

UNIVERSIDAD AUTÓNOMA DE CIUDAD JUÁREZ

INSTITUTO DE INGENIERÍA Y TECNOLOGÍA

Departamento de Posgrado

Maestría en Cómputo Aplicado



# Uso de Realidad Aumentada en Hospitales: Rondas Médicas

Proyecto de desarrollo que presenta como requisito para obtener el grado de

**MAESTRO EN CÓMPUTO APLICADO**

Ing. César Javier Maldonado Flores

**Director de Tesis**

Luis Felipe Fernández Martínez

Ciudad Juárez, Chihuahua, XX de mayo del 2020

## Contenido

<b>I.</b>	<b>Introducción.....</b>	<b>4</b>
1.1	Planteamiento de la oportunidad.....	4
1.2	Solución.....	5
1.2.1	Solución propuesta.....	6
1.2.2	Alcance y Limitaciones.....	6
1.2.3	Impacto.....	6
1.2.4	Metodología de la solución.....	7
1.3	Objetivos.....	8
1.3.1	Objetivos específicos.....	8
1.4	Justificación.....	8
<b>II.</b>	<b>Marco Teórico.....</b>	<b>9</b>
2.1	La Realidad Aumentada.....	9
2.1.1	Industria Manufacturera.....	14
2.1.2	Educación.....	15
2.1.3	Medicina.....	17
2.1.4	Microsoft Dynamics 365 Guides.....	18
2.2	Valor de negocio en RA.....	19
2.2.1	La estrategia de la Realidad aumentada.....	21
2.2.2	Despliegue de realidad aumentada.....	21
2.3	Mixed Reality Toolkit (MRTK).....	23
2.4	Rondas Medicas.....	24
<b>III.</b>	<b>Desarrollo.....</b>	<b>27</b>
3.1	Framework.....	27
3.1.1	Componente de Almacenamiento.....	28
3.1.2	Componente de Soporte.....	29
3.1.3	Componente de Authoring.....	30
3.1.4	Componente de Realidad Aumentada.....	30
3.2	Desarrollo sistema de almacenamiento.....	31
3.2.1	Sistema de almacenamiento – Implementación local.....	31
3.2.2	Sistema de almacenamiento – Implementación en Amazon aws.....	32
3.2.3	Sistema de base de datos.....	34
3.3	Desarrollo sistema de soporte.....	34
3.3.1	Metodología de desarrollo.....	35
3.4	Desarrollo sistema Authoring.....	47
3.4.1	Metodología de desarrollo.....	47

3.5	Desarrollo sistema de Realidad Aumentada .....	57
3.5.1	MixedRealityToolkit de Microsoft .....	57
3.5.2	Configuraciones de Unity .....	58
3.5.3	Prototipos en HoloLens .....	63
3.5.4	Aplicación de RA “Medical Rounds RA - Round” .....	65
IV.	Resultados .....	77
4.1	Marco de trabajo y sus cuatro sistemas .....	77
4.1.1	Componente de almacenamiento .....	78
4.1.2	Componente de soporte .....	78
4.1.3	Componente de authoring .....	78
4.1.4	Componente de RA .....	79
4.2	Presentación de resultados (instanciación del marco de trabajo) .....	80
V.	Conclusiones y trabajo futuro .....	77
	Referencias .....	<b>¡Error! Marcador no definido.</b>
	[Anexo A] .....	102
	[Anexo B] .....	102
	[Anexo C] .....	118
	[Anexo D] .....	133
	[Anexo E] .....	134
	[Anexo F] .....	140
	[Anexo G] .....	144

## **I. Introducción**

Hoy en día, la realidad aumentada (RA) se utiliza en muchas áreas de la interacción humana, como la medicina, la industria, el modelado, el diseño, Internet de las cosas, el aprendizaje en línea y el entretenimiento, entre otros [1] [2] [3]. A menudo se asocia con la realidad virtual (VR). Sin embargo, son dos términos diferentes. Se considera que la RA es la inserción de información virtual, como imágenes, texto, video, audio u objetos 3D, en una vista en tiempo real del mundo real [2] [4] [5]. Uno de los principales beneficios de la RA es la generación de experiencias a través de la visualización de una tarea, proceso o actividad. Esto genera conocimiento relevante para el usuario de RA [2]. Por lo tanto, las aplicaciones de RA deben enfocarse en recrear una interacción visual placentera sin estar saturadas de información [1].

Las aplicaciones desarrolladas para dispositivos de RA varían de muchas maneras, como el lenguaje de programación, la estructura, la funcionalidad, las características, las implementaciones, etc. Además, las capacidades de procesamiento y representación del dispositivo son un factor importante [2]. Por tanto, es una tarea complicada definir un modelo, arquitectura o marco base para las aplicaciones de RA. La experiencia es un factor importante para determinar cómo desarrollar una aplicación RA eficaz. Las aplicaciones de RA deben ser intuitivas, con interfaces que ofrezcan una experiencia de RA útil al usuario [6]. En este trabajo nos enfocamos en diseñar y desarrollar un marco de trabajo para aplicaciones de RA.

El marco no está dirigido a ningún área de dominio específico, tiene un enfoque general. Sin embargo, para propósitos de prueba se realiza una instanciación del marco de trabajo enfocado a las rondas médicas. Este trabajo de tesis se organiza de la siguiente manera. El capítulo 1 describe la introducción y planteamientos del proyecto. El capítulo 2 describe el marco teórico sobre el cual se basa del proyecto. El capítulo 3 describe el desarrollo del proyecto y de los sistemas de la instancia del marco de trabajo. En el capítulo 4 se presentan los resultados del trabajo de tesis. Y finalmente en el capítulo 5 se describen las conclusiones y trabajo futuro del proyecto.

### **1.1 Planteamiento de la oportunidad**

Los hospitales son instituciones de atención a la salud que brindan tratamiento a pacientes por medio de personal médico, enfermería especializada y equipo médico. Todos los días se realizan “rondas” las cuales consisten en una visita a los pacientes del hospital hecha

por médicos, enfermeros, estudiantes o especialistas que se considere necesario para la correcta atención de los pacientes, esto con el propósito de evaluar el estado de salud de los ingresados, conocer su evolución diaria y hacer las indicaciones necesarias para su recuperación y rehabilitación [7].

El principal objetivo de esta práctica médica es observar la evolución de los problemas de salud en los pacientes internados para tomar las medidas correspondientes para su rápida recuperación. Además, tiene el propósito de educar progresivamente al paciente, sus familiares y todo el personal que participa en esta práctica. Es necesario conocer y llevar a la práctica los reglamentos internos de trabajo del hospital en cuestión, además de los protocolos, estándares, guías y manuales de procedimientos de los servicios clínicos de hospitalización [7] [8].

El personal médico que realiza las rondas debe poseer información de los pacientes. La información debe contener, la condición actual en la que se encuentran, tratamientos y medicamentos suministrados. También, se debe realizar una revisión de sus signos vitales y estado de salud en general. Para lograr este propósito utilizan formatos impresos con la información del paciente en los cuales también registran sus observaciones o cambios detectados en la condición del interno. Si fuera necesario obtener más información del paciente como su historial clínico completo, antecedentes, etc. Sería necesario acudir al archivo clínico para obtener dichos datos [7].

El avance tecnológico ha beneficiado muchas áreas como la educación, el sector industrial, la agricultura, la medicina, la economía, etc. Ha mejorado la rapidez, la calidad y los costes de bienes y servicios. Además, ha cambiado la manera en que vemos y compartimos información. Por ejemplo, la realidad aumentada es una tecnología capaz de causar un gran impacto positivo en la forma de realizar las rondas médicas a través de la combinación del mundo físico con elementos virtuales aumentando la experiencia del usuario. La versatilidad que ofrecen algunos de los dispositivos en realidad aumentada le podría permitir al personal médico y de salud obtener y visualizar información de sus pacientes mientras conservan total libertad en las manos. Cambiando la manera en que se realizan las cosas, aumentado la cantidad de información disponible en el menor tiempo posible.

## **1.2 Solución**

### **1.2.1 Solución propuesta**

Se propone un marco de trabajo enfocado en un sistema de realidad aumentada el cual muestre información de pacientes como historial médico, expediente clínico, exploración física, etc. A personal médico durante las rondas médicas.

La información necesaria para las rondas médicas será accedida y compartida a través de 4 sistemas principales del marco de trabajo. Cada sistema será implementado individualmente y realizarán funciones específicas. El sistema de almacenamiento se encargará de almacenar y distribuir la información a los otros tres sistemas. El sistema de soporte administrará usuarios, roles, pacientes y su información y perfiles para el sistema de realidad aumentada. El sistema de “*authoring*” permitirá la creación y gestión de los perfiles para la ronda medica del sistema de realidad aumentada. El sistema de realidad aumentada accederá y mostrará la información de los pacientes y los perfiles del sistema de *authoring*.

### **1.2.2 Alcance y Limitaciones**

- Se diseñarán y desarrollarán 4 sistemas basados en el marco de desarrollo planteado en este documento.
- La información en el sistema de almacenamiento incluirá usuarios y sus roles, pacientes y su información genera, historial clínico, información de exploración física, pruebas y radiografías. Además, la información de los perfiles para el sistema de realidad aumentada.
- Los usuarios, su roles, permisos y accesos a otros sistemas solo podrán ser gestionados a través de un solo sistema.
- La diversidad de sistemas permite la distribución de tareas y actividades al personal adecuado.
- Los sistemas serán desplegados en el modelo de despliegue en la nube pública de AWS.
- En ninguno de los sistemas del marco de trabajo se incluyen módulos de procesos de análisis de la información.

### **1.2.3 Impacto**

- Acceso seguro a la información de pacientes por personal médico conforme a su rol y permisos.
- Acceso rápido y en mayor volumen de la información de pacientes durante las rondas médicas.

- Primer sistema de realidad aumentada para personal médico durante las rondas médicas, a través del dispositivo HoloLens, en hospitales de México.
- Ahorro de tiempo en búsqueda y revisión de archivos relacionados a un paciente durante las rondas médicas.
- Acceso a los sistemas desde cualquier dispositivo u horario sin la instalación de software.

#### **1.2.4 Metodología de la solución**

Como metodología de la solución se plantearon una serie de 7 fases o actividades globales. Con el propósito de desarrollar la solución, algunas de las fases se realizaron de manera individual y otras paralelas.

- Fase1 (Recopilación y revisión de información): Recopilación y revisión de diversas fuentes de información en los temas de “marco de trabajo realidad aumentada”, “Aplicaciones de realidad aumentada” y “rondas medicas”. Artículos, libros, documentos, etc. Que provean un antecedente en el cual formar la base del marco de trabajo.
- Fase 2 (Identificación de componentes): Identificación de componentes para el marco de trabajo a través de la revisión de diversos artículos enfocados a marcos de trabajo, arquitecturas o modelos para la realidad aumentada.
- Fase 3 (Desarrollo del marco de trabajo): Desarrollo de la estructura del marco de trabajo incorporando los componentes identificados anteriormente. Realizar un proceso interactivo de revisiones para determinar una clara y estable versión del marco de trabajo.
- Fase 4 (Desarrollo de los componentes del marco de trabajo): Proceso de diseño y desarrollo de cada uno de los componentes del marco de trabajo enfocados en el área de dominio “Rondas Medicas”.
  - Sistema de Almacenamiento.
  - Sistema de Soporte
  - Sistema de Authoring (Autoría).
  - Sistema de Realidad Aumentada.
- Fase 5 (Despliegue de los sistemas): En esta fase se despliegan los sistemas en sus respectivas plataformas de la nube de AWS.
- Fase 6 (Pruebas): Se realizarán las pruebas a los sistemas del marco de trabajos enfocados en el área de rondas médicas.

- Fase 7 (Documentación): La documentación se realizará continua y constantemente durante la realización de cada fase previamente descrita.

### **1.3 Objetivos**

Incorporar tecnologías emergentes como la realidad aumentada en entornos de hospital para las rondas médicas.

#### **1.3.1 Objetivos específicos**

- 1) Generar un marco de trabajo de realidad aumentada para construir perfiles con la información requerida de los pacientes durante las rondas médicas.
- 2) Crear una instancia del modelo de realidad aumentada.

### **1.4 Justificación**

Implementar tecnología de realidad aumentada en forma de una aplicación o prototipo de software para el área de hospitales y clínicas de México. Cambiando la manera en que el personal médico obtiene y ve la información de los pacientes durante las rondas médicas. Utilizando un dispositivo tecnológico especialmente diseñado para la realidad aumentada.

No existen antecedentes en la utilización de un dispositivo para la realidad aumentada para la asistencia de personal médico durante las rondas médicas en México. De esta manera, la obtención y vista de información de los pacientes como registros médicos, historial clínico, signos vitales, entre otros, se realiza de la forma más rápida posible.

## II. Marco Teórico

A continuación, se plantearán las bases sobre las que se desarrolla este proyecto de realidad aumentada. Es necesario tomar en consideración distintos elementos teóricos y tecnológicos. En la subsección 2.1, se describe la realidad aumentada como tecnología emergente, en la subsección 2.2 se describe el valor de negocio que crea la realidad aumentada y la subsección 2.3 se introduce al marco de trabajo “*Mixed Reality*”.

### 2.1 La Realidad Aumentada

La realidad aumentada (RA) suele ser asociada con la realidad virtual (RV) aun cuando estos dos términos son muy distintos. La realidad aumentada consiste en la inserción de información virtual como imágenes, texto, video, audio u objetos 3D a una visión del mundo real o físico en tiempo real [4]. Azuma nos explica que la RA posee 3 requisitos fundamentales [2]:

- 1) Combina contenido real y virtual
- 2) Es interactivo en tiempo real
- 3) Es registrado en 3D

Por otra parte, la RV consiste en la inmersión del usuario a una realidad virtual, donde el escenario, los objetos, imágenes, textos, etc. están generados de manera digital [3]. En un sistema de RV el usuario es sumergido completamente en un “mundo virtual”. Y es desconectado de la realidad física a través de un aparato diseñado con este propósito. Uno de los dispositivos de RV más famosos son el *Oculus Rift*, figura 2.1. Las interfaces de este tipo de dispositivos se diseñan para mejorar la interacción que tiene el usuario con los objetos virtuales. Su principal objetivo es el remplazo de la realidad con un ambiente virtual en el cual el usuario se desconecte del real [2].

Uno de los principales objetivos de los sistemas de RA consiste en mejorar y complementar la visión del mundo real con elementos virtuales, pero sin llegar a ser una inmersión completa como la RV. Se han diseñado una gran variedad de dispositivos con este fin en particular. Actualmente en el mercado existe una amplia variedad de dispositivos para utilizar aplicaciones tanto de RA como de RV. Sin embargo, algunas de las categorías más importantes en tecnología de RA son: Sensores, utilizados para la localización de objetos en el mundo físico que nos permite tener una interacción más

nítida y real con la RA. Dispositivos de visualización, que permiten la visualización de datos virtuales, lo que mejora la percepción del mundo real por parte del usuario [9].



Figura 2.1 Dispositivo de RA Oculus Rift [10].

La “Realidad Mixta” (RM) es un concepto propuesto por Milgram y Kishino en 1994, importante para describir la RA. Describieron a la RM como la fusión del mundo real y virtual. Además, también propusieron la realidad mixta continua (*Mixed Reality continuum*, en inglés) la cual es una taxonomía de los distintos niveles de fusión entre real y virtual que pueden ser alcanzados como se muestra en la figura 2.2 obtenida de [2]. Básicamente, el mundo real y virtual son puestos en extremos separados y dependiendo de a qué extremo nos acerquemos tendremos RA o RV. La RA está cerca del mundo real pues consiste en la integración de elementos virtuales en una vista del mundo real. Y la RV es la inmersión en un ambiente completamente virtual [2].



Figura 2.2 Realidad Mixta de Milgram y Kishino [2].

El concepto de realidad mixta descrita anteriormente fue ampliado por Mann (1994). De manera que definió un concepto llamado realidad mediada (MR, *Mediated Reality*) como un filtro o modificación a la vista del mundo real. A diferencia de la RA en la cual se agregan elementos virtuales a la vista en el mundo real. Una de las principales características de MR es la sustracción de información molesta o irrelevante para el usuario. De manera que fuese posible eliminar publicidad, anuncios o cualquier otra clase

de elementos reales que no fueran necesarios. En la figura 2.4 se muestra una representación gráfica de la realidad mediada [2].

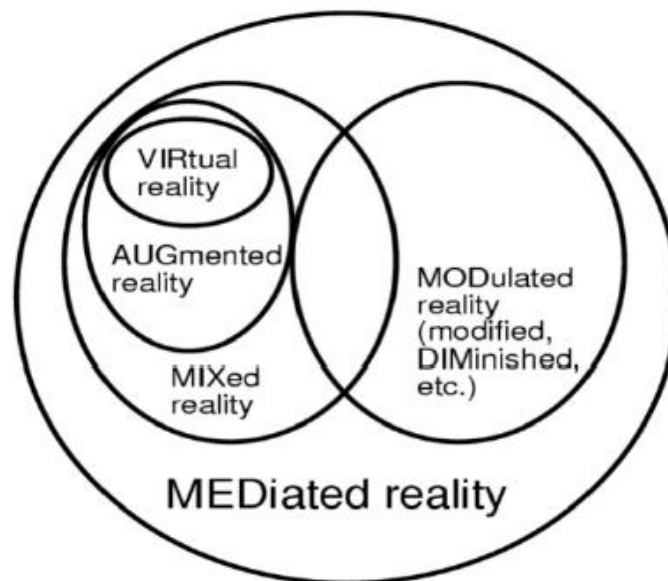


Figura 2.3 Realidad Mediada de Mann (1994) [2].

El concepto de RA ha ido cambiando y evolucionando a través de los años. Mientras más investigación y desarrollo se invertía en esta tecnología, también se fue extendiendo a otras áreas del dominio humano en las cuales no se pensó que podría ser aplicada [11].

Algunos autores consideran que la RA podría ser una poderosa herramienta para la transferencia de conocimientos. Nonaka define el conocimiento como la “información en contexto”. Por lo tanto, se podría asumir que la RA es capaz de transferir información y ponerla en contexto, y por lo tanto transferir conocimiento a los usuarios [11].

Algunas de las principales áreas de investigación de la RA se centran en 3 dominios específicos [12]:

- Autoría (*Authoring, A*): Consiste en las técnicas de software que se centran en la generación de contenido aumentado y mostrarlo en el mundo real.
- Conciencia del contexto (*Context-Awareness, CA*): Consiste en las técnicas de software que se centran en usar la información contextual para caracterizar contenido aumentado.

- Análisis de interacción (*Interaction-Analysis*, IA): Consiste en las técnicas de software que analizan el estatus de la interacción entre el usuario y el contenido aumentado para proveer retroalimentación o mejorar la interacción.

En [12] se menciona que no existe suficiente investigación o literatura de las tres áreas de realidad aumentada anteriores, o investigaciones dedicadas a explicar, explorar, analizar la relación entre ellas ni su capacidad de transferencia de conocimiento. Existen investigaciones y literatura enfocados en distintos proyectos de realidad aumentada que puedan caer en estas áreas. Sin embargo, requeriría una inmensa cantidad de trabajo si fuera necesario revisar todas las aplicaciones de realidad aumentada.

Existen una gran variedad de dispositivos para la implementación de realidad aumentada. Uno de ellos es a través de un dispositivo especialmente diseñado para la realidad aumentada como son los “*GoogleGlass*” y los “*HoloLens*” llamados “*headset*”, ver figura 2.3. Y otra es a través de dispositivos móviles como son los *smartphones*, *tablets*, etc. La principal diferencia entre ambos radica en la forma en que introducen la información extra que conforma la realidad aumentada. Los *headset* consisten principalmente de un par de pantallas de cristal frente al usuario en la cuales es mostrada la información adicional. Por otro lado, los *smartphones* necesitan generar la imagen del entorno digitalmente (Captura por la cámara) y la información adicional (contenido virtual) es plasmado sobre ese entorno. Esto conlleva a un gasto extra en procesamiento por parte de los dispositivos móviles y *tablets* [13].



Figura 2.3 HoloLens y GoogleGlass respectivamente (<https://www.microsoft.com/en-us/hololens>, <https://www.google.com/glass/start/>).

Las aplicaciones que utilizan tecnología de realidad mixta requieren “conocer” las posiciones relativas de la cámara y la escena en las que están trabajando. Cuando ocurren

un movimiento entre la cámara o la escena se debe hacer el seguimiento en tiempo real de los “seis grados de libertad” (ejes x, y, z de la traslación y rotación) que definen la posición y orientación de la cámara en relación con la escena o el desplazamiento 3D de un objeto en relación con la cámara [14].

En RA el rastreo de un elemento u objeto consiste en la constante identificación de la ubicación de dicho elemento. El rastreo en tercera dimensión (rastreo 3D) trata de recuperar los seis grados de libertad continuamente. Por otra parte, los rastreadores mecánicos tienen buena precisión en general, sin embargo, al usuario se ve restringido a un volumen limitado de trabajo [14].

La tecnología de rastreo de un objeto en una secuencia de video, tanto de una grabación como en tiempo real, requiere de su constante y continua localización cuando la cámara, el escenario o el objeto están en movimiento. El rastreo 3D continuamente trata de recuperar los seis grados de libertad de la cámara. Los rastreadores mecánicos son muy precisos, sin embargo, condicionan al usuario aun limitado volumen de trabajo. Los rastreadores de tipo magnético son muy vulnerables a las distorsiones provocadas por el metal y las ondas electromagnéticas. Los rastreadores ultrasónicos tienen a ser interrumpidos por el “ruido”. Por lo tanto, los rastreadores de visión tienen el potencial de proporcionar soluciones no invasivas, precisas y de bajo costo para este problema. La implementación de marcadores especiales o fiduciales<sup>1</sup> es ampliamente aceptada ya que incrementan en gran medida la precisión del rastreo. A los usuarios finales de realidad aumentada no les gustan esta clase de marcadores ya que son visibles en la escena y no siempre es posible modificar el ambiente antes que la aplicación tenga que ejecutarse [14].

Otro importante método para el rastreo es el basado en textura. Si un objeto está lo suficientemente texturizado la información puede ser entregada a través del flujo óptico, comparación de plantillas o punto de interés. El último es el más efectivo para aplicaciones de realidad mixta ya que se basa en emparejar características locales. Para el método a través de punto de interés en lugar de comparar todos los píxeles en una imagen solo unos cuantos son seleccionados inicialmente con algo denominado operador de interés. Reduciendo significativamente el tiempo de cómputo requerido mientras incrementa la fiabilidad si los píxeles son elegidos correctamente [14].

---

<sup>1</sup> Símbolos o marcas utilizados como puntos de referencia para realizar algún proceso o coordinación.

### 2.1.1 Industria Manufacturera

La RA es una tecnología que ha sido investigada durante muchas décadas y probada en distintas áreas de actividad humana. Sin embargo, la industria es un caso en el que la RA tuvo que ser implementada de una manera específica y particular. A diferencia de la educación, en la industria se requiere un análisis a profundidad del área en que se va a implementar y de todos los detalles concernientes a la funcionalidad [15].

Actualmente, la mayoría de las aplicaciones o sistemas de RA para el sector de la industria manufacturera están caracterizados como propietarios. El ambiente industrial es un campo en constante movimiento. Las personas trabajan para alcanzar metas de producción u operativas. Además, su principal interés es el crecimiento económico y se utiliza a la tecnología únicamente como una herramienta para alcanzar este propósito [15]. Algunas de las áreas en las que se espera que las aplicaciones de RA funcionen bien son las siguientes [15]: Diseño de producto, diseño de planta, entrenamiento, asistente de producción, seguro de calidad, logística de producción, mantenimiento remoto.

Para un área como la industria manufacturera no solo un apropiado sistema de software es importante para la aplicación de RA. También es necesario utilizar el hardware adecuando o que mejor pueda adaptarse a este entorno. En [16] se proponen una serie de clasificaciones de dispositivos de RA:

- Pantalla montada en la cabeza (HMD, *Headsets*): Consiste en un dispositivo de pantalla el cual es usado en la cabeza o sobre la cabeza.
- Dispositivos de mano: Dispositivo interactivo que se utiliza en una o dos manos y tiene una pantalla y cámara como los *smartphones*.
- Proyector: Los proyectores láser o LCD/LED pueden mostrar información visual en el mundo real sin necesidad de utilizar un dispositivo sobre sí mismo.
- Seguimiento de usuario: Está compuesto por sensores de movimiento para detectar el movimiento del usuario. Sin embargo, estos dispositivos no suelen ser muy exitosos en un ambiente industrial.
- Háptico y retroalimentación forzada: Dispositivos usables que proveen retroalimentación al usuario sin dejar de realizar la actividad que realiza.

Los sistemas de software de las aplicaciones de RA deben poseer dos elementos fundamentales, rastreo preciso y registro. Permiten que los elementos del mundo real y virtual estén alineados con precisión. Dicha precisión permite que las interacciones con los elementos e información virtual sean cómodas y precisas para el usuario lo que permite que pueda enfocarse en su actividad principal [17].

En [18] los autores introducen un proceso de manufactura en colaboración con la tecnología de RA. ARAM (*Augmented reality aided manufacturing*), “Fabricación asistida con realidad aumentada” es descrito como un subsistema especial el cual está conectado con la realización del producto de manufactura incluyendo programación de manipulación, herramientas de máquina, dispositivos de transporte y almacenamiento, medición, pruebas y diagnóstico de partes y producto ensamblado. En la figura 2.4 se muestra la “idea principal” de lo que conforma ARAM. Son cuatro áreas principales de la industria manufacturera en las que podría ser aplicada la RA para asistir en el proceso. La unión de las cuatro áreas es donde se encuentra ARAM asistiendo en control de robot, transporte y almacenamiento, pruebas y ensamblado.

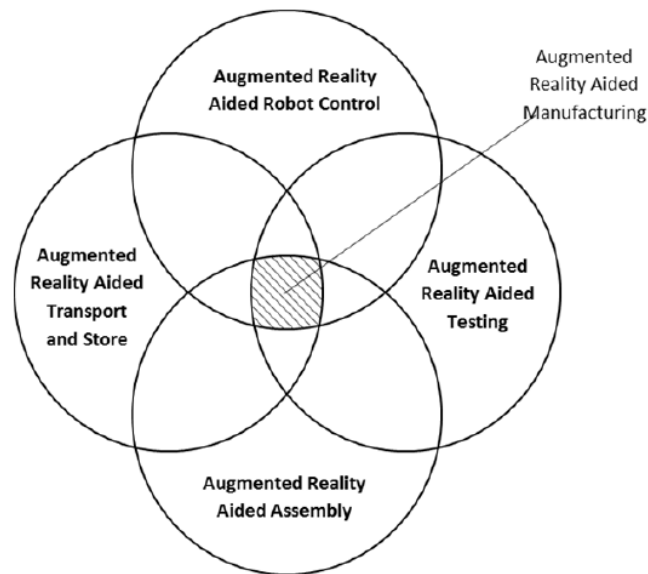


Figura 2.4 Fabricación asistida con realidad aumentada (ARAM) [18].

### 2.1.2 Educación

La educación es una de las áreas de actividad humana en las que mayor investigación se ha realizado con respecto a la RA. Al mejorar la percepción de los usuarios la RA ayuda al usuario a interactuar con su entorno y los elementos virtuales de

manera dinámica. Esta constante interacción estimula a los usuarios a aprender a través de la práctica. Dicho aprendizaje ha demostrado ser mucho más efectivo que el visual o auditivo [1].

Las realidades mixtas se han utilizado en la educación desde hace mucho tiempo. Pinturas murales en cuevas a través de paredes aumentadas es un ejemplo de la transferencia de conocimiento acerca de caza y supervivencia. La imagen aumentada como una solución para el usuario de tener una oportunidad de familiarizarse con conceptos que no se pueden lograr en el mundo físico [1].

Una de las características más importantes de RA en términos de la educación es el espacio flexible y orientado al estudiante, el cual proporciona oportunidades de aprendizaje. El aprendizaje ya no se limita por un espacio limitado y tradicional como un salón de clases o un laboratorio. Se mueve con los usuarios e interactúa con el mundo a su alrededor [1] [19].

Los sistemas de RA más comunes en el área de la educación son los basados en marca y sin marca. Los sistemas basados en marca están compuestos por 3 elementos, una marca o estampo de la cual se obtiene información, un “*gripper*” para obtener esa información de la marca y un cubo para información aumentada en información renderizada 3D en una pantalla. Por otro lado, los sistemas sin marca necesitan un sistema de rastreo que involucra GPS, “Brújula” y un dispositivo de reconocimiento de imagen. Sin embargo, los sistemas sin marcador suelen ser un poco menos precisos [19].

Varios investigadores han sugerido que los alumnos pueden fortalecer su motivación para el aprendizaje y mejorar sus prácticas educativas basadas en el realismo con la realidad virtual y aumentada. A pesar de que a ávido una gran cantidad de investigadores durante las últimas dos décadas, la adopción de RA en educación y capacitación sigue siendo un desafío debido a problemas con su integración con los métodos de aprendizaje tradicionales los costos para el desarrollo y mantenimiento del sistema de RA y la resistencia general a nuevas tecnologías [19].

La capacidad de mostrar información mediante el uso de objetos virtuales que el usuario no puede detectar físicamente utilizando sus sentidos es una característica de los dispositivos de RA “*headset*”. Los *headset* son el dispositivo de RA más conveniente para el aprendizaje, ya que permite que el usuario utilice sus manos para manipular elementos virtualizados que ve a través del dispositivo. Puede cambiar la posición, forma, tamaño

entre otras características virtuales usando los demos, manos y movimientos como sacudir e inclinar [20].

### 2.1.3 Medicina

La medicina es otra de las áreas de actividad humana en la que se ha realizado mucha investigación con respecto a la realidad aumentada. Es necesaria la capacidad de cuantificar y manipular la información del espacio físico del entorno en el que se esté trabajando para relacionar un conjunto de datos con otro es fundamental para la navegación quirúrgica, odontológica, entre otras áreas de la medicina. Donde los conjuntos de datos son las posiciones físicas o coordinadas en el espacio físico [21].

Existe una necesidad en la medicina de visualizar datos médicos y al paciente dentro del mismo espacio físico. Poder monitorear los signos vitales de un paciente durante una operación le permite a un cirujano realizar un trabajo mucho más preciso y con una respuesta más rápida [22].

En [21] se sugiere un sistema de RA para asistir a cirujanos en operaciones médicas. Se explica que una de las mejores maneras de conseguir un sistema de RA con las características necesarias para la tarea, sería un modelo de paciente híbrido. Este modelo consiste en crear una representación virtual del paciente el cual se “empalme” con el paciente en el mundo real. Lo cual permite señalar zonas específicas de la operación en el modelo virtual que sirvan como guía para el cirujano. En la figura 2.5 se muestra un empalamiento de un cuerpo real y uno virtual antes de una operación.

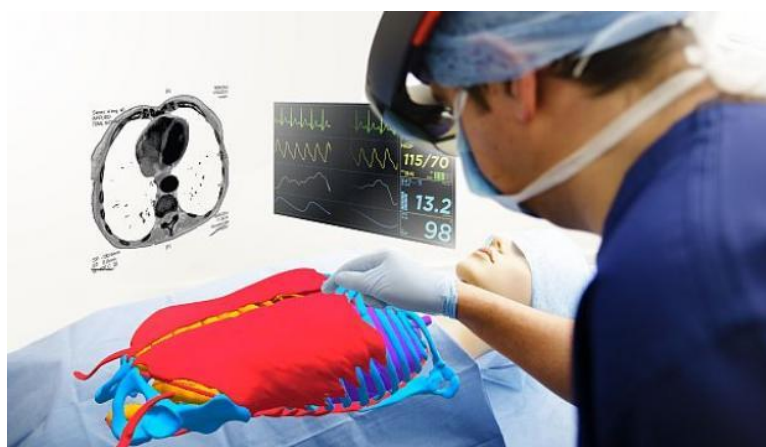


Figura 2.5 Empalamiento de un cuerpo real y uno virtual en RA (<https://healthcare-in-europe.com>).

La fusión de imágenes registradas en la misma pantalla ofrece lo mejor de dos mundos. Una característica adicional proporcionado por este enfoque puede ser una mejor comprensión de la imagen al visualizar el contexto anatómico (cuerpo del paciente) que no ha sido obvio antes. Este es el caso de la cámara endoscópica y las imágenes de ultrasonido, donde cada imagen corresponde solo a una pequeña área [22].

Existen algunos problemas para una correcta implementación de la realidad aumentada relacionadas al registro, rastreo y calibración. La exactitud con la que es posible empalmar un elemento virtual con uno real está limitado en muchos aspectos. Primero, el dispositivo que sea utilizado para implementar el sistema de RA. Si se trata de un dispositivo móvil es necesario utilizar una mano para poder operarlo a menos que se utilizara algún tipo de dispositivo adicional para sujetarlo. La precisión con un dispositivo móvil también es un problema ya que el dispositivo captura el mundo real lo procesa y después agrega los elementos virtuales por lo que hay un pequeño tiempo de espera. La solución óptima para esta clase de inconvenientes es usar un dispositivo tipo *Headset* que pueda ser usado por el cirujano y que le permita mover sus manos libremente. El dispositivo no captura y procesa el mundo real, al ser cristales frente a los ojos del cirujano el observa el mundo real en vivo, los elementos virtuales simplemente se agregan sobre esta imagen. Sin embargo, aún necesita que el hardware procese los elementos virtuales, por lo cual, el hardware deberá tener una capacidad de cómputo y renderización que le permita aplicar cambios en tiempo real [22].

Actualmente, en el área del cuidado de la salud se exige los mejores resultados para el paciente con la mayor rentabilidad posible. El tiempo que el estudiante de medicina pasa bajo la supervisión de un médico veterano es limitada. Y el riesgo de que estudiante inexperto comience operando un paciente vivo es muy peligroso. La RA podría dar solución a este problema proveyendo una manera para que el estudiante practique. La simulación brinda una oportunidad única para que se practiquen habilidades clínicas en un entorno de bajo riesgo antes de la experiencia en el quirófano [23].

#### **2.1.4 Microsoft Dynamics 365 Guides**

Hoy en día las empresas necesitan realizar una formación constante y exhaustiva del personal sin experiencia. Ya que es un proceso fundamental para una incorporación eficaz del nuevo personal de una empresa. Sin embargo, normalmente alcanzar un buen nivel de formación representa una inversión costosa en términos de tiempo y dinero para la empresa y los empleados. Microsoft *Guides* ofrece una nueva e innovadora forma de

entrenar de manera eficaz, lo cual permite que toda la inversión se realice por adelantado en el desarrollo del material. Y los empleados que requieran la formación o entrenamiento, pueden realizarla por su cuenta sin necesidad de que estén presentes los formadores. Sin problemas de programación ni gastos de viaje. La instrucción puede ser adaptada para las necesidades de la formación que deba ser impartida. Ya que la documentación y fuentes de información están disponibles como enlaces externos en la guía de capacitación [24].

Uno de los mejores ejemplos es el caso de *Toyota Motor Corporation* [25], ya que impulsan muchos cambios y adopta el proceso de mejora continua “*kaizen*”. Todo con el propósito de dar a sus clientes un servicio de mayor valor y calidad. En la última década, los vehículos se han vuelto significativamente más sofisticados. Y la empresa quería ayudar a los técnicos a ser lo más eficientes y precisos posibles en cuanto a los servicios y reparaciones cada vez más complicados. Por lo tanto, Toyota tomó la decisión de adoptar Microsoft HoloLens 2 y desarrollo cuatro diferentes entornos de realidad mixta utilizando “*Azure Object Anchors*”, “*Azure Remote Rendering*”, “*Dynamics 365 Remote Assist*” y “*Dynamics 365 Guides*”. A través de este nuevo enfoque la empresa mejora en aspectos como reducción de tiempos, mejora la calidad de sus servicios y brinda capacitación innovadora para técnicos en formación. Además, fomenta la colaboración de sus empleados cumpliendo con su filosofía de mejora continua.

La empresa utiliza *Dynamics 365 Guides* y los HoloLens para brindar capacitación e instrucciones en el trabajo a los técnicos de servicio a través de instrucciones holográficas. Por ejemplo, *Microsoft Guides* permite a los técnicos ver instrucciones paso a paso sobre un proceso o tarea determinada como instalar una pieza, desmantelar una pieza, ajustar maquinaria, etc. Los técnicos pueden consultar una variedad de guías con sus auriculares y confirmar los procedimientos en la práctica [25].

## **2.2 Valor de negocio en RA**

Actualmente la realidad aumentada ha avanzado lo suficiente para ser ampliamente implementada en distintos dispositivos tecnológicos de uso común como los teléfonos inteligentes. La facilidad con la que las aplicaciones de RA se difunden permiten que pueda volverse parte de los propios productos de las empresas interesadas en implementar esta tecnología. Con la RA también es posible mejorar el rendimiento en distintas áreas en la industria como el desarrollo y diseño de productos, la fabricación, monitorización, comercialización, etc. [26]

Aun cuando la realidad aumentada ya ha sido implementada en distintas áreas de actividad humana, las empresas y organizaciones pasan por alto una de sus principales características. La realidad aumentada puede ser implementada dentro del producto como una capacidad o característica adicional. De manera que se pueden diseñar productos con capacidades enfocadas en la realidad aumentada. Lo cual permite aumentar la cantidad de información y características que pueden proporcionar los productos [26].

Otra de las áreas en la que la realidad aumentada ha causado un gran impacto en una empresa es en recursos humanos, capacitación o entrenamiento. En [27], se presenta un caso en el cual la empresa Wal-Mart agregó a su programa de entrenamiento el uso de realidad virtual y aumentada. En su programa “evitar, denegar, defender” se describen las acciones a tomar, como escapar, protegerse o prepararse para luchar en respuesta a un tirador activo. El 3 de agosto de 2019, en una de las tiendas de Wal-Mart en el Paso, hubo un tiroteo que acabó con la vida de 22 personas. Después de este evento, los empleados de Wal-Mart que había recibido el entrenamiento en RV compartieron comentarios positivos del mismo. Con lo que podemos inferir que los empleados valoraron más el entrenamiento.

En el caso del desarrollo de productos la realidad aumentada muestra su mayor potencial durante el diseño. Durante años los ingenieros y personal encargado del desarrollo de productos de una empresa ha utilizado herramientas de diseño asistido por computadora (CAD, *Computer-aided design*). Creando modelos 3D, sin embargo, la interacción siempre fue a través de ventanas 2D de sus dispositivos lo cual no es intuitivo y dificulta la conceptualización de diseños. Por medio de la realidad aumentada es posible crear modelos tridimensionales y visualizarlos en el espacio físico real, permitiendo una interacción mucho más profunda e intuitiva [26].

La cantidad de áreas en las cuales se puede adoptar la tecnología de realidad aumentada únicamente va en aumento. Esta tecnología aumenta las capacidades y sentidos de los operadores humanos. Además, les proporciona mucha más información directamente a su campo de visión, conecta usuarios y expertos para aumentar la colaboración sin necesidad de ocupar las manos en dispositivos usables (*wearable*).

Algunas de las áreas del ámbito empresarial en las cuales la realidad aumentada se ha aplicado han obtenido resultados beneficiosos. Sin embargo, aún podrían generarse aún más beneficios en áreas como la industria manufacturera, logística, mercadotecnia y

ventas, la industria de la salud, servicios postventa, y recursos humanos. En especial la industria de la salud ya que la realidad aumentada entre las muchas ventajas que aporta la disponibilidad de la información es de gran importancia pues reduce la cantidad de tiempo que el personal médico necesita para obtener información de sus pacientes [26].

### **2.2.1 La estrategia de la Realidad aumentada**

El impacto que la tecnología tiene en las empresas en la actualidad es absoluto. Implementar nuevas tecnologías es base para la competitividad en las empresas. La realidad aumentada es una de estas tecnologías de gran impacto. La manera en que se hacen las cosas actualmente en las industrias y empresas cambia según la tecnología. La realidad aumentada proporciona el medio ideal para la transmisión, presentación y organización de la información de manera rápida y fácil. Además, algunos de los dispositivos “*headwear*” liberan las manos del usuario aumentando la cantidad de tareas que puede realizar. Estos dispositivos poseen una poderosa interfaz la cual muestra información y datos directamente en el campo de visión del usuario. Proporciona grandes beneficios como reducir costos de capacitación, servicio, montaje, diseño y otras fases de la cadena de valor [26].

Las aplicaciones de realidad aumentada enfocadas a la industria van en aumento. Cada día se hacen grandes inversiones en tecnologías de realidad aumentada en las industrias. Con la finalidad de reducir costos, aumentar la productividad, mejorar la eficiencia de una tarea o actividad. Todos estos beneficios se traducen como ahorros significativos. Algunas empresas optan por invertir en la creación de nuevos dispositivos de realidad aumentada. Otros utilizan dispositivos del mercado y solo se dedican al desarrollo de software para su empresa. Y finalmente hay empresas que no les interesan las tecnologías de la información y que solo quieren dedicarse a su negocio y utilizan las tecnologías únicamente como una característica de competitividad. Ellos adquieren dispositivos y software de realidad aumentada proporcionada por terceros [28].

### **2.2.2 Despliegue de realidad aumentada**

Los dispositivos que pueden utilizar aplicaciones de realidad aumentada ya se encuentran ampliamente difundidos en la sociedad. Y sucede lo mismo en la cadena de valor de las empresas y la industria. Poco a poco su uso se ve incrementado de manera continua y exponencial. Las capacidades y versatilidad que ofrece la tecnología de realidad aumentada permiten su introducción e implementación en una amplia variedad de áreas.

Desde las áreas administradas como recursos humanos, finanzas, etc. hasta áreas tecnológicas y de operación [26].

