

AUTO RIGER TOOL

ENTREGA PROYECTO FINAL

IGNACIO ARTÍMEZ AYUELA
GONZALO MARCO

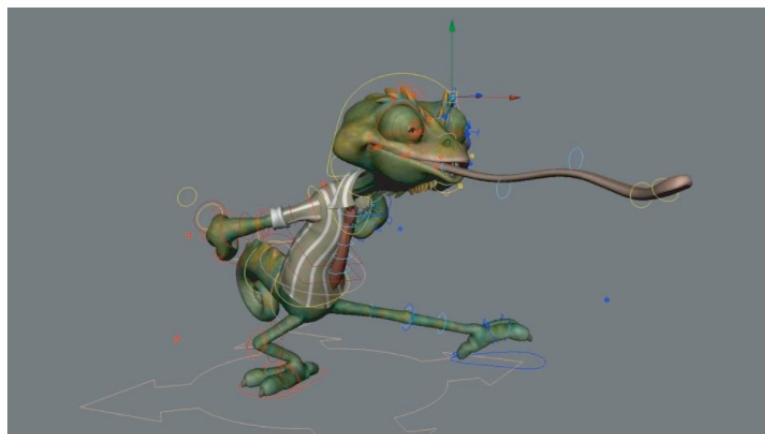
edix



INTRODUCCIÓN

El proyecto final está orientado a un público y una tecnología muy concreta. Debido a que ambos integrantes provienen del mundo de la animación y videojuegos, se vio una oportunidad para aprovechar los conocimientos adquiridos en las formación profesional, y combinarlo con lo que aprendieron en sus respectivos grados. Es por eso que se decidió crear un autorigger, mediante el cual los futuros usuarios pudieran establecer y riggear sus modelados para en un futuro poder animarlos más fácilmente.

Para entenderlo mejor, un rigging comienza con la importación de un modelado 3D, el cual queremos deformar de diferentes maneras para poder animarlo en un futuro. Dentro de ese modelo 3D, se introducen los huesos y los pesos. Tras esto, estas deformaciones deben proceder de algún sitio y seguir una lógica, unos movimientos concretos, y mover solo los vértices que se quieran para conseguir la animación. Es aquí donde entraría la herramienta de rigging.

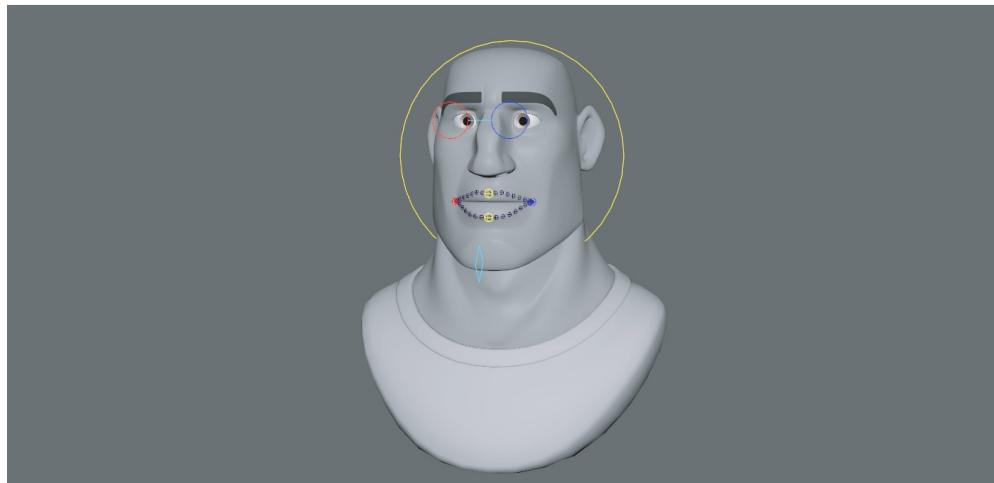


1. Ejemplo Rig - Película Disney

El proyecto consta de la herramienta en primer lugar, que es lo más importante. Esta herramienta podría ser gigante, pero en el tiempo que dado se ha tenido que centrar y focalizar los esfuerzos en algo que se pudiera entregar en plazo y que fuera funcional. Para entenderlo, se debe dividir un modelo humanoide de cuerpo entero en pequeños cachitos, boca, pie, dedos mano izquierda, párpados, etc. La herramienta cuenta por ahora con capacidad para riggear boca y ojos, así como la primera capa de deformación de la cabeza. En un futuro, se seguirá escalando funcionalidades, hasta abarcar un cuerpo entero, pero esto es un trabajo bastante largo y tedioso con bastantes procedimientos entre uno y otro. Esta herramienta está orientada a usuarios avanzados de rigging, ya que la descarga de la herramienta hasta llegar a su aplicación, implantación y uso requiere de varios pasos.

En segundo lugar, al tratarse de un producto, que aunque sea gratuito, se quiere llegar al máximo número de personas, y es aquí donde entra la segunda parte del proyecto, una página web, donde los usuarios puedan descargar la herramienta, pero además, tenga un lugar donde se les explique como funciona la herramienta de rigging, y se entienda para que les puede servir o hasta dónde pueden llegar, mediante un apartado de wiki.

El objetivo real y a corto plazo que se quiere alcanzar con este proyecto, es lanzar una herramienta documentada para poder rigger unas partes concretas de un modelado facial, y poco a poco ir completando hasta llegar a un punto que se pueda ofrecer un auto rig completo.



2. Rig Final - Creación Propia

AGRADECIMIENTOS

Nos gustaría agradecer este proyecto al centro, sobre todo a Raquel que es la que nos ha tenido que aguantar en cada trimestre del grado y nos ha dado los recursos y la oportunidad de profundizar en lo que hemos estudiado en nuestras carreras.

A nuestro compañeros, por ser el apoyo que aunque en la distancia ha sido indispensable.

PALABRAS CLAVE

Rigging

Es un proceso creativo crucial que inicia con un modelo o escultura digital tridimensional, necesaria para poder deformar y animar personajes de manera fácil y efectiva. Aunque como yo lo definiría es ponerle las cuerdas a una marioneta para poder moverla luego con esas cuerdas en vez de tener que moverlo a mano.

Joints

Una de las cosas más importantes a la hora de realizar un rig, se podría definir como los huesos o el esqueleto que va a llevar luego toda la deformación por encima.

Pesos

Los pesos son la influencia que hacen esos joints a la geometría mediante el pintado de los mismos, teniendo cada joint una influencia de 0 a 1 con respecto a uno o muchos joints.

Ribbon

Una construcción formada por un plano unos follicles y joints, los joints van debajo de los follicles en la jerarquía, lo que hace que cuando muevas los joints generales los controles se muevan con ella. Luego estos joints son los que pesan con la malla del personaje lo que hace que la deformen.

Wire

Se trata de una construcción muy parecida al ribbon en la que en vez de usar una nurb usamos una curva que unimos mediante nodos y esa curva es la que deforma la geometría.

Autorig

El nombre que se le asigna a la herramienta que es utilizada para hacer en 4 pasos todo un rig completo, sin tener que hacer todos los pasos uno a uno.

Sticky Lips

Una construcción muy utilizada en animación y muy necesaria para simular que el personaje está masticando o que tiene los labios pegados.

Character TD

Un director técnico de personajes, también conocido como TD de criaturas o animador técnico, es un tipo de director técnico en películas y videojuegos que se preocupa específicamente por personajes, criaturas y objetos mecánicos.

INDICE GENERAL

INTRODUCCIÓN	1
AGRADECIMIENTOS	3
PALABRAS CLAVE	4
RESUMEN	6
MÓDULOS FORMATIVOS APLICADOS	7
HERRAMIENTAS/LENGUAJES UTILIZADOS	8
COMPONENTES DEL EQUIPO	9
FASES DEL PROYECTO	10
Marzo	10
Abril	11
Mayo	20
CONCLUSIONES Y MEJORAS	31
BIBLIGRAFÍA	34

RESUMEN

El tema trabajado en este proyecto final de grado es por un lado y principalmente el rigging, además del diseño y programación web.

Respecto al primer tema, se ha escogido debido a que ambos integrantes del grupo cuentan con conocimientos de desarrollo y modelado 3D, lo cual combinado al grado superior de Desarrollo de Aplicaciones Multiplataformas, se decidió en aplicar esos conocimientos adquiridos y la programación para desarrollar una herramienta que pueda riggear y manipular en primer lugar la boca y los ojos de un modelado 3D. Esta herramienta es muy útil para aquellas personas que con conocimientos avanzados de rigging, ya que les permite agilizar su trabajo para poder animar después con mayor precisión.

