

Avatar hiperrealista para el apoyo al tratamiento de pacientes con problemas de cognición social

Hyper-realistic avatar to support the treatment of patients with
social cognition problems

Trabajo Fin de Grado
Grado en Ingeniería Informática



**VNiVERSIDAD
D SALAMANCA**
CAMPUS DE EXCELENCIA INTERNACIONAL

Memoria final

Alumno: Enrique Hernández Hernández

Tutores: Samuel Marcos Pablos y Francisco José García
Peñalvo

Resumen

Como resultado de investigación a lo largo de los últimos años se ha podido conocer la situación social en la que se encuentran muchas personas que tienen problemas en ciertas habilidades sociales que se requieren para las interrelaciones sociales entre humanos. Estos problemas crean ciertas necesidades en el ámbito terapéutico, las cuales pueden ser apoyadas mediante herramientas software que faciliten y apoyen durante el proceso de ayuda a este tipo de pacientes.

El problema de estos tipos de herramientas es que en ocasiones intentan ser autosuficientes, y que puedan ser usadas directamente por pacientes sin la necesidad de un terapeuta supervisándolo, lo cual en ocasiones puede generar un grado de insuficiencia en cuanto a ayudar al paciente se refiere. Por ello aplicaciones que sean utilizadas por terapeutas psicológicos durante su proceso de apoyo psicoemocional al paciente pueden resultar más prácticas. Otros problemas de las herramientas usadas por psicólogos son a menudo estáticas, lo cual provoca una necesidad de mucho material, y en ocasiones es imposible llegar a casos concretos que desee un terapeuta.

Cada vez la gran mayoría de la sociedad es más consciente de que las personas con problemas en sus habilidades de interacción social son personas totalmente capaces y solo necesitan un poco de ayuda por parte de la gente neurótica para que el esfuerzo que ellos llevan a cabo les permita integrarse en la sociedad. Por ello es un campo en el que el desarrollo de software puede apoyar de manera muy positiva no solo a la gente encargada de apoyar a estas personas, sino también a la propia evolución y adaptación de ellas.

Para el desarrollo de una aplicación orientada a terapeutas se busca centrar el diseño de interfaces en el usuario, para poder resolver necesidades de usuarios finales consiguiendo una experiencia lo más satisfactoria posible para las personas que hagan uso de la aplicación. Además, se busca el mínimo esfuerzo por parte del usuario para usarla y entender todas las funcionalidades de las que dispone la herramienta, sin sacrificar ninguna funcionalidad que se considere esencial o importante en una aplicación basada en el apoyo en tratamientos a personas con problemas en sus habilidades sociales.

En base a todo lo dicho se hace una propuesta de una herramienta de apoyo a terapeutas y psicólogos que permita a sus pacientes ver de una forma dinámica aprender a interpretar la interacción no verbal producida en las expresiones faciales de personas involucradas en una interacción social. Se considera un campo cada vez más importante y que tiene la necesidad de un apoyo constante para poder adaptar a cualquier tipo de persona sin discriminación ninguna. Por ello en la construcción de esta aplicación se aprovecha esta oportunidad de cara a ser una herramienta verdaderamente útil, que va a ser usada por terapeutas y psicólogos en sus labores de apoyo e integración.

Índice

Resumen	2
Índice	3
1. Estado del arte	7
1.1. Introducción.....	7
1.2. Comunicación no verbal.....	7
1.3. Avatares animados.....	8
1.4. Expresividad en interfaces de aplicaciones	9
1.4.1. El papel de las expresiones emocionales en la interacción social	9
1.4.2. Importancia de rostro en interfaces antropomórficas	10
1.4.3. Músculos de expresión en la anatomía facial humana.....	11
1.4.4. Sistema de codificación de acciones faciales	12
1.4.5. Expresiones faciales	14
1.5. Problemas identificados.....	16
1.5.1. Relaciones sociales (cognición social)	16
1.5.2. Soluciones software.....	17
1.6. Objetivos	18
1.7. Conclusiones	19
2. Metodología.....	19
3. Análisis de Requisitos	22
3.1. Búsqueda de requisitos y necesidades	22
3.2. Estudio de herramientas y material	23
3.2.1. Herramientas de implementación	23
3.2.2. Material necesario	24
3.3. Perfil de usuario y entorno	26
3.3.1. Estudio demográfico de terapeutas.....	26
3.3.2. Arquetipo de usuario	27
3.3.3. Entorno de usuario objetivo.....	27
3.4. Requisitos de la aplicación	27
3.4.1. Requisitos de almacenamiento de información	28
3.4.2. Requisitos funcionales.....	28
4. Diseño.....	29
4.1. Soluciones propuestas	29
4.1.1. Reproducción de expresiones faciales.....	29
4.1.2. Representación de expresiones faciales.....	29
4.1.3. Organización de la aplicación	29
4.1.4. Intensidad y duración de la expresión facial	30

4.2.	Elevator Pitch	31
4.3.	Flujo de tareas	31
4.3.1.	Reproducción de la expresión facial.....	31
4.3.2.	Editar expresiones faciales	32
4.3.3.	Cambiar modos de la aplicación.....	32
4.4.	Flujo de navegación de la aplicación.....	33
4.5.	Guías de estilo	34
5.	Prototipado	35
5.1.	Primer prototipo básico	36
5.2.	Prototipo en papel.....	37
5.3.	Prototipo en Unity	37
6.	Implementación.....	38
6.1.	Implementación AUs en Unity.....	39
6.2.	Tratamiento de las unidades de acción en el código.....	41
6.3.	Reproducción de la suma de unidades de acción implementadas: expresión facial.....	43
6.4.	Parámetros de reproducción	44
6.4.1.	Duración	44
6.4.2.	Intensidad	44
6.5.	Modo de reproducción y edición.....	45
6.5.1.	Modo Reproducción	45
6.5.2.	Modo Edición	45
6.6.	Almacenamiento de expresiones faciales	46
6.6.1.	Crear nueva expresión facial	46
6.6.2.	Guardar expresión facial.....	46
6.6.3.	Cargar expresión facial.....	46
6.7.	Unidades de acción antagónicas.....	46
7.	Mejoras implementadas.....	48
7.1.	Mejoras	48
7.2.	Nuevas funcionalidades.....	49
7.2.1.	Intensidad de unidad de acción.....	49
7.2.2.	Opciones de la aplicación.....	50
8.	Conclusión y líneas futuras	51
8.1.	Líneas futuras	51
9.	Bibliografía.....	53
ANEXOS.....		56
Definiciones.....		56
Expresiones faciales Universales		59

Diagramas.....	61
Objetivos	61
Requisitos Funcionales.....	65
Casos de Uso	71
Esquema de casos de uso.....	85
Notas de reuniones con tutor del TFG.....	87

Índice de tablas

Tabla 1: Músculos principales del rostro según su posición	11
Tabla 2: Ejemplos de diferentes unidades de acción (AUs).....	13
Tabla 3: Combinaciones de AUs para formar algunas expresiones faciales.....	15

Índice de ilustraciones

Ilustración 1: Avatar animado de la herramienta MetaHuman Creator.....	9
Ilustración 2: CodeMiko, streamer virtual de Twitch.....	9
Ilustración 3: Principales músculos en un rostro humano	12
Ilustración 4: Regla de medida de los niveles de la intensidad de las unidades de acción.....	14
Ilustración 5: Ejemplo de expresiones faciales universales.....	15
Ilustración 6: Aplicación usada para el aprendizaje de los niños en el ámbito de las relaciones sociales.	17
Ilustración 7: Ejemplos de expresiones faciales	18
Ilustración 8: Diagrama de la Ingeniería de la Usabilidad y la Accesibilidad.....	21
Ilustración 9: Rostro que se utiliza para implementar las unidades de acción	25
Ilustración 10: Animations Rigs del modelo que se está usando para parametrizar las unidades de acción	26
Ilustración 11: Porcentaje de intensidad de una expresión facial	30
Ilustración 12: Wireframe para la navegación básica de la aplicación entre sus diferentes funcionalidades y modos	34
Ilustración 13: Ejemplo de aplicación médica aplicando guía de estilo.....	35
Ilustración 14: Primer prototipo creado en Paint a modo de base de la interfaz de usuario.....	36
Ilustración 16: Primer prototipo de modo edición.....	38
Ilustración 17: Creación de nuevo Animation Clip para el GameObject que contiene los Rigs del rostro del avatar.	40
Ilustración 18: Grabación de un Animation Clip haciendo uso de las transiciones generada automáticamente.....	40
Ilustración 19: Ejemplo de máquina de estados de un Animator Controller en Unity para las animaciones.	41
Ilustración 20: Lista de capas del controlador de animaciones de Unity	42
Ilustración 21: Máquina de estado referente a la AU 1-InnerBrow Raiser que tiene 2 estados parado y el Animation Clip de la AU.....	43
Ilustración 22: Interfaz gráfica del modo reproducción, en el que se ve los elementos como lo son la barra de reproducción y botones para cambiar de modo.	44
Ilustración 23: Unidades de acción antagónicas, que no se podrían realizar simultáneamente.....	47
Ilustración 24: Mapa de unidades de acción incompatibles	47
Ilustración 25: Cuadro para ajustar la intensidad de cada AU perteneciente a una expresión facial	50

1. Estado del arte

1.1. Introducción

Durante el trabajo la atención se centra en las principales características de la comunicación humana, la cual se puede desarrollar en varios medios como pueden ser el habla, expresiones corporales o faciales, dirección de la mirada, etc. En concreto las expresiones faciales es la parte más importante durante el desarrollo de la aplicación. Entender los mecanismos que el ser humano emplea en sus relaciones es fundamental a la hora de desarrollar interfaces que sean fácilmente utilizables por los usuarios, ya que esto permite que el ser humano entienda la interacción con el programa de una manera mucho más natural y familiar, como podría entender una comunicación con otro ser humano.

La interacción es, de forma general, un intercambio de información. En el caso de los humanos, la comunicación surge como un proceso natural que permite intercambiar pensamientos, emociones, preferencias, etc. Lo interesante de este intercambio de información para el proyecto es el aporte de emociones a través de información no verbal por parte de los humanos. En este caso se da total importancia a las expresiones faciales y como estas de una manera no verbal aportan información que deberá interpretar la otra entidad que forme parte de la interacción.

Existen varios modelos que describen la comunicación entre dos entidades, algunos de ellos tienen un carácter más abstracto y general como puede ser el modelo de Shannon [Sh48] que detalla la comunicación entre dos entidades cualquiera y los elementos que conforman dicha interacción, como lo son fuente, código, emisor, canal, mensaje, receptor y contexto. Debido a su aceptación y carácter general se ha utilizado para describir interacciones entre humanos, aunque existen otros que se consideran más exactos. Podemos observar como el modelo de comunicación humana [Be91] modifica elementos con respecto al de Shannon e introduce otros nuevos, como lo es la retroalimentación, la cual es una condición necesaria para que se produzca comunicación entre dos entidades ya que se produce cuando existe cualquier tipo de respuesta por la parte receptora de la comunicación por medio de cualquier método explicado anteriormente.

La retroalimentación es un concepto importante en la comunicación, ya que sin ella solo existe un aporte de información por una de las partes de la interacción, pero no comunicación entre ambas.

1.2. Comunicación no verbal

Existe una gran diferencia entre la comunicación verbal y no verbal, que reside en la forma del mensaje que se envía a través del canal. Mientras que la comunicación verbal el mensaje se transmite a través de una forma lingüística utilizando un idioma que ambas partes de la comunicación entiendan, la no verbal es una forma de comunicación más universal que puede entender más fácilmente un humano sin necesidad de compartir un idioma con la otra parte. (Mast, 2007)

La comunicación no verbal está definida por signos, sonidos, gestos y no necesita de una estructura compleja. Pese a esto, en las comunicaciones humanas se suele usar la comunicación no verbal para acompañar a la verbal y aportar, con esta primera información, que no se dice con las palabras, o cierto contexto a una interacción entre dos entidades, generalmente humanas en este caso.

Si la comunicación ocurre entre dos humanos se le da total importancia a la comunicación no verbal, concretamente a las expresiones faciales. Aunque se ha dicho que tienen un carácter más universal y entendible por todas las personas, en ocasiones habrá gente que tenga limitaciones en el campo de las relaciones sociales que le cueste entender esta comunicación no verbal ya que no va acompañada de una explicación, por ejemplo, personas con autismo o síndrome de asperger que, no son capaces de entender expresiones sarcásticas tan fácilmente, pese a que la expresión facial lo deje claro. A esto se le llama problemas de cognición social, los cuales en muchas ocasiones surgen en esta comunicación no verbal.

1.3. Avatares animados

Una buena práctica para representar información de una manera que entienda un usuario universalmente puede ser usar un medio que muestre la respuesta en forma de las expresiones faciales, para tal propósito se usa un avatar lo más realista posible que sirva de medio para expresar la información no verbal necesaria para el desarrollo de los programas que hagan uso de esta característica.

Las interfaces sociales son aquellas que buscan interactuar con el usuario de una manera inteligente empleando un canal de comunicación que sea lo más natural posible para el humano. Deben de ser capaces de formar parte de comunicaciones sociales. Para lograr estos objetivos es necesario que tenga la capacidad de mostrar personalidad, emoción, personificación, diálogo, etc. No se busca crear una interfaz inteligente que aprenda del usuario, ya que este le aportará la información necesaria para el uso de la herramienta que se busca crear, pero sí que responda mostrando cierta personalidad y emoción mediante el uso de las expresiones faciales que el usuario desee reproducir.

En este campo se han producido grandes avances acompañando a la mejora de la renderización de gráficos y la potencia de los ordenadores actuales. Los avatares animados se han empleado para interactuar con humanos, se pueden nombrar varios ejemplos como lo son la herramienta que se encuentra en desarrollo de Metahuman (*Unreal Engine / MetaHuman*, s. f.) para el entorno de desarrollo de Unreal Engine el cual nos aporta avatares hiperrealistas para el desarrollo de videojuegos, películas animadas y diferentes desarrollos de aplicaciones que requieran de su uso. O bien podemos ver otros ejemplos que cada vez están más de moda en el campo de los *streamers*, los cuales en ocasiones para evitar mostrar su rostro crean avatares virtuales animados en tiempo real que muestra lo que expresa el *streamer* pero a través de uno de estos avatares.

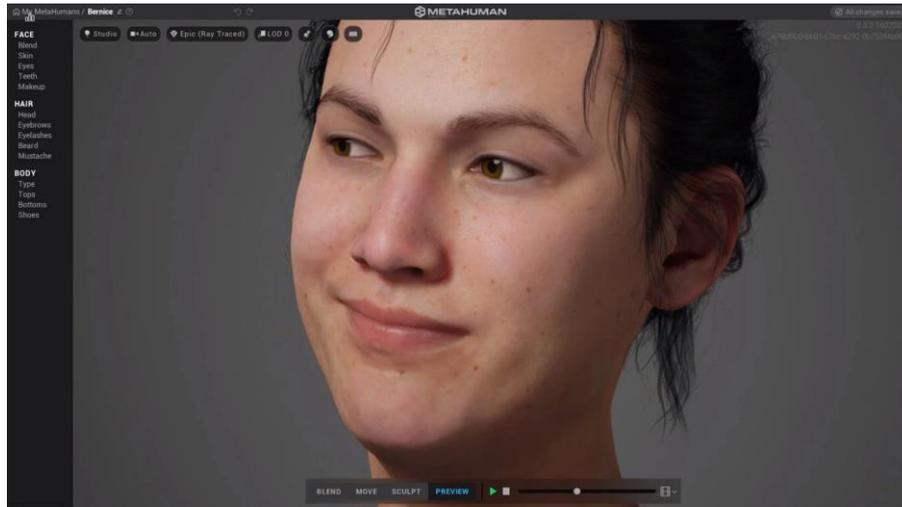


Ilustración 1: Avatar animado de la herramienta MetaHuman Creator



Ilustración 2: CodeMiko, streamer virtual de Twitch

Al final el objetivo de todas estas herramientas es hacer uso de un avatar realista que sea capaz de transmitir ciertas emociones a través de expresiones faciales, en definitiva, es lo que se busca en la aplicación, pero cambiando la manera de generar las expresiones faciales por una que sea asequible para los usuarios que se tengan como objetivo.

1.4. Expresividad en interfaces de aplicaciones

1.4.1. El papel de las expresiones emocionales en la interacción social

El uso de la comunicación no verbal para expresar el estado de ánimo de los interlocutores en las comunicaciones humanas lo convierte en un aspecto importante. Las expresiones de emoción realizadas mediante el movimiento de músculos de la cara o del cuerpo, forman parte de la comunicación humana, y van evolucionando para ayudar en las interacciones humanas de una manera más sencilla y eficaz.

Como ya se ha nombrado anteriormente, es importante la capacidad de los seres humanos de identificar las emociones representadas por un ser humano en una comunicación, dado que estas muchas veces son la explicación de ciertas acciones realizadas por el emisor de esta comunicación no verbal.

Muchas expresiones son innatas mientras que otras se aprenden a lo largo de la vida según estudios psicológicos. En recién nacidos es cierto que son capaces de representar ciertas emociones en los primeros años sin necesidad de haberlas aprendido de nadie en forma de respuesta a estímulos externos. A partir del tercer año de vida los padres son los encargados de enseñar el significado de las diferentes expresiones faciales mediante los diálogos que mantengan con el infante. Esto requiere capacidad de interpretación del pequeño la cual no siempre será la misma, causando esto un posible problema de comunicación durante su vida y siendo necesario más tiempo y herramientas más complejas que simplemente la interacción emocional con sus padres.

1.4.2. Importancia de rostro en interfaces sociales

Las interfaces sociales son aquellas que permiten la interacción con los usuarios tal y como lo harían los seres humanos entre ellos. Esto se puede conseguir mediante un rostro virtual que permita aportar información de respuesta a través de un canal utilizado en las relaciones sociales, como lo es el rostro mostrando las expresiones faciales.

El rostro es el canal que permite mostrar las emociones y estado anímico de una manera más concreta, por ello se debe considerar el rostro un elemento necesario en una interfaz antropológica a la hora de representar la información. El rostro es una zona compleja de la anatomía humana debido al gran número de músculos, y esto la convierte en una parte difícilmente parametrizable en una aplicación.

Este esfuerzo de implementación requiere de importantes razones por las que merezca la pena:

- Las expresiones faciales son universales y no requieren de ningún aprendizaje, sino que se reconocen inmediatamente.
- Es el punto central de la comunicación no verbal en interacciones sociales, con lo cual el usuario estará familiarizado con la interacción de la aplicación.
- El usuario puede comprobar rápidamente si el programa está entendiendo lo que hace, no lleva lugar a error.
- El rostro expresivo es de ayuda en determinadas situaciones, como lo puede ser en nuestro caso la enseñanza de la relación entre emociones y contextos con las expresiones faciales.

En base a esto, este tipo de interfaces podría ser la ideal para el intercambio de información bidireccional con los usuarios permitiendo que él sepa lo que hace exactamente al interactuar con la aplicación, y ésta le responda directamente con las expresiones faciales que desee. Ahora deberemos explicar los principales parámetros que conlustraciónn la apariencia de nuestro rostro a la hora de formar expresiones faciales.

1.4.3. Músculos de expresión en la anatomía facial humana

La cara se trata de una zona compleja en la anatomía, con alrededor de 20 músculos que se combinan para producir cambios visibles en el rostro. Los elementos más significativos son los huesos, músculos, la piel y los órganos exteriores, que son los ojos y la boca. Los músculos son los elementos más parametrizables en nuestro rostro y su contracción origina deformaciones en el rostro, así como movimiento en los órganos y piel que se encuentran en él. El problema que acompaña a los músculos es que existen más de cincuenta en nuestro rostro y su contracción es controlada en muchos de ellos, aunque algunos se mueven de forma involuntaria debido a la contracción y relajación de otros músculos.

Existen músculos principales, que son aquellos de los que se hace uso para generar expresiones humanas al contraerlos, aunque intervengan otros músculos de manera involuntaria al moverse estos principales. Estos músculos se agrupan según en qué zona de la cabeza se encuentran y se dividen en cinco grandes grupos.

Área	Músculos más importantes
1. Cabello y cejas	Occipital y frontal
2. Ojos y párpados	Elevador palpebral superior, orbicular del ojo, corrugador superciliar
3. Orejas	Auriculares anterior superior e inferior
4. Nariz	Piramidal, transverso, dilatador de las fosas nasales, elevadores común y propio, transverso
5. Labios y boca	Cuadrado del labio superior e inferior, canino, triangular, cigomáticos mayor y menor, buccinador, orbicular de la boca, risorio, mirtiforme

Tabla 1: Músculos principales del rostro según su posición

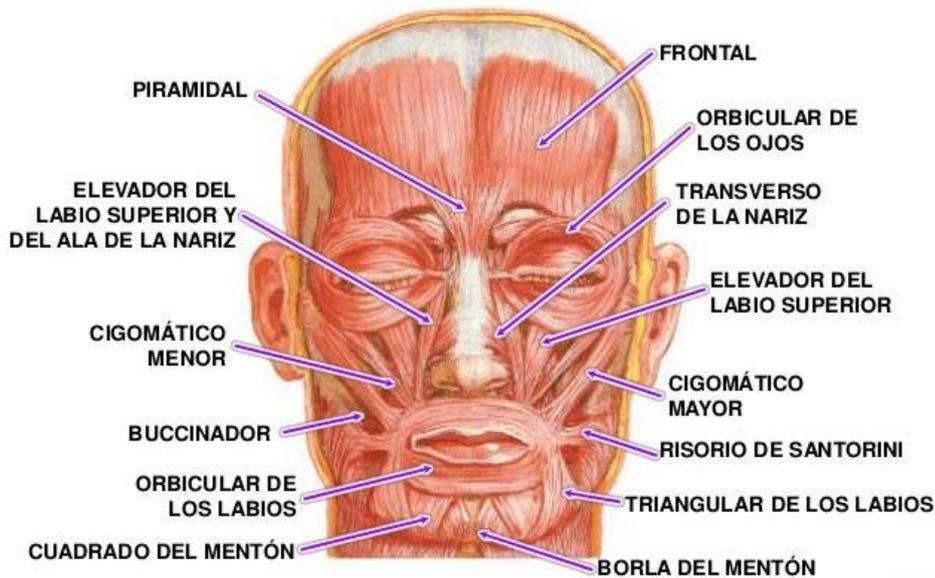


Ilustración 3: Principales músculos en un rostro humano

La mayoría de los músculos cuentan con una parte que los une al cráneo y otra móvil bajo la piel, la contracción de esta última parte origina una deformación en el rostro, como ya se ha dicho estas contracciones se pueden realizar de forma voluntaria o involuntaria. La apariencia del rostro tras estas contracciones es propia del individuo en función de sus características anatómicas y es complicado de generalizar, aunque sí existen ciertas formas genéricas aplicables a la mayoría de las personas.