En [29], se menciona que la forma más efectiva para la implementación exitosa de realidad aumentada en una compañía es a través de una hoja de ruta de implementación. En esta hoja de ruta se establecerá como la organización comenzará a capturar los beneficios mientras se desarrollan las capacidades necesarias para expandir su uso. Luego cuando se haya determinado la secuencia y el ritmo de adopción se deben considerar los desafíos técnicos como las habilidades organizativas involucradas, que varían de un contexto a otro. Algunos de los puntos específicos que cada organización debería considerar son [26]:

1. Capacidades de desarrollo: La experiencia es un factor de gran importancia en el desarrollo de aplicaciones o plataformas en realidad aumentada. Es necesario tener una imaginación capaz de imaginar escenarios, situaciones y usos de la realidad aumentada para satisfacer una necesidad o resolver un problema.
2. Contenido digital: Para crear experiencias de realidad aumentada interactivas y amigable con el usuario es necesario crear buenas interfaces. Por lo tanto, el contenido digital es de suma importancia pues esto representará el contenido que será posicionado en la vista del mundo real.
3. Reconocimiento del entorno físico: Para que haya una superposición del contenido digital con el contenido real las aplicaciones de realidad aumentada deben reconocer el espacio físico. Para ello el dispositivo debe poseer en *hardware* o *software* algo que le permite mapear el área vista, localización por GPS, marcadores, etc.
4. *Hardware* para la realidad aumentada: Cualquier teléfono “*Smartphone*” actualmente es capaz de ejecutar aplicaciones de realidad aumentada. Sin embargo, para experiencia más sofisticadas muchas empresas optan por dispositivos “*tablet*” ya que son más potentes, tienen una pantalla más amplia y tienen mejores gráficos. También, se han desarrollado dispositivos enfocados principalmente a la realidad aumentada como los “*headsets*”.
5. Modelo de desarrollo de software o modelo de publicación de contenido: Una de las maneras en las que una organización desarrolle su propia aplicación es a través de una aplicación independiente que pueden ser descargada, instalada y utilizada. El problema es que con cualquier cambio en la experiencia de realidad aumentada

los desarrolladores deberán reescribir la aplicación. Una alternativa es la utilización de software comercial para crear contenido de realidad aumentada. Una opción óptima cuando la aplicación requiere estar cambiando constantemente ya que el contenido es alojado en la nube y la aplicación puede actualizarse mientras tenga una conexión a internet.

### 2.3 Mixed Reality Toolkit (MRTK)

MRTK-Unity es un proyecto impulsado por Microsoft que proporciona un conjunto de componentes y características para acelerar el desarrollo de aplicaciones MR multiplataforma en Unity, en la figura 2.6 se muestra el logo de MRTK. Algunas de las características que posee MRTK son [30]:

- Proporciona los componentes básicos y fundamentales para el desarrollo de aplicaciones *Unity* en *HoloLens*, *Windows Mixed Reality* y *OpenVR*.
- Permite la creación rápida de prototipos a través de la simulación en el editor de Unity lo cual permite ver los cambios de inmediato.
- Diseñado para poder agregar nuevos componentes, además de brindar a los desarrolladores la capacidad de intercambiar componentes principales.
- Admite una amplia gama de plataformas entre las cuales se incluye:
  - *HoloLens* de Microsoft
  - *HoloLens 2* de Microsoft
  - *Windows Mixed Reality Headsets*
  - *OpenVR headsets* (HTC Vive / Oculus Rift)



Figura 2.5 Logo del framework Mixed Reality ToolKit.

Los dispositivos de realidad aumentada como los HoloLens permiten la creación de hologramas, los cuales son objetos hechos de luz y sonido que aparecen en el mundo que nos rodea y los muestra como si fueran reales. A través del dispositivo HoloLens, los hologramas responden a una serie de “gestos” como la mirada, comandos de voz, movimientos de mano, etc. Además, pueden interactuar con las superficies del mundo

real alrededor del usuario. Esto es gracias al sistema de mapeo de los HoloLens el cual conserva un plano tridimensional del mundo físico cercano [30].

MRTK no está diseñada para una sola audiencia en particular y uniforme: fue concebido para admitir casos de uso que van desde *hackatones* (encuentro de programadores) por primera vez hasta personas que actualmente están diseñando y desarrollando experiencias complejas y compartidas para una empresa. Es posible que se hayan escrito algunos códigos y API que se han optimizado para uno más que para el otro (es decir, algunas partes del MRTK parecen más optimizadas para la configuración de "un clic"), pero es importante tener en cuenta que algunas de ellas son simplemente por razones históricas y de recursos. Conforme el marco de trabajo MRTK evoluciona, las características que se crean deben diseñarse a escala para admitir nuevos casos de uso. Es importante tener en cuenta que el MRTK tiene requisitos para un correcto escalado a través de las experiencias de realidad aumentada y realidad virtual. Debe ser fácil construir aplicaciones que retrocedan en comportamiento cuando se implementan en un HoloLens 2 O un HoloLens 1, y debe ser fácil de construir una aplicación dirigida a *OpenVR* y *WVR* (entre otras plataformas). Algunas veces el equipo desarrollador del marco de trabajo de Microsoft puede enfocar una iteración particular en un sistema o plataforma en particular. Sin embargo, el objetivo a largo plazo es crear una amplia gama de apoyo para donde sea que las personas estén construyendo experiencias de realidad mixta [30].

## **2.4 Rondas Médicas**

En todos los hospitales y clínicas se realizan diariamente las rondas médicas, rondas de paciente, rondas multidisciplinarias o rondas interdisciplinarias. Son practicadas principalmente por médicos, enfermeras (os) y asistentes o estudiantes médicos. El propósito de las rondas medicas es evaluar el estado de salud de los pacientes, conocer su evolución diaria y hacer las indicaciones necesarias para su recuperación y rehabilitación. Involucran diversas disciplinas medicas que se unen para discutir la condición de los pacientes y determinar la atención adecuada [7].

Las rondas médicas normalmente son utilizadas como una herramienta educativa. Mantiene al personal médico enfocado en la misma información del paciente y su evolución. El personal que realiza las rondas debe por lo menos leer acerca de los

pacientes a los cuales visitara. De ser posible conocer su historial médico, pruebas recientes, hojas de monitoreo, registro de signos vitales, etc. En [31] se menciona que las rondas medicas se han utilizado como un medio para reunir a los miembros del equipo en unidades de atención al paciente y mejorar la colaboración con respeto al plan de atención.

Algunos autores [7] [8] [31] han determinado que las rondas medicas deben ser asistidas para ser realmente efectivas. En muchos de los casos las rondas diarias son realizados únicamente por un personal médico. Sin embargo, en algunas determinadas ocasiones es necesario que la ronda sea realizada por más de un personal médico e incluso por un equipo compuesto por integrantes de distintas especialidades médicas. En [7] los autores descubrieron que integrando a un farmacéutico durante las rondas diarias los errores de medicación fueron reducidos hasta en un 51%. Comentan que la reducción de errores se deba probablemente a que el farmacéutico obtiene una imagen clínica más completa del paciente después de participar en rondas médicas.

Algunos autores proponen las siguientes sugerencias para realizar rondas médicas de manera exitosa y lo implementan durante la educación del personal médico: Primero, identificar la información que se desconoce del paciente. El conocimiento de determinados pacientes varia de un personal médico a otro. Por lo tanto, es importante que se adquiera la información faltante como el historial médico, pruebas médicas, etc. Segundo, aprender rápidamente la información necesaria. En un hospital hay muchos pacientes y aprender la información de cada paciente a profundidad suele ser una tarea complicada. Sin embargo, suele ser un factor de vital importancia conocer información detallada del paciente. Tercero, proveer retroalimentación. La retroalimentación es una poderosa herramienta para asegurarse que todos estén recibiendo y transmitiendo información correctamente [8].

En [7], se describe la problemática de “errores de medicación” durante las rondas médicas. Este problema lo relacionan con la falta de experiencia del personal médico que realiza las rondas médicas. Y en su propuesta incluyen la participación del personal farmacéutico durante las rondas medicas como medida preventiva y solución al problema. Sin embargo, en México esta solución muchas veces no es practica ya que el personal médico disponible ya se encuentra ocupado. Incluso no hay personal farmacéutico en abundancia para enviar a alguien en cada una de las rondas. Por lo tanto, una solución mucho más eficiente al problema es la implementación tecnológica que auxilie al personal durante las rondas medicas a través de información adicional sobre el paciente de la ronda.

Lo cual permite al personal médico dar una mejor evaluación sobre el medicamento que se debe administrar al paciente.

### III. Desarrollo

A continuación, se presenta el desarrollo del proyecto y del marco de trabajo, sus componentes, instanciación e implementación. Además, se describen cada una de las subsecciones realizadas durante el desarrollo brevemente. En la subsección 3.1 se describe el marco de trabajo y sus componentes de manera detallada. En la subsección 3.2 se describe el subsistema de almacenamiento. En la subsección 3.3 el subsistema de soporte, en la subsección 3.4 el subsistema Desktop y por último en la subsección 3.5 se describe el subsistema de RA.

#### 3.1 Framework

Para el desarrollo de este marco de trabajo fue necesaria la definición de sus componentes principales a través de la revisión de diferentes marcos para RA. La mayoría de estos marcos difieren mucho unos de otros. Los autores de estos marcos los diseñaron tomando en consideración las necesidades del problema que tratan de resolver. Las características y tecnología utilizadas también son diferentes. El diseño del marco de trabajo propuesto está basado en la revisión de varios marcos, arquitecturas y soluciones modeladas, sin embargo, las soluciones propuesta en [32] y [33] fueron la mayor infancia en los componentes y funciones. En [32] los autores describen un marco con tres componentes principales, la plataforma “*Authoring*”, archivo de base de datos y plataforma de aplicación. El primer componente es utilizado por expertos a cargo de datos e información suministrada. El segundo componente es el lugar donde el primer y tercer componente almacenan y recuperan información, el archivo de base de datos. El tercer componente lo utilizan los usuarios del personal que necesitan la información del experto para completar sus tareas. El marco de trabajo propuesto se basa en la estructura de este componente donde hay un sistema para expertos, el componente de soporte. Un sistema para almacenar y recuperar datos, el componente de almacenamiento. Y un sistema que utiliza estos datos, el componente de *Authoring*. Además, agregamos otro componente para estructurar y organizar la información, el sistema de RA. En [33] se describe un marco para diseñar aplicaciones motivacionales de aprendizaje de RA. Proponen un conjunto de componentes necesarios para una aplicación de RA. El componente de RA en nuestro marco de trabajo fue basado siguiendo una estructura donde hay un flujo de componentes que trabajan para darle al usuario una mejor experiencia de realidad aumentada como en [33]. Sin embargo, en lugar de usar cuatro capas, solo nos enfocamos en las siguientes tres; autenticación, selección de perfil y vistas de datos, ya que nuestro componente no

pretende educar o mejorar el aprendizaje. Todos los componentes individuales se muestran en detalle en las siguientes subsecciones.

El marco de trabajo se plantea y desarrolla con la finalidad de construir una base sobre la cual generar instancias del marco con la información y datos que el usuario proporcione. Cada una de estas instancias seguirá la estructura del marco de trabajo, sin embargo, estará enfocada a un área o propósito particular dependiendo de la información proporcionada. En la figura 3.1 se muestra en su totalidad el marco de trabajo de realidad aumentada para la construcción de perfiles de datos específicos necesarios durante las rondas médicas en un hospital o clínica. Está conformado por 4 componentes principales los cuales trabajan juntos interactuando a través del componente de almacenamiento. Las partes que conforman el framework consiste en, componente de almacenamiento, componente de soporte, componente de authoring y componente de realidad aumentada.

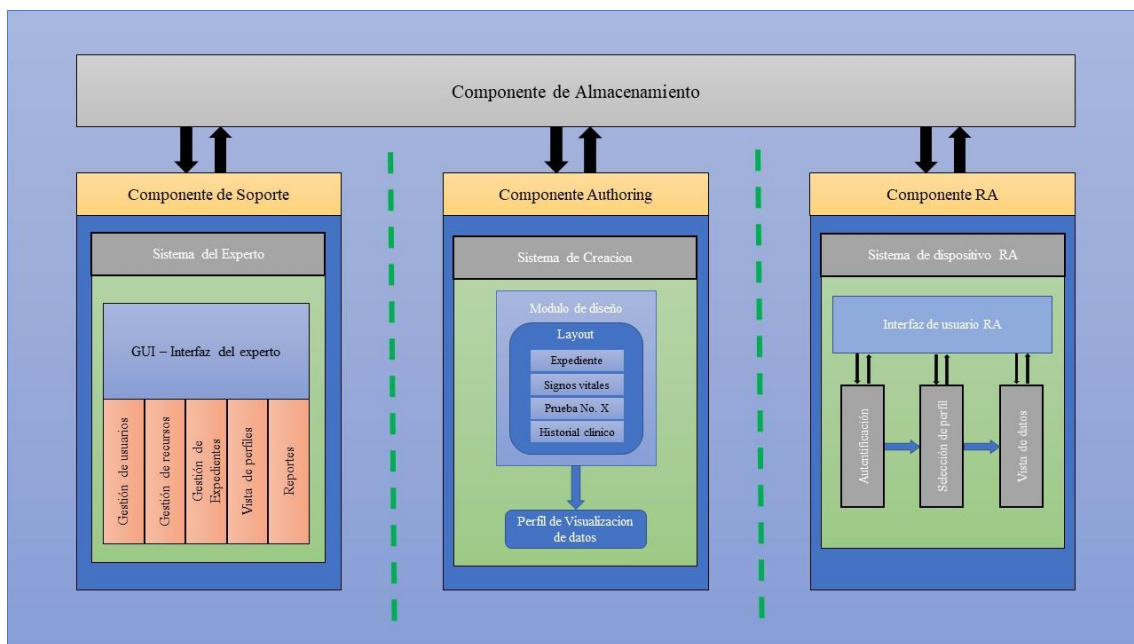


Figura 3.1 Marco de trabajo para la realidad aumentada (elaboración propia).

### 3.1.1 Componente de Almacenamiento

Es el componente base donde se almacenarán todos los datos utilizados y gestionados por los demás componentes del marco de trabajo. Los otros tres componentes (soporte, authoring y RA) se conectarán a dicho sistema para obtener, modificar y agregar datos. Es el único componente del marco de trabajo que no cuenta con otros subsistemas o componentes extras. De esta manera, se puede utilizar la tecnología y herramientas que se considere apropiadas para la información que se vaya a suministrar.

En la figura 3.2 se muestra la representación gráfica que se le da dentro del marco de trabajo.



Figura 3.2 Representación del componente de almacenamiento en el marco de trabajo.

### 3.1.2 Componente de Soporte

Componente que puede ser desarrollado como una aplicación móvil, aplicación de escritorio o aplicación web por la cual un administrador (experto o personal autorizado) podrá realizar algunas de las siguientes tareas; gestión de usuarios, gestión de recursos, gestión de expedientes, vistas de perfiles, reportes, etc.

Tiene como finalidad ser el punto de gestión de los datos de usuarios y pacientes. Se conecta directamente al componente de almacenamiento y permite añadir, modifica o borrar información si se cuenta con las credenciales apropiadas. Además, permite la creación de los usuarios de los componentes *Authoring* y Realidad Aumentada. En la figura 3.3 se muestra el componente de soporte en el marco de trabajo.

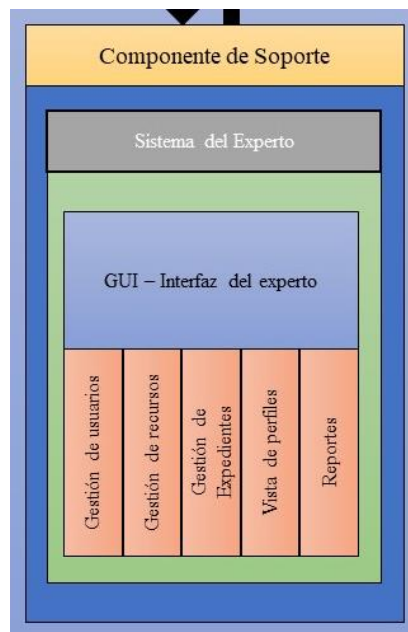
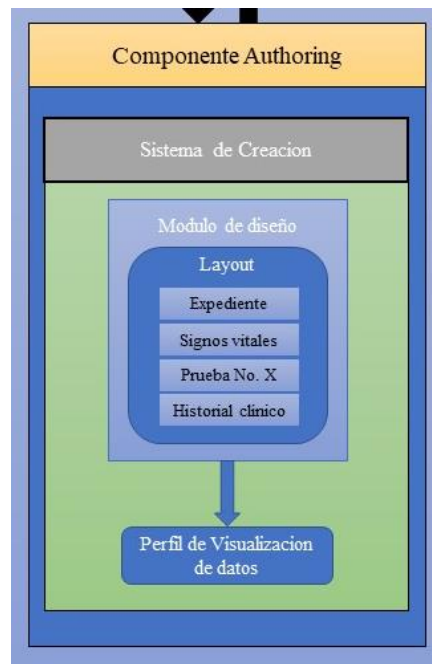


Figura 3.3 Componente de soporte en el marco de trabajo.

### 3.1.3 Componente de Authoring

El componente de “*Authoring*”, se encarga de la parte que crea los perfiles con los datos que ya han sido almacenados en el componente de almacenamiento. Está formado de un subsistema de creación en el cual se diseñan “*layouts*” por parte del personal médico que realiza las rondas. Puede utilizar distintos tipos de datos, expedientes clínicos, signos vitales, pruebas realizadas, etc.

Estos “perfiles” formados por *layouts* son transformados ha contenido gráfico para el subsistema de realidad aumentada. Los usuarios de este subsistema no necesitaran tener experiencia en la creación de software como tal. Solo será necesario que aporten la información que deseen que sea agregada al sistema de realidad aumentada. En la figura 3.4 se muestra la representación gráfica del componente de authoring en el marco de trabajo.

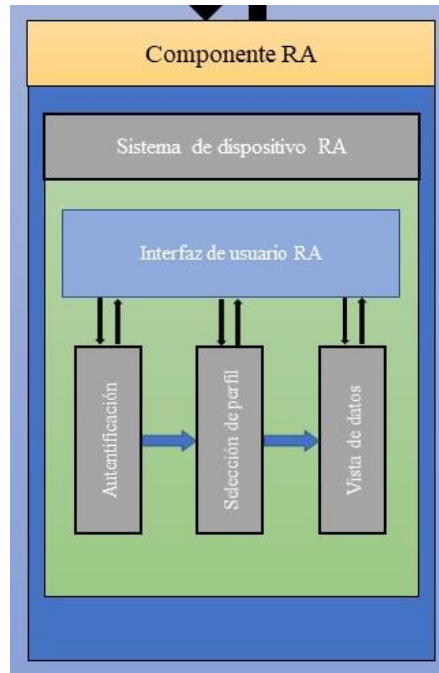


3.4 Componente de Authoring en el marco de trabajo.

### 3.1.4 Componente de Realidad Aumentada

El componente de “*RA*”, es el sistema ejecutándose en el dispositivo de realidad aumentada. Está formado de una interfaz de usuario para el médico o personal médico del hospital o clínica. Por medio de esta interfaz el usuario deberá autenticarse, seleccionar el perfil deseado y se mostrarán los datos del perfil. Los perfiles únicamente

podrán ser creados y modificados a través del sistema de Authoring. En la figura 3.5 se muestra el componente de RA en el marco de trabajo.



*Figura 3.5 Componente de Realidad Aumentada en el marco de trabajo.*

### **3.2 Desarrollo sistema de almacenamiento**

En el sistema de almacenamiento del marco de trabajo no se especifican subsistemas o componentes para su creación o implementación. Es la única parte con esta característica en la cual puedes elegir que herramientas, hardware y software utilizar. Para este proyecto se llevaron a cabo dos implementaciones del sistema de almacenamiento en dos subsecciones diferentes. La subsección 3.2.1 Implementación local y la subsección 3.2.2 Implementación en Amazon aws.

#### **3.2.1 Sistema de almacenamiento – Implementación local**

En el framework el sistema de almacenamiento siempre se dejó a la elección del personal encargado de implementarlo. Desde utilizar servidores locales o en la nube, bases de datos relacionales o no relacionales, etc.

Para la implementación local del sistema de almacenamiento del proyecto, se decidió utilizar una máquina virtual instanciada en un servidor en el LaNTI (Laboratorio

Nacional de Tecnologías de Información). Las especificaciones técnicas de dicha máquina virtual pueden ser consultadas en a) del [\[Anexo A.a\]](#).

Una vez preparado nuestro entorno del sistema de almacenamiento realizamos las configuraciones necesarias en `“/opt/lampp/etc/extra/httpd-xampp.conf”` para permitir el acceso al servidor. Debajo de la línea `“#since XAMPP 1.4.3”` cambiamos `“require”` por `“all granted”`. Ahora debemos configurar la seguridad, introducimos el comando `“sudo /opt/lampp/xampp security”` y se ejecuta un programa en la línea de comando para guiar al usuario en la configuración. Debemos configurar una contraseña para las páginas Xampp, desactivar la accesibilidad de *MySQL* a través de la red, configuramos una contraseña para `“MySQL/phpMyAdmin”`, configuramos la contraseña `“root”` de `“MySQL”` y con eso terminamos las configuraciones de seguridad. Nuestro sistema ahora es accesible a través de la dirección IP de nuestro servidor en el puerto 80 más `“/phpMyAdmin”` para utilizar la herramienta de la base de datos remotamente.

### **3.2.2 Sistema de almacenamiento – Implementación en Amazon aws**

El sistema de almacenamiento también se implementó en un ambiente de nube, específicamente en la nube de *Amazon Web Services* (AWS). Se realizó de esta manera ya que la implementación local del sistema no es accesible fuera de la red local donde se encuentra. Al no poseer permisos de administración de la red no fue posible permitir el acceso a través de una red externa como el Internet. Y, por lo tanto, fue necesario realizar una implementación que permitiera el acceso desde cualquier red conectada a internet que estuviese utilizando el dispositivo HoloLens.

Se eligió Amazon ya que su servicio de *Elastic Compute Cloud* (EC2) nos ofrece justo lo necesario para el sistema de almacenamiento. Amazon es una de las empresas con más tiempo proporcionando servicios de nube, cuentan con una amplia cantidad de regiones y zonas de disponibilidad, y cuentan con un amplio repertorio de instancias para cubrir casi cualquier necesidad. Para este proyecto se decidió empezar con una instancia de capa gratuita ya que los requerimientos iniciales del proyecto no son muy exigentes para el sistema de almacenamiento. Las especificaciones técnicas de la instancia utilizada se describen en el [\[Anexo A.b\]](#).

Para utilizar los servicios de aws es necesario crear una cuenta la cual requiere una verificación y un tiempo de espera de 24 horas antes de poder usarse. Una vez que ya podamos usar nuestra cuenta aws debemos ingresar a la consola de administración de aws

y buscar el servicio “EC2”. Seleccionamos “Instances” en el menú vertical de la parte izquierda y luego en el botón “Launch Instance”. Nos mostrara todas y cada uno de los *templates* disponibles para este caso seleccionamos “Free tier only” y buscamos el *template* de “Ubuntu Server 18.04 LTS” en 64 bit x86. En la siguiente página seleccionamos “Review and Launch” y nuestra instancia comenzara a crearse. Nos permitirá descargar un archivo “.pem” el cual utilizaremos luego para poder acceder a nuestra instancia por lo que debemos guardarlo. Una vez esté lista para ser utilizada (up and running) debemos conectarnos para realizar configuraciones e instalaciones. Para este propósito utilizaremos el software “Putty”, para Windows solo debemos descargar el instalador y ejecutarlo. Una vez hayamos instalado Putty, ejecutamos “PuttyGen” ya que debemos convertir nuestro archivo “.pem” en un archivo requerido por Putty para acceder por *ssh* a nuestra instancia. En PuttyGen nos dirigimos a “Type of key to generate” en la parte baja, en la sección de “Parameters” elegimos “RSA” como se muestra en la figura 3.6. Damos clic en “Load” y buscamos el archivo “.pem” que descargamos anteriormente, de no encontrarlo cambiamos el tipo de archivo por “All files”. Guardamos con el botón “save private key” y le damos un nombre al nuevo archivo.

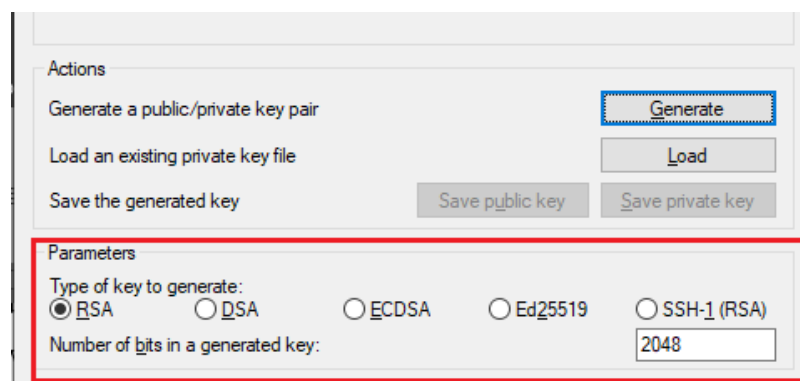


Figura 3.6 PuttyGen parameters.

Para poder conectarnos a nuestra instancia es necesario que busquemos los siguientes datos, nombre publico DNS o IPv4 de la instancia. Para encontrarlos en la sección de instancias al seleccionar la instancia en la parte de abajo aparecerá información de la instancia en una pestaña llamada descripción ahí se encuentra la información que buscamos. Ahora abrimos “Putty” nos dirigimos a “session” en la parte izquierda en la parte derecha en “Hostname” usaremos el siguiente formato “user\_name@public\_dns\_name” donde user\_name dependerá de la máquina que hayamos instalado, en nuestro caso es “ubuntu” ya que instalamos Ubuntu Server y el

“*public\_dns\_name*” es el que obtuvimos anteriormente de la instancia, aunque también podemos usar la dirección ipv4. Es importante saber que esta dirección no es estática y Amazon puede retirarla o cambiarla sin previo aviso. Para evitarlo debemos solicitar una dirección IP estática, pero eso generaría un cobro extra de 0.005 dólares la hora. Volviendo a *Putty* el puerto debe ser 22 y en tipo de conexión SSH. Nos dirigimos al panel izquierdo de categoría y nos vamos a *Connection* → *SSH* → *Auth*. Seleccionamos “*Browse*” y buscamos el archivo “.*ppk*” generado anteriormente. Damos clic en “*Open*” y habremos terminado, ahora podemos conectarnos remotamente a nuestra instancia EC2 por SSH. El siguiente paso consiste en realizar la instalación del software y herramientas del *stack* LAMP exactamente como se realizó en la subsección de implementación local.

### **3.2.3 Sistema de base de datos**

La base de datos en el sistema de almacenamiento la cual alimenta a los demás sistemas del marco de trabajo utiliza el motor de base de datos MariaDB en su versión 10.4.11. Los datos almacenados son datos de prueba basada en una fracción de la información que utilizaría un hospital de su personal y pacientes. El principal propósito de los datos es alimentar a los demás sistemas y definir sus funcionalidades y comportamientos. Ya que dependiendo de los tipos de datos las aplicaciones deberán ser específicamente diseñadas para poder utilizarlos y aprovecharlos. También, para mantener control de los usuarios y sus roles. Dicho control es muy importante ya que dependiendo de los datos que maneje la base de datos, podemos otorgar permisos de acceso, lectura y escritura a los usuarios. Los datos consisten en usuarios y roles (en este caso, personal médico), información general del paciente, examen físico, historial médico, pruebas de pacientes, radiografías y perfiles generados por el sistema de “*authoring*”.

### **3.3 Desarrollo sistema de soporte**

El sistema de soporte se diseñó como una aplicación web para acceder a través de internet utilizando una computadora o dispositivo móvil. Para acelerar el proceso de desarrollo del sistema, se utilizó el marco de trabajo Laravel. Que proporciona las herramientas y estructuras necesarias para formar la base del sistema. El sistema se implementó en una instancia EC2 de AWS al igual que el sistema de almacenamiento. De esta forma, se puede acceder a la aplicación desde cualquier computadora o dispositivo móvil conectado a internet. Es posible implementar los sistemas en una red privada si el usuario desea que solo sea accesible a través de su propia red.

### 3.3.1 Metodología de desarrollo

Para el desarrollo del sistema de soporte se seleccionó el método en cascada o metodología de cascada. La metodología de desarrollo consiste en ordenar rigurosamente las etapas del ciclo de vida del software, de forma que el inicio de cada etapa debe esperar la finalización de etapa anterior [34]. En la figura 3.7 se puede apreciar una representación gráfica de la metodología. Aunque la metodología de cascada es una de las más básicas de todas es al mismo tiempo una de las más utilizadas, permite seguir una serie de fases de manera ordenada y secuencial para el desarrollo del sistema, razón principal por la cual fue elegida para el desarrollo de los sistemas de soporte, *authoring* y RA.

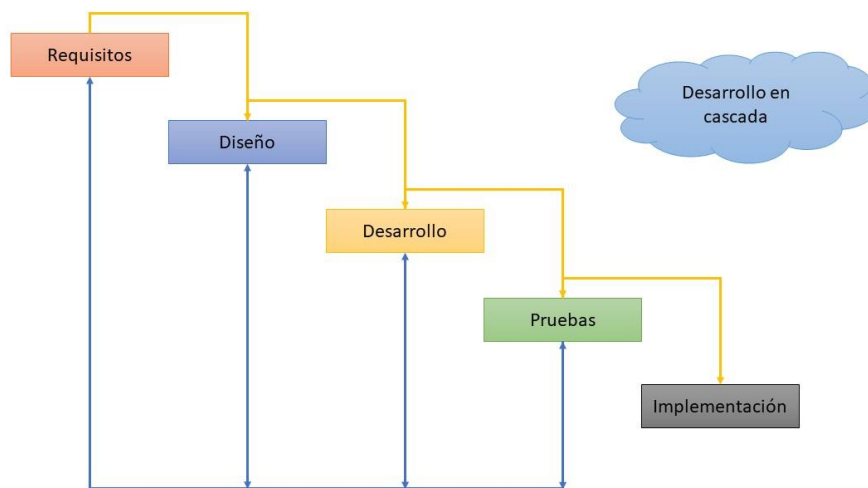


Figura 3.7 Representación gráfica de la metodología en cascada.

A continuación, se describen cada una de las fases de la metodología cascada y el avance del desarrollo del sistema.

- **Requisitos/Análisis:**

Para la primera fase de la metodología se generó un par de artefactos como base para el desarrollo del proyecto. Las características principales del prototipo fueron detenidamente analizadas a través de un documento de especificación de requisitos. También se generaron los casos de uso detallados del sistema. Con estos dos documentos se empezó a generar la base del prototipo del sistema.

Para el desarrollo se propuso el uso del marco de trabajo Laravel como base del sistema. Al trabajar con Laravel se trabaja con PHP, y se eligió LAMP como entorno de desarrollo y pruebas.

- Documento de especificación de requisitos. Como en casi cualquier metodología la primera etapa se enfoca en la búsqueda de los expertos del dominio y sus necesidades. El documento de especificación de requisito permite establecer las bases entre programadores y expertos del dominio, en lo que al proyecto de software se refiere. También, ayuda a los desarrolladores ajenos al proyecto a entender exactamente las necesidades y requerimientos del experto del dominio. Se desea identificar el prototipo deseado. El documento de especificación de requisitos completo puede ser consultado en a) del [\[Anexo B.a\]](#).

El documento de especificación de requisitos permite establecer bases entre los participantes del desarrollo (analistas, expertos de dominio, desarrolladores, diseñadores, etc.) en cuanto a las características del proyecto de software. Es el punto de información clave en el cual personas ajenas al proyecto pueden entender exactamente las necesidades y requerimientos del experto del dominio.

Los objetivos identificados por parte del experto consisten en:

- Desarrollar el prototipo de la aplicación para la gestión de usuarios de la aplicación, pacientes y su información médica.
  - La aplicación podrá ser accedida desde cualquier dispositivo con un navegador web conectado a internet.
  - Utilizar el framework Laravel de PHP.
- Documento casos de uso. Como segundo artefacto de la fase de requisitos/análisis se realizó la creación de un documento de los casos de uso basado en el documento de especificación de requisitos. Como resultado se redactaron los casos de uso detallados para el sistema de soporte. Los casos de uso detallados pueden ser consultados en a) del [\[Anexo C.a\]](#).

El principal propósito para generar el documento de casos de uso es encontrar los requisitos funcionales del sistema. De manera que pueda trazarse las características y funcionalidades específicas del sistema. Además, muestra los escenarios de éxito que describen el comportamiento de los actores con el sistema y del sistema a los actores.

Como ejemplo, en la tabla 3.1, se presenta el caso de uso C2 “Gestión de los datos de perfil de usuario”. El caso de uso muestra paso a paso el proceso de éxito en el cual un usuario se ha autenticado en el sistema y se muestra su información de perfil. En el caso de uso se muestra información importante como: un nombre y una descripción, también enlista los actores que están relacionados con él. Muestra el escenario principal de éxito (flujo principal de eventos) el cual consiste en cada una de las acciones realizadas por el actor y su correspondiente reacción en el sistema. Por último, se encuentran los “flujos alternos” los cuales representan acciones que no siguen el flujo del sistema y que pueden culminar en un caso de fracaso.

Tabla 3.1 Gestión de los datos de perfil del usuario.

<b>Caso de uso C2: Dashboard de usuario.</b> Los usuarios dentro del sistema tendrán la posibilidad de editar la información en sus perfiles.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet. El usuario debe estar autenticado en el sistema.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón de “ <i>Profile</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de gestión del perfil en la cual se mostrará un formulario editable en la cual el usuario podrá cambiar sus datos de perfil.
3: El usuario cambia los datos de su perfil y da clic en Actualizar. FA1	4: Si los datos ingresados son correctos el sistema lo redirigirá al <i>Dashboard</i> de usuario y mostrará un mensaje de éxito.
<b>Extensivos (Flujos alternativos)</b>	
FA1 – El usuario ingresa datos incorrectos como símbolos y números en el nombre. El sistema mostrara un mensaje de error en el formulario que indique el error cometido.	

- **Diseño:**

En la segunda fase de la metodología en cascada se tomaron como entrada los documentos generados durante la fase de análisis. Como resultado se generaron la base de datos y las pantallas o plantillas del sistema de soporte. La base de datos describe la interacción entre los diversos componentes de todos los sistemas del marco de trabajo y no solo los del sistema de soporte.

- Base de datos. La base de datos que se definió dentro del sistema de almacenamiento y que podrá ser accedida por los sistemas de soporte, authoring y RA es una base de datos relacional. Dicha base de datos almacena y proporciona acceso a puntos de datos relacionados entre sí. La base de datos relacional nos ofrece una manera intuitiva y directa de representar los datos en tablas. El modelo entidad relación que representa la base de datos general puede ser consultada en el [\[Anexo D\]](#).
- Pantallas del sistema de soporte (interfaz de usuario). Se diseñaron las pantallas de interfaz de la funcionalidad disponible para los distintos usuarios en el sistema. Las pantallas fueron diseñadas para el sistema de soporte el cual consiste en una aplicación web Laravel. Para la creación de las pantallas se tomaron como base los requisitos funcionales derivados de los casos de uso. El principal propósito del sistema de soporte es ser un punto central para realizar todas las gestiones de usuarios, pacientes, perfiles y su información. Por lo tanto, el principal enfoque de las pantallas consiste en la creación y edición de usuarios y pacientes.

A continuación, en la figura 3.8 se muestra una de las pantallas del sistema. En esta pantalla se muestra la interfaz del perfil de usuario la cual se muestra una vez que el usuario ha accedido a través de la pantalla de “login”. Y le permite al usuario modificar su información de perfil como nombre, usuario, email y contraseña.

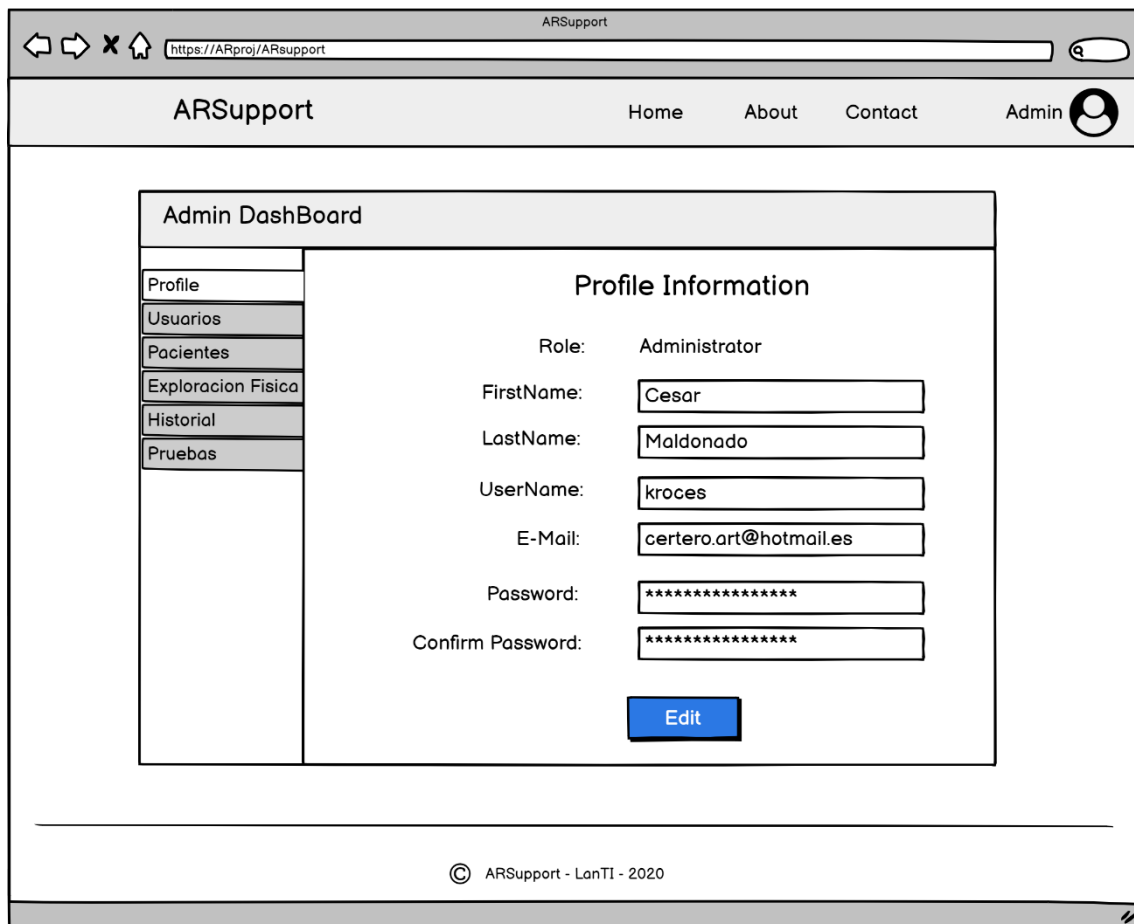


Figura 3.8. Pantalla del Dashboard de usuario.

Todas las pantallas que fueron diseñadas para el sistema de soporte y en las cuales se basó la aplicación web de soporte se encuentran al final de este documento en el [\[Anexo E\]](#).

- **Desarrollo:**

Durante esta fase de cascada se tomaron los artefactos generados en la fase de diseño como base para comenzar el desarrollo. A continuación, se describirán cada una de las subsecciones de esta fase. Para el desarrollo se utilizó el *framework* Laravel con la finalidad de generar una aplicación web escrita en PHP. La plataforma para el desarrollo consiste en una computadora local con 8 GB en RAM, 200 GB disco duro, procesador Intel i5-6500 y con sistema operativo Windows 10.

Las herramientas tecnológicas utilizadas para el desarrollo del sistema consisten en *Visual Studio Code*, *Xampp*, y *Mozilla Firefox*. Se usó la versión de *Visual Studio*

Code 1.46.1 con las extensiones *Laravel Blade Snippets* y *PHPStorm Formatter*. El *software* fue utilizado para poder montar la aplicación web durante las pruebas en el desarrollo. Al proporcionar un ambiente de servidor Apache, con base de datos MariaDB y PHP. Para Mozilla Firefox, uno de los navegadores favoritos a nivel mundial se utilizó la versión 80.0.1 para 64 bits.

- **Framework Laravel**

El framework Laravel es de código abierto diseñado para el desarrollo de aplicaciones y servicios web con PHP. En su página web es descrito como “marco de trabajo para aplicaciones web con una sintaxis elegante y expresiva” [35].

El principal objetivo de Laravel es ser un framework que permita el uso de una sintaxis refinada y expresiva para crear código de forma sencilla, evitando el “código espagueti” y permitiendo multitud de funcionalidades. La mayor parte de su estructura está formada por dependencias, especialmente *Symfony*, lo que implica que el desarrollo de Laravel dependa también del desarrollo de sus dependencias [35].

Para la aplicación web de soporte solo se utilizaron las características base del framework como son: sistema de ruteo, motor de plantillas, motor de controladores, clases añadidas para un mejor manejo de datos como lo son formatos de imágenes. Se comenzó con la versión 6 de Laravel al inicio del desarrollo, pero se migró a la versión 7 por tener más características y la versión 7.2.5 de PHP. En la figura 3.9 se muestra el logo de Laravel.



Figura 3.9. Logo del Framework Laravel ([laravel.com](http://laravel.com)).

- **Configuración de la plataforma de desarrollo**

Antes de comenzar con el desarrollo de la aplicación fue necesario una serie de configuraciones en algunas de las herramientas utilizadas. Estas

configuraciones fueron realizadas principalmente en el software *Xampp* y *Visual Studio Code*.

Una vez que se haya realizado la instalación de las herramientas de software *Xampp* y *Visual Studio Code*, se configuraran de la siguiente manera. En *Xampp* (Windows 10) se deben activar los módulos de Apache modificando el archivo “httpd.conf”. Desde el panel de control de *Xampp* en el renglón de Apache presionamos el botón “*Config*”. En la lista de opciones que aparece seleccionamos Apache (httpd.conf), lo cual abrirá el archivo de configuraciones que debemos modificar. Al inicio de este archivo habrá muchos “*LoadModule*”, algunos se encuentran comentados con #, debemos des comentar los módulos *deflate*, *expires* y el módulo *rewrite*.

A continuación, debemos configurar PHP, modificando el archivo php.ini. Desde el panel de control en el renglón de Apache presionamos nuevamente el botón “*Config*” y seleccionamos PHP (php.ini). Primero cambiaremos el tamaño máximo de subida que por defecto es de 2MB por uno mejor para un entorno de desarrollo, al menos 256MB. Cambiamos la línea “*post\_max\_size=8M*” por 256M y “*upload\_max\_filesize=2M*” por 256M. Y des comentaremos (Remove “;”) las siguientes extensiones, “*php\_mysqli.dll*”, “*php\_pdo\_mysql.dll*”, “*php\_soap.dll*”, “*php\_tidy.dll*” y “*php\_xsl.dll*”. Por último, añadimos nuestra zona horaria, “*date.timezone = ‘America/Ojinaga’*”.