Dentro de esta herramienta se podría profundizar mucho más, pero resulta muy difícil y costoso implementar un rigging a todo un modelo, ya que se debe adaptar a cada parte del modelado o del cuerpo humanoide para que sea fiel y las animaciones posteriores sean realistas a la realidad que el animador quiera plasmar. Por eso desde un primer momento, se decidió acotar el proyecto al tiempo que se continuaba haciendo algo viable y que fuera posible de terminar. Lo bueno es que la escalabilidad del proyecto es muy sencilla en cierto modo, debido a que "solo" se deberían ampliar las funcionalidades de la herramienta de riggin extendiéndola a otras partes del cuerpo. Sumado a esto, se debe aclarar que la mayoría de estos conocimientos no se disponían desde el comienzo, por lo que como se ve en el apartado de tecnologías utilizadas, se tuvo que recurrir a una gran variedad de ellas para poder completar el proyecto. Esta parte de aprendizaje en cierto grado resultó tediosa, pero también se debe de decir que se disfrutó aprendiendo y el resultado ha sido bastante acertado.

Sumado a esto, no se quería que esta herramienta solo ocupara un lugar en alguna página de riggin y fuera descargar y ya. Se quería que esta herramienta fuera accesible, si bien es cierto que este tipo de herramientas solo están orientadas a personas con niveles avanzados de riggin y modelado 3D, por ello se decidió aplicar otros conocimientos adquiridos en el grado de manera más directa, como es la creación de una página web, donde el usuario pudiera tanto descargar la herramienta, como aún más importante disponer de una wiki con todo el material didáctico respecto a la herramienta para que este pudiera implementarla y hacer un uso correcto de ella, así como echarle un primer vistazo antes de descargarla para comprobar si cumple con sus requisitos o no. Es una página web de consulta, por lo que se vió necesario implementar una base de datos, o sistemas complejos de páginas web, y reducirlo a lo que realmente se necesitaba.

MÓDULOS FORMATIVOS APLICADOS

Programación

Esta asignatura como es evidente, se ve plasmada en todos y cada uno de los pasos del desarrollo del proyecto. Dentro de esta, se explica más adelante como se ha distribuido el trabajo y con que lenguajes.

Entornos de Desarrollo

Esta asignatura brindó los conocimientos de Git básicos para estructurar y desarrollar un proyecto conjunto ya que el equipo estaba formado por más de una persona. En adición a esto se tuvo que aprender algunos comandos extras para perfeccionar el uso de la herramienta Git.

Lenguaje de marcas y sistemas de gestión de información

Esta asignatura se ha implementado de manera extensa así como las de programación, en la parte del desarrollo de la página web. Al igual que con el resto de módulos implementados, se ha llevado un poco más allá los conocimientos adquiridos para implementar lo que de verdad queríamos.

Desarrollo de interfaces

Dentro de este módulo se ha realizado una pequeña demo de la interfaz que tendrá la herramienta, la cual se verá más adelante.

Programación Multimedia

De este módulo se ha tomado las ideas base, no tanto las técnicas, para aplicarlo al proyecto, ya que si bien no se trata de una aplicación multiplataforma ni un videojuego, comparte varios factores con ellos.

Inglés

De este módulo, básicamente se ha decidido que como toda la industria de la informática, como la del 3D se mueven y comunican en inglés, se vio obligatorio desarrollar la herramienta en dicho idioma para en un futuro llegar al máximo de gente posible.

HERRAMIENTAS/LENGUAJES UTILIZADOS

Python

Lenguaje base usado en el desarrollo de la herramienta. Esto se vió obligatorio dada la tendencia del mercado en este sector por el uso predominante de este lenguaje además de que se ha podido ampliar nuestro conocimiento en él.

Maya

Máscara del motor gráfico más usado en sector de la animación, en el cual la herramienta ha sido desarrollada e implementada bajo su arquitectura, por ello no podría funcionar en otro motor o lenguaje.

GitHub

Software de control de versiones más conocido, con el cual se ha seguido el desarrollo del proyecto.

Qt

SDK basado en C++ para el desarrollo de la interfaz de nuestra herramienta dentro de Maya.

Pyside

Biblioteca que permite usar Qt dentro de Maya.

HTML/CSS

Lenguajes bases de etiquetado para el desarrollo básico de la página web.

Bootstrap

Front End framework, el cual ha permitido desarrollar y maquetar con mayor velocidad la página web.

JavaScript

Lenguaje de programación orientado al desarrollo web, para el uso de pequeños scripts integrados en la página web.

Pycharm

Editor el cual permite la sinergia entre Python, sus librerías y Maya.

Atom

Editor principal para el desarrollo del proyecto.

Scrum

Más que una herramienta, es la metodología que se ha seguido para el desarrollo del proyecto, mediante la asignación de tareas en tiempos concretos.

COMPONENTES DEL EQUIPO

Ignacio Artímez Ayuela

- Desarrollo Web
- Scrum Master
- Soporte Herramienta Rigging
- Maquetación



Gonzalo Marco Fernández

- Character TD
- Soporte Web



FASES DEL PROYECTO

MARZO

Entrega anteproyecto

Durante el mes de Marzo, comenzamos con el desarrollo de la idea lo que podría ser nuestro proyecto, final, el cual resultó ser bastante parecido a nuestra idea inicial.

Para ello y como primera entrega, tuvimos que desarrollar el anteproyecto, donde exponíamos la idea que teníamos y hasta donde queríamos llegar, lo que desembocó en esto:

"La principal necesidad de mercado que vemos es poder vender un autorig facial con una página dedicada 100% a ese programa y a su respectiva documentación, que a la vez sea llamativa para el usuario. Las razones por las que elegimos esta temática es porque tenemos una necesidad profesional en la que se necesita que todos los pasos que hay a la hora de hacer un rig de un personaje se automaticen y también que el usuario o persona que haga uso del script tenga una documentación donde se pueda ayudar para completarlo, a la par que intentar que sea lo más sencillo posible para el usuario. A parte es una herramienta que vamos a desarrollar para poder utilizarla a nivel profesional más adelante. Vamos a realizar una aplicación con la que se va a poder riggear un personaje con unos pocos botones sin tener que hacer paso a paso todas las partes que involucran lo que es el rigging de un personaje."



3. First Demo Interface

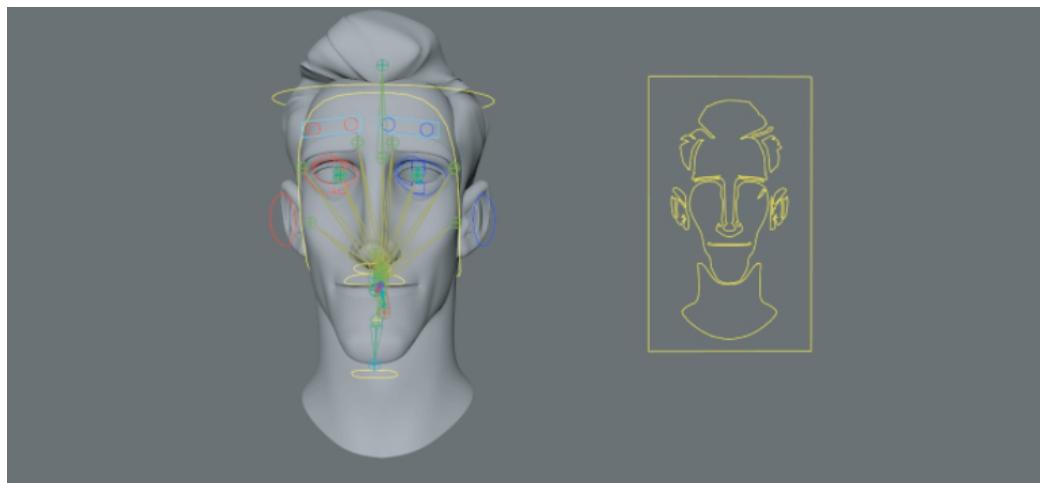
De esta manera definimos en cierto grado lo que queríamos que fuera nuestro proyecto, así como su finalidad a la hora de poder aplicarlo al mundo real, y no dejarlo meramente en el desarrollo de una aplicación básica para aprobar el proyecto. Desde el principio hemos querido llevar este proyecto a algo mucho más grande, aun que como se puede ver a lo largo del desarrollo en la metodología Scrum tuvimos que tomar decisiones para poder tener al final un proyecto algo más pequeño al planeado inicialmente pero mucho más pulido.

ABRIL**Semana 1 - Definición de opciones de cara al proyecto**

La primera semana de Abril decidimos que era más interesante ver lo que podíamos llegar a hacer antes que ponernos manos a la obra sin saber lo grande que podía llegar a ser nuestra herramienta.