1.4.4. Sistema de codificación de acciones faciales

El Facial Action Coding System (*Facial Action Coding System (FACS) - A Visual Guidebook*, 2019), o sistema de codificación de acciones faciales, es una descripción del movimiento del rostro humano basándose en el análisis anatómico y visual de la acción facial. El propósito del sistema de codificación de acciones faciales es conseguir una generalización que describa el movimiento del rostro basándose en que músculos intervienen en este, ya sea de manera voluntaria o involuntaria. Siempre teniendo en cuenta la apariencia visual y no los movimientos musculares que no se aprecian visualmente.

Se basa fundamentalmente en trabajos anteriores de dos investigadores, Duchenne y Hjorstjo. En los trabajos de Hjorstjo, se realizó una investigación usando a un conjunto de humanos para las pruebas. Éstos, aprendieron a contraer cada una de las diferentes regiones de manera independiente para poder capturarlo y tras su análisis por distintos observadores, determinar qué movimientos son distinguibles y el efecto visual que provocan en el rostro. Aquellas acciones que, aun siendo producidas por músculos distintos, no sean claramente diferentes según lo que aprecia un humano visualmente, se desechan.

El autor del FACS fue Paul Ekman, aunque su trabajo se basa en otros sistemas de codificación anteriores. Ekman es uno de los autores más renombrados de la psicología en términos de la expresión facial, y fue el encargado de parametrizar y definir el Sistema de codificación facial que se va a usar como base.

Como objetivo, el FACS describe las llamadas unidades de acción, en inglés Action Units o AUs, las cuales representan unidades mínimas de actividad que se genera con el movimiento del músculo que producen cambios momentáneos en la apariencia facial («Sistema de Codificación Facial», 2022) . Las llamadas AUs pueden ser generadas, descritas, reconocidas, a su vez estas acciones mínimas se pueden combinar para describir cualquier expresión facial que se forme en un rostro humano. Para el desarrollo de la generalización descrita por FACS con las unidades de acción, AUs, se determina el número de músculos que pueden ser activados de manera voluntaria sin solaparse con otros músculos, y cada acción de unidad muscular independiente da un resultado visual en el rostro. Como se acaba de explicar las partes mínimas que se diferencian son llamadas unidades de acción y no simplemente unidades musculares, esto es porque en muchas de las unidades de acción los movimientos generados por la contracción de un músculo generan la contracción de otros, como ya se ha explicado. Además, el mismo músculo puede generar varios resultados visuales diferentes, por ejemplo, levantar la ceja en la parte interior o exterior.

Mediante las unidades de acción mínimas y la descripción de cada una de ellas se desarrolla el sistema de codificación facial. En la Tabla 2 se ven algunos ejemplos de las AUs descritas mediante este método.

Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
					
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU 41	*AU 42	*AU 43	AU 44	AU 45	AU 46
					
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
					
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU 15	AU 16	AU 17	AU 18	AU 20	AU 22
					
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU 23	AU 24	*AU 25	*AU 26	*AU 27	AU 28
					
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

Tabla 2: Ejemplos de diferentes unidades de acción (AUs)

Una vez se ha visto la generalización y descripción de los movimientos musculares generando de esta manera las unidades de acción, se debe tener en cuenta que estas no tendrán una transición discreta entre activas o inactivas, sino que hay un cambio continuo desde que los músculos, que intervienen en una unidad de acción, pasan de estar totalmente relajados a totalmente contraídos. Para el tratamiento de esta intensidad de las unidades de acción se desarrolló una medición la cual estará compuesta por 5 niveles siendo el primero el inicio de la unidad de acción y el final llevarla hasta el máximo que nos permitan los músculos involucrados. Dichos niveles son etiquetados con las letras A, B, C, D y E y cada nivel es un intervalo, los cuales en cada nivel tendrán una longitud, siendo los niveles intermedios, C y D, los más amplios y los de los extremos, A, B y E, los más cortos.

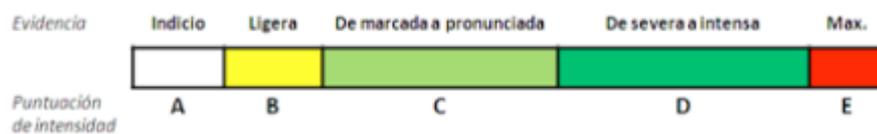


Ilustración 4: Regla de medida de los niveles de la intensidad de las unidades de acción.

1.4.5. Expresiones faciales

Se ha visto cómo se generan unidades mínimas de acción, AUs, que producen cambios visuales en el rostro de los seres humanos. Ahora se debe explicar cómo a partir de estas AUs se generan expresiones faciales complejas, que son las que como se explicó previamente, representan información, mayormente emocional, que muestra el interlocutor en una interacción social.

El primero en proponer que existían expresiones que se corresponden con las emociones primarias y que se podían categorizar fue Charles Darwin. Después de esto Ekman (*Expresiones faciales universales propuestas por Ekman (1972)*. | *Download Scientific Diagram*, 2017) postuló que existen 6 expresiones universales correspondiente cada una de ellas a las seis emociones primarias (alegría, enfado, tristeza, disgusto, sorpresa y miedo), las podemos observar en la Ilustración 5.

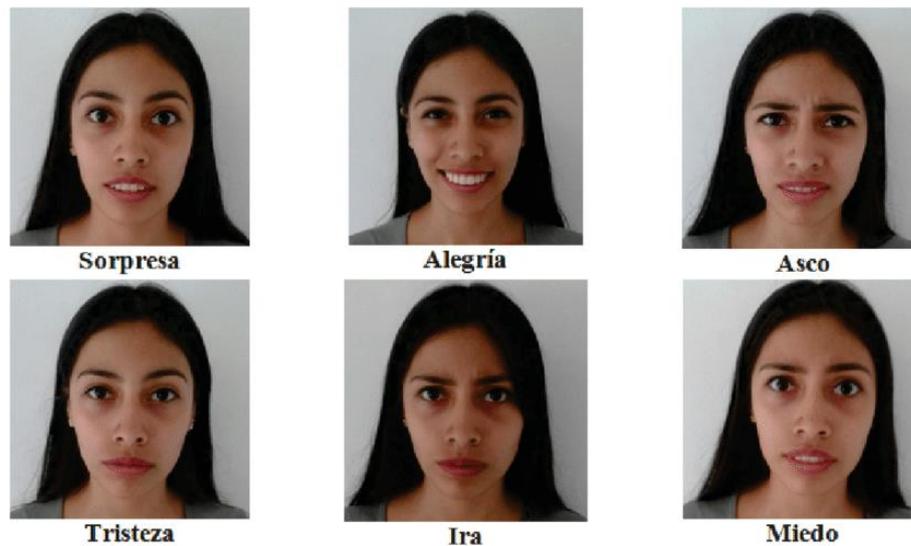


Ilustración 5: Ejemplo de expresiones faciales universales.

La forma de que se generen las expresiones faciales complejas en el rostro es mediante la combinación de movimientos musculares, para ello se usan las AUs, ya que estas parametrizan las contracciones de los músculos faciales en unidades de acción mínimas. Por ello, se pueden parametrizar las expresiones faciales complejas como la combinación lineal de AUs. En la Tabla se puede observar algunas emociones y su representación mediante las unidades de acción.

Emoción ↕	Unidades de acción ↕
Felicidad	6+12
Tristeza	1+4+15
Sorpresa	1+2+5B+26
Miedo	1+2+4+5+7+20+26
Coraje	4+5+7+23
Disgusto	9+15+16
Desprecio	R12A+R14A

Tabla 3: Combinaciones de unidades de acción mínimas para formar algunas expresiones faciales.

Como se observa en la Tabla 3 la felicidad está formada por las AUs 6 y 12, que están descritas como:

- AU6: levantamiento de mejillas
- AU12: estiramiento de la esquina del labio

Cada unidad de acción tiene su propia descripción... Además de esto las AUs pueden ir acompañadas de cierta información extra como el nivel de intensidad hasta el que se

activar cada una, A, B, C, D y E, o si es la parte derecha o izquierda en caso de ser una unidad de acción simétrica respecto del plano sagital del rostro, right y left, R o L.

Es importante recordar que este sistema de codificación no es más que una generalización descriptiva de cómo se generan expresiones faciales, no significa que siempre que un humano muestre una emoción vaya a realizar exactamente las mismas unidades de acción, con la misma intensidad. Por ello cada interacción social es única y sólo podría determinar la especificación de cada representación de las emociones observando al interlocutor que las muestra. («Sistema de Codificación Facial», 2022)

1.5. Problemas identificados

A continuación, se van a tratar los posibles problemas que pueden derivarse de todo lo explicado. Primero en el ámbito de las relaciones humanas, tanto porque se pueden producir estos problemas, como necesidades en su tratamiento que existen actualmente. Además, también se centra la atención en el ámbito software y la falta de herramientas que puede haber dentro del campo del estudio de las expresiones emocionales.

1.5.1. Relaciones sociales (cognición social)

Como se ha visto anteriormente las expresiones faciales son una herramienta básica y necesaria para las relaciones interpersonales e incluso para la supervivencia como seres sociales. Por ello posibles problemas en la fase de desarrollo y de aprendizaje de estas expresiones ya sea los reflejos innatos que se muestran desde bebés a un incorrecto aprendizaje de las aportadas por nuestros padres cuando se va evolucionando y creciendo, puede suponer un problema que marque el futuro de un ser humano.

La cognición social (Adolphs, 1999) es la capacidad de un ser humano de codificar, almacenar y recuperar información de situaciones sociales. La cognición social se refiere a la forma en la que se ve a los demás. Es una herramienta que sirve para entender las emociones, pensamientos, intenciones y conductas sociales de los demás. Todo esto supone una ventaja para poder adaptarse a un contexto y entender mejor lo que ocurre alrededor. Se recibe información que es interpretada para posteriormente a partir de esa información generar una respuesta adecuada.

El origen se produce desde que un bebe recién nacido comienza a interactuar con sus progenitores y está en continuo aprendizaje a partir de experiencias vividas. Esta capacidad requiere en ocasiones empatía y ser capaz de ponerse en el lugar de los demás para lograr entenderlos mejor. En muchas ocasiones ser capaz de entender lo que otra persona quiere decir requiere de ver y comprender cómo se encuentra esa persona en ese momento.

Todos los problemas que existen en el ámbito de la cognición social acarrearán una disfuncionalidad en esta capacidad, producen problemas a la hora de relacionarse socialmente en individuos que tengan estas disfuncionalidades sociales. Los problemas en esta capacidad suelen venir provocados por enfermedades mentales, problemas en el aprendizaje inicial que acarrearán un lastre el resto de la vida o falta de empatía que impide ponerse en el lugar de los demás.

Para la solución de estos problemas existen herramientas y terapias que permiten a los terapeutas ayudar a los pacientes a suplir el déficit que tienen en su cognición social (*La cognición social en niños y cómo la trabajamos con NeuronUP*, s. f.). En muchos casos esta falta de habilidades sociales se suele tratar en la infancia que es una etapa muy importante para el desarrollo de estas. En la Ilustración 6 podemos ver una herramienta usada para favorecer el desarrollo de esta capacidad social que permite a los niños mostrar lo que esperan y piensan de los demás («Herramientas para la rehabilitación de la cognición social en daño cerebral», 2020).



Ilustración 6: Aplicación usada para el aprendizaje de los niños en el ámbito de las relaciones sociales.

1.5.2. Soluciones software

Como se ha visto, en el estudio de las emociones y sus representaciones mediante expresiones faciales existen investigaciones y aportaciones software para los terapeutas y psicólogos que tratan de paliar e incluso solventar esos posibles problemas que pueden existir en pacientes con capacidades más reducidas en el ámbito de la cognición social. Existen muchos softwares, como el visto en la Ilustración 6 que permite a un terapeuta enseñar a los pacientes de una manera más interactiva y dinámica.

Existen desarrollos software relacionados con el estudio de las emociones que buscan enseñar a pacientes mediante juegos el tratamiento de la información no verbal que reciben. Tras una búsqueda de otras herramientas existentes (*Afectiva - Humanizing Technology*, s. f.) se llega a la conclusión de que no existe un software concreto que permita a un terapeuta mostrar una animación de un rostro humano a su paciente para explicarle de qué se trata, y a su vez que sea dinámica y totalmente editable permitiendo al terapeuta mostrar lo que él desee exactamente.

También existen muchas aplicaciones y herramientas que hacen uso de técnicas más estáticas (*Técnicas y programas de estimulación cognitiva*, 2018), como lo pueden ser las que muestran fotos de sujetos de prueba que muestran emociones concretas con unas

unidades de acción predeterminadas y una intensidad concreta sin permitirle al terapeuta ajustarla para explicar o mostrar momentos concretos que el desee de la manera más exacta posible. Vemos por ejemplo fotos de ejemplo que se usan para apoyarse en esta tarea como en la Ilustración 7



Ilustración 7: Ejemplos de expresiones faciales

No existen herramientas útiles que permitan a terapeutas mostrar emociones concretas a personas con problemas de cognición social, creadas a partir de unidades más mínimas de movimientos faciales, las AUs. Esto daría un control total a los médicos sobre qué es lo que quieren enseñar y el cómo.

1.6. Objetivos

No existen aplicaciones en concreto que permitan mediante un avatar hiperrealista crear una animación que sea la suma de las AUs que creen una expresión facial. Siempre se usan fotos que son estáticas o modelos que no resultan nada realista y es más complicado para los usuarios equiparar con rostros humanos.

Durante el desarrollo el objetivo principal es que la aplicación muestre expresiones faciales complejas que el terapeuta que la maneje determine que pueden provenir de emociones, mostrando estas de una forma no verbal, para que un paciente de la aplicación aprenda a interpretar qué quiere decir cada expresión facial sin necesidad de que quien la muestre explique qué quiere decir. Como ya se ha dicho estos pacientes son potencialmente gente con problemas de cognición social.

La idea es hacer algo original, que no exista y que supla un hueco que en ocasiones puede haber cuando un terapeuta quiera mostrar, a su paciente, una expresión en tiempo real de la manera más realista posible, sin necesidad de que sea estático como sería una foto o lo poco personalizable que sería el video de una persona representando una emoción en concreto, teniendo que depender además de este material.

La aplicación tiene dos propósitos, por un lado, el propósito real de la aplicación, para ayudar a gente con problemas de cognición social a interpretar de manera correcta las expresiones faciales; y por otro los objetivos de nuestra interfaz de usuario que cumpla todos los requisitos propuestos y que permita al usuario usar de manera asequible todas las funcionalidades implementadas a través de la aplicación, esta parte se tratará más adelante.

Los objetivos reales de la aplicación son poder reproducir expresiones faciales en el avatar, pudiendo el usuario elegir entre diferentes expresiones que tenga almacenadas y

que pueda reproducirlas parametrizando ciertos factores de la reproducción, como la velocidad, intensidad, ... Además, la aplicación debe permitir a los terapeutas, que deseen dar un uso más intensivo a esta, crear expresiones complejas a partir de las unidades de acción. Aunque la aplicación va a contener las expresiones faciales básicas para que el usuario elija si quiere crear expresiones faciales más complejas o solo hacer uso de las que ya dispone la aplicación.

La idea es una herramienta multifuncional, que muestre las transiciones del rostro facial de un estado normal hacia alguna emoción concreta y una intensidad determinada. Además, también permitirá mostrar momentos concretos de la expresión y ajustar la velocidad a la que se quiere mostrar para poder hacerlo más lento y dar la oportunidad al terapeuta de dar las explicaciones pertinentes.

Una de las necesidades de la aplicación es parametrizar las unidades de acción mínimas, que se han nombrado anteriormente, en un avatar hiperrealista, el cual mediante un interfaz permite al usuario, el terapeuta, combinar estos movimientos musculares para crear expresiones más complejas que posteriormente se puedan reproducir. Esta parte no es trivial, como se ha explicado previamente, no solo es compleja la implementación de las AUs en el avatar, sino que también lo es darle total control al usuario de una manera asequible para él.

Además de lo dicho, también se deberá de parametrizar la intensidad de las expresiones faciales, la cual es un valor de medida asignadas a las expresiones faciales para indicar el nivel de contracción de los músculos involucrados como se explicó anteriormente. La idea es que la aplicación permita al usuario reproducir la expresión con la intensidad que necesite o que desee ver en ese momento, y que fuera variable según lo que el terapeuta necesite.

1.7. Conclusiones

Considerando los apartados previos se llega a la conclusión de que es necesaria una herramienta que dote de medios a terapeutas en su tarea de enseñar a gente con problemas de cognición social para que aprendan a interpretar las expresiones faciales usadas como método de comunicación no verbal. Estas expresiones faciales permiten a los seres humanos aportar información de vital importancia en las relaciones interpersonales, y son de vital importancia para entender situaciones y acciones que ocurren en la vida cotidiana, por ello se considera muy importante y útil la creación de herramientas de este tipo.

2. Metodología

Para el desarrollo de la aplicación se va a centrar el proceso en el usuario objetivo de la aplicación buscando la mayor usabilidad posible, para ello es necesario centrar los esfuerzos desde la fase de conceptualización hasta el lanzamiento, y fases posteriores en centrarnos en lo que necesita el usuario y como es la manera más útil de mostrárselo.

Para basarse en el usuario, se debe hacer un estudio previo del tema y buscar posibles necesidades y objetivos que pueda cumplir el proyecto, una vez hecha esta búsqueda se ha analizado los requisitos que tiene la aplicación, hasta al final obtener los requisitos funcionales que hay que cumplir. Una vez hecho todo lo anterior, se comienza con el diseño basándose en la parte funcional obtenida, se intentan resolver problemas que hayan surgido para implementar toda la funcionalidad en una aplicación usable y centrada en las necesidades y conocimientos del usuario final. Una vez hecho esto se hacen las evaluaciones a partir del diseño creado mediante el uso de prototipos para probar ideas sin necesidad de gastar todo el tiempo necesario en su implementación y se va iterando de prototipos más básicos hacia un aspecto lo más parecido posible al final. Al final se explica de la implementación y de cómo se ha llevado a cabo aquellos aspectos más fundamentales del software.

En cuanto al diseño centrado en el usuario, DCU (Abrás et al., 2004) se sigue el método propuesto por Nielsen, cofundador de Nielsen Norman Group. Nielsen fue la primera persona que comenzó a utilizar el término diseño centrado en el usuario, como herramienta para diseñar las interfaces de usuario de aplicaciones. Siguiendo esta metodología el usuario es el centro de todo, por ello ha sido necesario ir realizando iteraciones y evaluaciones sobre sujetos emulando el usuario final. Además, Nielsen propone de 10 heurísticas para evaluar la usabilidad de las interfaces gráficas de usuario (Experience & Jakob Nielsen, 1994), las cuales son muy útiles para evaluar, a partir del prototipo funcional, que puntos son los óptimos de cara al usuario y cuales pueden mejorarse más.

Concretamente para el desarrollo de la aplicación se ha usado el Modelo de Proceso de la Ingeniería de la Usabilidad y la Accesibilidad, mediante el cual además de aportar usabilidad (Bevana et al., 1991) a la aplicación, también se trata de darle cierto grado de accesibilidad (IWARSSON & STÅHL, 2003) para garantizar que las funcionalidades de la herramienta estén disponibles de una manera asequible y fácil para el mayor número de usuarios posibles (hickeys, 2022). En este caso se tratará de que la aplicación sea accesible por terapeutas y psicólogos que son en su mayoría los que harán uso de la herramienta.

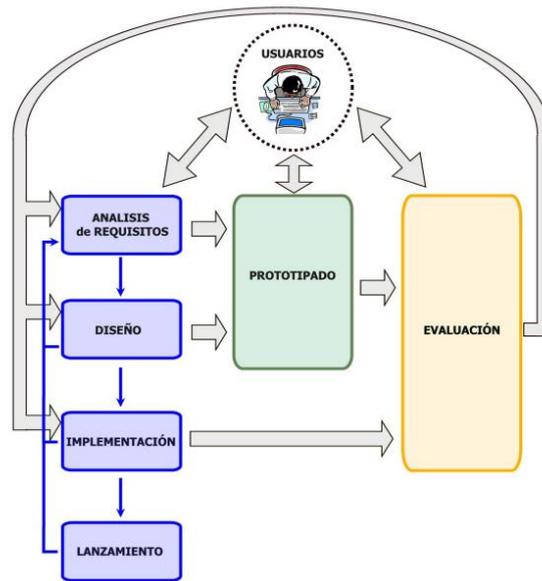


Ilustración 8: Diagrama de la Ingeniería de la Usabilidad y la Accesibilidad

Como vemos en la Ilustración 8 el modelo se basa en un ciclo secuencial e iterativo, ya que es necesario realizar pasos previos a otros, pero de manera iterativa para aplicar evaluaciones a la aplicación mediante las heurísticas, en nuestro caso mediremos la usabilidad mediante la heurística de Nielsen.

