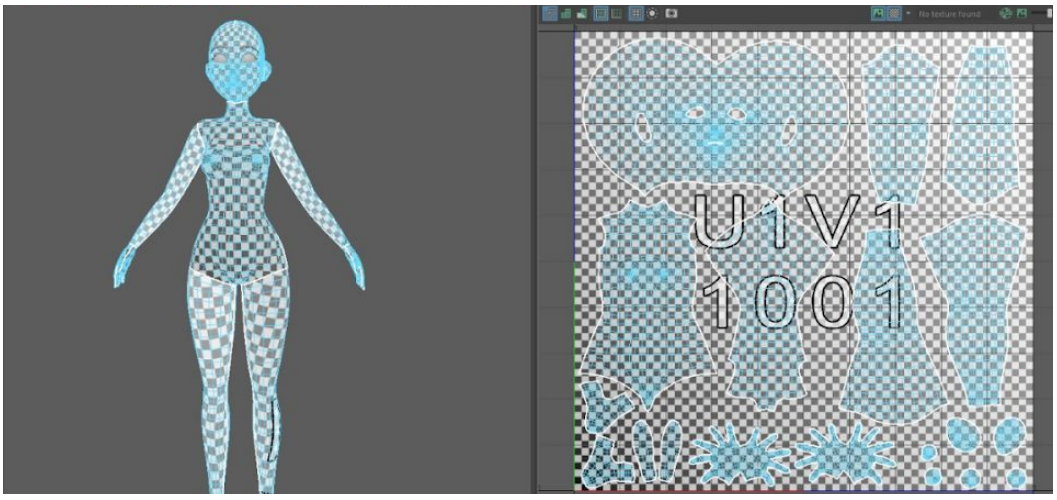


Fig 33. Final UV Mapping

4.1.1.8 Texture implementation

For the texture of the body and face, the technique of "hand painting" was used, which consists of painting with a brush on a 3D model directly.

In this case, the program used was Substance Painter.

- **Preparation for texture**

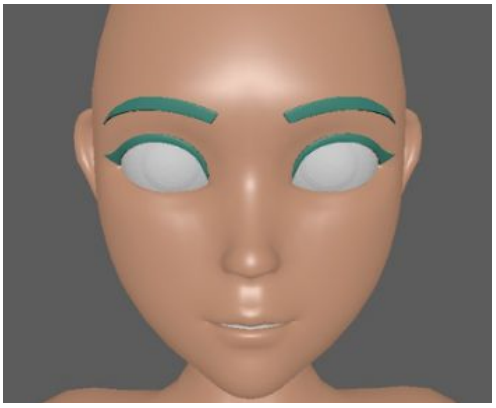


Fig 34. Preparation for texture

To be able to texturize in Substance, it was necessary to prepare the model in Maya, separating the different parts of the model in materials (As seen on Fig.34), so that Substance Painter generates different layers for each material and thus be able to work independently on them. The “soften edges” tool was also applied to soften the final aspect of the mesh.

- **Map baking**

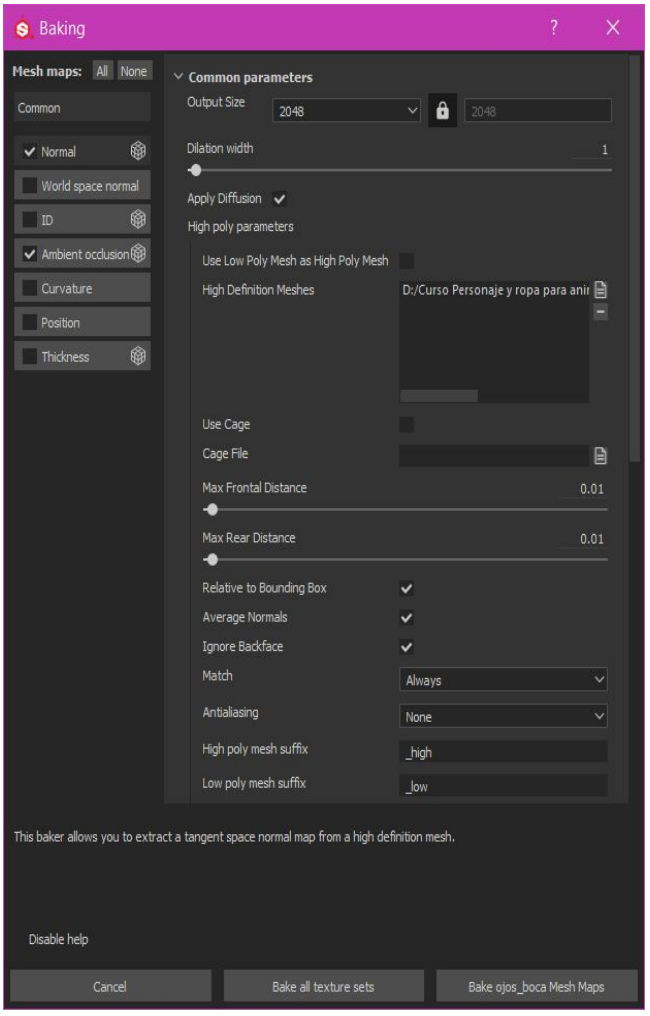


Fig.35 Map baking options

First, the low poly mesh that was developed in Maya was imported. Then, through the "Bake all texture sets" command (As seen on Fig.35), the normal maps and the "ambient occlusion map" were generated. This was needed in order to project the detail of the high poly mesh over the low poly to be used in the unity project.

After that, Photoshop was used to retouch some parts that did not adapt perfectly to the model (As seen on Fig.36 and Fig.37).

Fig.36 Normal map

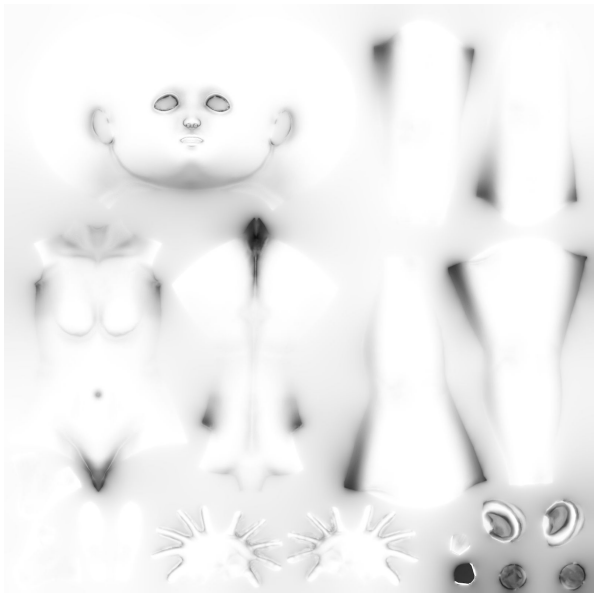


Fig.37 Ambient Occlusion Map

- **Hand painting**

In this case, a PBR (Physically Based Rendering) material was used for the body. It contains a set of digital textures that can be driven by parameters, allowing an infinity of variations from a single file.