Como último paso debemos configurar el usuario *root* de MySQL. En el panel de control y presionamos el botón “*shell*”, el cual, abrirá una consola de comandos donde se debe ingresar lo siguiente: “*mysqladmin – user=root password ‘nueva contraseña’*”.

Las configuraciones para la herramienta que usaremos para editar el código de la aplicación son únicamente la instalación de extensiones que nos facilitaran la edición del código. En el panel lateral izquierdo se encuentran una serie de iconos en vertical, seleccionamos el icono de extensiones. Se nos mostraran las extensiones ya instaladas y las recomendadas, en la barra de búsqueda de la parte superior pondremos “*Laravel Blade Snippets*”, y seleccionamos el botón instalar. Después, buscaremos la extensión “*PHPStorm Formatter*” y también la instalaremos.

Estas herramientas nos ayudarán a darle un formato más rápido a algunas de las “pantallas” (*blades*) del framework y nos permitirán tener un código más limpio y ordenado.

- **Desarrollo de la aplicación (Módulos)**

Una vez que el entorno de desarrollo ha sido configurado correctamente como se mostró en la sección anterior, se comenzara con el desarrollo de la aplicación en *Visual Studio Code*.

Este prototipo de aplicación fue desarrollado pensando en la necesidad de mantener diferentes roles para sus usuarios. Por lo tanto, los módulos necesarios para cumplir con los requisitos de la aplicación requieren ligeros cambios de apariencia y funcionalidad para cada uno de los usuarios. Para cada uno de los roles, la aplicación está compuesta de 3 módulos principales: módulo de perfil, módulos de gestión de usuarios y el módulo de gestión de pacientes.

El marco de trabajo Laravel trabaja con una arquitectura de carpetas avanzada, de modo que promueve la separación de los archivos con un orden correcto y definido, lo cual facilita el guiar a todos los integrantes del equipo de trabajo o desarrollo (si fuera necesario). Las principales características utilizadas para el desarrollo de la aplicación son: un sistema de rutas, un sistema de abstracción de base de datos, una abstracción del sistema de archivos, gestión de sesiones y sistema de autenticación.

Primero se deben realizar algunas configuraciones en el archivo de entorno (.env) el cual es un archivo de librería PHP el cual se encarga del control de la aplicación. La información en este archivo de configuración dependerá del desarrollador o el servidor donde será colocada la aplicación. Los valores que deben ser configurados son nombre de aplicación, llave de la aplicación, URL de la aplicación (localhost para desarrollo), conexión de base de datos (MySQL, SQLite, PostgreSQL, SQL Server), *host* de la base de datos (localhost para desarrollo), puerto de base de datos, nombre de base de datos, usuario y contraseña.

- Autenticación (*Authentication*).

Una de las ventajas de Laravel son sus librerías de rápida configuración. En esencia, las librerías de autenticación de Laravel están compuestas

por “guardias” y “proveedores”. Los guardias definen como se autentican los usuarios para cada solicitud. Por ejemplo, Laravel se envía con un protector de sesión que mantiene el estado mediante el almacenamiento de sesiones y cookies. Los proveedores definen como se recuperan los usuarios de su almacenamiento persistente. Laravel se envía con soporte para recuperar usuarios usando *Eloquent* y el generador de consultas de base de datos. Sin embargo, puede definir proveedores adicionales según sea necesario para su aplicación. Para la aplicación de soporte es necesaria la aplicación de distintos roles a los diferentes tipos de usuarios. Para poder obtener los datos de los usuarios, como email, nombre de usuario y contraseña es necesario crear una base de datos de donde obtenerlos. Los datos de los usuarios ya fueron definidos en la base de datos diseñada anteriormente y son id, role id, nombre y apellidos, nombre de usuario, email, contraseña, fecha de creación y fecha de actualización. Los tipos de usuario serán administrador, doctor, enfermera y estudiante, los cuales tendrán los roles id 1, 2, 3 y 4 respectivamente. Además, tendrán diferentes niveles de acceso y límites en la aplicación. Los administrativos poseerán acceso total y completo sobre la aplicación. No solo serán capaces de gestionar otros usuarios de la aplicación, también podrán desempeñar la gestión de pacientes y su información. Son el único tipo de usuario que podrá crear usuarios de su mismo nivel y rol, es decir, podrán crear nuevos usuarios de nivel administrativo. Los doctores podrán generar otros usuarios un nivel más bajo que su rol (enfermeras y estudiantes). También, podrán crear y eliminar pacientes y modificar sus datos clínicos y personales. Las enfermeras podrán crear únicamente nuevos usuarios de tipo “estudiante”. También poseerán el mismo nivel de acceso a los datos de los pacientes que un doctor. Por último, los estudiantes no podrán crear nuevos usuarios de ningún tipo, tampoco podrán crear o eliminar usuarios nuevos. Únicamente serán capaces de introducir información de los pacientes o actualizarla, pero no podrán eliminarla.

- Middleware.

En Laravel el *middleware* proporciona un mecanismo conveniente para filtrar las solicitudes HTTP que ingresan a su aplicación. Por ejemplo,

Laravel incluye un *middleware* que verifica que el usuario de su aplicación este autenticado. Si el usuario no está autenticado, el *middleware* redirigirá al usuario a la pantalla de inicio de sesión. Sin embargo, si el usuario está autenticado, el *middleware* permitirá que la solicitud continúe en la aplicación. Para esta aplicación se escribirá *middleware* adicional para permitir separar la funcionalidad para los diferentes tipos de usuarios. Ya que cada usuario estará limitado a las tareas de su rol el *middleware* nos permitirá realizar una separación sencilla y rápida. A través de la herramienta de *composer* crearemos los *middlewares* para cada uno de los tipos de usuarios (*AdminMiddleware*, *DoctorMiddleware*, *NursingMiddleware* y *StudentMiddleware*). Una vez creados los *Middleware* modificaremos el archivo de *middleware* global para definir nombres o apodos para utilizar los *middlewares* más adelante.

- Controladores.

Con Laravel en lugar de definir toda la lógica de manejo de solicitudes como cierres en archivos de ruta, preferimos organizar este comportamiento utilizando clases de controlador. Los controladores pueden agrupar la lógica de manejo de solicitudes relacionadas en una sola clase. Y los controladores los encontramos en el directorio *app/Http/Controllers* del proyecto de la aplicación. Para cada uno de los tipos de usuario se crearon 4 controladores que se encargan de manejar toda la funcionalidad de la aplicación. El controlador “Dashboard” es el principal encargado de las funciones y las rutas a las funcionalidades de gestión de perfil de usuario, gestión de usuarios y gestión de pacientes. El controlador de “perfil” se encarga de las funciones para la actualización de los datos personales y de la cuenta del usuario. El controlador de “usuarios” posee las funciones necesarias para realizar la completa gestión y creación de los usuarios de la aplicación dependiendo de su rol. Por último, el controlador de “pacientes” es el controlador con la mayor cantidad de funciones encargadas de la funcionalidad de gestión y creación de pacientes en la aplicación.

- Recursos/vistas.

Las vistas contienen el HTML proporcionado por la aplicación y separan la lógica de su controlador, lógica de aplicación de la lógica de presentación. Y estas vistas están almacenadas en el directorio `resources/views`. Sin embargo, las vistas en Laravel son diferentes de un archivo `“.html”` donde solo se escribe *scripts* html. “Blade” es el motor de plantillas simple pero potente que se proporciona con Laravel. A diferencia de otros motores de plantillas PHP populares, Blade no impide usar código PHP simple en sus vistas. De hecho, todas las vistas de Blade se compilan en código PHP simple y se almacenan en cache hasta que se modifican, lo que significa que Blade agrega esencialmente cero gastos generales a la aplicación. Todos los archivos de *view* usan la extensión de archivo `“.blade.php”`. Dos de los principales beneficios de utilizar Blade son la herencia de plantillas y las secciones. Esta cualidad nos permite seccionar nuestras vistas de tal manera que podemos utilizar una plantilla base y únicamente ir acoplando nuevas plantillas dependiendo de la situación requerida. Las vistas de cada uno de los usuarios de la aplicación representan la funcionalidad pura ya que les muestran a los usuarios las acciones disponibles. Las vistas son para mostrar el Dashboard, editar perfiles, gestionar usuarios, pacientes y su información. Todas y cada una de ellas escritas en formato `“.blade”`. Además, se utilizó la librería de estilos “Bootstrap” para facilitar la creación de vistas con una apariencia agradable para los usuarios.

- Rutas.

Una de las mayores ventajas de Laravel sobre otros frameworks es su sistema de ruteo. Todas las rutas de Laravel se definen en sus archivos de ruta, que se encuentran en el directorio `“routes”`. Estos archivos son cargados automáticamente por el marco. El archivo `“routes/web.php”` define rutas que son para su interfaz web. A estas rutas se les asigna el grupo de middleware web, que proporciona características como el estado de la sesión y la protección CSRF. Las rutas en `“routes/api.php”` no tienen estado y se les asigna el grupo de *middleware* api. Los métodos de enrutamiento permiten registrar rutas que responden a cualquier verbo *HTTP*, `get`, `post`, `put`, `patch`, `delete`, `options`. Para nuestra aplicación web se decidió utilizar grupos ya que estos permiten compartir atributos de ruta,

como middleware, en una gran cantidad de rutas sin necesidad de definir esos atributos en cada ruta individual. Los atributos compartidos se especifican en formato de matriz como primer parámetro del método “*Route::group*”. De esta manera podemos repetir nombres en las rutas sin que haya ningún conflicto ya que cada ruta se encontrara dentro de su grupo determinado administrador, doctor, enfermera y estudiante.

- **Pruebas:**

El desarrollo de la aplicación fue realizado utilizando en el editor de código *Visual Studio Code*. Sin embargo, la aplicación fue probada utilizando el servidor *Xampp* junto con las herramientas descritas en la sección 3.2 Implementación Local/AWS.

Las pruebas realizadas a la aplicación web se enfocan principalmente en la usabilidad, confiabilidad y rendimiento. Las pruebas fueron aplicadas a una muestra de 3 usuarios y consistió en utilizar la aplicación siguiendo una serie de instrucciones. Posteriormente, describieron sus opiniones del manejo de la aplicación y se obtuvieron los siguientes datos:

- Usabilidad: La usabilidad fue medida en tres diferentes métricas (exactitud, tiempo empleado, conformidad del usuario). La exactitud obtenida al aplicar el uso del prototipo a un usuario fue alta ya que el usuario no cometió errores al probar la funcionalidad de la aplicación. La aplicación fue diseñada de manera que se redujeran los errores por parte del usuario. El tiempo empleado por el usuario está dentro del margen establecido para realizar las operaciones definidas. El usuario recuerda los pasos que siguió para realizar las acciones correspondientes de la aplicación. En general, el usuario describió sentirse cómodo al usar la aplicación.
- Confiabilidad: Se realizó una serie de pruebas para medir la confiabilidad de la aplicación. Sin embargo, el prototipo fue diseñado para llevar a cabo una interacción limitada con el usuario (las acciones y usos que el usuario hace en la aplicación está preestablecida). Por lo tanto, cualquier acción que el usuario pueda realizar está establecida desde el diseño de la

aplicación, impidiendo al usuario salir de los parámetros de comportamiento fijados. Por lo tanto, la aplicación cumple con la confiabilidad deseada en cuando a la funcionalidad propia de la aplicación y sus componentes.

- Rendimiento: El rendimiento de la aplicación fue comprobado durante las pruebas de usabilidad. La aplicación durante las pruebas en local demostró que funcionaba bien bajo los recursos establecidos. Sin embargo, para determinar si la configuración de recursos en AWS es la adecuada se repitieron las pruebas en la aplicación alojada en la instancia EC2 de Amazon. Se determinó que con la infraestructura de AWS tanto de la instancia como la de red son más que suficientes para un grupo reducido de usuarios (entre 5 y 8 usuarios al mismo tiempo). Para un futuro incremento de usuarios será necesario un incremento del tipo de instancia acorde a la potencia requerida. AWS nos permite cambiar en cualquier momento el tipo de la instancia por lo cual no hay problemas de escalabilidad en el futuro.

- **Implementación:**

La aplicación fue implementada en su estación definitiva durante las pruebas de rendimiento. Para este fin fue utilizada una instancia EC2 t2.micro (instancia EC2 de 1vcpu, 1 GB de RAM), la potencia no es la ideal para un ambiente de producción, sin embargo, gracias a la escalabilidad proporcionada por AWS la instancia puede crecer en cualquier momento. La aplicación se encuentra en correcto funcionamiento y accesible a través de internet.

### **3.4 Desarrollo sistema Authoring**

El sistema de “authoring” se diseñó al igual que la aplicación de soporte como una aplicación web accedida a través de internet utilizando una computadora o dispositivo móvil. Se desarrolló utilizando el marco de trabajo Laravel. Que proporciona las herramientas y estructuras necesarias para forma la base del sistema. Además, el sistema se implementó en una instancia EC2 de AWS. Es posible implementar los sistemas en una red privada si el usuario desea que solo sea accesible a través de su propia red.

#### **3.4.1 Metodología de desarrollo**

Para el desarrollo de la aplicación de soporte y authoring se seleccionó el método en cascada o metodología de cascada. La metodología de desarrollo consiste en ordenar rigurosamente las etapas del ciclo de vida del software, de forma que el inicio de cada etapa debe esperar la finalización de etapa anterior [34]. En la figura 3.7 se muestra una ilustración de la metodología cascada.

- **Análisis:**

Para la fase de análisis de la metodología en cascada al igual que en el sistema de soporte se generó un par de artefactos para el sistema de authoring como base para el desarrollo. Se determinaron las características principales que el prototipo necesita a través de un documento de especificación de requisitos. Además, se generaron los casos de uso detallados del sistema.

Para el desarrollo se propuso el uso del marco de trabajo Laravel como base del sistema. Laravel utiliza PHP como lenguaje de programación base y se eligió LAMP como entorno de desarrollo y pruebas.

- Documento de especificación de requisitos. El enfoque principal de la fase de análisis es la extracción y análisis de los requisitos del sistema los cuales son obtenidos a través de los expertos del dominio y sus necesidades. A través del documento de especificación de requisito se establecieron las bases conceptuales del sistema de authoring. Lo cual permite transmitir información concerniente al sistema rápidamente entre programadores y expertos del dominio. Además, es de gran ayuda para informar a cualquier persona ajena al sistema y de esta forma entender rápidamente el sistema. El documento de especificación de requisitos completo del sistema de authoring puede ser consultado en b) del [\[Anexo B.b\]](#).

Como se menciona en el desarrollo del sistema de soporte el documento de especificación de requisitos permite establecer bases entre los participantes del desarrollo (analistas, expertos de dominio, desarrolladores, diseñadores, etc.) en cuanto a las características del proyecto de software. Por lo tanto, es importante su creación pues permite establecer los cimientos del futuro sistema.

Los objetivos identificados por parte del experto consisten en:

- Acceso únicamente a personal autorizado a través de un navegador web en cualquier dispositivo conectado a internet. Por medio de usuario y contraseña.
  - Permitir la gestión de perfiles para el sistema de realidad aumentada.
  - Permitir la revisión y copia de perfiles de otros usuarios del sistema de authoring.
- Documento casos de uso. El segundo artefacto generado para la fase de requisitos/análisis fue el documento de casos de uso, basado en el documento de especificación de requisitos. Como resultado se redactaron los casos de uso detallados para el sistema de soporte. Los casos de uso detallados pueden ser consultados en b) del [\[Anexo C.b\]](#).

El documento de casos de uso permite detallar paso a paso los escenarios en que interactúan el usuario y el sistema. De esta manera, se planea una ruta de interacción llamado escenario de éxito el cual dicta los pasos para que el caso sea exitoso. Además, se planean los posibles “flujos alternos” los cuales son una desviación en la ruta de éxito. Un documento de casos de uso bien detallado nos permitirá avanzar en el proceso de desarrollo y específicamente a la fase de diseño.

Como ejemplo, en la tabla 3.2, se presenta el caso de uso C7 “Creación de nuevo perfil”. El caso de uso muestra paso a paso el proceso de éxito en el cual un usuario crea un nuevo perfil. En el caso de uso se muestra información importante, nombre y una descripción, además enlista los actores que están relacionados con él. Muestra el escenario principal de éxito (flujo principal de eventos) el cual consiste en cada una de las acciones realizadas para lograr el objetivo de crear un nuevo perfil en el sistema de authoring. Por último, se encuentran los “flujos alternos” los cuales representan acciones que no siguen el flujo del sistema y que pueden culminar en un caso de fracaso.

*Tabla 3.2 Creación de nuevo perfil.*

**Caso de uso C7: Creación de nuevo perfil.** Los usuarios dentro del sistema tendrán la posibilidad de crear nuevos perfiles para la aplicación de realidad aumentada.

<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “ <i>My Profiles</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página todos sus perfiles en la cual se mostrará una lista con todos sus perfiles. FA1, FA2
3: El usuario da clic en el botón de la parte superior izquierda “Nuevo perfil”.	4: El sistema mostrará la página para la creación de un nuevo perfil.
5: El usuario rellena el campo de nombre del perfil y elige la información que deberá contener el perfil. Da clic en el botón de guardar nuevo perfil.	6: El sistema guardara los datos del nuevo perfil en la base de datos y mostrar un mensaje de perfil guardado exitosamente. FA3
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	
FA2 – El usuario no tiene ningún perfil en base de datos y su muestra una lista vacía.	
FA3 – Error al guardar el perfil en la base de datos, se muestra un mensaje de error.	

- **Diseño:**

En esta segunda fase de la metodología en cascada se tomaron los documentos de análisis de requisitos y casos de uso. Como resultado se generaron las pantallas o plantillas del sistema de authoring. La base de datos fue diseñada a partir de las necesidades del marco de trabajo. Sin embargo, elementos individuales son agregados dependiente de las necesidades del área de dominio al que sea enfocado.

- Base de datos. Definida dentro del sistema de almacenamiento y que podrá ser accedida por los sistemas de soporte, authoring y RA la cual es una base de datos relacional. Almacena y proporciona acceso a los datos necesarios por los sistemas externos a almacenamiento. La base de datos relacional nos ofrece una manera intuitiva y directa de representar los datos en tablas. El modelo entidad relación que representa la base de datos general puede ser consultada en el [\[Anexo D\]](#).

- Pantallas del sistema de authoring (interfaz de usuario). Las pantallas de interfaz de usuario que fueron diseñadas en base a los casos de uso de la fase de análisis. Las pantallas para el sistema de *authoring* consisten en una interfaz web de una aplicación de Laravel. El sistema de authoring solo puede ser accedido a través de la página de “*login*” y por medio de credenciales de usuario previamente definidas en el sistema de soporte. El principal propósito del sistema de soporte es ser un punto central para realizar todas las gestiones de usuarios, pacientes, perfiles y su información. Por otro lado, el sistema de authoring es el encargado de crear y gestionar los perfiles y su información necesaria para el sistema de realidad aumentada.

A continuación, en la figura 3.10 se muestra una de las pantallas del sistema de authoring. En esta pantalla se muestra la interfaz de usuario para crear nuevos perfiles utilizados por el sistema de realidad aumentada. Le permite al usuario agregar la información que desea utilizar en su perfil el cual se encuentra en la parte izquierda y lo arrastra al marco de la parte izquierda.

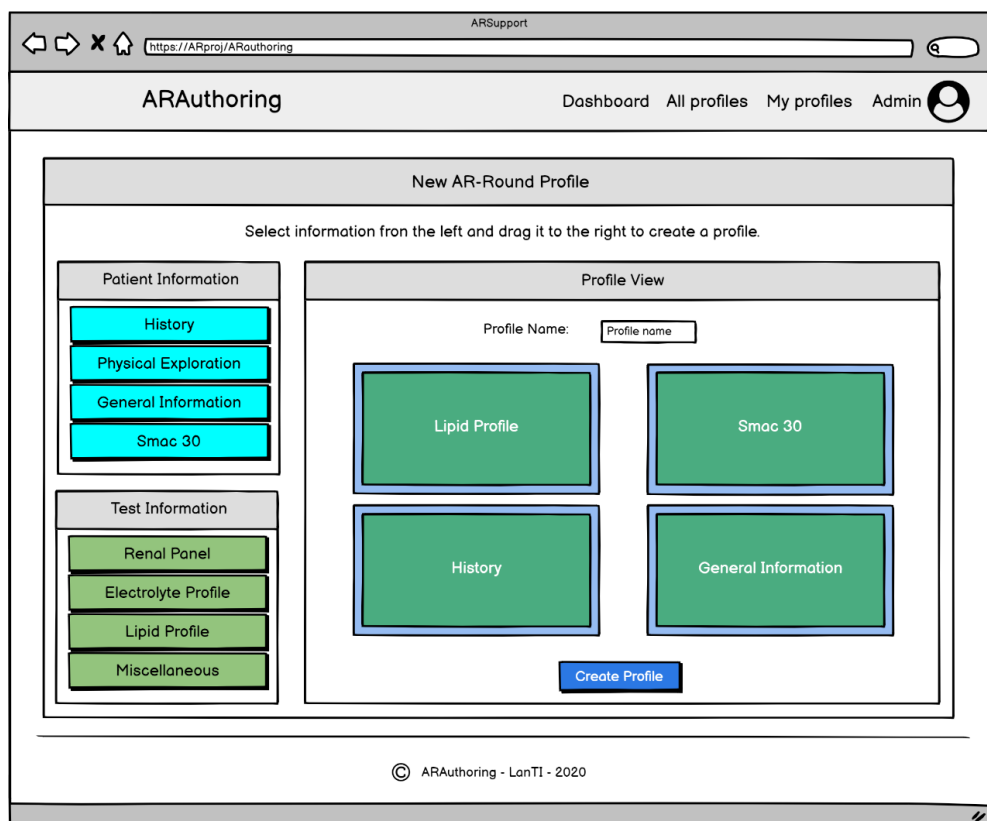


Figura 3.10. Pantalla de nuevo perfil.

Todas las pantallas que fueron diseñadas para el sistema de authoring y en las cuales se basó la aplicación web pueden ser consultadas al final de este documento en el [\[Anexo F\]](#).

- **Desarrollo:**

Las aplicaciones de soporte y authoring fueron pensadas ambas como aplicaciones web utilizando el marco de trabajo Laravel. Al igual con la aplicación de soporte, se utilizaron los artefactos generados durante las fases anteriores para desarrollar la aplicación. Las subsecciones de la fase de desarrollo son las mismas que en la aplicación de soporte. La plataforma para el desarrollo consistió en una computadora local con 8 GB en RAM, 200 GB disco duro, procesador Intel i5-6500 y con sistema operativo Windows 10.

Las herramientas tecnológicas utilizadas fueron *Visual Studio Code*, *Xampp*, y Mozilla Firefox.

- **Framework Laravel y Plataforma de desarrollo.**

Los sistemas de soporte y authoring comparten exactamente el mismo framework y plataforma de desarrollo. Durante las fases de análisis se decidió que ambos sistemas requerían ser aplicaciones web adaptables (“*Responsive*”). Adicionalmente la aplicación de authoring no puede ser utilizada sin un “nombre de usuario” y “contraseña” previamente definidos en la aplicación de soporte.

- **Desarrollo de la aplicación (Módulos).**

Toda la codificación de la aplicación de authoring fue realizada en la misma plataforma de desarrollo que authoring por lo tanto se utilizó el editor *Visual Studio Code* durante todo el desarrollo.

El prototipo de authoring fue desarrollado bajo la necesidad de una aplicación web en la cual distintos usuarios autorizados puedan ver y crear perfiles que después serán utilizados en el sistema de realidad aumentada.

Los usuarios que pueden acceder al sistema son autorizados por medio del sistema de soporte el cual se encarga de la gestión de usuarios, y la información de los pacientes. La aplicación de authoring está compuesta de dos módulos principales: modulo “vista de perfiles”, modulo “gestión de perfiles”.

Como se mencionó anteriormente durante el desarrollo del sistema de soporte en la subsección desarrollo. El marco de trabajo Laravel tiene una arquitectura de carpetas, las cuales nos permiten seccionar el desarrollo por las áreas importantes como la autenticación, middleware, controladores, vistas y rutas.

Inicialmente realizados la configuración de archivo de entorno (.env) encargado del control de la aplicación. La información contenida en “.env” dependerá de las necesidades del ambiente de desarrollo o producción donde será implementado.

- Autenticación (*Authentication*).

La autenticación para la aplicación de authoring será gestionada por medio de la aplicación de soporte. Por lo tanto, no se podrán dar de alta ni modificar información de los usuarios a través de authoring. Dichas tareas deberán ser manejadas a través de soporte. No existe realmente restricciones a nivel de usuario en el uso de la aplicación de authoring y podrá ser utilizada por completo por todos los usuarios (Administradores, Doctores, Enfermeras, Estudiantes). Sin registro de nuevos usuarios o cambios en la información de sus perfiles, únicamente pueden ser autenticados por la aplicación para obtener acceso.

- Middleware.

El marco Laravel nos permite la codificación de un *middleware* para filtrar las solicitudes HTTP que ingresan. De manera que el framework incluye un middleware, un archivo *php* en el cual los desarrolladores pueden agregar diferentes *middlewere* para realizar la autenticación de manera rápida y sencilla. El middleware redirigirá al usuario a la pantalla de inicio de sesión. Sin embargo, si el usuario está autenticado, el middleware permitirá que la solicitud continúe en la aplicación. Para esta aplicación se escribirá middleware adicional para permitir separar la funcionalidad para los diferentes tipos de usuarios. Sin embargo, esta cualidad no altera el funcionamiento o accesibilidad a las funciones de la aplicación. Fue realizado de esta manera previendo que en el futuro se puedan agregar aún más funcionalidades a la aplicación de authoring. Los *middlewares* son

creados utilizando la herramienta *composer* para cada uno de los tipos de usuario (*AdminMiddleware*, *DoctorMiddleware*, *NursingMiddleware* y *StudentMiddleware*). Después de crearlos realizaremos cambios en el archivo php *Middleware* global para definir nombres o apodos para utilizar los *middlewares* más adelante.

- Controladores.

Al igual que la aplicación de soporte los controladores pueden agrupar la lógica de manejo de solicitudes relacionadas en una sola clase. Los controladores de la aplicación de authoring están localizados en el directorio *app/Http/Controllers*. Para cada uno de los tipos de usuario se crearon 4 controladores que se encargan de manejar toda la funcionalidad de la aplicación. El controlador “Dashboard” es el principal encargado de las funciones y las rutas a las principales funcionalidades de la aplicación. En el caso de authoring son los perfiles globales de usuario y los perfiles del usuario. El controlador de “perfil” a diferencia del controlador en soporte se encarga de manejar la gestión de los perfiles para la aplicación de RA. Así como su creación, edición, vista y eliminación. Adicionalmente, cuenta con la función de “copiar” la cual permite al usuario copiar perfiles diseñados por otros usuarios de la aplicación.

- Recursos/Vistas.

Las vistas de la aplicación de authoring, las cuales contienen código html y css para formar la apariencia de las pantallas de la aplicación, se localizan en el directorio *resources/views*. Sin embargo, como se detalla en el segmento de la aplicación de soporte, las vistas en Laravel son diferentes de un archivo “.html” y en su lugar es un archivo “*Blade*” el cual es un motor de plantillas simple pero potente que proporciona Laravel. A diferencia de otros motores de plantillas PHP populares, *Blade* no impide usar código PHP simple en sus vistas. De hecho, todas las vistas de *Blade* se compilan en código PHP simple y se almacenan en cache hasta que se modifican, lo que significa que *Blade* agrega esencialmente cero gastos generales a la aplicación. Todos los archivos de *view* usan la extensión de archivo “.blade.php”. Las vistas de cada uno de los usuarios de la aplicación representan la funcionalidad pura ya que les muestran a los

usuarios las acciones disponibles. Las vistas de la aplicación de authoring mostraran el Dashboard de los usuarios, los perfiles globales en la aplicación (perfiles de todos los usuarios) y los perfiles personales del usuario. Una cualidad importante en las pantallas de los perfiles es el *script* “*json*” en los “*Blade.php*” para hacer que la experiencia del usuario sea lo más cómoda y sencilla posible durante la edición de los perfiles. Se agregaron algunas características que permiten al usuario arrastrar y acomodar objetos dentro de la página utilizando únicamente el puntero del ratón. Además, se utilizó la librería de estilos “Bootstrap” para facilitar la creación de vistas con una apariencia agradable para los usuarios.

- Rutas.

Todas las rutas de la aplicación de authoring están definidas en el archivo “*web.php*” localizado en el directorio “*routes/web.php*”. A estas rutas se les asigna el grupo de middleware web, que proporciona características como el estado de la sesión y la protección CSRF. Los métodos de enrutamiento permiten registrar rutas que responden a cualquier verbo *HTTP*, *get*, *post*, *put*, *patch*, *delete*, *options*. Al igual que en soporte se utilizaron grupos ya que nos permiten compartir atributos de ruta, como middleware, en una gran cantidad de rutas sin necesidad de definir esos atributos en cada ruta individual. Los atributos compartidos se especifican en formato de matriz como primer parámetro del método “*Route::group*”. De esta manera podemos repetir nombres en las rutas sin que haya ningún conflicto ya que cada ruta se encontrará dentro de su grupo determinado administrador, doctor, enfermera y estudiante.

- **Pruebas:**

La aplicación de authoring al igual que la de soporte fue probada utilizando un servidor Xampp junto con las herramientas descritas en la sección 3.2 Implementación Local/AWS.

Las pruebas realizadas a la aplicación web se enfocan principalmente en la usabilidad, confiabilidad y rendimiento. Las pruebas fueron aplicadas a una muestra de 3 usuarios y consistió en utilizar la aplicación siguiendo una serie de

instrucciones. Posteriormente, describieron sus opiniones del manejo de la aplicación y se obtuvieron los siguientes datos:

- Usabilidad: Esta cualidad fue medida en tres diferentes métricas (exactitud, tiempo empleado, conformidad del usuario). La exactitud consiste en las acciones del usuario sin cometer errores dentro de la aplicación. Los usuarios no cometieron errores al probar la funcionalidad de la aplicación. Esto es resultado de las características implementadas en la aplicación desde su fase de diseño. El tiempo que utilizó el usuario para probar las funcionales se encuentra en el margen adecuado en comparación con un usuario experto el cual conoce y ha utilizado la aplicación con anterioridad. Los usuarios recuerdan las acciones que siguieron para realizar las tareas o procesos en la aplicación. En general, el usuario describió sentirse cómodo al usar la aplicación.
- Confiabilidad: Al igual que la aplicación de soporte se realizaron una serie de pruebas para medir la confiabilidad de la aplicación. Sin embargo, ambos prototipos fueron diseñados para llevar a cabo una interacción limitada con el usuario (las acciones y usos que los usuarios realizan en las aplicaciones está preestablecida). Por lo tanto, cualquier acción que el usuario pueda realizar ya fue cuidadosamente establecida desde el diseño de la aplicación, impidiendo al usuario salir de los parámetros de comportamiento fijados. Por lo tanto, determinamos que la aplicación cumple con la confiabilidad deseada en cuando a la funcionalidad propia de la aplicación y sus componentes. De manera que el usuario nunca será capaz de cometer errores de “tipeo”.
- Rendimiento: El rendimiento de la aplicación fue comprobado durante las pruebas de usabilidad. La aplicación durante las pruebas en local demostró que funcionaba bien bajo los recursos establecidos. Sin embargo, para determinar si la configuración de recursos en AWS es la adecuada se repitieron las pruebas en la aplicación alojada en la instancia EC2 de Amazon. Se determinó que con la infraestructura de AWS tanto de la instancia como la de red son más que suficientes para un grupo reducido de usuarios (entre 5 y 8 usuarios al mismo tiempo). Para un futuro incremento de usuarios será necesario un incremento del tipo de instancia

acorde a la potencia requerida. AWS nos permite cambiar en cualquier momento el tipo de la instancia por lo cual no hay problemas de escalabilidad en el futuro. En la aplicación de authoring una buena parte de los recursos necesarios son manejados localmente en el dispositivo del usuario ya que la aplicación utiliza código “*json*” para la edición de los perfiles.

- **Implementación:**

Ambas implementaciones de la aplicación de *authoring* y soporte se realizaron en instancias EC2 t2.micro (instancia EC2 de 1vcpu, 1 GB RAM), la potencia no es ideal para un ambiente de producción con una cantidad moderada de usuarios (para esta instancia a partir de más de 10 usuarios), sin embargo, gracias a la escalabilidad proporcionada por AWS la instancia puede crecer en cualquier momento. La aplicación se encuentra en correcto funcionamiento y accesible a través de internet.

### **3.5 Desarrollo sistema de Realidad Aumentada**

Durante el desarrollo del sistema de realidad aumentada se realizan una serie de tareas y configuraciones las cuales se segmenta en diferentes subsecciones. En la subsección 3.5.1 *MixedRealityToolKit*, en 3.5.2 Configuraciones de *Unity*, en 3.5.3 Prototipos de HoloLens y por último en 3.5.4 los módulos y scripts de la aplicación “*Medical Rounds RA – Round*”.

El desarrollo del sistema de realidad aumentada fue realizado para el dispositivo de realidad aumentada HoloLens de Microsoft. Se utilizó el motor de videojuegos Unity en su versión 2018.4.6f1 para formar los elementos y escenarios virtuales que forman parte de la interfaz y el entorno de la aplicación. Además, se codificaron eventos, comportamientos y el funcionamiento principal en el lenguaje de programación C# utilizando la herramienta Visual Studio 2019. Se utilizó el marco de trabajo “*MixedRealityToolKit*” en su versión “*Foundation-v2.0.0-RC2*” para el manejo del dispositivo HoloLens.

#### **3.5.1 MixedRealityToolKit de Microsoft**

*MixedRealityToolKit* es un marco de trabajo diseñado por Microsoft para manejar distintos niveles de realidad mixta en varios dispositivos. Proporciona una colección de scripts,

interfaces y componentes diseñados para acelerar el desarrollo de aplicación de realidad aumentada para el dispositivo HoloLens en este caso. Su principal objetivo consiste en reducir las barreras de entrada para crear aplicaciones de realidad mixta y contribuir a la comunidad de desarrollo para su crecimiento.

El marco de trabajo está diseñado para la plataforma Unity. Incluye los componentes básicos necesarios para manejar aplicaciones dentro del dispositivo y de esta manera no tener que “reinventar la rueda” cada vez que se cree una aplicación de realidad aumentada. Para el desarrollo de aplicaciones dirigidas a los HoloLens, como mínimo, es necesaria la creación de un objeto cámara especialmente configurada, un cursor con la forma de una pequeña rueda y un administrador de entradas en Unity.

### **3.5.2 Configuraciones de Unity**

Para poder comenzar el desarrollo de aplicaciones de realidad aumentada para los HoloLens en Unity es necesario realizar una serie de configuraciones iniciales. A continuación, se muestran todas las configuraciones en orden cronológico.

Los HoloLens son un dispositivo completamente independiente de otros equipos de cómputo. Posee su propia versión de sistema operativo *Windows 10*, un procesador, tarjeta gráfica, disco duro, tarjeta de red, etc. Las aplicaciones que se ejecutan en los HoloLens pueden ser descargadas de la *Windows Store* o bien pueden ser construidas y compiladas en un equipo diferente para luego ser implementadas en los HoloLens. Esta última manera es la forma en la cual se realiza en este proyecto.

En Unity la configuración para la creación de la cámara y los eventos de entrada de los HoloLens puede ser fácilmente realizado a cabo mediante el *framework MixedRealityToolKit*. Creamos un proyecto nuevo de Unity en la versión “2018.4.6f1” en *templates* utilizamos 3D y presionamos *create* como se muestra en la figura 3.11.

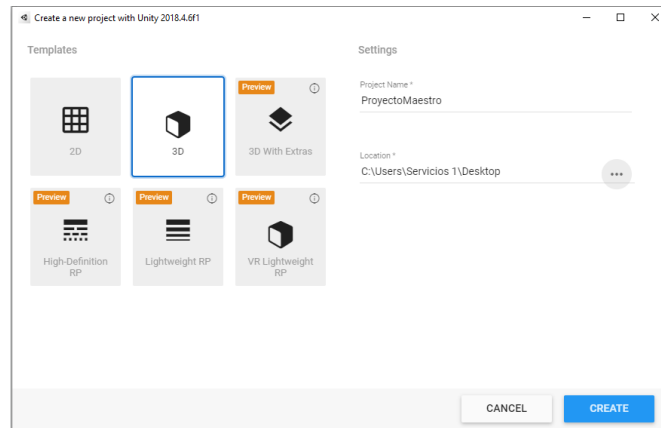


Figura 3.11 Creación de proyecto nuevo en Unity.

Primero, se importa el *framework* a través de la barra de menú *Assets* → *Import Package* → *Custom Package...* Y seleccionamos el paquete *Foundation* en su versión “v2.0.0-RC2”. En la figura 3.12 se muestra el proceso.

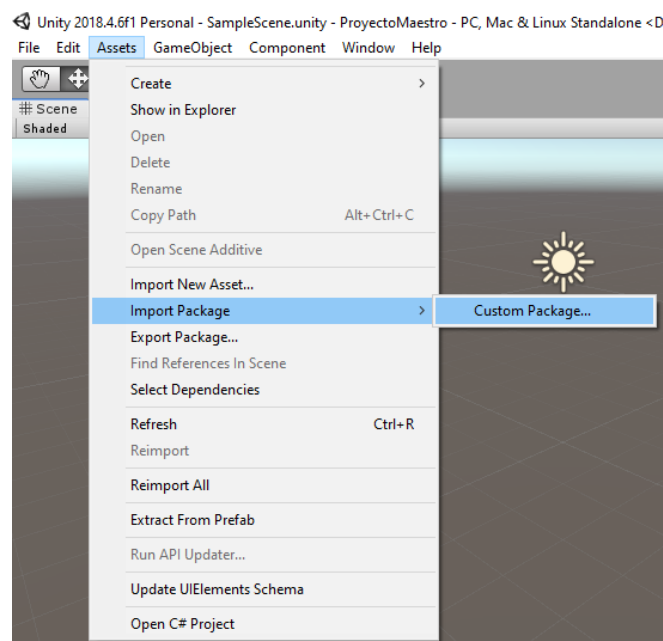


Figura 3.12 Selección de paquete del framework *MixedRealityToolkit*.

Ahora, debemos configurar la escena de Unity para que pueda usar la cámara y entradas de los HoloLens. Ahora en la barra de menú aparecerá un nuevo campo llamado “Mixed Reality Toolkit”. Seleccionamos este nuevo menú y tendrá dos opciones, seleccionamos “Add to Scene and Configure” como se muestra en la figura 3.13. Nos pedirá elegir un perfil para la configuración. Este perfil dependerá de la versión del dispositivo HoloLens con el que trabajemos. Para este proyecto se utiliza HoloLens 1 por lo tanto se selecciona el perfil “*DefaultMixedRealityToolkitConfigurationProfile*”. En caso de estar

utilizando el dispositivo HoloLens 2 se utilizaría el perfil “DefaultHoloLens2ConfigurationProfile”.

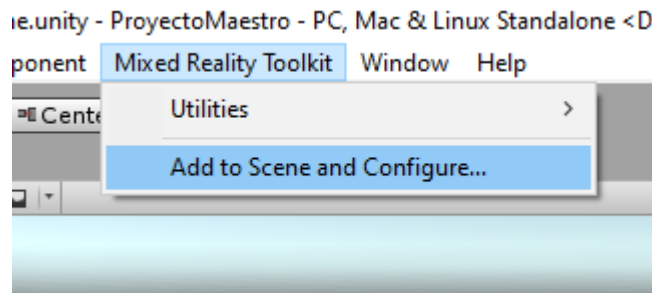


Figura 3.13 Añadir a la escena y configurar.

Ahora debemos realizar configuraciones de construcción del proyecto. Para ello nos dirigimos al menú y seleccionamos *File* → *BuildingSettings...* nos mostrará una ventana. Todas las escenas de nuestro proyecto deben estar en la construcción para poder visualizarlas una vez que la aplicación se esté ejecutando en los HoloLens. Para añadir las escenas seleccionamos “Add Open Scenes” como se muestra en la figura 3.14. En plataforma seleccionamos “Universal Windows Platform” con la siguiente configuración: “Target Device” HoloLens, “Architecture” x86, “Build Type” D3D, “Target SDK Version” Última instalada, “Minimum Platform Version”, 10.0.10240.0, “Visual Studio Version”, Última instalada, “Build and Run on” Máquina Local, “Build Configuration”, Release.

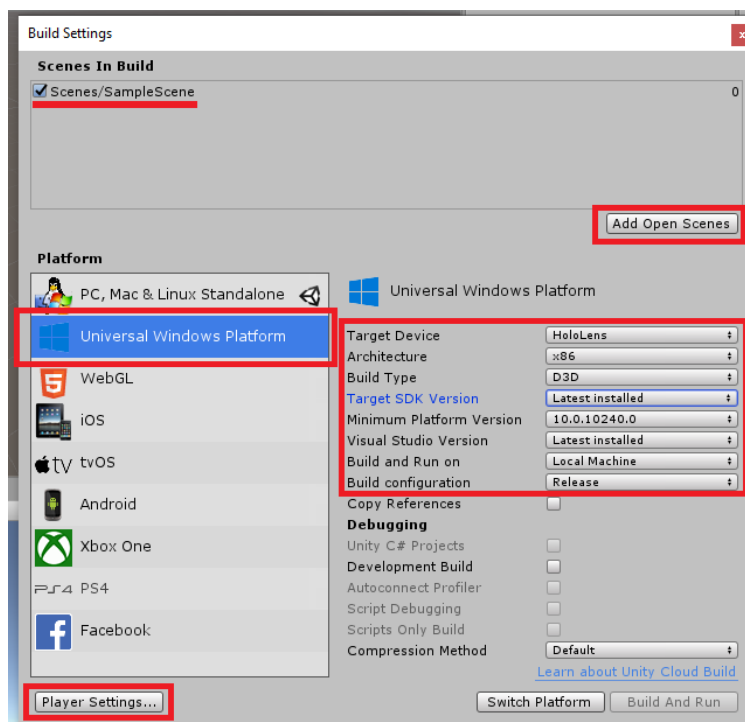


Figura 3.14 Ventana de configuración de construcción de la aplicación.

