

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE

ESCUELA POLITÉCNICA SUPERIOR DE ELCHE

GRADO EN INGENIERÍA INFORMÁTICA EN
TECNOLOGÍAS DE LA INFORMACIÓN



“Desarrollo de una aplicación de Realidad
Aumentada para ofrecer información
actualizada de fármacos a partir de sus envases”

TRABAJO FIN DE GRADO

Julio 2022

AUTOR: Mohamed Mohamed Mohamed

DIRECTOR: Manuel Quesada Martínez

Índice de contenido

1.- Introducción.....	6
1.2.- Tecnologías de la Información en el ámbito de la salud	7
1.3.- La revolución de la Realidad Aumentada	8
1.4.- Objetivos.....	11
2.- Materiales y métodos	13
2.1.- Realidad Aumentada	13
2.1.1.- Historia e inicio de la Realidad Aumentada.....	13
2.1.2.- Componentes hardware y software en RA.....	14
2.1.3.- Tipos de Realidad Aumentada	15
2.2.- Aplicaciones para la gestión de medicamentos	17
2.2.1.- Potencial interés del uso de RA en la salud	17
2.2.2.- Funcionalidad de aplicaciones para la gestión de medicamentos .	19
2.2.3.- Interoperabilidad de datos e integración con terceros.....	23
2.2.3.1.- Uso de la REST API de AEMPS CIMA.....	23
2.2.3.2.- Web Scraping para consultar interacciones	24
2.4.- Herramientas para el desarrollo de la RA.....	25
2.4.1.- Principales SDKs de RA: ARCore, ARKit y Vuforia.....	26
2.4.2.- Limitaciones de los patrones de RA para analizar medicamentos .	27
2.4.2.1.- Limitaciones de los patrones basados en imágenes	27
2.5. El Código Nacional del medicamento como patrón	28
2.6. Reconocimiento de patrones basados en texto	29
2.6.1. Patrones de reconocimiento de texto en Vuforia.....	29
2.6.2. OCR y Cognitive Services de Microsoft	29
2.7.- Unity para el desarrollo de la app móvil de RA.....	31
2.8.- Otras herramientas y utilizadas	32
2.9.- Propuesta de solución	34
3.- Resultados y discusión.....	35
3.1.- Análisis y diseño del software.....	35
3.1.1.- Metodología de desarrollo.....	35
3.1.2.- Captura de requisitos y casos de uso	38
3.1.3.- Diagrama de secuencia y diagrama de estado	39
3.1.4.- Diseño de la base de datos.....	40
3.1.5.- Prototipado.....	42

3.2.- Implementación en Unity de patrones OCR de RA	43
3.2.1.- Creación del proyecto Computer Vision	43
3.2.2.- Implementación del OCR de Azure en Unity	44
3.3.- Implementación de la aplicación móvil de RA	49
3.3.1.- Recuperando información desde la web de CIMA	49
3.3.2.- Población de la BBDD y acceso desde Unity	51
3.3.3.- Interfaz de usuario de la aplicación de RA	53
3.4.- Ejemplo de uso de la herramienta	55
3.4.1.- Inicio de sesión	55
3.4.2.- Registro de usuario	56
3.4.3.- Elección de Alergias	56
3.4.4.- Información sobre un medicamento	57
3.4.5.- Interacción entre medicamentos	58
3.4.6.- Cierre de sesión	59
4. Conclusiones y trabajo futuro	60
4.1.- Trabajo futuro	61
5. Bibliografía	63
ANEXO I: Descripción de los casos de uso	68
ANEXO II: Uso de patrones Vuforia	72

RESUMEN

En este proyecto hace uso de la tecnología de Realidad Aumentada (RA) que se espera que en los próximos años tenga más presencia entre la población, con grandes compañías como Apple, Microsoft, Xiaomi, etc. apostando por ella. La RA es una tecnología con un amplio espectro de utilización, desde el ámbito del entretenimiento hasta el ámbito de la salud, siendo este último el ámbito al que está enfocado este proyecto.

El principal objetivo de este proyecto es crear una aplicación de RA, para brindar al usuario información (extraída de fuentes fiables) sobre los medicamentos a partir del reconocimiento automático de sus envases. Esta labor ha requerido el análisis de aplicaciones y tecnologías relacionadas para definir un marco de trabajo que permita construir una versión funcional de la misma. Se han abordado tareas como la necesidad de mejorar los patrones de RA actualmente existentes, mediante la integración de la RA y el OCR de los Cognitive Services de Microsoft Azure para definir patrones basados en texto que nos permitan explotar el Código Nacional de los medicamentos para consultar datos fiables y mantenidos por terceros sobre información de un medicamento y sus interacciones, ya sea con alimentos o entre medicamentos. Concretamente, se han integrado en la aplicación consultas dinámicas a la web del Centro de Información de Medicamentos (CIMA) mediante la API REST y datos extraídos de otras webs usando técnicas de Web Scraping y almacenados de forma local. Implementar la primera versión de la aplicación ha requerido planificar y seguir el proceso mediante una metodología ágil de desarrollo de Software, pasando por las necesarias fases de análisis y diseño de software. Finalmente, hemos implementado la aplicación móvil de RA usando Unity, lo que nos ha permitido diseñar una aplicación dinámica e interactiva, creando una experiencia placentera para el usuario ya que no requerirá el acceso a páginas web a través de navegadores que requieren interacciones adicionales como rellenar un formulario.

En conclusión, este proyecto ha sido un reto interesante para nosotros ya que nos enfrentábamos a tecnologías y herramientas que no habíamos utilizado antes. Se ha implementado una aplicación funcional de utilidad para el paciente, ya que combina diferentes tecnologías alrededor de la RA para aumentar los envases de los medicamentos. Además, se han identificado diferentes vías de trabajo futuro que permitirían, por ejemplo, el lanzamiento de la aplicación al mercado, o publicar e integrar con las APIs de RA actuales la propuesta de patrones de RA basados en texto aquí desarrollada.

Palabras clave: realidad aumentada, RA, salud, medicamentos, interacción, unity, ocr, azure, cognitive services, cima

AGRADECIMIENTOS

Me gustaría dar gracias a mi familia por permitirme llegar hasta aquí, en especial a mis padres por tener paciencia y comprensión conmigo, y por criarme en un buen entorno familiar. Gracias, a mi hermano Hennan por todo el apoyo que me ha dado en estos años. Gracias, a mis abuelos por haberme criado, en los campamentos de refugiados saharauis, cuando mis padres no estaban.

Además, quiero decir que ha sido un gran placer el haber compartido estos años de carrera con mis compañeros de clase. Igualmente, me gustaría agradecer a mis profesores, en especial a Manuel Quesada Martínez por darme la oportunidad de realizar este trabajo de fin de grado con él cómo tutor, agradecerle por toda la ayuda que me ha brindado, la confianza puesta en mí y la gran dedicación en este proyecto.

Por último y no menos importante, agradecer a Alexis por la gran amistad y apoyo que me ha mostrado a lo largo de la carrera, lo cual ha permitido que estos años de carrera fueran más llevaderos. Gracias, a Julio, Horlando, Dmitry y Roberto por las risas y los grandes momentos, los cuales no se olvidarán.



1.- Introducción

La tecnología avanza a pasos agigantados y se introduce en el tejido de la sociedad de formas nunca vistas. Internet, móviles, ordenadores... son avances tecnológicos que nos llevan a un mundo cada vez más conectado, lo cual cambia la forma en la que nos relacionamos y comunicamos.

La palabra tecnología proviene de la unión de dos palabras griegas: «*tekne*», que es técnica, arte y, «*logia*», que significa destreza sobre algo. Desde un punto de vista global, la tecnología puede ser entendida como “*una respuesta al deseo del hombre de transformar el medio y mejorar su calidad de vida. Incluye conocimientos y técnicas desarrolladas a lo largo del tiempo que se utilizan de manera organizada con el fin de satisfacer alguna necesidad*” (Paula Nicole Roldán, 2017).

El interés del ser humano en la tecnología ha evolucionado a la par que ella. Por ejemplo, en el 2004 el 25% de adolescentes consideraban una de sus actividades favoritas la interacción con la computadora, cuando anteriormente era solo del 2% (Sobkin & Evstigneeva, 2004). Los avances en los dispositivos móviles con la llegada de los smartphones integrando una pantalla táctil y reduciendo su tamaño (ver Figura 1), cambió el comportamiento de millones de usuarios. En 2007, la aparición del primer iPhone, protagonizó un cambio radical en el día a día de las personas. En palabras de su creador Steve Jobs, “*Un iPod. Un teléfono. Un dispositivo para Internet. No son tres cosas distintas, se trata de un único dispositivo. Y vamos a llamarlo... iPhone*” (Steve Jobs, 2007). El impacto de este avance tecnológico fue tal que, en el año 2012 en España, entre los menores de 10 a 15 años el 87.1% ya utilizaba Internet y el 65,8% tenían un teléfono móvil (Jiménez et al., 2012).



Figura 1: Evolución del tamaño de los teléfonos móviles.

Una de las claves que potencia el uso cada vez más extendido de herramientas tecnológicas es su continua evolución. La mayoría de las personas jóvenes se pasan gran parte del tiempo delante de una pantalla digital, ya sea de un dispositivo móvil o de un ordenador, y esto es algo que hace más de 3 décadas atrás no pasaba. En los últimos años, la tecnología ha cambiado y seguirá

cambiando el cómo nos comunicamos entre nuestros conocidos y no tan conocidos. Podemos decir que, poco a poco, la tecnología se ha convertido en una parte indispensable de nosotros que se ha vuelto omnipresente en nuestra existencia. En un futuro inmediato, el avance de la tecnología continuará creciente y su incursión seguirá a otras partes de nuestra vida que aún no se han explorado. Con este trabajo se pretenden avanzar en esta línea, y para ello se desarrollará una aplicación que integre tecnologías innovadoras en procesos cotidianos relacionados con la salud y los medicamentos.

1.2.- Tecnologías de la Información en el ámbito de la salud

En el ámbito de la salud la HIT (Health Information Technology) o Tecnología de la Información Sanitaria en español, es el área de la sanidad que supervisa los sistemas tecnológicos que utilizan los profesionales sanitarios para gestionar los datos de los pacientes. *“HIT se refiere a los sistemas electrónicos que los profesionales sanitarios, y cada vez más los pacientes, utilizan para almacenar, compartir y analizar la información”* (ONC Health IT, 2011).

La HIT incluye tecnologías como las historias clínicas electrónicas (HCE) y las recetas electrónicas, así como herramientas tecnológicas que ayudan a los pacientes a alcanzar objetivos de salud como dejar de fumar o controlar la diabetes. Gracias a esta fusión de la tecnología y la asistencia sanitaria, se nos ha permitido crear HCE más precisas que permiten el seguimiento de los pacientes en diferentes centros sanitarios. Además, este enfoque ha dado a los pacientes un mayor control sobre su salud a través de aplicaciones y un mayor acceso a la información (ONC Health IT, 2011).

Por ejemplo, en la Figura 2 se puede apreciar el uso de una aplicación con un sistema de receta electrónica. Esta aplicación es un ejemplo de cómo el uso de una herramienta HIT puede felicitarle la vida tanto a un paciente como a su profesional sanitario. En este caso, el uso de recetas electrónicas tiene muchas ventajas, por ejemplo, facilitar el seguimiento del historial farmacoterapéutico al personal sanitario o dar accesibilidad al paciente a sus tratamientos (María Esther Fernández Sal, 2021).

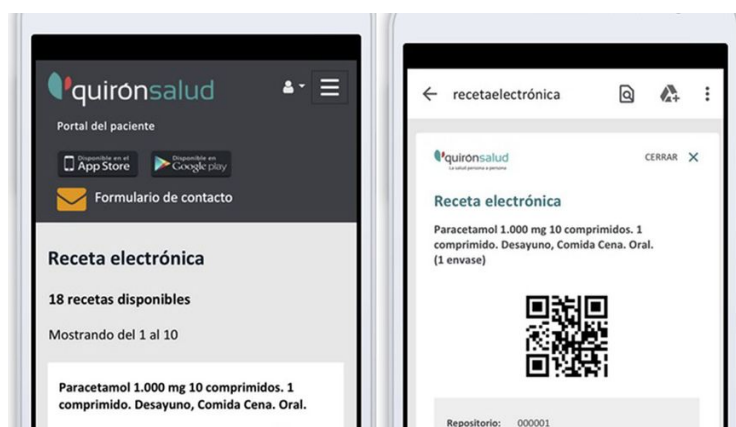


Figura 2: Ejemplo de receta electrónica de la aplicación QuirónSalud (Quironsalud.es, 2020).

Uno de los problemas a los que ha tenido que enfrentarse el sector de la HIT ha sido el tratamiento de datos sensibles de los usuarios. Aunque con los últimos avances el sector de la HIT abarca varios tipos de sistemas y de tecnología, en todos los casos ha enfocado sus esfuerzos en mejorar la atención del paciente y al mismo tiempo mantener su privacidad. Los avances en las HIT seguras han permitido a los médicos y a otros miembros del equipo de atención al paciente tener una comunicación mejor que nunca.

1.3.- La revolución de la Realidad Aumentada

A medida que la tecnología se desarrolla y expande, el campo de las tecnologías de la Información sanitaria también lo hará. Los trabajadores de este campo pronto trabajarán con tecnología de vanguardia como monitores de pacientes que informan a los médicos a través de un smartphone; estas notificaciones son a menudo generadas por medio de procesos internos que utilizan técnicas avanzadas de Inteligencia Artificial.

En este trabajo vamos a dirigir nuestra atención al uso de tecnologías de Realidad Aumentada, y en particular su impacto en el ámbito de la salud. La Realidad Aumentada (RA) se define como un recurso tecnológico que ofrece experiencias interactivas al usuario a partir de la combinación entre la dimensión virtual y la física, con la utilización de dispositivos digitales (Tatiana Grapsas, 2019).

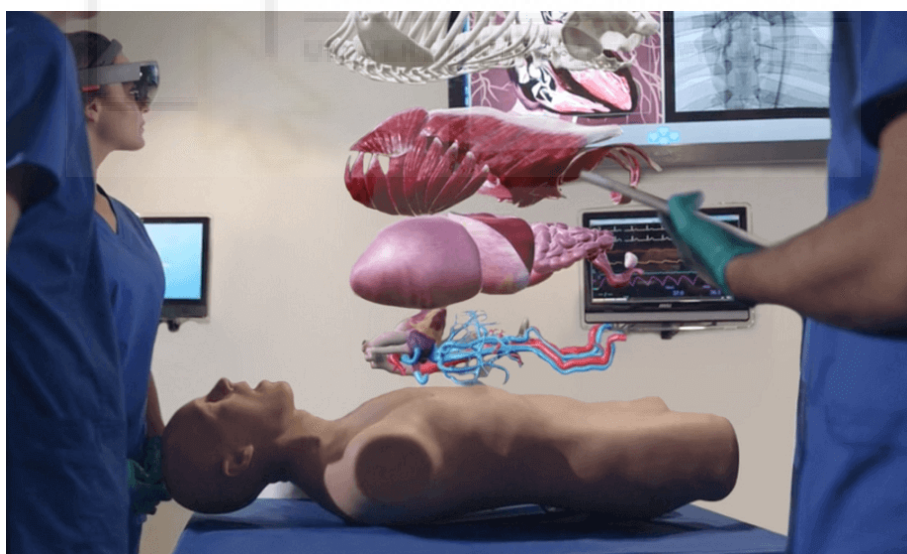


Figura 3: Ejemplo del uso de la RA en el ámbito de la educación sanitaria (Anastasia Morozova, 2022).

La Realidad Aumentada tiene mucho potencial, ya que puede beneficiar a la sociedad en la que vivimos desde diferentes puntos de vista como el entretenimiento, educación, moda, marketing, etc. El ámbito de la salud es uno de los primeros sectores en adoptar la Realidad Aumentada. Por ejemplo, hoy en día, muchas enfermeras y médicos la utilizan para mejorar la educación y los resultados de los pacientes (Tatiana Grapsas, 2019). En la Figura 3, podemos

ver como mediante el uso de la RA, el personal sanitario o los alumnos de medicina, pueden ver de forma detallada las diferentes partes del cuerpo humano en formato 3D... La RA también se utilizan para la educación de futuros profesionales sanitarios, permitiéndoles visualizar e interactuar con representaciones tridimensionales del cuerpo humano y su funcionamiento (Anastasia Morozova, 2022).

Desde el punto de vista de la ayuda al paciente, la RA permiten a los profesionales médicos ayudar a sus pacientes para entender el funcionamiento de los medicamentos o los procedimientos quirúrgicos a los que pueda ser sometido (Dean Madison, 2018). Aunque actualmente el número de soluciones destinadas a los pacientes es muy limitado, algunos trabajos como el que se muestra en la Figura 4 a) (extraída de (Lisa Nissen et al., 2018)) proponen el uso de la RA como una herramienta novedosa para mejorar la alfabetización en salud sobre antihipertensivos en personas que viven con diabetes mellitus tipo 2. También es una tecnología utilizada para el tratamiento de fobias, con aplicaciones como la de la Figura 4 b) (extraída de (AVRspot channel, 2018)) que permite al usuario simular la presencia de arañas sobre su mano que son superpuestas en la pantalla de un dispositivo móvil. En línea con (Lisa Nissen et al., 2018), en 2020 se presentó en (Martínez Parra, 2020) una solución de RA que utilizaba patrones basados en las cajas de los medicamentos para detectar interacciones entre ellos; sin embargo, la solución tecnológica planteada en (Martínez Parra, 2020) tenía problemas de escalabilidad ya que debían definirse patrones para cada caja, por lo que supone una limitación para la inclusión de nuevos productos o cambios en las cajas de los ya existentes.

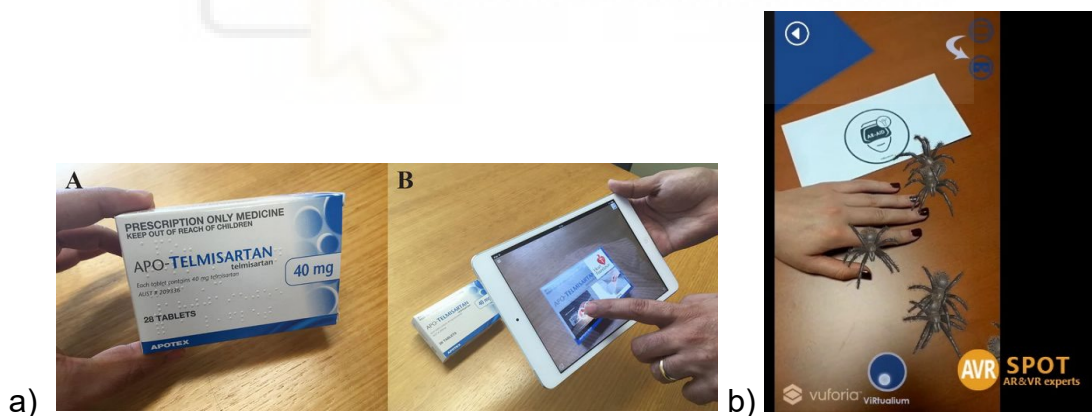


Figura 4: Ejemplo de uso de la RA para el tratamiento de pacientes.

Al hablar de RA, es inevitable su comparación con la Realidad Virtual (RV). Por ejemplo, aplicaciones como las de la Figura 4 b) habían sido previamente en entornos de RV y ahora están siendo migradas a RA. Mientras que la RA es una tecnología que amplía nuestro mundo físico añadiendo ciertas capas de información digital sobre él, la RV crea entornos completamente digitales para sustituir al real por uno virtual. Es decir, como se muestra en las Figuras 3 y 4 la RA añade sobre entorno real diferentes sonidos, gráficos, objetos 3D... (Alexander Sokhanych, 2021). Desde el punto de vista de su popularidad, en los

últimos años la RA ha conseguido un crecimiento más rápido que las tecnologías de RV gracias en parte a su accesibilidad.

Los dispositivos móviles actuales, con su gran capacidad de procesamiento y cámaras de alta calidad, se han convertido en la plataforma perfecta para consumir la RA. Esto hace que muchos usuarios, y casi sin quererlo, llevan en sus bolsillos un dispositivo tecnológico que pueden utilizar para acceder a las aplicaciones de RA sin necesidad de ningún accesorio adicional. Todo esto hace que la RA, poco a poco, esté más presente en nuestra vida, de una forma u otra. En la Figura 5 se puede observar la tendencia de crecimiento ascendente de usuarios activos de RA móvil en todo el mundo y su predicción para los próximos años.

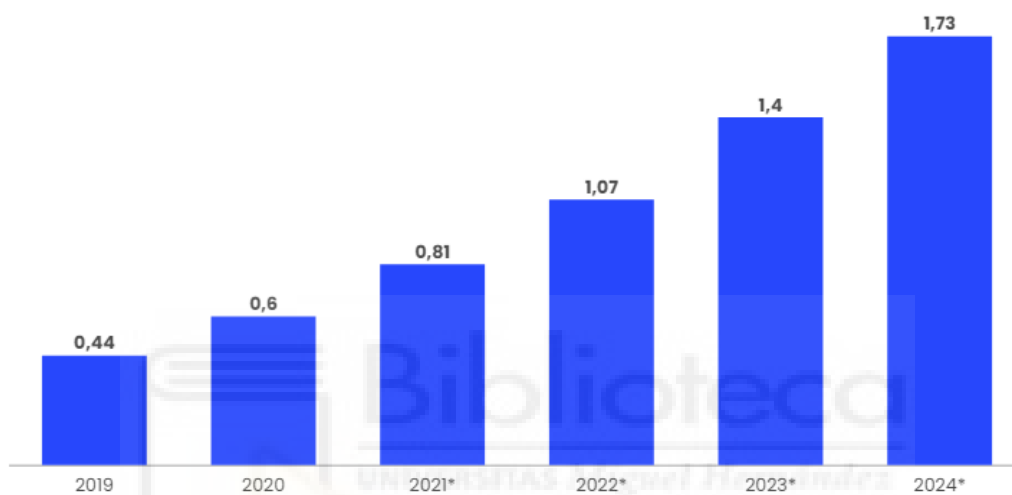


Figura 5: Número de usuarios activos de Realidad Aumentada móvil en todo el mundo desde 2019 hasta 2020 y estimación para 2021-2024 (en miles de millones) (Softtek.eu, 2021).