Then the details and additional shading for body, eyes, mouth and nails were layered (As seen on Fig.38).

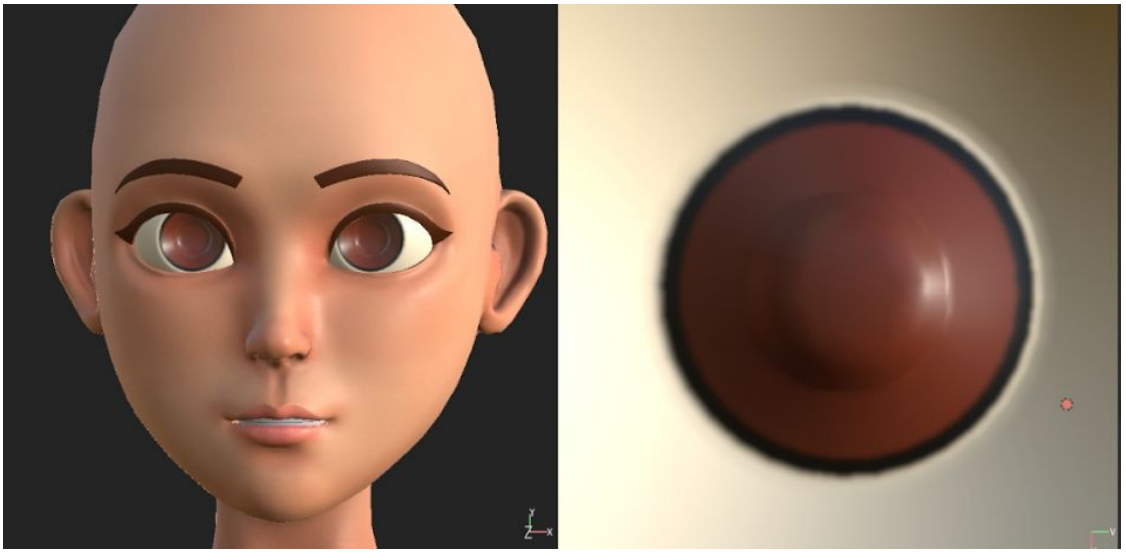


Fig 38. Visor 3D(left) and visor 2D(right)

- **Results**

Once the desired result had been obtained (As seen on Fig.39), the full texture of the body and the rest of the objects were exported with the command:"File> export textures> export".

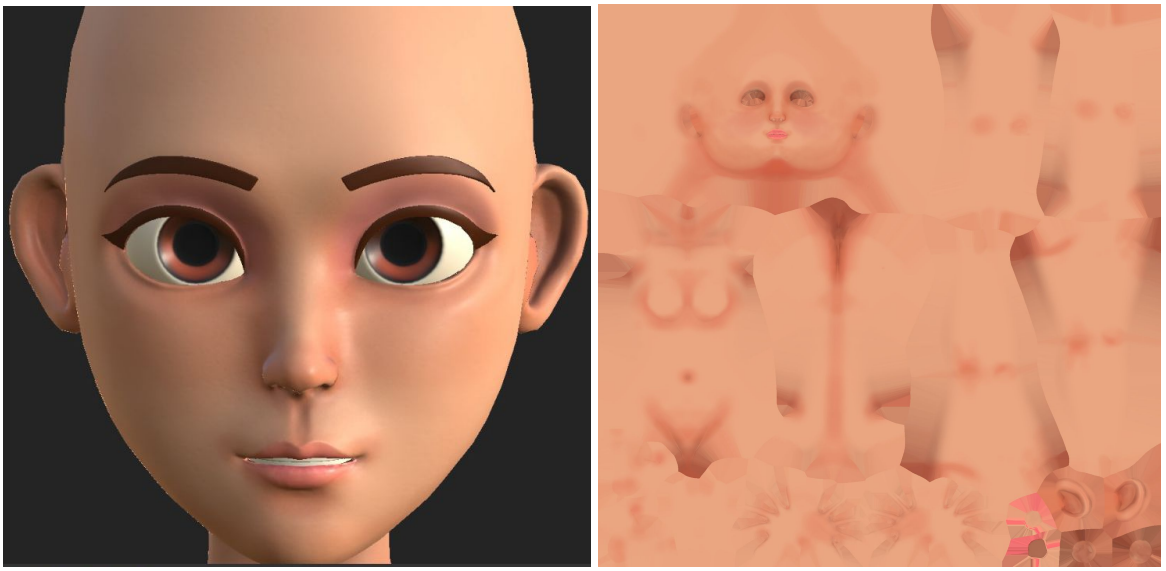


Fig 39. Face View and texture Map

For the clothes, hair and shoes, free seamless textures and Blinn materials from Maya were used.

4.1.2 Technical Artist development

This section covers the necessary tasks in order to develop the animation for the following blocks: body, hair and clothes.

In each of them, different techniques adapted to their specific needs were used, obtaining full compatibility with the Unity 3D graphics engine.

4.1.2.1 Body animation

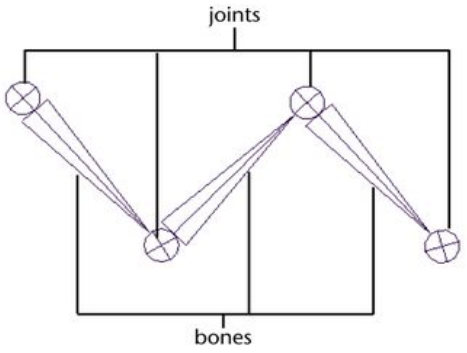


Fig 40. Joints and bones

In this section the workflow of the animation of the character is detailed. To do so, there will be shown the difference between the joints and bones (as seen on Fig.40).

Joints are the building blocks of skeletons and their points of articulation. Each joint can have one or more bones attached to it, and more than one child joint.

Joints let transforming a skeleton when posing and animating a bound model. On the other hand, bones are only visual cues that illustrate the relationships between joints.

Rigging

For the rigging of the body, the Mixamo tool was used, obtaining a base skeleton that will admit any animation of the Mixamo bank.

A total of 6 character animations were also adjusted: 1 for idle and 5 for costume or hairstyle change.Next, the bone skeleton and its adapted animations were exported in FBX to import it into Maya.