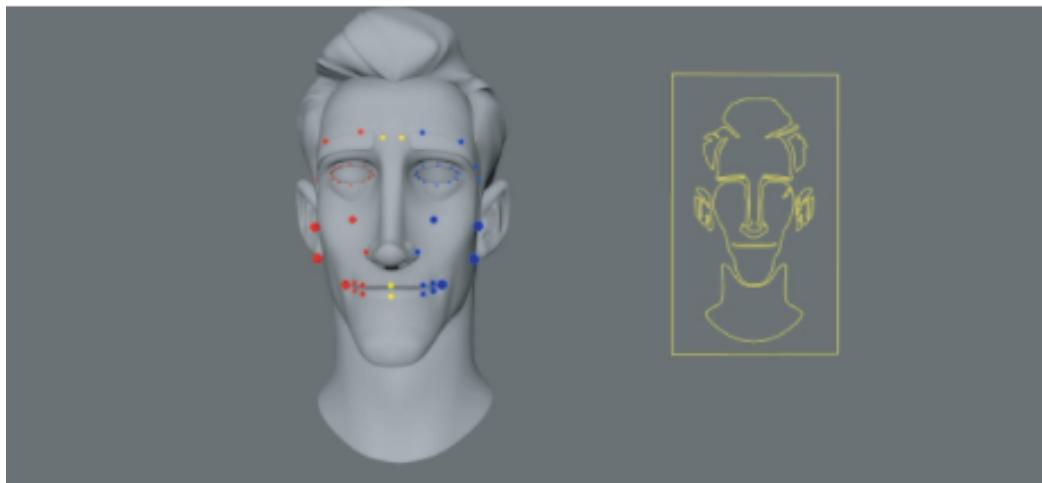
Para ello hemos tomado como referencia un trabajo de Gonzalo en el que se veía claramente todas las fases que tiene un rig facial o capas de deformación que tienen, que paso a describir a continuación:

Esta primera capa de deformación está formada solo por joints que son los que unen luego el rig corporal con el rig facial. Es una jerarquía de joints que mediante un skinCluster hace que esos joints o huesos deformen la geometría de nuestro personaje. Se trata de una construcción básica en la que no utilizamos ningún elemento externo a los joints.



4. Primera capa deformación

La segunda capa de deformación en este caso se trata de un wire. Un wire es un tipo de deformador que tiene maya en el que dado una geometría y una curva, cuando deformemos la curva nos va a变形ar la geometría, en esta construcción usamos cosas más avanzadas a la hora de que estos controladores de la segunda capa de deformación sigan a la primera capa de deformación.

ABRIL

5. Segunda capa deformación

La tercera capa de deformación se trata de blendshapes correctivos o keyshapes faciales, este apartado lo descartamos casi al momento, ya que para la realización de blendshapes se utilizan herramientas de esculpido y trabajo de división de lados por vértices, por lo que es un apartado avanzado para meter en la herramienta. Un blendshape correctivo trata de corregir una deformación de la primera capa, que no nos guste del todo o incluso mejorar o aportar algo a deformaciones ya existentes en nuestro rig.



6. Sin blendshape activado



7. Con blendshape activado

Tras analizar las capas de deformación de un rig finalizado ambos sabíamos que a parte de estas deformaciones se puede riggear de mil maneras diferentes, cosa que Gonzalo ha estado buscando durante toda la duración de este trabajo, ya que buscábamos las mejores construcciones para introducirlas a nuestra herramienta.

ABRIL

Durante la primera parte de la fase de desarrollo del proyecto, estuvo centrando más, como se puede comprobar en lo visto anteriormente, en la herramienta, ya que era el grueso del proyecto, pero se comenzó a dar unas primeras pinceladas sobre el alcance que se quería que tuviera la web. Para ello se realizó un brainstorming de las funciones que esta podría tener, hasta terminar descartando las que no se veían necesarias realmente y dejando algo sencillo, pero que cumpliera con su labor.

Funcionalidades iniciales de la web:

- Informar mediante texto
- Tutoriales prácticos
- Bot chat básico para dudas
- Informar mediante Videos
- Ofrecer descarga directa
- Ofrecer manera de contactar con nosotros
- Sistema de gestión de usuario con una base de datos
- Creación de tienda en la web

Finalmente, tras verse lo que sería ideal y comparándolo con el tiempo que se tenía se decidió reducir esa lista en la siguiente:

- Informar mediante texto
- Informar mediante Videos
- Ofrecer descarga directa
- Ofrecer manera de contactar con nosotros

Ahora ya se tenía una lista de las funcionalidades que la página web debía tener. A primera vista parece algo simple, algo que también se pensó en un inicio, pero, ¿realmente merecía la pena ampliar y complicar todo haciendo una web con tantas funcionalidades cuando ni de lejos hace falta? Es por eso que esas cuatro características parecieron más que suficientes para poder comenzar a desarrollar posteriormente la página web.

ABRIL

Semana 2 - Diseño Web

Una vez se tenía un plan de acción con todas las características que debía tener la web, se comenzó a pensar como debería ser. Pero antes de comenzar a desarrollar y picar código como tal, se realizó en primer lugar el workflow que queríamos que tuviera nuestra web, de esta manera a la hora de programarla se tendría una guía de diseño que agilizaría mucho el trabajo.

El objetivo de dicho workflow se dividía en dos partes:

Página principal:

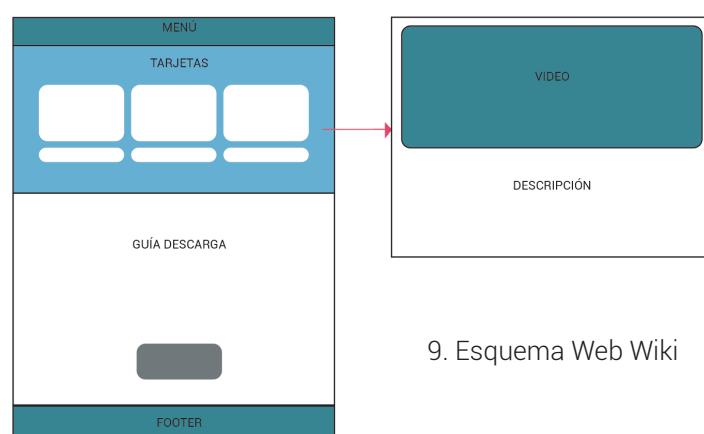
En esta página se tendría un primer vistazo a la herramienta, mediante una pequeña descripción del proyecto como tal, un apartado donde ambos integrantes del proyecto se introducirían y por último un formulario de contacto.



8. Esquema Web Principal

Wiki:

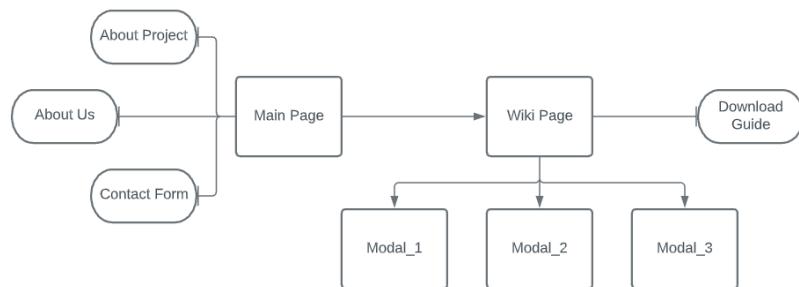
Esta es la parte que más importancia tiene de toda la página web. En ella se tienen unas tarjetas donde se explicará cada una de las diferentes funcionalidades de la herramienta, al hacerlo de esta manera, se está “modulizando”, lo que en un futuro será mucho más fácil escalar. Y además de esto una pequeña guía de instalación de la herramienta junto con el botón de descarga.



9. Esquema Web Wiki

ABRIL

Una vez se tenían los diseños esquemáticos de las dos páginas que compondrían la web, se realizó el sistema de workflow mediante el cual se definiría la sinergia entre ambas.



10. Diagrama UML Workflow Web

Diseño Herramienta

Tras esta primera semana de Abril en la que vimos todas las posibilidades que tenía esta herramienta a nivel de funcionalidad y de interfaz de usuario, decidimos que nos íbamos a centrar 100% en la segunda capa de deformación, aunque sí que es verdad que en las semanas posteriores se explicará mejor alguna construcción de las otras capas que finalmente decidimos descartar.