El primer paso es a partir de la investigación hecha previamente sobre el tema analizar los requisitos que la aplicación debe de satisfacer y las necesidades que debe cumplir de los usuarios. A partir de este análisis realizado se comienza el diseño de la aplicación, la cual es una de las fases más importantes, ya que es en la que se dé soporte a los usuarios finales, se diseña la parte de interacción del software para comunicar la implementación interna con el usuario, que al final es lo importante para la gente que vaya a hacer uso de la aplicación.

Una vez se han realizado estas dos fases se realizan los primeros prototipos con la información recabada en estos pasos previos. A partir de aquí se itera usando la información aportada por usuarios para mejorar el prototipo y crear otro solucionando los posibles errores que pudiera tener el anterior. Una vez se llega a los primeros prototipos funcionales, la información obtenida por los usuarios evaluados se aplicará para mejorar la usabilidad y experiencia del usuario. Para esto es necesario obtener la mayor información posible de los usuarios evaluados ya sea con comentarios que realicen o preguntas que se les interpongan.

Cuando se dispone de un prototipo que se considera correcto y que cumple con los requisitos necesarios para la herramienta, se comienza la implementación sobre los elementos e interfaz diseñada en ellos. Se realiza la implementación de los requisitos y a medida que se van cumpliendo, se evalúan heurísticamente lo que ya es la aplicación final, mediante las reuniones entre el desarrollador y el tutor que sigue el desarrollo de la aplicación. Usando los resultados obtenidos de estas evaluaciones se realizan mejoras de

la implementación del software y se añaden funcionalidades que sean útiles para el usuario final.

El último paso es el lanzamiento, aunque tras este se sigue recibiendo información de los usuarios que hagan uso de la aplicación, la cual se usa para solventar posibles errores en fases anteriores como puede ser en la implementación, o para mejorar aspectos que se encuentran implementados pero los usuarios consideran que se pueden mejorar para una mayor satisfacción final. El ciclo de desarrollo de la aplicación se continúa hasta que el desarrollador considere, ya que mediante las iteraciones se puede seguir mejorando e implementando nuevas funcionalidades que soliciten los usuarios finales.

Todos estos pasos siempre se siguen de manera iterativa, buscando estar en continuo desarrollo a la vez que se realizan mejoras sobre los pasos previos ya realizados. Este modelo de ingeniería de la usabilidad y accesibilidad se sigue de aquí en adelante hasta finalizar el desarrollo de la herramienta.

3. Análisis de Requisitos

3.1. Búsqueda de requisitos y necesidades

A partir de toda la investigación hecha hasta el momento se ha recabado bastante información relevante en referencia a los requisitos y necesidades que debe tener una aplicación que vaya a ser usada por un terapeuta que requiera de una herramienta que le ayude a enseñar a pacientes con problemas de interacción social el significado de las expresiones faciales y porque pueden producir y que significan.

Como requisito principal la aplicación deberá permitir como mínimo generar las seis expresiones faciales universales (alegría, enfado, tristeza, disgusto, sorpresa y miedo). Es importante recalcar que se estas expresiones se usan como punto de partida ya que son en muchas ocasiones la base para crear otras expresiones más complejas, ya sea quitando o añadiendo AUs que conforman la expresión. Este requisito va a acompañado de la necesidad de contar con un avatar 3D que cuente con un rostro facial lo más realista posible, y que permita de una manera asequible implementar las unidades de acción en él.

Otro de los requisitos de la aplicación es que permita al usuario interactuar con ella, ya que se busca algo dinámico que tenga total control, para ello se debe de tener en cuenta cuál será nuestro perfil de usuario, o sea las personas que potencialmente la van a usar.

Aparte de lo nombrado, la herramienta debe permitir al usuario no solo generar las seis expresiones universales en el avatar, sino también implementar nuevas expresiones más complejas, ya sea a partir de las existentes o desde cero, usando siempre las unidades de acción que se hayan implementado en el avatar. Esto tiene la necesidad de implementar el máximo número de unidades de acción en el avatar, para que el usuario disponga de la mayor variedad posible que desee de todo el catálogo de AUs disponibles según el sistema de codificación de acciones faciales (Oster & Ekman, 1981).

3.2. Estudio de herramientas y material

Como ya se ha dicho, no se han encontrado herramientas que realicen una tarea similar a la idea que se ha propuesto. Por ello existe la necesidad de crear un sistema totalmente nuevo e implementarlo desde cero. La idea es crear una aplicación multimedia que permita visualizar un avatar con miles de polígonos para que sea lo más realista posible. Además, mediante una interfaz gráfica permitirá al usuario navegar por la aplicación y efectuar cambios en el avatar.

Para realizar la implementación de las unidades de acción mínimas usaremos material ya existente en el cual nos basaremos, ya que existen estudios ya realizados y material existente en la web que mediante fotografías o videos cortos de seres humanos que mueven los músculos que se activan para cada una de las AUs descritas en el sistema de codificación de acciones faciales. Con lo cual la única preocupación es qué herramienta usar para su implementación.

Entonces para lo dicho anteriormente se necesita herramientas en las que se implementen la parte de la interfaz y cómo el usuario se comunicará con la aplicación; y la funcionalidad relacionada con las acciones de unidad y la combinación de estas para que se generen las expresiones faciales. Sumadas a estas herramientas de implementación se necesita el material para la codificación de las AUs, como lo es el avatar hiperrealista que se tratará de usar.

3.2.1. Herramientas de implementación

En este apartado se plantea qué herramientas usar para implementar los requisitos propuestos. Se puede hacer uso de una herramienta que permita el tratamiento de modelos 3D para lo referente a la parte funcional. y que el usuario pueda mediante una interfaz comunicarse con la aplicación para que produzca cambios en el modelo 3D. Una buena herramienta para tal propósito es un motor gráfico usado normalmente para el diseño de videojuegos o películas de animación. Estas herramientas permiten a los desarrolladores modificar propiedades de los modelos 3D como lo pueden ser, posiciones, rotaciones, escalas, etc. mediante la interfaz que ofrece a los usuarios.

Una de las opciones es Unreal Engine 4 (*Unreal Engine | The Most Powerful Real-Time 3D Creation Tool*, s. f.) y su herramienta de creación de humanos hiperrealistas, llamada MetaHumans Creator (*Unreal Engine | MetaHuman*, s. f.). Esta opción es muy conveniente ya que MetaHumans Creator aportaría el modelo hiperrealista ahorrando así un proceso de diseño o búsqueda de un rostro humano modelado en 3D, el problema de esta herramienta es que en principio solo puede usarse en Unreal Engine, lo cual permitirá al diseñador obtener un avatar hiperrealista en caso de usar esta herramienta. El inconveniente es que sobre esa herramienta el programador no tenía tanto conocimiento como el que dispone sobre Unity, lo cual le permite tener mayor soltura a la hora de prototipar e implementar en Unity.

Otra opción es desarrollar el programa sobre el motor gráfico Unity (Technologies, s. f.), que, pese a que no cuenta con una herramienta de creación de avatares hiperrealistas, como si lo hace Unreal Engine, sí que es una herramienta muy versátil, con muchísimo contenido.

La elección es usar Unity, debido a que es un motor mejor conocido por el programador y más versátil a la hora de crear interfaces más simples y aplicaciones que no requieran tantos requisitos hardware de los ordenadores en los que se use la aplicación. Además, Unity usa como lenguaje C# que es un lenguaje de programación multiparadigma mucho más asequible y cómodo de usar que C++, que es el lenguaje de programación utilizado por Unreal Engine.

Pese a que MetaHuman Creator no es compatible directamente con Unity se trató de importar los modelos de MetaHuman a Unity, ya que la herramienta podría ser muy útil para facilitar la tarea de implementar las unidades de acción en nuestro avatar. Para ello se utilizó Quixel, que es una herramienta que aporta contenido 3D para programas de tratamiento de modelos 3D, como lo son motores gráficos o programas de modelado 3D (Blender, Maya, AutoCAD). Quixel almacena la biblioteca de los modelos de humanos creados en MetaHuman Creator, además de aportarnos algunos de base. Desde Quixel se exportó uno de los humanos digitales a un programa de modelado llamado Maya. Tras este paso existe un plugin llamado FBX Exporter que permite exportar cualquier modelo 3D que esté en Maya a Unity.

En esta secuencia de acciones para lograr exportar el humano digital desde Quixel a Unity han surgido varios problemas. El primer problema es en el paso inicial a la hora de exportar de Quixel a Maya el modelo, una vez se han instalado las herramientas necesarias para importarlo hacia Maya, al abrirlo en este el modelo no tenía texturas, se veía un avatar en blanco sin el color de la piel, ojos, labios, etc. Tras un tiempo intentando esta solución se deshecho debido a los problemas y la clara incompatibilidad entre MetaHuman y otros programas que no sean Unreal Engine. Por ello se descarta y es necesario buscar un avatar lo más realista posible compatible con Unity.

3.2.2. Material necesario

Para implementar los requisitos del proyecto es necesario buscar un modelo 3D compatible con Unity, que además disponga de *rigging* suficientes, para modelar correctamente las unidades de acción en el rostro.

El *rigging* es el proceso por el cual se crea una estructura de articulaciones y huesos que permite deformar un modelo 3D, permitiendo de esta manera a los animadores modelar el movimiento del cuerpo, músculos, expresiones faciales y demás movimientos corporales de los avatares. Esta técnica se puede aplicar además de en humanoides digitales en animales, vehículos con engranajes y puertas y cualquier modelo que tenga algún tipo de articulación o movimiento asociado a él.

En definitiva, el objetivo es buscar un modelo 3D de un rostro con facciones humanas, lo más realista posible y que cuente con el *rigging* necesario para modelar las unidades de acción del sistema de codificación de expresiones faciales que deseemos implementar. Como esto aún no es algo que sea exacto ya que no se sabe cuántas AUs se van a implementar, se actuara como si se fueran a implementar la mayoría de ellas, siempre poniendo como mínimo que se permita implementar las expresiones faciales universales.

En la búsqueda se han encontrado bibliotecas de modelos 3D con los músculos faciales necesarios *riggeados*, el problema es que muchos de estos son de pago, y el objetivo del proyecto es intentar obtener un coste mínimo y estos modelos pueden ascender a doscientos dólares. Tras esto, se busca en la Asset Store de Unity (*Unity Asset Store - The Best Assets for Game Making*, s. f.), que es una biblioteca de contenido para el motor, aquí hay una sección dedicada a modelos 3D, como lo pueden ser avatares humanos.

La propia empresa de Unity ha creado un corto a modo de demo técnica de uno de sus motores y publicó todo el material usado en él. Uno de ellos es el modelo 3D del protagonista, el cual está totalmente animado, tanto el cuerpo como la cara. Es un avatar muy realista ya que sirvió para mostrar todo el potencial gráfico del motor. Con lo cual este avatar puede servir para animar su rostro e implementar las unidades de acción, además de ser totalmente compatible con Unity (*The Heretic*, 2022).



Ilustración 9: Rostro que se utiliza para implementar las unidades de acción

Ahora se debe importar a Unity, lo cual es un proceso bastante sencillo, ya que al estar en la Asset Store de Unity se puede hacer desde el propio motor, a través de una pestaña para acceder a la biblioteca personal y desde allí elegir un Asset e importarlo al proyecto. Una vez importado lo importante es saber si los *Animations Rigs* de los que dispone el modelo permiten crear las expresiones faciales básicas. Los *Animations Rigs* (*Pablos & Pablos*, 2011) son los puntos de un modelo que nos permite animarlo aplicando movimiento en estas zonas.

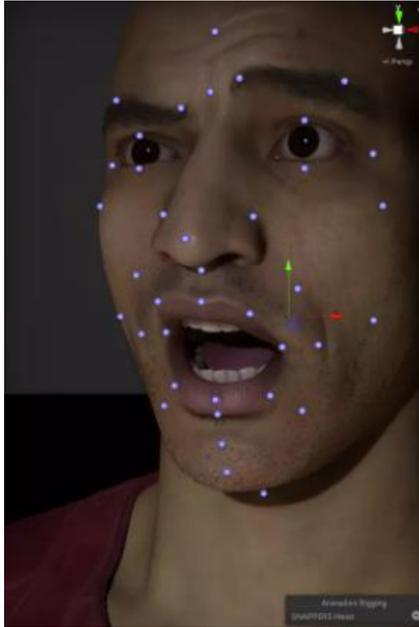


Ilustración 10: Animations Rigs del modelo que se está usando para parametrizar las unidades de acción

Tras la investigación ya realizada sobre FACS y leer tesis doctorales sobre el tema (Pablos & Pablos, 2011), se concluye que los Rigs de este modelo eran suficientes e incluso, en caso de ampliación, permitirán crear nuevas expresiones faciales por parte del terapeuta.

En conclusión, la herramienta de desarrollo de la que se va hacer uso es Unity, sirviendo este motor para implementar tanto la parte funcional de la aplicación, como la parte de la interfaz gráfica que permita al usuario comunicarse con la aplicación y recibir información de esta. Sumado a esto se va a usar el modelo “The Heretic: Digital Human” para implementar sobre él las unidades de acción y poder mostrar las expresiones faciales resultantes.

3.3. Perfil de usuario y entorno

Lo primero es identificar el tipo de usuario que potencialmente va a usar la aplicación y realizar un arquetipo del usuario. El principal objetivo de la aplicación son terapeutas y médicos psiquiatras que están tratando con pacientes con problemas de cognición social. Para ello hablaremos brevemente sobre estudios demográficos de la población para poder concluir edad y demás características que puedan afectar al diseño de la aplicación.

3.3.1. Estudio demográfico de terapeutas

Se realiza un breve estudio de la población objetiva para la aplicación buscando descubrir sus necesidades y conocimientos en función de la edad, estudios, y demás características que pueden resultar de interés (*El futuro de la profesión de psiquiatra en España*, s. f.).

Según estudios realizados en España hay un 20% de los psicólogos mayor de 60 años en el sistema público, lo cual hace ver una edad media avanzada en este sector. Esto significa

que las necesidades de nuestros usuarios son una aplicación más sencilla y familiar, buscando interfaces similares a otras que ya hayan usado previamente.

Aunque es cierto que la gente que se encuentre en estos puestos cuenta con un alto nivel de estudios, esto no quiere decir que tengan conocimientos en el ámbito informático, por ello se debe favorecer lo máximo posible la simplicidad de la aplicación. Cada vez aumenta más el tiempo de formación, pero el número de psicólogos y psiquiatras sigue siendo muy escaso, lo cual provoca que tengan una mayor carga de trabajo. Todo lo dicho significa que disponen de un tiempo muy limitado, tanto estudiantes como trabajadores del campo.

3.3.2. Arquetipo de usuario

En base al estudio demográfico realizado, se debe crear un entorno asequible para ser usado por personal no cualificado en el terreno de la informática y en un alto porcentaje gente de mayor edad que no dispone del conocimiento adquirido por gente de generaciones posteriores a ellos.

Debe de ser una aplicación familiar para terapeutas y personal médico, además de simple, esto es un problema de diseño ya que la información debe de permitir al usuario hacer todo de una manera liviana, se tratará más adelante.

Estas personas cuentan con tiempo limitado y necesitan de aplicaciones intuitivas, para no perder tiempo en aprender a usarlas, que mediante el uso de pocos clics lleguen a su objetivo sin tener que hacer un uso intensivo del programa.

3.3.3. Entorno de usuario objetivo

Para el entorno objetivo que pudiera mover la aplicación serian ordenadores que se pueden encontrar en hospitales, centros psiquiátricos y consultas médicas, pero esto debido a la potencia necesaria para cargar un modelo 3D muy complejo, e hiperrealista será complicado desde el punto de vista del rendimiento. Tal vez sea necesario equipos con unas mejores prestaciones para mover la aplicación, aunque una vez desarrollados las primeras versiones se puede abordar en qué tipo de ordenadores funciona correctamente, y posibles formas de optimización en máquinas con menos prestaciones.

3.4. Requisitos de la aplicación

A partir de los objetivos básicos de la aplicación que se han especificado y de las herramientas que se han elegido finalmente, se puede concretar unos requisitos funcionales que serán los que apliquemos en el diseño para continuar con el desarrollo.

A continuación, se ven los dos tipos de requisitos funcionales, obligatorios basados en los objetivos básicos de la aplicación a los cuales se les asigna una importancia alta y los requisitos opcionales que se le asigna una importancia media o baja según se consideren de mayor interés en la aplicación. Estos requisitos de menor importancia son posibles ampliaciones. Sumados a los los requisitos funcionales también se describen los de

almacenamiento de información, que son aquellos tipos de datos que serán necesarios crear y usar para representar la información necesaria de la aplicación.

3.4.1. Requisitos de almacenamiento de información

Para almacenar la información es necesario crear requisitos de información, los cuales indiquen qué datos concretamente va a contener para así poder hacer un tratamiento de ellos más adelante. La aplicación tiene la necesidad de permitir al usuario almacenar en disco expresiones faciales para que pueda guardar una expresión sin que se pierda cuando cierre la aplicación. Por ello es necesario un requisito de almacenamiento de información que guarde lo referente a las expresiones faciales. Además de eso se debe definir de qué forma se va a guardar las unidades de acción.

Lo primero es definir la unidad de acción, que es el tipo más básico, para ello hay que tener en cuenta que este dato es usado por el sistema para identificar las diferentes unidades de acción. Las AUs son representadas por una cadena de texto que contenga el nombre de cada una y que sea único, para de esta manera se puedan diferenciar.

El segundo requisito de información es la expresión facial, la cual debe de ser un tipo de dato complejo y permitir almacenarse. La expresión contiene un nombre que la identifica y es propia de ella para que el usuario y sistema sepan con qué expresión están tratando en cada momento. Además, contiene una lista del tipo de dato que se han definido para las unidades de acción, esta lista es variable en tiempo de ejecución y contiene en todo momento lo que el usuario haya elegido como las AUs que desea que formen parte de la expresión en cuestión.

3.4.2. Requisitos funcionales

A continuación se detallan los requisitos funcionales que la aplicación debe satisfacer para cumplir las necesidades y objetivos planteados. La aplicación debe reproducir y pausar expresiones faciales, así como seleccionar qué punto de la reproducción quiere elegir el usuario. Además, se puede determinar el tiempo que se desee que tarde la reproducción de las animaciones. El usuario también debe tener la posibilidad de cambiar la intensidad máxima de la expresión facial que se reproduce, para ajustarla a las necesidades de sus pacientes.

Por otro lado, se permite al usuario crear nuevas expresiones faciales, editarlas añadiendo y quitando las unidades de acción de la expresión en cuestión. También se deben poder guardar en disco, y cargar de disco las que hayan sido guardadas anteriormente.

4. Diseño

4.1. Soluciones propuestas

En esta fase se comienza a dar forma a todas las ideas y objetivos propuestos para la aplicación y los posibles problemas que puedan haber surgido a causa de los requisitos de cara a la funcionalidad y usabilidad por parte del usuario.

Lo primero que se hace es proponer una serie de ideas para algunos de los requisitos que necesiten de su propio diseño. La idea es proponer posibles soluciones para implementar los requisitos en la aplicación, y una vez se comienza la fase de prototipado poder aplicar en estas las ideas y ver cuál es la más correcta de cara al usuario mediante evaluaciones.

4.1.1. Reproducción de expresiones faciales

La base de la aplicación es que un usuario pueda reproducir expresiones faciales, con lo cual, lo primero permitir al usuario utilizar la base sin necesidad de que use todo lo demás. Por ello la solución de cara a los usuarios, es añadir en la aplicación de lanzamiento las expresiones universales, las cuales son las 6 más utilizadas y básicas: alegría, tristeza, ira, asco, miedo y sorpresa, pudiendo ampliar finalmente a alguna más que se considere muy importante. De esta forma los usuarios pueden elegir si le basta con esas expresiones faciales y no perder el tiempo creando otras nuevas o editando las ya existentes.

La reproducción debe ser intuitiva y familiar para el usuario. Para ello lo mejor es basar la interfaz en aplicaciones y herramientas ya existentes de reproducción de videos. Esto evita que la aplicación sea compleja y que le resulte familiar a todo el mundo que haya usado una de estas herramientas de reproducción previamente.

4.1.2. Representación de expresiones faciales

Como ya se ha mencionado, uno de los puntos más problemáticos y a la vez importantes para el desarrollo de la herramienta es la implementación de las unidades de acción. Como ya se habló anteriormente una expresión facial compleja es la suma de diferentes AUs en el rostro facial, por ello a la hora de implementarlo para que el usuario pueda crear nuevas expresiones, el selecciona de una lista de unidades de acción mínimas implementadas por el desarrollador. Las AUs seleccionadas serán parte de la expresión facial creándose para cada expresión una nueva lista de unidades de acción, el usuario además de poder añadir AUs puede quitarlas de las expresiones, pudiendo de esta forma personalizar mediante las unidades de acción una expresión facial compleja.

4.1.3. Organización de la aplicación

La aplicación de manera general tiene dos funcionalidades principales, por un lado, reproducir las expresiones faciales, y por otro editar y crear nuevas expresiones. El problema de esto es que se debe tener en cuenta que no se dispone de espacio ilimitado y hay que poner la información justa que no moleste ni agobie al usuario. Con lo cual se

debe explorar soluciones que permitan separar de alguna manera la información u ocultarla momentáneamente en función de lo que el usuario desee en cada momento.