- **Skinning**

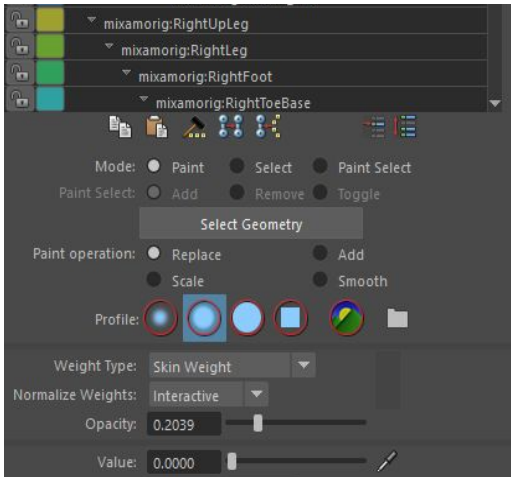


Fig 41. Paint skin weights tool

Though the Mixamo skeleton already supplies automatic skinning, the skin for every joint in the body was retouched by hand, thus adjusting mesh deformations.

For this, the command "Paint skin weights tool" was used (As seen on Fig.41). Using the brush and alternating between “replace” and “smooth” to paint the weights of each joint. The values that were painted set how much each joint influences the painted vertices relative to the other joints of the body. Once the movement of the body seems fluid when rotating a joint, this process is considered finished (As seen on Fig.42).

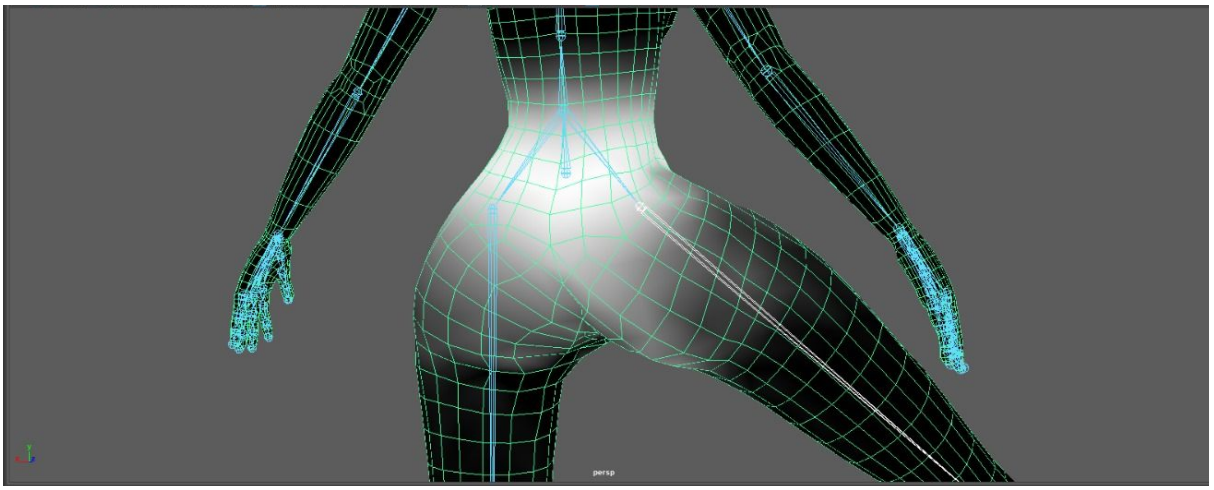


Fig 42. Hips joints skinning result.

- **Baked animations**

  One of the animations stored in FBX was imported to check the animation's fluidity.

4.1.2.2 Hair animation

In this section the animation of the hairstyle based on dynamic joints is detailed. To do so, it is important to have a fully animated body and the hair mesh imported into the same document.

- **Rigging**

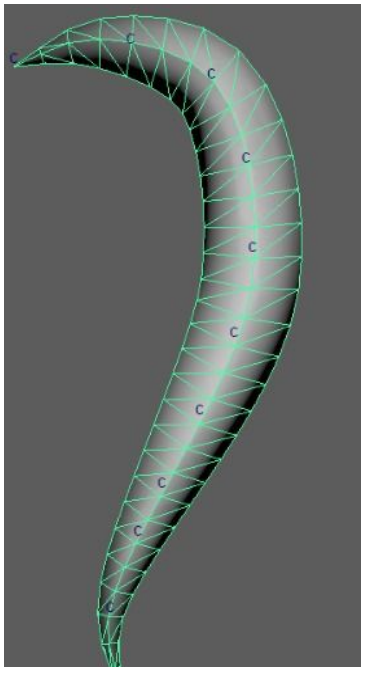


Fig 43. Cluster guides

The process of rigging for hair starts from scratch. A chain of joints was created for each strand.

For a tighter skinning, the joints must go right in the center of the mesh to which the skinning will be applied. To do this, edge loops were selected and for each one a "Cluster" was created (In the Rigging menu set, select Deform> (Create) Cluster). A cluster is a modifier that affects the selected geometry, but in this case it was used to find the midpoint between all the edges and then to position it in the center.

Using the clusters as guides, a chain of joints was created starting from the root of the tuft to the tips (As seen on Fig.43). To create the joints right in the position of the clusters, there has to be activated the "snap to point" command and then create the joints one after the other (In the Rigging menu set (press F3), select Skeleton> Create Joints).