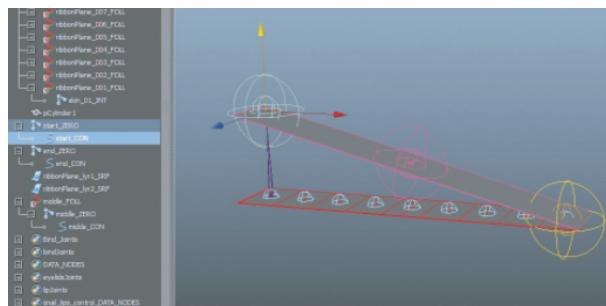
Esta decisión reducía el trabajo significativamente ya que nos tendríamos que centrar solo en varias construcciones, aunque estas construcciones son mucho más complicadas que la primera capa de deformación y tampoco nos reducía tanto las opciones ya que para el rig de la boca se pueden hacer 5 o 6 tipos de construcciones, igual para los ojos y las cejas.

ABRIL

Algunas de las construcciones que planteamos hacer fueron:

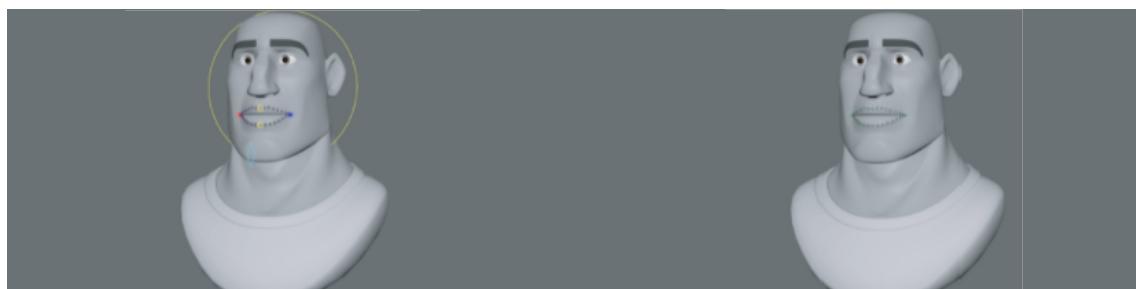
Wire: anteriormente citada en la semana 1 de Abril, en esta construcción veíamos pros y contras, ya que a nivel de posición la geometría se deforma perfectamente, pero a nivel de rotación y escala no funciona bien. Si que es verdad que al ser un deformador que va por encima del skinCluster no interfiere para nada con ello.

Ribbon: es muy parecida a la deformación por wire pero en este caso lo que deforma son joints con lo que conseguimos añadir la rotación y la escala a nuestra deformación. También un punto a favor de este tipo de construcción es que al utilizar un plano o nurb podemos duplicar dicho plano y a cada una aplicarle un deformador distinto como en la imagen de debajo.



11. Ejemplo Ribbon

Joints: una construcción en la que pasándole el número de loops que tenemos en nuestra geometría nos crea un locator por cada loop que es la guía para crear luego esta construcción,a parte de toda una red de nodos para que los joints se muevan como el animador quiera.



12. Ejemplo Joints

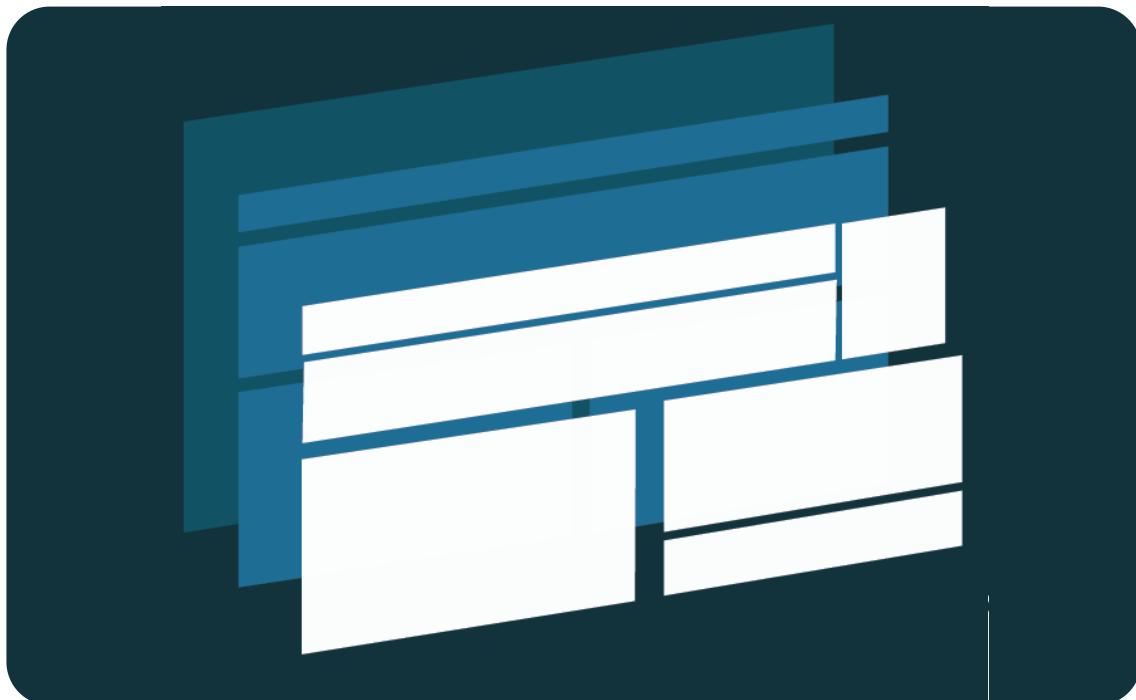
13. Ejemplo Guías

ABRIL**Semana 3 - Prototipo Web**

Una vez se tenía el diseño cerrado y elaborado, se pasó a la parte de prototipado, para ello se partió desde cero, mediante el uso de algunas de las tecnologías anteriormente mencionadas, sobre todo Bootstrap, librería la cual permitió tener en poco tiempo un primer prototipo de la página web, el cual era muy sencillo, que más que un prototipo como tal, era una pequeña base para seguir desarrollando la web.

Este prototipo realmente no duró mucho ya que al poco tiempo se procedió a picar la página web como tal al no tratarse de una web muy compleja.

Realmente esta semana no fue muy fructífera respecto al trabajo debido a asuntos personales de los integrantes.



14. Prototipo inicial Bootstrap

ABRIL**Semana 4- Primeros pasos rigging boca y cluster**

En esta cuarta semana de Abril se comenzó a realizar la construcción de la boca mezclando la primera capa de deformación con los ribbon, ya que primero se necesitaba una capa de joints que sean los controles generales.

A través de código se crearon los locators que sirven como guía para esa primera capa de deformación. Luego a través de curvas y de la nurb como se explicó anteriormente se crea la deformación ribbon.

Este tipo de sistema es muy acertado cómo podía quedar pero se pensó que no era el mejor sistema ya que se dependía de construcciones y cosas que un motor de videojuegos no soporta como son las curvas y las nurbs, por lo que al poco tiempo se empezó a buscar una solución que no dependiera de estas construcciones.



15. Sistemas aplicado

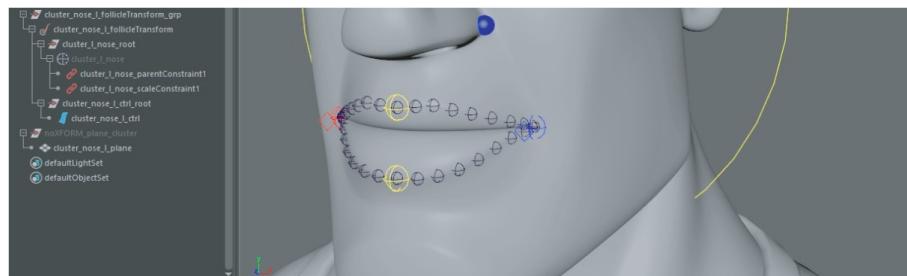
Cluster

Una construcción que se necesitaba que estuviese en la herramienta ya que no es solo para una zona concreta sino que se puede usar para las aletas nasales, como para los mofletes o los pómulos.

Cogiendo un vértice y pasándole un lado y un nombre se va a crear una estructura en la que se debe quitar la selección de ese vértice ya que se crea un

ABRIL

cluster se crea con los vértices que hubieses seleccionado al crearlo, por lo que se borraron y una vez terminada la ejecución de esta herramienta se debía meter la geometría entera en el cluster de nuevo para que pueda modificar toda la geometría.



16. Cluster Aplicado

MAYO**Semana 1 - Desarrollo rigging boca**

En esta cuarta semana de Abril se comenzó a realizar la construcción de la boca mezclando la primera capa de deformación con los ribbon, ya que primero se necesitaba una capa de joints que sean los controles generales.

A través de código se crearon los locators que sirven como guía para esa primera capa de deformación. Luego a través de curvas y de la nurb como se explicó anteriormente se crea la deformación ribbon.