No obstante, pese a la creciente popularidad de esta tecnología aún queda camino por recorrer, ya que aún no es una tecnología muy extendida entre profesionales médicos, empresas y pacientes. Eso significa, que en este campo hay una enorme oportunidad para que nuevas empresas o individuos, puedan crear nuevos productos o aplicaciones innovadoras explotando las capacidades de la RA. Creemos por ello que la historia de la RA acaba de comenzar y, dada la carencia de aplicaciones de RA destinadas a su uso por parte de los pacientes, creemos que sería una contribución interesante ofrecer nuevas soluciones que contribuyan a este campo.

Consideramos por tanto interesante, explorar y desarrollar una solución de RA para ofrecer a los pacientes información sobre los medicamentos de sus tratamientos para minimizar el riesgo de los pacientes, y con ello podríamos beneficiar nuestra sociedad en línea con el Objetivo de Desarrollo Sostenible 3 de Salud y Bienestar promovido por la Naciones Unidas (Naciones Unidas, 2022).

1.4.- Objetivos

En este proyecto se pretende utilizar la RA como una herramienta de gestión de medicamentos e información farmacológica, siendo nuestro principal objetivo:

- **Obj. General:** crear una aplicación de RA, para brindar al usuario información (extraída de fuentes fiables) de los fármacos prescritos por su médico. Esta información se mostrará a partir del reconocimiento automático de los envases de los medicamentos que está tomando.

Para conseguir este objetivo general será necesario alcanzar los siguientes objetivos secundarios:

- Trabajo de investigación para:
 - **Obj1:** definir un marco de trabajo a partir del análisis de tecnologías (SDKs y frameworks de RA...) y equipos (dispositivos móviles compatibles...) que nos permitan ir más allá de un marco teórico e implementar la aplicación en un dispositivo móvil.
 - **Obj2:** identificar aplicaciones relacionadas y analizar su funcionalidad desde el punto de vista de la gestión de medicamentos y ofrecer su información al usuario.
 - **Obj3:** identificar fuentes de información estructura que puedan ser consultadas y explotadas de manera automática e integrada por nuestra aplicación haciéndola escalable a nuevos medicamentos.
- Diseño y desarrollo de software:
 - **Obj4:** desarrollo de módulos para la explotación e integración de la información sobre medicamentos contenida en fuentes de datos externas y mantenidas por terceros.
 - **Obj5:** análisis, diseño e implementación para desarrollar un primer prototipo funcional de la aplicación de RA que pueda desplegarse en un dispositivo móvil.

Aunque este proyecto pertenece a la misma línea de investigación iniciada en (Martínez Parra, 2020), en este TFG se pretende abordar uno de los trabajos futuros allí planteados y que supone el reto de utilizar tecnología de RA en un entorno escalable en dos sentidos: (1) explorar patrones de RA que no estén basados en imágenes particulares sino en otro tipo de información, y (2) explorar fuentes de información que puedan ser enlazadas automáticamente con los patrones descritos en (1). Por ello, aunque Obj2 y Obj5 son una continuación de (Martínez Parra, 2020) las necesidades de este TFG requieren definir objetivos nuevos como Obj1, Obj3 y Obj4.

Además, personalmente, este proyecto me ha supuesto un reto, ya que es la primera vez que desarrollo para RA y trabajo con sus diferentes tecnologías relacionadas como, por ejemplo, la plataforma Unity. También es la primera vez que me enfrento a problemas cuya solución requiere del uso de servicios web reales como los *Cognitive Services* de Microsoft Azure, y la integración de datos reales como los ofrecidos por CIMA (Centro de Información de Medicamentos).



2.- Materiales y métodos

Cuando escuchamos sobre los nuevos desarrollos tecnológicos, ese entusiasmo que sentimos, poco a poco, se desvanece ya que, por lo general, esos inventos tienden a tener poco impacto en nuestras vidas diarias. Pero de vez en cuando, hay tecnologías que consiguen despegar para marcarse un jonrón, y una de esas es la Realidad aumentada.

2.1.- Realidad Aumentada

Continuando con el análisis de esta tecnología iniciado en el capítulo anterior, en esta sección profundizaremos más en ella.

2.1.1.- Historia e inicio de la Realidad Aumentada

Muy a menudo la Realidad Aumentada (RA) es presentada como una tecnología “futurista” o “del futuro”. Sin embargo, eso está muy lejos de la realidad dado que ese concepto existe desde hace años. Ya en la década de los 90, la visualización *head-up* podía ofrecer una experiencia básica de RA en los aviones militares, proporcionando a los pilotos información sobre la dirección, la velocidad y la altitud de los aviones (Ebrary.net, 2021) (ver Figura 6.a). Asimismo, en el 2009, el equipo de Pravaç Mistry del MIT Media Lab presentó SixthSense, la primera interfaz gestual portátil del mundo que permitía una integración casi perfecta de la información digital con la realidad (Pranav Mistry, 2021). En el 2013, se introdujeron las primeras gafas de RA del mundo, las Google Glass de Alphabet. Este dispositivo prometía mucho, sin embargo, no llegaron a cuajar porque, por un lado, su gran coste limitaba su uso a un ámbito exclusivo empresarial y, por otro lado, tenían muchos fallos de diseño en cuanto a la usabilidad, por lo cual pasó de un proyecto estrella a tener un retiro silencioso (Allan Vélez, 2021).

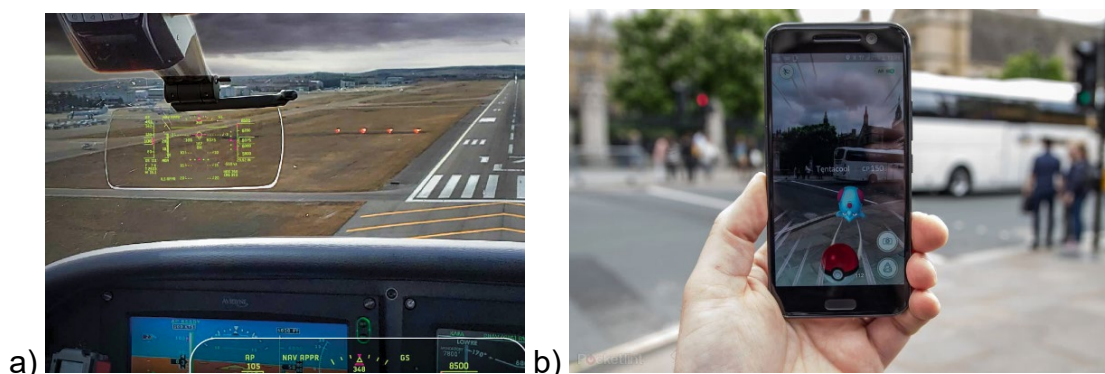


Figura 6: a) Head-up usado en aviación ofreciendo un añadido de información al piloto (Flying Contributor, 2019). b) Captura de la aplicación Pokemon GO (Chris Hall, 2022).

Iniciativas como (Allan Vélez, 2021; Ebrary.net, 2021; Pranav Mistry, 2021) pueden considerarse adelantadas a su tiempo, ya que el interés genuino por la RA empezó en el año 2016, cuando empezaron a aparecer los primeros

productos dirigidos al consumidor de a pie. Uno de los grandes ejemplos que impulsó el interés de una experiencia RA entre el público general fue Pokemon GO (Kavit Majithia, 2016), que fue una sensación mundial hasta el punto de que en la actualidad lleva más de 630 millones de descargas y ha generado más de 5 mil millones de dólares (Kieran Harris, 2021). Toda esta breve introducción a la historia de RA nos ha enseñado que para que una tecnología triunfe de verdad, tiene que tener éxito entre todos los públicos. De hecho, desde 2016 las empresas empezaron a reconocer el valor de esta tecnología, y la RA evolucionó allanando el terreno tanto para los consumidores como para las empresas.

2.1.2.- Componentes hardware y software en RA

Como hemos visto, la RA junta el mundo real y virtual para crear experiencias más inmersivas para los usuarios. La forma más común de RA se base en añadir datos u objetos digitales al mundo real mediante las cámaras de los dispositivos móviles o las gafas inteligentes para RA. Los componentes más importantes de la RA son (Fernanda Rodríguez, 2020):

- **Hardware:** su principal característica es captar y visualizar los datos.
 - **Entrada – Sensores:** para poder integrar el mundo real con el virtual, los sistemas de RA deben de tener la capacidad de captar de forma rápida y precisa la información sobre el mundo real. Los sensores nos permiten recoger la información y enviarla para su procesado. Los sensores pueden ser Ópticos (cámaras), Magnéticos (compases), Inerciales (acelerómetro, giroscopio) o de otro tipo (ej., GPS).
 - **Salida – Pantalla:** la RA necesita pantallas para poder ver el contenido virtual en el mundo real. Cuanta más calidad tiene la pantalla, más realista será el contenido. Los equipos para visualizar la RA se pueden dividir en Vestibles (cascos, gafas, visualización head-up, etc.) y No Vestibles (móviles, tabletas, proyectores, etc.).
- **Software:** su principal característica es interpretar los datos captados para transformarlos y aumentarlos.
 - **De alto nivel:** el software de RA son las aplicaciones y herramientas que podemos utilizar para acceder a la RA. Hoy en día existen herramientas de software que nos ayudan a desarrollar aplicaciones de RA, por ejemplo, ARKit, ARCore, Vuforia, ARtoolKit, EasyAR, LayAR, etc.
 - **De bajo nivel:** consiste en varias áreas, que pueden variar según la aplicación o la necesidad del desarrollador, por ejemplo, el procesamiento de imágenes, la Human-Computer Interaction o HCI, bibliotecas de programación, etc.

Según crezca el interés en la RA, los equipos que utilizamos para acceder a ella irán innovando en el futuro. Por ejemplo, hoy en día vemos más lentes y hardware inteligentes apareciendo en el mercado. En términos del hardware, recientemente, se está apostando por el desarrollo de gafas de RA, y en la actualidad destacan modelos como las HoloLens de Microsoft (Two Reality, 2021). Además, como pasó con los dispositivos móviles, el fabricante que podría revolucionar la RA es Apple quien ya está desarrollando sus gafas de RA (AppleInsider.com, 2022). No obstante, por el momento el uso de este tipo de dispositivos es exclusivamente empresarial ya que tienen un coste elevado. En cualquier caso, estas gafas, en un futuro, podrían destronar a los móviles en cuanto a ser el principal dispositivo para el consumo de AR, ya que permiten una experiencia más "manos libres". En la Figura 7 se muestran la evolución de la venta de cascos de realidad virtual y aumentada en los últimos años, donde se puede apreciar el potencial de la RA.

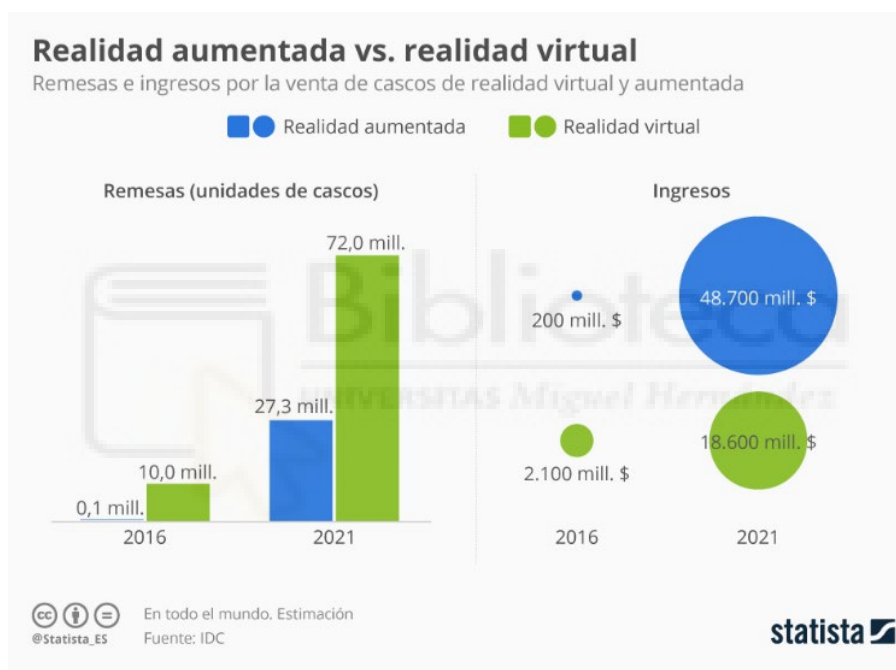


Figura 7: Realidad Aumentada vs Realidad Virtual. Remesas e ingresos por la venta de cascos de realidad virtual y aumentada (J.M. Sánchez, 2017).

2.1.3.- Tipos de Realidad Aumentada

El auge de la RA en años recientes ha causado la invención de diferentes tipos de aumento. Hay que comprender que la RA no es una tecnología monolítica. Es decir, las sutiles diferencias de implementación pueden lograr un papel fundamental en el éxito de una tecnología cuando esta se usa en un caso concreto. Por ejemplo, la RA es más que una herramienta que se utiliza en los juegos de los dispositivos móviles, también se ha comenzado a utilizar en los últimos años para otros casos. Por ejemplo, la RA se está utilizando en la asistencia a los cirujanos en los procedimientos o incluso para ayudar a los ingenieros y mecánicos en sus labores (Dr. Emilio Vicente & Dra. Yolanda Quijano, 2022; Ivy Decker, 2018).

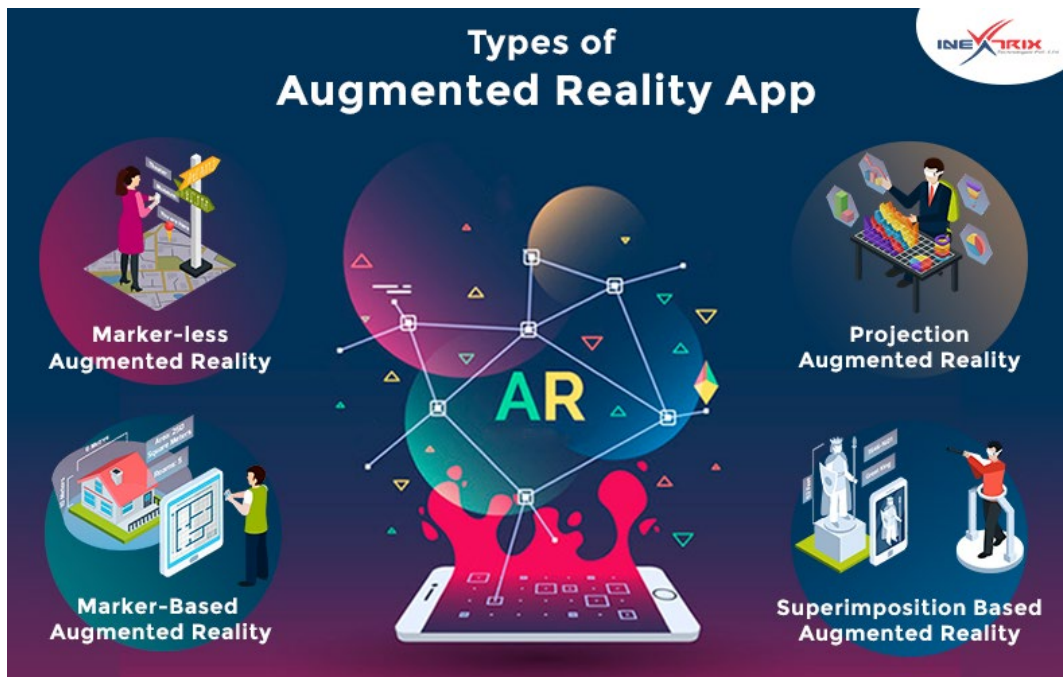


Figura 8: Tipos de Realidad Aumentada (INEXTRIX.com, 2019).

En la Figura 8 se pueden observar los distintos tipos de categorías en las que se puede clasificar las técnicas de RA. A continuación, explicamos en que consiste cada una, sus diferentes objetivos y casos de uso (Raykarimi.com, 2020):

- **RA basada en marcadores:** como su nombre indica, utiliza marcadores para el reconocimiento de imágenes a través de la cámara del móvil. Un marcador podría ser cualquier cosa, desde un código QR impreso hasta imágenes o texto. También puede calcular la posición y orientación de un marcador para posicionar el contenido. Así, una imagen puede aumentarse con un modelo 3D quedando asociadas por el marcador.
- **RA sin uso de marcadores:** también conocida como RA basada en la ubicación o en la posición, utiliza las cámaras, GPS, giroscopio, brújula y acelerómetro del dispositivo móvil para proporcionar datos basados en la ubicación del usuario. Estos datos determinarán el contenido de RA que se pueda encontrar en una zona determinada. Este tipo de RA suele producir mapas y direcciones para que el usuario se pueda guiar.
- **RA basada en proyecciones:** proyecta luz artificial sobre superficies físicas, esta tecnología puede ser la más atractiva ya que, en algunos casos, nos permite interactuar con la proyección. Por ejemplo, las proyecciones de las películas de Star Wars serían un ejemplo de este tipo.
- **RA basada en superposición:** puede sustituir la vista original por una aumentada, de manera total o parcial. El reconocimiento de objetos juega un papel vital, ya que sin él todo el concepto es sencillamente imposible. La aplicación IKEA Place es un ejemplo de este tipo, la cual nos permite colocar muebles digitales (con proporción 1 a 1 a los muebles reales) en el espacio de nuestra habitación (COSMOS, 2018).

2.2.- Aplicaciones para la gestión de medicamentos

En esta sección analizaremos la situación actual del uso de la RA como tecnología y su uso en aplicaciones relacionadas con la gestión de medicamentos.

2.2.1.- Potencial interés del uso de RA en la salud

Como hemos visto, la RA es una tecnología prometedora y con un gran futuro por delante. En cuanto al ámbito de la salud vemos que el mercado mundial de la RA en este ámbito se valoró en 609,60 millones de dólares y con una previsión de crecimiento entorno a los 4.237,60 millones de dólares en 2026 (FutureVisual.com, 2021). Estas previsiones nos indican que la RA en el ámbito de la salud no se trata de una moda pasajera, sino que ha que ha llegado para quedarse.

Aunque la RA, aún no ha acabado su despliegue en el mundo de la salud y medicina, actualmente, la RA puede tener un impacto positivo en todas las áreas del sector sanitario, desde el desarrollo de las habilidades de la próxima generación de médicos hasta la mejora de la detección, el diagnóstico y el tratamiento de las enfermedades. En (FutureVisual.com, 2021; IAT.es, 2020) se muestra como los diferentes tipos de RA se aplican en diferentes ámbitos sanitarios que resumimos a continuación:

- **Telemedicina:** permite a los profesionales de la salud evaluar, diagnosticar y tratar a los pacientes a distancia. Con la ayuda de la RA el profesional puede mostrarle al paciente cuál es el foco del dolor o dónde se hará la incisión.
- **Descubrimiento de medicamentos:** la RA se puede usar para facilitar el descubrimiento de medicamentos de forma más rápida y precisa. Esta tecnología, facilita el proceso de I+D+i en las empresas farmacéuticas para el descubrimiento de medicamentos y la investigación de múltiples tipos de enfermedades. Los científicos y otros expertos de la salud pueden utilizar la RA para evaluar datos genéticos relacionados con virus específicos. Por ejemplo, la empresa Sygnature Discovery con sede en Inglaterra, ha llegado a la fase de prototipado para un producto RA de este ámbito (Naveen Joshi, 2021).
- **Diagnóstico de enfermedades por imagen:** es uno de los ámbitos más importantes de la sanidad, ya que ayuda a los profesionales de la salud a detectar y diagnosticar enfermedades con precisión, desde las cardiopatías hasta el cáncer. Con la RA, por ejemplo, se puede realizar un mapeo detallado para el marcado preoperatorio de los límites de un tumor, esto nos ayuda a comprender la estructura de un tumor complejo.
- **Educación:** esta es y será una de las principales funciones de la RA en cuanto al ámbito de la salud. La RA ayudará en la formación de los profesionales de la salud, pues gracias a ella, se puede practicar los

conocimientos teóricos que se han impartido en un marco virtual. Con esto se puede reducir el coste al no necesitar materiales como maniqués y se logra incrementar la interacción y las capacidades de los alumnos. Tenemos ejemplos como RAFODIUN (García Jiménez & Casado Parada, 2019), el cual se trata de un proyecto I+D financiado por el Ministerio de Economía y puesto en marcha en la Facultad de Medicina de la Universidad de Sevilla. Este proyecto permite a los estudiantes aplicar contenido de RA en su formación.

- **Simulación para practicar operaciones quirúrgicas:** la RA facilita la preparación y el avance de operaciones de cirugía. Por ejemplo, SNAP (*Surgical Navigation Advanced Platform*) (Medgadget.com, 2014) es un producto que se integra con la tecnología de los quirófanos para ofrecer funciones avanzadas de 3D y RA. Con ella, los cirujanos pueden mejorar el rendimiento de sus cirugías y la RA permite mantener la precisión durante todo el procedimiento y simular todas las fases de cirugías complejas.
- **Terapias de rehabilitación:** en línea con el ejemplo mostrado en la Figura 4 b), con al RA se puede diseñar y visualizar situaciones virtuales seguras que después se pueden aplicar a una situación real para mejorar la recuperación del paciente de una manera más eficiente. Otros ejemplos, se pueden utilizar para el tratamiento del equilibrio y la prevención de caídas en geriatría o el tratamiento para recuperar la funcionalidad de las extremidades inferiores y superiores en los accidentes cerebrovasculares.
- **Localización de venas:** por ejemplo, con Accuvein (Accuvein.com, 2022) se ha logrado superar una de las situaciones más desagradables que tienden a sufrir los pacientes, los pinchazos fallidos a la hora de buscar venas. Accuvein muestra una imagen de la vasculatura y la superpone a la piel, ayudando así a los profesionales sanitarios para que encuentren de forma sencilla las venas a la hora de poner inyecciones o realizar análisis de sangre.