Finalmente seleccionamos “*Player Settings*” y nos abrirá la configuración en el inspector como se muestra en la figura 3.15. Seleccionamos la pestaña final en las configuraciones de la plataforma universal de Windows “*XR Settings*”. Activamos la casilla de “*Virtual Reality Supported*” y nos cargara un SDK de Windows. Seleccionamos la última casilla “*WSA Holographic Remoting*”. Ahora nos vamos a la pestaña de “*Publishing Settings*” y hasta capacidades. Seleccionamos “*InternetClient*”, “*InternetClientServer*”, “*WebCam*”, “*Proximity*”, “*Microphone*” y “*SpatialPerception*”. Luego más abajo en “*Supported Device Families*” seleccionamos “*Holographic*” y con esto ya podremos comenzar a trabajar en la aplicación.

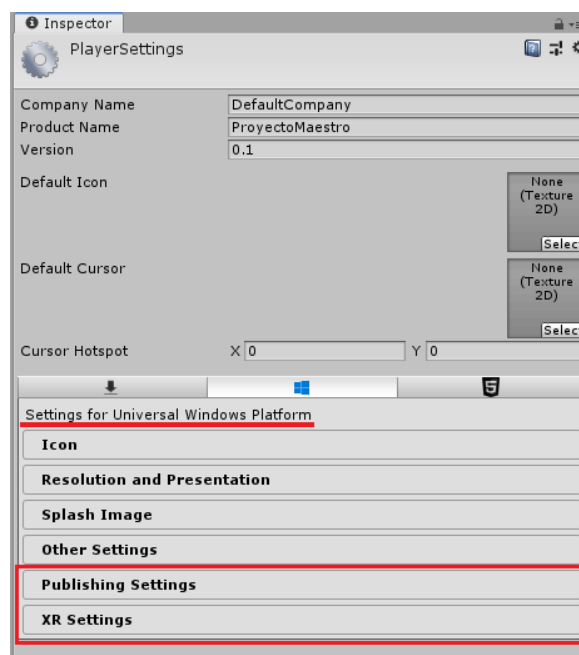


Figura 3.15 Configuraciones de “*Player*” en el inspector.

Una vez que ya contamos con un proyecto listo para probar en el dispositivo de los HoloLens debemos construir la solución, configurar, compilar e implementar. Primero, en *File* → *BuildSettings...* Presionamos el botón “*build*” de la ventana que se abrirá como se muestra en la figura 3.16. Ahora debemos darle un directorio en el cual guardara los archivos de la construcción. Creamos una carpeta llamada “*App*” y la seleccionamos, Unity comenzara a crear los archivos necesarios.

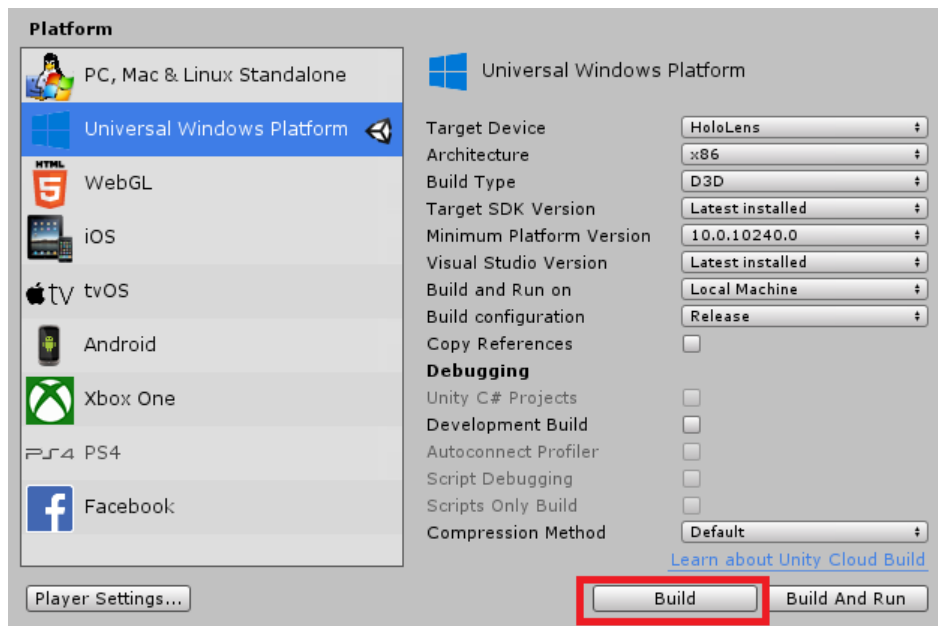


Figura 3.16 Ventana de configuración de la construcción.

Una vez finalizado este proceso en el directorio “App” se encontrará un archivo tipo “.sln” con el nombre del proyecto. Al abrirlo con Visual Studio debemos cambiar los parámetros de compilación en la parte superior central debajo de la barra de menú. En el “droplist” de configuraciones de soluciones seleccionamos “Release”, en plataformas de solución “x86” y “Máquina remota” para la ejecución como se muestra en la figura 3.17.

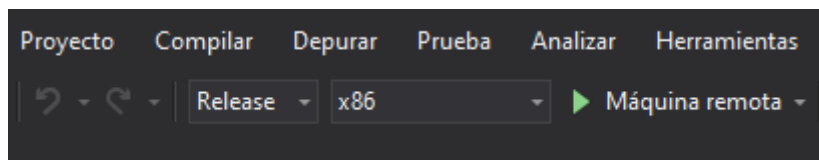


Figura 3.17 Configuración de compilación e implementación.

Por último, debemos especificar la dirección IP de la máquina remota donde será implementado en este caso los HoloLens. En la parte derecha se encuentra el explorador de soluciones ahí debería esta nuestra solución. La abrimos y seleccionamos el subdirectorio con el mismo nombre que nuestra solución, clic derecho y propiedades. En propiedades de configuración seleccionamos “Depuración” y en “Nombre de equipo” escribimos la dirección IP del dispositivo HoloLens como se muestra en la figura 3.18. Presionamos “Aceptar” y la configuración para compilar e implementar la aplicación en los HoloLens estará lista.

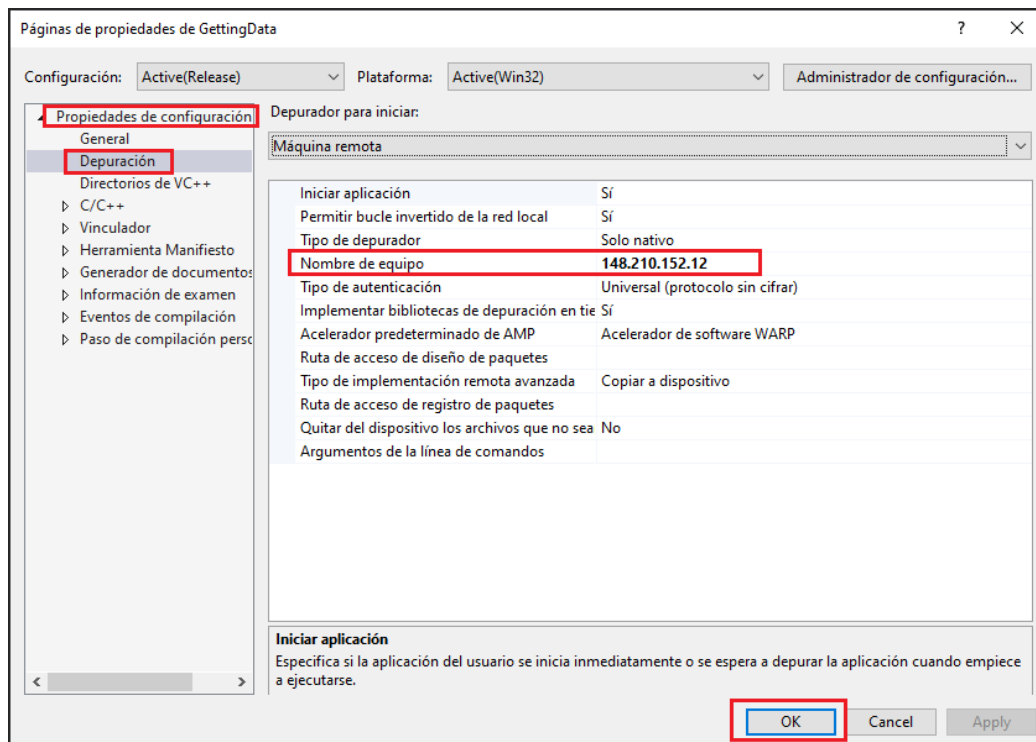


Figura 3.18 Configuración de la dirección IP del equipo remoto.

### 3.5.3 Prototipos en HoloLens

Para el desarrollo de la aplicación principal del sistema de realidad aumentada se desarrollaron algunos prototipos. Cada prototipo se enfoca en resolver segmentos que son partes de la aplicación principal los cuales son sistema de reconocimiento de imágenes, sistema de seguimiento y sistema de lectura de QRs.

#### ➤ Reconocimiento de imágenes

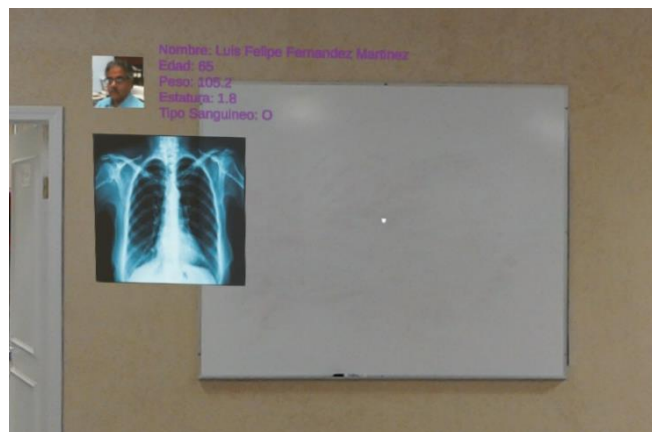
Se exploraron distintas opciones y funcionalidades para el desarrollo del sistema de realidad aumentada. Uno de ellos fue un prototipo de realidad aumentada para el reconocimiento de imágenes. Desarrollado en el motor Unity y el lenguaje C# a través de la herramienta Visual Studio 2019. Además, se utiliza el motor Vuforia el cual facilita el desarrollo de aplicaciones de realidad aumentada. Vuforia nos permite escanear objetos tridimensionales para poder reconocerlos a través de una aplicación. En este prototipo los objetos a ser reconocidos deben ser escaneados y cargados a una base de datos anteriormente. En la figura 3.19 se muestran dos objetos y encima de ellos elementos virtuales formados al ser reconocidos.



*Figura 3.19 Prototipo de reconocimiento de imágenes en realidad aumentada.*

#### ➤ Sistema de seguimiento de RA

Otro de los prototipos desarrollados consistió en un sistema que se conectara al sistema de almacenamiento y nos permitiera traer la información para poder visualizarla como elementos virtuales para la realidad aumentada. Al igual que el prototipo de reconocimiento se utilizó Unity y C#. Este prototipo se conecta a la base de datos y trae la información requerida para después mostrarla en una pequeña ventana que sigue el movimiento visual del usuario. En la figura 3.20 se muestra la ventana con la información obtenida del sistema de almacenamiento fija en la parte superior izquierda del campo de visión del usuario.



*Figura 3.20 Información extraída del sistema de almacenamiento visualizada en el sistema de RA.*

### ➤ Sistema de lectura de QRs

Se desarrolló un prototipo que escaneara y reconociera códigos QR los cuales nos permitirían obtener información de un paciente en particular sin necesidad de introducir nombres, ids, o cualquier otra forma de identificación que consuma tiempo. El escaneo de códigos QR por otro lado se realiza de manera casi instantánea. Para el desarrollo de este prototipo se utilizó Unity y C#. Además, fue necesario utilizar una librería llamada “zxing.unity” la cual nos permite utilizar un scanner de códigos QR entre otros. Sin embargo, la librería está diseñada principalmente para dispositivos móviles, escrita en Java. Fue necesario realizar algunas adecuaciones de código para poder hacerla funcional con un sistema de realidad aumentada para el dispositivo HoloLens. En la figura 3.21 se muestra el recuadro de escaneo y al lado la respuesta al escaneo.

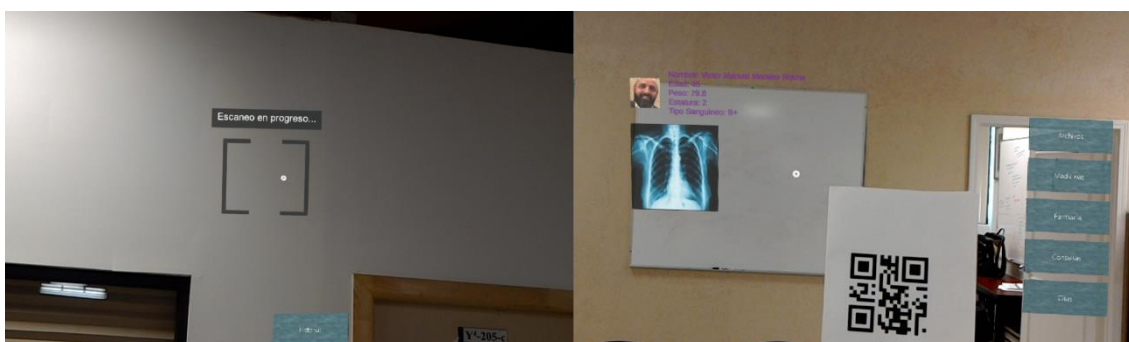


Figura 3.21 Prototipo de escaneo de códigos QR.

### 3.5.4 Aplicación de RA “Medical Rounds RA - Round”

El sistema de realidad aumentada fue nombrado como “*Medical Rounds, RA - Round*” fue diseñada como una aplicación nativa para el dispositivo de realidad aumentada *HoloLens*, por lo tanto, únicamente podrá ser utilizada a través de este dispositivo. Se desarrolló utilizando como base el marco de Microsoft MRTK. El sistema fue implementado en el dispositivo HoloLens de Laboratorio Nacional de las Tecnologías de la información (Lanti) y el Centro de la ingeniería del conocimiento y la ingeniería de software (CenICIS).

- **Metodología de desarrollo**

Para el desarrollo de la última aplicación (aplicación de realidad aumentada) se decidió continuar utilizando la metodología de desarrollo en cascada. En parte al ser una metodología confiable la cual nos mostró buenos resultados durante el desarrollo de las aplicaciones de soporte y authoring. Y por ser una metodología sencilla, pero practica con fases claramente definidas.

- **Análisis:**

Durante la fase de análisis del sistema de realidad aumentada en la metodología en cascada se generaron los artefactos de documento de especificación de requisitos y documento de casos de uso. Se determinaron las características y limitaciones clave del prototipo. Ya que a diferencia de los sistemas de authoring y soporte, el sistema de realidad aumentada se planeó para ser una aplicación para el dispositivo de realidad aumentada HoloLens. Sin embargo, este dispositivo nos limita en ciertos aspectos como se presenta en el documento de especificación de requisitos descrito en esta fase de la metodología.

Para el desarrollo se propuso el uso del marco de trabajo MRTK como se menciona en la sección 3.5.1. Además, la codificación será realizada en el lenguaje de programación C# en *Visual Studio* ya que se utilizará el motor de videojuegos *Unity* para la construcción de la aplicación.

- Documento de especificación de requisitos. Este documento particular tiene la intención de recabar y elaborar en la extracción y análisis de los requisitos para el sistema a desarrollar. A través del documento de especificación de requisito se establecieron las bases conceptuales del sistema de realidad aumentada. Además, este documento de requisitos también es fundamental para transmitir la información clave del proyecto a otras personas internas o externas al proyecto. El documento de especificación de requisitos completo del sistema de realidad aumentada puede ser consultado en c) del [\[Anexo B.c\]](#).

Los objetivos identificados para el sistema de realidad aumentada los cuales son tomados como los requisitos finales para el sistema son los siguientes:

- Módulo de “login”: El prototipo debe permitir el acceso a todo personal autorizado a través de sus credenciales de usuario. Los usuarios no podrán ser registrados en la aplicación de RA, únicamente a través del gestor de usuarios, sistema de soporte.

- El prototipo de RA permitirá realizar búsquedas por distintos medios: reconocimiento de QR, texto introducido a través de gestos HoloLens y por voz (nombre de paciente).
  - Una pequeña ventana “pop-up” mostrará una pequeña porción de información del paciente seleccionado. Seguirá la visión del usuario sin obstruirla.
  - Herramienta visual para mostrar la información del usuario. Consistirá en un recuadro o panel de 4 secciones donde podrán mostrarse 4 diferentes “ventanas” de información.
  - Opción para mostrar y ocultar la información del recuadro principal y el “pop-up con la información del paciente”.
  - Panel lateral con las diferentes secciones de información correspondiente al paciente. El usuario las seleccionara y deberán ser mostradas en el recuadro de visualización.
  - Vista de los perfiles del usuario. Se mostrará la información del paciente de acuerdo con el perfil seleccionado por el usuario.
- Documento casos de uso. El segundo artefacto generado fue el documento de casos de uso, una vez extraídos los requisitos clave para el desarrollo del sistema se crean los casos de uso basados en ellos. Como resultado se redactaron los casos de uso detallados para el sistema de realidad aumentada. Los casos de uso detallados pueden ser consultados en c) del [\[Anexo C.c\]](#).

El documento de casos de uso nos permitió repasar paso a paso los escenarios en que interactúan el usuario y el sistema. De esta manera, se planea una ruta de interacción llamado escenario de éxito el cual dicta los pasos para que el caso sea exitoso. Además, se planean los posibles “flujos alternos” los cuales son una desviación en la ruta de éxito. Un documento de casos de uso bien detallado nos permitirá avanzar en el proceso de desarrollo y específicamente a la fase de diseño.

En la tabla 3.3, se presenta el caso de uso C2 “Búsqueda de paciente”. El caso de uso muestra paso a paso el proceso de éxito en el cual un usuario realiza la búsqueda de un paciente en el sistema de realidad aumentada. Se describe que existen 3 tipos de búsqueda posible. Búsqueda por medio del código QR, búsqueda por medio del nombre o código de cuarto del paciente introducido en texto y finalmente la búsqueda a través de reconocimiento de voz utilizando el nombre del paciente. El escenario principal de éxito es la secuencia del usuario utilizando las 3 funciones de búsqueda y la respuesta del sistema a las acciones del usuario. Por último, se encuentran los “flujos alternos” los cuales representan acciones que no siguen el flujo del sistema y que pueden culminar en un caso de fracaso.

Tabla 3.3 Búsquedas de paciente.

<b>Caso de uso C2: Búsquedas de paciente.</b> Los usuarios una vez dentro del sistema serán presentados con una ventana de búsqueda. Aquí se podrán realizar búsqueda por escaneo de código QR, búsqueda introduciendo nombre o cuarto de paciente y búsqueda por reconocimiento de voz (nombre de paciente).	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Iniciar sesión en el sistema de realidad aumentada.	2: Mostrar la ventana de búsqueda con una breve introducción a los distintos tipos de búsqueda.
3: El usuario dice la palabra “scan”.	4: Se muestra un recuerdo para posicionar el código QR. Tras escanearlo el sistema muestra al paciente del QR. Si el QR pertenece un cuarto con varios pacientes se muestra una lista para seleccionar al usuario. FA1
5: El usuario dice la palabra “look for”.	6: El sistema muestra una ventana de grabación indicándole al usuario que puede decir el nombre del paciente a

	buscar. Se muestra al paciente encontrado. FA2
7: El usuario selecciona el icono de búsqueda en la ventana e introduce el nombre de paciente o código de cuarto “manualmente”.	8: El sistema realiza la búsqueda y muestra al paciente. Si la búsqueda arroja más de 1 resultado se muestra una lista de los pacientes para seleccionar por el usuario.
<b>Extensivos (Flujos alternativos)</b>	
FA1 – No se encuentran pacientes para el código escaneado. Falla al conectarse con el sistema de almacenamiento.	
FA2 – Fallo con los sistemas de reconocimiento de voz de Azure. No se encontraron pacientes con ese nombre.	

○ **Diseño:**

Para la segunda fase de la metodología, fase de diseño se tomaron como entrada el documento de especificación de requisitos y los casos de uso detallados. Como resultado se generaron las pantallas o plantillas del sistema de realidad aumentada. El sistema no contará con una base de datos particular, en su lugar, se conectará directamente con el sistema de almacenamiento para obtener los datos requeridos.

- Base de datos. El sistema se conecta directamente al sistema de almacenamiento implementado en la instancia EC2 de aws.
- Pantallas del sistema de realidad aumentada (interfaz de usuario). Las pantallas de interfaz de usuario fueron diseñadas en base a los casos de uso detallados ya que estos representan la funcionalidad que el sistema final debe poseer. Estas pantallas son presentadas de manera 2D, sin embargo, al tratarse de un dispositivo de hologramas como los HoloLens no hay forma de capturar la profundidad en las pantallas. Por lo tanto, estas sirven únicamente como guía en la estructura general del sistema de realidad aumentada.

A continuación, en la figura 3.22 se muestra una de las pantallas del sistema de authoring. En esta pantalla se muestra la interfaz la herramienta de visión de la información de pacientes “AroundFrame” en el cual se presentan los elementos: *pop-up* con información breve del paciente, marco de información central y panel lateral izquierdo con las secciones de información del paciente.

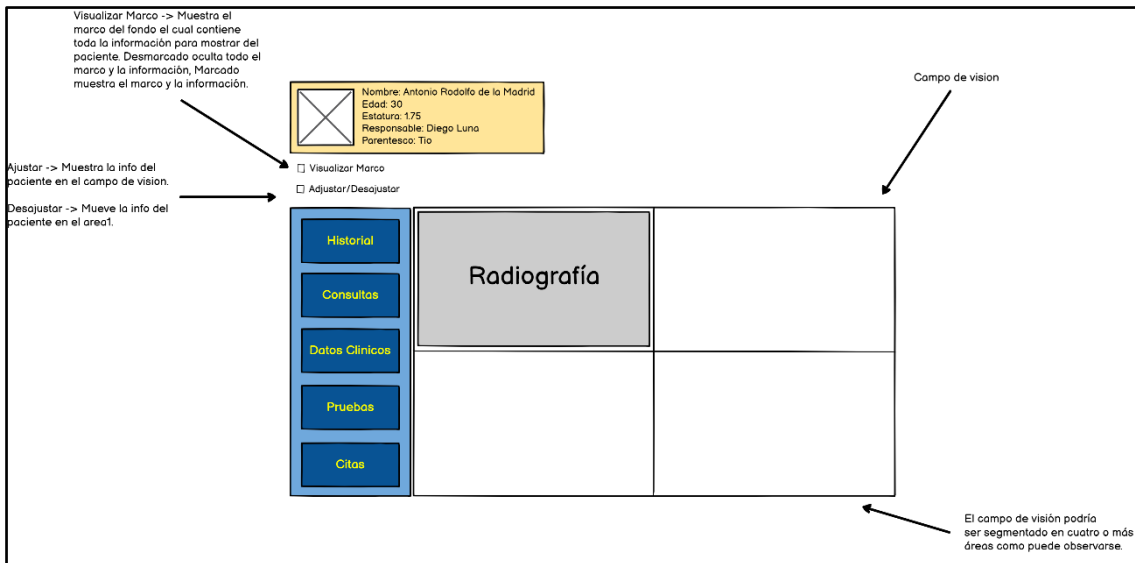


Figura 3.22 Pantalla “AroundFrame” del paciente seleccionado.

Todas las pantallas que fueron diseñadas para el sistema de realidad aumentada y en las cuales se basó la aplicación pueden ser consultadas al final de este documento en el [\[Anexo G\]](#).

○ **Desarrollo:**

Para el desarrollo de la aplicación de realidad aumentada se decidió utilizar el motor de videojuegos Unity ya que en su momento fue el único medio por el cual era posible desarrollar aplicaciones y programas para los HoloLens. El documento de casos de uso se utilizó para fijar las funcionalidades del sistema y las pantallas fueron utilizadas como una guía en el estilo y diseño de los componentes internos y gráficos de la aplicación. La plataforma para el desarrollo consistió en una computadora local con 8 GB de memoria RAM, 200 GB disco duro, procesador Intel i5-6500 y con sistema operativo Windows 10 y tarjeta gráfica GeForce GTX 970.

Las herramientas tecnológicas utilizadas fueron *Unity 2018.4.6f1* y *Visual Studio 2019*.

- **Framework MRTK y Plataforma de desarrollo.**

Como se describe en la sección 3.5.1 el marco de trabajo a utilizar es el *MixedRealityToolKit* de Microsoft ya que nos brinda las herramientas y elementos necesarios para desarrollar para la plataforma del dispositivo HoloLens. Además, en la sección 3.5.2 se describe el proceso de configuración para la plataforma de desarrollo en *Unity* y *Visual Studio*.

- **Desarrollo de la aplicación (Módulos).**

Toda la codificación de la aplicación de realidad aumentada se realizó en Visual Studio, mientras que el diseño y estructura de la interfaz de usuario fue realizada en Unity. Sin embargo, tras la compilación fue necesario realizar pruebas en el dispositivo HoloLens. Unity proporciona la posibilidad de una simulación interna para probar algunas de las características de la aplicación. Aun así, es imposible simular la realidad aumentada desde una perspectiva en primera persona como en los HoloLens.

El prototipo de realidad aumentada fue desarrollado bajo la necesidad de una aplicación dinámica e interactiva con los usuarios (personal médico) que pudiesen llevar con ellos mientras realizan las rondas médicas. Y que además le permitiera ver y revisar información importante del paciente de la ronda médica.

A continuación, se describen los módulos que conforman la aplicación de realidad aumentada.

- **Módulo de autenticación:** Un detalle importante a mencionar del marco de trabajo MRTK es la imposibilidad de eliminar cualquier escena que contenga el marco “instalado”. Esto se debe a que requiere una configuración sobre algunos de sus componentes que al momento de eliminarlo se pierde totalmente. Son valores y elementos fijos o estáticos durante la ejecución del sistema. Sin embargo, en Unity es posible tener varias escenas activas

al mismo tiempo. Gracias a esta cualidad podemos tener una escena principal con el framework mientras agregamos o eliminamos escenas con distintas funciones en la aplicación.

El primer módulo de la aplicación es el de autenticación. Al iniciarse la aplicación al usuario se le mostrará una escena de “pantalla de inicio de sesión” en la cual se deben introducir credenciales de autenticación (nombre de usuario y contraseña). Una vez introducidas las credenciales la interfaz llama el script de verificación el cual se conecta con el sistema de almacenamiento, ahí se encuentran los scripts PHP encargados de buscar la información del usuario en la base de datos. De encontrarse alguna coincidencia, devolverá un valor de tipo *bool* de verdadero, de otra manera será un valor falso. El script mostrará un mensaje de error en caso de falso o cargará la siguiente escena en caso de verdadero.

- **Módulo de selección de paciente y vista de datos:** Para el caso de las rondas médicas el usuario debe seleccionar un paciente para continuar con el flujo de la aplicación. La vista de datos sería un módulo por sí mismo. Sin embargo, debido a que la selección de un paciente inmediatamente cargara su número de identificación (id único cargado en el *GameManager*<sup>2</sup>) para poder cargar su información se decidió unirlos en uno solo. El primer submódulo del que se compone se encuentra en un objeto de juego (*AroundManager*) con un par de scripts especiales “*Qrcode*” “*voiceRec*” y “*AroundBehavior*”, el primer script se ocupa de realizar la búsqueda de pacientes por medio de códigos QR y el segundo activa el reconocimiento de voz para capturar

---

<sup>2</sup> Objeto de juego (con su correspondiente script) utilizado para manejar los diferentes estados del juego. Proporciona un mecanismo en segundo plano encargado de realizar transiciones entre objetos de juego e incluso escenas.

en voz el nombre del paciente. Ambos scripts trabajan con “*AroundBehavior*” el cual está encargado de conectarse al sistema de almacenamiento para obtener y comprobar los datos del paciente. “*AroundManager*” es un objeto de juego vacío e invisible, ya que como su nombre indica es el encargado únicamente de administrar funciones y eventos en segundo plano. El segundo submódulo se encuentra en el objeto de juego “*AroundPresentation*” el cual es un objeto “padre” con varios objetos de juego anidados. En este objeto padre se encuentra el hijo llamado “*PSearch*” el cual posee un script llamado “*Keyword*” el cual al ser llamado renderizará un teclado holográfico por el cual el usuario podrá ingresar el nombre o código del paciente. Como complemento, existe el objeto “*searchIconBTN*” el cual desencadena el evento de búsqueda de los datos ingresados. Una vez que se ha seleccionado el paciente entran los submódulos de vista de datos. El script “*AroundManager*” llama al nuevo objeto de juego “*AroundFrame*” el cual es el principal objeto encargado de la vista de datos del paciente. Se compone de los siguientes objetos hijos: “*LeftSection*”, “*MainFrame*” y “*Patient\_pop-up*”. En el objeto “*LeftSection*” posee objetos para realizar diversas tareas dependiendo de la situación. El objeto “*Patient\_pop-up*” es un pequeño mensaje flotante con información básica y una fotografía del paciente seleccionado. Con el hijo “*AttachBox*” puede reposicionarse a una posición menos “invasiva” para el usuario si así lo desea. El objeto hijo “*SeeFrame*” posee un script con una característica especial la cual se encarga de ocultar el objeto “*MainFrame*” de manera que no interfiera con la visión del usuario si así lo considerará. El objeto “*LeftPanel*” se compone de una serie de objetos con scripts que le permiten mostrar secciones con botones con la información del paciente seleccionado. Las secciones se dividen en: información general del paciente, pruebas

médicas, radiografías, información de exploración física, historial clínico y los perfiles. Por otra parte, “*MainFrame*” es el lugar en el cual se muestra la información seleccionada del “*LeftPanel*”, el objeto únicamente contiene un script “controlador” que se encarga de realizar tareas relacionadas con la vista de datos. Su principal objetivo es ser un contenedor de la información. Finalmente, el objeto “*ProfilePanel*” contiene el script “*Profiles*” el cual utiliza la información de “*AroundManager*” para cargar los perfiles del usuario del sistema a través del sistema de almacenamiento y mostrarlos en “*MainFrame*” al usuario.

- **Submódulo *AroundBehavior*:** Este submódulo es el principal encargado de la comunicación entre la aplicación de realidad aumentada y el sistema de almacenamiento. En el sistema de almacenamiento se encuentra un administrador de archivos el cual contiene una variedad de archivos PHP los cuales son utilizados para extraer la información de la base de datos y transmitirla al sistema de RA. Los archivos son: “*ExplorationDataRequest*” encargado de la información de exploración, “*XRayDataRequest*” encargado de las imágenes de radiografía de los pacientes, “*HistoryDataRequest*” datos del historial clínico, “*TestsDataRequest*” información de pruebas del paciente y “*PatientDataRequest*” información general del paciente. Además de los scripts de autenticación de usuarios.

- **Pruebas:**

Para realizar las pruebas de usabilidad, confiabilidad y rendimiento la aplicación de realidad aumentada deber ser compilada y construida para ser instalada en el dispositivo HoloLens. Las pruebas realizadas a la aplicación de realidad aumentada como se menciona anteriormente son en cuanto a usabilidad, confiabilidad y rendimiento. A los usuarios de las pruebas se les instruyo a seguir una serie de instrucciones. Posteriormente,

describieron sus opiniones del manejo de la aplicación y se obtuvieron los siguientes datos:

- Usabilidad: Antes de poder describir los detalles de las pruebas de usabilidad hay ciertas características del dispositivo HoloLens que deben ser expuestas. Los HoloLens son un dispositivo independiente el cual utiliza un sistema particular para realizar a cabo la interacción con el usuario. Una de sus características es “gaze” el cual es un punto fijo en medio de la vista de los HoloLens el cual tiene la misma función de un puntero de sistema operativo. Sin embargo, requiere de “gestos” para poder ser utilizado. Estos “gestos” son acciones llevadas a cabo con las manos las cuales desencadenan eventos en el sistema del dispositivo. Por esta razón cualquier usuario nuevo al uso del dispositivo debe ser introducido por medio de un tutorial. Solo el uso constante de los HoloLens nos permite dominar las acciones requeridas para desempeñar cómodamente las funcionalidades de cualquier aplicación en el dispositivo. La usabilidad fue medida al igual que en los sistemas de soporte y authoring por medio de tres métricas (exactitud, tiempo empleado, conformidad del usuario). La exactitud consiste en las acciones del usuario sin cometer errores dentro de la aplicación. Sin embargo, los usuarios, nuevos al dispositivo, presentaron algunas dificultades menores para realizar las acciones inicialmente. Aun así, no se cometieron errores por parte de los usuarios al probar las funcionalidades de la aplicación. El tiempo empleado por los usuarios fue un poco más alto del esperado, pero es justificable debido a que era la primera vez que utilizaban el dispositivo. Los usuarios recuerdan las acciones que siguieron para realizar las tareas o procesos en la aplicación. En general, el usuario únicamente expreso la dificultad de acostumbrarse a los gestos necesarios para manejar la aplicación.
- Confiabilidad: La aplicación de realidad aumentada fue diseñada y desarrollada para seguir un estricto proceso o secuencia de pasos para su uso. Además, al carecer de entradas de datos que deban ser

validadas (además de la autenticación) se redujeron los errores por parte del usuario. Por lo tanto, determinamos que la aplicación cumple con la confiabilidad deseada en cuanto a la funcionalidad propia de la aplicación y sus componentes.

- **Rendimiento:** El rendimiento de la aplicación es una cuestión de suma importancia cuando se desarrolla algo para un dispositivo con recursos limitados como los HoloLens. Inicialmente se utilizaron texturas y materiales gráficos no optimizados lo cual perjudicaba el rendimiento de la aplicación. Por lo tanto, se realizaron modificaciones y configuraciones en las texturas sin sacrificar calidad. Se optó por usar texturas translúcidas en lugar de sólidas con un brillo alto. De esta manera, no es necesario generar luces dentro del holograma para poder verlo claramente. Además, se cambió a una construcción de tipo “*Master*” en lugar de la comúnmente usada por la comunidad “*Release*”. La primera configuración está diseñada específicamente para Unity lo cual remueve elementos innecesarios que mejoran el rendimiento. Durante las pruebas de usabilidad se pudo comprobar la fluidez y dinamismo de la aplicación en el dispositivo. Y se comprobó que funciona correctamente con los recursos provistos.
- **Implementación:** La construcción y compilación de la aplicación de realidad aumentada se realizó en con los recursos descritos al inicio de la fase de desarrollo de la subsección 3.5.4. La implementación fue realizada únicamente en el dispositivo HoloLens del Lanti (Laboratorio Nacional de Tecnología de la Información).

## **IV. Resultados**

En el presente capítulo presenta los resultados obtenidos del proceso de instanciación del marco de trabajo enfocado a las rondas médicas. Se pretende mostrar la comunicación y colaboración realizada entre los distintos sistemas del marco.

También, se mostrará el proceso de funcionalidad e interacción de los sistemas desarrollados y los usuarios a través de una serie de ilustraciones. Se explicarán los detalles del diseño y estructura de los sistemas en el marco de trabajo.

### **4.1 Marco de trabajo y sus cuatro sistemas**

En el capítulo 2 “Marco teórico” presentamos las bases que tomamos para realizar este trabajo de tesis. El principal objetivo era crear un marco de trabajo enfocado a trabajar con aplicaciones de realidad aumentada. Después crearíamos una instancia de este marco de trabajo eligiendo un área de dominio. Como se presenta en la subsección 2.4 el área elegida fueron las rondas médicas realizadas por el personal médico de hospitales y clínicas.

El marco propuesto fue diseñado y estructurado de manera que exista un sistema para los expertos del dominio, el componente de soporte, un sistema para almacenar y obtener información, el componente de almacenamiento, un sistema encargado de utilizar estos datos, el componente de authoring y un sistema para estructurar y organizar la información que será utilizada, el sistema de RA.

Una aplicación de realidad aumentada requiere un conjunto de componentes para trabajar y comunicarse correctamente con los sistemas descritos en el marco de trabajo. Nosotros proponemos que una aplicación de RA para el marco debe poseer tres capas principales: autenticación, selección de perfil y vista de datos.

Proponemos un marco de trabajo para guiar el diseño y desarrollo de todo un ambiente o entorno en el cual podamos obtener, compartir y ver la información proporcionada. Con el objetivo de crear instancias basadas en este marco siguiendo su estructura. No está diseñado para un área específica o un tipo de información, su propósito es ser usado en distintas áreas y enfoques. Sin embargo, puede ser enfocado a un área específica durante la instanciación dependiendo de la información suministrada.

El marco promueve el diseño y desarrollo de cuatro componentes principales como su núcleo y están compuestos por diferentes sistemas. Cada sistema pertenece a un solo

componente particular. En el capítulo 3 se presentan las ilustraciones de los diferentes componentes del marco que se describen a continuación.

#### **4.1.1 Componente de almacenamiento**

El componente de almacenamiento es la base del marco descrito en este trabajo. Todos los datos utilizados y gestionados por los sistemas de los otros tres componentes se almacenan aquí. Cada sistema de los demás componentes se conectará a él para obtener, modificar o almacenar datos. Además, es el único sistema en todo el marco que no tiene otros módulos o subsistemas. De esta forma, se puede utilizar cualquier tipo de tecnología o herramientas para su desarrollo y construcción, dependiendo de la información y datos proporcionados.

#### **4.1.2 Componente de soporte**

El componente de soporte puede ser desarrollado e implementado como una aplicación móvil, una aplicación de escritorio o una aplicación web. Es utilizado por un administrador (experto o personal autorizado) para gestionar datos e información. Puede realizar algunas de las siguientes tareas: gestión de usuarios, gestión de recursos, gestión de archivos, vista de perfiles, informes, etc.

Su propósito es ser el punto de gestión de los datos de usuario de los sistemas de soporte, authoring y RA, además de la información del área de aplicación. Se conecta directamente al sistema de almacenamiento y permite agregar, modificar y eliminar información y datos, con los permisos correspondientes para cada tipo de usuario. Todas las cuentas de usuario para los sistemas de authoring y RA deben ser creadas por este sistema.

- **Sistema del experto**

El sistema del experto tiene una interfaz gráfica de usuario que se utiliza para interactuar con los módulos de administración de usuarios, recursos, perfiles, etc. Además, se utiliza para agregar cualquier cosa necesaria como parte de los datos desde el área de aplicación. Es posible agregar más módulos para satisfacer cualquier necesidad o para administrar datos específicos suministrados al marco de trabajo. Sin embargo, los módulos de gestión son esenciales, ya que a través de ellos le damos al usuario acceso a datos específicos y sistemas del marco.

#### **4.1.3 Componente de authoring**

Nosotros diseñamos el componente de authoring con base en las características y cualidades que debe tener una aplicación para crear contenido para RA. El contenido aumentado dependerá de los datos proporcionados al marco de trabajo. Nosotros proponemos una de las formas más básicas de generar contenido aumentado a través de perfiles de información compuestos por los datos en el sistema de almacenamiento.

- **Sistema de creación**

El personal experto o capacitado, preferiblemente acostumbrado al manejo de los datos del dominio, diseña los “*layouts*” utilizados para crear perfiles en el sistema de creación. Los datos pueden ser textuales, como manuales y documentación, o archivos multimedia como videos, imágenes, modelos 3d, etc. Los “perfiles”, formados por “*layouts*”, se transforman en contenido utilizado por el sistema de RA en el marco de trabajo. El experto o usuario del sistema de creación no necesita tener experiencia en la creación de software como tal. Solo será necesario que brinden la información que desean que se agregue al sistema de RA.

#### **4.1.4 Componente de RA**

Este es el sistema que se ejecuta en el dispositivo de RA. Consiste en una interfaz para los usuarios que necesitan ver los perfiles creados por el sistema de creación utilizando los datos suministrados previamente desde el componente de almacenamiento. A través de esta interfaz, el usuario debe autenticarse, seleccionar el perfil o los datos deseados y se mostrará la visualización de ese perfil. Los perfiles solo se pueden crear y modificar a través del sistema de creación (authoring). Los usuarios de este sistema no necesitan poseer experiencia en la creación de software como tal.

- Sistema de RA (sistema del dispositivo)

Este es el sistema que se ejecuta en el dispositivo de realidad aumentada. Se compone de tres subsistemas o módulos. En una aplicación de RA, la experiencia de interacción, normalmente conocida como “experiencia de RA” es la característica primordial que debe tener una aplicación de RA. A través de ella el usuario experimentará las ventajas que la RA puede proporcionar. El usuario interactúa con los módulos internos a través de la interfaz. A continuación, se detallan los 3 módulos principales.

- Autenticación: El módulo de autenticación es la primera interacción entre el usuario y el sistema de RA. A través de una interfaz, el usuario debe ingresar sus credenciales de acceso previamente administradas en el

sistema de soporte. Las credenciales de autenticación se pueden ingresar de manera “tradicional” por nombre de usuario y contraseña o mediante reconocimiento de voz.

- Selección de perfil: El módulo de selección de “perfil” o el módulo para seleccionar la estructura de los datos definidos en el sistema de authoring. Depende principalmente del tipo de información o datos suministrados al sistema de almacenamiento. Por ejemplo, consideremos las tareas de mantenimiento en un taller mecánico. Los perfiles pueden diseñarse para mantenimiento o reparación. Por otro lado, en un hospital, el personal médico podría seleccionar un paciente, una habitación, un medicamento, información de otro personal médico, etc. Se sugiere que la selección de datos se realice mediante el ingreso de información, reconocimiento de códigos de barras o códigos QR, o comandos de voz. De esta manera se facilita la tarea de búsqueda y la experiencia del usuario es mucho menos complicada. Siempre y cuando los datos que deba hablar el usuario no sean de alta sensibilidad o privados.
- Vista de los datos: El módulo de vista de los datos toma todos los datos del sistema de almacenamiento que requiera el usuario y los reestructura de acuerdo con lo que ha especificado en el perfil de authoring. La experiencia de interacción debe ser lo más placentera posible para el usuario. La visión del usuario no debe estar sobrecargada con contenido de realidad aumentada. Una sugerencia importante es delimitar un área en la que se presente información de tipo texto, y otra para videos, imágenes, etc.