Este tipo de sistema es muy acertado cómo podía quedar pero se pensó que no era el mejor sistema ya que se dependía de construcciones y cosas que un motor de videojuegos no soporta como son las curvas y las nurbs, por lo que al poco tiempo se empezó a buscar una solución que no dependiera de estas construcciones.



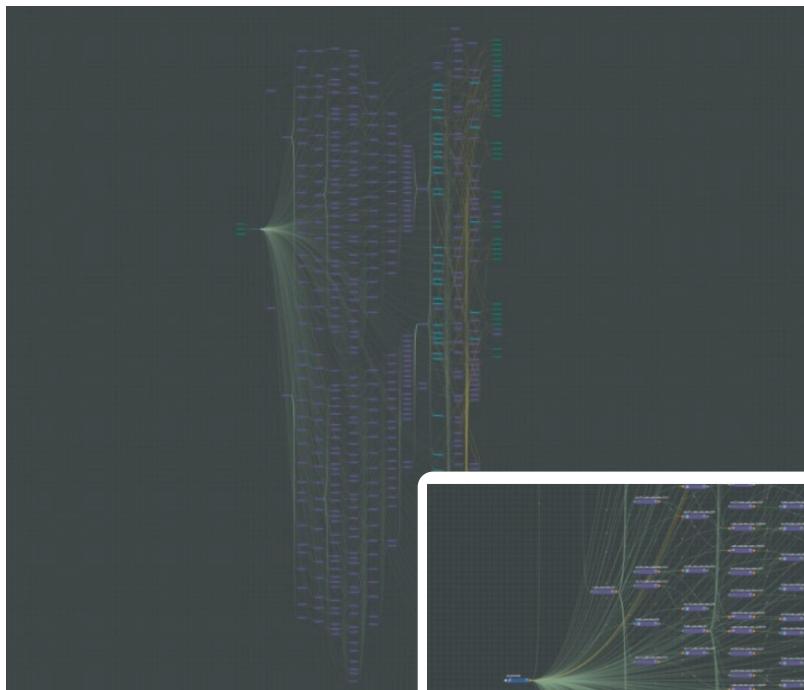
17. Guías Colocadas



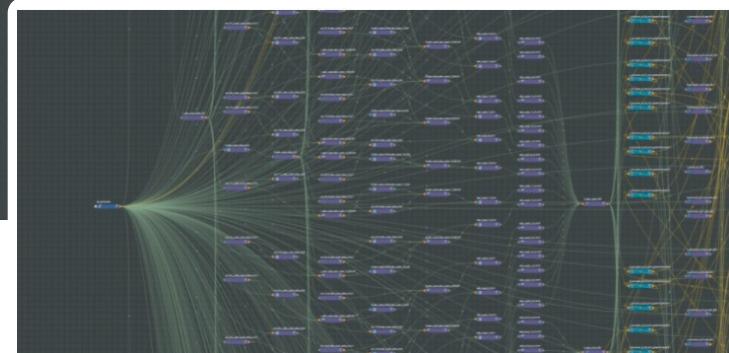
18. Construcción Terminada

MAYO

Como se puede comprobar en la imagen ampliada, todas las conexiones que la herramienta hace salen de un objeto, el que ha sido comentado antes que lleva todos los atributos con los que se puede modificar la posición y el comportamiento del rig de la boca.

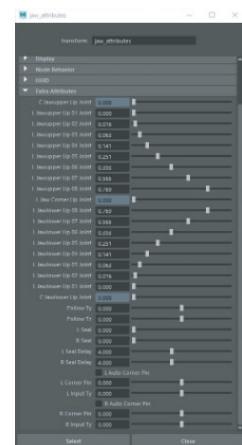


19. Red Nodal Completa



20. Red Nodal Ampliada

Como pueden comprobar en la imagen ampliada, todas las conexiones que la herramienta hace salen de un objeto, el que ha sido comentado antes que lleva todos los atributos con los que se puede modificar la posición y el comportamiento del rig de la boca.



21. Atributos Rig Boca

MAYO

Estructuración del código jaw_utils

Le pasan el número de guías que se quieren para hacer el rig, llamada "[createguides\(number\)](#)" tras esta función el usuario tiene dos funciones que devuelve todas las guías que han sido creadas y las mete en el diccionario para poder trabajar con ellas en los siguientes pasos de la construcción del rigging de la boca.

La siguiente función es llamada '[build\(\)](#)', en la que llaman al resto de funciones para terminar la construcción de la boca, creando las conexiones, los joint y finalmente en otra versión del código los controles.



```

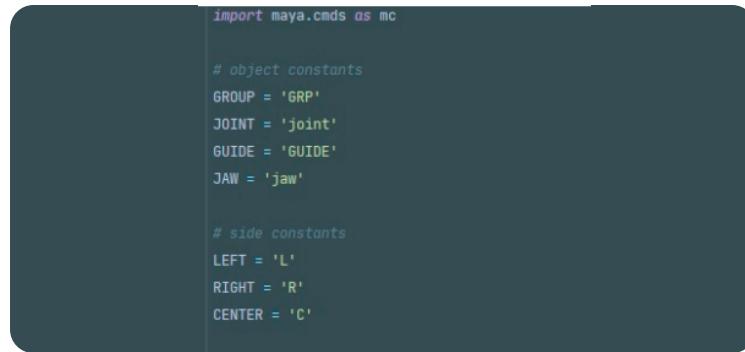
def createMinorJoints():
    """
    :return:
    """
    minor_joints = []
    for guide in lip_guides():
        mat = mc.xform(guide, q=True, m=True, ws=True)
        jnt = mcjoint(name=guide.replace(GUIDE, JOINT))
        mc.setAttr('{}.radius'.format(jnt), 0.5)
        mc.xform(jnt, m=mat, ws=True)

        # parent joints
        mc.parent(jnt, '{}_{}_lip_minor_{}'.format(CENTER, JAW, GROUP))
        minor_joints.append(jnt)
    return minor_joints

```

22. Función para crear Minorjoints

Como se puede comprobar en la imagen de abajo, se ha utilizado lo máximo posible el PEP 8 de python a la hora de definir variables y constantes y a la hora de formatear todas las referencias a variables en vez de sumar strings como se hace en java.



```

import maya.cmds as mc

# object constants
GROUP = 'GRP'
JOINT = 'joint'
GUIDE = 'GUIDE'
JAW = 'jaw'

# side constants
LEFT = 'L'
RIGHT = 'R'
CENTER = 'C'

```

23. Definición de constantes

MAYO

Desarrollo Web Principal

Para el desarrollo de la web principal se cogió tanto el diseño como el prototipo previo para comenzar el desarrollo final. Al haberse desarrollado de manera pau-tada y por partes, al llegar a este punto era cuestión de transformar todo en código. Para ello se procedió a picar el código con la librería Bootstrap, dando como resultado esta página principal con la característica de ser en scroll.

The screenshot displays the homepage of the AUTO RIGGER WEB project. At the top, there's a navigation bar with the title "AUTO RIGGER WEB" and links for "WIKI", "ABOUT", and "CONTACT". The main visual features a 3D character model with various yellow and blue circles overlaid, representing rigging points. To the right of the character, the word "AUTOTRIGGER" is prominently displayed, followed by the subtitle "Autotrigger Tool for Artists". Below this, there's a section titled "ABOUT THE PROJECT" which includes a brief description of the tool and a "W WIKI" button. The "ABOUT US" section features profiles for two team members: Ignacio Artimez (24 years old, Computer Science & Videogame Engineer) and Gonzalo Marco (22 years old, Creative Media Production). A message from the team states their motivation for creating the tool. The "CONTACT US" section provides fields for Name, Email Address, and Message, along with a "Send" button. At the bottom, there's a footer with sections for "LOCATION" (Madrid, Spain, Edix Educación Digital), "AROUND THE WEB" (social media icons), and "ABOUT" (Final Project DAM). The footer also includes a copyright notice: "Copyright © DAM Project 2021".

MAYO

Como se ha podido observar, el uso de esta librería nos permite estructurar de manera muy sencilla y rápida la web, una vez se tienen los conocimientos necesarios, y como podemos ver en el siguiente ejemplo todo está estructurado mediante cajas y propiedades concretas para conseguir el acabado necesario.



25. Esquema estructural Bootstrap

Por último se puede observar un pequeño formulario, el cual por ahora no está habilitado ni funcional, a la espera de obtener un servidor cuando la web se decida lanzar a los usuarios finales.