Aunque estos son solo pequeños ejemplos de la utilización de la RA en el ámbito de la salud, podemos ver como los beneficios tangibles ya están empezando a hacerse realidad y, por ende, el futuro de la RA en el sector sanitario nunca ha sido más saludable. Sin lugar a duda, aunque a la RA aún le queda camino por recorrer en el sector de la salud, estoy seguro de que, en un futuro no muy lejano, se podrán ver nuevos inventos que revolucionarán este sector. Además, estos inventos deberían beneficiar no solo a los profesionales, sino también a los pacientes gracias a que la tecnología y su incursión en el mundo cotidiano avanza a pasos agigantados. Por ejemplo, con el despliegue de la tecnología 5G (Yúbal Fernández, 2020), esperamos que la RA pase al siguiente nivel, pues será fundamental para abrir la puerta a nuevas prácticas de trabajo, por ejemplo, con el apoyo a la monitorización remota de los pacientes.

2.2.2.- Funcionalidad de aplicaciones para la gestión de medicamentos

En línea con los objetivos marcados en este proyecto, vamos a continuar nuestro análisis dirigiéndolo a descubrir que tipos de aplicaciones existen hoy en día dedicadas a ofrecer a los usuarios información sobre medicamentos como: ofrecer información general, las interacciones entre estos, si nos pueden indicar las alergias de medicamentos...

Para la búsqueda se ha utilizado Google Play, que es la plataforma de distribución digital de aplicaciones más grande de Android y es la más popular. No disponemos de dispositivos de Apple, por ello descartamos el uso de otras tiendas como Apple Store o similar. Además, se ha elegido Google Play porque es la tienda de dispositivos móviles con mayor número de usuarios y creemos que su uso es acertado.

En cuanto a la cadena de búsqueda, se ha utilizado como palabra de búsqueda "medicamentos". No hemos optado por búsquedas más específicas para no reducir el número de resultados, en su lugar proponemos como criterio de selección un análisis manual de las aplicaciones según los siguientes criterios: (1) que ofrezcan información sobre fármacos, (2) que ofrezcan información sobre interacciones y (3) que ofrezcan información sobre las alergias de los fármacos.

Durante el proceso de análisis, hemos visto que la mayoría de las aplicaciones analizadas o nos daban información sobre fármacos únicamente (por ejemplo, la aplicación Medicamentos (Morales Padrón, 2018) con más de 100 mil de descargas) o eran para el recordatorio para tomar los fármacos (por ejemplo, la aplicación Recordatorio de Medicación (MyTherapy, 2022) con más de 5 millones de descargas). Sin embargo, no hemos encontrado muchas aplicaciones que tengan la función de interacción con medicamentos o que nos ofrezcan información relacionada con alergias.

El análisis de aplicaciones anteriormente explicado nos ha permitido seleccionar cuatro aplicaciones cuya comparativa se incluye en la Tabla 1. Además de los tres criterios comentados, hemos incluido información sobre el uso y tipo de RA, así como si la aplicación es gratuita o requiere el pago por su uso. En la Figura 9 se incluye la captura de pantalla de estas cuatro aplicaciones.

Aplicación	¿De Pago?	Información de fármacos	Interacción fármacos	Alergias fármacos	RA
Vademecum	Si, pero gratuita para médicos colegiados	Si	Si	No	No
AEMPS CIMA	No	Si	No	No	Código barras
BOT PLUS	Si	Si	Si	No	No
iDoctus	Si	Si	Si	No	No

Tabla 1: Comparativa entre aplicaciones.

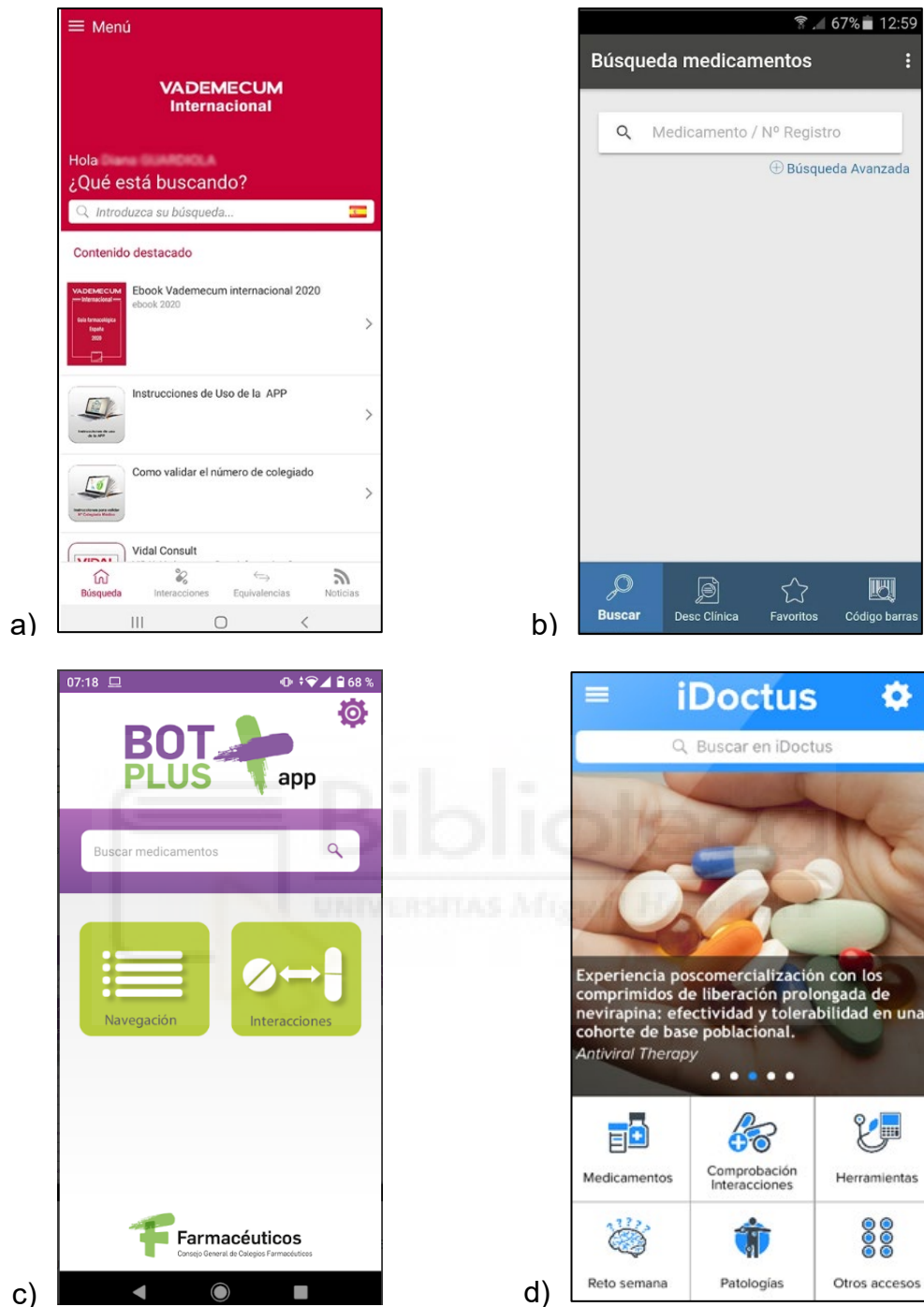


Figura 9: Captura de las cuatro aplicaciones seleccionadas para su análisis.

A continuación, se describen con más detalles los pros y contras de estas cuatro aplicaciones seleccionadas:

- **Vademecum** (Vademecum.es, 2011) (Figura 9 a)): es una aplicación de pago con opción gratuita para médicos colegiados. Ofrece dos módulos. El primero es el de la Guía Farmacológica, que nos permite buscar información sobre los fármacos y, el segundo es de Interacciones, que calcula las interacciones (basadas en la evidencia) entre dos o más medicamentos codificados según cuatro niveles gravedad.

El módulo de Interacciones tiene más de 12.300 medicamentos virtuales y más de 700 clases de interacción. La suscripción a cada módulo tiene carácter anual. El de Guía Farmacológica cuesta 29,99€ y el de Interacciones cuesta 14,99€. Esta aplicación no nos indica las alergias a los fármacos y no utiliza RA.

- **AEMPS CIMA** (AEMPS.gob.es, 2014) (Figura 9 b)): es una aplicación gratuita. Coordinada por la Agencia Española de Medicamentos y Productos Sanitarios (AEMPS) y Centro de Información online de Medicamentos (CIMA). Esta aplicación es mantenida por el gobierno y se creó con la intención de proporcionar puntualmente a los ciudadanos y a los profesionales de la salud toda la información posible sobre los medicamentos de forma comprensible para así, conseguir la correcta utilización de los mismos.

A día de hoy, AEMPS CIMA cuenta con 15.097 medicamentos, que además clasifica según los estados de autorizados, revocados o suspendidos temporalmente. Además, cuenta con 2.546 principios activos. Soporta el lenguaje Español e inglés. Como puntos negativos, la aplicación no tiene la función de interacción con medicamentos y no nos da la opción que nos indique las alergias a los medicamentos. Tampoco utiliza RA, solo utiliza un escaner de código de barras.

- **BOT PLUS** (Farmaceuticos.com, 2022) (Figura 9 c)): es una aplicación del Consejo General de Colegios Oficiales de Farmacéuticos, la cual está dirigida a farmacéuticos y profesionales de la salud. Esta aplicación les permite acceder de una manera fácil y rápida a la información sobre medicamentos y principios activos de uso humano.

Lo bueno de esta aplicación, es que incluye información de los medicamentos autorizados por la AEMPS, como la de CIMA, y los que están comercializados por los laboratorios titulares. Entre sus características y funcionalidades podemos encontrar: (1) búsqueda de información y administración sobre los fármacos y principios activos, (2) búsqueda de interacciones con medicamentos, (3) condiciones de disposición, facturación y precios de los fármacos, (4) advertencias de empleo y pictogramas y (5) posibilidad de uso *offline*.

Entre los puntos negativos encontrados: (1) no ofrece información sobre las alergias a los medicamentos, (2) no utiliza RA y (3) la aplicación es de pago. Para poder utilizar la aplicación se debe pagar una licencia con precio de 42€ para Farmacéuticos colegiados y 74€ para Farmacéuticos no colegiados. BOT PLUS también está en formato web, con diferentes características y precios de licencia respecto a la aplicación móvil.

- **iDoctus** (iDoctus.com, 2022) (Figura 9 d)): es una aplicación de pago y dirigida específicamente a los profesionales de la salud, la utilizan más de 440.000 profesionales en Latinoamérica y España. Esta aplicación utiliza un *vademecum* con más de 16.000 medicamentos, 2.600

principios activos y combinaciones utilizados en la farmacología. Además, cuenta con un comprobador de interacciones y tiene una funcionalidad llamada "Herramientas" la cual contiene más de 50 calculadoras y escalas que ayudan al profesional con la valoración de su paciente.

Es muy útil a la hora de recetar un medicamento u otro en función de cada paciente y de su historial clínico, porque ayuda a encontrar la dosis y el fármaco más adecuados y seguros en cada caso. Es una aplicación muy útil, para los profesionales a la hora de recetar medicamentos en función de cada paciente y su historial médico, pues con ello el profesional podrá encontrar la dosis y el medicamento correctos y seguros para cada paciente. También aporta a los profesionales artículos científicos de todas las especialidades, para que estos puedan estar al día de las novedades en el ámbito de la salud.

Podemos concluir que es una aplicación excelente para los profesionales de la salud, con muchas herramientas, información sobre medicamentos y sus interacciones. Por el contrario, es una aplicación de pago y restringida a su uso profesional, y no nos da la opción de que nos indique información relacionada con alergias. Tampoco utiliza RA.

Desde mi punto de vista, AEMPS CIMA es la mejor de las cuatro que hemos escogido y analizado. Se ha de tener en cuenta que, para la puesta en el mercado de un medicamento en España es obligatoria la autorización previa por la AEMPS o de la Comisión Europea, así como la comunicación de la puesta en el mercado por el Titular de la Autorización de Comercialización. Además, esta información se va actualizando durante el periodo de comercialización del medicamento con el fin de incluir la nueva información de la que se va disponiendo e incorporando a la información del medicamento una vez la AEMPS la evalúa y autoriza (AEMPS.gob.es, 2014). Dado que AEMPS CIMA está gestionada por AEMPS, esta aplicación cuenta en todo momento con la información actualizada del estado actual de todos los medicamentos autorizados en España. Es por ello, que se convierte en una aplicación mantenida por terceros, con fuentes fiables y que deseamos explotar como BBDD de nuestra aplicación de RA en la medida de lo posible.

Merece la pena destacar que ninguna de las aplicaciones analizadas hace un uso avanzado de RA, por lo que una aplicación que informe de medicamentos abstrayendo al usuario de complejas búsquedas y usando RA podría ser de utilidad.

Finalmente, por lo general, las aplicaciones disponibles no integran información sobre las alergias que un paciente puede tener a determinados principios activos que son el compuesto principal de los medicamentos. Por ello, consideramos que, además de la RA, permitir al usuario indicar a qué principios activos tiene alergia y utilizar la RA aumentar un medicamento indicando si tiene o no alergia a él es una contribución relevante.

2.2.3.- Interoperabilidad de datos e integración con terceros

En un principio, valoramos la posibilidad de extender las aplicaciones ya existentes. Sin embargo, todas las seleccionadas son aplicaciones propietarias no permiten integrar nuevos módulos. Ante esta situación modificamos nuestra pregunta, si no podemos extender las aplicaciones ya existentes ¿podemos construir una aplicación propia que se nutra de la información ya disponible y mantenida por terceros? Para ello, aunque no lo hemos incluido entre los criterios de selección anteriores, un aspecto importante en nuestro análisis es la capacidad de interoperabilidad de las aplicaciones disponibles.

2.2.3.1.- Uso de la REST API de AEMPS CIMA

El análisis de las aplicaciones móviles anteriores nos llevó en algunos casos como el de AEMPS CIMA hasta sus portales web corporativos. En estos sitios, además de la información contenida en las aplicaciones móviles ofrecían más información y funcionalidad. Por ejemplo, la web de AEMPS ofrece la posibilidad de realizar consultas a través de un formulario web. Además, hemos detectado que podemos explotar la REST API que utiliza CIMA para poder sacar toda la información de los medicamentos que proporciona la AEMPS (ver Figura 10).

Formato de los datos devueltos

Toda la información devuelta por los servicios REST descritos a continuación se encuentra en formato [JSON](#) y está codificada en UTF-8.

Las fechas devueltas estarán codificadas usando [Unix Epoch](#) (POSIX Time) GMT+2:00

Especificación

Medicamentos

Recursos

Recurso	Descripción
GET medicamentos?{condiciones}	Devuelve una lista de medicamentos para las condiciones enviadas
GET medicamento?{condiciones}	Devuelve la información de un medicamento pasando el nº de registro o el código nacional
POST buscarEnFichaTecnica	Recibe en formato JSON como cuerpo de la petición una serie de texto a buscar en las secciones y devuelve una lista de medicamentos

Figura 10: REST API de CIMA (AEMPS.gob.es, 2018).

La web CIMA de la AEMPS, implementa una interfaz REST, la cual nos puede permitir hacer consultas de la información de los medicamentos que están autorizados por la AEMPS. Con esta interfaz podremos, por ejemplo, conocer si un medicamento tiene algún problema de suministro o la lista de problemas de suministro en un momento dado. De forma general, como respuesta a la invocación de los servicios de la interfaz REST, esta nos devolverá un JSON codificado en UTF-8 que podríamos explotar posteriormente. También, podría devolvernos los siguientes códigos HTTP (Fernando Torres, 2019): (1) 400 - Los

parámetros no tienen el formato esperado (o no se han incluido) o (2) 404 - El código nacional no existe.

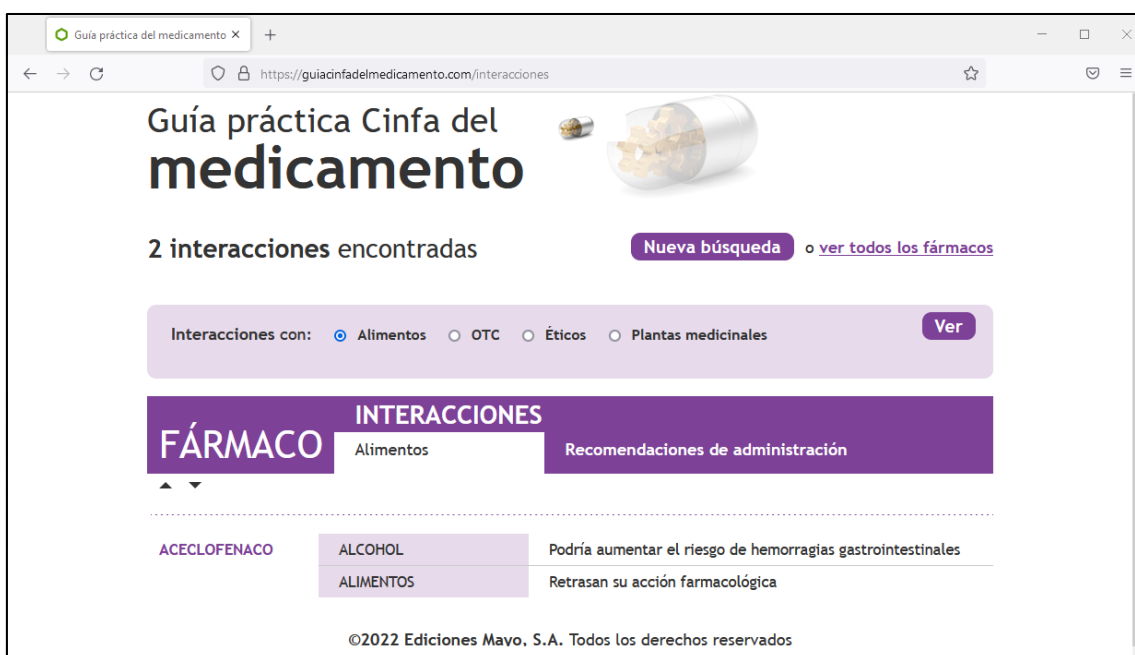
Ahora realizaremos un ejemplo de cómo son algunas de las llamadas y respuestas de esta API. En primer lugar, debemos de hacer la llamada, por ejemplo, GET `cima.aemps.es/cima/rest/psuministro/v2/cn/974931`, siendo 974931 en Código Nacional del medicamento (cuya utilidad la explicaremos posteriormente). Seguidamente, la API nos devolverá un JSON, como podemos apreciar en la Figura 11.

```
4 {
5   "cn": "974931",
6   "nombre": "NOVANTRONE 2 mg/ml CONCENTRADO PARA SOLUCION PARA PERFUSION , 1 vial de 10 ml",
7   "tipoProblemaSuministro": 10,
8   "fini": 1648677600000,
9   "ffin": 1659045600000,
10  "activo": true,
11  "observ": "La AEMPS ha autorizado unidades por comercialización excepcional. "
12 }
```

Figura 11: Respuesta de la API REST.

2.2.3.2.- Web Scraping para consultar interacciones

Como se comentó anteriormente, hemos identificado la carencia de información relacionada con alergias a determinando principios activos, así como información sobre interacciones entre medicamentos o alimentos. No hemos encontrado servicios como CIMA REST API que ofrezcan información relacionada. La información más útil en este sentido la hemos encontrado en la página web “Guía práctica del medicamento” (guiacinfadelmedicamento.com, 2022) de CINFA, ya que las que utiliza la aplicación Vademecum son de pago.



The screenshot shows a web browser displaying the 'Guía práctica del medicamento' website. The page title is 'Guía práctica Cinfa del medicamento'. Below the title, it indicates '2 interacciones encontradas' and provides a search button 'Nueva búsqueda' and a link 'o ver todos los fármacos'. There are radio buttons for 'Alimentos' (selected), 'OTC', 'Éticos', and 'Plantas medicinales', with a 'Ver' button. The main content area is titled 'INTERACCIONES' and has tabs for 'FÁRMACO', 'Alimentos', and 'Recomendaciones de administración'. Under the 'FÁRMACO' tab, 'ACECLOFENACO' is listed. Under the 'Alimentos' tab, there are two entries: 'ALCOHOL' with the note 'Podría aumentar el riesgo de hemorragias gastrointestinales' and 'ALIMENTOS' with the note 'Retrasan su acción farmacológica'. The footer contains the copyright notice '©2022 Ediciones Mavo, S.A. Todos los derechos reservados'.

Figura 12: Página web donde están las tablas de las interacciones.

En la Figura 12 se muestra como ejemplo la consulta del fármaco de las interacciones que produce el fármaco “ACECLOFENACO” con distintos alimentos. La información que nos ofrece el portal es que dicho fármaco interactúa con el alcohol y que el consumo de alimentos podría retrasar su acción farmacológica. A diferencia del servicio REST anteriormente descrito, en el portal la Guía Práctica del Medicamento los datos están directamente en tablas almacenadas como código HTML (ver ejemplo en Figura 13). El problema es que debemos realizar consultas independientes que requieren de la intervención del usuario para obtener las interacciones de cada fármaco. Por ello exploramos el uso de técnicas de *Web Scraping* que es una técnica que se utiliza para la extracción y almacenamiento de información de cualquier página web a través de un programa de software (Sergio Koller, 2021).

```

<table id="results">
  <tbody>
    <tr class>
      <th>ACECLOFENACO</th>
      <td class="bgcolor">
        <table>
          <tbody>
            <tr class>
              <td class="column1 onecolumn">ALCOHOL</td>
              <td>Podría aumentar el riesgo de hemorragias gastrointestinales</td>
            </tr>
            <tr class="noborder">
              <td class="column1 onecolumn">ALIMENTOS</td>
              <td>Retrasan su acción farmacológica</td>
            </tr>
          </tbody>
        </table>
      </td>
    </tr>
    <tr class>...</tr>
    <tr class>...</tr>
    <tr class>...</tr>
    <tr class>...</tr>
  </tbody>
</table>

```

Figura 13: Tablas en HTML de las interacciones de medicamentos con alimentos.

2.4.- Herramientas para el desarrollo de la RA

Como ya hemos dicho, nuestra aplicación tiene un gran componente de experiencias en RA, ya que es uno de sus puntos diferenciadores frente a otras aplicaciones. Creemos que una de las ventajas respecto a la competencia analizada es que al utilizar RA el usuario no necesita escribir el nombre del medicamento para obtener información de este, o podríamos aumentar otros aspectos relacionados con alergias a principio activos e interacciones.

En este apartado analizaremos qué tipo de herramientas tenemos disponibles para crear aplicaciones de RA, y concretamente, como podemos explotar los distintos tipos de RA para reconocer los envases de los medicamentos.