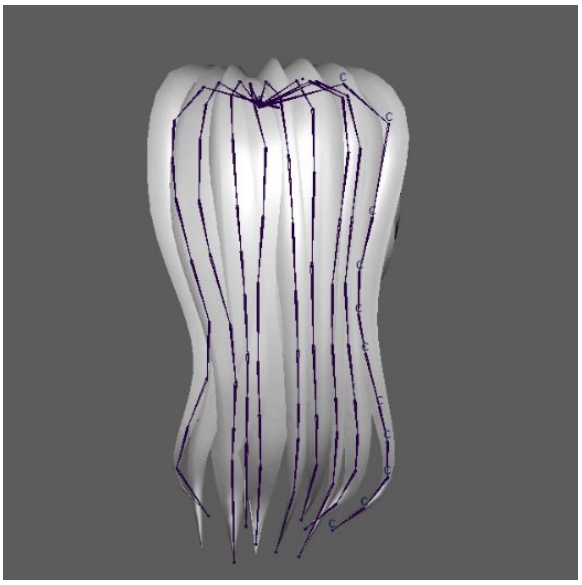


Fig 44. Joint chains

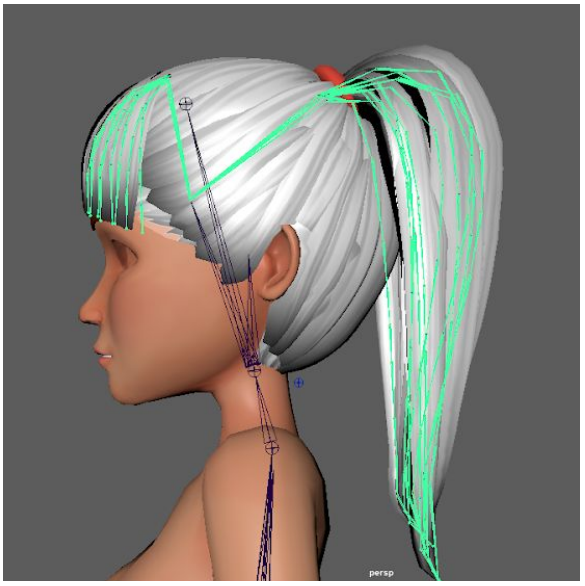


Fig 45. Head as parent of joint chains

After that, the chains of joints (As seen on Fig.44) were bridged to the joint of the head (As seen on Fig.45). In this way, when the body moves through the animations, the joints of the hair will move accordingly.

- **Skinning**

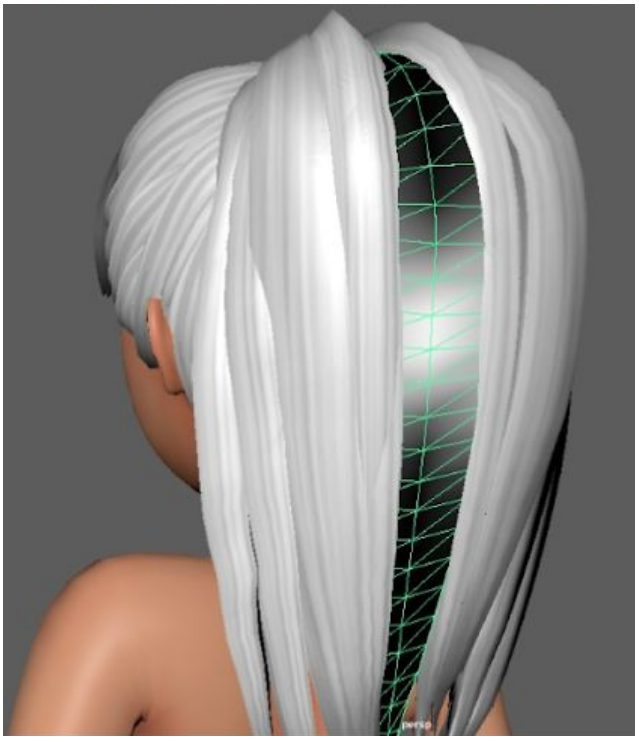


Fig 46. Hair Skinning

First of all, a "Freeze transformations" was applied to the mesh of the hair to set to zero its translations and rotations, which is useful to return the model to its original point.

After that, a "Bind skin" was applied selecting the hair mesh and the joint of the head (In the Rigging menu, select Skin> bind skin). The bind links the movement of the vertices of the mesh with that of the joints, thus allowing the movement of the model, according to the joints.

To obtain a fluid movement in the tufts, it was necessary to soften the joint weights using the "Flood" command of "Paint Skin Weights Tool".

- **Dynamic joints**

Dynamic joints move by taking the movement information of NURBS curves. For this it is necessary to transmit the information with the following procedure (As seen on Fig.47).

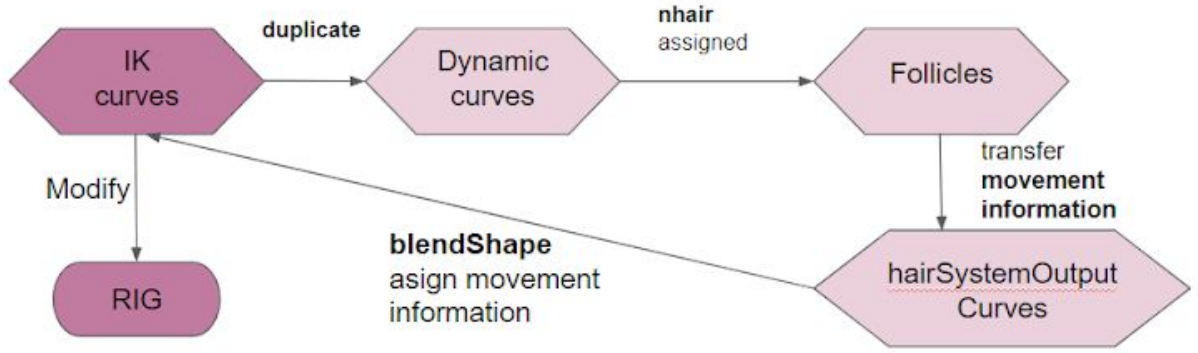
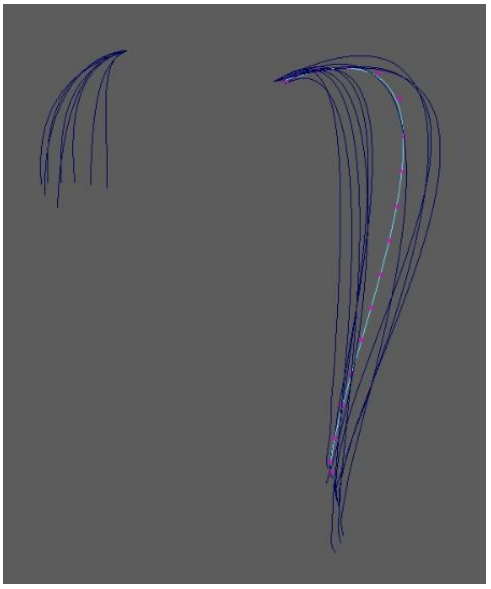


Fig 47. Dynamic Joints Diagram

1. IK Curves



First, an IK spline was created that starts at the root and ends at the tip of each tuft (In the Rigging menu set, select Skeleton > Create IK Spline Handle).

A spline IK handle lets you control a joint chain with a NURBS curve. All joints driven by the spline IK handle stay on and follow the curve (As seen on Fig.48).

The number of points on each curve can be edited to increase or decrease the fluidity of the curve movement (at the time of creating the curve, specify the "Number of Spans").

Fig 48. Curves generated by IK

2. Dynamic curves with nhair

The IK curves were duplicated by grouping them under the name of “Dynamic Curves”. A nhair system was then created and applied to the dynamic curves (Mode"fx">nhair>make selected curves dynamic).

As a result of this nhair system, a follicle has been created on each selected curve and a group of curves called "hairSystem1OutputCurves".