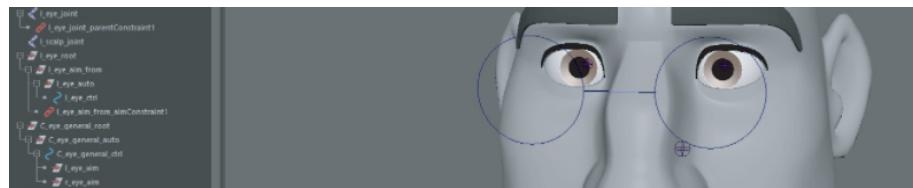
MAYO

Semana 2 - Desarrollo Ojo

El desarrollo del ojo se centró en la construcción de un joint por cada ojo el cual controlaba la rotación del mismo y apunta con el aimConstraint al objeto que tiene los controles en su jerarquía.

Esta construcción era mucho más sencilla que las anteriores ya que no necesita ninguna construcción avanzada es solo un constraint y trabajo de jerarquía.

En este caso el usuario tiene que seleccionar la geometría del ojo izquierda y con la tecla shift seleccionada, seleccione la geometría del ojo derecho, tras la ejecución del método con esa selección, el rig del ojo es creado solo con dicha función.



26. Ejemplo construcción del ojo

```

if len(eye_gns) == 0:
    mc.confirmDialog(title="Empty selection", message="You have to select the eyeBall mesh", button=['OK'])
    return

else:
    eye_cns = mc.cluster()
    mc.select(dt=True)
    joint_ey = mc.joint(n="{}_{0}_{}_".format(side, EYE, JOINT), rad=0.5)
    mc.mirrorTransform(joint_ey, eye_cns)
    scalp = mc.duplicate(joint_ey, n="{}_{0}_scalp_{}".format(side, JOINT))
    mc.delete(eye_cns)
    # Creating eye size constraint
    ctrl = mc.circle(n="{}_{0}_ctrl".format(side, EYE))
    mc.matchTransform(ctrl, scalp)
    mc.hilite(ctrl)
    mc.select("ctrl[0]", r=True)
    mc.move(0, 0, 5, r=True)
    mc.rotate(0, 0, 90, r=True)
    mc.parentConstraint(ctrl[0], joint_ey)
    cns.clean(ctrl[0])
    mc.select(clear=True)
    from_ey = mc.createNode('transform', n="{}_{0}_vis_from".format(side, EYE))
    mc.matchTransform(from_ey, "{}_{0}_root".format(side, EYE))
    mc.parent(from_ey, "{}_{0}_root".format(side, EYE))
    mc.parent("D_D_auto".format(side, EYE), from_ey)
    mc.select(clear=True)
    min_gpr = mc.createNode('transform', n="{}_{0}_min".format(side, EYE))
    mc.matchTransform(min_gpr, ctrl[0])
    mc.move(0, 0, -5, r=True)
    mc.absoluteConstraint(min_gpr, eye_cns)
    mc.setAttr("({}_{0}_min).axisFrom", axisConstraint1.axisVector1, forest(side,EYE), 0)
    mc.setAttr("({}_{0}_min).axisFrom", axisConstraint1.axisVector2, forest(side,EYE), 1)
    mc.setAttr("({}_{0}_min).axisFrom", axisConstraint1.worldUpType, forest(side,EYE), 2)
    mc.connectAttr("({}_{0}_root.worldMatrix[0]".format(side,EYE), "({}_{0}_min.axisFrom.worldMatrix)".format(side,EYE))
    mc.select(dt=True)
    general_grp_name = "{}_{0}_general_ctrl".format(EENTER, EYE)

```

27. Código construcción ojo

MAYO

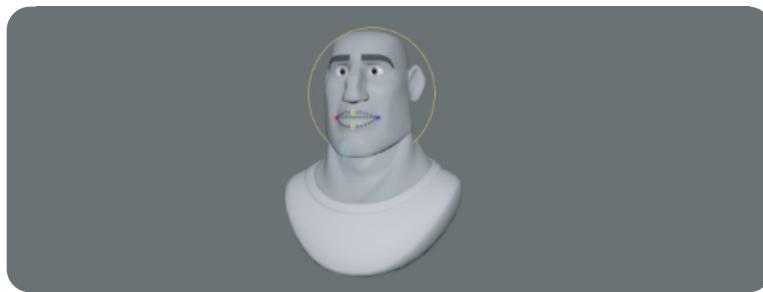
Se comprueba que haya geometría seleccionada antes de empezar con la construcción del sistema en sí, si no existe dicha selección una ventana modal o cuadro de diálogo es ejecutado parando el programa hasta que el usuario pulse aceptar.

Una vez comprobada la selección se crea un cluster directamente para obtener la posición central de las geometrías que el usuario ha seleccionado lo que hace que el usuario no tenga que realizar ningún paso más una vez ejecutada dicha función.

Desarrollo rigging boca

El rigging de la boca estaba casi finalizado pero, el usuario tendría que seleccionar un objeto que no es un controlador para poder modificar los atributos anteriormente citados y poder cambiar la deformación del rig de la boca.

Por lo que se añaden controles y atributos a dichos controles para que el usuario no tenga que seleccionar nada del rig que no quieran tocar.



28. Construcción de la boca con controles

```

jmc.attr <= 'jmc_attributes'
{
    ...
}

ctrl_law = mc.circle(n='O_O_ctrl').format(CENTER, 500, res[1], k, fl, segment, sections, shB, r1)[0]
mc.parent('OShape'.format(ctrl_law), 'O_O_C').format(CENTER, 300, 200T, r=True, s=True)
mc.setAttr('ctrl_law')
mc.milite('OShape'.format(ctrl_law))
mc.select('OShape_cv(7)'.format(ctrl_law))
mc.hew(k, -1, 0, r=True)
mc.select('ctrl_law')

ctrl_invert = mc.circle(n='O_O_invert_ctrl').format(CENTER, 500)[0]
mc.parent('OShape'.format(ctrl_invert), 'O_invert_C'.format(ctrl_invert)).format(CENTER, 300, 200T, r=True, s=True)
mc.setAttr('ctrl_invert')
mc.milite('OShape'.format(ctrl_invert))
mc.select('OShape_cv(7)'.format(ctrl_invert))
mc.hew(k, 1, 0, r=True)
mc.select('ctrl_invert')

ctrl_law = mc.circle(n='O_O_ctrl').format(CENTER, 500, 200T)
sel = mc.ls(sl=True)[0]

# adding follow attr
mc.addAttribute('O_follow', 'no="______', 'asUser', 'asFollow', k=False)
mc.addAttribute('O_follow_tv', 'at=double', 'min=-10, max=10, dv=4, k=True)
mc.addAttribute('O_follow_tv2', 'at=double', 'min=-10, max=10, dv=4, k=True)
mc.connectAttr('O_follow_tv'.format(sel), 'O_follow_tv'.format(sel))
mc.connectAttr('O_follow_tv2'.format(sel), 'O_follow_tv2'.format(sel))
mc.setAttr('O_follow_tv'.format(sel), 0)
mc.setAttr('O_follow_tv2'.format(sel), 0)

mc.select('ctrl_law')

for side in ['L']:
    a corner
    {
        ctrlCorner_law = mc.circle(n='O_O_broadCorner_ctrl').format(side, 500, res[1], fl, segment, sections, shB, r1)[0]
        mc.parent('OShape'.format(ctrlCorner_law), 'O_O_broadCorner_C'.format(side)).format(side, 300, 200T, r=True, s=True)
        mc.setAttr('ctrlCorner_law')
        mc.milite('OShape'.format(ctrlCorner_law))
        mc.select('OShape_cv(7)'.format(ctrlCorner_law))
        mc.hew(k, 1, 0, r=True)
    }
}

```

29. Código para la creación de dichos contro-

MAYO

Desarrollo Web Wiki

Una vez desarrollada la página principal de la página web, se procede a realizar y programar la segunda y más importante, la referente a la wiki. Dentro de esta página se pretende introducir las tarjetas que a su vez contienen las descripciones y videos de cada tipo de parte de la herramienta desarrollada.

Se siguió la misma metodología que con la anterior parte, mediante el uso de la estructura de bootstrap como se pudo ver en el esquema respectivo.

