APLICACIÓ WEB PER A LA

CREACIÓ DE PERSONATGES 3D

Treball Fi de Grau de

Javier Portabella Clariana

Director: Eva Valls Garolera

Grau en Enginyeria en Sistemes Audiovisuals

Curs 2024-2025



*[Pàgina en blanc]*

*[Tingueu en compte:*

*(i) No s’han de deixar pàgines blanques al cos del treball (índex fins la conclusió).*

*(ii) Utilitzar interlineat senzill i tipus de lletra Times New Roman.*

*(iii) Per a aquesta secció principal (cos del treball), la restricció del límit de pàgines és de 30 pàgines + 5% o 10% de marge.*

*(iv) La pàgina del final de capítol compta com a mitja pàgina.*

*(v) Podeu afegir un annex d'informació de suport a l’apèndix per incloure totes les dades/informació que necessiteu mostrar. El apèndix no té un límit de pàgines. Si tens informació a l’apèndix, has de fer referència en el text principal a la informació continguda en l’apèndix seguint aquest exemple:*

*document principal: “La taxa de natalitat als països europeus està disminuint molt ràpidament (vegeu Apèndix-1). Això vol dir que tot i que la gent viu més temps (figura Apèndix-1), amb el temps la mida de la població disminuirà".*

*(vi) Les referències s'inclouen a la secció de Bibliografia i s'exclouen de la limitació del nombre de pàgines. ]*

Dedicatòria [opcional] [mida 12]

Lorem ipsum

**Agraïments**

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

**Resum[mida 14]**

Resum del treball en la llengua que es presenti [mida 11]

Extensió màxima recomanada: 150 paraules.

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.

**Resum [en una 2a llengua. Ex. Resumen][mida 14]**

Resum del treball en una llengua diferent a la utilitzada [mida 11]

Extensió màxima recomanada: 150 paraules

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.

**Resum [en una 3a llengua. Ex. Abstract][mida 14]**

Resum del treball en una llengua diferent a la utilitzada [mida 11]

Extensió màxima recomanada: 150 paraules

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint obcaecat cupiditat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua.

**Contents**

[Introduction 1](#_Toc198829645)

[1.1 Motivation 1](#_Toc198829646)

[1.2 Objectives 2](#_Toc198829647)

[Theoretical Foundations 3](#_Toc198829648)

[2.1 Fundamentals of 3D Modeling 3](#_Toc198829649)

[Meshes 3](#_Toc198829650)

[Materials 4](#_Toc198829651)

[Polygons and Vertices 4](#_Toc198829652)

[2.2 Morph Targets and Texture Mapping in 3D 5](#_Toc198829653)

[Morphing 5](#_Toc198829654)

[Morph Targets 5](#_Toc198829655)

[UV Mapping 6](#_Toc198829656)

[2.3 Principles of User Interface Design for 3D Applications 6](#_Toc198829657)

[2.4 Overview of Relevant Technologies 7](#_Toc198829658)

[Three.js 7](#_Toc198829659)

[Blender 7](#_Toc198829660)

[Lexgui.js 7](#_Toc198829661)

[State of the art 8](#_Toc198829662)

[3.1 Overview of Web-based Applications for 3D Modeling 8](#_Toc198829663)

[3.2 Review of existing 3D Character Creators 9](#_Toc198829664)

[MetaHuman Creator 10](#_Toc198829665)

[Ready Player Me (RPM) 10](#_Toc198829666)

[3.3 Other Avatar Customization Areas: Social Media and Video Games 11](#_Toc198829667)

[3.4 Comparison of RPM, MetaHuman Creator and others 11](#_Toc198829668)

[3.5 Morphing Techniques 13](#_Toc198829669)

[Conclusions i treball futur[mida 20] 14](#_Toc198829670)

**Llista de figures [opcional] [mida 14]**

*[Fes la llista amb la mateixa estructura mostrada a l’índex, amb mida 11. Les figures i gràfics al cos del treball es titulen a la part d’abaix, amb mida 10]*

**Llista de taules [opcional] [mida 14]**

*[Fes la llista amb la mateixa estructura mostrada a l’índex, amb mida 11. Les taules al cos del treball es titulen a la part d’adalt, amb mida 10]*

**Chapter 1**

# Introduction

The fast growth of immersive technologies and interactive media has led to more and more users becoming interested in the creation of personalized and realistic three-dimensional characters. The customization of 3D avatars is becoming essential in some fields like gaming, virtual reality (VR), social platforms, and more. This demand was what made me innovate to make it more accessible with a web interface.

This project focuses precisely on this. It takes advantage of the innovation opportunity presented and develops a web application for creating customizable 3D characters. It will extend the functionality of the API provided by **Ready Player Me (RPM)**[1]. RPM is a platform where each user can create its own avatar by choosing between prebuilt assets such as 3D face and body meshes and modifying them in infinite ways, enabling developers to integrate avatar creation into various platforms. However, its customization capabilities are limited, particularly for users seeking advanced features like detailed facial modifications, complex aging effects, or some hair personalization.