La primera solución es crear varias pantallas y que el usuario pueda navegar entre ellas en función de lo que desee hacer. Están disponibles el modo reproducción el cual permite que el avatar reproduzca las expresiones faciales complejas que tengamos para mostrarlas al paciente. Este modo contará con las seis expresiones faciales universales y las que el usuario haya decidido crear. El segundo, es el modo edición, desde el cual el usuario puede crear nuevas expresiones faciales o modificar las ya existentes añadiendo o retirando AUs que ya están parametrizadas en el avatar, listas para que el terapeuta las use.

Otra posible solución al problema es que todos los paneles fueran ocultables por el usuario y que en función de si desea reproducir o editar, el pudiera elegir qué paneles en concreto desplegará en cada momento. Esta solución tiene la ventaja de que nunca habría funcionalidades redundantes, sino que cada función tiene su panel y si deseas utilizarla debe desplegar el panel, pero tiene la contra de que es más complejo, ya que un usuario inexperto se encontraría con muchos paneles en un inicio y no sabría cual usar y cuál cerrar.

4.1.4. Intensidad y duración de la expresión facial

Otro problema generado a nivel funcional de la aplicación es que en el momento de reproducir las expresiones faciales, el usuario puede elegir la intensidad que desee que tenga la expresión, y el tiempo que quiere que dure, pero esto supone un problema, ya que la intensidad no es más que el punto máximo al que llega una expresión, y la duración afecta a los segundos que tarda en reproducirse dicha expresión.

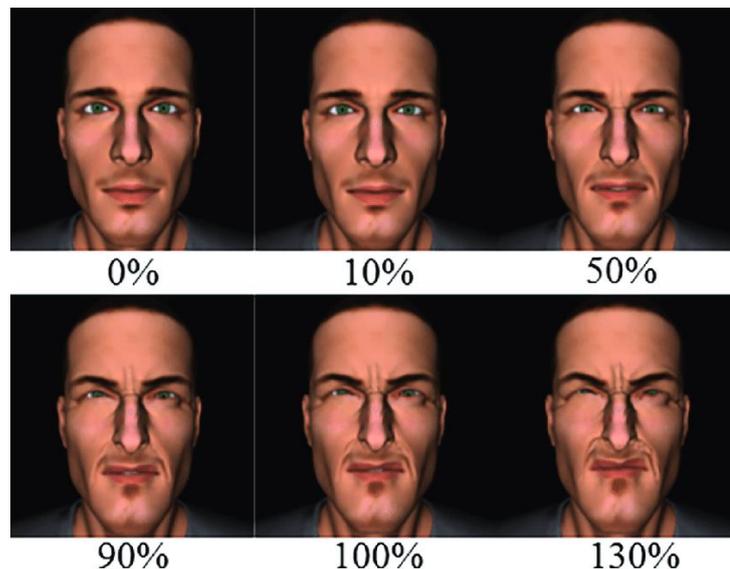


Ilustración 11: Porcentaje de intensidad de una expresión facial

En la Ilustración 11 se ve cómo en función del porcentaje el movimiento que realizan los músculos en la expresión es más forzado o relajado. El problema se plantea cuando el usuario pone una intensidad del 50% pero quiere que dure el doble de lo que duraría

normalmente, entonces si sobre el 100% de la expresión queremos que haga la mitad de intensidad, pero luego esa mitad dura el doble en teoría debería de durar lo mismo que la expresión a la máxima intensidad, pero llegando hasta la mitad de la contracción de los músculos involucrados. El problema es que tal vez el usuario quería que durara el doble de lo normal sin importarle la intensidad que aplique. Con lo cual la opción más apropiada es que sean cosas independientes, si el usuario pide que dure el doble la animación que se reproduzca durará el doble de lo que dura normalmente, y luego si quiere una intensidad del 50% entonces con la duración ya elegida, el máximo de la animación será la mitad de la contracción de los músculos.

4.2. Elevator Pitch

A continuación, el objetivo es crear una frase a modo de resumen del proyecto, que sea asequible para mostrar la idea general y los puntos más importantes del proyecto en cuanto a su funcionalidad y diseño, en un breve tiempo. Esta frase resulta muy útil para hacer un resumen de manera muy general pero lo suficientemente explicativa para definir de lo que es capaz la herramienta. Lo positivo del elevator pitch sobre el proyecto es que tiene en cuenta los usuarios objetivos, ya que son gente que no dispone de mucho tiempo para gastar y es de mucha utilidad el poder explicar y resumir el software en el menor tiempo posible.

La aplicación permite a partir de la suma de movimientos musculares, ya implementados, crear expresiones faciales totalmente personalizables, las cuales se pueden reproducir eligiendo la intensidad y duración de estas para poder ayudar a los pacientes a identificar claramente los distintos estados de ánimo que tiene una persona con la información no verbal que aportan las expresiones faciales.

4.3. Flujo de tareas

Una vez propuestas ya las ideas y sabiendo cómo se van a implementar las acciones, se comienza a diseñar el flujo de las tareas que el usuario puede realizar. Para ello se parte de los requisitos funcionales ya establecidos anteriormente, y apoyándose en el diseño realizado para saber las acciones que puede realizar el usuario en cada momento. Aquí se usan los casos de uso, que son una secuencia de acciones que el usuario puede realizar que tendrá un resultado observable por el usuario. Estos casos de uso forman un flujo de tareas que permite al usuario navegar a través de la aplicación.

Estas tareas se pueden clasificar en varios grupos en función de qué usos puede dar el usuario de cada uno de ellos: reproducción de la expresión facial, editar expresiones faciales, cambiar modos de la aplicación.

4.3.1. Reproducción de la expresión facial

En esta clasificación están los casos de uso que permiten reproducir y pausar la expresión en el modo reproducción. Estos casos de uso se implementan sobre un botón que cambia

entre uno u otro en función del estado en el que se encuentre la reproducción. Si la aplicación está reproduciendo la expresión facial el botón permite pararla y en caso de que esté parada el botón permite reanudarla.

Además, mediante un slider el usuario elige la intensidad de la reproducción, además de poder activarlo y que se vaya reproduciendo con el tiempo, se puede mover el slider para ir a un punto en concreto de la expresión facial. Sumado a esto la herramienta permite al usuario elegir un tiempo de duración el cual será el que tarde desde el inicio de la reproducción de la expresión hasta el final, para que el terapeuta pueda elegir si quiere que el avatar contraiga los músculos que forman parte de la expresión de una manera más lenta o rápida en función de las necesidades que tenga a la hora de enseñarlo a su paciente.

4.3.2. Editar expresiones faciales

Dentro de este campo se encuentran las tareas referentes a editar la expresión que se está actualmente editando. Para ello hay un caso de uso que permite añadir de la lista de unidades de acción totales a la lista de AUs que forman parte de la expresión actual mediante doble clic del usuario en las ranuras que contiene cada unidad de acción. De la misma manera, pero de forma inversa permite quitar las unidades de acción de la expresión que se está editando para que vuelvan a la lista de AUs totales.

Además, existen tres casos de uso referentes al tratamiento de las expresiones faciales en disco, los cuales se implementan en un botón cada uno. El primero es crear una nueva expresión facial lo cual creará el tipo de dato y lo inicializará con la información necesaria para guardarlo en disco cuando sea solicitado por el usuario. El segundo es guardar la expresión facial el cual añadirá a la lista de expresiones faciales ya guardadas en disco la que se está actualmente editando. Y el tercero es cargar una expresión facial para lo cual el usuario seleccionara entre la lista de expresiones almacenadas en disco cuál de ellas quiere cargar para editar o reproducir.

4.3.3. Cambiar modos de la aplicación

Para esta funcionalidad se necesitan dos casos de uso. El primero sirve para cambiar a modo edición, acción que solo se puede llevar a cabo en caso de que el usuario se encuentre en el modo reproducción la aplicación, y que cambia la información mostrada en la interfaz para que el usuario pueda editar la expresión facial actual, así como guardarla, crear una nueva o cargar otra.

El segundo permite al usuario cambiar al modo reproducción, sólo puede llevarse a cabo esta acción si la aplicación se encuentra en el modo edición, y muestra al usuario en pantalla la información necesaria para reproducir la expresión, como lo es el panel que contenga el slider de reproducción y duración y los botones de reproducción/pausa y demás modos de reproducción.

4.4. Flujo de navegación de la aplicación

El flujo de la aplicación debe seguir las ideas propuestas durante el diseño para su posterior implementación en prototipos y en la aplicación final. Se debe diseñar el flujo mediante el que el usuario se moverá por los distintos entornos, pantallas y paneles de la aplicación mediante la interfaz. Para diseñar el flujo, lo primero es describir brevemente cada posible estado o pantalla en el que puede estar la aplicación, para luego decir estando en cada uno a cuáles puede navegar.

Existen dos diferentes modos, edición y reproducción. En el modo edición el usuario puede editar una expresión que esta activa en ese momento, guardarla, crear una nueva, y estar introduciendo el nombre de la expresión facial nueva, o cargar una expresión guardada en disco y seleccionando cual desea cargar o eliminar de todas las disponibles que están almacenadas. En el modo reproducción las animaciones pueden estar paradas o activas.

Se empieza por el estado en el que se encuentra la aplicación inicialmente, estando en el modo reproducción, con la expresión parada, el usuario en este modo tiene la opción de pulsar un botón que le permita reanudar o pausar la reproducción de las animaciones. Mediante el uso de otro botón podrá cambiar al modo edición de la aplicación, desde allí se puede crear una nueva que lleva a un nuevo estado la aplicación, en el que espera que el usuario introduzca el nombre de la nueva expresión facial, pudiendo confirmar o cancelar crearla, ambas opciones devuelven al usuario al modo edición. Desde aquí se puede abrir una nueva expresión de disco, lo cual abre un panel que muestra todas las expresiones almacenadas en disco para que el usuario pueda elegir si cargar alguna de ellas o borrarlas, mediante clic en el slot de cada expresión o en un botón que permite eliminarla cada una de ellas. Si el usuario decide cargar una, la aplicación vuelve al modo edición, pero con unidades de acción en la expresión correspondiente a lo que haya cargado. Si decide eliminar una expresión tiene que confirmar la acción, pudiendo también cancelarla y volviendo con ambas opciones al panel de carga de expresiones, panel desde el cual también puede volver al modo edición sin necesidad de hacer nada, tan solo cerrando este panel. Por último, desde el modo edición se permite al usuario cambiar el nombre de la expresión facial que se está editando.

Una vez definido todo se puede crear un pequeño esquema visual que sirve como un prototipo muy básico en el que se ve como navegar a través de la aplicación, en la Ilustración 12 se ve como con el uso de botones y clics del usuario se puede cambiar los diferentes modos de la herramienta y usar las funcionalidades principales como lo son reproducir y parar o añadir y quitar unidades de acción de la expresión que se está editando. Las flechas rojas indican los cambios que se producen al hacer clic sobre la zona de origen de la flecha.

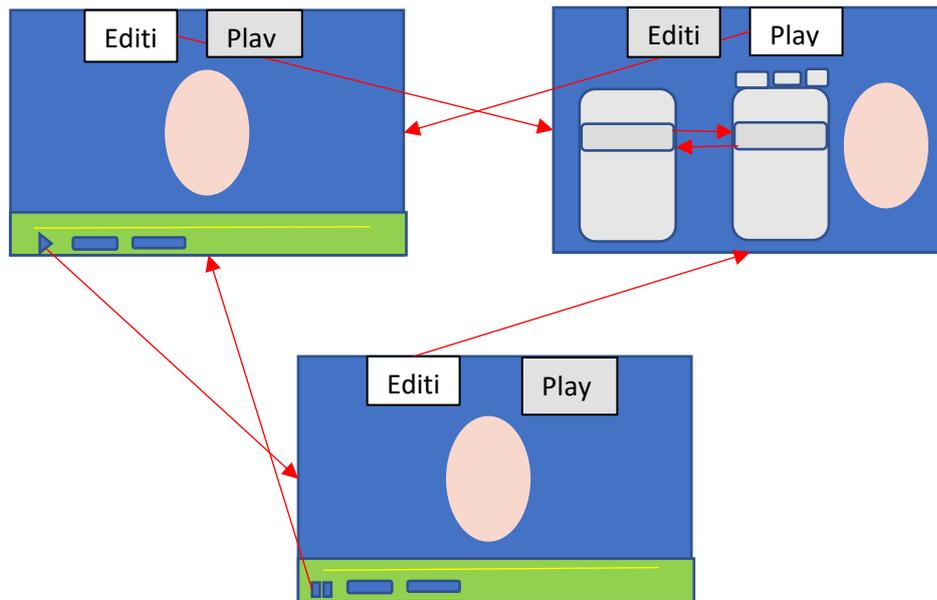


Ilustración 12: Wireframe para la navegación básica de la aplicación entre sus diferentes funcionalidades y modos

4.5. Guías de estilo

Esta fase se basa en tratar de buscar estudios previos realizados en cuanto estilos usados en aplicaciones del mismo campo. Estas investigaciones reciben el nombre de guías de estilo, y son manuales que permiten a los diseñadores de interfaces de usuario realizar interfaces lo óptimas posibles basándose en el usuario. Las guías de estilo también sirven para que las empresas puedan crear su propio sello y aplicarlo en todos sus productos.

Como la aplicación se encuentra dentro del campo de la psicología, se puede englobar dentro del marco de aplicaciones sanitarias, por ello se realiza un pequeño estudio sobre las características que debe tener un programa dentro del ámbito de la medicina para resultar más familiar a los usuarios. Además de esto aplicar estas guías de estilo hace que la aplicación tenga una esencia propia a partir de los estilos visuales utilizados y permita identificar la aplicación gracias a esto.

El primer paso es analizar el público objetivo de la aplicación, previamente se realizó un arquetipo del usuario final en base a estudios realizados, por ello ya se dispone de la información necesaria para identificar los usuarios. Tras esto se investigan en páginas web y documentación guías de estilo que definan terminología, colores y tipografías que se usen en aplicaciones cuyo público objetivo sea similar al del programa y estén dentro del ámbito de la medicina.

El resultado de esta búsqueda son varios documentos y blogs (*What Are the Best UI Designs for Medical Apps?*, 2021) que definen cómo deben ser estas características dentro de la interfaz del programa. A partir de esta información obtenida se debe construir la interfaz gráfica de la aplicación para así lograr un estilo similar a aplicaciones de este campo.



Ilustración 13: Ejemplo de aplicación médica aplicando guía de estilo

En la Ilustración 13 se ven unas pantallas de una aplicación que ha usado colores azules y verdes, los cuales son colores muy predominantes en aplicaciones médicas. Además, hace uso de botones que se recalcan mucho e iconos fácilmente entendibles ya que las personas que pueden hacer uso de la aplicación no necesitan tener conocimientos previos sobre aplicaciones y navegación en ellas. Por último, se ve un diseño muy limpio, sin estar sobrecargado, pero mostrando toda la información necesaria, incluso en caso de ser oportuno se crean listas para que el usuario pueda navegar por toda la información relevante.

5. Prototipado

Se realizan una serie de bocetos y conceptos en diferentes medios que plasman la interfaz de usuario y la manera en la que el usuario se tiene que comunicar con la aplicación. El objetivo principal es probar estos prototipos con personas que den un feedback a la vez de ver como quedan las ideas y soluciones propuestas durante el diseño plasmadas en una interfaz. Es necesario medir la usabilidad de cada prototipo y llegar a un prototipo final que satisfaga las máximas necesidades del usuario posibles. (Baumer et al., 1996)

Para el prototipado se ha seguido un proceso iterativo de tal manera que se llegue a un prototipo óptimo para comenzar a implementar la funcionalidad sobre él. Lo ideal es satisfacer todas las necesidades y requisitos funcionales que se han descrito, siendo el centro de atención el usuario objetivo para tratar de cumplir todas sus necesidades y resolver de la manera óptima los posibles problemas que hayan podido surgir en fases anteriores.

En cuanto a los encuestados hay que tratar de buscar un equilibrio entre el tiempo del que se dispone, y se quiere utilizar, y los resultados que se obtienen de las evaluaciones. Según Nielsen el número ideal de sujetos a los que evaluar por cada prototipo es cinco, y a partir de este número se pueden sacar conclusiones relevantes de cada prototipo, como posibles mejoras y fallos que pueda haber. (Turner et al., 2006)

5.1. Primer prototipo básico

La primera idea es realizar un prototipo básico con una herramienta sencilla que permite hacerse una idea inicial del diseño de la aplicación, usando esta como punto de partida para realizar prototipos más complejos.

Este prototipo cuenta con las funcionalidades básicas de la aplicación, como lo son una lista con todas las unidades de acción disponibles que se hayan implementado, pudiendo el usuario seleccionar entre ellas para añadirlas a la expresión facial que se está editando en ese momento, además de un pequeño panel de control que podría editar ciertos valores del entorno de la expresión, esto no es un requisito funcional, pero está introducido en el prototipo para ver como quedaría y si se entiende el concepto de un pequeño menú no relacionado con las expresiones en cuestión. Además, en el medio se encuentra el rostro, haciendo que el usuario centre la atención en él ya que es el punto más importante de la herramienta, el cómo va a actuar el rostro en función de la entrada que se dé mediante la interfaz el usuario. A mayores se ha colocado una barra bajo el rostro con cinco niveles, haciendo referencia a un slider que permite seleccionar la intensidad con la que el usuario busque que el avatar reproduzca la expresión facial.

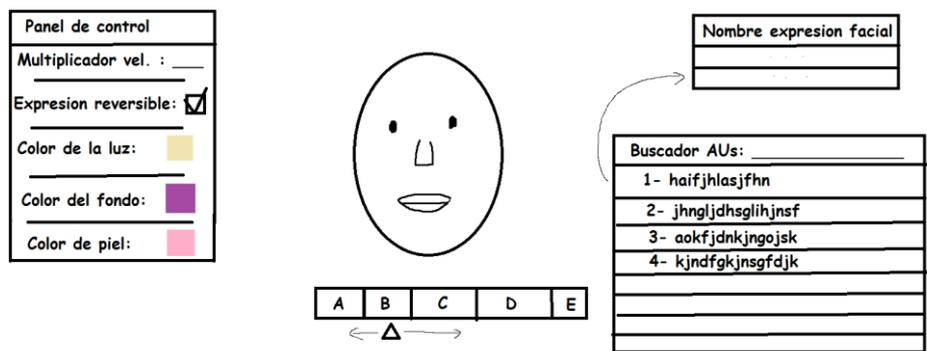


Ilustración 14: Primer prototipo creado en Paint a modo de base de la interfaz de usuario

El prototipo que vemos en la Ilustración 14 y que se ha descrito no dispone de todas las herramientas y funcionalidades que se han descrito en los requisitos y el diseño, ya que es un prototipo muy básico que se usa a modo de base para ir puliendo y añadiendo funcionalidades de la aplicación.

En el prototipo de la Ilustración 14 se puede ver la idea de los paneles, la cual resulta útil de cara a no sobrecargar la pantalla de información y que todo se encuentre organizado en su panel. Esto es necesario debido a la necesidad de mostrar mucha información en pantalla, ya que la herramienta da una funcionalidad compleja de cara al usuario y se debe simplificar lo máximo posible.

A su vez en los paneles se ha introducido un scroll que permite en una parte de la pantalla tener toda la información necesaria referente a la lista de unidades de acción sin necesidad de ocupar todo el tamaño de la lista completa. Esta lista de AUs es de la que el usuario dispone a la hora de editar las expresiones para añadirlas.

Se ve también una flecha indicándonos un scroll de los paneles que contienen las unidades de acción que aparecen en el buscador hacia el panel que contendrá las AUs de la expresión facial en ese momento, esta idea se debe probar en sucesivos prototipos más interactivos y plantearse soluciones como que sea un panel arrastrable por el usuario; o el doble clic sobre la AU que se desea añadir a la expresión facial para ver qué propuesta consideran los usuarios como más intuitiva y familiar.

El prototipo muestra la idea básica de la aplicación, se basa en no sobrecargar al usuario mostrando toda la información en pantalla, sino intentar minimizar esta sin quitar ninguna función necesaria de la aplicación. A la vez tratar de que sea intuitiva, buscando que el usuario gaste el menor tiempo posible en usar el programa. Para ello a continuación se hará un prototipo en papel el cual permitirá hacer las primeras pruebas con usuario que dará una idea aproximada de cuáles son los aspectos que se deben mejorar y cuales es conveniente mantener para satisfacer a los usuarios.

5.2. Prototipo en papel

En esta fase haciendo uso del prototipo anterior se construye uno sobre papel para hacer pruebas con usuarios que aporten información. Este prototipo es útil para resolver dudas de cómo es óptimo de cara al usuario implementar las ideas propuestas en el diseño.

En el prototipo a papel se busca conocer si al usuario le resulta cómodo la interacción con la información mediante paneles mostrados en pantalla con listas dinámicas que se mueven de la lista de unidades de acción totales a la lista de AUs que componen la expresión, y viceversa. Además, también se evalúa la forma que tiene el usuario de interactuar con cada ranura de las unidades de acción teniendo como opciones que lo pueda arrastrar de un panel a otro o que se muevan de un panel a otro haciendo doble clic en cada unidad de acción. Ahora que ya están introducidos más elementos funcionales de los que se habló en los requisitos como una caja de reproducción que permita al usuario controlar la reproducción.