2.4.1.- Principales SDKs de RA: ARCore, ARKit y Vuforia

Hay tres principales SDKs de RA que son los más utilizados. A continuación, procederemos a explicar en detalle en qué consisten estas herramientas:

ARCore (Google Developers, 2022): es la herramienta de Google para poder crear experiencias de RA. Principalmente, ARCore hace dos cosas, (1) rastrear la posición del dispositivo móvil mientras se mueve y (2) construir su propia comprensión del mundo real. ARCore tiene tres características principales: (1) el seguimiento de movimiento, lo cual permite al dispositivo móvil comprender y rastrear su posición en relación con el mundo real, (2) la comprensión ambiental que permite al dispositivo móvil detectar el tamaño y la ubicación de todo tipo de superficies, y (3) la estimación de luz, lo cual permite al dispositivo móvil calcular las condiciones de iluminación actuales del entorno real.

- **ARKit** (Apple Developer, 2022): es la herramienta de Apple para poder crear experiencias de RA. La condición principal para toda experiencia RA, es la habilidad de mantener en todo momento una conexión entre el mundo real y el mundo virtual. Para mantener esta conexión Apple utiliza una técnica llamada Odometría Visual Inercial (VIO). ARKit es la última versión ha lanzado Apple, y proporciona una serie de características únicas como: (1) API de profundidad, (2) AR instantánea, (3) captura de movimiento, (4) geometría de la escena y (5) oclusión de personas. Las dos primeras características utilizan el scanner LIDAR, un sensor que solo lo utilizan los dispositivos móviles de Apple. Cada una de estas características contribuye al realismo de la RA de Apple y permite la colocación instantánea de objetos virtuales que se mezclan perfectamente con el entorno del mundo real.
- **Vuforia Engine** (ptc.com, 2021): es una herramienta software muy potente y usada para crear aplicaciones que proporcionen experiencia en RA, pertenece a la compañía PTC Inc. Una de sus grandes ventajas es que se puede utilizar en un gran número de dispositivos, no es exclusiva a una sola plataforma, por tanto, con ella se puede llegar a más público. Entre sus características de RA, tenemos: el reconocimiento dinámico dirigido a imágenes, objetos 3D y entornos para proporcionar flexibilidad en el desarrollo

Cada SDKs tiene sus propias ventajas y desventajas, y dependiendo de para qué mercado queremos desarrollar nuestra aplicación de experiencia de RA, tendremos que elegir uno por encima de los otros. Podemos decir que ARKit es el que ofrece una mejor experiencia RA, ya que tiene características exclusivas que las demás no tiene. Además, para los dispositivos a los que va dirigidos, tienen un sensor que no se encuentra en los dispositivos de la competencia, lo cual ofrece otro plus. El mayor problema de ARKit es que su uso está restringido al ecosistema de Apple, por ello, si se quiere desarrollar aplicaciones multiplataforma (desde Android, IOS hasta Windows), la única opción disponible

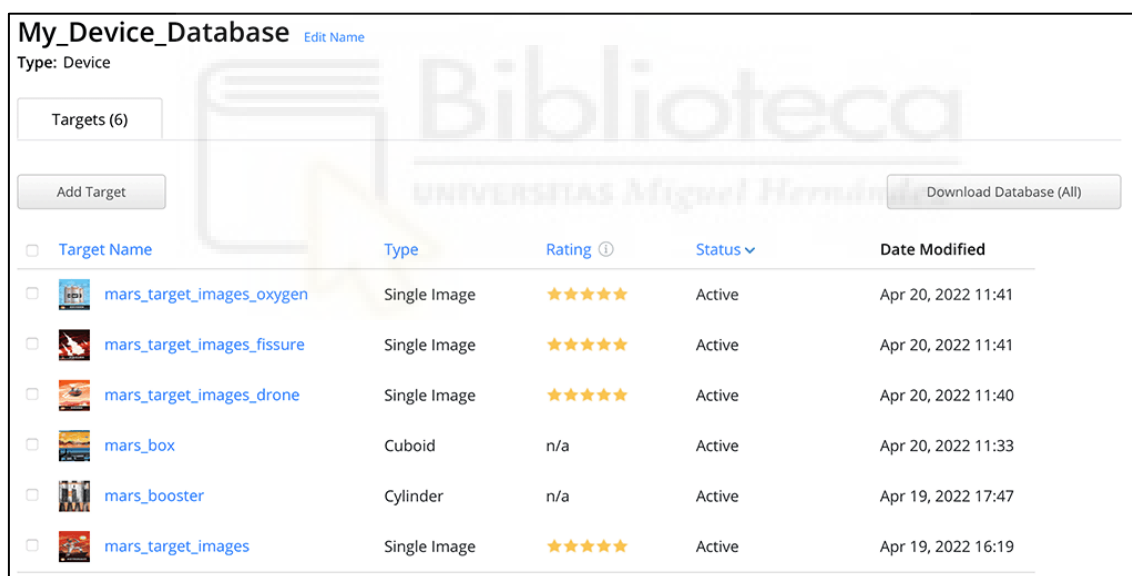
será el uso de Vuforia Engine aunque se tengan que sacrificar algunas características con esta elección.

2.4.2.- Limitaciones de los patrones de RA para analizar medicamentos

En esta sección se analizan las posibilidades ofrecidas por las principales API de RA para conseguir reconocimiento de los embalajes de los medicamentos. Como se comentó en la introducción, en este trabajo queremos además dar con una solución alternativa a la propuesta en (Martínez Parra, 2020), cuya principal limitación era la falta de escalabilidad de la solución y dificultad de adaptarse a cambios en los envases.

2.4.2.1.- Limitaciones de los patrones basados en imágenes

Una práctica habitual en el desarrollo de aplicaciones de RA usando las APIs actuales es el uso de marcadores basados en imágenes. Los distintos SDKs mencionados en el apartado anterior ofrecen funciones para poder detectar un objeto. Por ejemplo, ARCore con “*Augmented Images*” nos da la posibilidad de crear aplicaciones de RA, las cuales son capaces de reconocer imágenes que están previamente registradas, y anclar contenido virtual en ellas (Google, 2019).



The screenshot shows the Vuforia My_Device_Database interface. At the top, it says "My_Device_Database" with an "Edit Name" link. Below that, it indicates "Type: Device". There are buttons for "Targets (6)", "Add Target", and "Download Database (All)". A table lists the targets with columns for Target Name, Type, Rating, Status, and Date Modified.







<input type="checkbox"/>	Target Name	Type	Rating	Status	Date Modified
<input type="checkbox"/>	 mars_target_images_oxygen	Single Image	★★★★★	Active	Apr 20, 2022 11:41
<input type="checkbox"/>	 mars_target_images_fissure	Single Image	★★★★★	Active	Apr 20, 2022 11:41
<input type="checkbox"/>	 mars_target_images_drone	Single Image	★★★★★	Active	Apr 20, 2022 11:40
<input type="checkbox"/>	 mars_box	Cuboid	n/a	Active	Apr 20, 2022 11:33
<input type="checkbox"/>	 mars_booster	Cylinder	n/a	Active	Apr 19, 2022 17:47
<input type="checkbox"/>	 mars_target_images	Single Image	★★★★★	Active	Apr 19, 2022 16:19

Figura 14: Ejemplo de BBDD de Vuforia con seis patrones (extraída de (Vuforia, 2021)).

En nuestro caso, nos centraremos en mirar cómo funciona el reconocimiento de objetos en Vuforia por ser la única alternativa de las tres multiplataforma. Vuforia utiliza los llamados “*Image Targets*” para los que la API rastrea las características naturales extraídas de la imagen de la cámara con una base de datos de imágenes definidas como patrones previamente (Vuforia, 2022). En la Figura 14 se puede ver una BBDD de Vuforia compuesta por 6 patrones, y se observa como en la columna *Rating* se etiqueta cada patrón según sus propiedades para que su reconocimiento sea bueno. Este tipo de patrones fueron los que se utilizaron previamente en (Martínez Parra, 2020), y sus problemas de

escalabilidad se deben al hecho de que, para utilizar los “*Image Targets*” en el reconocimiento de embalajes de medicamentos, debemos ir añadiéndolos uno a uno. Además, se deberían definir tantos patrones como tipos de embalajes existan, y si se producen cambios en ellos la base de datos debería volver a reconstruirse.

Como alternativa, hemos explorado la posibilidad de otro tipo de patrones que puedan ser mapeados directamente con la información de los medicamentos contenida en recursos como los de CIMA.

2.5. El Código Nacional del medicamento como patrón

Analizando las distintas posibilidades que tenemos a la hora de identificar de forma única a un medicamento, llegamos a un identificador único: el Código Nacional (CN). El Código Nacional es un identificador único de cada medicamento, el cual aparece en la parte superior derecha de sus envases (ver Figura 15 a)). Hemos indagado en las posibilidades que nos da utilizar este identificador único y hemos descubierto que, si disponemos de este código podremos utilizarlo como ID para hacer las consultas a CIMA mediante su API REST, y esto nos permite sacar toda información posible del mismo y usarla para aumentar la realidad con nuestra aplicación.



Figura 15: a) Posición del Código Nacional en la caja de un medicamento. b) Resultado de la respuesta de la API REST con el CN de argumento.

Mediante esta propuesta conseguimos que nuestra solución sea escalable y adaptable a los cambios que puedan sufrir las cajas de los medicamentos, ya que nuestra aplicación de RA únicamente debería ser capaz de detectar y extraer dicho código y a partir de él se podría aumentar la realidad integrando en la aplicación la información extraída de la API.

2.6. Reconocimiento de patrones basados en texto

Dado que el Código Nacional es una cadena alfanumérica, decidimos en este trabajo explorar, como alternativa a los patrones basados en imágenes, opciones para la detección de texto en aplicaciones de RA.

2.6.1. Patrones de reconocimiento de texto en Vuforia

El reconocimiento de texto que ofrece Vuforia no es viable para nosotros, ya que debemos de dedicarle una cantidad enorme de tiempo, pues para su correcto funcionamiento debemos de ir añadiendo cada palabra a Vuforia para que la reconozca. Además, como podemos ver en la Figura 16, los patrones basados en texto de Vuforia son limitados, ya que, por un lado, no soporta palabras en español y, por otro lado, como hemos comentado las palabras a reconocer se tienen que ir añadiendo una a una.

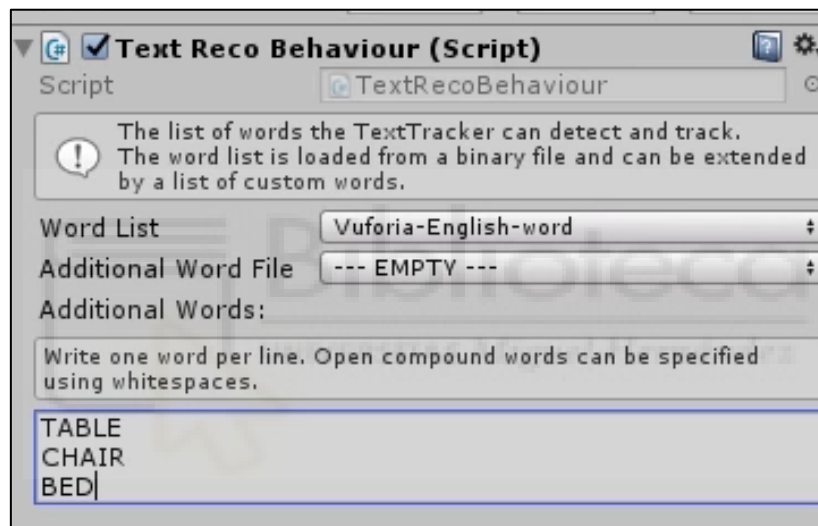


Figura 16: Patrones basados en texto de Vuforia.

2.6.2. OCR y Cognitive Services de Microsoft

Esta limitación de Vuforia ha motivado que hayamos redirigido nuestra búsqueda a analizar las capacidades de otras técnicas de reconocimiento de texto tradicionales para saber si son viables de integrar en nuestra aplicación de RA, reemplazando el uso de los patrones más populares usados en RA. Esta búsqueda nos ha llevado a la tecnología OCR (Optical Character Recognition) que se utiliza para convertir prácticamente cualquier tipo de imagen que contenga texto escrito (mecanografiado, manuscrito o impreso) en datos de texto legibles por máquina (Joshua Harris, 2018).

Buscando herramientas relacionadas con OCR, hemos llegado al servicio Computer Vision de Azure (Microsoft Azure, 2021) que nos da acceso a algoritmos avanzados que procesan imágenes y devuelven información basada en características visuales que son de nuestro interés.

Computer Vision tiene 3 características principales:

- **Extracción de texto (OCR):** extrae texto impreso y manuscrito de imágenes y documentos en varios idiomas y estilos de escritura.
- **Reconocimiento de imágenes:** permite extraer muchas características visuales de imágenes, por ejemplo, objetos, caras, contenido para adultos o descripciones de texto generadas automáticamente.
- **Análisis espacial:** analiza la presencia y el movimiento de personas en una fuente de vídeo, lo cual nos podría permitir, por ejemplo, controlar el aforo, la distancia social o el uso de mascarillas.

Nosotros nos centraremos en la característica de extracción de texto (OCR) ya que es la que nos permite automatizar la detección del Código Nacional. Este servicio utiliza las capacidades de los servidores de Azure para hacer OCR. Esta solución utiliza internamente modelos de aprendizaje automático cuyo uso es abstraído a los usuarios a través de la Computer Vision API v3.0. Para llegar a hacer OCR se puede utilizar la nueva Read API o la API OCR heredada. Read API es última tecnología OCR (Microsoft Azure, 2022). Básicamente el funcionamiento del OCR de Computer Vision, se basa en que le pasamos una imagen al servidor y este nos devuelve un JSON con el texto que haya encontrado en la imagen. En la Figura 17 se muestra un diagrama resumen de las distintas etapas del proceso.

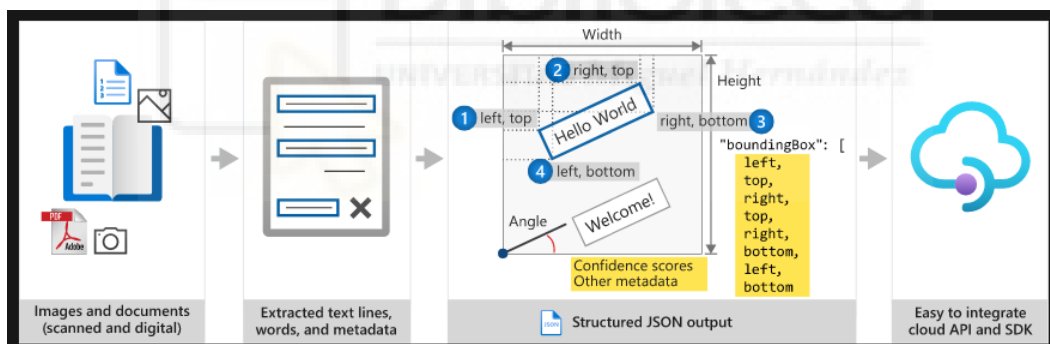


Figura 17: OCR de Computer Vision.

Computer Vision tiene dos planes uno Gratis y otro Estándar. Dentro del Estándar hay tres niveles. En la Figura 18 se muestran las características ofrecidas por cada nivel. El precio depende de cuantas transacciones hagamos, y una transacción es básicamente la utilización de una característica. El plan gratuito puede utilizar las características del nivel "Estándar" con limitaciones.

Desde un punto de vista técnico, como Computer Vision utiliza imágenes y documentos como entrada, estos deben tener en cuenta los siguientes requisitos para poder hacer OCR sobre ellos: (1) los archivos deben ser de los siguientes formatos: JPEG, PNG, BMP, PDF y TIFF, (2) en el caso de que los archivos sean PDF y TIFF, solo se procesan hasta 2000 páginas (la opción gratuita solo 2 páginas), y (3) los archivos deben tener un tamaño inferior a 500 MB (la opción gratuita solo hasta 4 MB), sus dimensiones deben ser entre 50x50 y 10.000 x 10.000 píxeles.

Instancia	Características	Precio
Gratis - Web/Container 20 transacciones por minuto		5.000 transacciones gratis por mes
S1 - Web/Container 10 transacciones por segundo	Etiqueta Face GetThumbnail Color Tipo de imagen GetAreaOfInterest	0-1M transacciones - €0,951 por 1000 transacciones 1-10M transacciones - €0,619 por 1000 transacciones 10-100M transacciones - €0,571 por 1000 transacciones 100M+ transacciones - €0,381 por 1000 transacciones
S2 - Web/Container	OCR Adultos Celebridad Lugar de interés Detectar, objetos Marca	0-1M transacciones - €0,951 por 1000 transacciones 1-10M transacciones - €0,619 por 1000 transacciones 10-100M transacciones - €0,571 por 1000 transacciones 100M+ transacciones - €0,381 por 1000 transacciones
S3 - Web/Container	Describir* Lectura	0-1M transacciones - €1,427 por 1000 transacciones 1M+ transacciones - €0,571 por 1000 transacciones
	Análisis espacial	Gratis durante la versión preliminar

Figura 18: Listado de precios de Computer Vision

El servicio nos permite extraer texto de imágenes que contienen texto impreso, texto manuscrito, dígitos y símbolos monetarios. En cuanto a los idiomas que soporta Computer Vision para hacer OCR, tenemos que, para el OCR de texto impreso tiene compatibilidad con inglés, francés, alemán, italiano, portugués, español, chino, japonés, coreano, ruso, árabe, hindi y otros idiomas internacionales que usan scripts latinos, cirílicos, árabes y devanagari. Además, el OCR de texto manuscrito tiene compatibilidad con inglés, chino simplificado, francés, alemán, italiano, japonés, coreano, portugués y español.

Como vemos en este resumen de las características del OCR de Computer Vision de Azure, es una tecnología muy interesante que puede tener muchas aplicaciones y que, en nuestro caso, como veremos más adelante en el apartado 3.2. nos servirá para poder extraer el texto que hay en los embalajes de los medicamentos. Nos hemos decantado por el OCR que ofrecen los Cognitive Services de Microsoft Azure, por la gran documentación que ofrecen y porque es relativamente fácil utilizar.

2.7.- Unity para el desarrollo de la app móvil de RA

Unity, antes llamado Unity 3D, es un motor de desarrollo de videojuegos (Iván Asensio, 2021). Con motor de videojuego o *Game Engine* nos referimos a un software que permite el diseño y la creación de un entorno interactivo.

Unity, nos permite desarrollar para diversas plataformas (PC, videoconsolas, móviles, etc.) gracias a un editor visual y la programación vía scripting (C#). Entre sus funcionalidades destacadas encontramos: (1) Motor gráfico el cual nos permite renderizar gráficos 2D y 3D, (2) Motor físico el cual nos permite simular las leyes de la física, (3) Animaciones, ya sea de objetos o personajes, (4) Sonidos, (5) Inteligencia Artificial, o (6) Programación vía scripting.

En Unity toda la creación del video juego gira en torno a su editor visual, el cual contiene elementos como:

- **Hierarchy view o Vista de la Jerarquía:** muestra todos los elementos de una escena, como la Cámara, interfaz de usuario, personajes, etc.
- **Scene view o Vista de la escena:** permite visualizar la escena, además de poder mover los elementos de la escena y mover la cámara.
- **Project View o Carpeta de Proyecto:** es nuestra carpeta del proyecto, en donde guardamos los scripts, las escenas, los modelos, los assets, etc.
- **Game View o Vista del Juego:** se muestra la vista final de la escena, al pulsar el botón de Play Unity compilará los scripts y nos mostrará, en esta pestaña, la escena final, donde podremos probar las funcionalidades de la aplicación.
- **Inspector View o Inspector:** esta pestaña nos mostrará toda la información del objeto que estemos seleccionando en ese momento. Desde los componentes hasta los scripts que estén asociados al objeto.
- **Console View o Vista de la Consola:** es la pestaña que nos mostrará los mensajes de debug o los errores de compilación.

La programación de Unity está basada en scripts en el lenguaje de programación de C#, aunque también se pueden utilizar otros lenguajes como Unity Script o Boo, pero principalmente se utiliza el de C# ya que es el que más usado por los usuarios. Para poder usar estos scripts en la aplicación se tienen que asociar a un GameObject, que es un objeto fundamental en Unity que puede representar personajes, props o el escenario. Al asociar el script al GameObject, este consigue una funcionalidad o funcionalidades definidas por el usuario.

En este proyecto, para la aplicación final se ha utilizado principalmente Unity y scripts en C#. Unity tiene una documentación muy completa y extensa que aplanan la curva de aprendizaje. También tiene una gran comunidad de usuarios, por lo que cualquier duda o error que uno se encuentre, ya está solucionado. Además, Unity tiene una interfaz muy intuitiva y tiene todo tipo de *plugins* para poder utilizar. Asimismo, contiene los SDK enfocados al desarrollo de la RA, como ARKit, ARCore o Vuforia. Por ello se ha decidido utilizar Unity, ya que pensamos que utilizar otro framework solo nos llevaría más tiempo en comparación a Unity.

2.8.- Otras herramientas y utilizadas

Además de las herramientas y servicios anteriormente mencionados, en este proyecto se ha utilizado otras herramientas como:

- **Notepad++** (notepad-plus-plus.org, 2022): en su versión 8.4.2, para la escritura de los scripts en C# de los que se compone la aplicación móvil de RA implementada en Unity.

- **MySQL de Mono Project** (mono-project.com, 2021): para la conexión de manera local de la aplicación móvil y la base de datos requerida para almacenar información asociada a los usuarios.
- **DB Browser** (sqlitebrowser.org, 2021): es una herramienta visual de alta calidad y de código abierto para crear, diseñar y editar archivos de bases de datos compatibles con SQLite.
- **Lucidchart** (lucidchart.com, 2022): se ha utilizado esta herramienta para la creación del diagrama de E/R de la BBDD y otros de los diagramas creados en la parte de análisis y diseño de software.
- **Python con Selenium** (selenium.dev, 2022) **y Octoparse** (octoparse.es, 2022): las dos primeras herramientas se han utilizado para realizar el Web Scraping de las interacciones, pero como las tablas no están en una dirección web fija, sino que las genera la web según las opciones que elija el usuario, se ha utilizado la herramienta Octoparse para la extracción de los datos que hay en las tablas.
- **SimpleJSON** (Bunny83, 2021): es un framework en C# puro sin dependencias de Unity u otras bibliotecas, el cual lo hemos utilizado para manipular los JSON. SimpleJSON, es un codificador y decodificador de JSON sencillo, rápido, completo, correcto y extensible. Este framework puede analizar cualquier JSON válido y no requiere ninguna clase serializable personalizada.