To overcome these improvements, this project will design a user-friendly web interface that allows users to interactively customize avatars with intuitive controls. Also, it will introduce new customization options, such as facial feature interpolation, wrinkle texture application for aging effects, and recoloring options.

It also takes advantage of the final degree project done by Oriol Jiménez Ayguadé, a graduate in computer engineering. Our projects have similar approaches and goals, so I have been able to pick up where he left off.

The project's implementation will focus on two core areas:

1. **User interface development**: making sure the user interaction is an accessible, fluid, and intuitive customization process.
2. **Algorithmic innovation**: improving the interpolation of morph targets, application of new textures and modification of the preset ones.

## 1.1 Motivation

When the opportunity came up to focus my final degree project on 3D avatar customization, one of the first things that came to mind was the film *Ready Player One*[2]. This movie left a lasting impression on me from the moment I watched it upon its release in 2018. The story follows Wade Owen Watts, a young gamer from the year 2045 who, like the rest of humanity, finds solace in the OASIS virtual reality metaverse, preferring it over an increasingly bleak real world. In OASIS, everyone can have an avatar that represents them exactly how they wish to be seen.

The idea of a completely customizable virtual identity was what most impressed me about Ready Player One. Although the movie is, at its core, a critique of society, I was captivated by the idea that everyone could have a personalized virtual version of themselves, shaped entirely by their own choices and imagination. This notion of freedom through the customization and design of the characters, where the only impediment is our imagination, is what inspires the vision for this project. The goal is to bring a slice of this fictional concept into reality by granting users this freedom to design their own 3D avatars.

## 1.2 Objectives

The main goal of this project is to build a web application that lets users design, customize, and download their own 3D avatars. I have set out a series of particular goals to facilitate this transition:

* **New customization options**: The app will have features beyond what the Ready Player Me (RPM) interface offers. Users will be able to interpolate between facial traits from several avatars to produce a new one, add ageing effects like wrinkles or grey hair and adjust other characteristics such skin tone, eye color, clothing, and more.
* **User-friendly design**: A key aspect of this project is making a simple and intuitive interface. So that, even those with no experience in 3D modeling, can easily navigate the application while keeping it accessible and straightforward.
* **Avatar export options**: Users will have the possibility to download their final creations in formats that work with other platforms, enhancing the app's utility and versatility.
* **Improved performance and accessibility**: The platform will be web-based and lightweight, unlike heavier tools like Metahuman Creator, ensuring a smooth experience even on less powerful devices.
* **Diverse representation**: greater inclusivity is sought, offering customization options with a variety of skin tones, facial structures, genders, and styles.

With these objectives in mind, the project aims to blend customization creativity with technical ease, providing a practical tool that makes avatar customization more accessible to global users.

**Chapter 2**

# Theoretical Foundations

This section focuses on providing the theoretical foundations of the work. It is about explaining the fundamental concepts, basic technologies, and scientific or technical principles that will be used in the project. Here, the essential information that serves as a foundation for understanding the problem and the available tools is introduced.

## 2.1 Fundamentals of 3D Modeling

The process of 3D modeling consists of the creation of three-dimensional representations of objects using specialized computer software. In traditional modeling, 3D objects are created using surface representations such as polygonal meshes, NURBS patches[[1]](#footnote-1) or subdivision surfaces.

### Meshes

A mesh is a collection of vertices, edges, and faces that define the shape of a 3D object. The vertices are the most basic elements of a mesh from which we can obtain the others. It is a set of single points or positions in 3D space that are stored in arrays of coordinates, each array generally representing an object.

Secondly, there are the edges, which are the straight lines connecting two vertices. They are usually invisible on the rendered image, however, in the wireframe view, the edges are seen as “wires” connecting dots and creating the mesh.

Finally, the faces are the flat surfaces enclosed by edges, typically forming triangles, quadrangles or n-gons. They are what is seen when rendering the mesh. If an area between n vertices does not contain a face, it will simply be transparent or nonexistent in the rendered image.

Un dibujo de una persona con un paraguas

El contenido generado por IA puede ser incorrecto.

Figure 2.1: Representation of vertices, edges and faces of a cube.

In order to help shade a mesh (among other things), it is very common to use normals. These elements are defined as the perpendicular direction to a face (usually), a line, a tangent line for a point on a curve, or a tangent plane for a point on a surface. There are two ways of shading normals, in a smooth or flat way.

When a mesh uses smooth shading, normals are interpolated across the vertices of the mesh. This creates seamless transitions between adjacent polygons, producing a more natural and realistic appearance.

On the other hand, flat shading applies uniform lighting across each face of a mesh, which is ideal for models with distinct flat surfaces, such as cubes or pyramids.