Se han obtenido conclusiones sobre las pruebas realizadas con usuarios de sus posibles dudas sobre la implementación. Lo primero sobre los paneles es. Además, el clic sobre las unidades de acción para añadir y quitar de las expresiones faciales. A su vez se llega a la conclusión de implementar dos modos diferentes para diferenciar la funcionalidad, uno de estos es el modo reproducción donde se ven las herramientas necesarias para reproducir las animaciones de la expresión facial, por otro lado, el modo edición con los paneles necesarios para editar las expresiones faciales.

5.3. Prototipo en Unity

Por último, como Unity es una herramienta tan versátil nos aporta una serie de herramientas ya creadas, como botones, paneles, cajas de texto y demás elementos básicos que permiten prototipar de una manera muy cómoda y sencilla, e incluso utilizar lo creado para luego implementar la funcionalidad en el prototipo y cambiarlo visualmente para seguir las guías de estilo nombradas anteriormente.

En el prototipo se han implementado los paneles referentes a la expresión facial que se está editando y las unidades de acción que contiene esta; y a la lista total de unidades de acción. Además, se han implementado dos modos de cara a que el usuario sepa lo que está haciendo en cada momento si reproducir una expresión ya creada o crear y editar una. Aparte de estas dos funcionalidades esta implementado un menú de la aplicación que permite cerrar el programa, guardar una expresión, abrir una ya existente o crear una nueva.

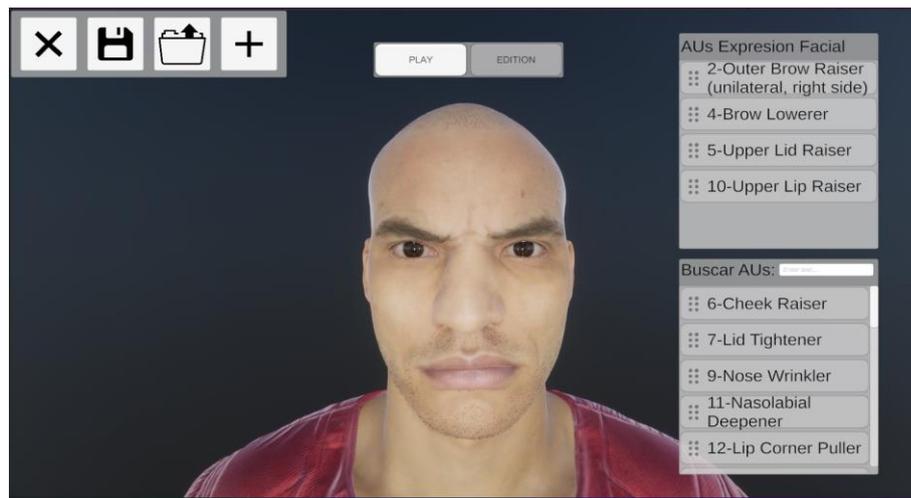


Ilustración 15: Primer prototipo de modo edición

En la Ilustración 16 vemos todo lo dicho anteriormente, además del avatar elegido previamente, y se puede apreciar la forma en la que van a verse los paneles dentro de la aplicación y los botones sobre el avatar. Aún es necesario implementar el modo reproducción en el que haya una barra de herramientas con lo referente a la reproducción de las animaciones que formen parte de la expresión facial.

Sobre este prototipo las respuestas por parte de los usuarios usados para evaluar las heurísticas son positivas, implementar dos modos ha sido óptimo, para que el usuario rápidamente identifique si está solo viendo la expresión, o editando. A su vez un panel para el menú es muy útil para que el usuario diferencie la funcionalidad esencial de crear o reproducir las expresiones de las funcionalidades extras que aporta el menú. Pese a esto el menú está mal distribuido, ya que mezcla funcionalidades referentes a la aplicación, como es cerrarla, y la funcionalidad de las expresiones faciales, como es guardarla o crear nuevas, por ello en posteriores diseños de la interfaz se redistribuyen los botones en zonas más adecuadas en la interfaz.

6. Implementación

Ahora se comienza el desarrollo de la aplicación haciendo uso de Unity como herramienta principal y el material nombrado previamente. El primer paso es parametrizar las unidades de acción en Unity usando el avatar elegido. Una vez estén las AUs elegidas

implementadas, deberemos de crear una interfaz que permita al usuario elegir cualquiera de ellas.

Simplemente es importante conocer algunos conceptos importantes dentro del motor como lo es la escena, que es el lugar donde van colocados todos los modelos 3D, funcionalidad y demás componentes que se quieran cargar sin necesidad de refrescar nada en la aplicación. Además de escena es importante conocer el concepto de GameObject que es cualquier objeto que se encuentre ubicado en una escena de Unity el cual puede tener componentes, estos componentes puede ser funcionalidad, mallas 3D, reproductores de audio y animaciones, etc. Por último, un concepto importante son los Assets, que son aquellos materiales que podemos usar en nuestras escenas, interfaces o componentes directamente.

6.1. Implementación de unidades de acción en Unity

Unity tiene una herramienta de crear animaciones a partir de los Riggs de un modelo 3D. Estas animaciones se pueden manejar desde el código. La idea es que cada una de las unidades de acción se implementen en un Animation Clip.

En Unity las animaciones están compuestas por Animations Clips, que son la animación en sí, que permite que se graben desde el motor cuando editamos ciertas propiedades de los objetos de una escena, pudiendo luego reproducir estas transformaciones. Por otro lado, está el Animator Controller que es el cerebro de las animaciones, es la parte encargada de crear una máquina de estados que haga cambiar de unos Animations Clips a otros usando variables y condiciones sobre estas, las variables se pueden editar mediante código.

Una vez explicados estos dos conceptos, lo primero que se debe hacer es implementar las unidades de acción en Animations Clips, para después diseñar la lógica que transicione entre un Animation Clip del rostro relajado a uno de una unidad de acción.

Para grabar el Clip Unity proporciona una herramienta muy útil, mediante la cual seleccionaremos un GameObject de la escena de Unity y crearemos un nuevo archivo en el cual podremos guardar las transformaciones que se realicen durante un tiempo determinado en el GameObject, permitiendo reproducirlo como si fuera un clip de video en el que se ve una transición que pasa realiza los pasos intermedios.

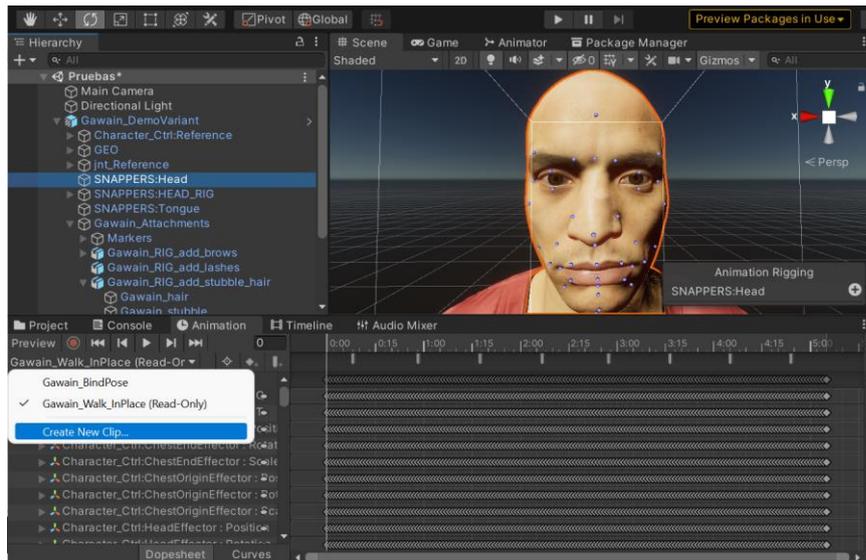


Ilustración 16: Creación de nuevo Animation Clip para el GameObject que contiene los Riggs del rostro del avatar.

En la Ilustración 17 se ve cómo está seleccionado en la jerarquía de GameObjects de Unity el SNAPPERS:Head que es aquel que contiene los Riggs de la musculatura de la cara, como se puede apreciar en la escena le salen puntos azules en la cara que son aquellos que se pueden mover. Una vez hecho esto en la herramienta de crear Animations Clips y en el desplegable se selecciona la opción “Create New Clip...” para comenzar a crear las unidades de acción.

Una vez creado el clip correspondiente a una AU se comienza a grabar esta sobre el rostro del avatar, el movimiento de los músculos correspondientes, para esto deberemos de pulsar el botón de “Record”, representado con un círculo rojo.



Ilustración 17: Grabación de un Animation Clip haciendo uso de las transiciones generada automáticamente

En la Ilustración 18 se puede ver como se graban los key frame, que son los rombos en grises bajo la barra de tiempos. Estos Key Frames es un fotograma que se toma como referencia y a través del Unity calcula las transiciones intermedias entre cada una de las propiedades. En el ejemplo de la Ilustración 18 se ve que hay 4 propiedades de los que se han guardado Key Frames, que son aquellas que se van a alterar a lo largo de esta animación, en este caso es la posición de los GameObject referidos a las cejas izquierda y derecha (Brow_L y Brow_R), además se aprecia que al final del Animation Clip, el rostro queda con las cejas levantadas, mientras que antes de nada como se ve en la Ilustración 17 la cara se encontraba relajada sin contraer ningún músculo.

La idea es que aquellas AUs que se deseen implementar crear su respectivo Animation Clip y ponerle una duración estándar de 1 segundo, más tarde se puede editar sin problema, y grabar mediante keyframes la transformación de las posiciones de los músculos que están involucrados en las unidades de acción. De esta forma no se afecta que se aumente la duración de los Animations Clips.

Ahora se va una por una con todas las AUs a implementar hasta tenerlas todas en forma de Animations Clips para poder controlarlas desde nuestro Animator Controller y por ende desde nuestro código.

6.2. Tratamiento de las unidades de acción en el código

Una vez conseguido parametrizar las unidades de acción dentro de Unity mediante Animations Clips, hay que diseñar el Animator Controller de tal forma que las transiciones entre estado relajado y alguna de las AUs se produzcan mediante condiciones de variables

La complejidad en este punto es el hecho de que el Animator Controller tiene que ser capaz de reproducir simultáneamente distintos Animations Clips, cosa que creando una única máquina de estados sería imposible, ya que las transiciones entre los Animations Clips imposibilitan que este en varios estados a la vez.

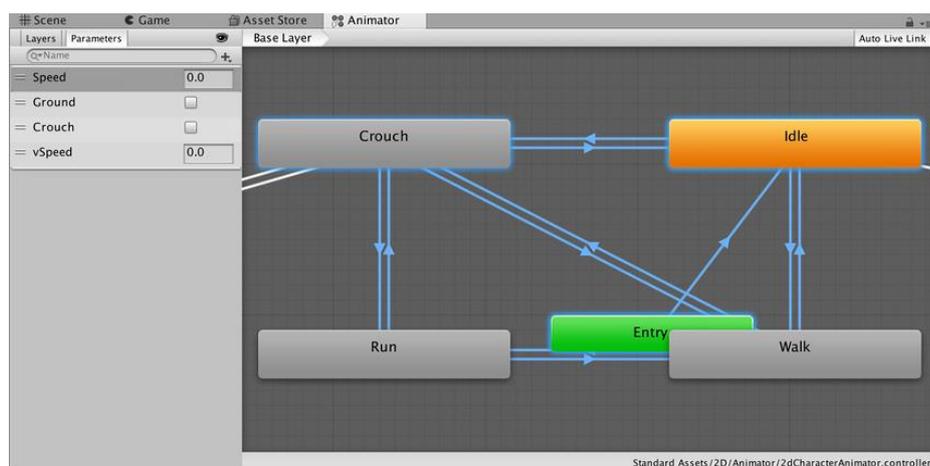


Ilustración 18: Ejemplo de máquina de estados de un Animator Controller en Unity para las animaciones.

Como se ve en la Ilustración 19 la máquina de estados va pasando de estado a estado mediante transiciones que se realizan si se cumple una cierta condición. Nosotros

debemos hacer que varios estados, o Animations Clips, se solapen en simultáneo si se cumplen las condiciones correspondientes, para ello podemos crear varias máquinas de estado diferentes, las cuales tengan dos estados, uno relajado y otro una unidad de acción, para que desde el código se pueda solapar todas las máquinas de estado reproduciendo de esta forma varias animaciones a la vez.

El Animator Controller permite crear varias capas, que son niveles de animación, y a cada capa asignarle un peso que va de 0 a 1, este peso es la importancia que tienen unas capas sobre otras. En el caso de las unidades de acción todas tienen la misma importancia y por ello debemos de asignar a todas las capas el mismo peso, o sea el máximo.

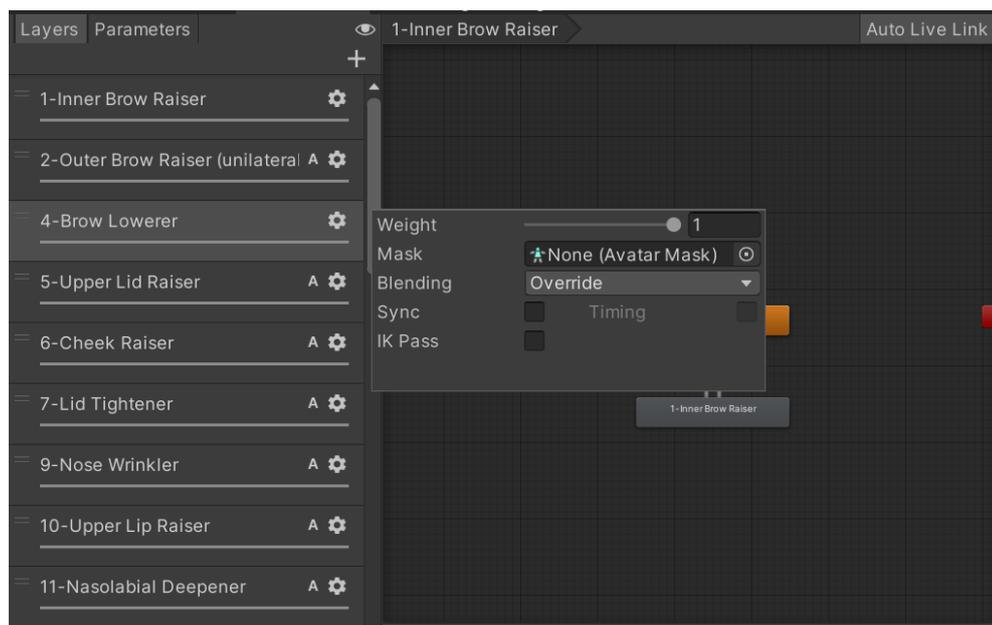


Ilustración 19: Lista de capas del controlador de animaciones de Unity

Como se ve en la Ilustración 20 lo primero es crear capas para cada AU que queramos implementar y asignarle a todas un peso de uno. Una vez hecho esto en cada uno de ellos se debe crear una máquina de estados compuesta por el estado parado, Bind y el Animation Clip correspondiente a la unidad de acción.

Luego en cada capa añadiremos los dos estados, el parado y la unidad de acción correspondiente a cada capa, y se ponen dos condiciones, de ida y vuelta al estado de parado. Esta condición la controlaremos con una variable booleana que accionamos desde código, la cual se denomina como condición de activación. Además de la variable booleana asignada a las transiciones de ida y vuelta, se declara una variable float cuyo valor puede estar entre 0 y 1, la variable está enlazada al tiempo de reproducción del Animation Clip, esta variable se denomina tiempo de movimiento.

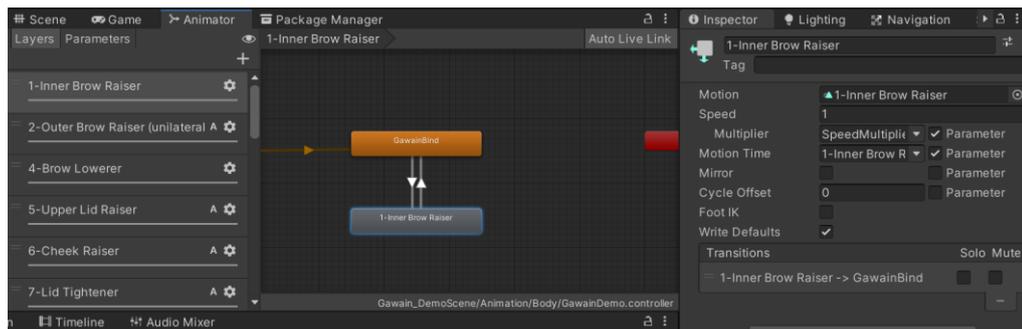


Ilustración 20: Máquina de estado referente a la AU 1-InnerBrow Raiser que tiene 2 estados parado y el Animation Clip de la AU.

Como vemos en la Ilustración 21 hay un parámetro habilitado llamado Motion Time, esta variable se enlaza a una variable del Animator Controller, la que se ha denominado tiempo de movimiento. A partir de las dos variables nombradas se permite que cuando el usuario hace clic en una nueva AU que quiere añadir a la expresión facial actual se active la condición de activación, y un hilo paralelo se encargará de ir incrementando el valor del tiempo de movimiento para que así esta se reproduzca a la velocidad deseada y hasta la intensidad deseada.

Aparte de todo esto las AUs deben de estar identificadas de alguna forma en el código para saber cuándo se selecciona una de cuál se trata. Para ello tendremos una lista de cadenas dinámica que contendrá inicialmente el nombre de todas las AUs implementadas. Cada vez que usemos las AUs su tratamiento será mediante la cadena que las identifica, cuando lo añadamos a una expresión simplemente añadiremos la cadena a la lista de AUs de la expresión, y mediante el nombre se puede activar la condición de activación y modificar el tiempo de movimiento, ya que ambas variables están denominadas por el nombre la AU y el nombre de la AU + “<Time>” respectivamente, para que vía código solo sabiendo el nombre se puedan modificar fácilmente.

6.3. Reproducción de la suma de unidades de acción implementadas: expresión facial

Una vez parametrizadas las variables en el Animator Controller es necesario reproducir las capas que se deseen de este, en función de que unidades de acción quiera el usuario que se activen.

El primer paso es implementar una interfaz que permita al usuario haciendo clic a una lista de AUs añadirla a la expresión actual y que esta se reproduzca. Por ello se debe implementar que el clic en cada AU active su respectiva condición de activación en la capa correspondiente a la unidad de acción, y un clic una vez ya está la AU añadida a la expresión facial desactive esta condición de activación para que esa máquina de estado cambie del Animation Clip de la AU al de parado.

Una vez hecho esto se debe implementar un hilo en paralelo encargado de ir incrementando el tiempo de movimiento del Animation Clip de cada AU que esté activa en la expresión actual, para que de esta forma las AUs estén coordinadas y se tenga un

temporizador elegido para darle una duración máxima al conjunto de animaciones, unidades de acción, habilitadas.

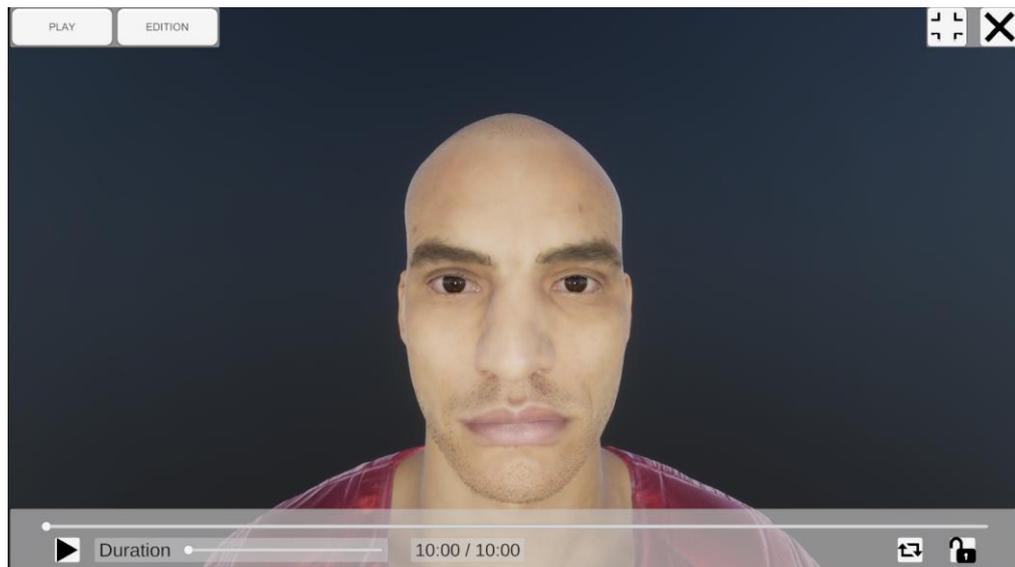


Ilustración 21: Interfaz gráfica del modo reproducción, en el que se ve los elementos como lo son la barra de reproducción y botones para cambiar de modo.

6.4. Parámetros de reproducción

Se implementan varios parámetros de reproducción que permiten al usuario elegir la duración e intensidad máximas de la expresión facial. Además de estos dos hay un botón de bucle que hará que la expresión facial se reproduzca continuamente y no pare nunca.

6.4.1. Duración

La duración se implementa mediante un slider para que el usuario pueda moverlo del mínimo al máximo. Esto provocará que en caso de estar en el punto mínimo la suma de las animaciones de las unidades faciales, o sea la expresión facial dure el menor tiempo posible, asignado a 1 segundo, mientras que si el usuario aumenta la duración al máximo la expresión facial durará el mayor tiempo posible, siendo 5 segundos.

Esta herramienta es útil para que el usuario tenga más tiempo para ver los cambios que sufre el rostro a lo largo de las animaciones, o para asignar más un valor realista y más breve.