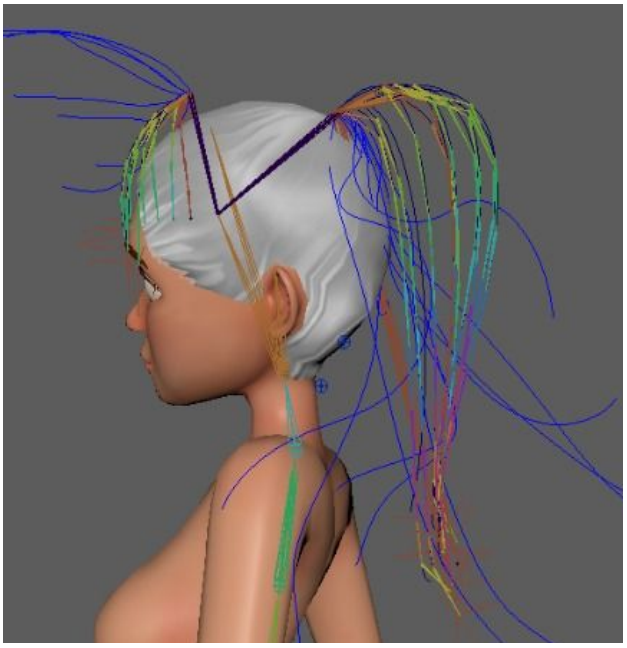


Fig 49. Dynamic Curves

The follicles generated by the new "hairSystem" can vary their hair simulation by modifying the parameters of the "hairSystemshape".

Then all the follicles were selected and in the "Channel Box" the "Point Lock" was modified by setting it to"Base", so the follicles will move freely attached only to the head (As seen on Fig.49).

To reduce the movement of the curve and attract it to the follicle, attributes such as the "Curve Attract" (Attribute editor>dynamic properties> Start Curve Attract) were tweaked with the selected position=1 and attraction position=0.

3.System output curves to IK curves

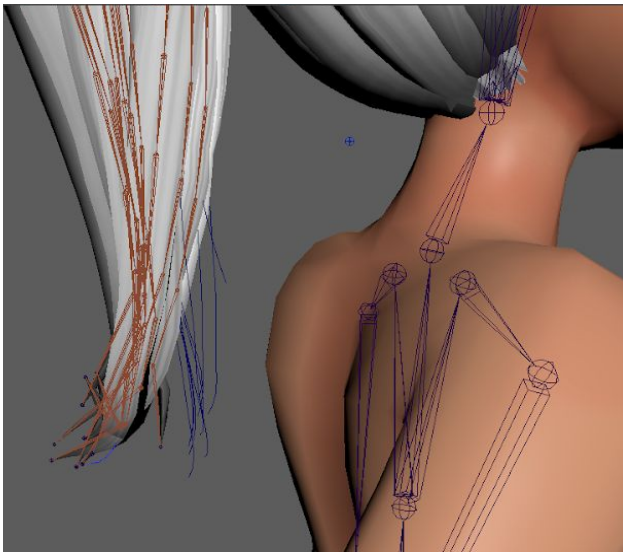


Fig. 50. Moving hair joints

As can be seen in the figure 49 follicles move, but the joints (in multicolor) won't move just yet. The next step is to transfer the information from the follicles to the joints.

For each initial IK curve, the information of its final eponymous curve, the System output curve, was applied, thus completing the process of transferring the dynamic curve to the chain of joints of each strand. This same process was repeated for all strands.

- **Bake and export**

When we had carried out the same process in the other two hairstyles of the project, related to the body rig, it was considered that the "complex rig" had already been completed and was then exported along with the hairstyles in FBX. When exporting, the bake animation box was checked from the second 0 until the end of the animation in fps, saving the new generated joints animation.

4.1.2.3 Cloth animation

This section details how the clothes were adapted to the body and with what method its animation is achieved. For tight clothing a skinning was applied as shown below and for clothes with more dynamism such as a skirt the animation was achieved using an ncloth simulation, which is more realistic than a skinning.

- **Skinning**

For the skirt skinning (As seen on Fig.51), it was selected as well as the complete body/hair rig and then were applied a "bind skin". After that, both the body (the source skin) and the skirt (destination skin) were selected in order to apply the "Copy skin weights" command (Skin >Copy skin weights at the Rigging panel).

Thus, it is guaranteed that clipping will be avoided as the body and clothes share the same movement information supplied by the skinning.

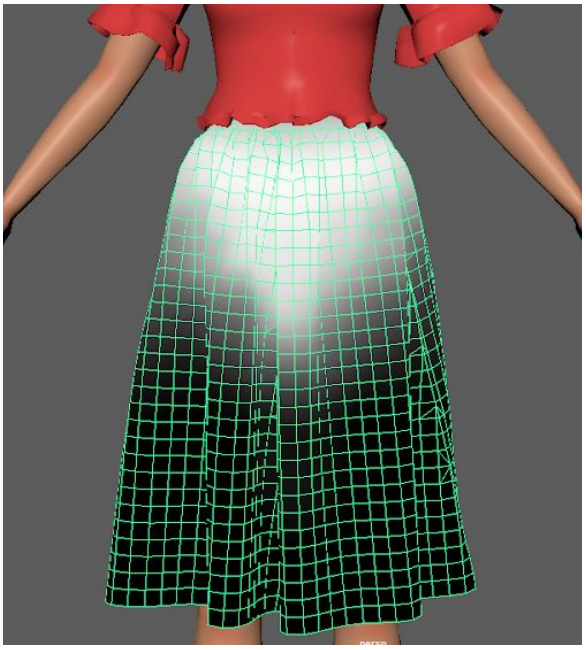


Fig 51. Skirt skinning

- **Simulation**

Fig 52. Skirt Collisions

The animation of the clothes could be left alone with the skin, but concerning the skirts it is more desirable to apply a simulation in order to improve realism.

For a cloth simulation, the skirt was selected and then "ncloth" was applied (At the FX planel, nCloth > Create ncloth).

As it is not desirable for the clothes to fall to the floor, some constraints (See Fig.54) were applied around the waist "Point to surface"(FX Panel, nConstraints>Point to Surface).

After that, the animation fluidity can be checked by pressing "Play forwards". If the result was not successful, the "nClothshape" parameters can be modified depending on the type of fabric.