Para la parte de la interfaz de usuario, no se ha utilizado ninguna librería de terceros, ya que en Unity las mejores librerías de terceros son de pago en la *Asset Store*. Se ha optado por la creación de la interfaz con las herramientas nativas de Unity. Finalmente, hay que mencionar que se ha desplegado la aplicación RA en un móvil Samsung Galaxy S8. Todas estas herramientas se recogen visualmente en la Figura 19.



Figura 19: Diagrama resumen con los logotipos las tecnologías y herramientas propuestas para el desarrollo de la aplicación.

2.9.- Propuesta de solución

Para finalizar este capítulo y ya analizadas las diferentes opciones de desarrollo de RA, se ha creado un diagrama, como se ve la Figura 20, que muestra a modo de resumen la arquitectura hardware y software de la aplicación móvil de RA desarrollada. En los siguientes capítulos se describirán en más profundidad los aspectos relevantes de la implementación e integración de estas tecnologías.

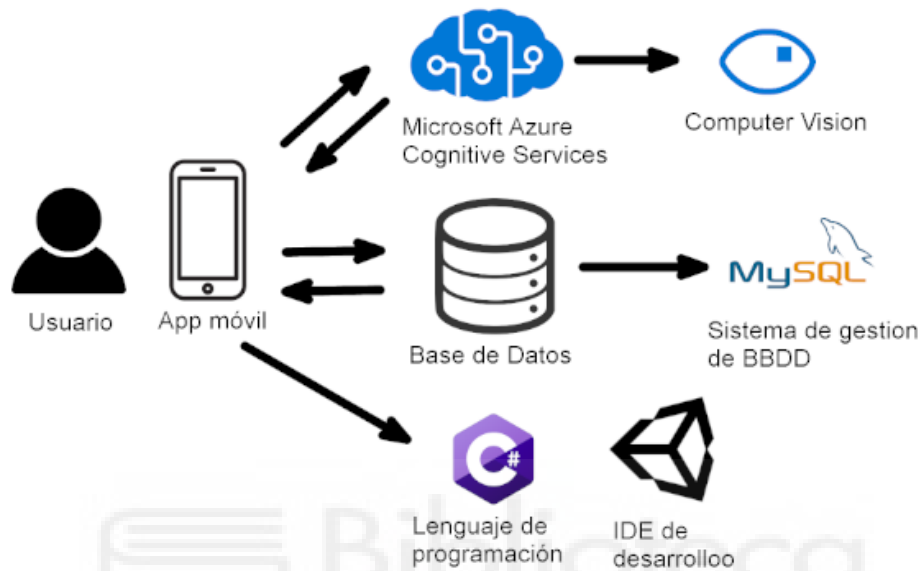


Figura 20: Arquitectura hardware y software de la solución desarrollada.

3.- Resultados y discusión

En este capítulo describiremos los resultados obtenidos incluyendo los aspectos más destacables de la fase de diseño y desarrollo de software. Para finalizar el capítulo se describirá la aplicación desarrollada describiendo su funcionalidad a través de una selección de capturas de pantalla.

3.1.- Análisis y diseño del software

En esta sección contaremos algunos aspectos relevantes relacionados la metodología de desarrollo utilizada para la creación de nuestra aplicación, y posteriormente describiremos algunos aspectos relacionados con el análisis y diseño del software.

3.1.1.- Metodología de desarrollo

Se ha utilizado la metodología de desarrollo ágil Scrum (proyectosagiles.org, 2022). Scrum es una estructura de desarrollo incremental, es decir, el ciclo de desarrollo del proyecto se descompone en una serie de pequeños proyectos divididos en 3 etapas (análisis, desarrollo y testing). Además, las iteraciones o *sprints*, son unos de sus elementos principales. Como se ve en la Figura 21, los *sprints* son entregas parciales y regulares del producto final en las que se fija el incremento de funcionalidad del producto. Además, los *sprints* están priorizados por el beneficio que aportan al receptor del proyecto.



Figura 21: Funcionamiento del Sprint de Scrum.

Se ha elegido Scrum como metodología de trabajo por estar especialmente indicada para proyectos en entornos complejos donde los requisitos cambian constantemente ya que estos están poco definidos, y donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales. Por ejemplo, nuestro proyecto es abierto ya que se puede llegar a la solución del problema de diversas formas, ya sea mediante detección de objetos con Vuforia o la utilización de OCR de Cognitive Services de Azure.

En el caso de este proyecto, no ser un equipo de desarrollo grande ha permitido simplificar la planificación de este. Para cada *sprint* se concretan en una primera reunión los requisitos a alcanzar. Una vez finalizado el *sprint* el equipo vuelve a reunirse analizando los siguientes aspectos:

- **Revisión:** se presentan los requisitos completados en la iteración en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados, ya desde la primera iteración, se va replanificando el proyecto.
- **Retrospectiva:** se analiza cómo ha sido nuestra manera de trabajar y cuáles son los problemas que podrían impedirnos progresar adecuadamente, para así mejorar la productividad de manera continua.

En este proyecto, hemos realizado un total de 10 iteraciones o fases cuyo cronograma se puede consultar en la Figura 22.

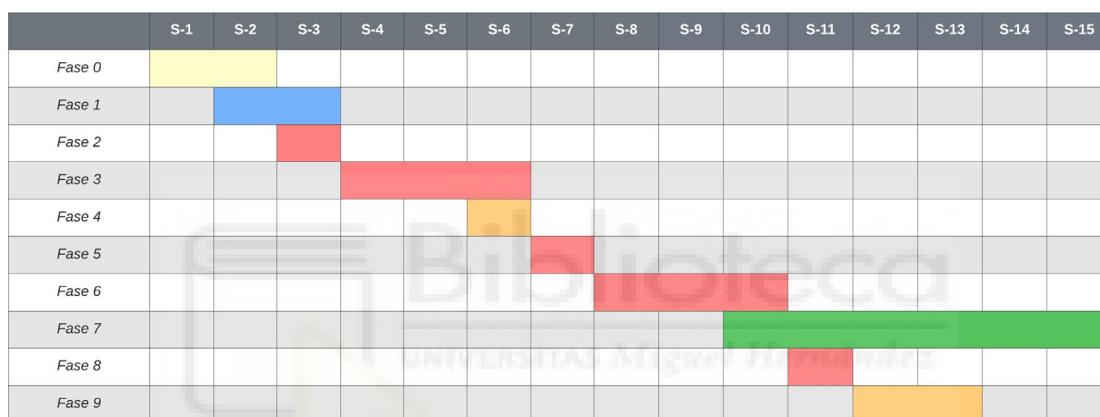


Figura 22: Diagrama de Grantt

A continuación, definimos con más detalle cada una de las 10 iteraciones realizadas, incluyendo su descripción y aspectos relacionados con el grado de consecución de los requisitos:

- **Implementación fase 0:**
 Experimentar con RA en Unity, se debe montar una pequeña aplicación de RA y buscar un banco de imágenes de las cajas de los medicamentos.
 - Duración aproximada: 2 semanas.
 - ¿Se han completado los requisitos?
 - Parcialmente. No se encuentra el banco de imágenes.
- **Implementación fase 1:**
 Reconocimiento de objetos mediante Vuforia, detección de patrones alternativos a *Image Target* como patrones de texto. Controlar la captura de pantalla de la aplicación.
 - Duración aproximada: 2 semanas.
 - ¿Se han completado los requisitos?

- Parcialmente. El uso de patrones de texto de Vuforia no funciona como esperamos. Se descarta su uso y se buscan alternativas compatibles con OCR.
- **Implementación fase 2:**

Aplicación realiza capturas de pantalla automáticamente cada x segundos y guarda la captura en un fichero privado de la aplicación.

 - Duración aproximada: 1 semana.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 3:**

Realizar OCR con la aplicación mediante Cognitive Services de Azure.

 - Duración aproximada: 2 semanas.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 4:**

Ajustes de la implementación del OCR y testing. Se testea el correcto funcionamiento del OCR y sobre todo si el OCR recoge correctamente el Código Nacional de un medicamento.

 - Duración aproximada: 1 semana.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 5:**

Conectar base de datos a Unity.

 - Duración aproximada: 1 semana.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 6:**

Ingeniería inversa, con la ayuda del OCR, hacer consultas a la web de CIMA desde Unity Rest. También, se debe hacer Web Scraping para sacar los datos de las interacciones que hay en las tablas de la web guiacinfadelmedicamento.com.

 - Duración aproximada: 3 semanas.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 7:**

Generación de la memoria final y documentación.

 - Duración aproximada: 6 semanas.
 - ¿Se han completado los requisitos?: Sí.
- **Implementación fase 8:**

Implementar la posibilidad de iniciar sesión y de la creación de usuario en la aplicación. También, el usuario debe tener la posibilidad de indicar las alergias que tenga a ciertos principios activos que hay en los medicamentos.

 - Duración aproximada: 1 semana.
 - ¿Se han completado los requisitos?: Sí.

- **Implementación fase 9:**

Testing de la aplicación final, ver si todas las funcionalidades de la aplicación funcionan correctamente.

- Duración aproximada: 2 semana
- ¿Se han completado los requisitos?: Sí.

3.1.2.- Captura de requisitos y casos de uso

En este apartado se definirán los casos de uso que resumen las características de la aplicación que se han ido implementando a lo largo del proyecto de manera incremental. En la Figura 23 se resumen los distintos actores y casos de usos definidos. Como se puede observar tenemos 3 actores: (1) Usuario, (2) Microsoft Azure y (3) CIMA. Estos 2 últimos serían actores secundarios, los cuales participan en el caso de uso y el Usuario es un actor principal, es decir, inicia el caso de uso y el cual tendrá acceso a las funcionalidades de la aplicación si está registrado. El usuario, podrá iniciar sesión cuando se haya registrado. También, a la hora de registrarse, para que este se complete, el usuario tendrá que indicar si tiene o no alergia a algunos de los principios activos que se indican.

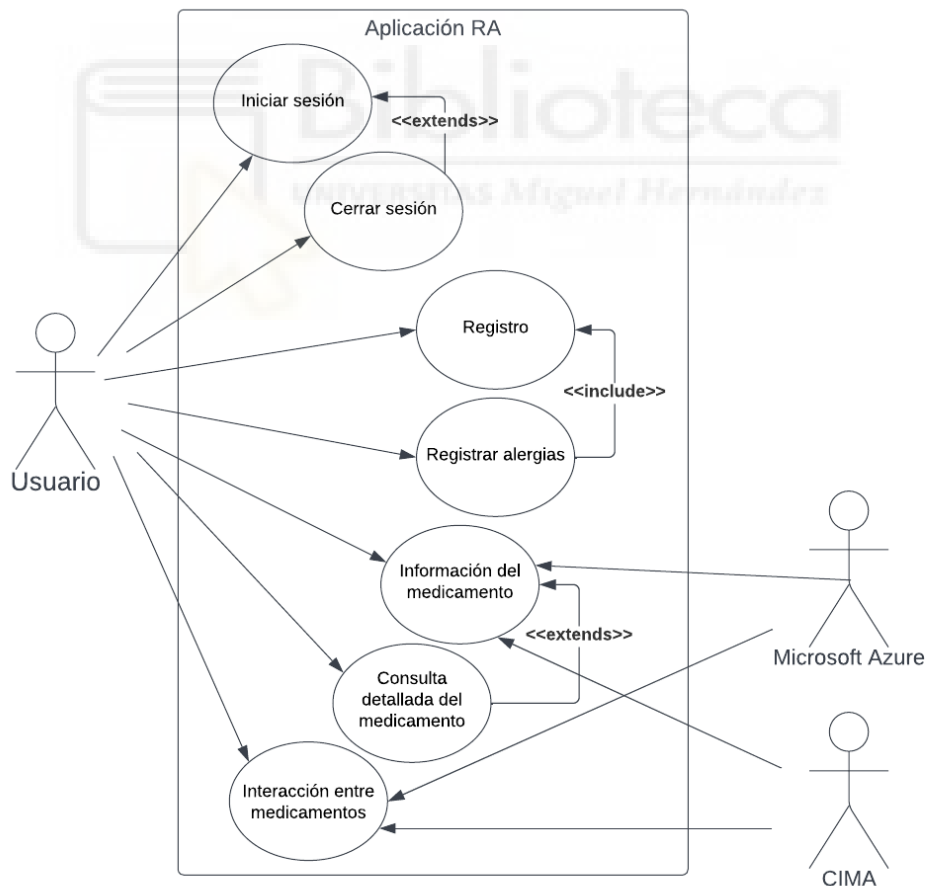


Figura 23: Diagrama de casos de uso.

La aplicación tiene la funcionalidad dividida en dos partes bien diferenciadas: (1) ver información del medicamento, y (2) consultar interacciones. La funcionalidad

del punto (1) permite al usuario puede ver información detallada del medicamento, donde se le indicará si ese medicamento se sigue comercializando, los principios activos que tenga el medicamento, la alergia que tenga a alguno de esos principios activos (si el principio activo del medicamento coincide con el que el usuario registro en las alergias) o las interacciones con alimentos que tenga ese medicamento. La funcionalidad del punto (2) permite al usuario consultar las interacciones de los medicamentos. Con esta opción, el usuario puede ver las interacciones al apuntar con la cámara del móvil a dos medicamentos.

En la Tabla 2 se muestra, a modo de ejemplo, el Caso de Uso 1 desarrollado de la aplicación. Para realizar esta descripción hemos utilizado las plantillas que hemos utilizado en la asignatura de Ingeniería del Software (Rodríguez Sala et al., 2019). El resto de los casos de uso se podrán encontrar en el Anexo I.

CU-01	Registro	
Actores	Todos los usuarios no registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario no registrado quiera registrarse en la aplicación de RA	
Precondición	El usuario no debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario entra en la aplicación, donde lo primero que le saldrá es la pantalla de Iniciar Sesión
	2	El usuario pincha en el botón de "Crear cuenta nueva" lo que lo lleva a la pantalla de Registro
	3	El usuario rellena los campos de nombre, apellidos, teléfono, dirección de correo y contraseña
	4	El sistema comprueba que los datos introducidos no están guardados en la base de datos
	5	Si el correo electrónico ya existe, el sistema imprime un mensaje de error y permite al usuario repetir el intento (pasos 3-4)
	6	Si el correo electrónico no existe, el sistema redirige al usuario a la pantalla de selección de alergias
Postcondición	Con la cuenta validada el usuario puede introducir si quiere las alergias que tenga a ciertos principios activos	
Excepciones	Paso	Acción
	6	El usuario cancela la operación. El caso de uso termina

Tabla 2: Casos de Uso 1: Registro.

3.1.3.- Diagrama de secuencia y diagrama de estado

En la Figura 24 se muestra el diagrama de secuencia. Estos de diagramas son un tipo de diagrama de interacción que describe el comportamiento dinámico del sistema, haciendo especial énfasis en las secuencias de mensajes que intercambian los diferentes objetos.

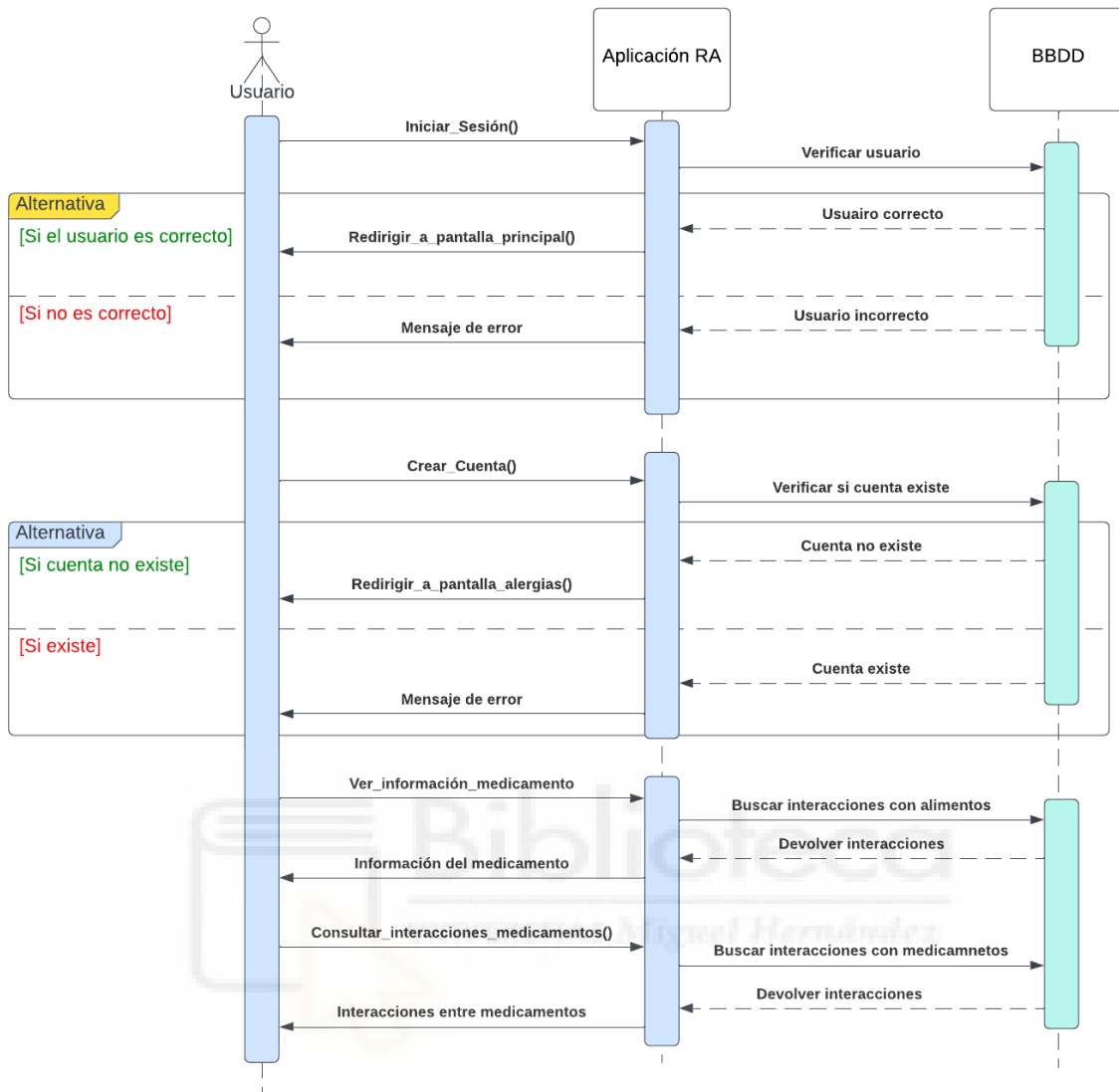


Figura 24: Diagrama de secuencia

En la Figura 25, se muestra un ejemplo de diagrama de secuencia que nos ayuda a visualizar de una mejor forma las transiciones entre diversos objetos.

3.1.4.- Diseño de la base de datos

El diseño de la base de datos es una parte importante en el desarrollo del software, ya que, si hay un mal diseño de la base de datos afectará negativamente el desempeño del software que estemos creando.

En nuestro caso, nuestro diseño de la base de datos no es complejo ya que hemos externalizado una parte crítica como es la información relacionada con los medicamentos. Esto nos permite no tener que almacenar dicha información, en su lugar, la consultaremos dinámicamente a medida que se identifique el CN de medicamentos explotando el fichero de datos JSON que nos devuelven los servicios. Pese a la externalización de esta información clave, hemos tenido que definir una BDD para registrar datos internos.

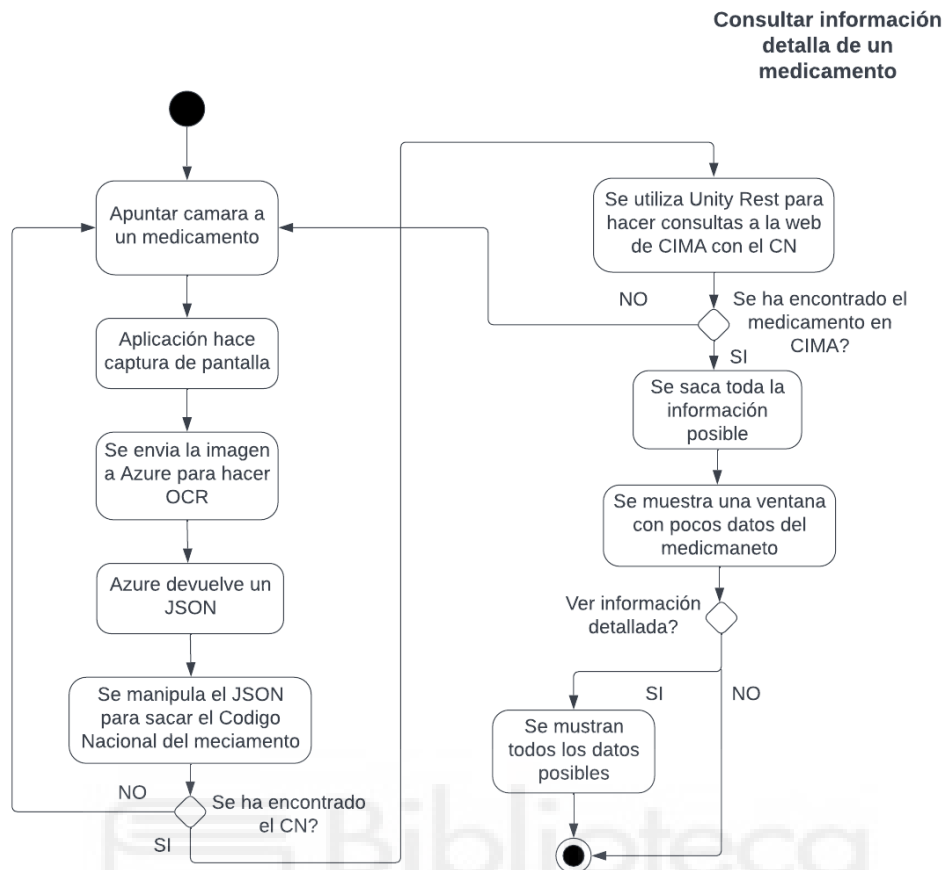


Figura 25: Diagrama de estado: Consultar información detallada de un medicamento.