#### **4.2 Presentación de resultados (instanciación del marco de trabajo)**

En la siguiente subsección se describen los resultados obtenidos al término de este trabajo de tesis uso de realidad aumentada en hospitales: rondas médicas. Estos resultados consisten en una serie de capturas de pantalla en las que se muestra el funcionamiento de los sistemas instanciados del marco de trabajo. Durante el desarrollo de los distintos sistemas no se realizaron cambios en las funcionalidades planeadas ni se agregaron o removieron. Sin embargo, si hubo algunos cambios estéticos y de estructura en comparación a las pantallas de interfaz creadas en las fases de diseño.

#### 4.2.1 Sistema de almacenamiento

Como se mencionó en la sección 3.2 desarrollo del sistema de almacenamiento, se utilizó una estancia EC2 de aws para su implementación. La instancia cumple la función de un servidor de base de datos y archivos en el cual se almacena toda la información del marco de trabajo. En la figura 4.1, se muestra el gestor de la base de datos del sistema de almacenamiento en el cual se encuentra la base de datos “hospital”. La información necesaria para los otros tres sistemas se encuentra en esta base de datos. Información de los usuarios (personal médico), pacientes (exploración física, historial clínico, radiografías y pruebas) y perfiles para el sistema de RA.

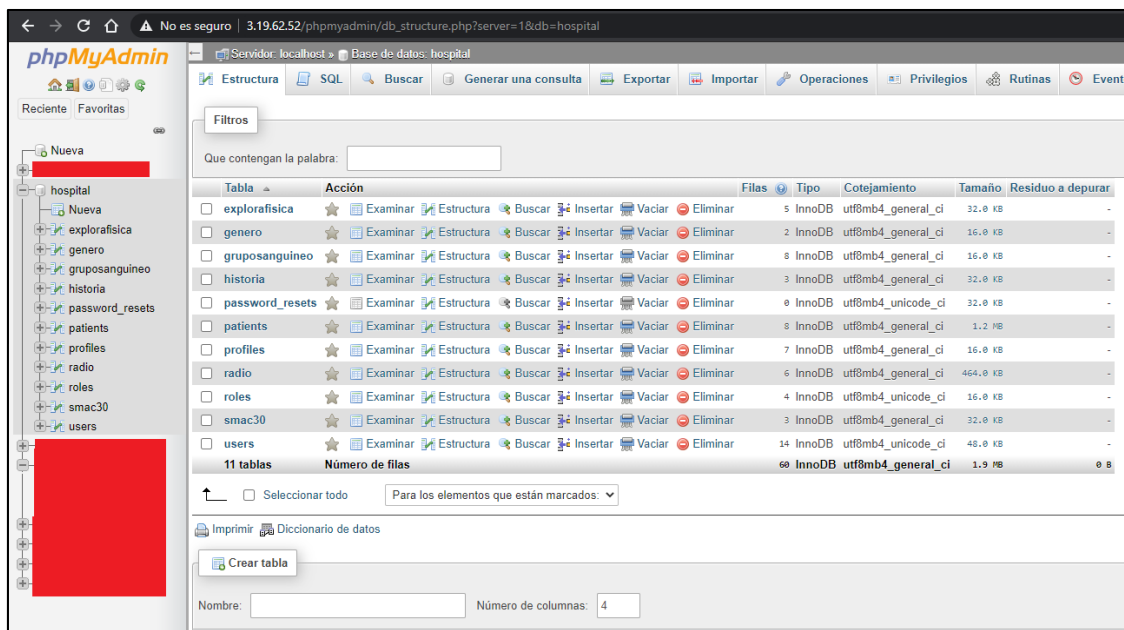


Figura 4.1 Gestor de base de datos del sistema de almacenamiento.

Además de la base de datos el sistema de RA requiere acceso a una serie de archivos PHP en el sistema de almacenamiento para llevar a cabo algunas de sus funciones. En la figura 4.2, se muestra los archivos en el sistema de almacenamiento. Para más detalles de estos archivos consultar el submódulo “AroundBehavior” en “Desarrollo” de la subsección 3.5.4 (Aplicación de RA “Medical Rounds RA-Round”).

```

root@ip-172-31-29-195:/home# cd /opt/
root@ip-172-31-29-195:/opt# cd lampp/
root@ip-172-31-29-195:/opt/lampp# ls
README-wsrep  ctlscrip.sh  info  manager-linux-x64.run  properties.ini
README.md     docs         lampp  manual                 sbin
RELEASENOTES  error        lib    modules                share
THIRDPARTY    etc          lib64  mysql                  www
apache2       htdocs      libexec  pear                   uninstall
bin           icons       licenses  php                    uninstall.dat
build         img         logs     phpmyadmin             var
cgi-bin       include     man      proftpd                xampp
root@ip-172-31-29-195:/opt/lampp# cd htdocs/
root@ip-172-31-29-195:/opt/lampp/htdocs# ls
ARauthoring  ARsupport.zip  bitnami.css  img  webalizer
ARauthoring.zip  HelloUnity    dashboard    index.php
ARsupport     applications.html  favicon.ico  proj
root@ip-172-31-29-195:/opt/lampp/htdocs# cd HelloUnity/
root@ip-172-31-29-195:/opt/lampp/htdocs/HelloUnity# ls
ExplorationDataRequest  ProfileDataRequest  helloworld.php  radiodata.php
HistoryDataRequest      RadioDataRequest    login.php
PatientDataRequest      TestDataRequest     patientdata.php
root@ip-172-31-29-195:/opt/lampp/htdocs/HelloUnity#

```

Figura 4.2 Archivos PHP del sistema de Almacenamiento.

#### 4.2.2 Sistema de soporte

Al igual que el sistema de almacenamiento, el sistema de soporte esta hospedado en una instancia EC2 de aws. La dirección IP de está instancia es dinámica, por lo tanto, puede cambiar en cualquier momento si así lo requiere aws. Sin embargo, pueden asignarse instancias estáticas con un consto mensual de 5\$ para sistemas fijos y en producción. En la figura 4.3, se muestra la página de bienvenida del sistema de soporte. Se diseñó de esta manera para poder llenar la página de información introductoria en caso de que fuera necesario.

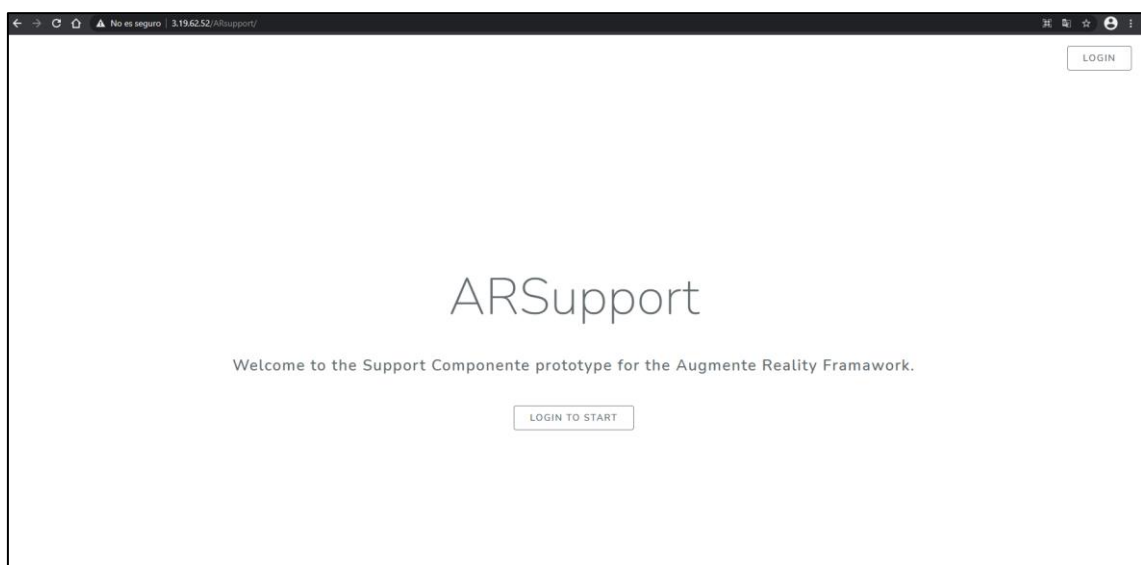
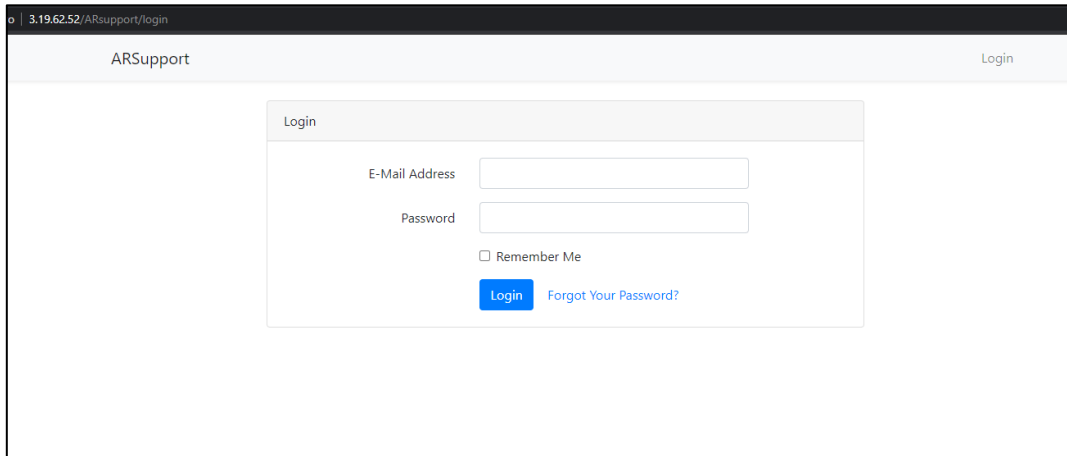


Figura 4.3 Página de bienvenida del sistema de soporte.

El sistema de soporte solo da entrada a personal autorizado (personal médico y administrativo) como se muestra en la figura 4.4.



*Figura 4.4 Inicio de sesión del sistema de soporte.*

Una vez que el usuario se ha autenticado correctamente será llevado al “Dashboard” de la aplicación. En la figura 4.5, se muestra el Dashboard de usuario con las secciones perfil del usuario, usuarios y pacientes.

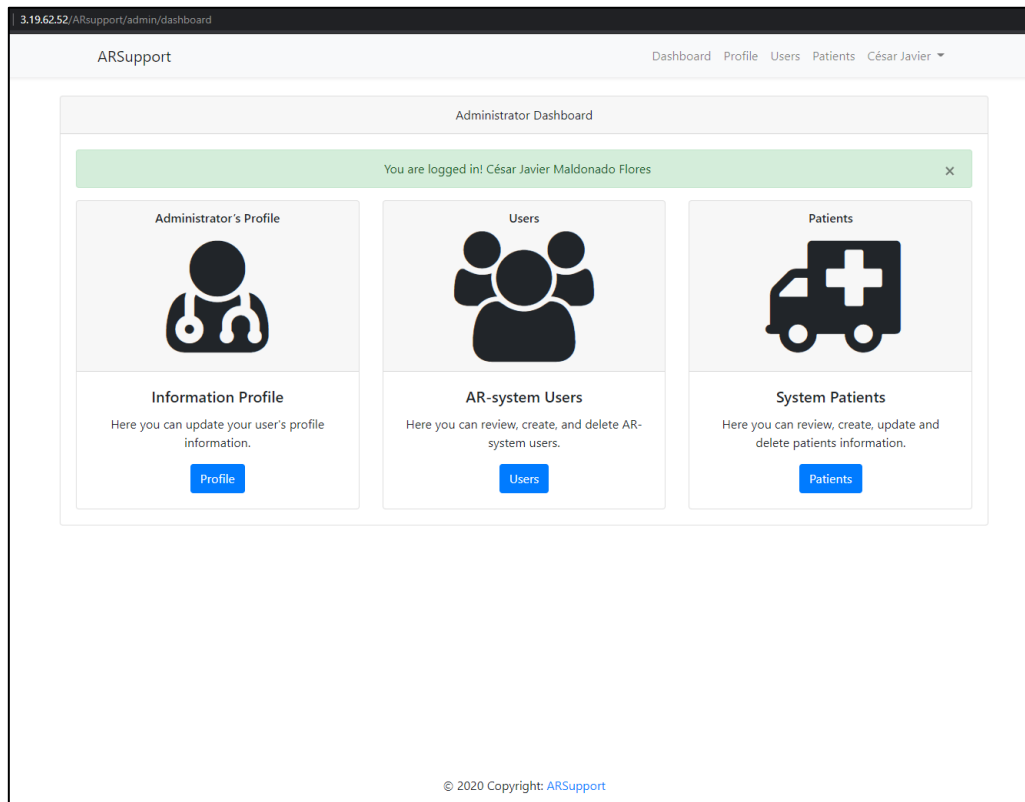


Figura 4.5 Dashboard de usuario del sistema de soporte.

El usuario podrá consultar la información de su perfil en la sección “*profile*”. Dependiendo del tipo de usuario se mostrará administrado, doctor, enfermera o estudiantes. En la figura 4.6, se muestran los datos disponibles del perfil de usuario, role, primer y segundo nombre, nombre de usuario, dirección de correo electrónico, contraseña y su confirmación.

The screenshot displays the 'Administrator Information Profile' page in the ARSupport system. The page header includes 'ARSupport' and navigation links for 'Dashboard', 'Profile', 'Users', 'Patients', and 'César Javier'. The main content area features a profile card with a medical icon and the following fields:

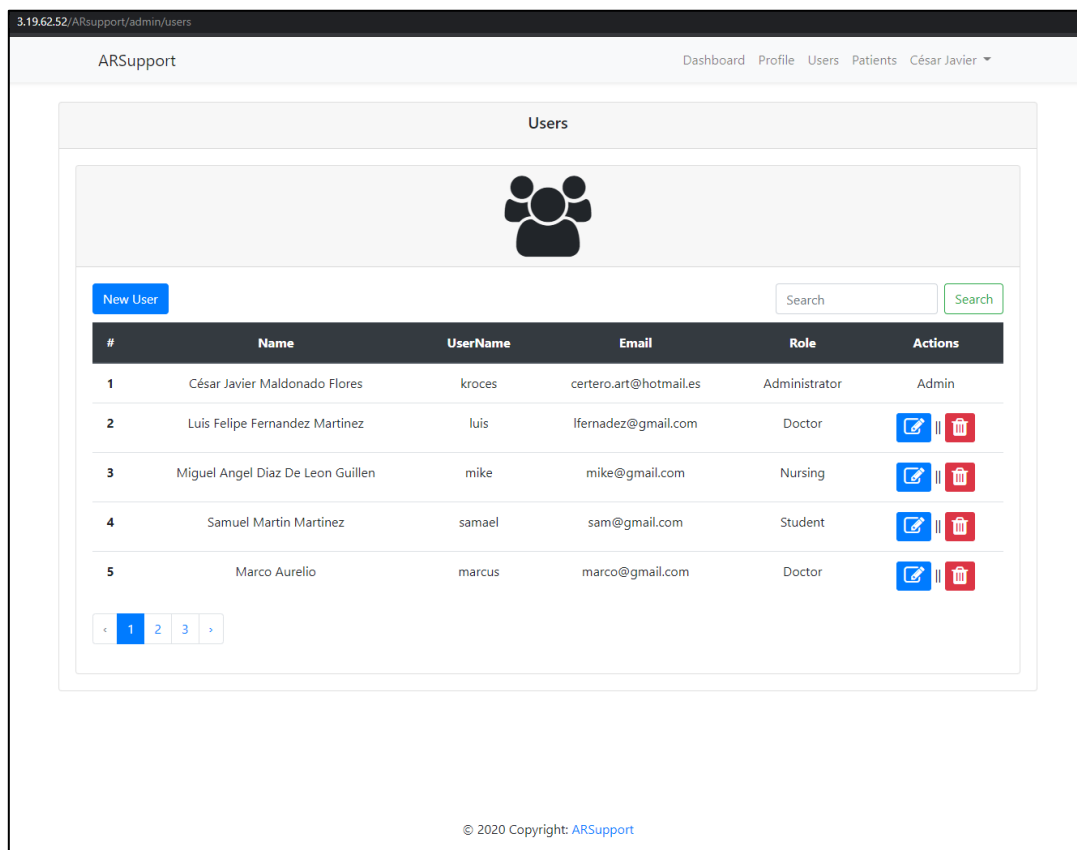
Role:	Admin
FirstName:	César Javier
LastName:	Maldonado Flores
UserName:	kroces
E-Mail Address:	centero.art@hotmail.es
Password:	
Confirm Password:	

An 'Update' button is located below the password fields. The footer contains the text '© 2020 Copyright: ARSupport'.

Figura 4.6 Perfil de usuario en el sistema de soporte.

La segunda sección del Dashboard se llama usuarios y se refiere a los usuarios (personal médico y administrativo). Cada usuario posee un nivel de acceso o rol en la aplicación, existen administradores, doctores, enfermeras y estudiantes. Los administradores son el rango más alto del sistema. Pueden crear nuevos usuarios de todas las categorías, incluso administradores. Los doctores pueden crear usuarios de tipo enfermera y estudiante. Poseen acceso total al sistema de soporte, ya que pueden

crear nuevos pacientes y llenar su información. Las enfermeras pueden crear usuarios estudiantes únicamente. Al ser parte importante del funcionamiento de un hospital y durante las rondas médicas pueden crear nuevos pacientes y modificar su información. Por último, los estudiantes son el rol con menos permisos dentro del sistema. No pueden crear nuevos usuarios ni pacientes, tampoco pueden eliminar su información. Únicamente son capaces de capturar datos para pacientes nuevos o recién ingresados. En la figura 4.7, se muestra la página de usuarios en la cual aparecen listados los usuarios del sistema. Según los permisos de los usuarios aparecerá el botón “nuevo usuario” y los botones de “editar” y “eliminar” junto a la información del usuario. También es posible realizar búsqueda de usuarios con la barra de búsqueda a través del nombre, nombre de usuario y correo electrónico.



*Figura 4.7 Página de usuarios del sistema de soporte.*

En la figura 4.8, se muestra el formulario para la creación de un nuevo usuario del sistema de soporte. El usuario creador debe poseer los permisos correspondientes para poder crear nuevos usuarios.

ARSupport Dashboard Profile Users Patients César Javier ▾

Register New User

Role:	<input type="text" value="Doctor"/>
FirstName:	<input type="text" value="Alejandro"/>
LastName:	<input type="text" value="Castillo"/>
UserName:	<input type="text" value="alex"/>
E-Mail Address:	<input type="text" value="alexded@hotmail.com"/>
Password:	<input type="password" value="....."/>
Confirm Password:	<input type="password" value="....."/>

[Register New User](#)


© 2020 Copyright: ARSupport

*Figura 4.8 Formulario de nuevo usuario del sistema de soporte.*

La tercera sección del Dashboard de usuario son los pacientes del sistema. Al igual que la página de usuarios se carga una lista con todos los pacientes registrados del sistema. Sin embargo, según los permisos de un usuario se podrán realizar las siguientes acciones: crear nuevos pacientes, introducir información de los pacientes (historial clínico, exploración física, pruebas, radiografías e información general), modificar información de los pacientes, y eliminar pacientes. Los administradores y doctores podrán realizar todas las tareas anteriormente mencionadas. Las enfermeras podrán dar de alta pacientes y modificar su información, pero no podrán eliminar pacientes. Los estudiantes no podrán dar de alta pacientes, ni eliminarlos, ni introducir su información por primera vez, solo podrán modificar información de los pacientes ya existentes. En la figura 4.9, se muestra la página de pacientes del sistema de soporte. El contenido de las “acciones” por cada renglón de paciente se verán afectado dependiendo del tipo de usuario.

ARSupport Dashboard Profile Users Patients César Javier ▾

### Patients



New Patient

Search

#	Name	Gender	Room	Actions
1	Cesar Javier Flores Maldonado	Male	LNT0001	<div style="background-color: #34495e; color: white; padding: 2px; border-radius: 3px; display: inline-block;">Actions ▾</div> <ul style="list-style-type: none"> <li>Physical Exploration</li> <li style="background-color: #e6e6e6;">Medical Record</li> <li>Smac30</li> <li>X-rays</li> <li>Review</li> <li>Edit</li> <li>Delete</li> </ul>
2	Luis Felipe Martinez Fernandez	Male	LNT0001	
3	Miguel Angel Guillen Diaz de Leon	Male	LNT0001	
4	Juan Flores Corona	Male	LNT0001	
5	Maria Castillo Carmona	Female	LNT0001	

<
1
2
>

© 2020 Copyright: ARSupport

*Figura 4.9* Página de pacientes del sistema de soporte.

Cada una de las acciones disponibles para los pacientes consiste en un formulario en el cual se debe llenar la información del paciente. En la figura 4.10, se muestra el formulario para la exploración física de un paciente.

ARSupport Dashboard Profile Users Patients César Javier ▾

Edit Patient Physical Exploration

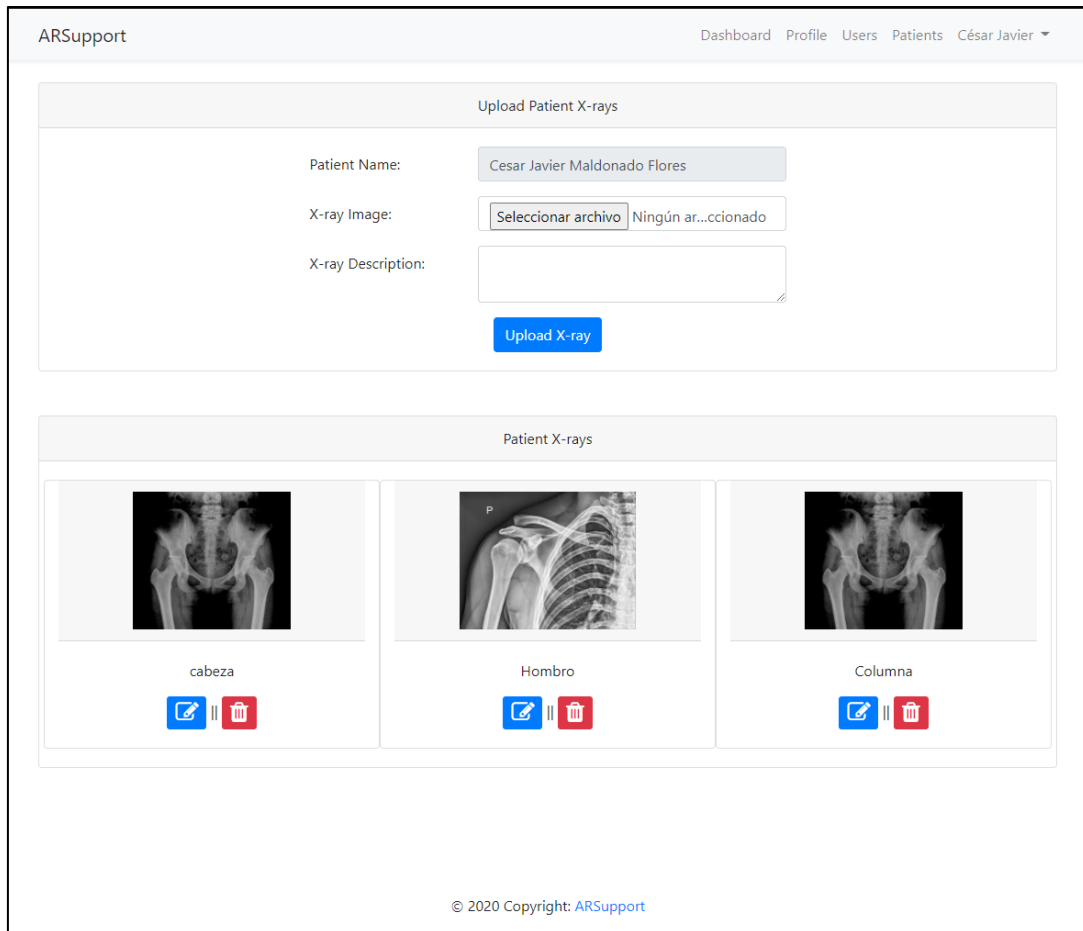
Habits:	<input type="text" value="Sentado 6 horas al día."/>
Stature:	<input type="text" value="1,78"/>
Weight:	<input type="text" value="140"/>
Glucose:	<input type="text" value="5,1"/>
Cholesterol:	<input type="text" value="4,2"/>
Hemoglobin:	<input type="text" value="32,1"/>
Body mass index:	<input type="text" value="45,6"/>
Body mass index status:	<input type="text" value="insuficiente"/>
Abdominal Perimeter:	<input type="text" value="63"/>
Head Circunference:	<input type="text" value="160"/>
Head:	<input type="text" value="normal"/>
Neck:	<input type="text" value="normal"/>
Chest:	<input type="text" value="normal"/>
Abdomen:	<input type="text" value="normal"/>
Extremities:	<input type="text" value="normal"/>
Genitals:	<input type="text" value="normal"/>
Skin:	<input type="text" value="moreno"/>

[Update Patient Physical Exploration](#)

© 2020 Copyright: ARSupport

*Figura 4.10 Formulario de Exploración física del paciente en el sistema de soporte.*

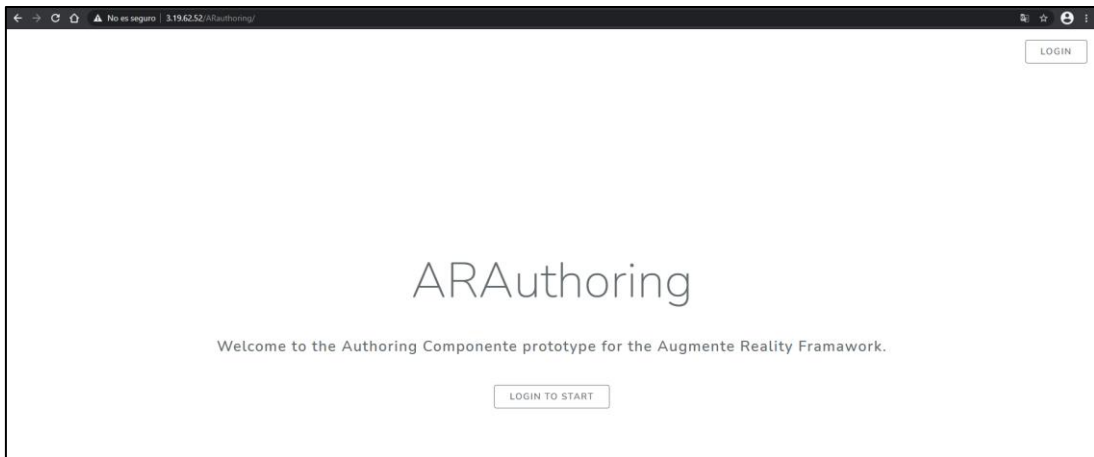
Una de las acciones importantes es la de radiografías del paciente. Para esta sección se diseñó una vista completa donde se pueden agregar nuevas radiografías y ver las radiografías ya existentes. Además, se agregó la funcionalidad de poder editar o modificar estas radiografías según el tipo de usuario. En la figura 4.11, se muestra la página de radiografías.



*Figura 4.11 Página de radiografías del paciente del sistema de soporte.*

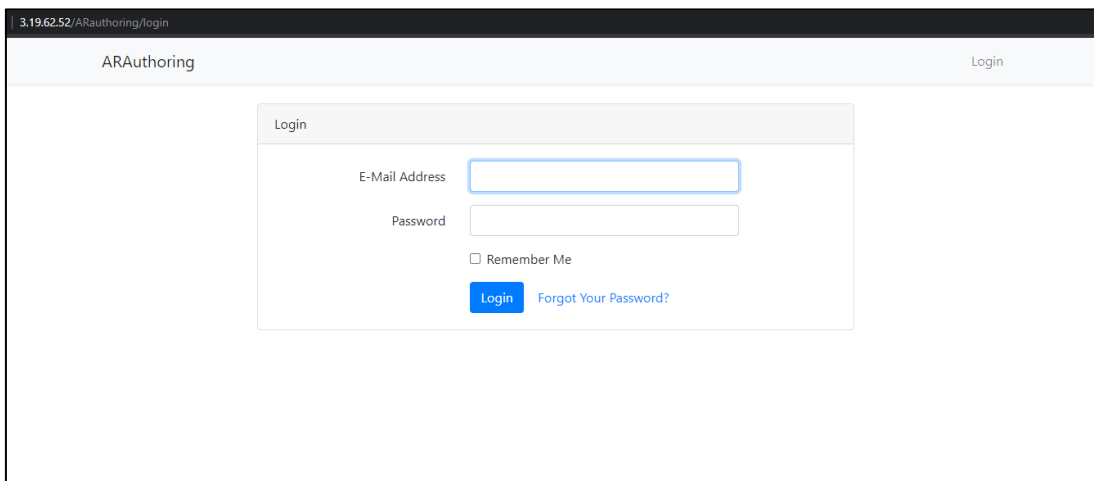
#### 4.2.3 Sistema de authoring

Las instancias EC2 son un medio sumamente útil para realizar implementaciones de tipo servidor como se mencionó en el sistema de almacenamiento. Por lo tanto, se decidió utilizar una instancia EC2 para el sistema de authoring. En la figura 4.12, se muestra la página de bienvenida del sistema de authoring. Al igual que soporte su función es la de presentar información introductoria a usuarios de internet u otro personal médico. En caso de que el sistema a implementar sea una herramienta interna del hospital o centro médico esta característica podría ser eliminada y permitir el acceso únicamente por ciertas redes o red interna.



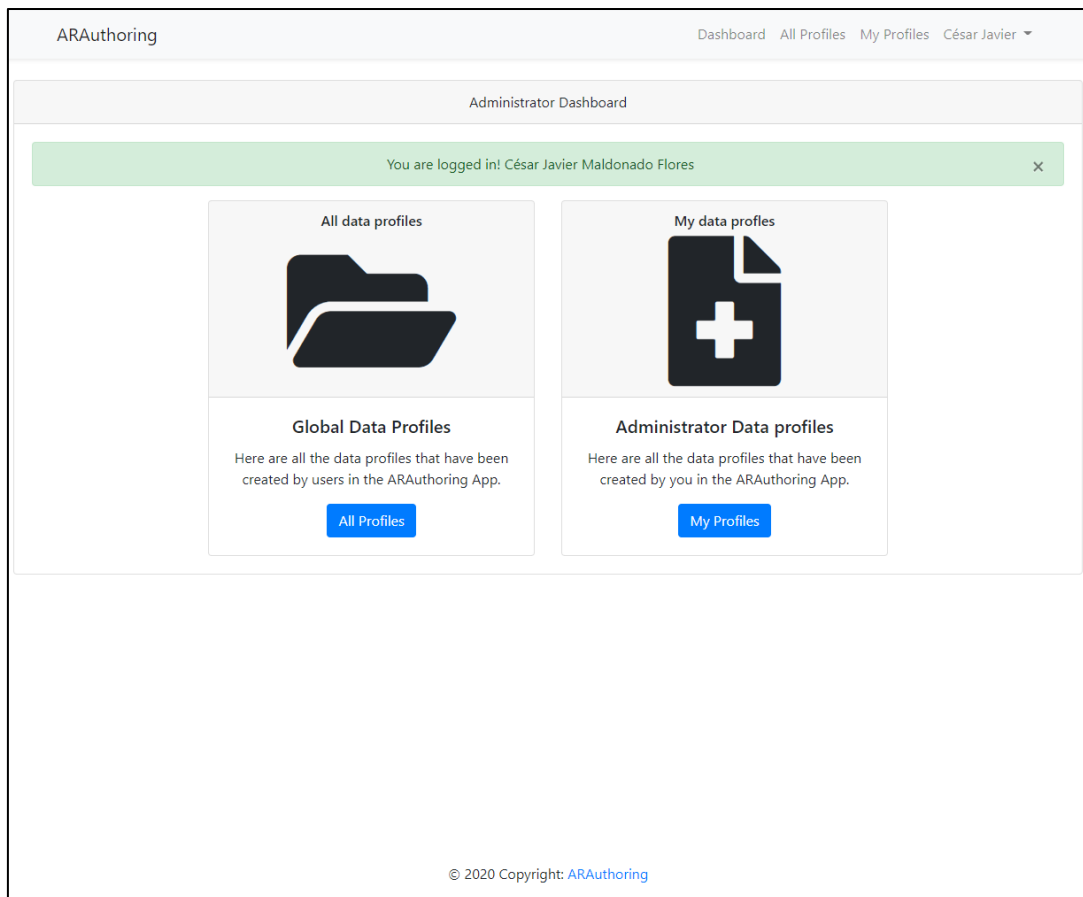
*Figura 4.12 Pagina de bienvenida del sistema de authoring.*

Para ingresar al sistema de authoring únicamente se da entrada a personal autorizado el cual también utilizara el sistema de RA. En la figura 4.13, se muestra la página de inicio de sesión del sistema de *authoring*.



*Figura 4.13 Página de inicio de sesión del sistema de authoring.*

Una vez el usuario se ha autenticado el sistema mostrará la página de Dashboard del usuario con un mensaje de bienvenida. En la figura 4.14, se muestra la página de Dashboard del sistema de authoring con las secciones “todos los perfiles” y “mis perfiles”.




*Figura 4.14 Página de Dashboard del usuario del sistema de authoring.*

En la sección de los perfiles de datos globales se le mostrará al usuario una página con el listado de todos los perfiles de datos públicos en el sistema de authoring. Estos perfiles pertenecen a todos los usuarios del sistema y únicamente ellos pueden editarlos. Sin embargo, cualquier usuario puede revisarlo y hacer una copia propia a su propia cuenta de usuario. En la figura 4.15, se muestra la página de los perfiles de datos globales. Es posible realizar búsquedas de perfiles por: nombre de perfil o nombre de propietario.

ARAuthoring Dashboard All Profiles My Profiles César Javier ▾

### All AR-Authoring Profiles



New Profile

Search

Search

#	Name	Upper left Frame	Upper right Frame	Lower left Frame	Lower right Frame	Actions
1	First Test	electro	general	smac	history	<a href="#" style="color: white; text-decoration: none;">View</a>
2	Second Test	renal	physical	history	smac	<a href="#" style="color: white; text-decoration: none;">View</a>
3	nextprofile	renal	lipid	electro	misc	<a href="#" style="color: white; text-decoration: none;">View</a>
4	without infder	renal	lipid	electro		<a href="#" style="color: white; text-decoration: none;">View</a>
5	El precioso	history	physical	general	electro	<a href="#" style="color: white; text-decoration: none;">View</a>

< 1 2 >

© 2020 Copyright: ARAuthoring

*Figura 4.15 Página de perfiles globales del sistema de authoring.*

Si un usuario abre un perfil que no le pertenece no podrá modificar el perfil. Sin embargo, si podrá crear una copia propia la cual podrá modificar. En la figura 4.16, se muestra la página de “revisión” del sistema de authoring en la cual existe el botón “*copy profile*” para hacer una copia del perfil seleccionado.

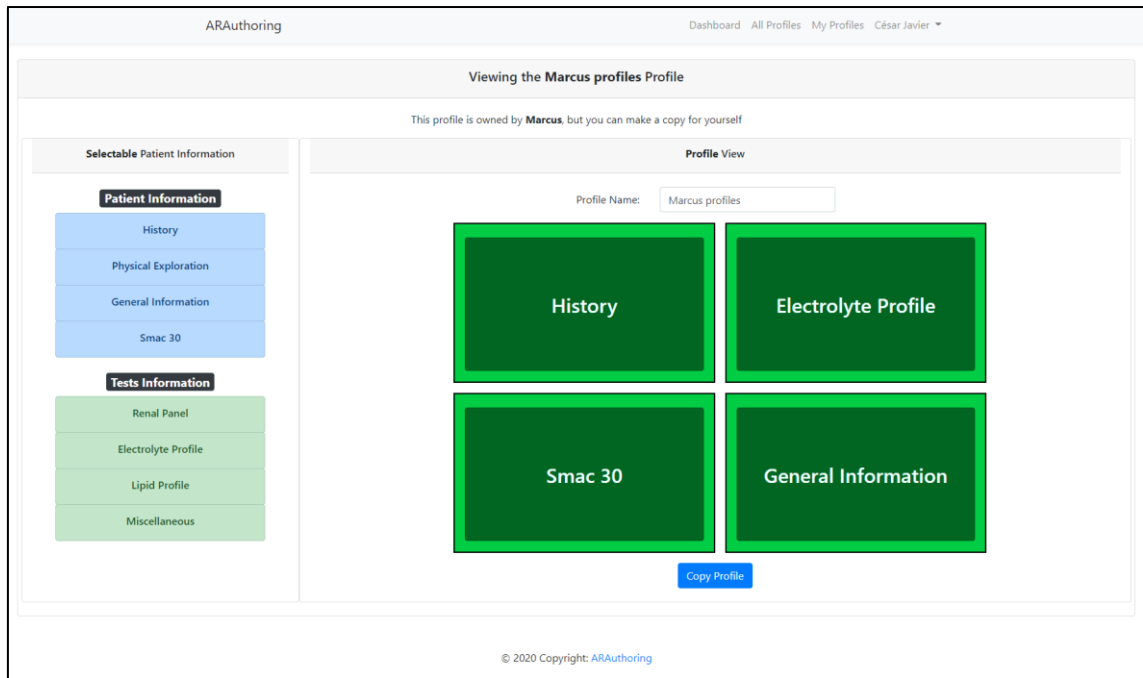


Figura 4.16 Página de revisión del perfil seleccionado en el sistema de authoring.

Los usuarios tienen la posibilidad de crear nuevos perfiles cuando lo deseen. La página en el sistema de authoring para la creación de estos perfiles se realizó utilizando código “json” el cual nos permitió desarrollar una página de creación de perfiles dinámica y fácil de usar. En la figura 4.17, se muestra la página de nuevo perfil del sistema de *authoring*.

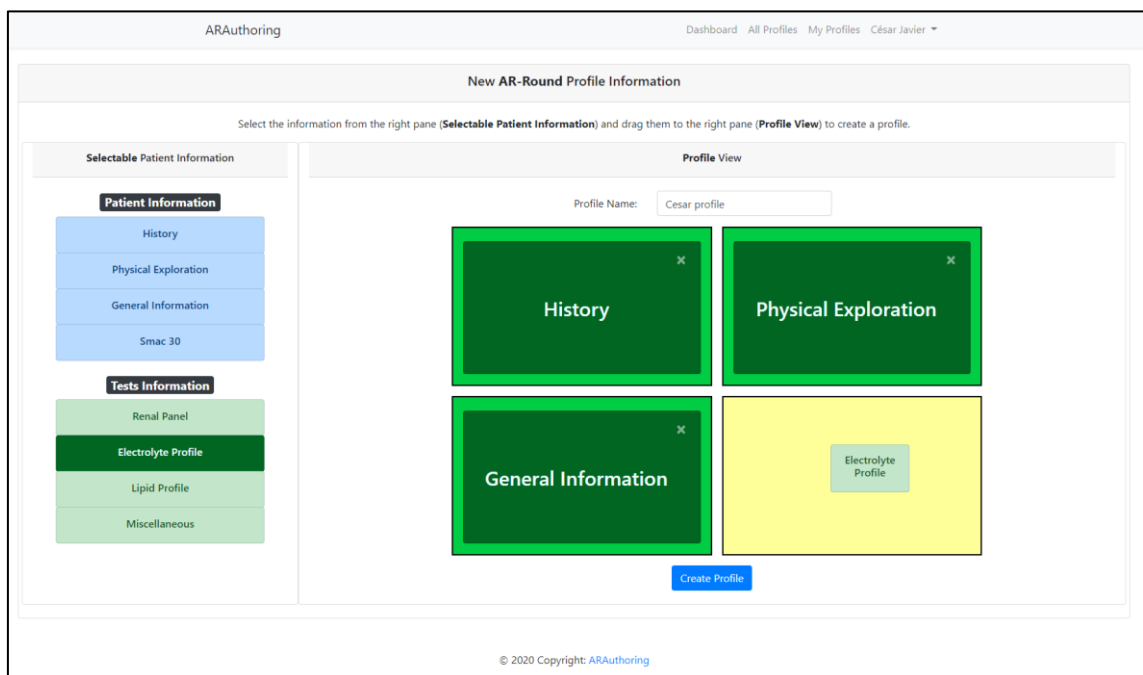


Figura 4.18 Nuevo perfil del sistema de authoring.

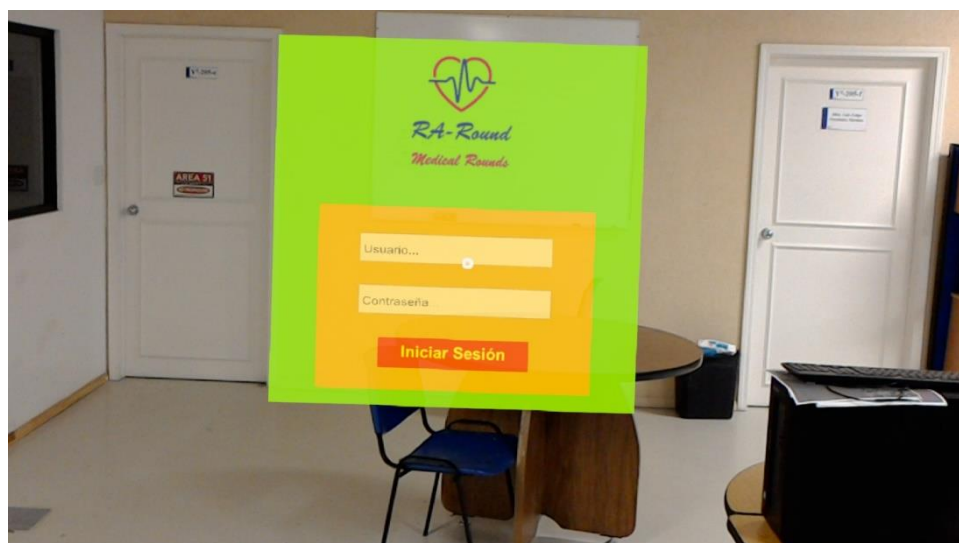
#### 4.2.4 Sistema de RA

Por último, el sistema de realidad aumentada fue desarrollada como una aplicación para el dispositivo de Microsoft HoloLens v1. Una vez comienza la ejecución del sistema de RA se muestra el logo de la aplicación “RA-Round” como se muestra en la figura 4.19.