Forma, Círculo

El contenido generado por IA puede ser incorrecto.

Figure 2.2: Smooth vs flat shading.

### Materials

In computer graphics, materials define how a 3D object's surface interacts with light, determining its visual properties like color, shininess, transparency, and texture. While the mesh provides the object's shape, the material controls whether it appears matte, glossy, metallic, or translucent by simulating real-world physics (e.g., reflections, roughness, or light scattering).

Textures, which are images mapped onto materials, add fine details like scratches or patterns, but the material itself dictates how those textures respond to lighting. Together, materials and textures transform basic geometry into realistic or stylized surfaces, from dull stone to reflective metal, enabling lifelike or artistic visuals in 3D Modeling. Some examples of these types of materials include those shown at Figure 2.3:

Imagen que contiene tabla, plato, presentado, hecho

El contenido generado por IA puede ser incorrecto.

Figure 2.3: Sphere with different textures and materials.

### Polygons and Vertices

Polygons are the fundamental building blocks of 3D models, forming the surfaces that define an object’s shape. A polygon is a flat, two-dimensional shape composed of straight edges (edges) and corners (vertices). The simplest polygon is a triangle, consisting of three vertices and three edges. More complex polygons, such as quadrangles (quads) or n-gons (polygons with more than four sides), can also be used, but triangles are the most common due to their stability in rendering and computational efficiency.

Vertices are the most basic elements in a 3D mesh. Each vertex represents a single point in 3D space, defined by its coordinates (X, Y, Z). These points serve as the foundation for constructing edges and faces, ultimately shaping the model.

The number of polygons in a model (polycount) affects both performance and visual quality:

* Low-poly models: Use fewer polygons, making them efficient for real-time applications (e.g., mobile games, VR).
* High-poly models: Contain dense geometry for fine details, often used in pre-rendered scenes (e.g., films, high-end visualizations). A well-optimized model balances detail and performance by strategically distributing polygons where needed (e.g., more on facial features, fewer on flat surfaces).

## 2.2 Morph Targets and Texture Mapping in 3D

This section explores two essential techniques in 3D modeling: morph targets (also known as blend shapes) and texture mapping. Morph targets enable smooth deformations of a mesh, while texture mapping enhances surface details without increasing geometric complexity.

### Morphing

Morphing is a 3D animation technique that smoothly transitions between different shapes of a model by directly manipulating its vertices—unlike skeletal animation, which uses a bone rig. It's especially useful for subtle effects like facial expressions, offering fine control over individual features.

The process involves storing multiple versions (morph targets) of a base mesh, each representing a specific deformation—like a smile or frown. Animators blend these shapes using weight values to create smooth, expressive movements. For example, increasing the weight of a "smile" target transitions a neutral face into a grin.

In this project, morphing is applied not between different versions of the same base mesh, but between corresponding parts of two distinct 3D models. The weight values of the targets are determined by the position of a slider in the interface.

### Morph Targets

Also known as blend shapes, morph targets are alternate versions of a base mesh used to create animations by blending deformations. They're crucial for character animation, especially in rendering realistic facial expressions.

Artists sculpt each target mesh to represent specific expressions. However, in this project, when talking about morph targets, we refer to the different noses, eyes, ears, etc; of the characters. These can then be blended in real time, e.g. combining 70% "nose1" with 30% "nose2" to form a unique look.

Imagen de la pantalla de un computador

El contenido generado por IA puede ser incorrecto.Pantalla de computadora con una imagen de una caricatura de una persona

El contenido generado por IA puede ser incorrecto.Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

*Figure 2.4: Two morph targets combined (first and second pictures) to obtain the third.*

All morph targets must share the same vertex count to blend correctly. If this were not fulfilled, we would have problems with the correspondence of vertices. Despite limitations, morph targets are essential for achieving lifelike animation in games and films.

### UV Mapping

UV mapping is the process of deploying a 3D mesh onto a 2D surface, called a UV map, which allows textures to be assigned to points on the model with precision. Each vertex of the model corresponds to a UV coordinate that indicates which part of the image (or texture) is projected onto its surface.

This system has been especially useful in my project for applying masks and texture changes, as well as color filters. UV mapping allows for precise adjustments in texture properties, making it easier to apply and modify color filters.

## 2.3 Principles of User Interface Design for 3D Applications

A well-designed interface not only needs to be visually clear but also allow users to interact with 3D elements in an intuitive and efficient way.

One of the main principles is **simplicity**. Even though the code of the interface may be very complex, the platform should be presented in an accessible way to the users. To accomplish this, elements should offer visual feedback and have logical interaction. In this project a good option for this could be a slider or a Voronoi Cell[[2]](#footnote-2) panel, because they are functional and are easy to understand. These controls must also be clearly labeled and put together according to the part of the face or body they affect.