The screenshot shows a dark-themed web page for the 'Auto Rigger' project. At the top, there's a navigation bar with 'AUTO RIGGER' on the left and 'WIKI' in the center. Below the navigation, there are three image cards: 'CLUSTER' (a hand holding a cluster tool), 'MOUTH' (a mouth), and 'EYES' (an eye). The main content area is titled 'HOW TO INSTALL' and contains four numbered steps:

1. Click on download on the web page and unzip the files.
2. Save the directory where you have unzipped the files.
3. Open your Maya and go to the script editor and type this code:

```
import sys
#Download path
path = "C:\Users\luis\Documents\maya\scripts\Autorigger\eyelids_utils"
for module_name in sys.modules.keys():
    if module_name == 'eyelids_utils':
        del(sys.modules[module_name])
if path not in sys.path:
    sys.path.append(path)
import eyelids_utils
eyelids_utils.funcion()
```

4. Execute the script with the function you want depending on the construction you want.

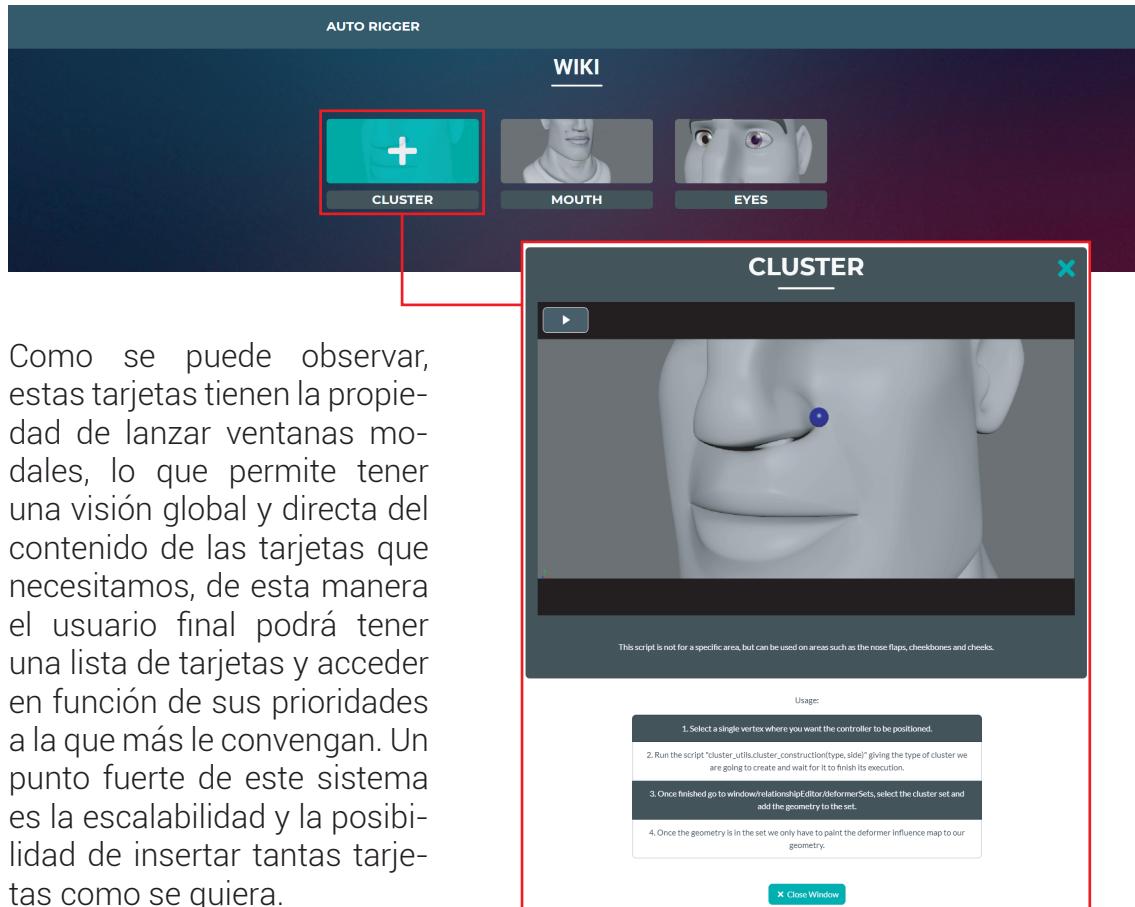
At the bottom of the page, there's a 'Download!' button with a download icon. The footer section includes 'LOCATION' (Madrid, Spain, Edix Educación Digital), 'AROUND THE WEB' (links to Twitter, LinkedIn, and YouTube), and 'ABOUT' (Final Project DAM). A small copyright notice at the very bottom reads 'Copyright © DAM Project 2021'.

30. Página Principal Wiki

A excepción de la página principal esta cuenta con ciertas características añadidas.

MAYO

Como característica principal encontramos las ventanas modales.



31.Sistema ventana Modal

A nivel código es importante remarcar que este sistema es muy simple, usando la propiedad modal de etiqueta, al pinchar el programa sabrá a qué modal llamar y la pondrá en pantalla.

```

<!-- Wiki Cluster-->
<div class="col-md-6 col-lg-4 mb-5">
  <div class="portfolio-item mx-auto" data-toggle="modal" data-target="#clusterModal">
    <div class="portfolio-item-caption d-flex align-items-center justify-content-center h-100 w-100">
      <div class="portfolio-item-caption-content text-center text-white"><i class="fas fa-plus fa-3x"></i></div>
      
    </div>
    <div class="mt-3"></div>
    <h3 class="py-1 portfolio-modal-title text-secondary text-uppercase mb-0 text-center bg-dark text-light rounded" id="CLUSTER">CLUSTER</h3>
  </div>
</div>
<!-- Cluster Modal-->
<div class="portfolio-modal modal fade" id="clusterModal" tabindex="-1" role="dialog" aria-labelledby="portfolioModalLabel" aria-hidden="true">
  <div class="modal-dialog modal-xl" role="document">
    <div class="modal-content">
      <div class="modal-body text-center">
        <div class="text-align-center">
          <div class="col-lg-12 bg-dark rounded pt-3">
            <button class="close btn-un" type="button" data-dismiss="modal" aria-label="Close">
              <span aria-hidden="true"><i class="fas fa-times"></i></span>
            </button>
          </div>
          <h2 class="portfolio-modal-title text-secondary text-uppercase mb-0 text-light" id="CLUSTER">CLUSTER</h2>
          <div class="divider-custom-white">
            <div class="divider-custom-line-white text-light"></div>
            </div>
          <!-- Cluster Video -->
          <video id="my-video" class="video-js vjs-fluid" controls preload="auto" width="1000" poster="assets/img/wiki_cluster.png" data-setup="{}" muted>
            <source src="assets/videos/video_cluster.mp4" type="video/mp4" />
          </video>
        </div>
      </div>
    </div>
  </div>
</div>

```

32. Código Modales

MAYO

Semana 3 - Ultimar Herramienta

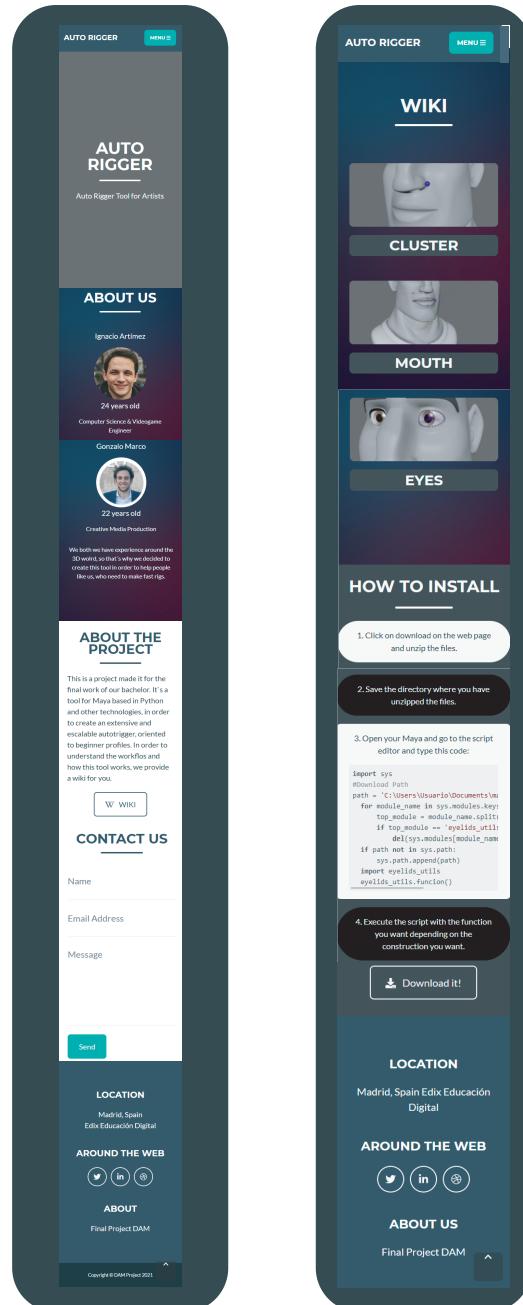
Durante la última semana del proyecto se tomaron varias decisiones con respecto al código que se presentaría definitivamente ya que habían otras construcciones en las que se estaba trabajando pero finalmente no se entregan como parte de la herramienta.