6.4.2. Intensidad

La intensidad se implementa, al igual que la duración, con un slider. En este caso el slider permite al usuario asignar el máximo de contracción de cada una de las unidades de acción que conforman la expresión facial que se está reproduciendo. El funcionamiento del slider es que en el punto más alto las animaciones de las AUs lleguen hasta su máximo, contrayendo así todo lo posible los músculos del avatar, mientras que en el valor mínimo

del slider no se reproduzca nada ya que la intensidad sería 0, por ende, la contracción de los músculos involucrados en la expresión facial es inexistente.

Esta herramienta es útil para el terapeuta ya que como ya se habló, las expresiones faciales en el mundo real nunca llevan los músculos a su máxima contracción, sino que son movimientos algo más naturales y no tan forzados, exceptuando expresiones faciales en concreto, como puede ser la de sorpresa que funciona a modo de acto reflejo.

6.5. Modo de reproducción y edición

La aplicación como se ha dicho anteriormente requiere de dos modos para el usuario, el modo reproducción, a través del cual podrá reproducir expresiones faciales ya creadas; por otro lado, el modo edición, en el cual se pueden crear nuevas expresiones, editar las ya existentes y guardar la que se está editando actualmente.

Para esto la idea es implementar dos botones que funcionan a modo de pestaña los cuales permiten al pulsar alterar la posición del avatar, además de sus parámetros de reproducción, y los paneles que se muestran de funcionalidades.

6.5.1. Modo Reproducción

Implementa la funcionalidad descrita anteriormente de Reproducción de la suma de AUs implementadas, así como los parámetros de reproducción. El usuario puede reproducir las expresiones faciales guardadas desde este modo, así como editar sus parámetros de reproducción para que se vea de una manera diferente.

Se muestra una barra de reproducción en la que hay un botón de “Play/Pause” para reproducir o pausar la animación en sí como si fuese un video. A su vez hay un slider el cual avanza a medida que la animación se reproduce, como si fuera un indicador de un reproductor de video de porque zona del video vamos pues igual, este slider está enlazado a la variable nombrada anteriormente de tiempo de movimiento.

Además de estos dos elementos, se han implementado sliders que controlan la duración e intensidad de la aplicación, las cuales hacen referencia a el tiempo máximo de duración para una intensidad dada y la intensidad máxima de contracción de los músculos que interaccionan en la expresión facial, respectivamente. Por último, se añade un botón que sirve para reproducir la expresión en bucle, o sea que cuando el slider de la reproducción llegue al final volverá rápidamente al comienzo y se reproducirá la expresión de nuevo.

6.5.2. Modo Edición

El modo edición permite al usuario crear nuevas AUs, guardarlas y cargar otras almacenadas en disco, así como borrar las guardadas y cambiar el nombre a la actual.

La idea principal es implementar dos paneles, uno que contenga la lista de AUs totales, y un buscador para que el usuario pueda buscar directamente una unidad de acción en concreto. El otro panel contiene una lista de las AUs que forman la expresión facial que

se está editando en ese momento, pudiendo mediante el doble clic llevar las AUs de un panel al otro, en función de las que quiera añadir o quitar el usuario a la expresión facial.

6.6. Almacenamiento de expresiones faciales

Para el almacenamiento de expresiones faciales, es necesario crear un tipo de dato que almacene la lista de unidades de acción que contiene la expresión y una cadena que sirva como nombre identificador de cada expresión para cuando el usuario cargue las expresiones del disco sepa cual está eligiendo.

6.6.1. Crear nueva expresión facial

Se instancia un nuevo objeto de la clase que almacena la expresión facial, para ello pide al usuario un nombre mediante un pop-up y así ya se tiene una nueva expresión, ahora el usuario debe seleccionar de la lista de AUs cual quiere añadir y se guardan en la nueva instancia de manera temporal hasta que el usuario decida guardarlo en el disco.

6.6.2. Guardar expresión facial

La clase que almacena la información de la expresión es serializable, de esta forma se puede almacenar en un fichero binario en el disco. Cada vez que el usuario guarde una nueva expresión se carga el fichero con todas las expresiones almacenadas en disco y se comprueba si la que estamos guardando es nueva o comparte nombre con una existente, para de esta forma saber si hay que añadir una nueva expresión o sobrescribir una ya existente.

Una vez alterada la lista que se ha cargado de instancias de expresiones se vuelve a serializar para guardarla en disco y que quede de esta forma el fichero binario en disco actualizado con los cambios guardados.

6.6.3. Cargar expresión facial

Hay una ruta definida en disco en la cual se encuentra el fichero con la lista de expresiones que se han guardado, la idea es cargar todas y quedarnos con el nombre de cada expresión para mostrarle una lista en la interfaz al usuario para que él pueda elegir por el nombre que expresión es la que desea cargar.

Una vez el usuario elige la expresión según el nombre, se cargará toda la información de la clase, tanto el nombre como la lista de AUs que la componen, y la asignará a la instancia que se está editando y reproduciendo en ese momento.

6.7. Unidades de acción antagónicas

Cuando hablamos de expresiones faciales y de suma de unidades de acción hay que ser conscientes que hay ciertos movimientos musculares que son contrarios a otros, aquellos que mueven músculos de la misma zona, pero la contracción tenga sentidos opuestos. Por

ejemplo, a la hora de sonreír según el FACS se activa la AU 12 - Lip Corner Puller que eleva las comisuras de los labios hacia arriba, mientras que hay otra unidad de acción que mueve los extremos de los labios hacia abajo indicando tristeza, es la AU 15 - Lip Corner Depressor.

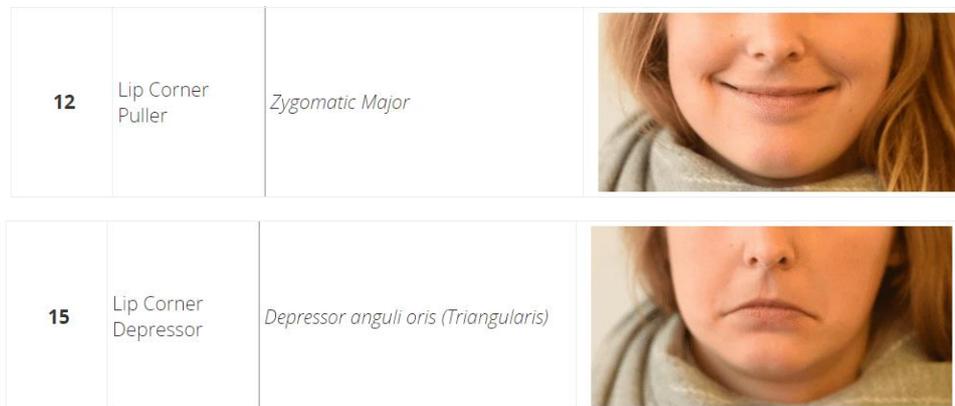


Ilustración 22: Unidades de acción antagónicas, que no se podrían realizar simultáneamente.

Como se ve en la Ilustración hay AUs que no se pueden reproducir simultáneamente ya que el resultado sería antinatural. La solución a esto es implementar un mapa que indique por cada una de las unidades mínimas de acción implementadas, una lista de cuales son incompatibles con ellas. En el código se implementa con un mapa siendo la clave la unidad de acción en cuestión y el valor una lista de unidades de acción incompatibles con la clave.

```

1   Inner Brow Raiser => 4
4   Brow Lowerer => 1
5   Upper Lid Raiser => 7
7   Lid Tightener => 5
10  Upper Lip Raiser => 11 13
11  Nasolabial Deepener => 10 13
12  Lip Corner Puller => 13 14 15 18
13  Cheek Puffer => 10 11 12 15 18
14  Dimpler => 12
15  Lip Corner Depressor => 12 13 18
16  Lower Lip Depressor => 17
17  Chin Raiser => 16
18  Lip Puckerer => 12 13 15 20
20  Lip stretcher => 18 22 23 24
22  Lip Funneler => 20 23 24
23  Lip Tightener => 20 22 24
24  Lip Pressor => 20 22 23

```

Ilustración 23: Mapa de unidades de acción incompatibles

En la Ilustración 24 se puede observar el mapa creado de AUs incompatibles entre ellas, esta diseñado a partir de que dos unidades de acción utilicen un musculo, pero en direcciones contrarias provocando eso una incompatibilidad entre ellas

7. Mejoras implementadas

Como se ha explicado anteriormente, el desarrollo de la aplicación ha seguido un proceso iterativo sobre la secuencia de pasos que se han seguido. Por ello, aunque no se plasme en el orden correcto a lo largo del prototipado e implementación se han hecho unas evaluaciones heurísticas sobre la usabilidad que nos permite a la vez que se han ido implementado los requisitos de la aplicación realizar mejoras en necesidades que ya se habían implementado. Además de realizarse mejoras sobre pasos previos, también se han incluido nuevas funcionalidades que se ha considerado que complementan a otras o ayudan a cumplir las heurísticas de las evaluaciones.

7.1. Mejoras

Las mejoras implementadas durante la implementación se han hecho acorde a las evaluaciones heurísticas basándose en las diez normas de la usabilidad de Nielsen para crear interfaces gráficas. Para hablar de ellas se dirá cuál es la mejora y que heurística favorece de cara a mejorar la experiencia del usuario.

Se han añadido un asterisco en el botón de guardar para representar cuando la aplicación se encuentra en un estado diferente al que hay guardado en disco, para que de esta manera el usuario sepa que está en un estado inseguro, este asterisco desaparece cuando el usuario guarda en disco la información volátil. Además, los botones de los modos edición y reproducción se quedan clicados en función del modo en el que se encuentre el usuario para que en cada momento solo pueda ir al otro modo y sepa en cual esta. En el modo reproducción los botones de la barra inferior cambian de iconos en función del estado en el que se encuentre, si esta pausada o no, si están bloqueados los paneles o en modo bucle. Todo esto favorece al usuario a tener visibilidad del estado del sistema siendo capaz de ser consciente de que hace el sistema en cada momento y que puede hacer el a partir de ese estado.

En cuanto al guardado, en caso de que en el botón se encuentre el asterisco, indicando que hay algún cambio no guardado si el usuario intenta abrir una nueva expresión o cerrar la aplicación provocando ambas acciones la pérdida de la información sin guardar le mostrara un mensaje indicando que no ha guardado y si lo desea hacer en ese momento antes de que sea irreversible. Esto favorece a la prevención de errores por parte del usuario para que si se equivoca al realizar una acción que no deseaba o se le olvidaba un paso previo la aplicación se lo pueda recordar.

Los cuadros de advertencia de no guardado permiten marcar para que no se vuelvan a mostrar, por si el usuario no desea que le pida confirmación siempre, además que desde las opciones de la aplicación se puede elegir sobre todos los cuadros de advertencia. En las opciones también permite al usuario elegir que atajos de teclado quiere que tenga cada acción para que pueda cambiarlos al gusto. Todo ello da al usuario el control y la libertad que desee en la aplicación.

Al aplicar las guías de estilo de aplicaciones médicas como ya se explicó se cumple una de las heurísticas que requiere que el usuario reconozca aspectos de la aplicación en lugar de tener que recordarlos. La idea de usar como referencia interfaces de usuario ya creadas o guías de estilo es para favorecer la familiarización por parte del usuario con la aplicación y no obligarle a recordar el funcionamiento de la aplicación, sino que pueda usar conocimientos previos que tenga.

Además, los atajos de teclado favorecen a la flexibilidad y eficiencia de uso del software. Gracias a ellos permitimos a usuarios menos experimentados que puedan usar la interfaz gráfica normalmente mediante los botones y el uso del ratón, pero si hay usuarios que quieren ahorrarse tiempo y navegar más rápido por la aplicación pueden hacer uso de los atajos de teclado.

Estos son algunas de las heurísticas en las que se ha hecho hincapié por considerarse de mayor importancia en una aplicación dentro del ámbito médico. Además de esto a continuación se explican nuevas funcionalidades implementadas para ampliar el uso de la herramienta y que sea más completa, así como para favorecer las heurísticas que se han aplicado para evaluar la interfaz de la aplicación.

7.2. Nuevas funcionalidades

Ahora se habla de las nuevas funcionalidades que han surgido a lo largo de la implementación y las evaluaciones de esta para favorecer la usabilidad de la aplicación e implementar nuevas funciones que pueden resultar muy útiles para los terapeutas y el propósito de la herramienta.

7.2.1. Intensidad de unidad de acción

Las expresiones faciales creadas por los usuarios pueden ajustar su intensidad total en el reproductor por si el terapeuta desea mostrar expresiones que contraigan los músculos involucrados al máximo o a un menor porcentaje. Esta nueva funcionalidad permite al usuario dentro del modo editar a cada una de las unidades de acción que forman las expresiones faciales, para de esta forma poder crear expresiones totalmente modificables y hacerlas lo más cercanas a las que puede mostrar un rostro humano real. Esta funcionalidad además será útil para que el terapeuta pueda mostrar expresiones más realistas o más exageradas que pueden significar sarcasmo, por ejemplo.

Para ello se ha implementado una lista de números decimales que van del 0 al 1 siendo 0 una intensidad nula y 1 la máxima. Esta lista será de un tamaño dinámico en función de las unidades de acción que formen la expresión que el usuario está editando. Cuando el usuario haga clic sobre cada slot que contiene la unidad de acción de la expresión facial saldrá un cuadro que le permitirá mediante una slider ajustar la intensidad de la AU concreta. Desde el código cada posición de la lista dinámicas que contiene las AUs tendrá una intensidad correspondiente almacenada en la misma posición de la lista de intensidades respectivamente.

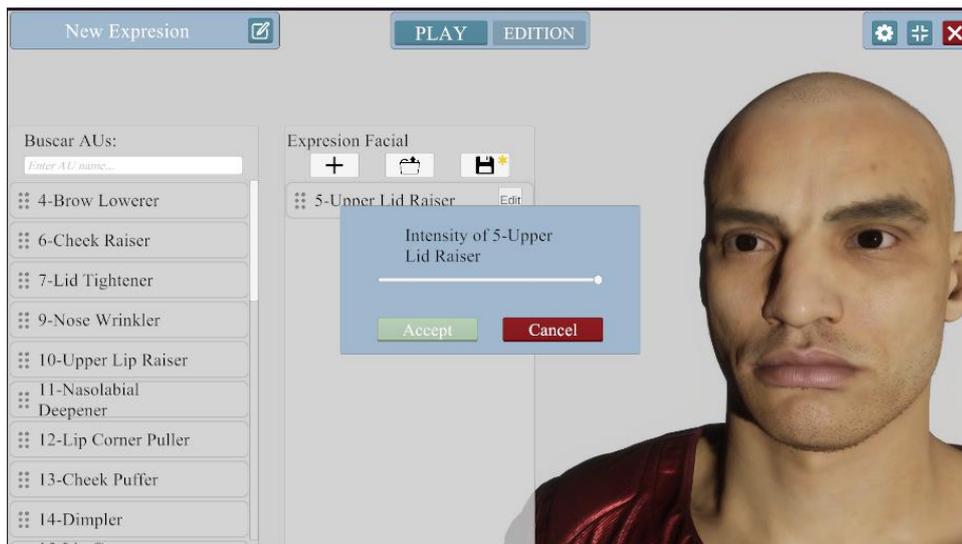


Ilustración 24: Cuadro para ajustar la intensidad de cada AU perteneciente a una expresión facial

A la hora de guardar las expresiones faciales en disco a la clase creada para ese propósito se le añade una lista dinámica que hará referencia a la lista que contiene cada expresión para indicar la intensidad de cada una de sus AUs, para que cuando el usuario la recupera del disco mediante la función de cargar, también se muestren las intensidades de cada unidad de acción guardada.

7.2.2. Opciones de la aplicación

Las opciones de la aplicación no implican una nueva funcionalidad para el terapeuta de cara a la herramienta, pero permitirá al usuario tener un mayor control sobre las funcionalidades que ya están implementadas además de editar atajos y preferencias por defecto.

Para ello se crea una clase que almacene de manera estática en ella todas las opciones elegidas por el usuario en tiempo de ejecución de la aplicación, y cada vez que edite una opción se cambiara el valor almacenado de la opción en la clase estática. Además de esto se guardará en disco para que al arrancar la aplicación se carguen todas las opciones de disco a como las había dejado el usuario la última vez que la uso.

Para guardar estas opciones se usa la clase de Unity PlayerPrefs, que permite guardar cadenas, enteros y numero decimales en disco y recuperarlos de una manera muy sencilla, como si se tratara de una variable que se recupera y en caso de no haber sido guardad nunca se le asigna un valor por defecto. Estos valores por defecto serán los que el desarrollador ha asignado por defecto a la aplicación, como lo es mostrar todos los cuadros de confirmación antes de perder cambios no guardados o los atajos de teclado asignados.

8. Conclusión y líneas futuras

Por último, es necesario hacer una breve conclusión del resultado obtenido y líneas futuras de la aplicación para continuar mejorando y solucionando los errores que pueda tener. El principal objetivo para esta primera iteración del desarrollo de la aplicación era diseñar e implementar una herramienta capaz de ser usada por terapeutas para crear expresiones faciales totalmente reproducibles a partir de las unidades de acción mínimas, y a su vez reproducir estas expresiones creadas y las universales con las que la aplicación cuenta disponibles de base. Además del objetivo principal se han diseñado e implementado ideas para favorecer la usabilidad de los requisitos de la aplicación y nuevos objetivos que favorecen la tarea para la que se ha desarrollado la herramienta.

8.1. Líneas futuras

La aplicación desarrollada puede ser mantenida y actualizada a la vez que se implementen nuevas funcionalidades que resulten útiles para el propósito de los terapeutas. A partir de la información recibida por los usuarios, una vez se lance la aplicación, comenzaría el proceso de solucionar todos los posibles fallos que hasta ese momento han permanecido ocultos a los ojos del desarrollador. La idea es usar el feedback que aporten los usuarios finales para solucionar errores y mejorar aspectos de la aplicación.

La primera mejora debería ser implementar de una manera más precisa y realista las unidades de acción sobre el avatar, ya que, aunque ya hay bastantes implementadas y se entiende los músculos que mueven cada una, es conveniente que sean lo más detalladas posibles y que un experto en animaciones se encargue de ello. Gracias a Unity este paso solo requeriría de crear las animaciones de las unidades de acción e intercambiarlas por las que ya hay, o añadirlas si fueran nuevas, ya que la funcionalidad del programa es igual y adaptable a cambios de los Animations Clips de las AUs.

Una funcionalidad a tener en cuenta en un futuro próximo es permitir al terapeuta estar en el modo editar en su pantalla del ordenador, mientras en otra pantalla el paciente ve el modo reproducción, de esta forma el terapeuta podría ajustar todos los parámetros en tiempo real mientras se los muestra a su paciente. Como la aplicación busca ser lo más inmersiva posible en el modo reproducción, para que el paciente le resulte más realista sería de gran ayuda implementar este modo en realidad virtual o aumentada haciendo uso de unas gafas dedicadas a tal propósito (Estrella, 2011) como lo pueden ser las Oculus Quest 2 o las HTC Vive. (Borges et al., 2018)

En cuanto a las mejoras o más funcionalidades que se puedan implementar es necesario ahora que la versión inicial ya se ha terminado de desarrollar es momento de que lo prueben usuarios finales de la aplicación para usar su información junto con la versión inicial que ya hay desarrollada de la aplicación para continuar mejorándola a lo largo de iteraciones que se realizaran.

En definitiva, la aplicación permite continuar su desarrollo a lo largo del tiempo, ya que es una herramienta muy útil que puede ser usada de una manera muy eficaz por parte de psicólogos y terapeutas con sus pacientes con problemas de cognición social.

9. Bibliografía

- ¿ *Qué es Rigging* ? (2011, abril 13). <http://www.artzuza.com/2011/04/character-animation-technical-director.html>
- Abras, C., Maloney-Krichmar, D., & Preece, J. (2004). *User-centered design*. 15.
- Adolphs, R. (1999). Social cognition and the human brain. *Trends in Cognitive Sciences*, 3(12), 11.
- Agarwala, A., Hertzmann, A., Salesin, D. H., & Seitz, S. M. (2004). Keyframe-based tracking for rotoscoping and animation. *ACM Transactions on Graphics*, 23(3), 584-591. <https://doi.org/10.1145/1015706.1015764>
- Baumer, D., Bischofberger, W., Lichter, H., & Zullighoven, H. (1996). User interface prototyping-concepts, tools, and experience. *Proceedings of IEEE 18th International Conference on Software Engineering*, 532-541. <https://doi.org/10.1109/ICSE.1996.493447>
- Bevana, N., Kirakowskib, J., & Maissela, J. (1991). What is Usability. *In Proceedings of the 4th International Conference on HCI*.
- Borges, M., Symington, A., Coltin, B., Smith, T., & Ventura, R. (2018). HTC Vive: Analysis and Accuracy Improvement. *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2610-2615. <https://doi.org/10.1109/IROS.2018.8593707>
- Coan, A. P. D. of P. J. A., Coan, J. A., Allen, J. J. B., Ph.D, J. J. B. A., & Allen, P. D. of P. J. J. B. (2007). *Handbook of Emotion Elicitation and Assessment*. Oxford University Press, USA.
- Estrella, A. D. (2011). *Inmersión mental y realidad virtual*. 4.
- Experience, W. L. in R.-B. U., & Jakob Nielsen. (1994, abril 24). *10 Usability Heuristics for User Interface Design*. Nielsen Norman Group. <https://www.nngroup.com/articles/ten-usability-heuristics/>
- Expresiones faciales universales propuestas por Ekman (1972)*. | [Download Scientific Diagram](#). (2017, marzo). https://www.researchgate.net/figure/Figura-2-Expresiones-faciales-universales-propuestas-por-Ekman-1972_fig1_321901828

Herramientas para la rehabilitación de la cognición social en daño cerebral. (2020, septiembre 8). *Centros de Daño Cerebral de Hospitales vithas*. <https://neurorhb.com/blog-dano-cerebral/herramientas-para-la-rehabilitacion-de-la-cognicion-social-en-dano-cerebral/>

hickeys. (2022, mayo 11). *Accesibilidad (conceptos básicos de diseño)—Win32 apps*. <https://docs.microsoft.com/es-es/windows/win32/uxguide/inter-accessibility>

IWARSSON, S., & STÅHL, A. (2003). Accessibility, usability and universal design—Positioning and definition of concepts describing person-environment relationships. *Disability and Rehabilitation*, 25(2), 57-66. <https://doi.org/10.1080/dre.25.2.57.66>

La cognición social en niños y cómo la trabajamos con NeuronUP. (s. f.). Recuperado 4 de julio de 2022, de <https://www.neuronup.com/actividades-de-neurorrehabilitacion/actividades-para-funciones-cognitivas/actividades-de-cognicion-social/la-cognicion-social-en-ninos-y-como-la-trabajamos-con-neuronup/>

Mast, M. S. (2007). On the importance of nonverbal communication in the physician–patient interaction. *Patient Education and Counseling*, 67(3), 315-318. <https://doi.org/10.1016/j.pec.2007.03.005>

Oster, H., & Ekman, P. (1981). Expresiones faciales de la emoción. *Studies in Psychology = Estudios de Psicología*, 7, 115-143.