**Real-time feedback** is another essential aspect. Any change users make (like clicking in a recoloring button to change the eyes color) should be instantly visible, to help them understand the effect of each interaction. To implement this, we will get helped by the Three.js library, which makes it possible to implement real-time rendering directly in the browser while interacting with the server.

Interaction techniques such as zooming, rotating and translating with respect to the model are essential to be aware of what we are editing and how we are influencing the scene. These functions should be implemented through intuitive gestures (touchscreen swipes with both fingers, scrolling, separating them, etc.) and supported by visual cues.

Following the study made by Lee et al. (2009) on usability principles and best practices for the user interface design of complex 3D parametric tools [5], we can understand the importance of minimizing cognitive load, organizing control groups logically, and (as mentioned above) ensuring immediate feedback from the system. While their work focuses on architectural applications, many of the principles identified are directly applicable to an avatar customization interface, where the balance between complexity and usability is critical to achieving user goals.

## 2.4 Overview of Relevant Technologies

In order to implement interactive 3D applications in web environments, first we need to go through the technologies that support rendering and real-time interaction. Web-based 3D graphics rely mainly on WebGL[[3]](#footnote-3), however, using it directly can be complex and time-consuming. For this reason, high-level libraries such as Three.js or Babylon.js have emerged to simplify the process of building and managing 3D scenes. This section provides an overview of how these technologies enable user-friendly and interactive avatar customization interfaces.

### Three.js

Three.js [9] is a high-level JavaScript library designed to simplify the development of 3D graphics in the browser. Through a well-structured API, developers can create complex 3D scenes that include objects, lights, cameras, animations, and interactions; which are essential components for applications such as avatar customization and facial morphing.

As said above, WebGL is the underlying technology used to render 3D graphics in the browser, but it requires a deep understanding of graphics programming and a significant amount of code to produce even simple scenes. Three.js acts as an abstraction layer over WebGL, saving developers the work of writing complex code and allowing them to focus on design and interaction logic.

Compared to other libraries like Babylon.js, which is more focused on high-performance game development, Three.js offers a simpler API that is ideal for web-based tools requiring high-quality visuals and usability, as discussed in the previous sections.

### Blender

Blender [10] is an open-source 3D creation suite. It supports the entirety of the 3D pipeline: modeling, animation, rendering, etc. In this project, it has been used to identify the exact vertex indices needed to perform interpolation between morph targets accurately.

### Lexgui.js

LexGUI [11] is a lightweight graphical user interface library used for real-time control of parameters in web-based 3D applications. In this project, it has been integrated to allow users to interact with a 2D control map (Map2D), where we can add characters in the edges and divide the space like a simplified Voronoi diagram. By moving the control point across the map, the system interpolates between multiple avatars using weighted morph targets, enabling smooth transitions and hybrid faces.

**Chapter 3**

# State of the Art

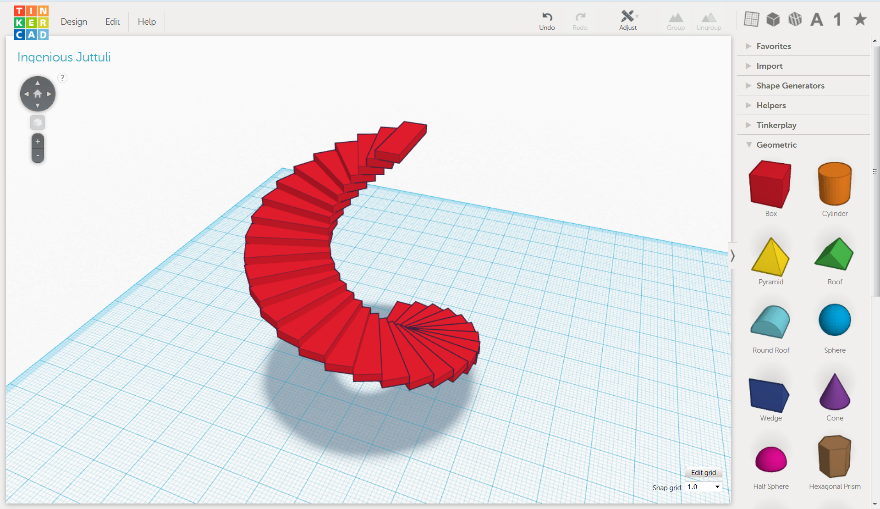
This section explores current tools and technologies for 3D avatar creation, focusing on web-based solutions. There is a comparison of major platforms in terms of visual quality, customization flexibility, and accessibility. Special attention is given to MetaHuman Creator [12] and Ready Player Me [1]. This review shows the new features my project aims to address with enhanced in-browser editing and interpolation.

## 3.1 Overview of Web-based Applications for 3D Modeling

Web-based 3D modeling tools have matured significantly in recent years. Early attempts like Clara.io [13] demonstrated that full-featured 3D design and rendering could run entirely in a browser. Today a range of online tools exist, from simple educational editors to complex sculpting suites.