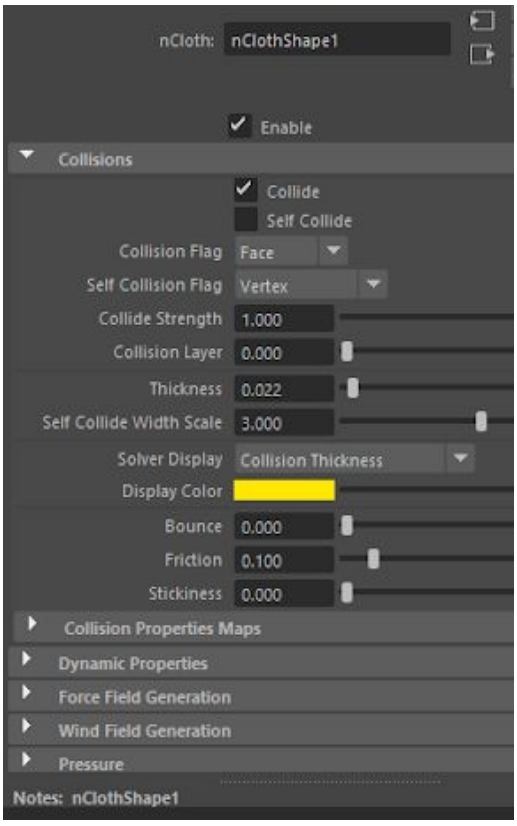


Fig.53 Parameters

Apart from the named parameters, the collision of the fabric with the skirt (As seen on Fig.52) is also editable. In this case a "Thickness" of 0.022 (See Fig.53) was assigned and it was verified that it was not clipping with the body.

For the simulation to consider a collisionable object not penetrable to the body, a "nrigid" was assigned to it.

In this way, when simulating the skirt it will collide with the body without crossing it, respecting the movement of the legs and arms without crossing them (See Fig.55).

Finally, for a more realistic simulation, the "self collide" option of the "nClothshape" was activated so that it does not intersect / clip with itself.

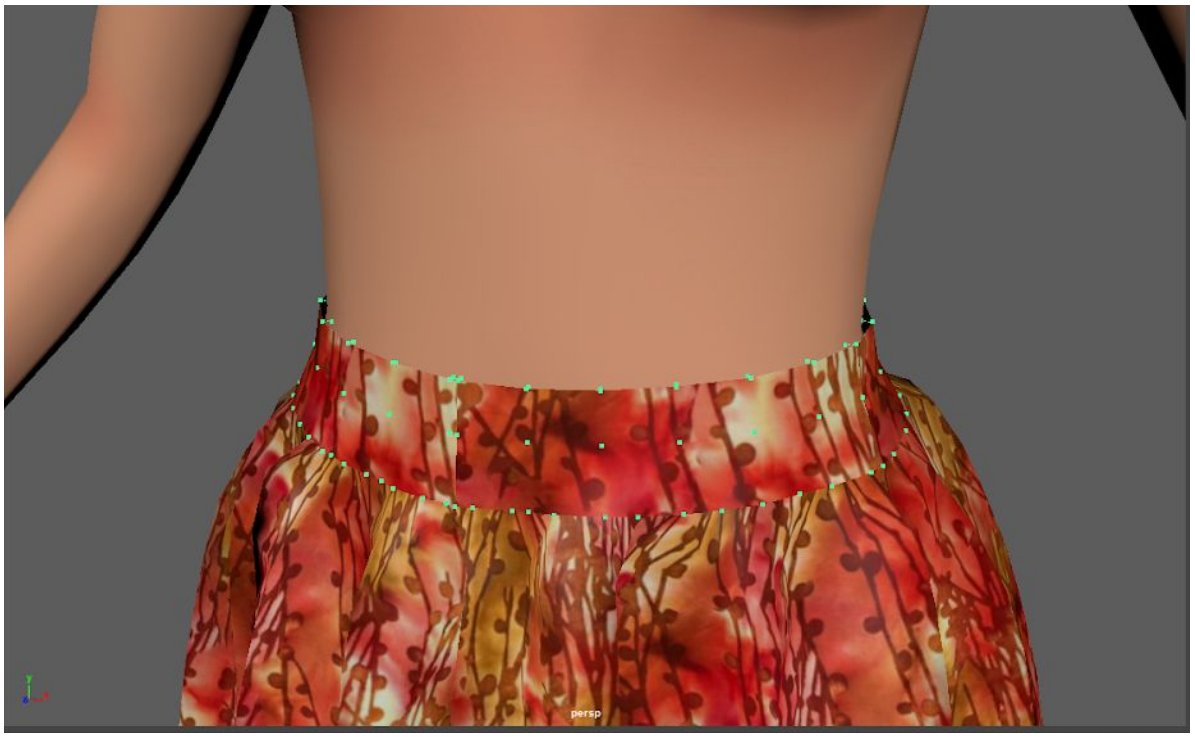


Fig. 54 nConstraints around waist

Fig. 55 Skirt Simulation

- **Cache and export**

To export the cloth simulation, as it is not possible to save it in FBX, and taking into account that Unity does not support Maya cache (mc), the alembic cache format was used. (FX Panel, cache>alembic Cache>Export selection to alembic).

This alembic file (.abc)[16] only saves information about a particular animation for each type of cloth. However, it is very useful to import cloth animations into Unity 3D.

At the export menu, the animation bake has to be set from the frame 0 to the end of the animation. "No normals" and "UV write" options have to be checked at the "Advanced options" as well.

4.1.3 Technical development

This project has three canvases that the user can interact with:

- ➢ Main menu Canvas
- ➢ Character editor Canvas
- ➢ Photocall canvas

The main functionalities of these canvases are as follow:

- **Camera System**

  Camera Rotation:

  The user can rotate the character around its Y axis at any moment.

  By clicking the left button and dragging left or right, the rotation is applied and it will follow the mouse's direction.

  Camera movement:

  Three different positions have been specified to implement the camera's movement: "MainMenuCameraPosition", "StandardCameraPosition" and "ZoomInFaceCameraPosition". The user can change its position between the main and the standard by pressing the zoom button.

  The "ZoomInFaceCameraPosition" is the one that can be seen on the main canvas. The camera will move from one position to the other. Unity coroutines have been used to perform this movement. This way, the transition between these positions is made smoothly, offering a more attractive and fluid experience.

- **Main menu Canvas**

This canvas is just an introduction to the game. The camera is focused on the character from a near distance. By pressing the X button, the player exits the game. By pressing the PLAY button, the player enters the edition canvas. As soon as the edition canvas is enabled, the camera leaves its "MainMenuCameraPosition" position and moves to the base position: "StandardCameraPosition", following the movement system detailed before.