Pablos, S. M., & Pablos, S. M. (2011). Avatar virtual realista con capacidad perceptiva orientado a la robótica social [[Http://purl.org/dc/dcmitype/Text](http://purl.org/dc/dcmitype/Text), Universidad de Valladolid]. En *Avatar virtual realista con capacidad perceptiva orientado a la robótica social*. <https://produccioncientifica.usal.es/documentos/619ca010a08dbd1b8f9eed7d>

Roza, S. (2021, enero 8). La universalidad de las expresiones faciales de la emoción. *ASENOVE*. <https://www.asenove.es/universalidad-expresiones-faciales-emocion/>

Técnicas y programas de estimulación cognitiva. (2018, julio 31). Bitbrain. <https://www.bitbrain.com/es/blog/tecnicas-programas-estimulacion-cognitiva>

The Heretic: Digital Human | Tutorial Projects | Unity Asset Store. (2022, enero 10).

<https://assetstore.unity.com/packages/essentials/tutorial-projects/the-heretic-digital-human-168620>

Turner, C. W., Lewis, J. R., & Nielsen, J. (2006). *Edited by Waldemar Karwowski, Boca Raton, FL: CRC Press Determining Usability Test Sample Size.*

Unity—Manual: Animation Clip (Clip de Animación). (2016).

<https://docs.unity3d.com/es/530/Manual/class-AnimationClip.html>

Unity—Manual: Animator Controller. (2016). <https://docs.unity3d.com/es/530/Manual/class-AnimatorController.html>

Unity—Manual: GameObject. (2016). <https://docs.unity3d.com/es/530/Manual/class-GameObject.html>

Unity—Manual: Usando Components. (2016).

<https://docs.unity3d.com/es/530/Manual/UsingComponents.html>

Unreal Engine | MetaHuman. (s. f.). Unreal Engine. Recuperado 4 de julio de 2022, de

<https://www.unrealengine.com/en-US/metahuman>

What are the Best UI Designs for Medical Apps? (2021, marzo 25). VironIT.

<https://vironit.com/what-are-the-best-ui-designs-for-medical-apps>

ANEXOS

Definiciones

Sistema de Codificación Facial, FACS: es un sistema para denominar movimientos faciales humanos por su apariencia en la cara, basado en un sistema desarrollado originalmente por un anatomista sueco llamado Carl- Herman Hjortsjö.¹ Fue tomado después por Paul Ekman y Wallace V. Friesen, y publicado en 1978 Ekman, Friesen, and Joseph C. Hager publicaron una importante actualización de la Programación y Codificación facial en 2002.³ Los movimientos individuales de los músculos faciales son codificadas por ' FACS ' desde leves cambios instantáneos en la apariencia facial. («Sistema de Codificación Facial», 2022)

Unidad de acción mínima, unidad de acción, AU: contracción o la relajación de uno o más músculos. Son una serie de descriptores de acción, que se diferencian de las unidades específicas de acción debido a que la codificación facial no ha especificado la base muscular para la acción y tienen comportamientos específicos no distinguidos con tanta precisión como lo han hecho por la AU. (Coan et al., 2007)

Expresión facial: es uno de los medios más importantes para expresar emociones y estados de movimiento y no como un objeto estático podemos conseguir una mejor comprensión de lo que nos comunican los demás. También hacemos juicios sobre la personalidad y otros rasgos de la gente en función de lo que vemos en sus caras. No toda la comunicación que se transmite a través de la expresión facial es susceptible de ser percibida conscientemente por el interlocutor; sin embargo, sí se sabe que las impresiones que obtenemos de los otros también están influidas por los movimientos imperceptibles de su comunicación verbal. («Expresión facial», 2022)

Expresión facial universal: La evidencia científica nos indica que hay 7 expresiones faciales de la emoción que son universales y se manifiestan en cualquier persona independientemente de su raza, cultura, nacionalidad, religión, género o cualquier otra variable demográfica. Éstas son: Ira, Desprecio, Miedo, Asco, Alegría, Tristeza y Sorpresa. Además, se considera que estas expresiones pueden tener un aprendizaje innato ya que no está relacionado con sus factores socioculturales. (Roza, 2021)

Expresión facial compleja: son aquellas expresiones que no forman parte dentro del grupo de las expresiones faciales universales. Pueden ser derivadas de las otras o expresiones totalmente nuevas construidas a partir de la contracción de músculos diferentes. Estas expresiones según la evidencia científica si están relacionadas con el entorno de las personas y sus costumbres y cultura, se considera que son aprendidas por los progenitores y a una edad muy temprana.

Rigging: es el proceso de crear un sistema de controles digitales y agregárselos a un modelo 3D para que así pueda ser animado fácil y eficientemente. Este es un paso crucial dentro del proceso de la creación de una animación 3D. (*¿ Qué es Rigging ?*, 2011)

La estructura necesaria para poder deformar y animar personajes. De la misma forma que haríamos con una marioneta, se crea una estructura de huesos y controles dentro de una geometría para que los animadores puedan animar a el cuerpo, músculos, faciales, piel, pero también la ropa y el cabello de los personajes mediante dichos Riggs. (Cortés, 2021)

Game Object: son objetos fundamentales en Unity que representan personajes, props, y el escenario. Estos no logran nada por sí mismos, pero funcionan como contenedoras para Components, que implementan la verdadera funcionalidad. Por ejemplo, un objeto Light es creado al adjuntar un componente Light a un GameObject. (*Unity - Manual: GameObject*, 2016)

Components: son las tuercas y tornillos de los objetos y comportamientos de un juego. Son las piezas funcionales de cada GameObject. Si todavía no entiende la relación entre Components y GameObjects, lea la página de GameObjects antes de seguir adelante. (*Unity - Manual: Usando Components*, 2016).

Animation Clip: son los pequeños bloques de construcción de la animación en Unity. Ellos representan las piezas aisladas del movimiento, como lo son RunLeft (Correr a la izquierda), Jump (Saltar), o Crawl (Gatear), y pueden ser manipulado y combinado en varias maneras para producir resultados al final vivos (vea Animation State Machines, Animator Controller, o Blend Trees). Estos Clip son el movimiento del esqueleto del modelo en cuestión con una duración y transformación de sus propiedades. (*Unity - Manual: Animation Clip (Clip de Animación)*, 2016)

Animator Controller: es creado dentro de Unity y permite arreglar y mantener un conjunto de animaciones para un personaje u objeto. En la mayoría de las situaciones, es normal tener múltiples animaciones y cambiarles entre sí cuando ciertas condiciones de juego ocurren. Desde la herramienta el usuario puede crear maquinas de estado en cada capa que permita reproducir simultáneamente varios Animations Clips a la vez. Las transiciones de la maquina de estados pueden ser mediante condiciones o cuando una animación termina transaccione a otro estado. En definitiva, es el cerebro que sigue Unity para reproducir las animaciones de sus modelos. (*Unity - Manual: Animator Controller*, 2016)

Keyframes: durante la comprensión de un video el key frame es aquel fotograma que se toma como referencia con el fin de almacenar solo dicho fotograma y a partir de este almacenar los siguientes fotogramas para ahorrar espacio. Esta técnica permite generar movimiento a través de fotogramas clave que almacenan un estado en un determinado momento. En Unity estos keyframes almacenan propiedades en cada momento de las animaciones para que unity

calcule las transformaciones intermedias y así poder crear el movimiento. (Agarwala et al., 2004)

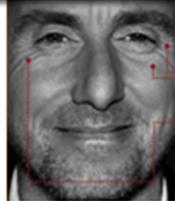
Expresiones faciales Universales

Las expresiones faciales universales son aquellas que según investigaciones realizadas (*Expresiones faciales universales propuestas por Ekman (1972). | Download Scientific Diagram, 2017*) tiene ciertas características comunes: tienen desencadenantes psicológicos subyacentes, lo cual significa que no son expresiones faciales aprendidas, sino que son adquiridas de manera innata; tienen muestras fisiológicas únicas; pruebas cognitivas comunes en todas las culturas; sentimientos y experiencias interculturales.

Los humanos no solo disponen de las expresiones faciales universales, sino que son capaces de representar muchos tipos de expresiones faciales más complejas que representan diferentes emociones. La diferencia de estas expresiones complejas con las universales reside en las características que se han nombrado, que son compartidas por las universales, pero no las tienen las complejas, o al menos aun no se ha sido capaz de identificarlas. En definitiva, la clave cuando se habla de expresiones faciales universales frente a las complejas es la capacidad de ser innatas o aprendidas por condiciones culturales respectivamente.

Nivel 1

Las SIETE expresiones faciales básicas según Paul Ekam

	ALEGRÍA →	
	TRISTEZA Párpados superiores caídos, el entrecejo se arruga, ligera caída de los labios.	
	IRA →	
	SORPRESA Los párpados superiores suben, pero los inferiores no están tensos. La mandíbula suele caer.	
	ASCO →	
	MIEDO Párpados superiores elevados al máximo e inferiores tensos. Las cejas levantadas se acercan. Los labios se alargan hacia atrás.	
	DESPRECIO →	

Expresiones faciales universales según Paul Ekam explicadas

Como vemos en la Ilustración según Paul Ekam, que ha sido un psicólogo pionero en el estudio de las emociones, existen siete expresiones faciales universales, las cuales son formadas mediante la combinación de varias unidades de acción mínimas recogidas en el FACS.

Emoción ↕	Unidades de acción ↕
Felicidad	6+12
Tristeza	1+4+15
Sorpresa	1+2+5B+26
Miedo	1+2+4+5+7+20+26
Coraje	4+5+7+23
Disgusto	9+15+16
Desprecio	R12A+R14A

Combinación de unidades de acción que forman las expresiones faciales universales

En la Ilustración vemos que la felicidad está formada por la combinación de dos unidades de acción, que son 6 - Levantamiento de mejillas y 12 - Tiramiento labial esquinal, que combinadas crean una expresión de felicidad en el rostro humano. La tristeza está formada por: 1 - Levantamiento interior de ceja, 4 - Bajar cejas, 15 - Depresión labial esquinal, las cuales al combinarse forman en el rostro humano la expresión de tristeza. La sorpresa formada por las AUs: 1 - Levantamiento interior de ceja, 2 - Levantamiento exterior de ceja, 5 - Levantamiento del párpado superior y 26 - Caída de la mandíbula, con estos movimientos musculares se representa la emoción de sorpresa. El miedo es una de las más complejas dentro de las universales, ya que está formada por muchas unidades de acción, como lo son 1 - Levantamiento interior de ceja, 2 - Levantamiento exterior de ceja, 4 - Bajar cejas, 5 - Levantamiento del párpado superior, 7 - Apretar párpados, 20 - Apretar labios y 26 - Caída de la mandíbula, representando todas ellas combinadas el miedo en un humano de manera no verbal. El coraje se representa mediante las unidades de acción siguientes: 4 - Bajar cejas, 5 - Levantamiento del párpado superior, 7 - Apretar párpados, 23 - Embudo labial, formando estas 4 AUs la sensación de coraje en un rostro humano. El disgusto está formado por tres movimientos musculares mínimos que son 9 - Arrugar la nariz, 15 - Depresión labial esquinal y 16 - Depresión labial frontal formando así la expresión facial de disgusto. Por último, dentro de las expresiones faciales universales está el desprecio, la cual se forma por 12 - Tiramiento labial esquinal derecho y 14 - Hoyuelo facial derecho, formando así la representación de la emoción de desprecio en el rostro humano.

Todas las expresiones faciales, incluidas las complejas se forman igual que estas, mediante la combinación de unidades de acción mínimas que acaban creando una apariencia en el rostro que representa la emoción en cuestión que se siente en cada momento. Como vemos aparte de definir cada unidad de acción se define también en caso de ser unidades de acción espejo en qué lado de la cara es si derecho o izquierdo. Además de esta combinación se puede aplicar una intensidad para representar un mayor o menor grado de esa emoción.

Diagramas

Objetivos

OBJ-0001	Reproducción de expresiones faciales
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación deberá permitir reproducir una expresión ya creada. Será necesario que el usuario decida cuándo quiere reproducirla y cuando quiere pararla.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

OBJ-002	Selección de la intensidad de la expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación permitirá elegir una intensidad de la expresión a la que queremos que se reproduzca la intensidad, de esta forma buscaremos que no siempre la expresión vaya de la intensidad mínima hasta la máxima, sino que el usuario elija.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

OBJ-003	Creación de expresiones faciales personalizables
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación permitirá crear expresiones personalizables por parte del usuario a partir de unidades de acción. El usuario seleccionará la suma de estas que quiere que forme la expresión en cuestión que quiera crear.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

OBJ-004	Almacenar las expresiones faciales
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación dejará no solo editar expresiones que nos aporte de inicio, sino también crear nuevas a nosotros, guardarlas en disco y cargarlas.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

OBJ-005	Parametrizar valores de la reproducción de expresiones faciales
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación deberá permitir al usuario elegir cuánto quiere que dure la reproducción de una expresión, desde que intensidad hasta que otra queremos que se reproduzca.
Subobjetivos	Ninguno
Importancia	Media
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Este objetivo es algo muy amplio pero opcional, no se debe cumplir todo, pero una vez hechos los primeros prototipos de la aplicación decidir qué parámetros queremos dejar al usuario que elija en la reproducción.

OBJ-0005	Parametrizar intensidad de unidades de acción independientes
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Descripción	La aplicación dejará seleccionar la intensidad independiente de cada unidades de acción que se añadan a las expresiones faciales que cree el usuario
Subobjetivos	Ninguno
Importancia	Baja
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

Requisitos Funcionales

FRQ-0001	Reproducir expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0001 Reproducción de expresiones faciales
Descripción	La aplicación permitirá al usuario elegir la expresión facial que desea ver en ese momento y reproducirla de tal manera que pueda verla
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0002	Pausar expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0001 Reproducción de expresiones faciales
Descripción	La aplicación permitirá al usuario pausar en un momento en el que la expresión se esté reproduciendo para así ver la expresión en momentos determinados y no siempre como una animación
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0003	Seleccionar intensidad de la expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0002 Selección de la intensidad de la expresión facial
Descripción	La aplicación dará la opción al usuario de cuál es la intensidad de la expresión facial a la hora de reproducirla ya que no siempre deseara que sea igual.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0004	Seleccionar unidades de acción para añadir a expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables
Descripción	La aplicación permitirá al usuario elegir que unidades de acción son las que serán la suma de lo que formará una expresión facial que esté editando.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0005	Seleccionar unidades de acción para quitar a expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables
Descripción	La aplicación permitirá al usuario elegir que unidades de acción de las que ha añadido para la creación de la expresión facial quiere quitar para que no forme parte de esta.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0006	Crear nueva expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0004 Almacenar las expresiones faciales
Descripción	La aplicación permitirá al usuario crear nuevas expresiones faciales totalmente personalizables, cada una estará identificada con un nombre.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0007	Cargar expresión facial existente
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0004 Almacenar las expresiones faciales
Descripción	La aplicación permitirá al usuario recuperar expresiones que él haya creado o que vengan en el programa base para que las reproduzca o las edite.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-

Comentarios	Ninguno
--------------------	---------

FRQ-0008	Guardar expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0004 Almacenar las expresiones faciales
Descripción	La aplicación permitirá guardar una expresión nueva que se haya creado o la edición de una ya existente para sobrescribir una vez el usuario haya editado esta expresión facial.
Subobjetivos	Ninguno
Importancia	Alta
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0009	Elegir tiempo de reproducción de expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0005 Parametrizar valores de la reproducción de expresiones faciales
Descripción	La aplicación permitirá al usuario elegir si quiere que la reproducción de la expresión facial dure más o menos.
Subobjetivos	Ninguno
Importancia	Media
Urgencia	PD
Estado	-
Estabilidad	-

Comentarios	Ninguno
--------------------	---------

FRQ-0010	Elegir modo de reproducción de expresión facial
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0005 Parametrizar valores de la reproducción de expresiones faciales
Descripción	La aplicación permitirá al usuario elegir si quiere que la reproducción vaya de inicio y a fin y finalice, o que se reproduzca en bucle.
Subobjetivos	Ninguno
Importancia	Media
Urgencia	PD
Estado	-
Estabilidad	-
Comentarios	Ninguno

FRQ-0011	Elegir intensidad de las unidades de acción
Versión	1.0
Autores	Enrique Hernández Hernández
Fuentes	
Dependencias	OBJ-0006 Parametrizar intensidad de unidades de acción independientes
Descripción	La aplicación permitirá al usuario elegir cuando está editando o creando expresiones faciales la intensidad máxima de cada unidad de acción en esa expresión.
Subobjetivos	Ninguno
Importancia	Baja
Urgencia	PD
Estado	-

Estabilidad	-
Comentarios	Ninguno

Casos de Uso

UC-001	Reproducir expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0001 Reproducción de expresiones faciales	
Descripción	La aplicación permitirá al usuario elegir la expresión facial que desea ver en ese momento y reproducirla de tal manera que pueda verla	
Precondición	Que la expresión facial este pausada	
Secuencia normal	Paso	
	1	El sistema activará la reproducción de la animación de la expresión facial
Postcondición	Que la expresión facial se esté reproduciendo	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	

Comentarios	Ninguno
--------------------	---------

UC-002	Pausar expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-001 Reproducción de expresiones faciales	
Descripción	La aplicación permitirá al usuario pausar en un momento en el que la expresión se esté reproduciendo para así ver la expresión en momentos determinados y no siempre como una animación	
Precondición	Que la expresión facial se esté reproduciendo	
Secuencia normal	Paso	
	1	El sistema pausara la reproducción de la animación de la expresión facial
Postcondición	Que la expresión facial se pare	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-003	Seleccionar intensidad de la expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-002 Selección de la intensidad de la expresión facial	
Descripción	La aplicación dará la opción al usuario de cuál es la intensidad de la expresión facial a la hora de reproducirla ya que no siempre deseara que sea igual.	
Precondición	Estar en modo reproducción	
Secuencia normal	Paso	
	1	El sistema cambia el porcentaje hasta el que se llega a reproducir las animaciones de las unidades de acción
Postcondición	Que el porcentaje máximo de la animación de las unidades de acción cambie	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-004	Seleccionar unidades de acción para añadir a expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables	
Descripción	La aplicación permitirá al usuario elegir que unidades de acción son las que serán la suma de lo que formará una expresión facial que esté editando.	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario selecciona de una colección de unidades de acción cual quiere añadir a la expresión
	2	El sistema añade la unidad de acción a la expresión facial actual y esta será la suma de todas las añadidas
Postcondición	La expresión facial será la suma de todas las unidades de acción anteriores más la nueva añadida	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-005	Seleccionar unidades de acción para quitar de la expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables	
Descripción	La aplicación permitirá al usuario elegir que unidades de acción de las que ha añadido para la creación de la expresión facial quiere quitar para que no forme parte de esta.	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario selecciona de una colección de unidades de acción pertenecientes a una expresión cual quiere quitar de la expresión
	2	El sistema quita la unidad de acción de la expresión facial actual
Postcondición	La expresión facial será la suma de todas las unidades de acción añadidas menos la que se acaba de quitar	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-006	Crear nueva expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0004 Almacenar las expresiones faciales	
Descripción	La aplicación permitirá al usuario crear nuevas expresiones faciales totalmente personalizables, cada una estará identificada con un nombre.	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario selecciona crear una nueva expresión facial y le da nombre
	2	El sistema almacena el nombre de la expresión y espera que el usuario la edite
Postcondición	Se creará la expresión facial nueva y vacía	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-007	Cargar expresión facial existente	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0004 Almacenar las expresiones faciales	
Descripción	La aplicación permitirá al usuario recuperar expresiones que él haya creado o que vengan en el programa base para que las reproduzca o las edite.	
Precondición		
Secuencia normal	Paso	
	1	El usuario elegirá de una colección de expresiones faciales guardadas en disco cual quiere recuperar.
	2	El sistema cargará la expresión facial en memoria
Postcondición	La colección de unidades de acción cargadas en memoria son las de la expresión seleccionada	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	
Comentarios	Ninguno	