*Figura 4.19 Logo de la aplicación de RA.*

El usuario debe iniciar sesión en el sistema de RA para continuar. En la figura 4.20, se muestra el formulario “tradicional” de inicio de sesión. El usuario debe introducir usuario y contraseña.



*Figura 4.20 Inicio de sesión del sistema de RA.*

Una vez el usuario se ha autenticado correctamente el sistema requiere que seleccione un paciente. Como se describió en el módulo de selección de paciente de “desarrollo” de la sección 3.4.5 existen tres maneras de realizar la búsqueda de pacientes. La primera es a través del escaneo de código QR. En caso de que sea un solo paciente el sistema cargara al paciente directamente. De tratarse de varios pacientes al usuario se le mostrará una lista para que pueda elegir al paciente. La segunda forma de búsqueda es escribiendo el nombre del paciente o código de cuarto. Y la tercera búsqueda es a través del reconocimiento de voz, el usuario podrá pronunciar el nombre del paciente. En la figura 4.21, se muestra el resultado de haber realizado una búsqueda con más de un resultado.

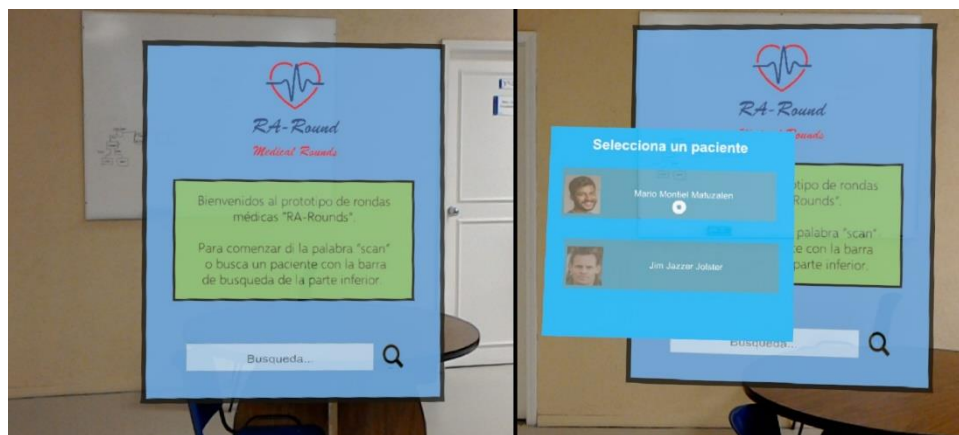


Figura 4.21 Resultados de búsqueda en el sistema de RA.

Tras haber seleccionado una paciente de la lista de resultados el sistema mostrará la pantalla principal de la vista de datos. En la figura 4.22, se puede observar los distintos elementos de esta sección. En la parte superior izquierda se encuentra el “pop-up” con la foto e información del paciente seleccionado. La cual puede ser ajustada a esa posición si así se requiere con el cuadro de selección debajo “ajuste/desajuste”. El siguiente cuadro de selección permite al usuario ocultar el marco en caso de que requiera ese espacio para observar mejor el mundo real. El panel izquierdo contiene todas y cada una de las secciones de información del paciente. Al seleccionar una de ellas el sistema lo mostrará en el “MainFrame” del lado derecho en forma de ventana de información.

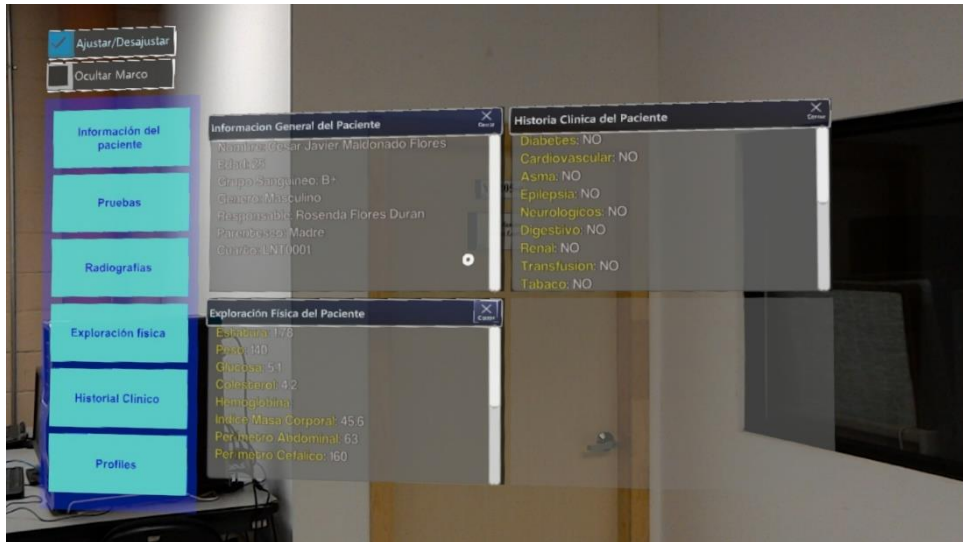


Figura 4.22 Vista de datos del sistema de RA.

Finalmente, en la sección “perfiles” al ser seleccionada se le mostrará al usuario una lista de todos los perfiles en su cuenta en el mismo panel izquierdo. Cada uno de estos perfiles cerrará cualquier ventana que haya sido abierta por el usuario y mostrará solo aquellas con información del perfil (del paciente seleccionado al inicio). De esta manera el usuario tendrá acceso a distintas clases de información a través de diferentes perfiles. En la figura 4.23, se muestra un perfil seleccionado en el sistema de RA.

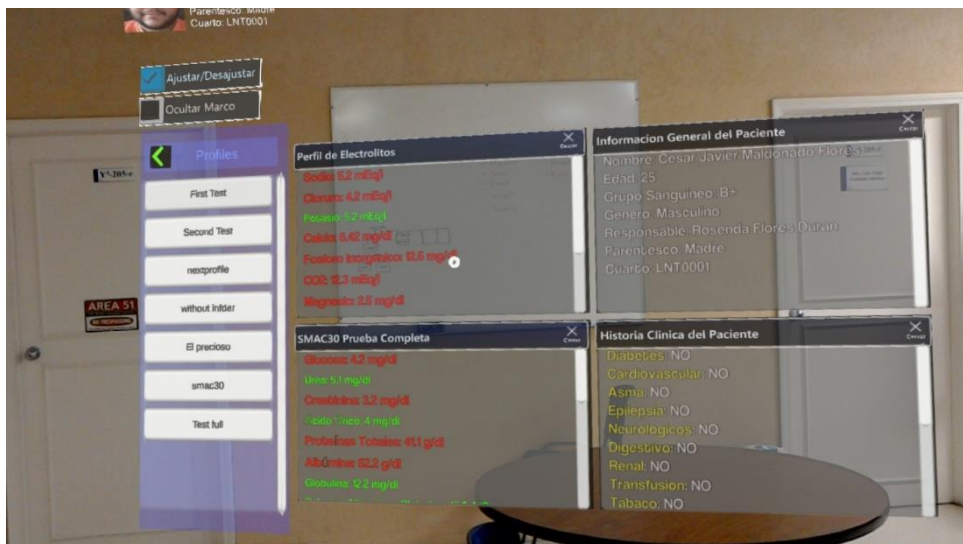


Figura 4.23 Perfil seleccionado del paciente en el sistema de RA.

## V. Conclusiones y trabajo futuro

En este trabajo de tesis presentamos un marco de trabajo enfocado a la colaboración o comunicación con un sistema de RA. Descubrimos que las estructuras, componentes y características varían mucho según los requisitos, el enfoque o las características para cada enfoque particular. El objetivo fue desarrollar un marco para las aplicaciones de RA enfocado a cualquier área de dominio (sin un área de dominio fijo) donde los datos y la información deban mostrarse a los usuarios a través de dispositivos de RA, para proporcionar al usuario rápidamente información que fuera fácil de ver. Como resultado, el marco propuesto consistió en un marco con cuatro componentes diferentes: almacenamiento, soporte, *authoring* y RA. Trabajando juntos para permitir al usuario administrar los datos, crear "perfiles" para el sistema de RA y mostrar los datos al usuario.

Una vez terminado el marco, creamos una instancia para probarlo. La instancia se centró en el área de dominio "Hospitales de México – Rondas Médicas". El sistema de RA se centró en ver los datos y la información del paciente durante las rondas médicas del personal del hospital. La instancia consistió en el diseño y desarrollo de 4 sistemas diferentes. El sistema de almacenamiento se encarga de almacenar los datos del usuario (personal médico), los pacientes y su información médica (historial clínico, exploración física, pruebas, radiografías e información general) y los perfiles del sistema de RA. El sistema de soporte se encarga de gestionar los usuarios y la información de dominio que serán utilizados por los demás sistemas. El sistema de *authoring* se encarga de gestionar y visualizar los perfiles del sistema de RA. El sistema de RA es responsable de mostrar los datos y la información del paciente a los usuarios autorizados.

En conclusión, segmentar el marco en cuatro componentes principales nos da solidez. Dado que las posibles áreas de dominio son amplias y variadas, los cuatro componentes nos permiten centrarnos en determinadas actividades y tareas en cada uno de ellos. El componente de almacenamiento no restringe ni limita las herramientas, el software o la implementación que se pueden utilizar. En este trabajo mostramos dos de estas implementaciones, localmente y en un entorno de nube pública. El componente de soporte forma los roles y permisos que los usuarios poseen y otorgan a otros usuarios, restringiendo el acceso a la información al personal no autorizado y permitiendo que solo los usuarios con ciertos roles y permisos accedan o modifiquen la información. El componente de *authoring* proporciona un entorno ideal para crear y administrar perfiles que mostrarán información en un orden o secuencia en el sistema de RA. Sin embargo,

según la información que deba mostrarse, este componente puede requerir características o comportamientos adicionales. Para este trabajo de tesis se ha propuesto un diseño simple de datos, como texto, imágenes y objetos 3d. El componente de RA es donde el usuario verá la información y los datos del área de dominio. El marco no especifica un dispositivo de RA que deba ser utilizado. Pero los módulos que debe contener se proponen utilizando los datos del componente de almacenamiento gestionados por los otros sistemas del framework.

La RA ha sido clasificada como una tecnología innovadora con un gran potencial para casi cualquier área de interacción humana. Se debe explorar y trabajar en su potencial para abordar intuitivamente las necesidades de los usuarios. En el futuro, este marco podría mejorarse para que el componente de RA pueda usarse en dispositivos diseñados principal o exclusivamente para la RA. También puede ser posible reformar el marco para cubrir aún más áreas de dominio con características específicas, agregando o modificando componentes.

En cuanto al trabajo futuro para los sistemas instanciados del marco de trabajo enfocados a las rondas médicas. Existen cambios o complementos que podríamos agregar para mejorar tanto la funcionalidad como la experiencia de usuario. El sistema de almacenamiento podría ser expandido en cuanto a diseño contenido en su base de datos. Al tratarse de un modelo relacional hay cosas que no podríamos hacer o que tendríamos que buscar una manera creativa de hacerlo. Aunque como se mencionó se diseñó de esta manera para que el usuario que lo implementara pudiera utilizar cualquier cosa que necesitara. Para el sistema de soporte, en este caso particular, requeriría un gran gasto de tiempo en la captura de datos. Por lo tanto, agregar una funcionalidad en la cual puedan escanearse documentos existentes y digitalizarlos para extraer la información requerida es otra propuesta. De esta manera se ahorraría tiempo y esfuerzo por parte del personal médico. El sistema de authoring y RA dependen en gran medida de lo que se necesite mostrar y por lo tanto de la información introducida por el sistema de soporte y almacenada por el sistema de almacenamiento. Incluir algún otro tipo de información requeriría cambiar un poco las especificaciones de los perfiles que se muestran en el sistema de RA. Por lo tanto, para este sistema se propone un rediseño de su estructura en el que se pueda mostrar una mayor cantidad de información sin sacrificar la experiencia del usuario.

## Referencias

- [1] A. Iftene y D. Trandabat, «Enhancing the Attractiveness of Learning through Augmented Reality,» *Procedia Computer Science*, vol. 129, pp. 166-175, 2018.
- [2] M. Billinghurst, A. Clark y G. Lee, «A Survey of Augmented Reality,» *Foundation and Trends in Human-Computer Interaction*, vol. 8, nº 2-3, pp. 73-272, 2014.
- [3] R. T. Azuma, «A Survey of Augmented Reality,» *Presence: Virtual and Augmented Reality*, vol. 6, nº 4, pp. 355-385, 1997.
- [4] E. P. Pajares Ortega, «Diseño de actividades didácticas con Realidad Aumentada,» *UNED*, pp. 1-204, 2015.
- [5] B. Kashif, K. Osman, E. Aiman y K. U. Samee, «Potentials, trends, and prospects in edge,» *Computer Networks*, vol. 130, pp. 94-120, 2018.
- [6] E. Quintero, P. Salinas y E. González, «Augmented Reality app for Calculus: A proposal for the Development of Spatial Visualization,» *Procedia Computer Science*, vol. 75, pp. 301-305, 2015.
- [7] K. K. SCARSI, M. A. FOTIS y G. A. NOSKIN, «Pharmacist participation in medical rounds reduces medication errors,» *Am J Health-Syst Pharm*, vol. 59, pp. 2089-2092, 2002.
- [8] D. M. Irby y L. Wilkerson, «Teaching Rounds: Teaching when time is limited,» *ResearchGate*, vol. 336, pp. 384-387, 2008.
- [9] I. Zendjebil, F.-E. Ababsa, J.-Y. Didier y M. Hachet, «Outdoor Augmented Reality: State of the Art and Issues,» de *10th ACM/IEEE Virtual Reality International Conference (VRIC 2008)*, Laval, France, 2008.
- [10] Oculus, «Oculus,» Facebook Technologies, [En línea]. Available: <https://www.oculus.com/>. [Último acceso: 10 Septiembre 2019].
- [11] G. Jason, «Library spaces and smart buildings: technology, metrics, and iterative design,» *Chicago, IL: ALA TechSource*, vol. 54, nº 1, pp. 3-33, 2018.
- [12] I. Fernández del Amo, J. Ahmet Erkoyuncu, R. Roy, R. Palmarini y D. Onoufriou, «A systematic review of Augmented Reality content-related techniques for knowledge transfer in maintenance applications,» *ELSEVIER - Computer in industry*, vol. 103, pp. 47-71, 2018.
- [13] F. Zheng, T. Whitted, A. Lastra, P. Lincoln, A. State, A. Maimone y H. Fuchs, «Minimizing Latency for Augmented Reality Displays: Frames Considered Harmful,» de *IEEE International Symposium on Mixed and Augmented Reality 2014*, Munich, 2014.
- [14] M. Haller, M. Billinghurst y B. Thomas, *Emerging Technologies of Augmented Reality Interfaces and Design*, Pennsylvania : Idea Group Inc., 2007.

- [15] M. Quandt, B. Knoke, C. Gorltd, M. Freitag y K. -. D. Thoben, «General Requirements for Industrial Augmented Reality Applications,» *Procedia CIRP*, vol. 72, nº 51, pp. 1130-1135, 2018.
- [16] P. Caricato, L. Colizzi, M. Gnani y A. Guerrieri, «Augmented reality applications in manufacturing: a multi-criteria decision model for performance analysis,» *The International Federation of Automatic Control*, vol. 19, pp. 24-29, 2014.
- [17] A. Nee y S. Ong, «Virtual and Augmented Reality Applications in Manufacturing,» *The International Federation of Automatic Control*, vol. 7, pp. 19-21, 2013.
- [18] J. Novak-Marcincina, J. Barnaa, M. Janaka y L. Novakova-Marcincinova, «Augmented Reality Aided Manufacturing,» *ScienceDirect*, vol. 25, pp. 23-31, 2013.
- [19] K. Lee, «Augmented Reality in Education and Training,» *University of Northern Colorado*, vol. 56, pp. 13-21, 2012.
- [20] M. Kesima y Y. Ozarslanb, «Augmented reality in education: current technologies and the potential for education,» *ScienceDirect*, vol. 47, pp. 297-302, 2012.
- [21] J. Shuhaiber, «Augmented Reality in Surgery,» *American Medical Association*, vol. 139, pp. 170-174, 2004.
- [22] T. Sielhorst, M. Feuerstein y N. Navab, «Advanced Medical Displays: A Literature Review of Augmented Reality,» *Journal of Display Technology*, vol. 4, pp. 451-467, 2008.
- [23] C. Lee y G. Kwok Chu Wong, «Virtual reality and augmented reality in the management of intracranial tumors: A review,» *Journal of Clinical Neuroscience*, vol. 62, pp. 14-20, 2018.
- [24] B. Holmes, «Overview of Dynamics 365 Guides - Dynamics 365 Mixed Reality | Microsoft Docs,» Microsoft, 03 February 2021. [En línea]. Available: <https://docs.microsoft.com/en-us/dynamics365/mixed-reality/guides/>. [Último acceso: 10 March 2021].
- [25] K. Kayano, «Microsoft Customer Story - Toyota simplifies complex repairs and accelerates training with mixed reality on HoloLens 2,» Microsoft, 20 April 2021. [En línea]. Available: <https://customers.microsoft.com/en-us/story/862482-toyota-motor-company-manufacturing-azure-holo-lens>. [Último acceso: 25 April 2021].
- [26] M. E. Porter y J. E. Heppelmann, «Why Every Organization Needs an Augmented Reality Strategy,» *Harvard Business Review*, vol. 95, nº 6, pp. 46-57, 2017.
- [27] P. Thibodeau, «Walmart, FedEx make case for VR training,» SearchHRSoftware, 10 Octubre 2019. [En línea]. Available: <https://searchhrsoftware.techtarget.com/news/252472139/Walmart-FedEx-make-case-for-VR-training>. [Último acceso: 20 Noviembre 2019].
- [28] R. Schoenfelder y D. Schmalstieg, «Augmented Reality for Industrial Building Acceptance,» de *IEEE Virtual Reality Conference*, Reno, NE, USA, 2008.

- [29] H. Regenbrecht, G. Baratoff y W. Wilke, «Augmented reality projects in the automotive and aerospace industries,» *IEEE computer graphics and applications*, vol. 25, nº 6, pp. 48-56, 2005.
- [30] Microsoft, «Mixed Reality Toolkit-Unity,» Microsoft, 21 Enero 2015. [En línea]. Available: <https://github.com/microsoft/MixedRealityToolkit-Unity>. [Último acceso: 15 Octubre 2019].
- [31] M. M. Kevin J. O’Leary, M. Ryan Buck, M. Helene M. Fligel, R. M. Corinne Haviley, M. R. C. Maureen E. Slade, M. Matthew P. Landler, M. Nita Kulkarni, M. M. Keiki Hinami, P. M. Jungwha Lee, M. Samuel E. Cohen y M. Mark V. Williams, «Structured Interdisciplinary Rounds in a Medical Teaching Unit,» *HEALTH CARE REFORM*, vol. 171, nº 7, pp. 678-684, 2011.
- [32] J. Ahmet Erkoyuncu, I. Fernaández del Amo y M. Dalle Mura, «Improving efficiency of industrial maintenance with context aware adaptive authoring in augmented reality,» *ScienceDirect*, vol. 66, nº 1, pp. 465-468, 2017.
- [33] J. Bacca, S. Baldiris, R. Fabregat y K. , «Framework for designing motivational augmented reality applications in vocational education and training,» *Australasian Journal of Educational Technology*, vol. 35, nº 3, pp. 102-117, 2019.
- [34] J. Cervantes Ojeda y M. d. C. Gómez Fuentes, «Taxonomía de los modelos y metodologías de desarrollo de software más utilizados,» *Redalyc*, vol. 52, pp. 37-47, 2012.
- [35] M. Stauffer, *Laravel Up & Running A Framework for Building Modern PHP Apps*, Sebastopol: O’Reilly Media, 2019.
- [36] H. Lasi y H.-G. Kemper, «Industry 4.0,» *Business & Information Systems Engineering*, vol. 4, pp. 239-242, 2014.
- [37] C. Echard, «Ensuring Software Integrity in IoT Devices,» *Journal of Information Technology & Software Engineering*, vol. 7, nº 5, pp. 1-3, 2017.
- [38] P. Fraga-Lamas, T. M. Fernández-Caramés, Ó. Blanco-Novoa y M. A. Vilar-Montesinos, «A Review on Industrial Augmented Reality Systems for the Industry 4.0 Shipyard,» *IEEE Access*, vol. 6, pp. 13358-13375, 2018.
- [39] D. Aschenbrenner, N. Maltry, J. Kimmel, M. Albert, J. Scharnagl y K. Schilling, «ARTab – using Virtual and Augmented Reality Methods for an improved Situation,» *ScienceDirect*, vol. 49, nº 30, pp. 204-209, 2016.
- [40] D. W. F. van Krevelen y R. Poelman, «A Survey of Augmented Reality Technologies, Applications and Limitations,» *The International Journal of Virtual Reality*, 2010.
- [41] L. H y H.-G. K, «Industry 4.0,» *Springer*, vol. 56, pp. 239-242, 2014.

## [Anexo A]

- a. La Máquina Virtual local cuenta con 8 Gb de memoria, 150 Gb de disco duro y 2 *vCPU*. Como sistema operativo se optó por Ubuntu server 18.04 LTS por ser una opción de código abierto el cual funciona con casi cualquier hardware o plataformas de virtualización. Además, es útil para una gran variedad de tareas como el hospedaje de sitio web, compartir archivos y contenedores. Las tecnologías utilizadas fueron PHP 7.4.2, Apache 2.4.41, *MariaDB* 10.4.11 y *phpMyAdmin* 5.0.1. Instaladas a través del *stack* LAMP de BITNAMI sobre el servidor Ubuntu.
- b. La instancia EC2 de AWS utilizada es la t2.micro la cual cuenta con 1 GiB de memoria, EBS de 30 GB (Almacenamiento), 1 *vCPU* de procesador. Como sistema operativo se utiliza Ubuntu server 18.04 LTS. Las tecnologías utilizadas fueron PHP 7.4.2, Apache 2.4.41, *MariaDB* 10.4.11 y *phpMyAdmin* 5.0.1.

## [Anexo B]

- a. Documento de especificación de requisitos (Sistema de Soporte).

---

# Especificación de requisitos de software

## Sistema de soporte

Autor: César Javier Maldonado Flores --- 183137

## 1. Introducción

El presente documento de especificación de requerimientos de software (SRS) del sistema a construir surge para ser una guía exploratoria en el sistema de soporte del marco de trabajo propuesto. También pretende ser de ayuda para entender los requisitos mínimos necesarios.

Los requisitos a continuación descritos han sido definidos y propuestos por el profesor Luis Felipe Fernández-Martínez y el estudiante César Javier Maldonado Flores. Se describirá en forma

detallada las interfaces de usuario, del sistema, del hardware y comunicaciones, así como de los requerimientos, atributos entre otros.

### 1.1. Propósito

- Permitir establecer las bases necesarias para comunicar y compartir la información del proyecto entre programadores, expertos u otro individuo ajeno al proyecto.
- Este documento pretende ser de ayuda para que otros desarrolladores ajenos al proyecto puedan entender exactamente las necesidades y requerimientos del experto del dominio con respecto al software.

### 1.2. Alcance

- Identificar el prototipo del sistema deseado. Prototipo del sistema de soporte para la gestión de usuarios, pacientes y perfiles del marco de trabajo.
- Objetivos
  - Desarrollar el prototipo del sistema de soporte para gestionar los recursos descritos en el marco de trabajo propuesto.
  - Realizar una aplicación web para el prototipo del sistema.
  - Ser la base de la gestión de usuarios y recursos de todos los sistemas del marco de trabajo.

### 1.3. Personal involucrado

Nombre	César Javier Maldonado Flores
Rol	Analista, Diseñador, Programador
Categoría Profesional	Programador
Responsabilidades	Analizar los requisitos del prototipo, diseño del prototipo, programación de los módulos del prototipo.
Información de contacto	<a href="mailto:al183137@alumnos.uacj.mx">al183137@alumnos.uacj.mx</a>

Nombre	Luis Felipe Fernández-Martínez
Rol	Mánager del proyecto
Categoría Profesional	*No participa en el desarrollo
Responsabilidades	Suministrar información relevante del proyecto
Información de contacto	<a href="mailto:lfernand@uacj.mx">lfernand@uacj.mx</a>

## 1.4. Definiciones, acrónimos y abreviaturas

### Definiciones

**Laravel:** Es un marco de trabajo de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y 7. Su filosofía es la creación de código PHP de forma elegante y simple.

**PHP:** Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

**JavaScript:** Lenguaje de programación interpretado, se define como orientado a objetos.

**jQuery:** Es una librería multiplataforma de JavaScript, la cual permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX.

**AJAX:** Técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente mientras se mantiene la comunicación asincrónica con el servidor en segundo plano.

**Interfaz gráfica:** Medio que permite la comunicación entre el usuario y el sistema.

**Prototipo:** Ejemplar o primer molde que se realiza de un producto, objeto o software.

### Acrónimos.

Ninguno.

### Abreviaturas.

**HW:** HardWare

**SW:** SoftWare

**Sr.** Señor

**Sra.** Señora

**Ing.** Ingeniero(a)

**Dr.** Doctor(a)

## 1.5. Resumen

El documento de especificación de requisitos de software está compuesto de la siguiente manera.

- **Introducción:** En esta sección se detallan los objetivos que tiene el documento y de nuestro prototipo en forma general.
- **Descripción General:** Describe una perspectiva general del prototipo a desarrollarse, como también las características del usuario y las limitaciones que podría tener.

- **Requerimientos Específicos:** Muestra paso a paso todos los requerimientos que el usuario desea en el producto final. Para el cual se ha utilizado el “Prototipo 2 del Estándar IEEE 380”.

## 2. Descripción general

### 2.1. Perspectiva del producto

El prototipo que se va a desarrollar es independiente, y tendrá un diseño modular para la gestión de las diferentes áreas dentro del mismo.

### 2.2. Características de los usuarios

Tipo de usuario	Cualquier persona con autorización y que pueda usar una computadora o dispositivo con navegador web y acceso a internet.
Formación	Breve información de uso y objetivos de la aplicación.
Habilidades	Hacer uso de páginas o portales web.
Actividades	Acceder a la aplicación y usar las funcionalidades establecidas.

**\* Cualquier persona que cumpla con las condiciones anteriormente listadas podrá hacer uso del prototipo a desarrollar.**

### 2.3. Restricciones

El prototipo será desarrollado utilizando el marco de trabajo Laravel en el lenguaje de programación PHP, orientado a objetos. El sistema requerirá del sistema de almacenamiento para guardar y obtener información. La metodología para el desarrollo estará basada en cascadas por ser una de las metodologías más usadas para el desarrollo de software a pequeña y mediana escala.

### 2.4. Evolución previsible del sistema

Implementación de funcionalidades adicionales para una captura más amplia de información. Implementación de *frameworks* o *apis* libres para la expansión y mejoramiento de las características útiles del prototipo.

## 3. Requisitos específicos

R1: El prototipo debe permitir el acceso a todo personal autorizado a través de un navegador web en cualquier dispositivo conectado a internet. Por medio de usuario y contraseña.

R2: El prototipo permitirá la gestión de usuarios del sistema (Personal autorizado con diferentes niveles de autorización).

R3: El prototipo permitirá la gestión del perfil del usuario.

R4: El sistema permitirá la gestión de los pacientes del hospital, clínica, centro médico, etc.

R5: El prototipo permitirá acceso únicamente a personal autorizado de la institución médica a la que pertenezca.

### **3.1. Requisitos comunes de las interfaces**

#### **3.1.1. Interfaces de usuario**

Para la interfaz de usuario se utilizará un modelo simple compuesto de formularios para guardar y obtener información del usuario del prototipo, otros usuarios del prototipo, y pacientes de la institución médica. El prototipo será “responsivo” por lo tanto, podrá ser utilizado en una amplia variedad de dispositivos.

#### **3.1.2. Interfaces de hardware**

**Dispositivo con acceso a internet y navegador web (Computadora, Smartphone, Tablet, etc.):** El prototipo será desarrollado para cualquier dispositivo conectado a internet con un navegador web para visualizarlo.

#### **3.1.3. Interfaces de software**

Navegador web.

### **3.2. Requisitos no funcionales**

#### **3.2.1. Requisitos de rendimiento**

El prototipo deberá ser ejecutado de manera fluida con los recursos de la nube pública de AWS. En una instancia de máquina virtual Ubuntu LTS 16.04. Además, los datos deberán ser guardados y obtenidos del sistema de almacenamiento.

#### **3.2.2. Seguridad**

La seguridad del sistema es por:

- El sistema estará hospedado en una instancia EC2 de AWS.
- El prototipo únicamente podrá ser accedido por personal autorizado a través de usuario y contraseña.
- La información de los usuarios y pacientes estará respaldada en el sistema de almacenamiento el cual estará en otra instancia EC2 de AWS.

### **3.2.3. Disponibilidad**

El prototipo será desarrollado tomando en cuenta las especificaciones, requerimientos, reglas y objetivos del experto del dominio. La accesibilidad dependerá del servicio de proveedor de nube. El cual tiene un alto índice de accesibilidad.

### **3.2.4. Mantenibilidad**

No es posible realizar mantenimiento directo en la instancia EC2 donde se encontrará hospedado. Sin embargo, es posible realizar ajustes y actualizaciones y después actualizar la versión en EC2.

### **3.2.5. Portabilidad**

El prototipo consiste en una aplicación web “responsiva” realizada utilizando el framework Laravel, por lo tanto, podrá ser migrada a cualquier otro proveedor de nube o incluso a sistemas locales de ser necesario. Al ser una aplicación web responsiva podrá ser accedido a través de cualquier dispositivo con navegador web y acceso a internet.

- b. Documento de especificación de requisitos (Sistema de Authoring).

---

## **Especificación de requisitos de software**

### **Sistema de authoring**

Autor: César Javier Maldonado Flores --- 183137

## **1. Introducción**

El presente documento de especificación de requerimientos de software (SRS) del sistema a construir surge para ser una guía exploratoria en el sistema de authoring (segundo sistema) del marco de trabajo propuesto. También pretende ser de ayuda para entender los requisitos mínimos necesarios tanto funcionales como no funcionales para el futuro desarrollo del sistema (prototipo).

Los requisitos a continuación descritos han sido definidos y propuestos por el profesor Luis Felipe Fernández-Martínez y el estudiante César Javier Maldonado Flores. Se describirá en forma detallada las interfaces de usuario, de sistema, de hardware y comunicaciones, así como de los requerimientos, atributos, entre otros.

### 1.1. Propósito

- Permitir establecer las bases necesarias para comunicar y compartir la información del proyecto entre programadores, expertos u otro individuo ajeno al proyecto.
- Este documento pretende ser de ayuda para que otros desarrolladores ajenos al proyecto puedan entender exactamente las necesidades y requerimientos del experto del dominio con respecto al software.

### 1.2. Alcance

- Identificar el prototipo del sistema deseado. Prototipo del sistema de authoring para la gestión los perfiles utilizados por el sistema de realidad aumentada descrito en el marco de trabajo.
- Objetivos:
  - Desarrollar el prototipo del sistema de authoring para gestionar los perfiles creados por los usuarios (usuarios gestionados en soporte) descritos en el marco de trabajo propuesto.
  - Realizar una aplicación web para el prototipo del sistema.
  - Implementar la aplicación como el principal medio para el diseño y creación de perfiles para el sistema de RA.

### 1.3. Personal involucrado

Nombre	César Javier Maldonado Flores
Rol	Analista, Diseñador, Programador
Categoría Profesional	Programador
Responsabilidades	Analizar los requisitos del prototipo, diseño del prototipo, programación de los módulos del prototipo.
Información de contacto	<a href="mailto:al183137@alumnos.uacj.mx">al183137@alumnos.uacj.mx</a>

Nombre	Luis Felipe Fernández-Martínez
Rol	Mánager del proyecto
Categoría Profesional	*No participa en el desarrollo (codificación)
Responsabilidades	Suministrar información relevante del proyecto
Información de contacto	<a href="mailto:lfernand@uacj.mx">lfernand@uacj.mx</a>

## 1.4. Definiciones, acrónimos y abreviaturas

### Definiciones

**Laravel:** Es un marco de trabajo de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y 7. Su filosofía es la creación de código PHP de forma elegante y simple.

**PHP:** Es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML.

**JavaScript:** Lenguaje de programación interpretado, se define como orientado a objetos.

**jQuery:** Es una librería multiplataforma de JavaScript, la cual permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX.

**AJAX:** Técnica de desarrollo web para crear aplicaciones interactivas o RIA (*Rich Internet Applications*). Estas aplicaciones se ejecutan en el cliente mientras se mantiene la comunicación asincrónica con el servidor en segundo plano.

**Interfaz gráfica:** Medio que permite la comunicación entre el usuario y el sistema.

**Prototipo:** Ejemplar o primer molde que se realiza de un producto, objeto o software.

**Perfil (para Aplicación de RA):** Estructura formada por cuatro partes las cuales forman un marco utilizado en la aplicación de RA del marco de trabajo. Puede contener distintas formas de información dependiendo de los datos proporcionados por el área del dominio enfocado.

### Acrónimos.

Ninguno.

### Abreviaturas.

**HW:** HardWare

**SW:** SoftWare

**Sr.** Señor

Sra. Señora

Ing. Ingeniero(a)

Dr. Doctor(a)

### 1.5. Resumen

El documento de especificación de requisitos de software está compuesto de la siguiente manera.

- Introducción: En esta sección se detallan los objetivos que tiene el documento y de nuestro prototipo en forma general.
- Descripción General: Describe una perspectiva general del prototipo a desarrollarse, como también las características del usuario y las limitaciones que podría tener.
- Requerimientos Específicos: Muestra paso a paso todos los requerimientos que el usuario desea en el producto final. Para el cual se ha utilizado el “Prototipo 2 del Estándar IEEE 380”.

## 2. Descripción general

### 2.1. Perspectiva del producto

El prototipo que se va a desarrollar es independiente, y tendrá un diseño modular para la gestión de las diferentes áreas dentro del mismo.

### 2.2. Características de los usuarios

Tipo de usuario	Cualquier persona previamente registrada en el sistema de soporte con una computadora o dispositivo con navegador web y acceso a internet.
Formación	Breve información de uso y objetivos de la aplicación.
Habilidades	Hacer uso de páginas o portales web.
Actividades	Acceder a la aplicación y usar las funcionalidades establecidas.

**\* Cualquier persona que cumpla con las condiciones anteriormente listadas podrá hacer uso del prototipo a desarrollar.**

### 2.3. Restricciones

El prototipo será desarrollado utilizando el marco de trabajo Laravel en el lenguaje de programación PHP, orientado a objetos. El prototipo requerirá del sistema de almacenamiento para guardar y obtener información. Además, será necesario que el sistema de soporte esté en funcionamiento para generar los usuarios del sistema de authoring. La metodología para el

desarrollo estará basada en cascadas por ser una de las metodologías más usadas para el desarrollo de software a pequeña y mediana escala.

## **2.4. Evolución previsible del sistema**

Implementación de módulos adicionales para el diseño y creación de perfiles con nuevas características o adiciones de información. Implementación de *frameworks* o *apis* libres para la expansión y mejoramiento de las características útiles del prototipo.

# **3. Requisitos específicos**

R1: Los usuarios de prototipo deberán ser dados de alta en el sistema de soporte.

R2: El prototipo debe permitir el acceso a todo personal autorizado a través de un navegador web en cualquier dispositivo conectado a internet. Por medio de usuario y contraseña.

R3: El prototipo permitirá la gestión de perfiles para el sistema de realidad aumentada.

R4: El sistema permitirá revisar y copiar el perfil de otro usuario del sistema de authoring.

R5: El prototipo permitirá acceso únicamente a personal autorizado que utilizará el sistema de realidad aumentado.

## **3.1. Requisitos comunes de las interfaces**

### **3.1.1. Interfaces de usuario**

Para la interfaz de usuario se utilizará un modelo simple compuesto de objetos *jquery* los cuales permitirán crear los perfiles de manera dinámica y sencilla en el navegador web. De esta manera cualquiera podrá utilizar la aplicación sin necesidad de tener conocimientos previos o algún tipo de entrenamiento, ya que será muy intuitiva. El prototipo será “responsivo” por lo tanto, podrá ser utilizado en una amplia variedad de dispositivos.

### **3.1.2. Interfaces de hardware**

**Dispositivo con acceso a internet y navegador web (Computadora, Smartphone, Tablet, etc.):** El prototipo será desarrollado para cualquier dispositivo conectado a internet con un navegador web para visualizarlo.

### **3.1.3. Interfaces de software**

Navegador web.

## **3.2. Requisitos no funcionales**

### **3.2.1. Requisitos de rendimiento**

El prototipo deberá ser ejecutado de manera fluida con los recursos de la nube pública de AWS. En una instancia de máquina virtual Ubuntu LTS 16.04. Además, los datos deberán ser guardados y obtenidos del sistema de almacenamiento.

### **3.2.2. Seguridad**

La seguridad del sistema es por:

- El sistema estará hospedado en una instancia EC2 de AWS.
- El prototipo únicamente podrá ser accedido por personal autorizado a través de usuario y contraseña.
- La información de los usuarios y pacientes estará respaldada en el sistema de almacenamiento el cual estará en otra instancia EC2 de AWS.

### **3.2.3. Disponibilidad**

El prototipo será desarrollado tomando en cuenta las especificaciones, requerimientos, reglas y objetivos del experto del dominio. La accesibilidad dependerá del servicio de proveedor de nube. El cual tiene un alto índice de accesibilidad.

### **3.2.4. Mantenibilidad**

No es posible realizar mantenimiento directo en la instancia EC2 donde se encontrará hospedado. Sin embargo, es posible realizar ajustes y actualizaciones y después actualizar la versión en EC2.

### **3.2.5. Portabilidad**

El prototipo consiste en una aplicación web “responsiva” realizada utilizando el framework Laravel, por lo tanto, podrá ser migrada a cualquier otro proveedor de nube o incluso a sistemas locales de ser necesario. Al ser una aplicación web responsiva podrá ser accedido a través de cualquier dispositivo con navegador web y acceso a internet.

- c. Documento de especificación de requisitos (Sistema de RA).

---

# **Especificación de requisitos de software**

## **Sistema de Realidad Aumentada**

Autor: César Javier Maldonado Flores --- 183137

### **1. Introducción**

El presente documento de especificación de requerimientos de software (SRS) del sistema a construir surge para ser una guía exploratoria en el sistema de realidad aumentada del marco de trabajo propuesto. También pretende ser de ayuda para entender los requisitos mínimos necesarios.

Los requisitos a continuación descritos han sido definidos y propuestos por el profesor Luis Felipe Fernández-Martínez y el estudiante César Javier Maldonado Flores. Se describirá en forma detallada las interfaces de usuario, del sistema, del hardware y comunicaciones, así como de los requerimientos, atributos entre otros.

#### **1.1. Propósito**

A continuación, se describe el propósito de este documento de requerimientos de software.

- El principal propósito de este documento de requerimientos de software consiste en asentar las bases necesarias para comunicar y compartir la información del proyecto entre programadores, expertos u otro individuo ajeno al proyecto.
- Este documento pretende ser de ayuda para que otros agentes externos del proyecto entender y familiarizarse fácil y rápidamente con las necesidades y requerimientos del experto del dominio con respecto al software a desarrollar.

#### **1.2. Alcance**

A continuación, se listan el alcance del sistema prototipo a desarrollar y sus objetivos.

- Identificar los requerimientos y necesidades para prototipo del sistema deseado. El prototipo del sistema de realidad aumentada para personal médico durante las rondas médicas y la vista de perfiles de información diseñados específicamente para el sistema.
- Objetivos

- Desarrollar el prototipo del sistema de realidad aumentada para personal médico durante las rondas médicas.
- Realizar una aplicación para el dispositivo de realidad aumentada HoloLens.
- Mostrar los perfiles de información creados en el sistema de authoring específicamente diseñados para este sistema.

### 1.3. Personal involucrado

Nombre	César Javier Maldonado Flores
Rol	Analista, Diseñador, Programador
Categoría Profesional	Programador
Responsabilidades	Analizar los requisitos del prototipo, diseño del prototipo, programación de los módulos del prototipo.
Información de contacto	<a href="mailto:al183137@alumnos.uacj.mx">al183137@alumnos.uacj.mx</a>

Nombre	Luis Felipe Fernández-Martínez
Rol	Mánager del proyecto
Categoría Profesional	*No participa en el desarrollo
Responsabilidades	Suministrar información relevante del proyecto
Información de contacto	<a href="mailto:lfernand@uacj.mx">lfernand@uacj.mx</a>

### 1.4. Definiciones, acrónimos y abreviaturas

#### Definiciones

**HoloLens:** Dispositivo de realidad mixta independiente que ejecuta Windows 10. Redefine la información personal a través de experiencias holográficas. Combina ópticas y sensores de última generación para ofrecer hologramas 3D anclados al mundo real que nos rodea.

**C#:** Lenguaje de programación multiparadigma desarrollado y estandarizado por la empresa Microsoft como parte de su plataforma .NET, diseñado para la infraestructura de lenguaje común.

**Unity:** Es un motor de videojuegos multiplataforma, disponible como plataforma de desarrollo para Windows, MacOS y Linux.

**Interfaz gráfica:** Medio que permite la comunicación entre el usuario y el sistema.

**Prototipo:** Ejemplar o primer molde que se realiza de un producto, objeto o software.

**Perfil authoring:** Conjunto de información creado y diseñado en el sistema de authoring para ser utilizado exclusivamente por el sistema de realidad aumentada. Muestra información selecta sobre un paciente durante las rondas médicas.

## **Acrónimos.**

Ninguno.

## **Abreviaturas.**

**HW:** HardWare

**SW:** SoftWare

**Sr.** Señor

**Sra.** Señora

**Ing.** Ingeniero(a)

**Dr.** Doctor(a)

### **1.5. Resumen**

El documento de especificación de requisitos de software está compuesto de la siguiente manera.

- **Introducción:** En esta sección se detallan los objetivos que tiene el documento y de nuestro prototipo en forma general.
- **Descripción General:** Describe una perspectiva general del prototipo a desarrollarse, como también las características del usuario y las limitaciones que podría tener.
- **Requerimientos Específicos:** Muestra paso a paso todos los requerimientos que el usuario desea en el producto final. Para el cual se ha utilizado el “Prototipo 2 del Estándar IEEE 380”.