However, these general-purpose modelers typically lack the specialized controls needed for intuitive avatar creation. For example, TinkerCAD [14] is a beginner-oriented web app that lets users build models out of primitive shapes. It is highly restricted to basic geometry and cannot produce organic forms: as one user notes, it is “limited to simple geometric shapes” and cannot create a convincing human face. TinkerCAD also offers no surface texturing or UV mapping – colors can only be applied to whole blocks, not detailed skin or wrinkles.

* **TinkerCAD** (Autodesk): An easy drag-and-drop editor for beginners. It excels at blocky, mechanical models (using boxes, cylinders, etc.), but *cannot* sculpt natural forms or fine features. It has no UV textures or material maps, only solid colors. In practice this means TinkerCAD can’t create a realistic head or facial details, making it unsuitable for avatar customization beyond toy-like figures.



*Figure 3.1: TinkerCAD interface.*

* **SculptGL** (Stéphane Ginier) [15]: A web-based sculpting application offering advanced brushes and dynamic topology. SculptGL provides many modeling tools (brush, inflate, smooth, grab, twist, etc.) and even vertex painting (PBR color/roughness), allowing freeform organic modeling in the browser. In contrast to TinkerCAD’s block approach, SculptGL can carve and refine complex shapes. However, it’s essentially a digital analog sculpting studio (like a lightweight Blender sculpt mode), so it relies on the user’s skill; there are no high-level “nose slider” or “face width slider” controls. In short, SculptGL can form faces, but only through manual sculpting rather than intuitive parameter tweaking.

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

*Figure 3.2: SculptGL interface.*

* **Clara.io** (Exocortex): A once-promising professional website for modeling, animation and rendering. It offered a full desktop-like interface in-browser, as seen in Figure 3.1, with features like sub-object mesh editing, skeletal animation, and integrated V-Ray rendering. Clara.io could import/export many formats and even allowed photo-realistic cloud rendering. However, it was not specialized for character creation, and being very powerful made it complex and resource-hungry. (In fact, Clara.io was shut down in 2022, partly due to the limitations of running such heavy tools in a browser). Its general modeling workflow is far more complex than the slider-based controls our project envisions.

Pantalla de juego de computadora

El contenido generado por IA puede ser incorrecto.

*Figure 3.3: Clara.io interface.*

These tools illustrate the trade-offs: TinkerCAD is simple but too rudimentary for realistic faces, SculptGL is flexible but demands artistic effort, and Clara.io was powerful but generic and complicated. None provide an intuitive avatar customization interface. In contrast, this project’s browser is designed specifically to let users **directly manipulate character features** (e.g. nose shape, eyes color) with smooth interpolation and texture controls, something the above tools do not offer.

## 3.2 Review of existing 3D Character Creators

Beyond general modelers, there are tools dedicated to building 3D characters. These range from professional desktop suites (e.g. Reallusion’s Character Creator or Blender’s character plugins) to online or app-based avatar makers. Many game studios have character pipelines (e.g. Adobe Fuse, Mixamo) and hobbyist tools (DAZ3D, VRoid) as well. For this study, I am focusing on leading web-enabled or easy-access creators that resemble the goals of this project. In particular, MetaHuman Creator and Ready Player Me stand out as prominent examples, offering user-friendly workflows on different ends of the realism spectrum. We discuss each in detail below.

### MetaHuman Creator

MetaHuman Creator (by Epic Games) is a cloud-based character design app aimed at photorealism. It advertises itself as letting anyone “quickly and intuitively create photorealistic digital humans” in minutes. Using a web browser, users select from high-resolution scanned “preset” faces and bodies, then adjust sliders for body and face shapes or blend features together. This blending can be done with a Voronoi Cell as we can see in Figure 3.4. The system guarantees all results are “physically plausible” (based on real human scans), so the output looks remarkably lifelike. Characters come fully rigged with realistic skin, hair, eyes, teeth, and clothing, and even include eight levels of detail for game performance.



*Figure 3.4: MetaHuman interface, with the Voronoi Cell to interpolate.*

Because it is an Epic tool, the output is tightly integrated with Unreal Engine – designers download their MetaHuman via Quixel Bridge[[4]](#footnote-4). The interface is graphical and interactive, with blend shape previews and a library of hairstyles, uniforms, etc. Epic emphasizes ease of use: “MetaHuman Creator is a cloud-based app to make MetaHumans in minutes starting from [a preset]”.

**Limitations:** Despite its power, MetaHuman has key constraints. It is *locked into realistic humans*: any feature adjustments are constrained by the scanned data. As the MetaHuman docs note, only “physically plausible adjustments” can be made, meaning you can’t, for example, enlarge the eyes like a cartoon or add alien features without external editing. In practice, artists workaround this by exporting to Blender/Unreal and manually sculpting (Epic even provides a “Mesh to MetaHuman” plugin to convert a custom mesh, but results depend on the input).