Por lo que en la carpeta descargable de la web se adjuntan el script de jaw_utils y de eye_utils.

Ultimar Web

Respecto a la web, en un principio se tenía el vídeo insertado como un gif, por lo que el usuario no tenía control sobre ello. Es por eso que decidimos incluir una librería externa la cual permitió a la web ofrecer un reproductor de vídeo insertado en él mismo. Además de esto se terminó de pulir otros aspectos para que como se puede a continuación la página es completamente responsive.

Así como un botón de descarga, otro para scroll instantáneo al top de la página, un menú plegado y otros ajustes.

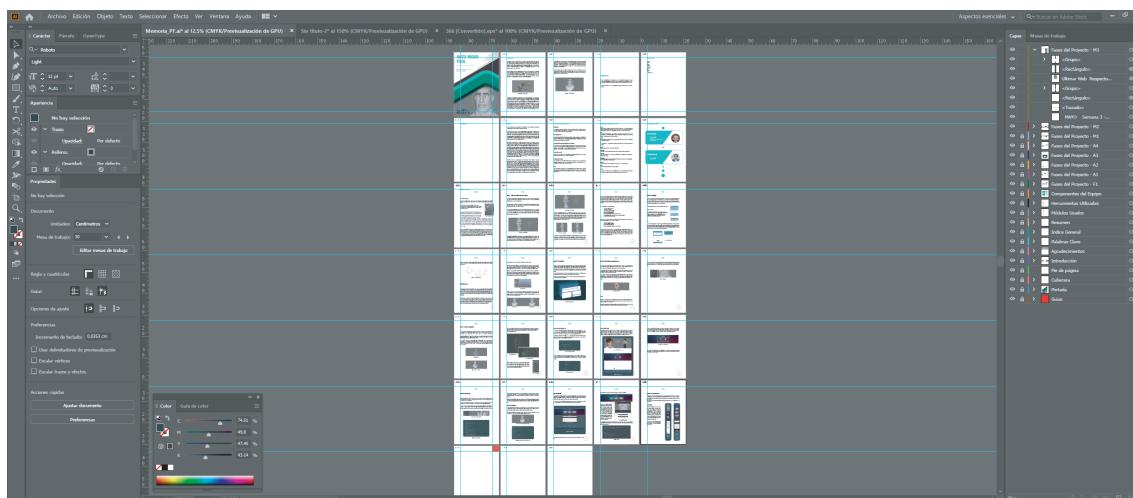


MAYO

Memoria

Durante la última semana del proyecto, hemos realizado la memoria. Si bien este paso no es tan importante como la realización de la propia herramienta o la página web, se vio oportuno realizar una perfecta maquetación y sobre todo descripción del proyecto, debido a que la naturaleza del mismo si tiene que ver con el grado superior impartido, se tratan tecnologías mucho más novedosas y apartadas al mismo.

Para la creación de la memoria se optó por el software Illustrator, estructurada sobre la metodología scrum basada en sprints de semanas en las cuales se desarrolló todo el proyecto.



34. Maquetación Memoria

CONCLUSIONES Y MEJORAS

Como conclusión del proyecto, en términos generales, se ha conseguido completar el objetivo que se tenía de crear una herramienta funcional de rigging junto con una página web. Se debe destacar, que debido al tiempo dado para la realización del proyecto, se tomaron al principio del mismo una serie de decisiones para lograr acortar y obtener una lista de tareas real que permitiera entrar en tiempo sin sacrificar de manera agresiva contenido.

Durante el proyecto se ha llevado una metodología scrum, la cual ha permitido al equipo llevar al día las tareas, y lo que es más importante, poder compaginarlo con el trabajo y otras tareas. Gracias a esto se ha podido completar el proyecto sin dejarse ninguna funcionalidad no prediseñada por el camino

En conclusión el equipo está muy contento con el proyecto final y como ha quedado, además de ser un proyecto en el que se ha disfrutado trabajando debido a la originalidad del mismo y poder aprender cosas nuevas.

Mejoras Web

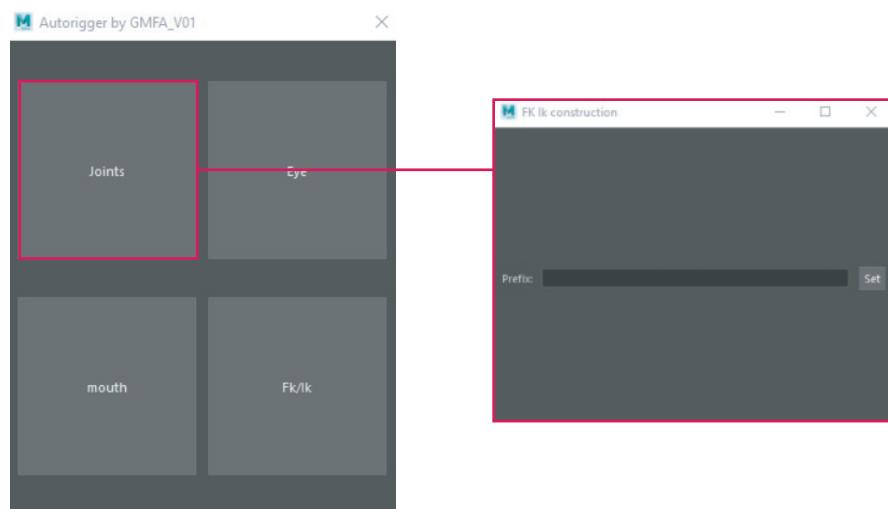
Respecto a la web, como se describió al principio de esta memoria, se tenían planteadas varias ideas para hacerla más completa, pero debido al tiempo que se tenía sumado a que algunas de esas funcionalidades no eran críticas para su funcionamiento.

Por ello, a la hora de escalar la web para sumarle mejoras se harían las siguientes.

- Tutoriales prácticos
- Bot chat básico para dudas
- Ofrecer descarga directa
- Sistema de gestión de usuario con una base de datos
- Creación de tienda en la web
- Crear foro para los usuarios

Mejoras Herramienta

La primera mejora que se buscaría implementar sería terminar la interfaz en la que se estaba trabajando, ya que por tema de tiempos y tamaño del trabajo se decidió no incluirla pero la interfaz de usuario estaba programada como una interfaz con ventanas emergentes dependiendo de la opción que el usuario pulsase.



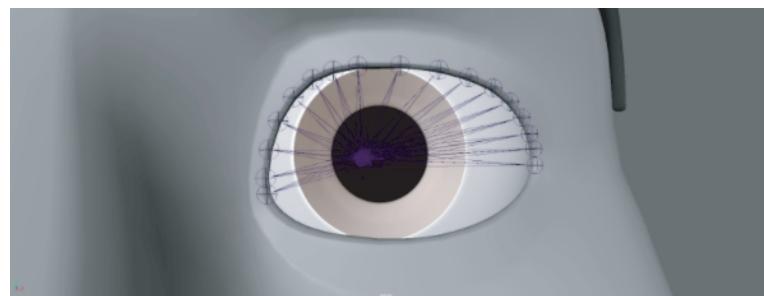
35. Ventanas UI

Al pulsar la ventana al usuario le saldría una ventana emergente diferente dependiendo de la opción pulsada y de los botones y opciones que se necesiten en esa opción, esta ventana no es modal, es decir, que no bloquea la ejecución del programa y el usuario va a poder seguir utilizando todo lo que tiene detrás de estas ventanas. También se añadiría un diseño más llamativo a dicha ventana, cambiando el icono de los botones.

A estas mejoras se añadiría el crear una primera ventana que fuese de login en la que solo podrían hacer login los usuarios dados de alta en la web o los usuarios con licencia.

Con respecto a la funcionalidad de la herramienta se añadiría la construcción de los párpados, que a diferencia del rig de la boca el usuario seleccionará el loop de vértices que quiera para el rig de estos y a partir de esa selección se crearían el resto de objetos del rig.

El rig de las cejas también sería implementado con alguna de las deformaciones que comentamos anteriormente, en este caso podría ser tanto ribbon como wire. Por último se implementaría el rig corporal que aunque no se haya incluido en este trabajo, está a media construcción.



36. Construcción Párpados



37. Rig corporal finalizado

BIBLIOGRAFÍA

Animum

<https://www.animum3d.com/productos/curso-online-rigging-maya-3dsmax-personajes-3d/>

Pyside2 for maya

<https://zurbrigg.com/tutorials>

Bootstrap Docs

<https://getbootstrap.com/docs/4.0/getting-started/introduction/>

W3Schools Docs

<https://www.w3schools.com/html/default.asp>

Stack Overflow Docs

<https://stackoverflow.com/>

Video JS Docs

<https://docs.videojs.com/index.html>