## **2. Descripción general**

### **2.1.Perspectiva del producto**

El prototipo que se va a desarrollar es independiente, y tendrá un diseño modular para la gestión de las diferentes áreas dentro del mismo.

### **2.2. Características de los usuarios**

Tipo de usuario	Cualquier persona con autorización, principalmente personal médico, y con el dispositivo HoloLens conectado a internet con una copia instalada del sistema de RA.
Formación	Breve tutorial en el uso de “gestos” para los HoloLens. Además, del propósito y funcionalidades de sistema de RA.
Habilidades	Ninguna en particular.

Actividades	Utilizar el dispositivo HoloLens, acceder al sistema de rondas médicas y utilizar sus funcionalidades.
-------------	--

**\* Cualquier persona que cumpla con las condiciones anteriormente listadas podrá hacer uso del prototipo a desarrollar.**

### 2.3. Restricciones

El prototipo será desarrollado utilizando el motor de videojuegos Unity en el lenguaje de programación C#, orientado a objetos. El sistema requerirá del sistema de almacenamiento para guardar y obtener información. Los perfiles utilizados por el sistema deberán estar registrados para el usuario del dispositivo. La metodología para el desarrollo estará basada en cascadas por ser una de las metodologías más usadas para el desarrollo de software a pequeña y mediana escala.

### 2.4. Evolución futura del sistema

Implementación de funcionalidades adicionales útiles para el personal médico durante una ronda médica. Ampliación de las características de los perfiles para una mejor y selecta información de los pacientes. Implementación de *apis* o software externo para agregar funcionalidades útiles al sistema como mapa del hospital o clínica, captura de información a través del dispositivo HoloLens, etc.

## 3. Requisitos específicos

R1: Módulo de “*login*”: El prototipo debe permitir el acceso a todo personal autorizado a través de sus credenciales de usuario. Los usuarios no podrán ser registrados en la aplicación de RA, únicamente a través del gestor de usuarios, sistema de soporte.

R2: El prototipo de RA permitirá realizar búsquedas por distintos medios: reconocimiento de QR, texto introducido a través de gestos HoloLens y por voz (nombre de paciente).

R3: Una pequeña ventana “pop-up” mostrará una pequeña porción de información del paciente seleccionado. Seguirá la visión del usuario sin obstruirla.

R4: Herramienta visual para mostrar la información del usuario. Consistirá en un recuadro o panel de 4 secciones donde podrán mostrarse 4 diferentes “ventanas” de información.

R5: Opción para mostrar y ocultar la información del recuadro principal y el “pop-up con la información del paciente”

R6: Panel lateral con las diferentes secciones de información correspondiente al paciente. El usuario las seleccionará y deberán ser mostradas en el recuadro de visualización.

R7: Vista de los perfiles del usuario. Se mostrará la información del paciente de acuerdo con el perfil seleccionado por el usuario.

### **3.1. Requisitos comunes de las interfaces**

#### **3.1.1. Interfaces de usuario**

Para la interfaz de usuario se utilizará un modelo simple compuesto de formularios para guardar y obtener información del usuario del prototipo, otros usuarios del prototipo, y pacientes de la institución médica. El prototipo será “responsivo” por lo tanto, podrá ser utilizado en una amplia variedad de dispositivos.

#### **3.1.2. Interfaces de hardware**

**Dispositivo HoloLens con acceso a internet:** El prototipo será desarrollado para el uso exclusivo del dispositivo de realidad aumentada HoloLens con conexión a internet.

#### **3.1.3. Interfaces de software**

Sistema Operativo Windows 10 en el dispositivo HoloLens.

### **3.2. Requisitos no funcionales**

Ninguno.

#### **3.2.1. Requisitos de rendimiento**

El prototipo deberá ser ejecutado de manera fluida con los recursos proporcionados por el dispositivo de realidad aumentada. Además, los datos utilizados deberán ser obtenidos del sistema de almacenamiento.

#### **3.2.2. Seguridad**

La seguridad del sistema es por:

- El sistema estará compilado por Visual Studio.
- El sistema estará instalado en el dispositivo HoloLens.

- El prototipo únicamente podrá ser accedido por personal autorizado a través de usuario y contraseña.
- La información de los usuarios y pacientes estará respaldada en el sistema de almacenamiento el cual estará en una instancia EC2 de AWS.

### **3.2.3. Disponibilidad**

El prototipo será desarrollado tomando en cuenta las especificaciones, requerimientos, reglas y objetivos del experto del dominio. La accesibilidad dependerá del servicio de proveedor de nube (El sistema se comunica constantemente con el sistema de almacenamiento). El cual tiene un alto índice de accesibilidad.

### **3.2.4. Mantenibilidad**

Desafortunadamente HoloLens 1 es aún un dispositivo de pruebas y no es posible realizar instalaciones remotas o por red. Por lo tanto, es necesario realizar actualizaciones o cambios conectando el dispositivo a una computadora.

### **3.2.5. Portabilidad**

El dispositivo HoloLens puede ser utilizado en cualquier parte en la que pueda estar una computadora. Posee una batería interna de media capacidad que le permite al usuario su uso constante por varias horas. Sin embargo, la aplicación solo puede ser utilizada entre dispositivos HoloLens y no otros como computadoras o teléfonos móviles.

[Anexo C]

- a. Documento de casos de uso (Sistema de Soporte).

---

**Casos de uso detallados**

**Sistema de soporte**

Autor: César Javier Maldonado Flores --- 183137

Tabla C.1 Identificación de usuario a través de página de inicio de sesión

<b>Caso de uso C1: Identificación de usuario a través de página de inicio de sesión.</b> Los usuarios autorizados del sistema podrán acceder a través de una página de inicio la cual pedirá nombre de usuario y contraseña como credenciales de acceso. De ser correcto se le permitirá el acceso al sistema de lo contrario se mostrará un mensaje de error en nombre de usuario o contraseña.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Acceder a la página del prototipo de soporte.	2: Mostrar la página de inicio de sesión.
3: Introducir el nombre de usuario (email), contraseña y presionar el botón de “iniciar sesión”.	4: Comprobar las credenciales de usuario y dar acceso al sistema. FA1
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Las credenciales son incorrectas, se muestra la página de inicio de sesión con un mensaje de “nombre de usuario o contraseña incorrecta”.	

Tabla C.2 Dashboard de usuario.

<b>Caso de uso C2: Dashboard de usuario.</b> Los usuarios dentro del sistema serán presentados con un <i>Dashboard</i> dependiendo de su nivel de autorización en el sistema (Administradores, doctores, enfermeras (os) y estudiantes (otros)).	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Iniciar sesión en el sistema de soporte.	2: Mostrar el Dashboard del usuario según su nivel de autorización.
<b>Extensivos (Flujos alternativos)</b>	
Ninguno	

Tabla C.3 Gestión de los datos de perfil del usuario.

<b>Caso de uso C3: Gestión de los datos de perfil del usuario.</b> Los usuarios dentro del sistema tendrán la posibilidad de editar la información en sus perfiles.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón de “ <i>Profile</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de gestión del perfil en la cual se mostrará un formulario editable en la cual el usuario podrá cambiar sus datos de perfil.
3: El usuario cambia los datos de su perfil y da clic en Actualizar. FA1	4: Si los datos ingresados son correctos el sistema lo redirigirá al <i>Dashboard</i> de usuario y mostrará un mensaje de éxito.
<b>Extensivos (Flujos alternativos)</b>	
FA1 – El usuario ingresa datos incorrectos como símbolos y números en el nombre. El sistema mostrará un mensaje de error en el formulario que indique el error cometido.	

Tabla C.4 Gestión de usuarios (personal médico) del sistema.

<b>Caso de uso C2: Gestión de usuarios del sistema.</b> Los usuarios dentro del sistema tendrán la posibilidad gestionar otros usuarios que estén por debajo de su nivel de autorización (crear, modificar, eliminar).	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: En el Dashboard del usuario da clic en el botón “ <i>users</i> ” o en la barra de navegación.	2: Se redirigirá al usuario a la página de usuarios del sistema.
3: El usuario da clic en el botón “ <i>new user</i> ”.	4: El sistema mostrará un formulario para ingresar los datos del nuevo usuario (rol, primer nombre, apellido, nombre de usuario, e-mail, contraseña y confirmación).
5: El usuario ingresa los datos del usuario y da clic en “registrar nuevo usuario”. FA1	6: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de usuarios y muestra un mensaje de éxito. FA2

7: El usuario da clic en el botón de “editar” de alguno de los usuarios mostrados.	8: El sistema muestra el formulario con la información del usuario seleccionado.
9: El usuario modifica los datos del usuario y da clic en actualizar usuario. FA1	10: El sistema comprueba los datos ingresados y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de usuarios y muestra un mensaje de éxito. FA2
11: El usuario da clic en el botón “Eliminar usuario”.	12: El sistema muestra una alerta de seguridad “Estás seguro que deseas eliminar este usuario?”.
13: El usuario da clic en Aceptar. FA3	14: El sistema muestra un mensaje de usuario eliminado correctamente.
15: El usuario escribe algo en la barra de búsqueda en la página de usuarios y da clic en “Search”.	16: El sistema muestra todas las coincidencias de usuarios encontradas con la o las palabras escritas. FA4
<b>Extensivos (Flujos alternativos)</b>	
FA1 – El usuario ingresa datos incorrectos como símbolos y números en el nombre, etc. El sistema mostrará un mensaje de error en el formulario que indique el error cometido.	
FA2 – Error al guardar los datos en el sistema de almacenamiento, el sistema redirigirá al usuario a la página de usuarios y mostrará un mensaje de error.	
FA3 – El usuario da clic en cancelar, el sistema cierra la alerta.	
FA4 – El sistema no encuentra ninguna coincidencia y muestra una página en blanco.	

Tabla C.5 Gestión de los pacientes del sistema.

<b>Caso de uso C5: Gestión de los pacientes del sistema.</b> Los usuarios dentro del sistema tendrán la posibilidad de editar la información de los pacientes siempre y cuando tenga la autorización para hacerlo. Los estudiantes solo pueden ingresar nueva información de un paciente ya existente. Los demás (administrador, doctor y enfermera) pueden crear nuevos pacientes, eliminarlos y actualizarlos.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>

1: En el Dashboard del usuario da clic en el botón “ <i>patients</i> ” o en la barra de navegación.	2: Se redirigirá al usuario a la página de pacientes del sistema.
3: El usuario da clic en el botón “ <i>new patient</i> ”.	4: El sistema mostrará un formulario para ingresar los datos del nuevo paciente (curp, primer nombre, apellido, edad, foto, fecha de nacimiento, género, tipo de sangre, calle, número exterior, colonia, código postal, número de teléfono, responsable, relación con responsable y cuarto).
5: El usuario ingresa los datos y da clic en registrar nuevo paciente. FA1	6: El sistema comprobara los datos y los guardara en el sistema de almacenamiento. Redirigirá al usuario a la página de pacientes y mostrará un mensaje de paciente registrado correctamente. FA2
7: El usuario da clic en el botón acciones de algún paciente y selecciona “ <i>Editar</i> ”.	8: El sistema muestra un formulario con la información del paciente.
9: El usuario modifica los datos del paciente y da clic en actualizar paciente. FA1	10: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirigirá al usuario a la página de pacientes y mostrará un mensaje de éxito. FA2
11: El usuario da clic en el botón acciones y selecciona historial médico.	12: El sistema muestra un formulario con la información del historial médico. En caso de un nuevo paciente aparecerá en blanco.
13. El usuario modifica o ingresa los datos del historial médico del paciente y da clic en Agregar o Actualizar dado el caso. FA1	14: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de pacientes y mostrará un mensaje de éxito. FA2
11: El usuario da clic en el botón acciones y selecciona exploración física.	12: El sistema muestra un formulario con la información de la exploración física del paciente. En caso de un nuevo paciente aparecerá en blanco.
13. El usuario modifica o ingresa los datos del historial médico del paciente y da clic en Agregar o Actualizar dado el caso. FA1	14: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de pacientes y mostrara un mensaje de éxito. FA2

15: El usuario da clic en el botón acciones y selecciona smac 30.	16: El sistema muestra un formulario con la información del smac 30 del paciente. En caso de un nuevo paciente aparecerá en blanco.
17. El usuario modifica o ingresa los datos del smac30 del paciente y da clic en Agregar o Actualizar dado el caso. FA1	18: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de pacientes y mostrará un mensaje de éxito. FA2
19: El usuario da clic en el botón acciones y selecciona radiografías.	20: El sistema muestra un formulario con la información necesaria para subir una radiografía. Y en la parte inferior muestra las radiografías del paciente. En caso de un nuevo paciente solo aparece el formulario para subir nuevas radiografías.
21. El usuario selecciona una radiografía le da una descripción y da clic en subir radiografía.	22: El sistema comprueba los datos y los guarda en el sistema de almacenamiento. Redirige al usuario a la página de radiografías del paciente. FA2
23: El usuario (en la página de radiografías del paciente) selecciona una de las radiografías y da clic en editar.	24: El sistema mostrará el formulario con la radiografía a editar.
25: El usuario modifica los datos de la radiografía y da clic en actualizar. FA1	26: El sistema comprueba los datos y guarda en el sistema de almacenamiento. Redirige al usuario a la página de radiografías del paciente.
27: El usuario selecciona el botón eliminar radiografía.	28: El sistema mostrará una alerta de comprobación de eliminación.
29: El usuario da clic en Aceptar. FA3	30: El sistema elimina la radiografía seleccionada. Redirige al usuario a la página de radiografías y muestra un mensaje de éxito.
31: El usuario da clic (en la página de pacientes) el botón de acciones y selecciona eliminar.	32: El sistema muestra una alerta de comprobación antes de eliminar al paciente.

33: El usuario da clic en Aceptar. FA3	34: El sistema elimina al paciente del sistema de almacenamiento y muestra un mensaje de éxito.
35: El usuario escribe algo en la barra de búsqueda de la página de pacientes y da clic en "Search".	36: El sistema muestra todas las coincidencias de pacientes encontradas con la o las palabras escritas. FA4
<b>Extensivos (Flujos alternativos)</b>	
FA1 – El usuario ingresa datos incorrectos como símbolos y números en el nombre, etc. El sistema mostrará un mensaje de error en el formulario que indique el error cometido.	
FA2 – Error al guardar los datos en el sistema de almacenamiento, el sistema redirigirá al usuario a la página de usuarios y mostrará un mensaje de error.	
FA3 – El usuario da clic en cancelar, el sistema cierra la alerta.	
FA4 – El sistema no encuentra ninguna coincidencia y muestra una página en blanco.	

b. Documento de casos de uso (Sistema de Authoring).

---

## Casos de uso detallados

### Sistema de authoring

Autor: César Javier Maldonado Flores --- 183137

*Tabla C.1 Autenticación de usuario a través de página de inicio de sesión*

<b>Caso de uso C1: Identificación de usuario a través de página de inicio de sesión.</b> Los usuarios autorizados del sistema de authoring podrán acceder a través de una página de inicio la cual pedirá nombre de usuario y contraseña como credenciales de acceso.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>

1: Acceder a la página del prototipo de authoring.	2: Mostrar la página de inicio de sesión.
3: Introducir el nombre de usuario (email), contraseña y presionar el botón de “iniciar sesión”.	4: Comprobar las credenciales de usuario y dar acceso al sistema. FA1
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Las credenciales son incorrectas, se muestra la página de inicio de sesión con un mensaje de “nombre de usuario o contraseña incorrecta”.	

Tabla C.2 Dashboard de usuario en authoring.

<b>Caso de uso C2: Dashboard de usuario.</b> Los usuarios dentro del sistema serán presentados con un <i>Dashboard</i> .	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Iniciar sesión en el sistema de authoring.	2: Mostrar el Dashboard del usuario según su nivel de autorización. Con las opciones de “todos los perfiles” y “mis perfiles”.
<b>Extensivos (Flujos alternativos)</b>	
Ninguno	

Tabla C.3 Lista de todos los perfiles del sistema.

<b>Caso de uso C3: Lista de perfiles del sistema.</b> Los usuarios dentro del sistema tendrán la posibilidad de ver la lista todos los perfiles creados por los usuarios dentro del sistema.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “All Profiles” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de los perfiles del sistema donde se mostrará una lista con todos los perfiles registrados por los usuarios del sistema. FA1
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	

Tabla C.4 Lista de los perfiles del usuario.

<b>Caso de uso C4: Lista de perfiles de usuario.</b> Los usuarios dentro del sistema tendrán la posibilidad de ver la lista todos los perfiles creados por los usuarios dentro del sistema.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “ <i>My Profiles</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de los perfiles del usuario donde se mostrará una lista con todos los perfiles registrados del usuario. FA1
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	

Tabla C.5 Ver perfil y copiar perfil en la lista de los perfiles del sistema.

<b>Caso de uso C5: Ver perfil y copiar perfil.</b> Los usuarios dentro del sistema tendrán la posibilidad de ver la información detallada de cada uno de los perfiles y copiar el perfil a su cuenta de usuario.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “ <i>All Profiles</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de los perfiles donde se mostrará una lista con todos los perfiles del sistema. FA1
3: Dar clic en el botón “ <i>view</i> ” de alguno de los perfiles listados.	4: El sistema mostrará la página de vista de perfil con toda la información del perfil.
3: El usuario Da clic en el botón “ <i>Copy Profile</i> ” de la parte inferior central.	4: El sistema copiará la información del perfil en base de datos en la cuenta de usuario destino y mostrar un mensaje de éxito. FA2
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	
FA2 – Error al guardar en la base de datos, se muestra mensaje de error.	

Tabla C.6 Búsqueda de perfil.

<b>Caso de uso C6: Búsqueda de perfil.</b> Los usuarios dentro del sistema tendrán la posibilidad de buscar perfiles desde las listas en las páginas de “todos los perfiles” y “mis perfiles”.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “ <i>All Profiles</i> ” o “ <i>My Profiles</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página de los perfiles seleccionada. Donde se mostrará el listado de perfiles. FA1
3: Dar clic en la barra de búsqueda de la parte superior derecha de la lista de perfiles. Escribir la cadena de búsqueda y dar clic en el icono de lente.	4: El sistema mostrará los perfiles que coincida con la cadena de búsqueda del usuario. FA2
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	
FA2 – La cadena de búsqueda no arrojo ningún resultado, se muestra una lista vacía.	

Tabla C.7 Creación de nuevo perfil.

<b>Caso de uso C7: Creación de nuevo perfil.</b> Los usuarios dentro del sistema tendrán la posibilidad de crear nuevos perfiles para la aplicación de realidad aumentada.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “ <i>My Profiles</i> ” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página con todos sus perfiles en la cual se mostrará una lista con todos sus perfiles. FA1, FA2
3: El usuario da clic en el botón de la parte superior izquierda “Nuevo perfil”.	4: El sistema mostrará la página para la creación de un nuevo perfil.
5: El usuario rellena el campo de nombre del perfil y elige la información que deberá contener el perfil. Da clic en el botón de guardar nuevo perfil.	6: El sistema guardará los datos del nuevo perfil en la base de datos y mostrar un mensaje de perfil guardado exitosamente. FA3
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles.	
FA2 – El usuario no tiene ningún perfil en base de datos y su muestra una lista vacía.	

FA3 – Error al guardar el perfil en la base de datos, se muestra un mensaje de error.

Tabla C.8 Gestión de perfiles de usuario.

<b>Caso de uso C8: Búsqueda de perfil.</b> Los usuarios tendrán la posibilidad de modificar y eliminar sus perfiles dentro del sistema.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo con navegador web y acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Dar clic en el botón “My Profiles” en el <i>Dashboard</i> del usuario o en la barra navegadora de la página.	2: Se redirigirá al usuario a la página con todos sus perfiles en la cual se mostrará una lista con todos sus perfiles. FA1, FA2
3: El usuario da clic en el botón “editar” de alguno de los perfiles listado en la página.	4: El sistema mostrará la página con la información del perfil.
5: El usuario podrá modificar la información del perfil y renombrarlo. Después, presiona el botón “actualizar perfil” de la parte inferior.	6: El sistema guardará los nuevos datos del perfil en la base de datos y mostrar un mensaje de perfil actualizado exitosamente. FA3
7: El usuario regresa a la página de todos los perfiles y da clic en el botón “eliminar” de alguno de los perfiles listados.	7: Se muestra un mensaje de alerta. “Estás seguro que quieres eliminar este perfil?” con las opciones aceptar y cancelar.
8: El usuario da clic en aceptar del mensaje mostrado por el sistema.	9: El sistema borra el perfil de la base de datos y muestra un mensaje de “El perfil fue exitosamente eliminado”. FA4
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Error de conexión con la base de datos y no se pudo mostrar la lista de perfiles. FA2 – El usuario no tiene ningún perfil en base de datos y su muestra una lista vacía. FA3 – Error al actualizar el perfil en la base de datos, se muestra un mensaje de error. FA4 – Error al borrar el perfil de la base de datos.	

c. Documento de casos de uso (Sistema de RA).

Tabla C.1 Autenticación de usuario a través de página de inicio de sesión

<b>Caso de uso C1: Autenticación de usuario a través de página de inicio de sesión.</b> Los usuarios (personal médico) autorizados del sistema podrán acceder a través de una pantalla de inicio la cual pedirá nombre de usuario y contraseña como credenciales de acceso. De ser correcto se le permitirá el acceso al sistema de lo contrario se mostrará un mensaje de error en nombre de usuario o contraseña.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar el dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Ejecutar la aplicación de RA en el dispositivo HoloLens.	2: La aplicación mostrará la pantalla de inicio de sesión.
3: Introducir el nombre de usuario, contraseña y presionar el botón de “iniciar sesión”.	4: Comprobar las credenciales de usuario y dar acceso al sistema. FA1, FA2
<b>Extensivos (Flujos alternativos)</b>	
FA1 – Las credenciales son incorrectas, se muestra un mensaje de “nombre de usuario o contraseña incorrecta”.	
FA2 – Error en la conexión con la base de datos (sin conexión a internet, instancia EC2 no disponible).	

Tabla C.2 Búsquedas de paciente.

<b>Caso de uso C2: Búsquedas de paciente.</b> Los usuarios una vez dentro del sistema serán presentados con una ventana de búsqueda. Aquí se podrán realizar búsqueda por escaneo de código QR, búsqueda introduciendo nombre o cuarto de paciente y búsqueda por reconocimiento de voz (nombre de paciente).	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>

1: Iniciar sesión en el sistema de realidad aumentada.	2: Mostrar la ventana de búsqueda con una breve introducción a los distintos tipos de búsqueda.
3: El usuario dice la palabra “scan”.	4: Se muestra un recuerdo para posicionar el código QR. Tras escanearlo el sistema muestra al paciente del QR. Si el QR pertenece un cuarto con varios pacientes se muestra una lista para seleccionar al usuario. FA1
5: El usuario dice la palabra “look for”.	6: El sistema muestra una ventana de grabación indicándole al usuario que puede decir el nombre del paciente a buscar. Se muestra al paciente encontrado. FA2
7: El usuario selecciona el icono de búsqueda en la ventana e introduce el nombre de paciente o código de cuarto “manualmente”.	8: El sistema realiza la búsqueda y muestra al paciente. Si la búsqueda arroja más de 1 resultado se muestra una lista de los pacientes para seleccionar por el usuario.
<b>Extensivos (Flujos alternativos)</b>	
FA1 – No se encuentran pacientes para el código escaneado. Falla al conectarse con el sistema de almacenamiento.	
FA2 – Fallo con los sistemas de reconocimiento de voz de Azure. No se encontraron pacientes con ese nombre.	

Tabla C.3 Ventana pop-up con casilla de selección de visibilidad.

<b>Caso de uso C3: Ventana pop-up con casilla de selección de visibilidad.</b> Una vez seleccionado el usuario de la búsqueda se mostrará una pequeña ventana pop-up con una foto del paciente y una pequeña porción de información. Y también una casilla de selección con el cual mostrar u ocultar dicho <i>pop-up</i> .	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: Tras la búsqueda el usuario selecciona el paciente.	2: Se mostrará un <i>pop-up</i> (pequeña ventana) con la foto e información del paciente.

	Además, se mostrará una casilla de selección para ocultar/mostrar el pop-up.
3: El usuario da clic en la casilla de selección.	4: Se oculta el pop-up del paciente hasta que el usuario vuelva a seleccionar la casilla de selección.
<b>Extensivos (Flujos alternativos)</b>	
Ninguno.	

Tabla C.4 Marco principal de visión de la información del paciente.

<b>Caso de uso C4: Marco de visión de la información del paciente.</b> Una vez el paciente ha sido seleccionado su “Número de identificación” (digito utilizado internamente por la aplicación) se encontrará activo. Lo que significa que cualquier tipo de información solicitada será acerca de este paciente en específico. En marco principal se mostrará la información seleccionada de dicho paciente.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: El usuario selecciona el paciente.	2: Se Muestra el marco principal con 4 diferentes secciones.
3: El usuario da clic en cualquiera de los botones en el panel izquierdo.	4: Se muestra la información seleccionada del paciente en el primer recuadro del marco (superior izquierdo).
5: El selecciona otra de los botones para mostrar información del usuario.	6: Se muestra la ventana en el segmento superior derecho. Así sucesivamente hasta llenar los cuatro segmentos del marco. FA1
7: El usuario da clic en el botón [X] de la parte superior de la ventana de información.	8: Se cierra la ventana de información y se libera ese espacio del marco.
<b>Extensivos (Flujos alternativos)</b>	
FA1 – El marco tiene 4 segmentos ocupados por ventanas. No es posible abrir más ventanas hasta cerrar alguna de las actualmente abiertas.	

Tabla C.6 Panel lateral izquierdo con secciones de la información del paciente.

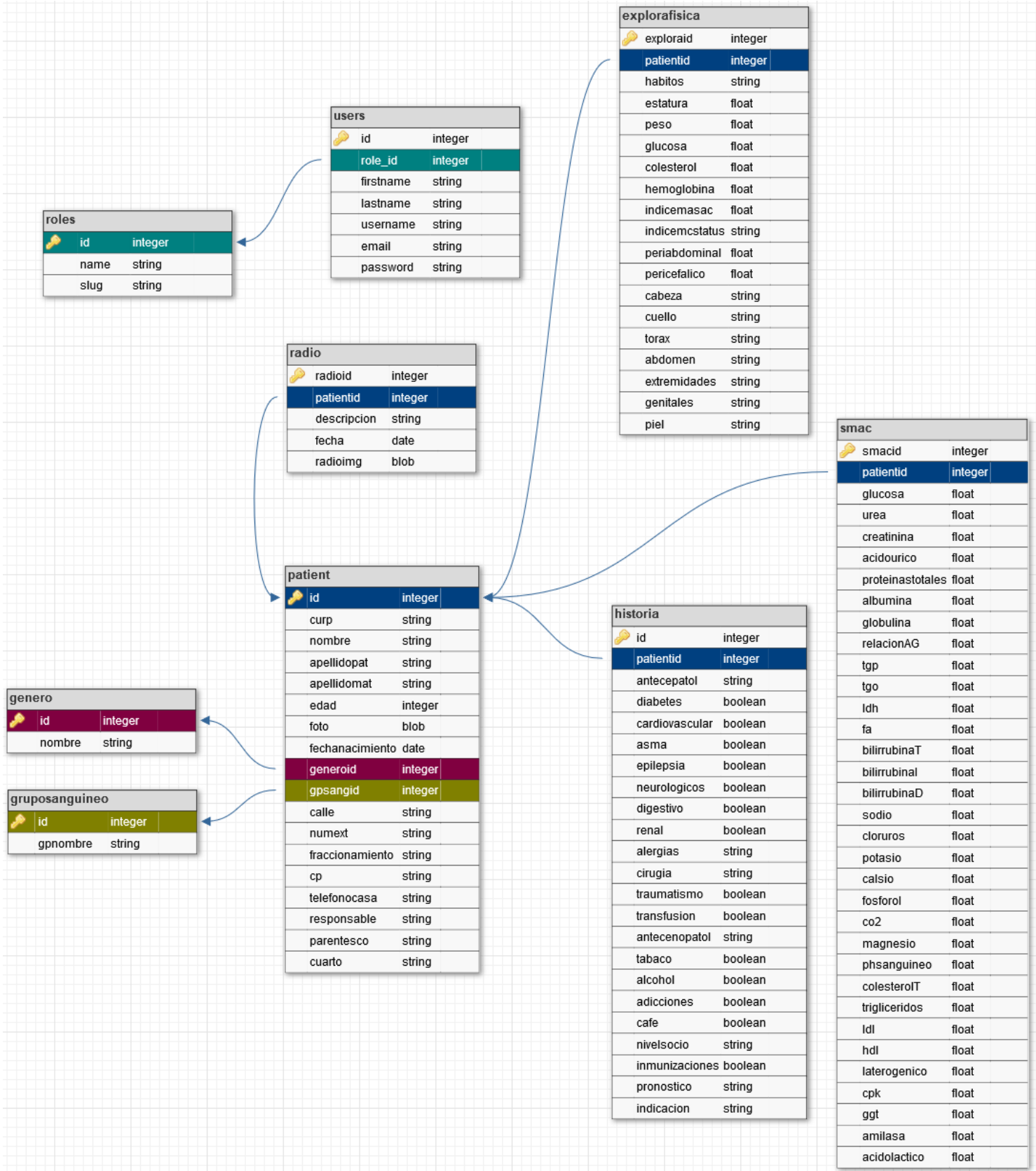
<b>Caso de uso C6: Panel lateral izquierdo con secciones de la información del paciente.</b>	
Panel con secciones de la información disponible para los pacientes registrados en el sistema de almacenamiento.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: El usuario una vez seleccionado el paciente se mostrará un panel lateral en la parte izquierda. El usuario selecciona cualquier sección de información.	2: Se mostrará la información seleccionada del paciente en el marco principal.
<b>Extensivos (Flujos alternativos)</b>	
Ninguna.	

*Tabla C.7 Vista de perfiles del usuario.*

<b>Caso de uso C7: Vista de perfiles del usuario.</b> Los perfiles de usuario que utiliza la aplicación de realidad aumentada deben ser previamente diseñados y creados en el sistema de authoring. Consisten en una serie de “secciones” de información dividida en 4 segmentos.	
<b>Actores:</b> Usuario, Prototipo.	
<b>Precondiciones:</b> Utilizar un dispositivo HoloLens con acceso a internet.	
<b>Escenario Principal de Éxito (Flujo Básico)</b>	
<b>Acción del actor</b>	<b>Responsabilidad del sistema</b>
1: El usuario selecciona al paciente. Selecciona la sección “perfiles” del panel izquierdo.	2: Se mostrarán los perfiles del usuario listados sobre el panel izquierdo.
3: El usuario selecciona un perfil de la lista.	4: Se muestran la información del perfil en el marco principal de acuerdo con su diseño. FA1
<b>Extensivos (Flujos alternativos)</b>	
Ninguna.	

# [Anexo D]

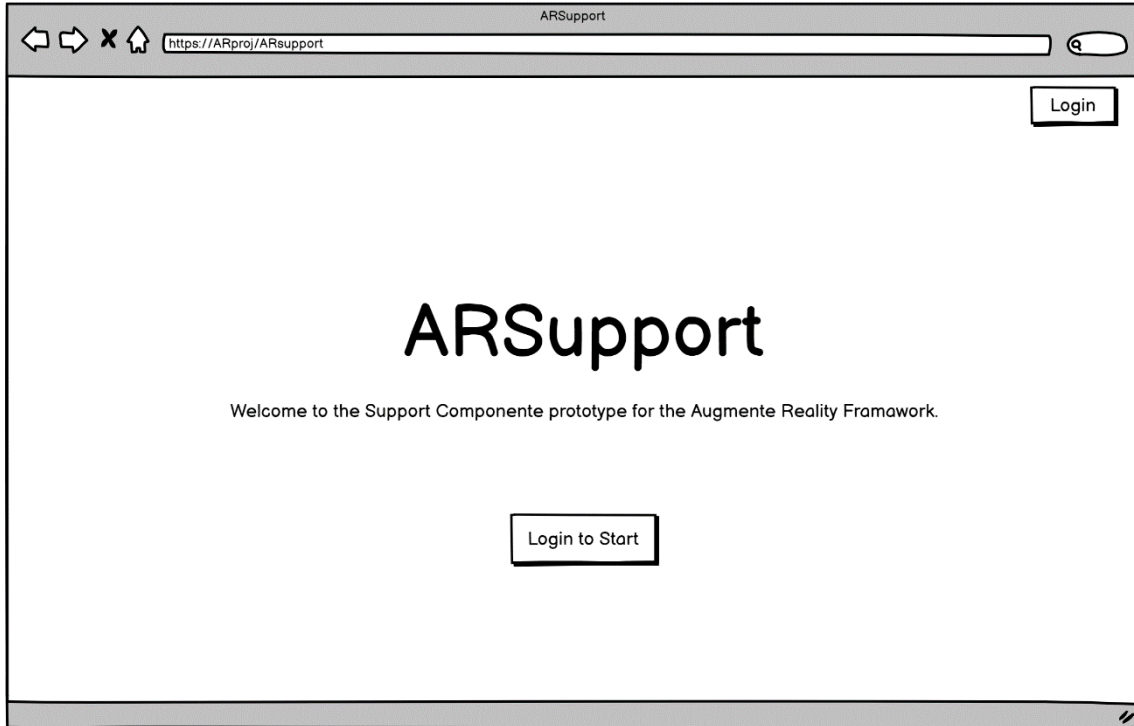
Base de datos de los cuatro sistemas del marco de trabajo.



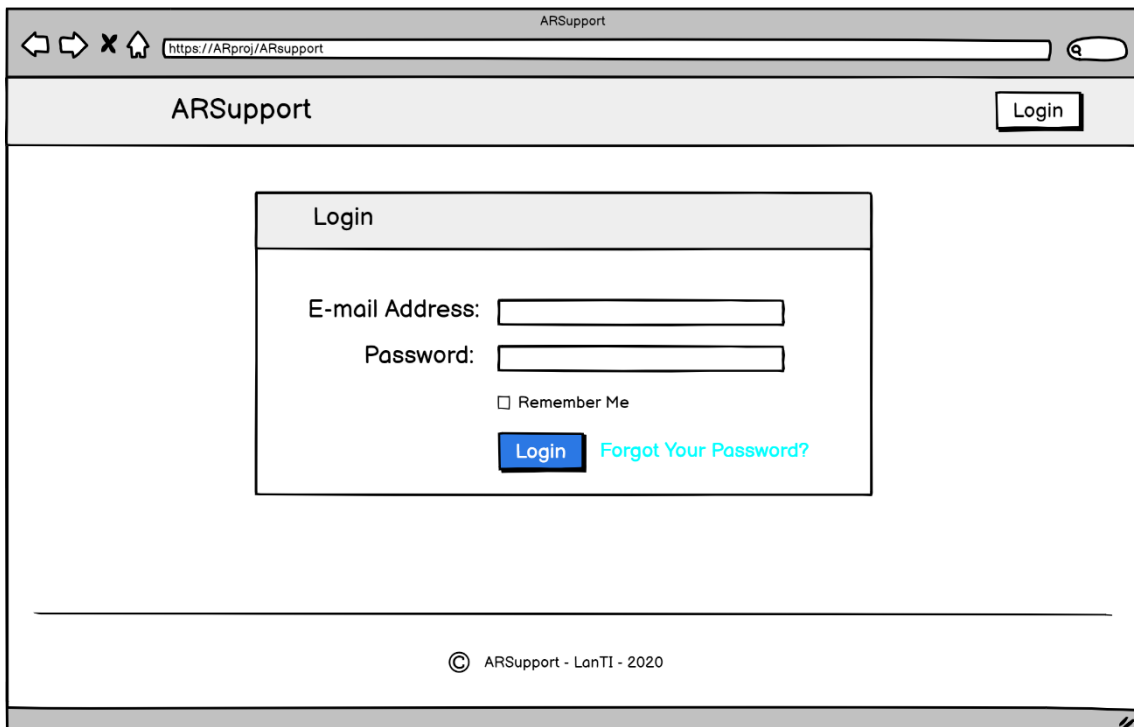
[Anexo E]

Pantallas de la interfaz de usuario de la aplicación web de soporte.

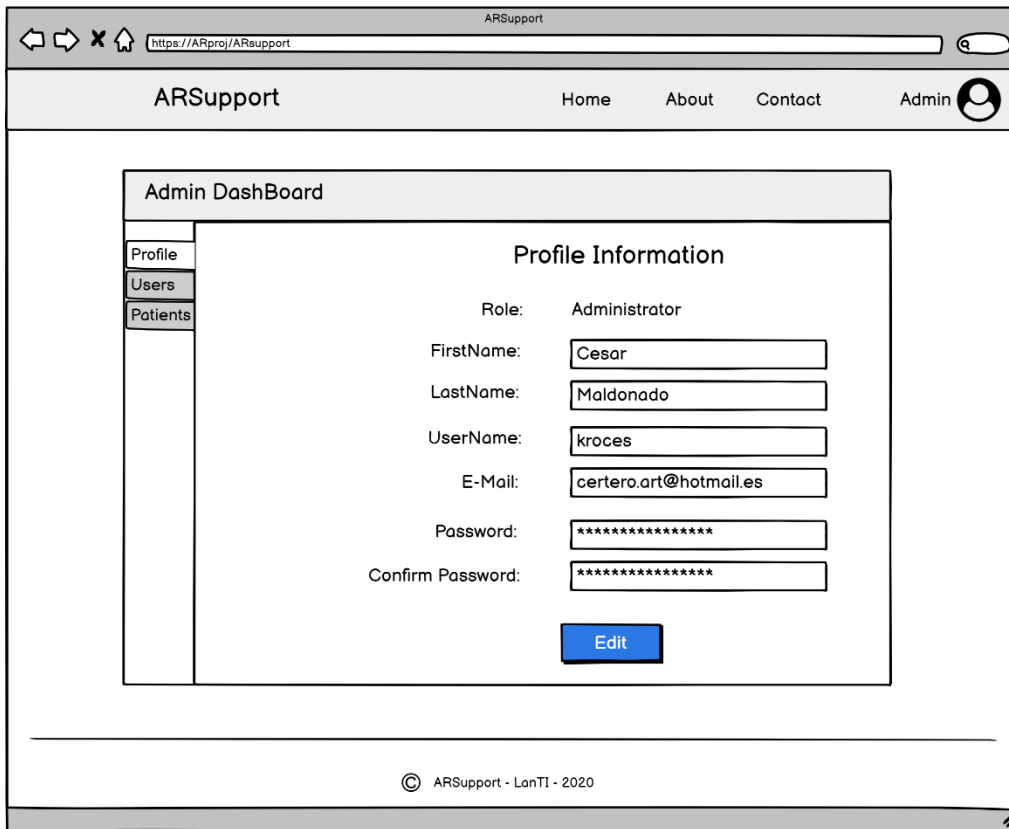
Pantalla de Bienvenida:



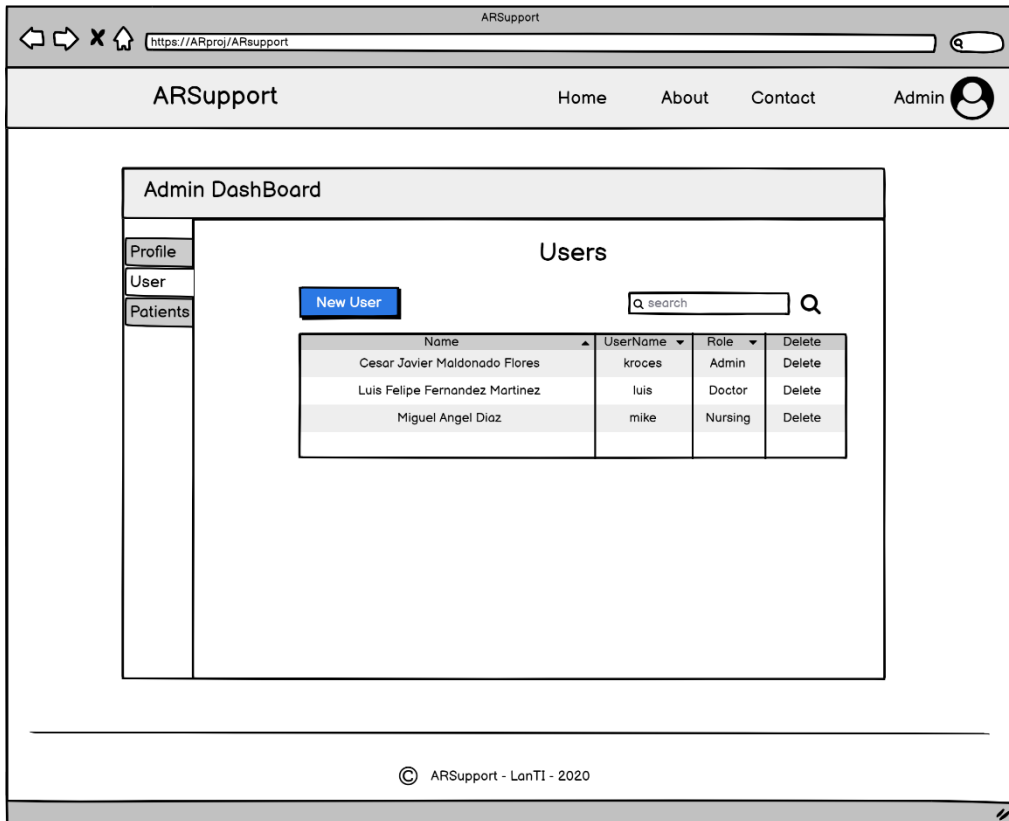
Pantalla de autenticación de usuario:



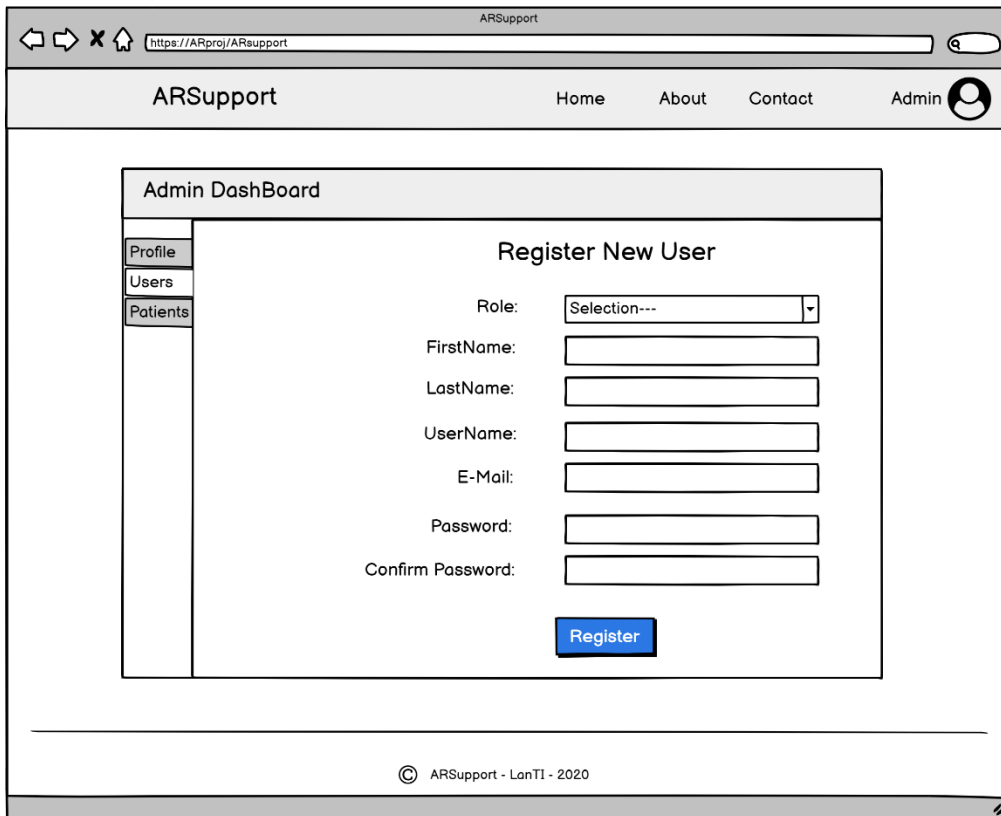
Pantalla de Dashboard de usuario:



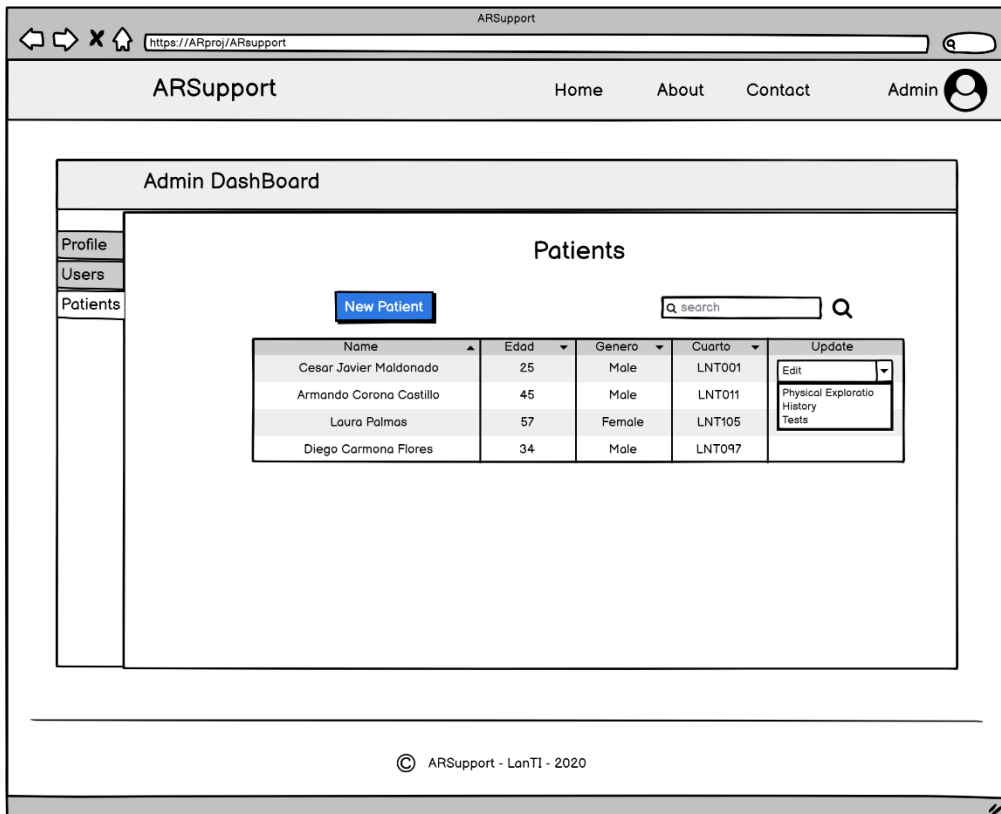
Pantalla de usuarios del sistema:



Pantalla de registro de nuevo usuario:



Pantalla de pacientes:



Pantalla de registro de nuevo paciente:

The screenshot shows a web browser window with the URL `https://ARproj/ARsupport`. The page title is "ARSupport" and the navigation menu includes "Home", "About", "Contact", and "Admin" (with a user profile icon). The main content area is titled "Admin DashBoard" and contains a sidebar with "Profile", "Users", and "Patients" (the active page). The central form is titled "Register New Patient" and includes the following fields:

- Foto:  Select the image...
- Nombre:
- Apellido Paterno:
- Apellido Materno:
- CURP:
- Edad:
- Fecha de Nacimiento:
- Genero:
- Grupo Sanguineo:
- Calle:
- No.
- Fraccionamiento:
- Codigo Postal:
- Telefono:
- Responsable:
- Parentesco:
- Cuarto:

A blue "Register" button is located at the bottom of the form. The footer of the page contains the copyright notice: © ARSupport - LanTI - 2020.