Another practical issue is performance: because the interface streams complex 3D assets from Epic’s servers, users have reported it can feel slow or laggy – “painfully slow” to download characters, in one user’s words. Finally, MetaHuman is not freely usable in all engines – it is geared to Unreal, so integrating it elsewhere can be complex. In summary, MetaHuman offers unmatched realism and detail, but at the cost of requiring the Unreal ecosystem, broad scans-only presets, and a more heavyweight cloud workflow. It has less flexibility for stylized or extreme customization and is not browser-native (you use a browser interface but need Unreal to animate or export the result).

### Ready Player Me (RPM)

Ready Player Me is a web-based avatar creation platform focused on cross-application use (especially social/VR/AR). Users typically start in a web view (on PC or mobile), where they can either upload a selfie or start from a preset to generate an avatar. The interface then offers hundreds of customization options: from hair, clothing, accessories to body proportions and color schemes.

Pantalla de video juego de una persona

El contenido generado por IA puede ser incorrecto.

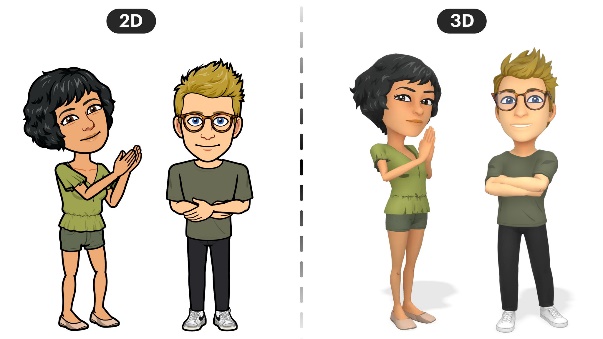
*Figure 3.5: Ready Player Me interface.*

Once finished, RPM gives the option to download the avatar as a fully skinned, rigged 3D model (in GLB or GLTF format) ready to plug into games or apps. It supports both full-body avatars (entire body with legs) and half-body (upper torso and head) versions, optimized for real-time performance. Developers can integrate RPM via SDKs[[5]](#footnote-5) (Unity, Unreal, React, etc.) or simply by loading the GLB file from the provided URL. Many games use RPM for user avatars, thanks to this ease of use and the cross-platform asset pipeline.

**Limitations:** RPM’s design prioritizes convenience and broad compatibility over fine detail. The visual style is stylized/cartoony, not photorealistic (in line with contemporary games like Fortnite[[6]](#footnote-6)). Facial customization is limited to choosing pre-made head types or adjusting overall proportions; there are no sliders for individual nose or mouth shapes. Internally, RPM avatars do have blend shapes (for example, the SDK uses predefined “EyeBlinkLeft”/“EyeBlinkRight” morphs to animate blinking), but users cannot create new face morphs on the fly. Texture changes are likewise pre-set (skin tone options, clothing textures) rather than user-painted. In short, while RPM provides a great breadth of outfits and hairstyle assets with easy export, it offers less flexibility over facial details and no dynamic aging or fine-grained feature interpolation. It trades fine expressiveness for speed and platform reach.

## 3.3 Other Avatar Customization Areas: Social Media and Video Games

Many social apps and games include simple avatar systems as well. These are generally aimed at casual audiences and offer very limited customization compared to MetaHuman or RPM. For example:

* **Bitmoji (Snapchat):** it was an expressive 2D cartoon avatar system until the company changed the style of the avatars to 3D ones in July 2023. Users pick from a set of face shapes, eye styles, and outfits to create a personal emoji-like avatar.

It's optimized for messaging and augmented reality (AR) filters, not 3D modeling; the character's appearance is more cartoonish than realistic. The control depth is modest (hair, clothes, facial features from preset libraries) and there is no true 3D sculpting.

Figure 3.6: 2D and 3D Bitmojis comparison.

* **Meta Avatars (Facebook/Meta):** Personal avatars for Meta’s social and VR platforms. Similarly to the last Bitmojis, these use a smooth style and are designed for friendly social interactions (reacting to messages, profiles). Users can adjust facial features (eyes, nose, etc.), hair, and clothing through a simplified UI, but all options are predefined. The focus is on diversity and expression in social media contexts rather than fine detail.

Figure 3.7: Group of Meta avatars.

\_

* **Nintendo Mii:** An avatar system that has become a classic due to its use on the Wii and Switch consoles. Mii Maker allows players to customize very basic facial characteristics: choosing from predefined face shapes, hairstyles, and accessory templates. Users navigate through options like “face”, “eyebrows,” “nose,” “facial hair,” etc., but each choice snaps to a canned graphic. The result is a blocky, low-detail character. \_

Figure 3.8: Nintendo Mii face editor.