En la Figura 26 se muestra el diagrama E/R donde tenemos los siguientes elementos que describimos a continuación:

- **Tabla “Usuarios”:** en que guardaremos los datos del usuario necesarios para realizar el login. Esta tabla está formada por una clave ID, Email, Contraseña, Nombre, Apellidos y Teléfono del usuario.
- **Tabla “Principio”:** con un ID de clave y el Nombre del principio activo, en esta tabla se guardan todos los principios que hemos recolectado a través del Web Scraping de las tablas de la web “Guía CINFA del Medicamento”.
- **Tabla “Lista_Alergias”:** como se ha comentado antes, el usuario puede elegir a que principios activos tiene alergia, por ello, se ha creado esta tabla en la que guardaremos los principios a los que el usuario tenga alergia. Tiene como clave el ID y tiene como clave foránea tanto el ID del usuario como el ID del principio activo al que el usuario tenga alergia.
- **Tabla “Interacciones_Alimentos”:** nos permiten almacenar las interacciones que hay entre un principio activo y distintos alimentos.
- **Tabla “Interacciones_Medicamentos”:** nos permiten almacenar las interacciones entre principio activo y principio activo.

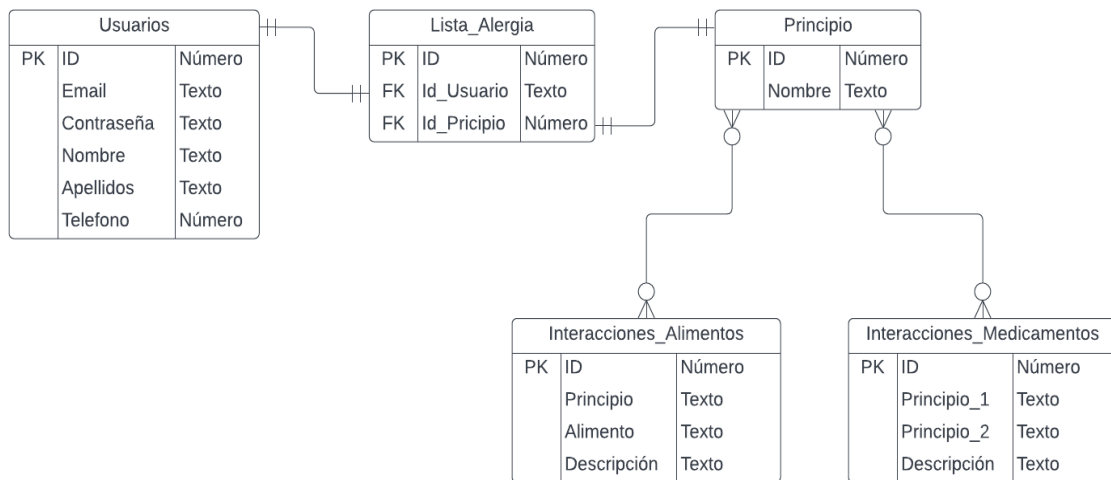


Figura 26: Diagrama de Entidad-Relación.

3.1.5.- Prototipado

Para el prototipo de la aplicación, se decidió hacer unos esbozos para tener una visión más clara de cómo sería la interfaz de usuario de la aplicación. El prototipado nos ayuda a ordenar nuestras ideas, explorar diferentes conceptos o diseños. También, nos ayuda a detectar posibles problemas o carencias que pueda tener el diseño antes de comenzar la fase de programación del proyecto. Con ello, podemos probar, ajustar y perfeccionar los elementos de la interfaz de usuario con los que van a interactuar los usuarios.

En la Figura 27, se pueden ver varios ejemplos del prototipado de las diferentes pantallas de la aplicación. Este prototipado se podría haber hecho mediante programas dedicados a realizar prototipos de la interfaz de usuario, pero muchos de esos programas requieren pagar ya sea el producto o la clave para su utilización, así que se decidió por algo sencillo y barato.

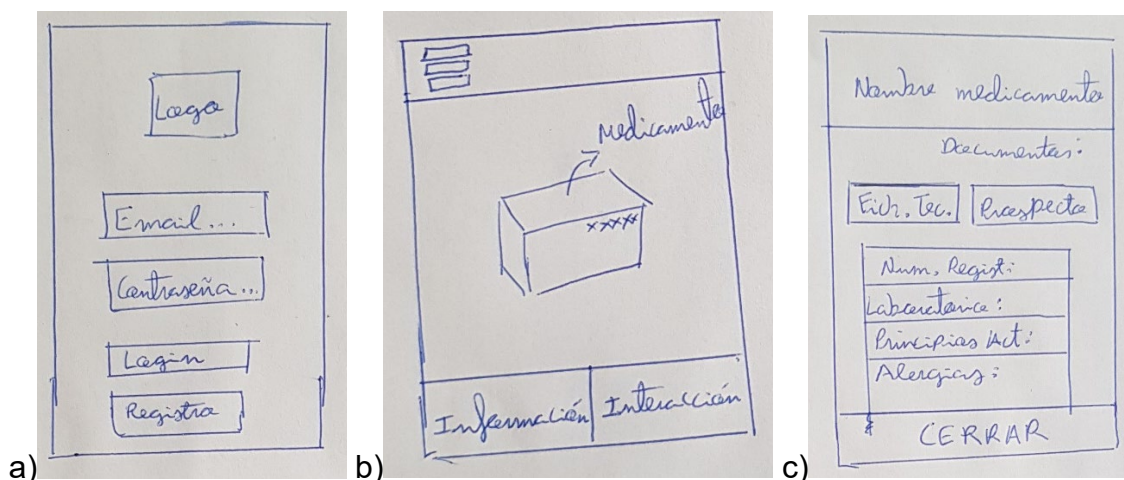


Figura 27: a) Prototipo de la pantalla de inicio. b) Prototipo de la pantalla principal. c) Prototipo de la información detallada de un medicamento.

En la Figura 28, tenemos el mapa de navegación de la aplicación, donde vemos las conexiones entre las diferentes pantallas.

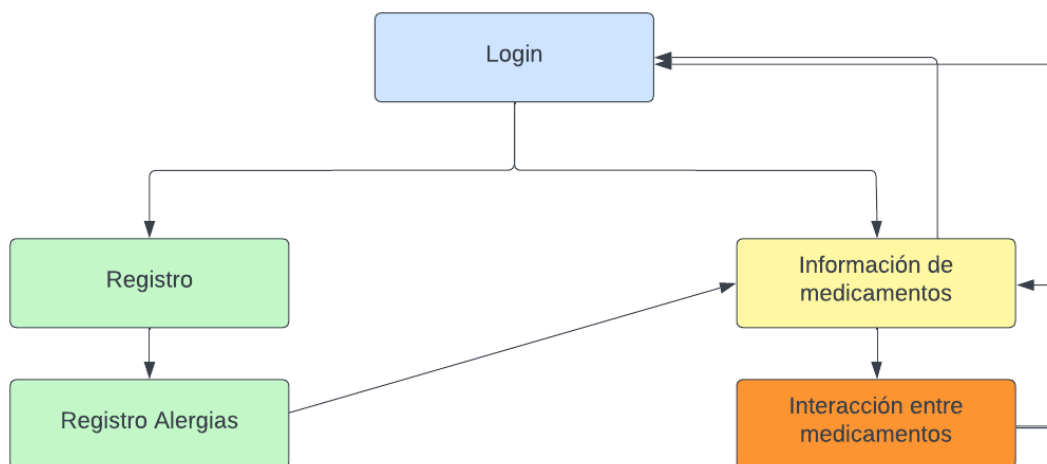


Figura 28: Mapa de navegación de la aplicación

3.2.- Implementación en Unity de patrones OCR de RA

En esta sección describiremos los resultados de las pruebas donde estábamos buscando el mejor método para poder montar nuestra aplicación de RA: (1) la detección de objetos mediante Vuforia o (2) la utilización de OCR de Cognitive Services de Azure.

En el Anexo II se describen el proceso realizado para crear una base de datos de prueba basada en los *Image Target*. Hemos estimado un tiempo de 10 minutos para dar de alta o modificar la información asociada a cada envase, lo que haría difícil escalar la aplicación para soportar miles de medicamentos. Aunque se consiga semi-automatizar, definir patrones basados en imágenes también haría difícil tener que estar al día de los cambios de los distintos envases, lo que vuelve a suponer un problema para su gestión actualizada.

Dado que los patrones de texto de Vuforia planteaban problemas similares, nos enfrentamos a la tarea de implementar nuestros propios patrones de texto para RA, y a continuación describimos las partes más relevantes del proceso para que pueda ser en un futuro reutilizado por otros.

3.2.1.- Creación del proyecto Computer Vision

Para trabajar con el OCR de Microsoft Azure, debemos de crear un proyecto de Computer Vision, que es la herramienta de los Cognitive Services de Azure. Con la creación del proyecto se nos dará las claves para poder hacer llamadas al servidor de Azure. Para ello iremos a la web de Azure, elegimos la herramienta Computer Vision y le damos a crear.

Como se ve en la Figura 29, en la creación del proyecto nos pedirá la suscripción, en nuestro caso utilizaremos la que nos proporciona la Universidad Miguel Hernández de Elche. Después, debemos escoger el recurso que en nuestro caso se ha creado desde cero. También, debemos elegir la Región y el Nombre del proyecto. Finalmente tenemos que elegir la tarifa, para nuestro proyecto elegiremos la gratuita, la tarifa F0.

Detalles del proyecto

Suscripción * ⓘ Azure for Students

Grupo de recursos * ⓘ UnityGrupoRecursos
Crear nuevo

Detalles de la instancia

Región ⓘ West Europe

Nombre * ⓘ UnityComputerVisionF01

Plan de tarifa * ⓘ Free F0 (20 Calls per minute, 5K Calls per month)

Figura 29: Proyecto de Computer Vision.

Una vez ya creado el proyecto, ya tendremos nuestras claves para poder hacer las llamadas al servidor (ver Figura 30).

Mostrar claves

Clave 1
.....

Clave 2
.....

Ubicación o región ⓘ westeurope

Extremo
<https://unitycomputervisionf0.cognitiveservices.azure.com/>

Figura 30: Claves de Computer Vision.

3.2.2.- Implementación del OCR de Azure en Unity

Después de obtener las claves, debemos de ir a Unity y crear un script en C#, el cual en nuestro caso llamaremos `CamaraController`. Para tomar una foto, vamos a crear una nueva función llamada `TakePicture` (ver Figura 31). Esta función va a ser diferente a una normal y eso es porque estamos haciendo una co-rutina que usa `IEnumerator` en lugar de `void`. Una co-rutina, es una función que podemos pausar a mitad de camino y esperar a que ocurran cosas específicas antes de continuar. Necesitamos esperar al final del cuadro para tomar una foto.

Como podemos ver en la Figura 31:

- 1) Lo primero será esperar al final del cuadro para que todos los píxeles sean renderizados.
- 2) Creamos un nuevo `Texture2D` que es básicamente una copia de la textura de la cámara.
- 3) Leemos los píxeles de la textura de la cámara y los aplicamos
- 4) Convertimos la textura en una matriz de bytes, ya que será la forma en la que vamos a enviar la imagen a la API.
- 5) Enviamos los datos de bytes a la API de Computer Vision, la cual llamaremos en una función de otro script que crearemos más adelante.

```
IEnumerator TakePicture ()
{
    //1)
    yield return new WaitForEndOfFrame();

    //2)
    Texture2D screenTex = new Texture2D(camTex.width, camTex.height);

    //3)
    screenTex.SetPixels(camTex.GetPixels());
    screenTex.Apply();

    //4)
    byte[] byteData = screenTex.EncodeToPNG();

    //5)
    AppManager.instance.StartCoroutine("GetImageData", byteData);
}
```

Figura 31: Código de la función `TakePicture`.

Ahora, crearemos otro script en C# que llamaremos `AppManager` con el cual nos conectaremos a la API de `ComputerVision` y obtendremos el texto extraído de los datos de la imagen que hemos creado en el script anterior. Como vamos a hacer llamadas al servidor de Azure (ver Figura 32) necesitaremos añadir el namespace `UnityEngine.Networking` y también añadiremos el namespace `SimpleJSON` cuyos archivos debemos de añadir al proyecto de Unity.

También, debemos crear una variable para poder enlazar desde Unity la clave que hemos creado anteriormente, en nuestro caso será una subclave o `subKey` (para la clave de suscripción) y otra variable para la dirección de la API. A continuación, se incluye el texto de visualización que mostrará el texto que extraemos de la imagen. Finalmente, debemos instanciar la clase `AppManager` para poder llamar una de sus funciones en el script que creamos anteriormente.

```

using UnityEngine.Networking;
using SimpleJSON;

public class AppManager : MonoBehaviour
{
    // Computer Vision subscription key
    public string subKey;

    // Computer Vision API url
    public string url;

    // texto en pantalla que muestra el texto del OCR
    public TextMeshProUGUI uiText;

    // instance
    public static AppManager instance;

    void Awake ()
    {
        // set the instance
        instance = this;
    }
}

```

Figura 32: Código del script AppManager

Antes de seguir tenemos que vincular los datos de estas 3 variables desde Unity. Como se aprecia en la Figura 33, añadimos la subclave que hemos creado anteriormente, también debemos de añadir la URL en nuestro caso será el extremo que aparece en la Figura 30 pero debemos de añadirle “vision/v3.2/ocr” para hacer OCR y finalmente un objeto de Unity “Text Mesh Pro” el cual es un elemento de UI.

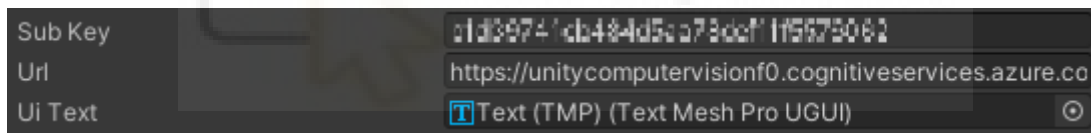


Figura 33: Acceso a las variables desde Unity

La función principal de este script la llamaremos `GetImageData` (ver Figura 34). Esta función, recibe una imagen como una matriz de bytes, la envía a la API, la API nos devuelve un archivo JSON, ese JSON lo convierte en un objeto `SimpleJSON` y extrae el texto de él. Esta función también es una co-rutina. Esto se debe a que haremos una pausa en la función mientras esperamos una respuesta del servidor.

Como podemos ver en la Figura 34, 1) creamos un nuevo `UnityWebRequest`, esta es una clase hecha por Unity y nos permite hacer peticiones a servidores. En nuestro caso, queremos enviar los datos de la imagen a la API. También establecemos el método a `POST`, lo cual significa que estamos enviando datos al servidor para ser calculados. 2) creamos un “download handler”, el cual nos permite acceder a los datos que se devuelven de la solicitud de la API, en nuestro caso un JSON. 3) además de recibir datos, también queremos enviarlos (en este caso la matriz de bytes de la imagen) y eso se lo haremos con un “upload handler”.

Seguidamente, en el paso 4) estableceremos una cabecera que contendrá nuestra subclave para identificar quiénes somos. 5) enviamos la petición web y esperamos el resultado. 6) obtenemos los datos de la respuesta del servidor y los convertimos en un objeto JSON utilizando `SimpleJSON`. Finalmente, con 7) obtenemos el texto del JSON, manipulándolo en la función `GetTextFromJSON` que veremos más adelante e imprimimos el texto en pantalla para que se visualice mejor el resultado del OCR.

```
public IEnumerator GetImageData (byte[] imageData)
{
    // 1)
    UnityWebRequest webReq = new UnityWebRequest (url);
    webReq.method = UnityWebRequest.kHttpVerbPOST;

    // 2)
    webReq.downloadHandler = new DownloadHandlerBuffer ();

    // 3)
    webReq.uploadHandler = new UploadHandlerRaw (imageData);
    webReq.uploadHandler.contentType = "application/octet-stream";

    // 4)
    webReq.SetRequestHeader ("Ocp-Apim-Subscription-Key", subKey);

    // 5)
    yield return webReq.SendWebRequest ();

    // 6)
    JSONNode jsonData = JSON.Parse (webReq.downloadHandler.text);

    // 7)
    string imageText = GetTextFromJSON (jsonData);
    uiText.text = imageText;
}
```

Figura 34: Código de la función `GetImageData`

Como se ha comentado antes, para manipular el JSON que nos devuelve la API de Azure, debemos crear una función llamada `GetTextFromJSON`, la cual tomará el archivo JSON y obtener sólo el texto de él. Como vemos en la Figura 35, esa la estructura del archivo JSON que se nos devuelve. Lo que queremos acceder es a todas las propiedades de texto.

Ya creada la función `GetTextFromJSON` y como se aprecia del código de la Figura 36, 1) accedemos a las líneas que hay dentro de las regiones, 2) hacemos un bucle a través de cada línea, 3) otro bucle dentro para cada palabra y finalmente 4) accedemos al texto que hay en las palabras y lo añadimos.

```
"language": "en",
"textAngle": 0.0,
"orientation": "Up",
"regions": [
  {
    "boundingBox": "370,510,378,40",
    "lines": [
      {
        "boundingBox": "370,510,378,40",
        "words": [
          {
            "boundingBox": "370,510,103,40",
            "text": "test"
          },
          {
            "boundingBox": "492,510,256,40",
            "text": "sentence"
          }
        ]
      }
    ]
  }
]
```

Figura 35: Estructura del archivo JSON.

```
string GetTextFromJSON (JSONNode jsonData)
{
    string text = "";
    JSONNode lines = jsonData["regions"][0]["lines"];

    // bucle a través de cada línea
    foreach(JSONNode line in lines.Children)
    {
        // bucle a través de cada palabra en la línea
        foreach(JSONNode word in line["words"].Children)
        {
            // Añadimos texto
            text += word["text"] + " ";
        }
    }
    return text;
}
```

Figura 36: Código de la función GetTextFromJSON.

Finalmente, en la Figura 37 podemos ver el resultado final, en donde apuntamos a un medicamento y la aplicación toma una captura, se hace OCR a la captura y se muestra el texto del OCR en pantalla.

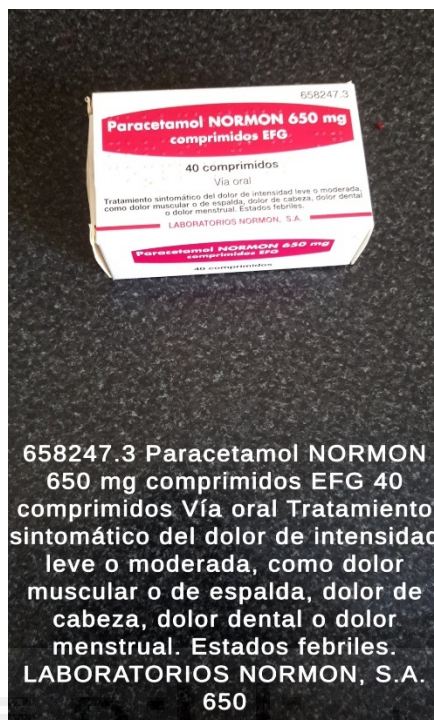


Figura 37: Ejemplo del uso de OCR.

Una vez tenemos el texto podremos explotarlo para hacer consultas a la API REST de CIMA. El proceso descrito en este apartado deberemos repetirlo cada 5 segundos para conseguir el comportamiento real de un patrón de RA

3.3.- Implementación de la aplicación móvil de RA

En este punto se describen otros aspectos relevantes de la implementación de la aplicación móvil de RA y su interfaz gráfica.

3.3.1.- Recuperando información desde la web de CIMA

A partir del texto obtenido con el patrón OCR de RA descrito en la sección anterior debemos hacer dos consultas a la web de CIMA para obtener los datos del medicamento que se encuentra frente a la cámara de nuestro dispositivo móvil. La primera consulta será con el Código Nacional para poder conseguir el número de registro que usa internamente la web de CIMA para cada medicamento. Es a través de esta segunda consulta con el número de registro con la que nos devuelve un JSON con más información del medicamento.

Para la consulta con el Código Nacional seguiremos justamente donde lo dejamos con la parte del OCR. Como podemos ver en la Figura 38, el texto que obtuvimos con el OCR lo partimos, para luego sacar el Código Nacional, el cual

es un número entero de 6 dígitos. Utilizamos la URL de la API REST utilizando el código nacional para la consulta.

Aplicamos los mismos conceptos que hicimos en la parte de OCR, donde creamos un nuevo `UnityWebRequest` y un "download handler". Luego, enviamos la petición a la API REST y esperamos el resultado, seguidamente obtenemos los datos de la respuesta de la API y los convertimos en un objeto JSON utilizando `SimpleJSON`. Finalmente, obtenemos el número de registro con la función `GetTextFromJSON2`.

```
string imageText = GetTextFromJSON(jsonData);
//Final parte OCR
//Inicio parte consulta con la web de CIMA
string[] cadenaToken = imageText.Split(new[] { '.', ' ' });
for (int i = 0; i < cadenaToken.Length; i++)
{
    if (int.TryParse(cadenaToken[i], out int num) && cadenaToken[i].Length == 6) {
        codNacional = cadenaToken[i];
        //LLAMAMOS A LA API REST de CIMA
        string URL = "https://cima.aemps.es/cima/rest/medicamentos?&multiple="+codNacional;

        UnityWebRequest request = UnityWebRequest.Get(URL);

        request.downloadHandler = new DownloadHandlerBuffer();

        yield return request.SendWebRequest();

        JSONNode jsonData2 = JSON.Parse(request.downloadHandler.text);

        string numRegistro = GetTextFromJSON2(jsonData2);
    }
}
```

Figura 38: Código de la consulta a la web de CIMA con el CN.