Pantalla de Exploración física:

The screenshot shows a web browser window with the URL `https://ARproj/ARsupport`. The page title is "ARSupport" and the navigation menu includes "Home", "About", "Contact", and "Admin" (with a user icon). The main content area is titled "Admin DashBoard" and contains a sidebar with "Profile", "Users", and "Patients" (the active section). The main form is titled "Edit Patient Physical Exploration" and contains the following fields:

- Habits:
- Stature:
- Weight:
- Glucose:
- Cholesterol:
- Hemoglobin:
- Body mass index:
- Body mass index status:
- Abdominal Perimeter:
- Head Circunference:
- Head:
- Neck:
- Chest:
- Abdomen:
- Extremities:
- Genitals:
- Skin:

At the bottom of the form is a blue button labeled "Update Patient Physical Info".

© ARSupport - LanTI - 2020

Pantalla de historial médico:

ARSupport

Home About Contact Admin

Admin DashBoard

Profile  
Users  
Patients

### Edit Patient Medical History

Pathological History:

No Pathological history:

Diabetes:

Cardiovascular:

Asthma:

Epilepsy:

Neurological:

Digestive:

Renal:

Transfusion:

Allergies:

Surgery:

Trauma:

Tobacco:

Alcohol:

Addictions:

Coffee:

Socioeconomic level:

Immunizations:

Medical prognosis:

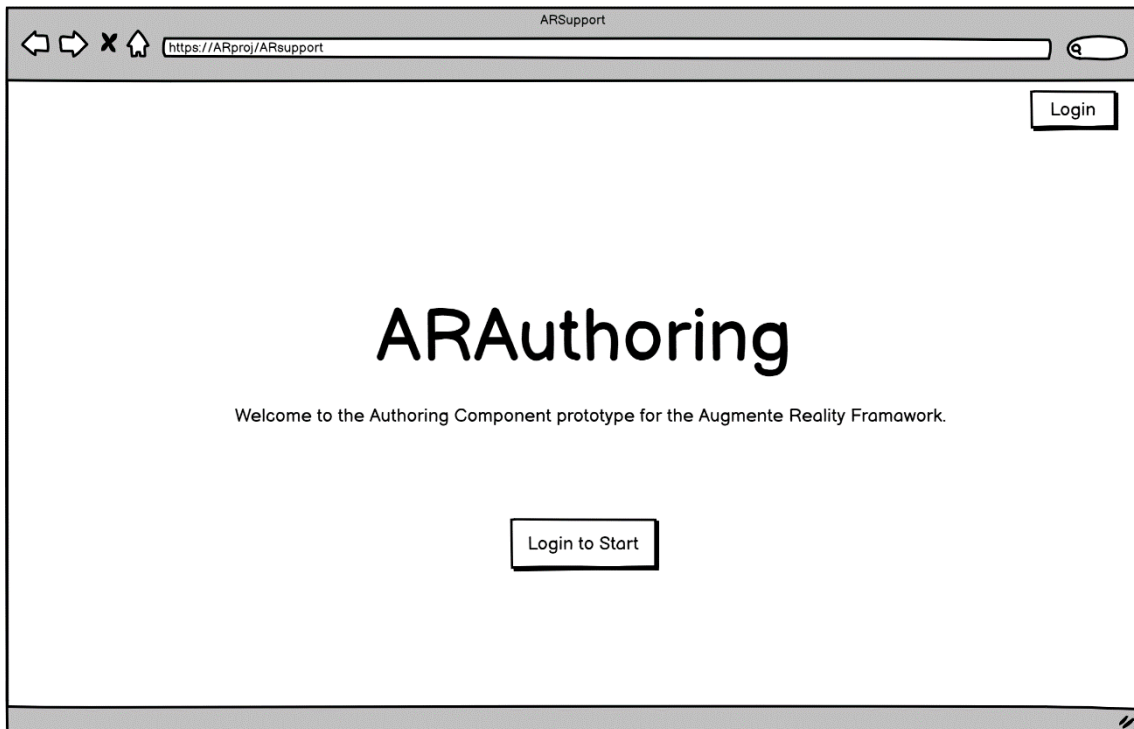
[Update Patient Physical Info](#)

© ARSupport - LanTI - 2020

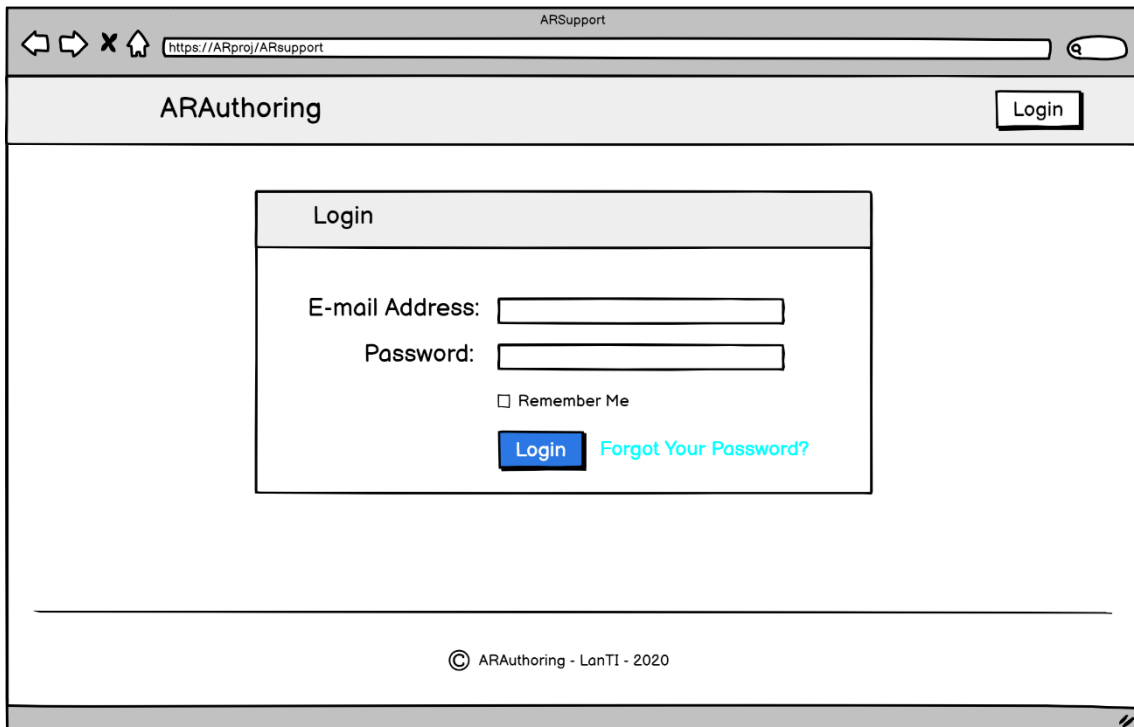
[Anexo F]

Pantallas de la interfaz de usuario de la aplicación web de authoring.

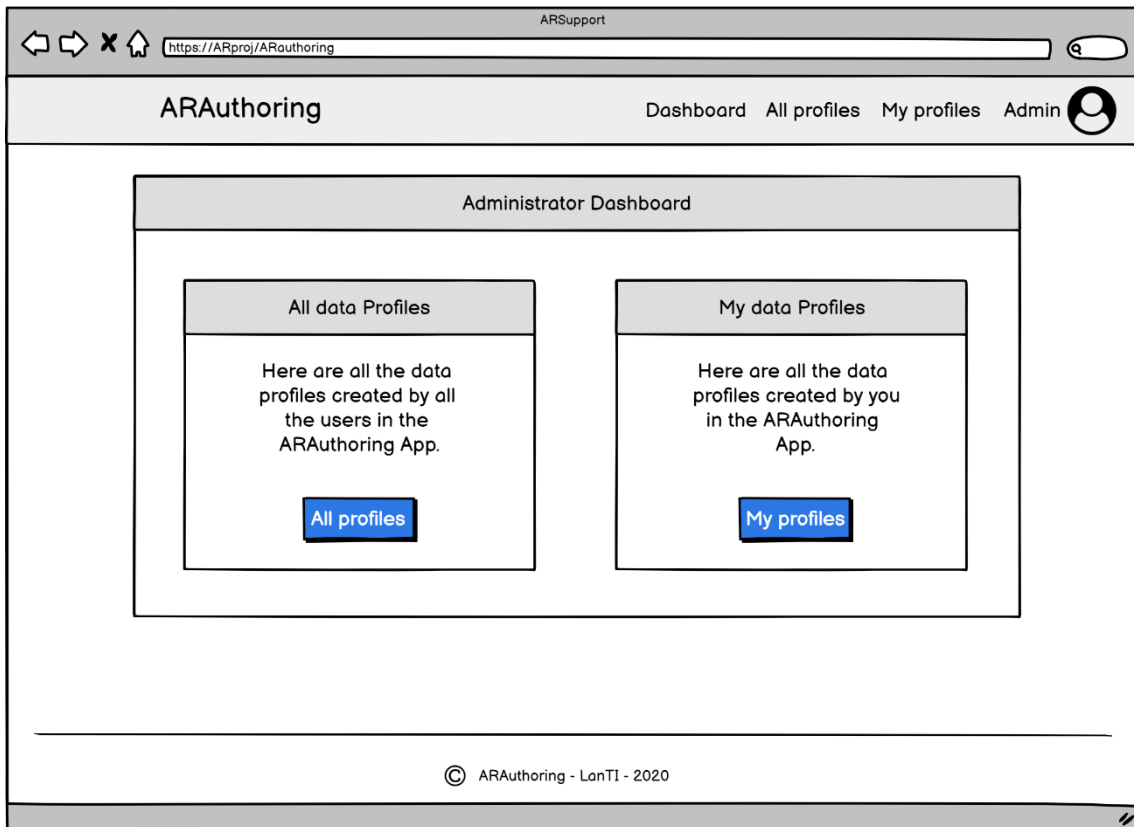
Pantalla de Bienvenida:



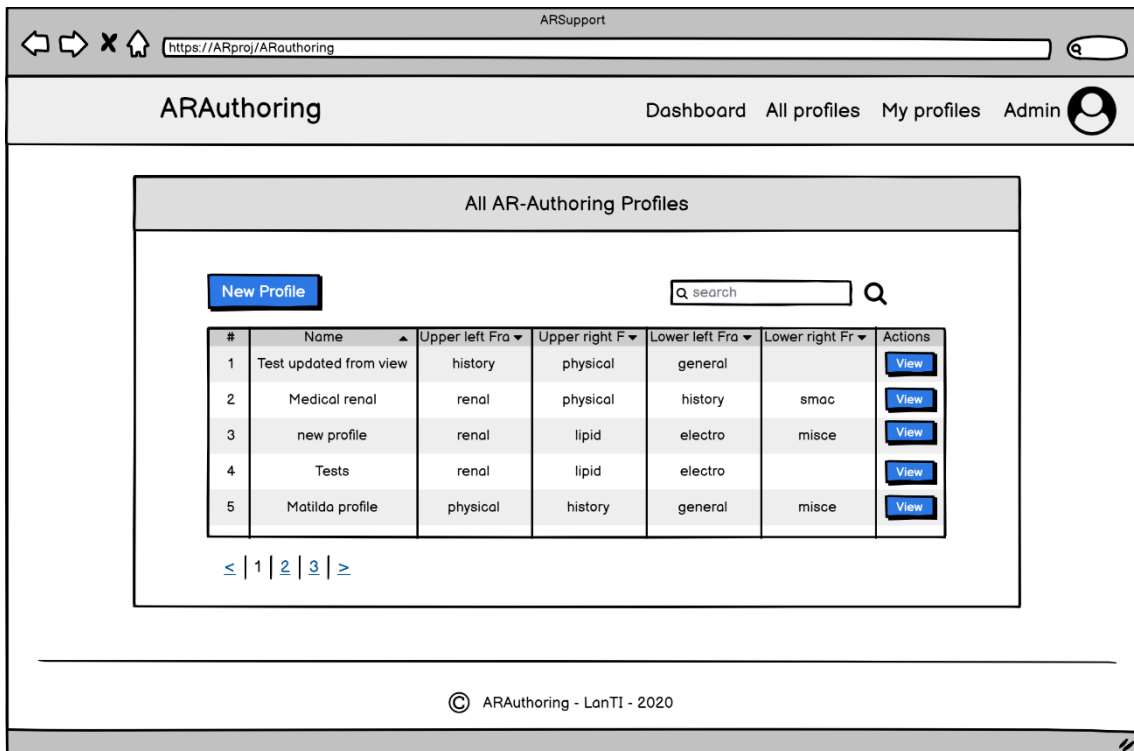
Pantalla de autenticación de usuario:



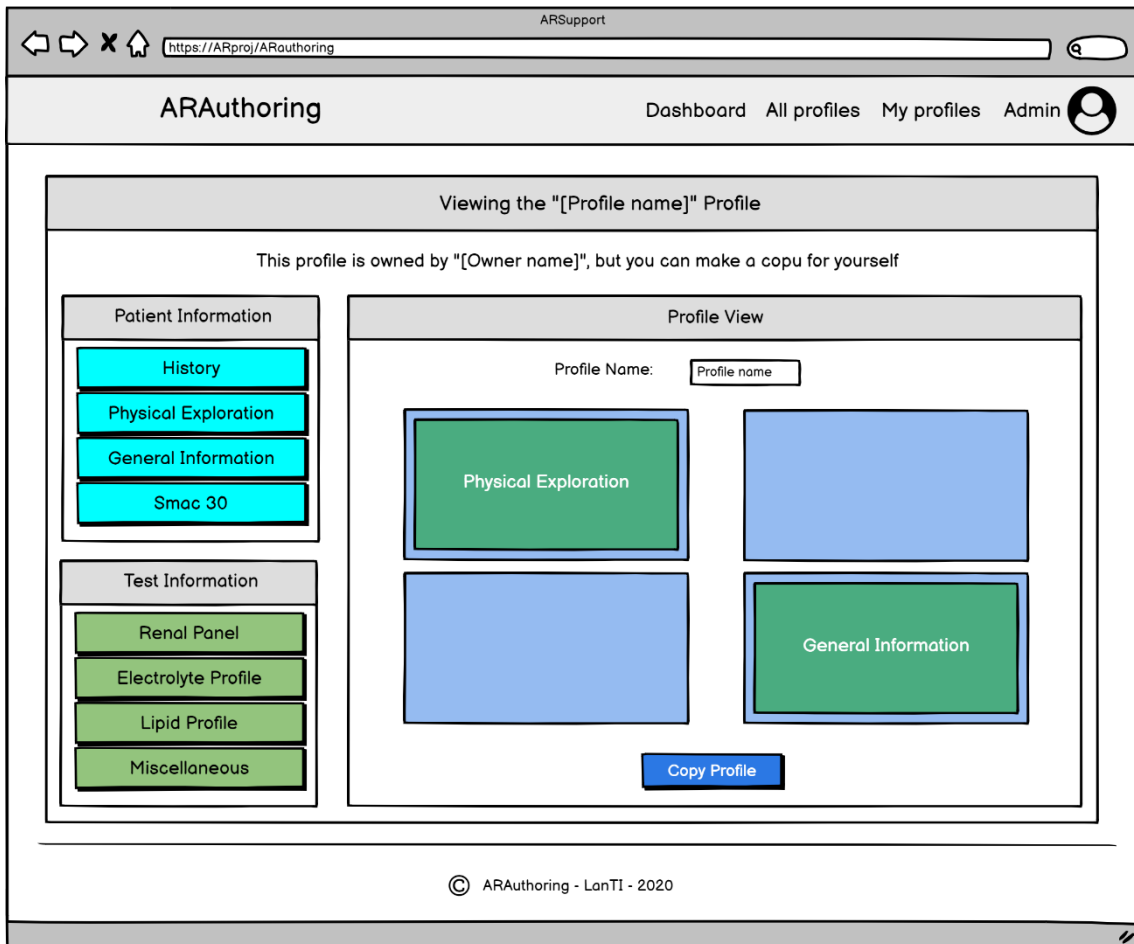
Pantalla *Dashboard* de authoring:



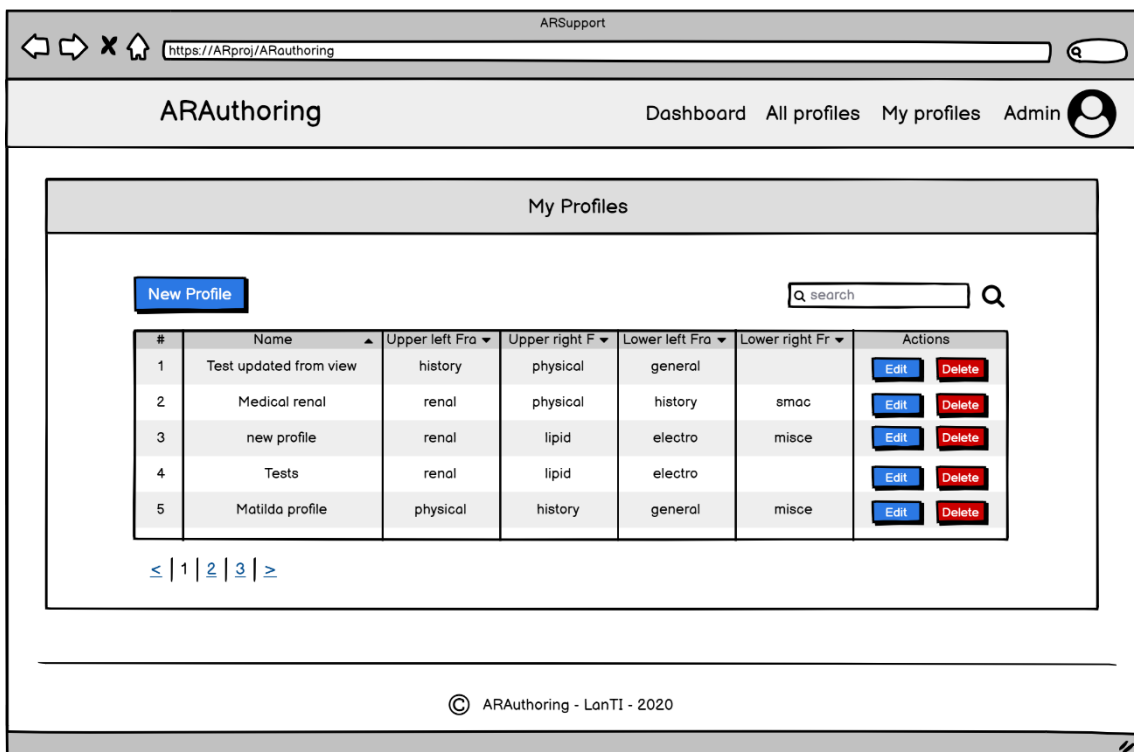
Pantalla “Todos los perfiles”:



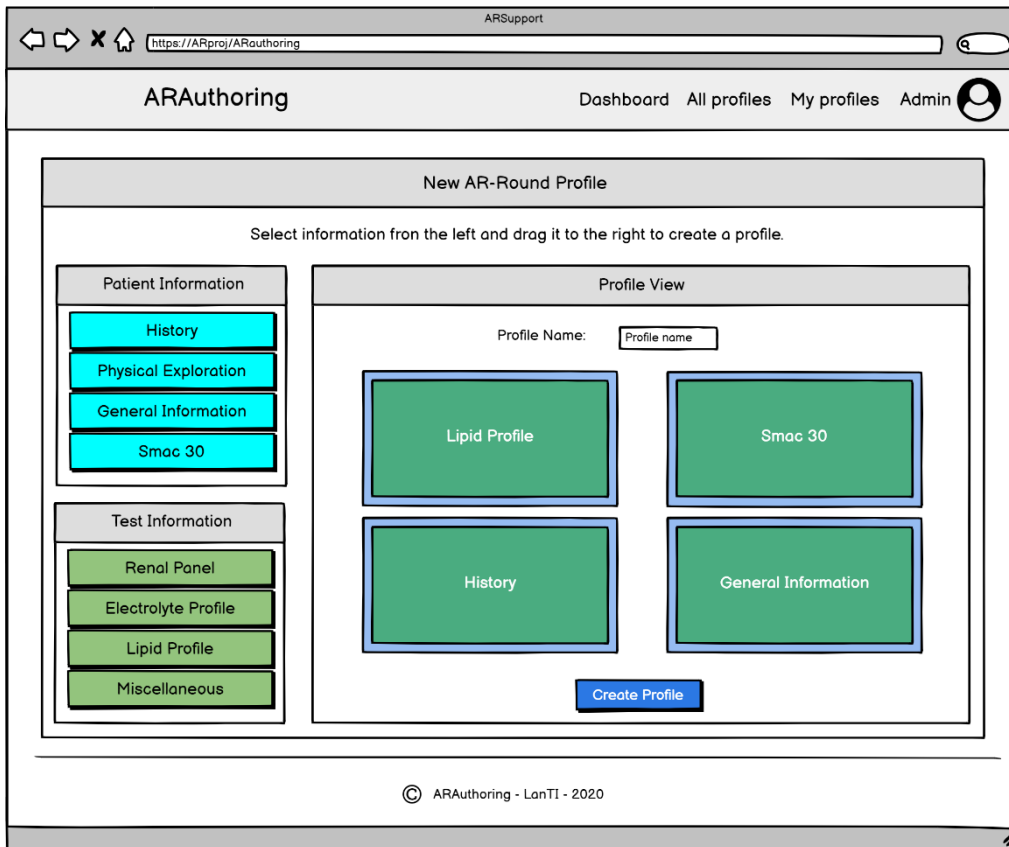
Pantalla de “Vista de perfil”:



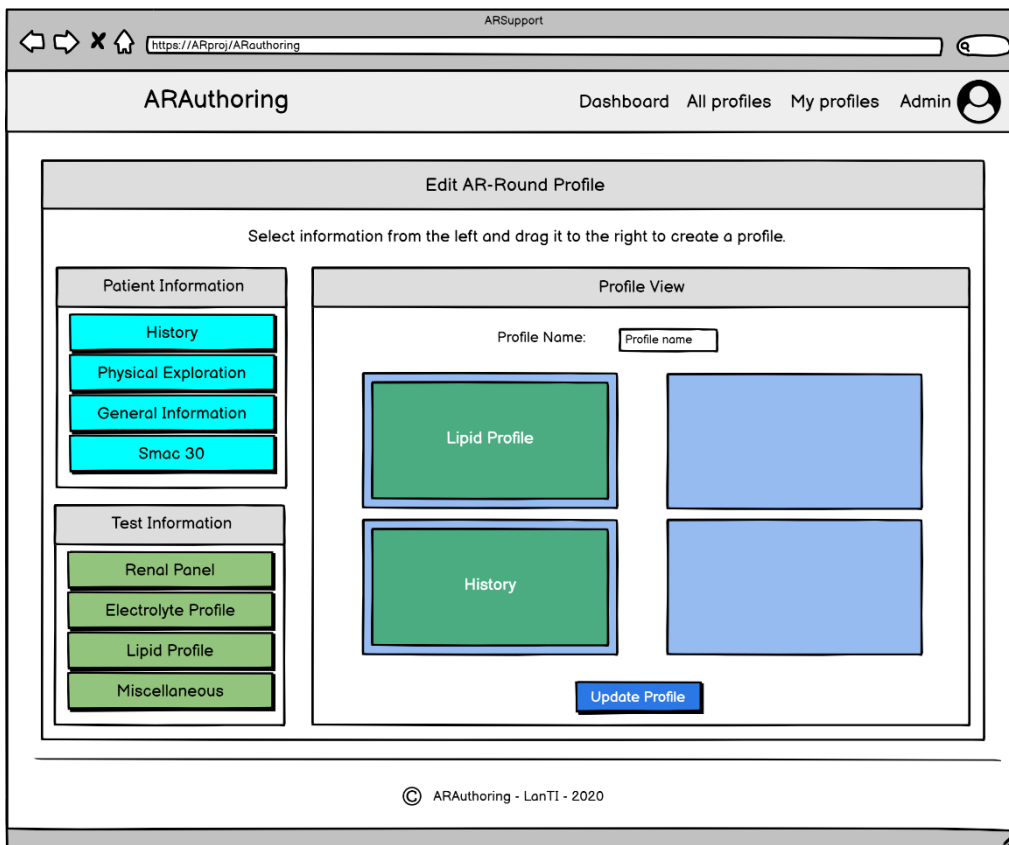
Pantalla de “Mis perfiles”:



Pantalla de “Crear perfil nuevo”:



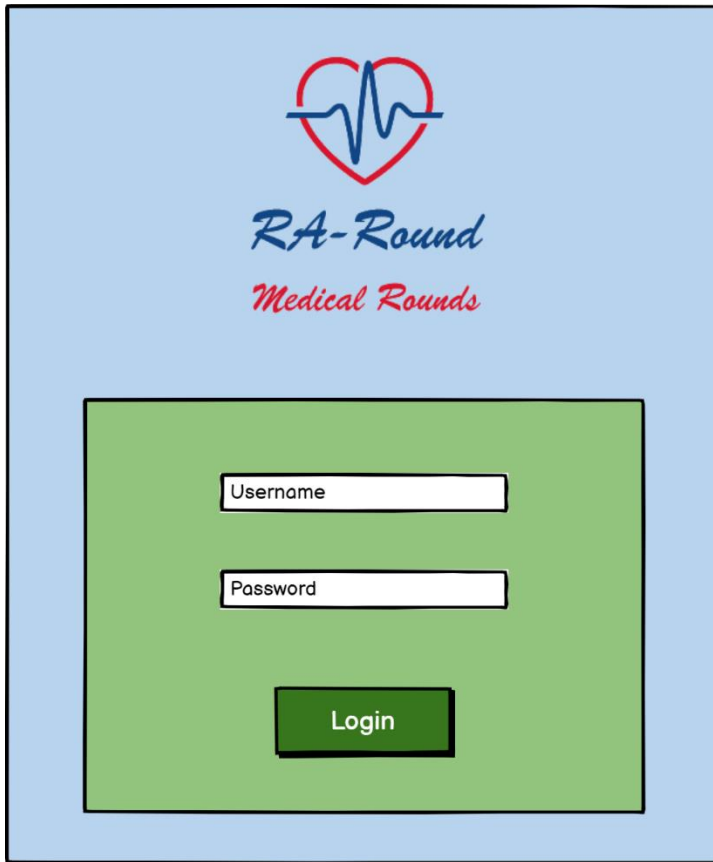
Pantalla de “Actualizar Perfil”:



[Anexo G]

Pantallas de la interfaz de usuario de la aplicación de realidad aumentada.

Pantalla “Inicio de sesión”:

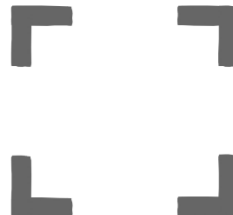


Pantalla “Búsqueda de paciente”:

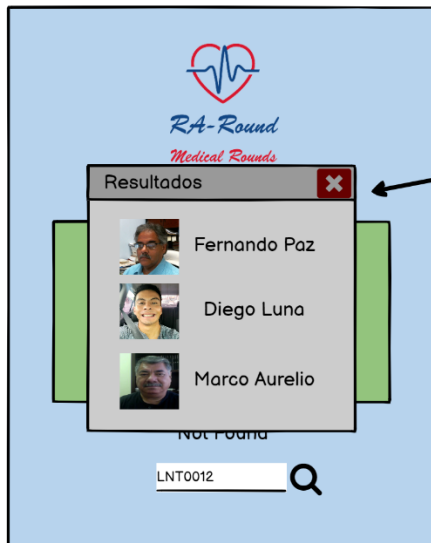


**SCAN -> Voice Command**

Apariencia de del lector que aparece para escanear los codigos QR. Aparece a proximadamente a un metro de distancia en frente del usuario de los hololens.



## SEARCH



Podría aparecer un objeto de lista con los pacientes de la búsqueda.

## LOOK FOR-> Voice Command

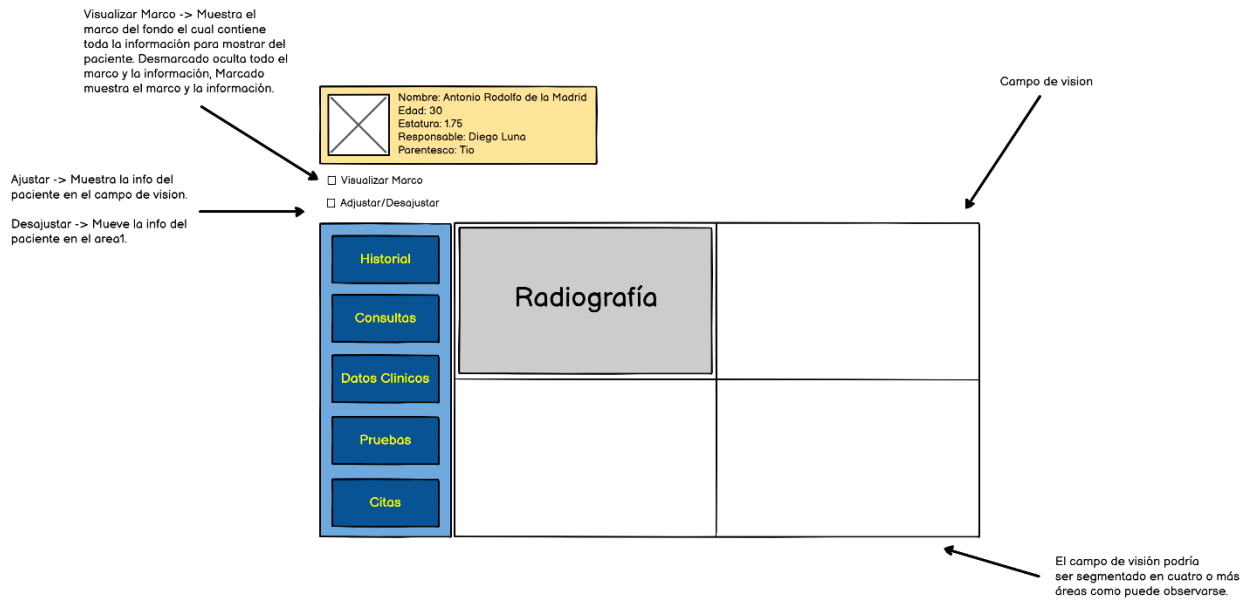
Apaciencia del comando "Look for" el cual reconoce la voz y busca al paciente por nombre.



Listening to

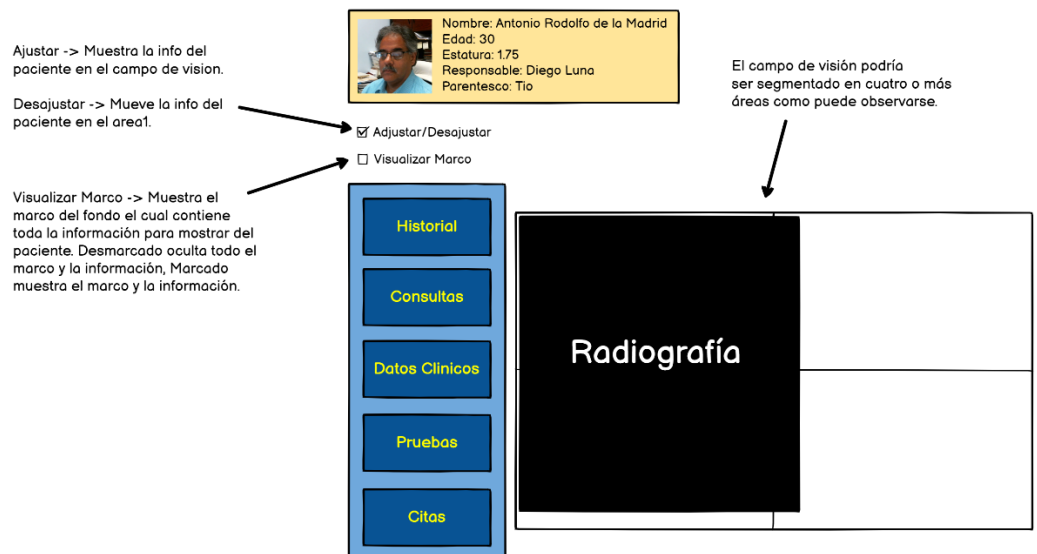
"Diego Luna"

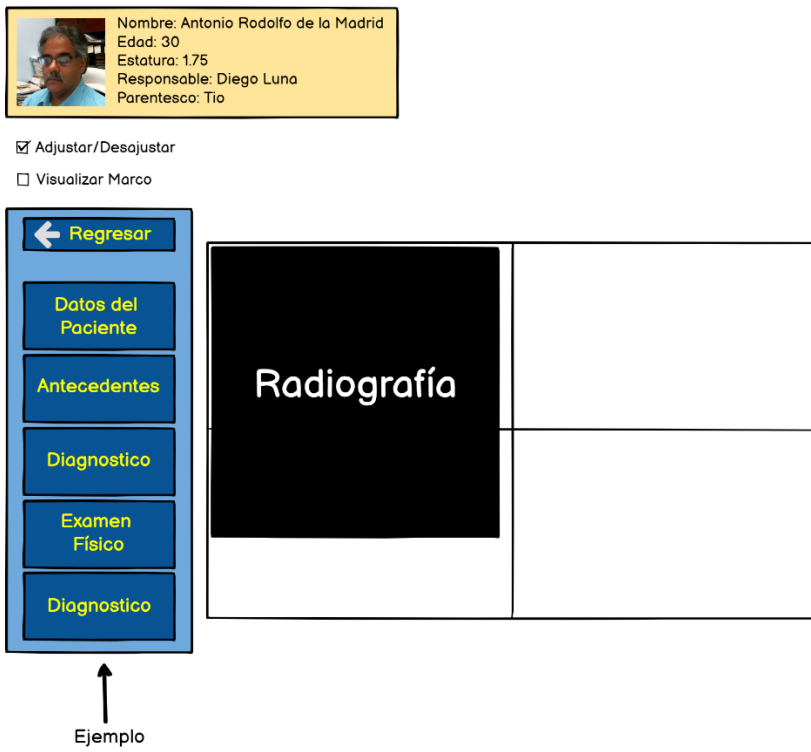
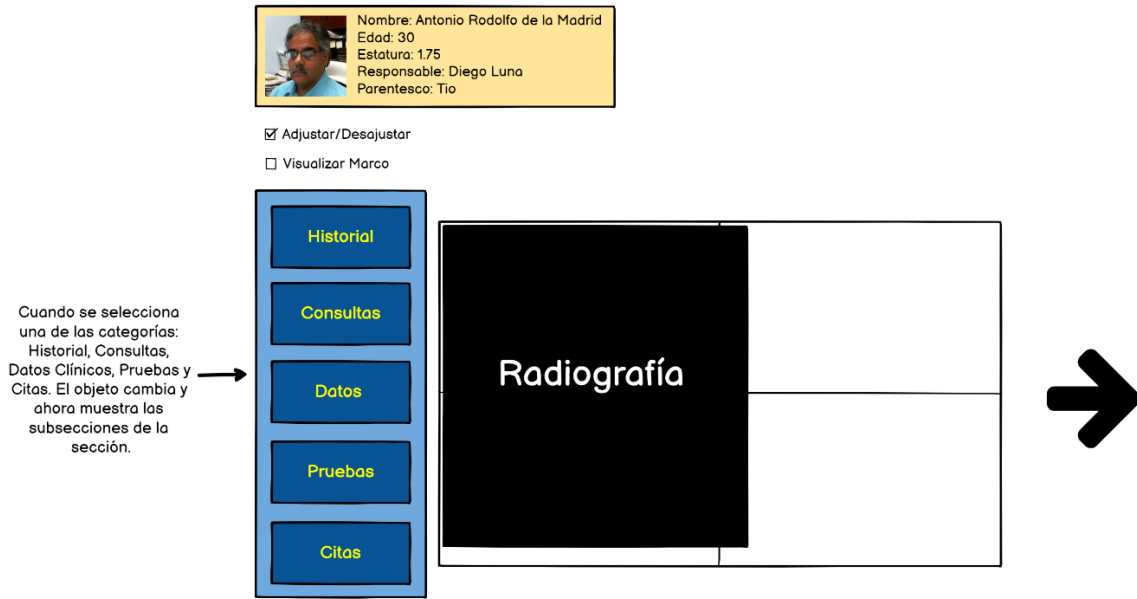
Pantalla "paciente seleccionado":



## Pantalla "AroundFrame":

Para mostrar esta parte de la aplicación un paciente debe haber sido seleccionado a través del escáner de QR o por búsqueda manual.





Pantalla “perfiles del usuario”:

 Nombre: Antonio Rodolfo de la Madrid  
Edad: 30  
Estatura: 1.75  
Responsable: Diego Luna  
Parentesco: Tio

- Visualizar Marco
- Ajustar/Desajustar

El perfil Seleccionado se muestra en marco principal

Perfiles disponibles del usuario.

< Perfiles

- First
- Sugar
- Tests
- Lipid
- General
- History and test

Glucosa: 5.6ml  
weight: 78 kg  
Height: 1.8 mts  
Blood: A+  
Age: 40  
Sodio: 0.4 mg