Fig.56 Main menu Canvas

- **Character editor canvas**

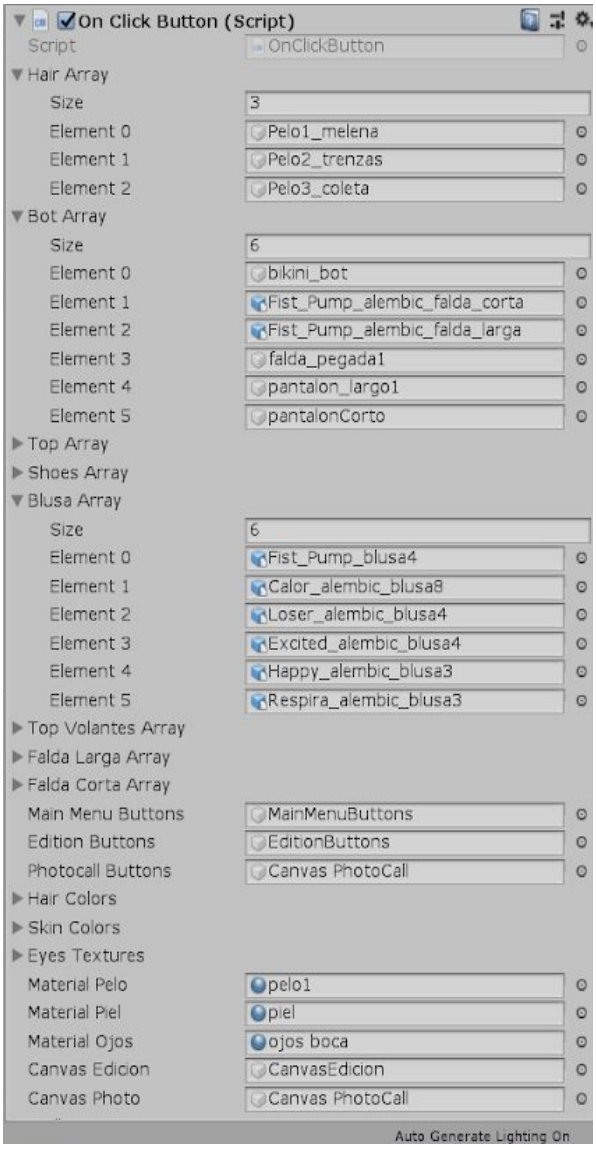


Fig.57 Unity inspector preview

All the functions involved in the character edition have been developed into one script: "OnClickButton". This way, it is easier to add more animations and complements by just modifying the arrays found on this script.

The buttons for this canvas (see Fig. 58) are as follow:

Color Edition buttons:

An array of color elements has been developed for this functionality. These colors are applied to the albedo attribute of the material that is assigned to the object.

Change animated elements buttons:

By pressing the clothes change or the hairstyle change buttons a new animation is triggered on the animation controller's animation loop. Anytime the user makes a new change, the current element is hidden and the next one is shown.

There are two types of clothes elements: Normal and special.

Normal clothes have a skinned mesh and special ones, alembic files (.abc).

Normal elements fit the body perfectly and have little impact on the engine's performance.

Alembic elements have a higher computational cost, but they offer a much more realistic animation. Anytime a cloth special element is selected, the script accesses its corresponding array and its special animation is played.

ZoomIn button:

By pressing the "Zoom in" button, the "cameraMovement" script is invoked thus the camera moves to the "ZoomInFaceCameraPosition" position. Pressing this button again takes the camera back to its standard position.

GoToPhoto button:

By pressing this button, the edition canvas is hidden and the photocall canvas and its wooden floor gameobject are displayed.

GoToMain button:

By pressing this button, the player is taken back to the main menu. The edition canvas is no longer displayed.

Fig.58 Character editor canvas

- **Photocall canvas**

Change Background button:

The user can change the background pattern by pressing the "change background" button (Fig. 59). This functionality is managed by an array of textures which is applied to the albedo attribute of the material assigned to the wall object. The textures used on this background are from "bgfonts"[17].

Snapshot button:

A snapshot can also be taken by pressing the "snapshot" button. This snapshot is saved in the same folder where the main application is located for the player to see it. A message is displayed whenever a new snapshot is taken.

Animation button:

By pressing the animation change button, a new animation is triggered on the animation controller. This way, the user can take a new snapshot while the character is posing with the selected elements.

Back to editor button:

By pressing this button, the photocall canvas is hidden and the character edition canvas is displayed.

Fig.59 Photocall canvas

- **Animator controller**

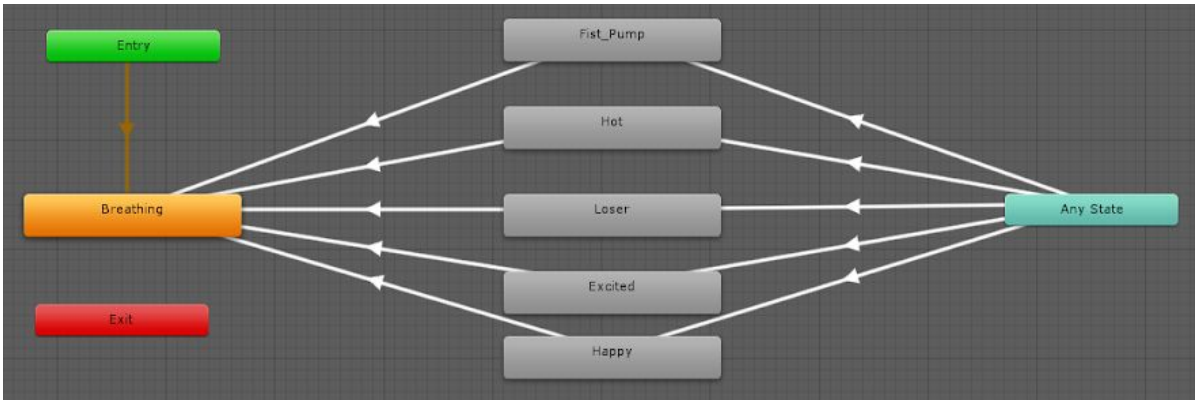


Fig.60 Animation Controller

The default animation that is always displayed is the idle: “breathing”. Whenever a button is clicked, the script manages the change that has been made by accessing to the corresponding array. The current element is then hidden and the next one is shown. The animation trigger is set to true. This way, the animation controller is able to detect that a change has been made and the next animation is played. Once the animation has finished, the animation trigger is set to false and the animation controller plays the “breathing” animation indefinitely until a new change is made.

After that, the next time the button is pressed, the following animation in the list will be played. This way, the character will always play a different animation. Animations load from frame 0 each time they are triggered. If the user changes one of the elements while an animation is being played, it will stop, causing the animation controller to start the next animation.