UC-008	Guardar expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0004 Almacenar las expresiones faciales	
Descripción	La aplicación permitirá guardar una expresión nueva que se haya creado o la edición de una ya existente para sobrescribir una vez el usuario haya editado esta expresión facial.	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario selecciona guardar la expresión que está actualmente editando
	2	El sistema guardará en disco la información de la memoria que contiene todo lo referente a la expresión facial
Postcondición	La colección de unidades de acción cargadas en memoria se almacenan en disco	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	

Comentarios	Ninguno
--------------------	---------

UC-009	Elegir tiempo de reproducción de expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0005 Parametrizar valores de la reproducción de expresiones faciales	
Descripción	La aplicación permitirá al usuario elegir si quiere que la reproducción de la expresión facial dure más o menos.	
Precondición	Estar en modo reproducción	
Secuencia normal	Paso	
	1	El usuario selecciona la duración que quiere para la animación total de la expresión facial.
	2	El sistema cambiará el tiempo máximo que tarda de en ir desde el inicio hasta el final de la colección de animaciones que forman la expresión facial
Postcondición	La colección de unidades de acción cargadas en memoria se almacenan en disco	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Media	
Urgencia	PD	
Estado	-	

Estabilidad	-
Comentarios	Ninguno

UC-010	Elegir modo de reproducción de expresión facial	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0005 Parametrizar valores de la reproducción de expresiones faciales	
Descripción	La aplicación permitirá al usuario elegir si quiere que la reproducción vaya de inicio y a fin y finalice, o que se reproduzca en bucle.	
Precondición	Estar en modo reproducción	
Secuencia normal	Paso	
	1	El usuario selecciona si quiere cambiar el modo de reproducción
	2	El sistema cambia el modo de reproducción
Postcondición	El modo de reproducción cambia	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Media	
Urgencia	PD	
Estado	-	

Estabilidad	-	
Comentarios	Ninguno	
UC-011	Elegir intensidad de las unidades de acción	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0006 Parametrizar intensidad de unidades de acción	
Descripción	La aplicación permitirá al usuario elegir si quiere que la reproducción vaya de inicio y a fin y finalice, o que se reproduzca en bucle.	
Precondición	Haber realizado mínimo 1 vez el UC-004 Seleccionar unidades de acción para añadir a expresión facial	
Secuencia normal	Paso	
	1	Tras el usuario realizar el UC-004 al usuario se le permite elegir la intensidad máxima que quiere que tenga esa unidad de acción determinada
	2	El sistema asigna a esa unidad de acción dentro de la expresión facial un porcentaje máximo.
Postcondición	Que cambie el porcentaje máximo de una unidad de acción	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Baja	
Urgencia	PD	
Estado	-	

Estabilidad	-
Comentarios	Ninguno

UC-012	Cambiar a modo edición	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables	
Descripción	La aplicación cambiará la información mostrada al usuario para permitirle editar las expresiones faciales	
Precondición	Estar en modo reproducción	
Secuencia normal	Paso	
	1	El usuario selecciona cambiar al modo edición
	2	El sistema cambia la información en pantalla
Postcondición	Estar en modo edición	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	

Comentarios	Ninguno
--------------------	---------

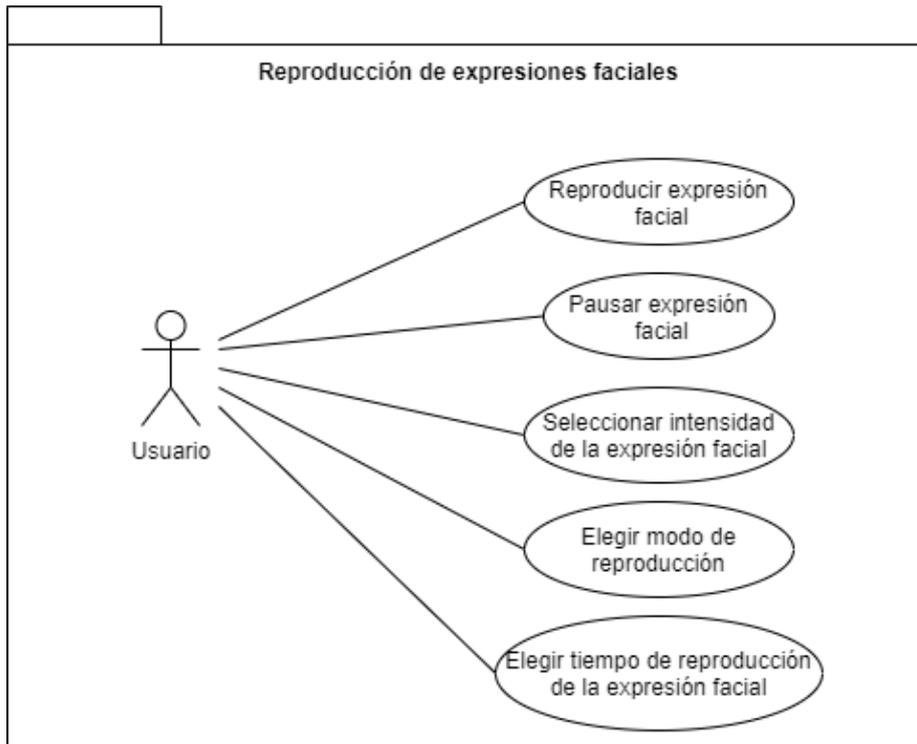
UC-013	Cambiar a modo reproducción	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0001 Reproducción de expresiones faciales	
Descripción	La aplicación cambiará la información mostrada al usuario para permitirle editar las expresiones faciales	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario selecciona cambiar al modo reproducción
	2	El sistema cambia la información en pantalla
Postcondición	Estar en modo reproducción	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	
Estabilidad	-	

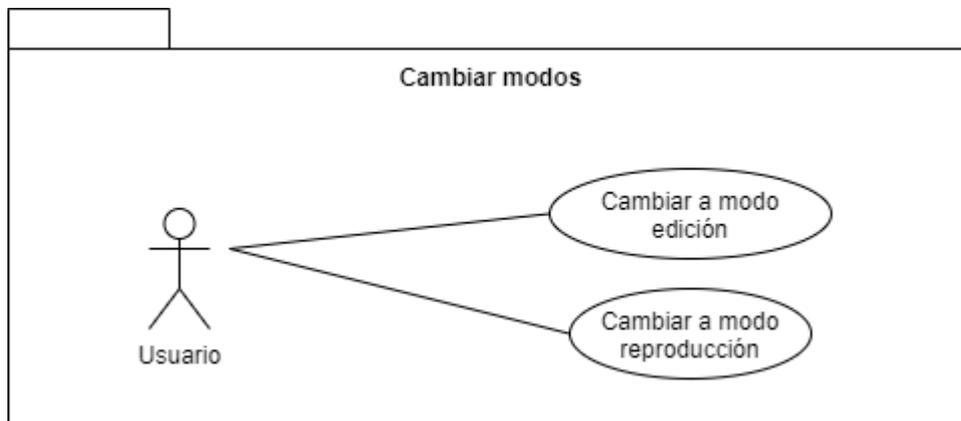
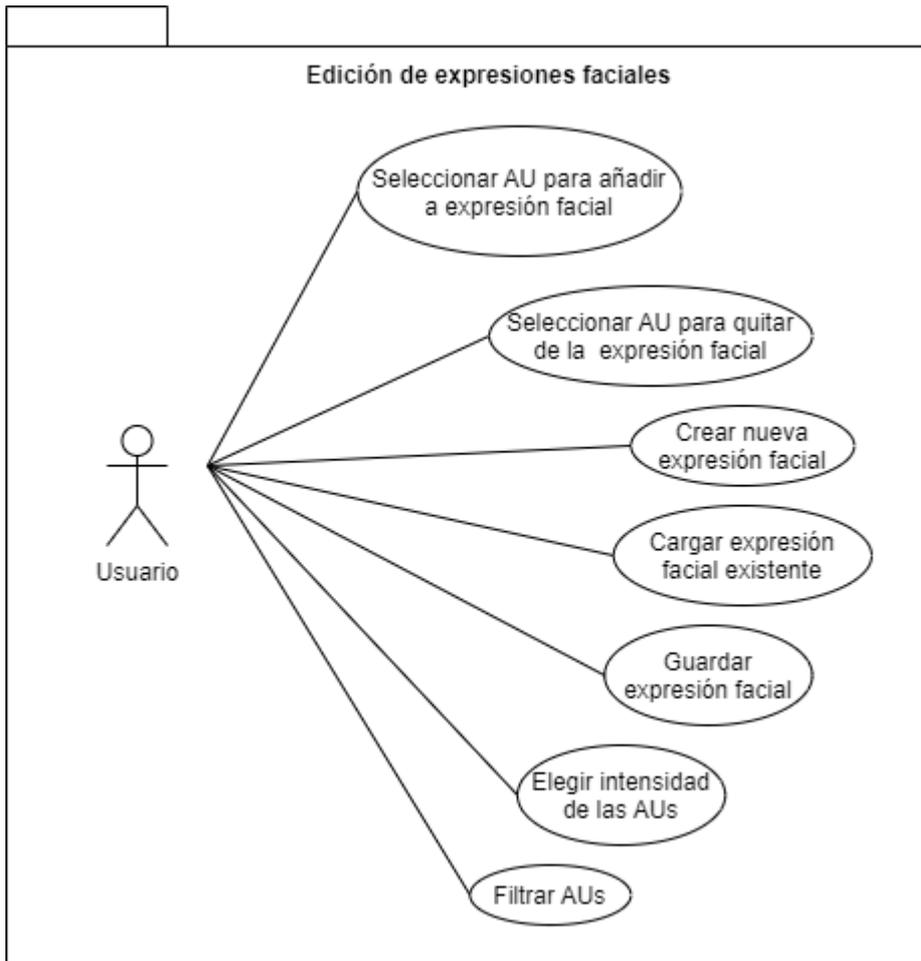
Comentarios	Ninguno
--------------------	---------

UC-014	Filtrar unidades de acción	
Versión	1.0	
Autores	Enrique Hernández Hernández	
Fuentes		
Dependencias	OBJ-0003 Creación de expresiones faciales personalizables	
Descripción	La aplicación tendrá una colección de todas las unidades de acción disponibles para crear expresiones faciales, y el usuario podrá filtrar esta colección por el nombre de cada una de las unidades de acción	
Precondición	Estar en modo edición	
Secuencia normal	Paso	
	1	El usuario introduce el filtro para buscar unidades de acción en concreto
	2	El sistema filtra la colección de unidades de acción en función de lo introducido por el usuario
Postcondición	Colección de unidades de acción filtradas	
Excepciones	Paso	
Rendimiento	Paso	
Importancia	Alta	
Urgencia	PD	
Estado	-	

Estabilidad	-
Comentarios	Ninguno

Esquema de casos de uso





Notas de reuniones con tutor del TFG

PRIMERA REUNIÓN

=====

¿Qué es lo primero que hay que hacer?

Si empezamos por el DCU, tengo que partir de cero o hacer las encuestas destinadas hacia la idea que tenemos

¿Qué versión de Unity deberíamos de usar (una más moderna, o una más estable)?

Mas estable

¿De cara al informe nos basaremos en que nos centramos?

En el diseño centrado en el usuario

Funcionalidades extras para añadir: guardar expresiones para que sean recuperables por el usuario

SEGUNDA REUNIÓN

=====

El tutor recomienda seguir esquema de ingeniería de la usabilidad y accesibilidad

Se plantea el debate de si debíamos usar Unreal Engine 4 y la herramienta de MetaHumans o usar Unity y buscar un avatar hiperrealista que nos permitiera modelar las expresiones faciales necesarias para la aplicación.

Hemos dado un plazo para encontrar un avatar realista que nos diera la funcionalidad necesaria para la aplicación y que fuera compatible con el motor que quería usar, que era Unity. Encontré un modelo investigando en la tienda de Unity ([link](#)).

Partes de la memoria del TFG:

Introducción de qué queremos hacer

Estado del arte (que hay hecho por ahí)

Buscar necesidades (más bien requisitos)

Decisiones de diseño

Desarrollo

Pruebas con usuarios

Conclusiones

La aplicación como mínimo debe tener implementada cara y con una interfaz seamos capaces de mover partes de esa cara al porcentaje que deseemos. Además, poder ajustar las emociones totales que permitan controlar el porcentaje.

El tutor sugiere como objetivo para la próxima reunión parametrizar la cara, refiriéndose a:

Crear unidades independientes de las AUs

Poder activar las AUs desde la interfaz y también las expresiones universales

Combinar AUs para obtener expresiones complejas

Temporizar las AUs (introducir tiempo en el que deseamos hacer la expresión para que sea más rápido o mas lento)

Poder crear combinaciones de AUs para expresiones complejas (opcional)

Poder crear modo experto y modo simple (opcional)

Sobre la memoria lo primero es forzarse a hacer el índice para que después sea más fácil escribir y distribuir el trabajo que se ha hecho.

-Necesidad del proyecto

-Estado del arte

-Soluciones estudiadas

-Diseño

-Programación de la solución

-Pruebas

Según el tutor es recomendable ir escribiendo de forma esquemática la memoria para que a la hora de desarrollarla sea más fácil y este toda la información plasmada.

=====

Dudas planteadas al tutor durante la reunión:

En el documento, tengo que hacer un tutorial de cómo voy haciendo todo en Unity o simplemente explicarlo de una manera general

El tutor recomienda explicar las cosas mas destacables de la aplicación para que se vea el trabajo desarrollado y el esfuerzo dedicado.

¿Podemos ir empezando con las pruebas de usuario para probar interfaces (como hacerlo con papel o como porque es una interfaz muy compleja y para ordenador es más difícil)?

Si, es recomendable que se hagan prototipos para mejorar al máximo posible en la versión final la usabilidad de la aplicación

¿Explico desde un inicio que es lo que vamos a hacer o lo voy desarrollando a lo largo del trabajo?

La idea principal debe de estar explicada al inicio, pero si durante el desarrollo surgen ideas que mejoran el software es mas conveniente añadirlo posteriormente en la memoria.

Se comenta con el tutor algunas unidades de acción que han dado problemas a la hora de implementarse:

2 Outer Brow Raiser (unilateral, right side): no se si tengo que hacer una para cada ceja, ya que luego se podran juntar (y así levantar las dos a la vez)

11 Nasolabial Deepener: no hay músculos para hacer eso

22 (with AU25) Lip Funneler, 23 Lip Tightener: se parecen bastante y es complicado hacerlo MUY diferente

25 Lips part, 26 (with AU25) Jaw Drop: me parecen la misma, creo que la diferencia es que en una se ven los dientes de abajo

45 Blink, 46 Wink: las veo iguales que cerrar los ojos (43 Eyes Closed) pero más rápido

El tutor sugiere revisar la lista de las AUs más importantes para descartar las parecidas y evitar mal entendidos por parte de los usuarios

Explicación de la idea de interfaz que había, de la cual surge la idea de crear dos modos para mayor comodidad del usuario. Uno dedicado a la reproducción de la expresión facial que se este cargada en ese momento. Otro dedicado a editar las expresiones faciales mediante el uso de las unidades de acción.

Además se habla de problemas que han surgido desde la anterior reunión sobre parametrizar el rostro del avatar:

- Mezclar AUs que hagan movimientos contrarios (mover labio hacia abajo y hacia arriba) -> soluciones ajustar los pesos
- Duración en frames y tiempo porque en ordenadores más lentos puede dar problemas
- Coger imagen de musculatura de la cara para ver cómo funcionan

De cara a la búsqueda de necesidades se basan en búsqueda bibliográfica, hablar con usuarios (en artículos) y Samuel. Además, el tutor ha hablado con personal sanitario, encontrando el problema de que los estudios previos se basaban en avatares poco realistas. Además, hay estudios anteriores que son similares pero que no hacen exactamente la tarea a la que vamos a dedicar la herramienta.

REUNIÓN 27/04/2022

=====

El tutor propone como algo opcional, pero que resultaría muy útil para ampliar la funcionalidad de la herramienta, poder ajustar intensidad a cada AU de cada expresión

El primer paso a partir de ahora es toda la funcionalidad ya implementada darle un estilo consistente para tener una versión inicial, y a partir de ella poder añadir toda la funcionalidad extra.

Funcionalidades IU que debatimos con el tutor que sería interesante implementar:

- AL mover ratón que se quite la "pantalla completa"
- Que los botones están asignados a teclas (reproducir r, pausar p ejemplo)
- Que el slider de intensidad sea como en YouTube reproduciendo un video, pero la expresión
- Usar modelo CRAP
- Mirar menús similares (photoshop), y basar todo en guías de diseño (y explicarlo)

PRÓXIMA REUNIÓN

=====

Resolución de dudas por parte del tutor sobre como escribir las definiciones en la memoria, se acuerda escribir un anexo dedicado a ello.

¿En la memoria es necesario hablar de Requisitos (de almacenamiento, no funcionales y funcionales)?

Es conveniente dejar constancia de todos los requisitos necesarios para la correcta implementación de la aplicación.

Se debate la idea de que el usuario pueda elegir una intensidad máxima desde el modo reproducción para que la animación de la expresión facial no vaya mas de ese punto de intensidad.

¿A la hora de hacer la parte de INSO de la memoria cuando hay que hablar de los actores, pongo 1 que sea el único usuario y listo?

Si, ya que es el único actor involucrado en la aplicación.

¿En qué parte podemos hablar del Elevator Pitch?

Dentro del diseño, aunque se puede escribir en varios sitios ya que no es mas que una frase que explica la funcionalidad básica de la aplicación de forma breve y concisa

¿En diseño ya se puede empezar hablar de la interfaz y de cómo va a ser? ¿También a nivel funcional?

Si, tras el análisis de los requisitos se debe de empezar a diseñar como van a implementarse esos requisitos, es conveniente que el proceso de desarrollo mezcle la parte funcional y de diseño unida ya que una cosa no puede existir sin la otra.

Se habla sobre añadir otro slider al lado del de duración que indique la intensidad máxima de la expresión. Además de manera visual se puede añadir una animación a sliders de duración e intensidad que solo sean la palabra y al ponerlo encima se despliegue el panel con el slider como en YouTube

Se acuerda que cuando la animación está en modo el bucle, la vuelta al estado relajado del rostro sea más natural.

REUNIÓN 15/06/2022

=====

Se repasa y comentan aspectos de la memoria sobre la estructura y referencias. Además, también se pregunta sobre el estilo que debe de seguir, que fuente usar, tamaño, etc. El tutor recomienda que las imágenes sean auto explicativas y se sepa porque están ahí solo leyendo su título.

El tutor recomienda crear un menú sobre el panel referente a las expresiones faciales que permita realizar las funcionalidades de guardar, cargar y crear nueva expresión para seguir el modelo CRAP y respetar la cercanía de los elementos dentro de la interfaz.

¿Cómo hacemos para que el usuario pueda abrir nueva expresión sin ir al modo reproducción?

El tutor recomienda añadir dos zonas donde el usuario pueda abrir una expresión ya guardada previamente, una de ellas es crear un panel que le permita hacerlo también en el modo reproducción.

¿Cada vez que guardamos preguntamos al usuario si está seguro de que quiere sobrescribir lo anterior guardado?

Para respetar la heurística de prevención de errores, sería muy recomendable crear cuadros de aviso que el usuario decidiera si quiere volver a verlos o no en cualquier momento. Aunque no es una funcionalidad básica sería muy recomendable de cara a satisfacer la usabilidad

Se encuentra por parte del tutor un error en la cámara de Unity, que falla al crear el primer plano sobre la cara del avatar, dando una imagen deformada de el. Se debe solucionar para que la cámara haga un primer plano correcto y no de una sensación extraña al mirar al avatar.

Sobre el diseño, pregunto si esta bien las guías de estilo aplicadas y si va por buen camino de cara a que resulte familiar y atractiva para los usuarios. El tutor recomienda añadir sombreado a los botones y buscar guías de estilo más amplias como lo son Fluent Design o Material para probar ideas que vengan explicadas y ver como quedan.

El tutor recomienda que añadir atajos de teclado para facilitar a usuarios que tengan mayor manejo sobre la aplicación, además indica que lo óptimo sería que el usuario editara cuales quiere asignar a cada uno, aunque no es esencial.

Sobre guardar los archivos y sobrescribir se llega a la conclusión de que lo recomendable es si el usuario cambia algo y no lo guarda aparezca un asterisco en el botón de guardar indicando que se puede guardar y si sales o cambias de expresión avise al usuario de que puede perder los cambios.

Para diseñar la intensidad de las AUs independientes dentro del modo editar el tutor recomienda poner en cada slot de las unidades de acción pertenecientes a la expresión un botón que lance una ventana emergente que permita con un slider editar la intensidad de esa AU.

REUNION 01/07/22

=====

En cuanto al cuadro de aviso de no haber guardado los cambios surge un problema con los colores y colocación de los botones que dan las opciones. El tutor recomienda usar una x en la ventana para que el usuario la pueda cancelar la operación y dos botones indicando si desea continuarla guardando o sin guardar.

También se pregunta sobre si es conveniente que cuando las opciones están abiertas los paneles no puedan esconderse en el modo reproducción, concluyendo que si porque si el usuario esta editando las opciones y se esconden puede ser contraproducente a nivel usabilidad.

En la presentación del TFG es conveniente probar Unity para enseñar trabajo que se ha realizado en el motor y el ejecutable para que se vea como funciona la aplicación final

Según el tutor cuando se escriben las mejoras implementadas es importante que quede claro que están dentro del proceso iterativo del DCU, y no es algo aparte. Además, recomienda hablar de las métricas de usabilidad de Nielsen y la experiencia de usuario.

Es necesario añadir un diagrama de tareas o wireframe en Flujo de tareas dentro de la memoria. Se pueden usar los esquemas de caso de uso ya diseñados para construir el wireframe ya que son bastantes similares.