Por último, para esta parte de las consultas a la web de CIMA, vemos en la Figura 39 la última consulta de las 2 que tenemos que hacer. En este caso el proceso será exactamente el mismo que en la anterior consulta, pero en vez del código nacional utilizamos el número de registro y la función `GetTextFromJSON3` no devuelve nada, pero ya dentro de ella guardamos toda la información que necesitamos del medicamento.

```
string URL2 = "https://cima.aemps.es/cima/rest/medicamento?nregistro="+
numRegistro+"&carganotas=1&cargamateriales=1";

UnityWebRequest request2 = UnityWebRequest.Get(URL2);

request2.downloadHandler = new DownloadHandlerBuffer();

yield return request2.SendWebRequest();

JSONNode jsonData3 = JSON.Parse(request2.downloadHandler.text);

GetTextFromJSON3(jsonData3);
```

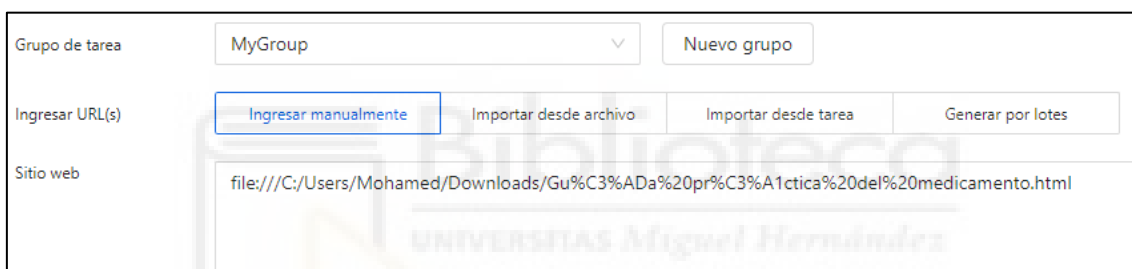
Figura 39: Código de la consulta a la web de CIMA con el número de registro.

3.3.2.- Población de la BBDD y acceso desde Unity

Además de los datos de cada medicamento extraído de CIMA, se han detectado requisitos descritos en los casos de uso que requieren de la implementación de una BBDD. Hemos decidido acceder a esta BBDD de manera local para que las consultas de las interacciones extraídas mediante Web Scraping y almacenadas de forma estática puedan realizarse en el mismo dispositivo.

Como se comentó anteriormente, para el Web Scraping, en un principio se utilizó Python con Selenium para sacar los datos de las interacciones que hay en las tablas del portal la Guía Práctica del Medicamento, pero como las tablas no están en una dirección web fija, sino que las genera la web según las opciones que elija el usuario, se ha decidido utilizar la herramienta Octoparse para la extracción de los datos que hay en las tablas.

Para poder sacar la información de las tablas con Octoparse, debemos de descargar el HTML de la tabla generada para que no haya problemas a la hora de extraer la información. Una vez descargadas ponemos el enlace local del HTML descargado como se muestra en la Figura 40.



The screenshot shows the Octoparse web interface. At the top, there is a dropdown menu for 'Grupo de tarea' set to 'MyGroup' and a 'Nuevo grupo' button. Below that, there are buttons for 'Ingresar URL(s)', 'Importar desde archivo', 'Importar desde tarea', and 'Generar por lotes'. The 'Ingresar URL(s)' button is highlighted. Underneath, the 'Sitio web' field contains the local file path: 'file:///C:/Users/Mohamed/Downloads/Gu%C3%ADa%20pr%C3%A1ctica%20del%20medicamento.html'. A watermark for 'Biblioteca UNIVERSITAS Miguel Hernández' is visible in the background.

Figura 40: Ejemplo de como indicarle a Octoparse una página HTML para hacerle Web Scraping.



The screenshot shows a web browser displaying a table of drug interactions. The table has two columns: the first column lists drug names and the second column lists their interactions. A data preview window is open at the bottom of the browser, showing the first three rows of the table.

Núm.	Medicamento	Interacción
1	ACECLOFENACO	ALCOHOL
2	ACENOCUMAROL	ALCOHOL LECHE DE SO.
3	ACETILCISTEÍNA	RICOS EN CAL RICOS EN HIE

Below the table, there is a data preview window with the following text: 'Vista previa de datos: 120 línea(s) de datos & 1 campos capturados'. The preview shows the first three rows of the table.

Figura 41: Web Scraping a una columna de una tabla con Octoparse.

En esta web nos encontramos con un problema, ya que como se vio en la Figura 13, sus tablas tienen otras tablas dentro, lo cual hace difícil el sacar toda la tabla de manera correcta. Por ello, decidimos sacar la información de una tabla columna a columna (ver Figura 41), para después juntar esas columnas en un Excel. Finalmente, una vez tenemos el Excel con todos los datos, lo convertimos a un CSV y mediante la herramienta DB Browser importamos los datos a una tabla de la BBDD.

Para integrar la BBDD en Unity, lo primero que debemos de hacer es descargar y pegar en la carpeta de "Plugins", todos los archivos del plugin de MySQL de Mono Project para Unity. Después, en el script de `AppManager` debemos de añadir las diferentes librerías para MySQL de Mono Project (ver Figura 42). Después, creamos las variables para MySQL (ver Figura 43). Y como la BBDD tiene que funcionar en Android, debemos de crear en Unity una carpeta llamada "StreamingAssets"; todos los archivos colocados en esa carpeta se copiarán literalmente en una carpeta concreta del dispositivo de destino que en nuestro caso serán la BBDD local.

Finalmente, pondremos el código que se ve en la Figura 44, que lo que hace es poner la ubicación donde ponemos la BBDD que hemos copiado en Android como la ubicación para hacer consultas más tarde.

```
using Mono.Data.Sqlite;
using System;
using System.Data;
using System.IO;
```

Figura 42: Librerías para el funcionamiento de MySQL.

```
//mysql
private string conn, sqlQuery;
IDbConnection dbconn;
IDbCommand dbcmd;
private IDataReader reader;
string DatabaseName = "BaseDatos.db";
```

Figura 43: Variables de MySQL.

```
//Application database Path android
string filepath = Application.persistentDataPath + "/" + DatabaseName;
if (!File.Exists(filepath))
{
    WWW loadDB = new WWW("jar:file://" + Application.dataPath + "!/assets/BaseDatos.db");
    while (!loadDB.isDone) { }
    // then save to Application.persistentDataPath
    File.WriteAllBytes(filepath, loadDB.bytes);
}

conn = "URI=file:" + filepath;
```

Figura 44: Código de MySQL para la ubicación de la BBDD.

En la Figura 45, tenemos un ejemplo de consulta con la BBDD para obtener las interacciones con Alimentos que un principio activo tenga. Si hay interacciones, guardaremos en dos ArrayList, el alimento y la descripción de la interacción.

```
using (dbconn = new SqliteConnection(conn))
{
    string Alim_Search, Desc_Search;
    dbconn.Open(); //Open connection to the database.
    IDbCommand dbcmd = dbconn.CreateCommand();
    string sqlQuery = "SELECT alimento, descripcion " + "FROM Alimentos " +
    "WHERE principio LIKE '%" + principio + "%'";
    dbcmd.CommandText = sqlQuery;
    IDataReader reader = dbcmd.ExecuteReader();
    while (reader.Read())
    {
        Alim_Search = reader.GetString(0);
        Desc_Search = reader.GetString(1);
        listaAlimentos.Add(Alim_Search);
        listaDescripcion.Add(Desc_Search);
        hayInteracciones=true;
    }
    reader.Close();
    reader = null;
    dbcmd.Dispose();
    dbcmd = null;
    dbconn.Close();
}
}
```

Figura 45: Ejemplo de consulta con BBDD para interacciones con alimentos.

3.3.3.- Interfaz de usuario de la aplicación de RA

Ya terminadas las consultas tanto para la API de Computer Vision de Azure como para la API REST de CIMA, en este punto nos dedicaremos a hablar de la interfaz de usuario. Se ha optado por la creación de la interfaz con las herramientas de Unity. En la Figura 46 se muestra a modo de ejemplo como se ha montado la escena de la pantalla de entrada a la aplicación. Tenemos un GameObject, en el cual ponemos un "Canvas" y dentro de ese Canvas vamos colocando los diferentes elementos de la interfaz de usuario. Por ejemplo, tenemos dos "Input Field", tanto para el email como para la contraseña, tenemos también dos botones y una imagen con el logo de la aplicación.

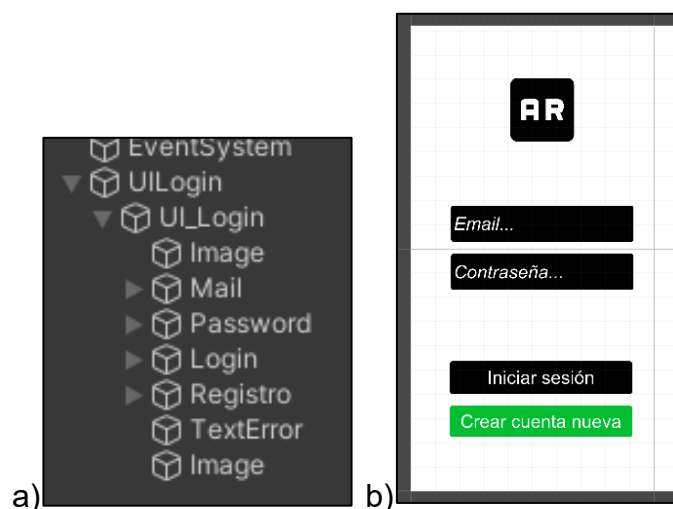


Figura 46: Ejemplo de interfaz de usuario.

Para crear un Canvas debemos ir al `Hierarchy view`, clic derecho del ratón → `UI` → `Canvas`, el Canvas se creará vacío y para ir añadiéndole elementos solo tenemos que hacer: clic derecho del ratón → `UI` → y elegir el elemento de `UI` que queremos. Ahora miraremos como se ha dispuesto la cámara, pues el usuario pasará la mayor parte del tiempo en las pantallas de información o interacción de los medicamentos, donde se estará ejecutando la cámara y haciendo capturas cada 5 segundos.

Como estamos utilizando el OCR de Azure, no necesitamos una cámara con funcionalidades de RA, la `Main Camera` nos hace el trabajo. El problema de la `Main Camera` es que no nos enseña lo que se está viendo por la cámara principal, para ello debemos de crear en un `Canvas` el llamado `Raw Image` (clic con el botón derecho del ratón en → `UI` → `Raw Image`). Al crearlo debemos de ajustarlo al tamaño del canvas, como se ve en la Figura 47.

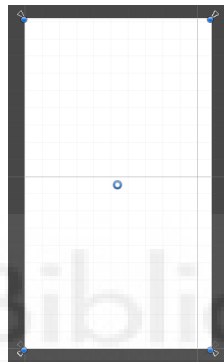


Figura 47: Ejemplo del uso de `Raw Image` interfaz de usuario.

Una vez creado el `Raw Image` debemos ir al script `CameraController` y poner el código que se ve en la Figura 48. Primero creamos 2 variables, una de tipo `RawImage`, que es a la que aplicaremos la textura de la cámara y otra de tipo `WebCamTexture` para que muestre todo lo que nuestra cámara este viendo. La variable “`camTex`” se usará para darle una anchura y altura a la `Texture2D` que se vio anteriormente en la Figura 31. Finalmente, en la función `Start()` debemos de crear la textura de la cámara.

```
// UI RawImage al que aplicamos la textura de la cámara web
public RawImage cameraProjection;

// textura que muestra lo que nuestra cámara está viendo
private WebCamTexture camTex;

void Start ()
{
    // creamos la textura de la cámara
    camTex = new WebCamTexture(Screen.width, Screen.height);
    cameraProjection.texture = camTex;
    camTex.Play();
}
```

Figura 48: Código para la textura de la cámara.

También debemos destacar que esta aplicación está creada a partir de varias escenas (por ejemplo, una escena es la pantalla de inicio, otra la de registro) y para cambiar entre escenas debemos de utilizar un “SceneManager”, para ello debemos de utilizar el namespace “UnityEngine.SceneManagement” y poner el código que se ve en la Figura 49.

```
public void LoadScene(string sceneName)
{
    SceneManager.LoadScene(sceneName);
}
```

Figura 49: SceneManager para cambiar entre escenas.

3.4.- Ejemplo de uso de la herramienta

En este punto, se describirán a modo de tutorial las diferentes pantallas que conforman la versión final de la aplicación desarrollada.

3.4.1.- Inicio de sesión

Esta es la primera pantalla que el usuario encontrará al abrir la aplicación. El usuario una vez registrado, puede iniciar sesión en la aplicación móvil. Como se ve en la Figura 50, el usuario deberá rellenar los 2 campos (email y contraseña) para poder iniciar sesión. Si el usuario o la contraseña son incorrectos, saltará un mensaje de error. Por último, desde esta pantalla el usuario puede acceder a la pantalla de registro.

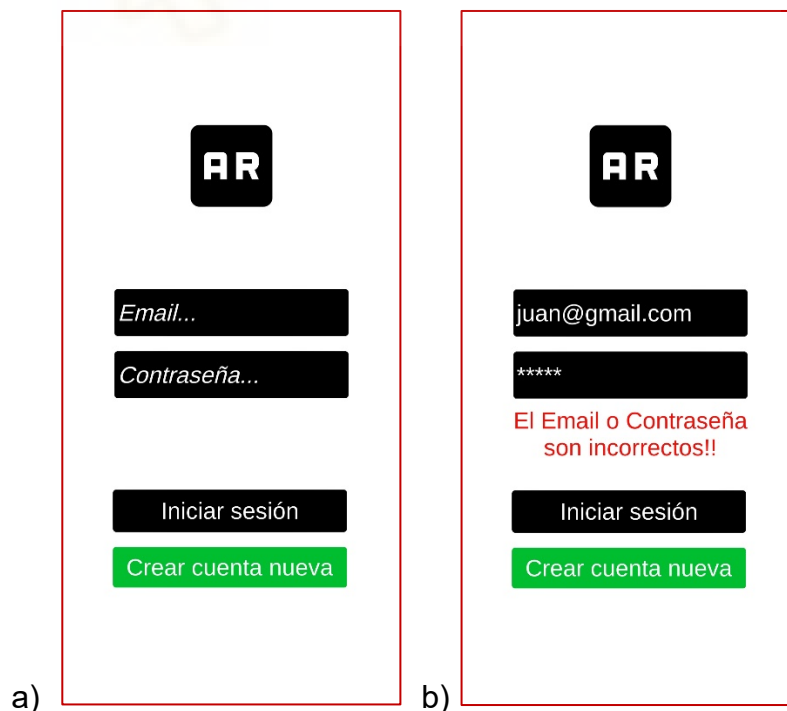


Figura 50: Pantalla de inicio de sesión.

3.4.2.- Registro de usuario

En esta pantalla (ver Figura 51), el usuario podrá registrar su cuenta. Para ello, deberá de rellenar los campos de Nombre, Apellidos, Teléfono, Email y Contraseña. Si el Email ya se está registrado, saltará un mensaje error. También, se podrá reiniciar los campos o volver a la pantalla de inicio de sesión. Finalmente, si el usuario inicia sesión se le llevará a la pantalla principal de la aplicación (la de información sobre los medicamentos).

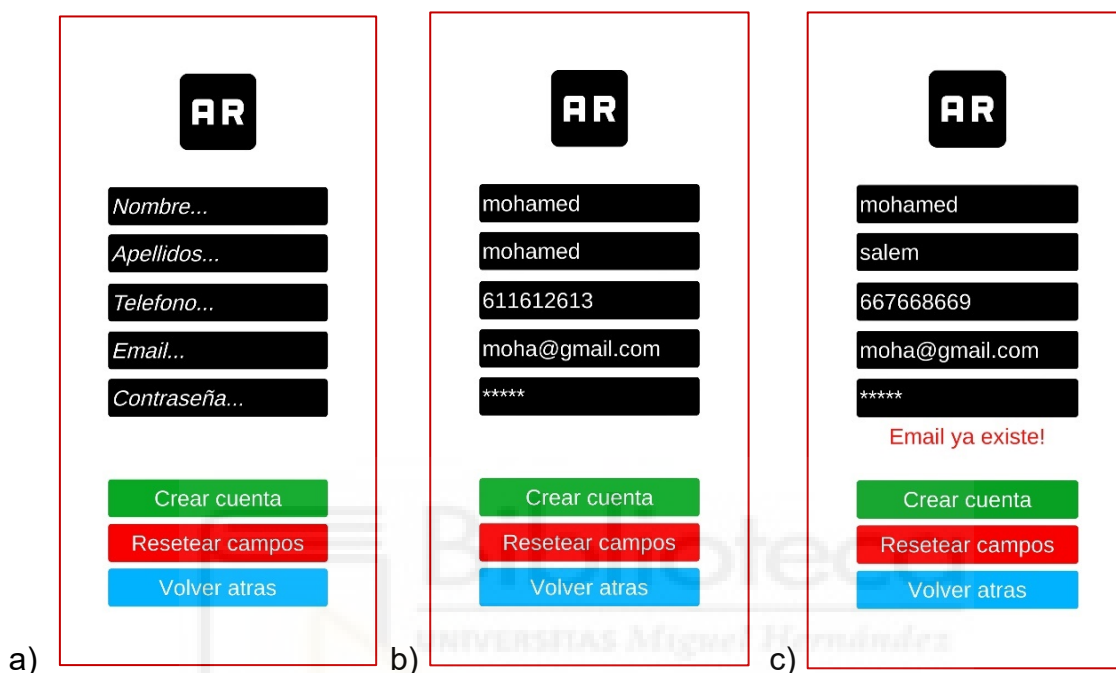


Figura 51: Registro de cuenta.

3.4.3.- Elección de Alergias

En esta pantalla (ver Figura 52), el usuario podrá escoger los principios activos a los que tenga alergias. Para ello, se dispone de un dropdown para escoger un principio activo. Una vez ya encontrado el principio, se escoge y se añade, al pulsar el botón de “Añadir”, nos saldrá una ventana emergente para aceptar o denegar el añadir el principio activo. Si se acepta, se añadirá a la lista. También, se dispone de la posibilidad de “Cerrar” que cancelará el registro de cuenta y nos llevará de vuelta a la pantalla de registro. Finalmente, se dispone del botón de “Finalizar” para finalizar el registro y nos llevará a la pantalla principal de la aplicación.

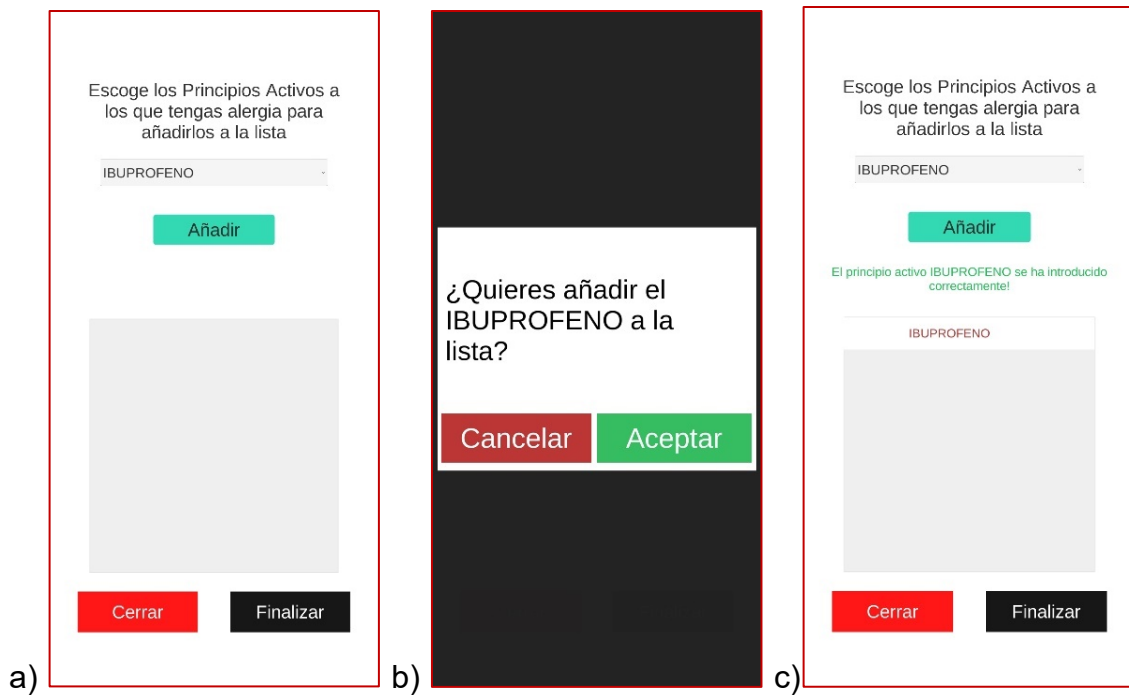


Figura 52: Elección de alergias.

3.4.4.- Información sobre un medicamento

En esta pantalla (ver Figura 53) el usuario podrá consultar la información de un medicamento. El usuario debe apuntar con la cámara del móvil a un medicamento y la aplicación de manera transparente al usuario aumentará el medicamento con la información extraída de manera automática de la web de CIMA (ver Figura 53 b)).

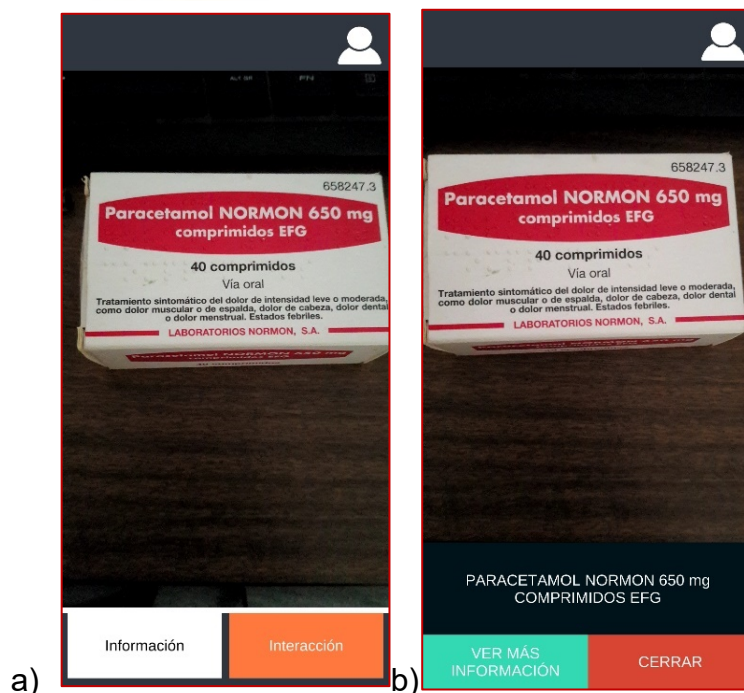


Figura 53: Pantalla de información de los medicamentos.