- **Alembic animations**

For the alembic animations performed by the simulated clothes, a Unity plugin was used to read the ".abc" format. Since it only saves information about the bake made with the clothing simulation, the animation of each cloth was treated as a special animation. In this case it does not depend on the rig (just as normal animations), so it is not managed along with the other animations that actually depend on it.

- **Project details**

Below are the details of the quantifiable elements:

Codelines	765
Scripts	3
Hair meshes	3
Top cloth meshes	6
Bot cloth meshes	6
Shoes meshes	3
Parts of body meshes	7
Special alembic meshes	24
Animations	6
Textures	30
Icons	20
Materials	15

4.2 Results

It is expected that this application achieves the following results.

- To make a complete character artist workflow (sculpted, retopology, uv, Hand painting textures, hair, cloth...)
- To create a stylized design for body, hair and clothes.
- Animations of clothes and hair.
- Introducing those animations in a Unity UI that allows the user to see them applied in clothes and hair.
- To allow the user to change hair and clothes of the character through buttons.
- Illumination and favorable environment for the cartoon immersion in the character editor.
- Improve the character editor to allow changing the color of the eyes, hair and skin.
- Secondary screen called "Photo Call" in which the user is able to capture a screenshot of the character, which will be posing.

Fig.61 Start Screen

As for the character workflow, It was found really interesting to work with a complete set of applications to design and implement the character. It has been a long way, but the character is, at its final stage, fully sculpted and animated. The stylized design suits perfectly the character and its level of detail is much better than expected.

The different animations that the character performs are fluid and dynamic and both cloth and hair react perfectly to them as if they were actually a part of the character.

One of the hardest challenges to overcome during the development of this project was to import and implement the animations and the main character into a HUD in Unity

and interact with it. Changing any part of the character starts a new animation every time and the button menu is perfectly integrated into the environment.

Fig.62 Character Editor canvas

The illumination system that has been chosen for this project fits perfectly with the character style.

Fig.63 Character Editor with character animation

Fig.64 Character Editor with character animation from behind

As seen on the figure 64, rotating the character is a functionality that has been successfully introduced. Just by dragging the mouse, the character reacts by rotating around its center.

Fig.65 Character Editor after applied changes

Changing the skin and hair color is really easy to do and the user experience is as simple as intuitive. Once the user is satisfied with the design, accessing the photo call screen is a click away.

Fig.66 Photo Call Screen Result

The Photo Call screen (As seen on Fig.66) is very easy to manage. The snapshot capability enables saving the current state of the character into the local storage so it can be used later.

Fig.67 Snapshot message

Overall, the final result has been successful and all the challenges that were found and later overcome during the process were worthy (As seen on Fig.67).

"This page has been left intentionally in blank".

5. Final planning

As a result of the accused learning curve and a long workflow, the tasks have doubled the time planned for each of them. However, it was taken into account from the beginning. This is why the planning ended a month before the final deadline.

The project must be planned so that its execution is around 300h but, in the end, it has been necessary more than 600h for its complete realization. The month of June did not enter the planning, but it was necessary to write the entire memory and fix all possible bugs.

So that only 300h had been dedicated, one could have considered focusing the animation only on hair or only on clothes. By covering both aspects and having to deal with new problems and failures, the time of completion and that of research were doubled.

At the beginning of the planning it was suggested that 5 hours a day should be dedicated from Monday to Friday and that during the festive weeks the project would not be advanced. However, when the development of the project reached its halfway, There were spent 8 hours a day and the project was being worked upon on weekends and holidays.

Thanks to the hard work and to a new plan, the deadline has been successfully reached.

"This page has been left intentionally in blank".

6. Conclusions and future work

6.1 Conclusions

Despite the complexity of the project and the difficulties encountered in the process, it is considered that the work performed meets the expectations of a professional finish.

We have managed to complete the complete workflow of a character for video games from its modeling stage to user interaction in a HUD developed in a video game engine.

It has also been able to animate hairstyles and clothes fluently (taking into account the duration of the project).

Changes of clothes and hairstyle driven by the UI works perfectly.

The project displays stylized and beautiful aesthetics.

For this project, it has been possible to manage a vast number of tools necessary for the professional video game developer in a record time and dealing successfully with all the setbacks that this entails.

Although there have been used more hours than originally planned in quality of learning and resolution of errors, valuable knowledge has been obtained and enjoyed throughout the process.

In general terms the result has been better than expected and reflects the large amount of work and effort invested.

6.2 Future work

This project is easily scalable and improvable. The improvements that for lack of time did not enter the planning of the project are detailed below:

- In the future, it will be desirable to expand the content of the project in terms of the number of clothing and hair elements, as well as extra accessories.
- The capability to choose clothing textures would also be very attractive and improving the UI icons would give the project a more professional finish.
- Some buttons will have added more audio elements and effects.
- A zoom button will also be added to take a closer look at the character details.
- It is also intended to enable the option to change light exposure and export the FBX model with its final design.
- As the last modification, it would be desirable to adapt what has been learned throughout this project to a male character by adapting clothes and hairstyles to his gender.

"This page has been left intentionally in blank".

7. Bibliography

In this chapter the Google Drive repository and all the links of interest will be listed.

7.1 Repository

Github has a strict limitation for files over 100MB. Therefore, the project has been uploaded to Google Drive as some animations exceed that file size. Here is the URL:

- https://drive.google.com/drive/folders/1yoxRG8cW8syrR_4dYYEIq-ROINUbNW3K?usp=sharing

7.2 Links

[1] Reference game: http://www.juegos.com/juego/cute-moe-3d-dress-up

[2] Photoshop official website: https://www.adobe.com/es/

[3] Game-icon website: https://game-icons.net/

[4] Zbrush official website: http://pixologic.com/

[5] Maya Software website 2019: https://www.autodesk.com/products/maya/overview

[6] Vroid Studio official website: https://studio.vroid.com/

[7] Blender 2.8 official website: https://builder.blender.org/download/

[8] Substance Painter 2019 official website: https://www.substance3d.com/

[9] Marvelous Designer 2018 official website: https://www.marvelousdesigner.com/

[10] Adobe Mixamo official website: https://www.mixamo.com/#/

[11] Unity 3D 2019.1.2f1 official website: https://unity.com/

[12] Visual Studio Code official website: https://code.visualstudio.com/

[13] Google Docs official website: https://www.google.es/intl/es/docs/about/

[14] Google Drive official website: https://www.google.com/intl/es_ALL/drive/

[15] Meeting point for Spanish developers: https://www.stratos-ad.com/

[16] Alembic official website: https://www.alembic.io/

[17] Free textures website: http://bgfons.com/