These examples illustrate that mainstream consumer avatars are typically **simplistic and stylized**, designed for ease of use on phones or consoles. They lack the high polygon counts, realistic shaders, or procedural morphs of tools like MetaHuman. Their target audiences (kids, social users) prefer a “fun” look and rapid creation. In comparison, MetaHuman and RPM target more serious creation (games/VR/film), and offer much deeper visual fidelity. Bitmoji and Mii are very approachable but limited in feature depth.

## 3.4 Comparison of RPM, MetaHuman Creator and others

The table below summarizes key differences between these character systems, highlighting where our web-based avatar project fits in:

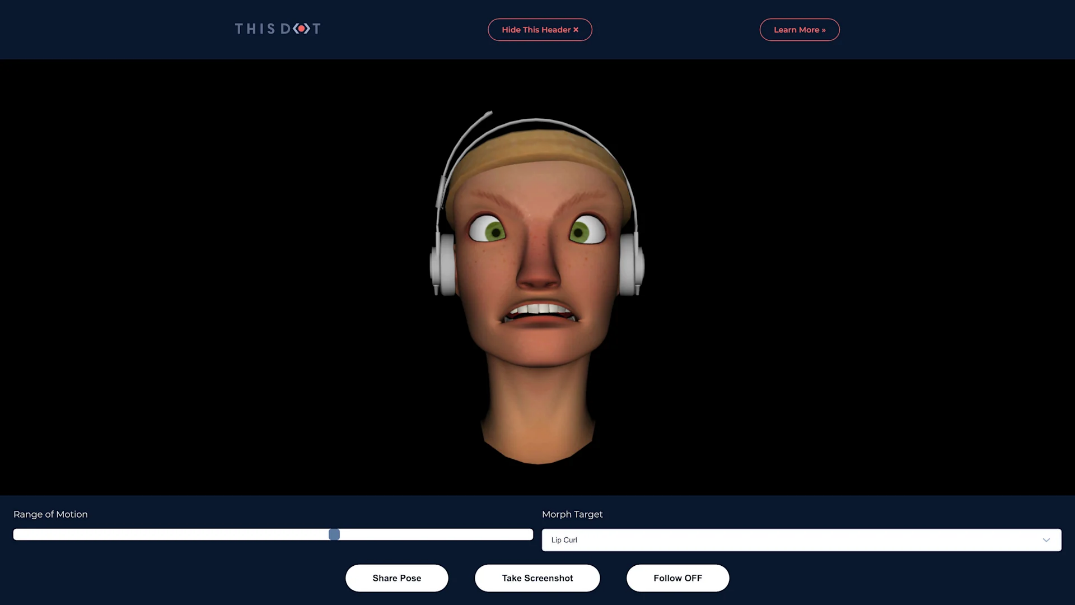
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Metahuman | Ready pLayer Me | Social Media Avatars | this project |
| Platform & Integration | Cloud service tied to Epic’s ecosystem (requires an Unreal Engine pipeline). | Web-native and offers SDKs, with output as a standard GLB model | Run entirely on mobile or console apps. | Purely browser-based (no engine required) and can integrate with any WebGL environment. |
| Visual Realism | Photorealistic humans (scanned skin, hair, detailed expressions). | Stylized 3D avatar (between cartoons and photo-like). Essentially the “Fortnite” style of human. | Cartoonish 2D/low-poly, focusing on clear identity rather than detail. | Middle-ground realism: higher fidelity than cartoons, but achievable in real time, with the ability to vary style by user. |
| Customization Controls | Many sliders for facial and body shape, but all are constrained by its scan-based presets. It does not expose raw vertex editing or texture painting. | Allows users to pick from vast libraries of hair, clothes, and some face presets, but no custom morph targets. | Only a handful of toggles (eye shape, hair style, etc. from menus). | Real-time interpolation sliders and texture maps (sliding between “wide nose” and “narrow nose” meshes or dialing wrinkle intensity on skin) |
| Export & Use | Fully rigged but primarily intended for Unreal Engine (export via Quixel/UE Bridge). | RPM avatars are optimized for game engines (GLB with skeleton) and can be easily imported into Unity/UE. | Bitmoji avatars are 2D images/GIFs, and Mii are console-specific assets. | Our avatars will be standard WebGL models (e.g. glTF) that can be exported for use in other 3D environments as well. |
| Accessibility & Audience | MetaHuman (free but tied to Epic) is aimed at professional developers/creators. | RPM is developer-friendly (free tiers) and targets cross-platform games/VR. | Bitmoji/Mii target casual social players. | Our project is aimed at end users and creators looking for precise control in a web application. It combines the ease of use of RPM with the detail of MetaHuman, offering intuitive in-browser controls over shape and texture that existing systems don't offer. |

## 3.5 Morphing Techniques

At the core of feature customization is mesh morphing. MetaHuman, Ready Player Me, and similar tools rely on blend shapes (morph targets) under the hood. In MetaHuman, each face comes with a set of predefined morph targets for expressions and body shape (though the Creator UI hides these details). Once exported to Unreal, the character uses control rigs and morph targets for animation and subtle deformations. Ready Player Me avatars include blend shapes for facial animations (e.g. eye blinking) and support mappings to ARKit/Oculus face rigs, but these are fixed – the user cannot introduce new ones.

Embedding morphing in a web app is challenging because it means recalculating vertex positions on the fly. For example, Google’s “Ginger” demo (2016) was a browser-based character with live sliders to tweak facial features. As one write-up notes [18], Ginger shows “a customizable head that uses a slider to tweak various facial features, much like a character customization system one would find in a video game”.

Figure 3.9: Ginger demo with sliders and morph target drop-down



This works, but it required careful optimization (and even then, was presented as a demo rather than a full product). In a real-time web environment, performance and memory are limited, so any large mesh with hundreds of blend shapes can become slow.

This project’s proposed flexible browser-based interpolation tackles this by computing intermediate meshes directly in WebGL. Instead of fixed presets, we allow arbitrary blends between example shapes. For instance, sliding a “nose width” control would linearly interpolate vertex positions between narrow and wide nose models. We also plan to blend textures (e.g. adding wrinkles) dynamically in the fragment shader. This approach is not currently available in existing avatar creators – they either use a fixed set of presets (RPM, MetaHuman) or require offline sculpting. By implementing a custom morph-and-texture pipeline in JavaScript/WebGL, our system gives users smooth, continuous control in real time, something not offered by the surveyed tools.

**Chapter 4**

# Implementation Process

**Chapter 5**

# Final Design

[Es pot separar conclusions i treball futur en capítols diferents, sense sobrepassar les 30 pàgines del cos de treball.]

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

**Chapter 6**

# Results

[Es pot separar conclusions i treball futur en capítols diferents, sense sobrepassar les 30 pàgines del cos de treball.]

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

**Chapter 7**

# Conclusions & Future Work

[Es pot separar conclusions i treball futur en capítols diferents, sense sobrepassar les 30 pàgines del cos de treball.]

Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.Lorem ipsum dolor sit amet, consectetur adipisici elit, sed eiusmod tempor incidunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquid ex ea commodi consequat. Quis aute iure reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

**Bibliography**

*Cal documentar les fonts bibliogràfiques utilitzades amb format APA.*

[*https://guiesbibtic.upf.edu/models-citacio/APA*](https://guiesbibtic.upf.edu/models-citacio/APA)

*La Biblioteca de la UPF ofereix el gestor de bibliografies Mendeley, que us permet crear la vostra base de dades personal de referències bibliogràfiques en línia, importar referències automàticament des de diferents recursos d'informació, extreure llistes i generar bibliografies en diferents estils de citació, i incorporar les citacions i llistes de bibliografia als vostres documents de text*

[*Gestor de bibliografies Mendeley*](http://guiesbibtic.upf.edu/mendeley)

*Exemples de cita bibliogràfica [mida 11]*

Pons, E. i Vernet, J. (2009). La llengua de l’ensenyament a les CCAA amb llengua pròpia. *Revista d’Estudis Autonòmics i Federals* (REAF), (8), 144-191.

American Psychological Association (2021). *Apa Style*. Recuperat 7 de juny del 2022, des de<https://apastyle.apa.org/>

Text de la cita bibliogràfica [mida 11]

**Apèndix**

[Seguiu una estructura lògica que coincideixi amb la secció del cos principal. No hi ha límit de pàgines.

• Llista de continguts

• 1a secció

• 2a secció

• Referències (només d’informació contiguda a l’apèndix)]

1. NURBS patches: NURBS patches are individual segments or portions of a surface that are defined by a NURBS model (Non-Uniform Rational B-Splines). These patches are used to create complex 3D shapes by combining multiple smaller NURBS patches together. [3] [↑](#footnote-ref-1)
2. A Voronoi cell is a region around a point (called a seed) that contains all the nearby positions closer to that point than to any other. When you place several seeds on a plane, the space is divided so that each area belongs to the closest seed — this division is called a Voronoi diagram. [4] [↑](#footnote-ref-2)
3. WebGL (Web Graphics Library) is a JavaScript API that allows rendering 3D and 2D graphics within a web browser without the need for additional plugins. [↑](#footnote-ref-3)
4. A tool provided by Epic Games that acts as a content manager. It allows users to easily import 3D assets (including MetaHumans) into Unreal Engine (UE). [↑](#footnote-ref-4)
5. Software Development Kits are pre-made packages of code and tools that make it easier to add certain features (in this case, RPM avatars) without building everything from scratch. [↑](#footnote-ref-5)
6. Fortnite is a battle royale game developed by Epic Games, released in 2017. [↑](#footnote-ref-6)