Además, se podrá ver más información de un medicamento, al pulsar en el botón de “VER MÁS INFORMACIÓN”. Eso hará que nos salga toda la información posible de un medicamento (ver Figura 54), por ejemplo, si se sigue comercializando, si necesita receta o no, sus documentos, principios activos, si se tiene alergia o no (ej. Figura 54 b)), si hay interacciones con alimentos, etc. Finalmente, si se pulsa en el botón de “Interacción” nos llevará a la pantalla de interacciones entre medicamentos.

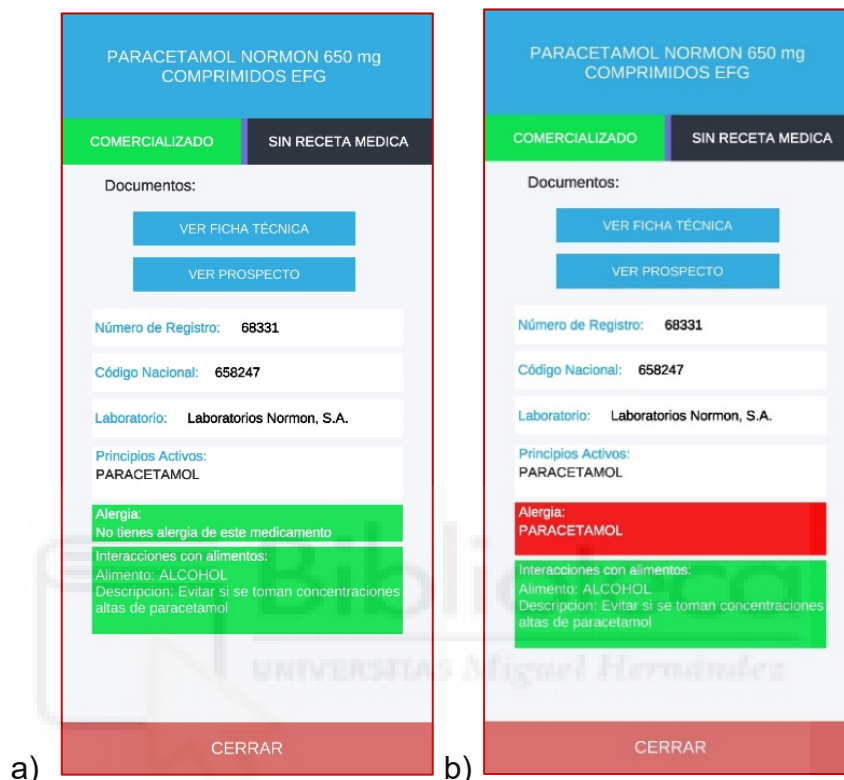


Figura 54: Información detallada de un medicamento.

3.4.5.- Interacción entre medicamentos

Finalmente, tenemos la última pantalla (ver Figura 55), donde el usuario podrá consultar las interacciones que puede haber entre medicamentos. Para ello, el usuario deberá apuntar a dos envases, uno a uno. La aplicación, por detrás realizará el mismo proceso que se describió en el punto anterior para sacar los principios activos que hay para cada medicamento y con esos principios se harán consultas a la base de datos para saber si hay interacciones entre ellos. Finalmente, si se pulsa en el botón de “Información” nos llevará a la pantalla de información de medicamentos.



Figura 55: Pantalla de interacción entre medicamentos.

3.4.6.- Cierre de sesión

Finalmente, desde esta y otras pantallas, si se pulsa en el icono de usuario, nos saldrá la posibilidad de cerrar sesión (ver Figura 56).

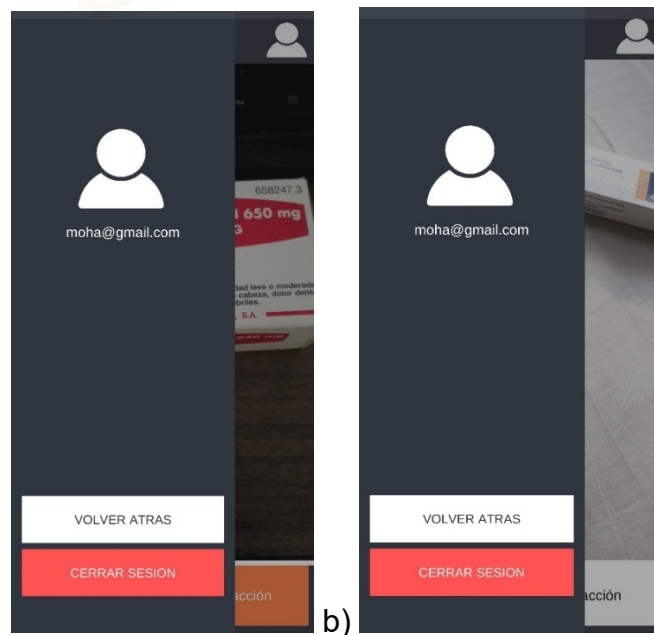


Figura 56: a) Opción de cerrar sesión desde la pantalla de “Información de medcamentos”. b) Opción de cerrar sesión desde la pantalla de “Interracción entre medcamentos”.

4. Conclusiones y trabajo futuro

Este proyecto, ha consistido en el desarrollo de una aplicación de RA que permite a los usuarios obtener información sobre medicamentos y saber si tienen alergia a dichos medicamentos a través del análisis de sus principios activos; también, permite saber que interacciones hay entre los medicamentos. Podemos decir, que esta aplicación puede ser muy útil para los usuarios para obtener información sobre un medicamento de manera directa, sin necesidad de rellenar ningún formulario o consultar distintas páginas webs sobre ese medicamento.

El desarrollo de esta aplicación ha requerido un análisis a fondo de ciertas aplicaciones y métodos para ofrecer información sobre un medicamento. Estas aplicaciones analizadas, abordaban principalmente: la opción de obtener información sobre un medicamento, posibilidad de ver las interacciones entre medicamentos y ver la disponibilidad que hay de un medicamento. Sin embargo, la mayoría de las aplicaciones de este ámbito necesitan que el usuario complete un formulario para consultar la información de un medicamento o para saber las interacciones entre medicamentos, y solo una de ellas hace uso de la RA mediante código de barras. Por ello, incluir una RA de más alto nivel y más compleja, es novedoso en este ámbito.

Desde el punto de vista técnico, podemos decir que el uso de tecnologías de la RA nos ha supuesto un esfuerzo importante. Pues, comparado con otras tecnologías vistas a lo largo del Grado, esta tecnología es nueva para nosotros y se ha tenido que navegar por la documentación de los diferentes SDKs.

En cuanto a la metodología para gestionar el proyecto, se ha decidido usar una metodología ágil de desarrollo del software. Gracias a esta metodología, hemos podido abordar los requisitos cambiantes que ha tenido este proyecto en los diferentes *sprints*. En nuestro caso, nos ha ayudado a realizar pruebas y analizar las dos formas de reconocimiento de medicamentos, para saber cuál de ellas es la más eficiente, más eficaz y la de mayor calidad para montar nuestra aplicación.

A lo largo del proyecto, se han encontrado diferentes problemas y limitaciones de los métodos que se han utilizado. En concreto, se ha confirmado que el método de reconocimiento de un medicamento por Vuforia no es del todo viable, ya que, si por un lado es un método muy bueno para el reconocimiento de objetos, tiene una limitación de escalabilidad. Este problema de escalabilidad se debe a que en el reconocimiento de embalajes de medicamentos debemos de utilizar los Image Target y se deben de ir añadiendo uno a uno. Además, se deberían definir tantos patrones como tipos de embalajes existan, y si se producen cambios en ellos la base de datos debería volver a reconstruirse.

Para solventar esta limitación, se decidió utilizar el método del OCR, el cual es un método muy potente y versátil, también nos permite una gran escalabilidad para nuestra aplicación al poder utilizar como referencia el Código Nacional de un medicamento. Este código lo podemos explotar para hacer consultas sobre la web de CIMA, es decir no necesitamos una BBDD propia que contenga toda la información de los medicamentos.

Pero como todo tiene limitaciones, por ejemplo, la calidad de la imagen a la que se le va a hacer OCR debe de ser aceptable. Si el texto que queremos reconocer en la imagen es borroso, tiene poca resolución o tiene otro problema no se conseguirá un resultado satisfactorio del OCR. También, se ha visto la limitación de que, si se intenta hacer OCR a más de un medicamento, la API no reconoce de manera correcta el texto de los medicamentos, por ello es mejor hacer OCR de medicamento en medicamento.

A pesar de esta limitación del OCR, hemos conseguido implementar una primera versión de la aplicación 100% funcional que permite gestionar de manera dinámica las interacciones que hay de los medicamentos y puede obtener información de un medicamento de manera directa ya que utilizamos la RA y estamos conectados a una web que se actualiza regularmente.

También, al conectarnos a un servidor de un tercero, en este caso Azure de Microsoft, nos supone algunos problemas de privacidad para el usuario. Ya que hay que tener en cuenta que cada 5 segundos la aplicación realiza una captura y la envía a Azure, esas capturas pueden contener datos personales. Por ello, por cuestiones de privacidad, cuando un usuario se registre deberá aceptar los términos y condiciones de Azure Microsoft. Además, aunque la versión gratuita de Azure nos ha permitido llevar a cabo este proyecto, se debería tener en cuenta su precio para valorar si es necesario monetizar nuestra aplicación para cubrir los gastos que se puedan generar.

4.1.- Trabajo futuro

Como ya hemos comentado, con este proyecto se ha conseguido desarrollar una primera versión de una aplicación móvil de RA que nos permite consultar la información de un medicamento y saber las interacciones que hay entre medicamentos. A partir de las pruebas realizadas con ella, se han identificado varias condiciones que permiten definir las siguientes líneas de trabajo.

- **Ampliar las funcionalidades de la aplicación de RA:** sería interesante en un proyecto futuro añadir otras funcionalidades como recordatorios para la toma de medicamentos. Otra funcionalidad, sería poder ver las últimas interacciones que se ha hecho o poder ver los últimos medicamentos de los que se ha buscado información.
- **Mejora de la BBDD de las interacciones:** en un proyecto futuro se podría mejorar la BBDD que utilizamos para las interacciones tanto con alimentos como entre medicamentos. También, para ello se podría utilizar un API de terceros que tenga la BBDD actualizada al día. Pues la BBDD que utilizamos actualmente no cubre todos los medicamentos y no se actualiza diariamente.
- **Alternativas gratuitas para el OCR:** en este proyecto utilizamos un OCR de pago, el cual nos salía a nosotros gratuito al ser estudiantes de la UMH. Si no fuéramos estudiantes, eso podría ser un obstáculo para nuestro proyecto, pues tendríamos que estar pagando mes a mes, por ello se

surgiere para un trabajo futuro buscar otras librerías que sean gratuitas para realizar OCR y que se puedan implementar en un script en C#.

- **Publicación de los patrones de RA basados en texto:** hoy en día no existen limitaciones en cuanto a la inclusión de librerías de patrones de RA basados en texto por parte de Vuforia. Por ello en un trabajo futuro se podría publicar la implementación del OCR que hemos hecho como una librería independiente en abierto en un repositorio como BitBucket (BitBucket, 2022), o incluso abordar si se podrían definir este tipo de patrones integrados en Vuforia.
- **Mejora de la autenticación de los usuarios:** en un trabajo futuro estaría bien mejorar el Login y Registro de usuarios. En este proyecto se ha gestionado de manera local para evitar que desplegar un servidor de bases de datos con el que conectar. A corto plazo, nos gustaría implementar este servicio a través de APIs de terceros como la autenticación con cuentas de Google o Microsoft, que gracias a su popularidad evitaría a nuestros usuarios tener que gestionar nuevas contraseñas.



5. Bibliografía

- Accuvein.com. (2022). *Accuvein*. <https://www.accuvein.com/>
- AEMPS.gob.es. (2014). *La AEMPS presenta una aplicación para dispositivos móviles para acercar la información de los medicamentos a profesionales y ciudadanos*. https://www.aemps.gob.es/informa/ni-aemps_01-2014-aplicacion-movil/
- AEMPS.gob.es. (2018). *CIMA REST API*. https://sede.aemps.gob.es/docs/CIMA-REST-API_1_19.pdf
- Alexander Sokhanych. (2021). *What is Augmented Reality (AR) and How does it work*. <https://thinkmobiles.com/blog/what-is-augmented-reality/>
- Allan Vélez. (2021). *Google Glass: de proyecto estrella a un retiro silencioso*. <https://es.digitaltrends.com/tendencias/google-glass-por-que-fracaso-lentes-inteligentes/>
- Anastasia Morozova. (2022). *Use of AR in Healthcare Education*. <https://jasoren.com/ar-in-healthcare-education/>
- Apple Developer. (2022). *Augmented Reality*. <https://developer.apple.com/augmented-reality/>
- AppleInsider.com. (2022). *Apple Glass*. <https://appleinsider.com/inside/apple-glass>
- AVRspot channel. (2018). *AR Aid application - augmented reality phobia treatment application*. https://www.youtube.com/watch?v=OhK4rkRQ67M&ab_channel=AVRspotchannel
- BitBucket. (2022). *BitBucket*. <https://bitbucket.org/product/>
- Bunny83. (2021). *SimpleJSON*. <https://github.com/Bunny83/SimpleJSON>
- Chris Hall. (2022). *Los mejores consejos y trucos de Pokémon Go*. <https://www.pocket-lint.com/es-es/videojuegos/noticias/138270-mejores-consejos-de-pokemon-go>
- COSMOS. (2018). *Ikea Place, su aplicación de realidad aumentada para decorar tu casa llega a los móviles Android con ARCore*. <https://www.xatakandroid.com/aplicaciones-android/ikea-place-su-aplicacion-de-realidad-aumentada-para-decorar-tu-casa-llega-a-los-moviles-android-con-arcore>
- Dean Madison. (2018). *The future of augmented reality in healthcare*. <https://healthmanagement.org/c/healthmanagement/issuearticle/the-future-of-augmented-reality-in-healthcare>
- Dr. Emilio Vicente, & Dra. Yolanda Quijano. (2022). *Realidad aumentada: la relación imprescindible entre cirugía y tecnología*.

- https://blogs.alimente.elconfidencial.com/bienestar/cirugia-de-precision/2021-07-13/realidad-aumentada-relacion-imprescindible-cirugia-tecnologia_3179724/
- Ebrary.net. (2021). *Navigation and Control*.
https://ebrary.net/123174/sociology/navigation_control
- Farmaceuticos.com. (2022). *Catálogo de productos y precios 2022 BOT PLUS*.
<https://www.farmaceuticos.com/wp-content/uploads/2021/03/Catalogo-Precios-Productos-2022.pdf>
- Fernanda Rodríguez. (2020). *AR 101 — Components of the Augmented Reality System (Part 3)*. <https://mafda.medium.com/ar-101-components-of-the-augmented-reality-system-part-3-878c71e68069>
- Fernando Torres. (2019). *CIMA REST API*.
<https://www.aemps.gob.es/informa/notasInformativas/medicamentosUsoHumano/problemasSuministro/2019/docs/CIMA-problemas-suministro.pdf>
- Flying Contributor. (2019). *SkyDisplay Head-Up Display Nears Certification Finish Line*. <https://www.flyingmag.com/skydisplay-nears-certification/>
- FutureVisual.com. (2021). *Benefits of Augmented Reality in Healthcare*.
<https://www.futurevisual.com/blog/benefits-augmented-reality-healthcare/>
- García Jiménez, F., & Casado Parada, I. (2019). *Rafodiun*.
<https://ra.sav.us.es/index.php/realidad-aumentada/36-rafodiun>
- Google. (2019). *ARCore Augmented Images*.
<https://clmirror.storage.googleapis.com/codelabs/augimg-intro/index.html#0>
- Google Developers. (2022). *Descripción general de ARCore y entornos de desarrollo compatibles*. <https://developers.google.com/ar/develop>
- guiacinfadelmedicamento.com. (2022). *Guía práctica Cinfa del Medicamento*.
<https://guiacinfadelmedicamento.com/>
- IAT.es. (2020). *LA REALIDAD AUMENTADA EN MEDICINA SALVA Y MEJORA VIDAS*. <https://iat.es/tecnologias/realidad-aumentada/medicina/>
- iDoctus.com. (2022). *IDoctus*. <https://int-public.idoctus.com/>
- INEXTRIX.com. (2019). *Understanding the Types of Augmented Reality App*.
<https://inextrix.com/blog/understanding-types-augmented-reality-app>
- Iván Asensio. (2021). *Qué es Unity y para qué sirve*.
<https://www.masterd.es/blog/que-es-unity-3d-tutorial>
- Ivy Decker. (2018). *Augmented Reality for Automotive DIY*.
<https://blog.anthonythomas.com/augmented-reality-for-automotive-diy>
- Jiménez, M., Piqueras, J., Mateu, O., Carballo, J., Orgilés, M., & Espada, J. (2012). *Diferencias de sexo, características de personalidad y*

afrentamiento en el uso de Internet, el móvil y los videojuegos en la adolescencia. Health and Addictions (0 ed., Vol. 0).

J.M. Sánchez. (2017). *La realidad virtual frente a la realidad aumentada: ¿cuál es de verdad el futuro a seguir?*

https://www.abc.es/tecnologia/informatica/soluciones/abci-facebook-realidad-virtual-frente-realidad-aumentada-cual-verdad-futuro-seguir-201704192150_noticia.html

Joshua Harris. (2018). *What Is OCR And What Is It Used For?*

<https://docparser.com/blog/what-is-ocr/>

Kavit Majithia. (2016). *Pokémon Go – the AR revolution is real.*

<https://www.mobileworldlive.com/blog/pokemon-go-the-revolution-is-real>

Kieran Harris. (2021). *5 mil millones de dólares en 5 años: la trayectoria de Pokémon Go.* <https://www.gamereactor.es/5-mil-millones-de-dolares-en-5-anos-la-trayectoria-de-pokemon-go/>

Lisa Nissen, Alireza Ahmadvand, Judy Drennan, Jean Burgess, Michele Clark, David Kavanagh, Kara Burns, Sarah Howard, Fleur Kelly, & Chris Campbell. (2018). *Novel augmented reality solution for improving health literacy around antihypertensives in people living with type 2 diabetes mellitus: protocol of a technology evaluation study .*

<https://bmjopen.bmj.com/content/8/4/e019422>

lucidchart.com. (2022). *Lucidchart.* <https://www.lucidchart.com>

Maria Esther Fernández Sal. (2021). *La receta electrónica.*

<https://revistamedica.com/receta-electronica-ventajas/>

Martínez Parra, E. (2020). *Desarrollo de una herramienta para la detección de interacciones entre medicamentos mediante Realidad Aumentada.*

<http://dspace.umh.es/jspui/handle/11000/8541>

Medgadget.com. (2014). *Surgical Navigation Advanced Platform (SNAP) for Intra-Op Visualization of Patient's Brain.*

<https://www.medgadget.com/2014/07/surgical-navigation-advanced-platform-snap-for-intra-op-visualization-of-patients-brain.html>

Microsoft Azure. (2021). *¿Qué es Computer Vision?*

<https://docs.microsoft.com/es-es/azure/cognitive-services/computer-vision/overview>

Microsoft Azure. (2022). *Computer Vision API (v3.2).*

mono-project.com. (2021). *MySQL.* <https://www.mono-project.com/docs/database-access/providers/mysql/>

Morales Padrón, A. (2018). *Medicamentos.*

<https://play.google.com/store/apps/details?id=es.amp.medicamentos&gl=ES>

- MyTherapy. (2022). *Recordatorio de Medicación*.
<https://play.google.com/store/apps/details?id=eu.smartpatient.mytherapy&gl=ES>
- Naciones Unidas. (2022). *Objetivo 3: Garantizar una vida sana y promover el bienestar para todos en todas las edades*.
<https://www.un.org/sustainabledevelopment/es/health/>
- Naveen Joshi. (2021). *4 Applications Of AR In Pharmaceuticals*.
<https://www.allerin.com/blog/4-applications-of-ar-in-pharmaceuticals>
- notepad-plus-plus.org. (2022). *What is Notepad++*. <https://notepad-plus-plus.org/>
- octoparse.es. (2022). *Herramienta de Web Scraping Gratis*.
<https://www.octoparse.es/>
- ONC Health IT. (2011). *Health IT: Advancing America's Health Care*.
<https://www.healthit.gov/sites/default/files/pdf/health-information-technology-fact-sheet.pdf>
- Paula Nicole Roldán. (2017). *Tecnología*. Economipedia.Com.
<https://economipedia.com/definiciones/tecnologia.html>
- Pranav Mistry. (2021). *SixthSense*.
<https://www.pranavmistry.com/archived/projects/sixthsense/>
- proyectosagiles.org. (2022). *Qué es SCRUM*. <https://proyectosagiles.org/que-es-scrum/>
- ptc.com. (2021). *Desarrolle experiencias de RA con Vuforia Engine*.
<https://www.ptc.com/es/products/vuforia/vuforia-engine>
- Quironsalud.es. (2020). *Quironsalud ya trabaja con la receta electrónica privada*. <https://www.quironsalud.es/en/virtual-press-room/press-releases/quironsalud-trabaja-receta-electronica-privada>
- Raykarimi.com. (2020). *Tipos de realidad aumentada*.
<https://www.raykarimi.com/es/types-of-augmented-reality>
- Rodriguez Sala, J. J., Martinez, J. P., & Torres Lopez, J. A. (2019). *Ingeniería del Software*.
https://www.umh.es/contenido/estudios/:asi_g_2792_S8/datos_es.html
- selenium.dev. (2022). *Selenium automates browsers. That's it!*
<https://www.selenium.dev/>
- Sergio Koller. (2021). *¿Qué es el web scraping, para qué sirve y cómo funciona?* <https://seranking.com/es/blog/web-scraping/>
- Sobkin, V., & Evstigneeva, I. (2004). *The Place of the New Information Technologies in the Life of Today's Adolescent. Russian Education and Society* (0 ed., Vol. 0).

- Softtek.eu. (2021). *¿Cuáles son los diferentes tipos de Realidad Aumentada?*
<https://softtek.eu/tech-magazine/user-experience/cuales-son-los-diferentes-tipos-de-realidad-aumentada/>
- sqlitebrowser.org. (2021). *DB Browser for SQLite*. <https://sqlitebrowser.org/>
- Steve Jobs. (2007). *Macworld 2007*. Allaboutstevejobs.
https://allaboutstevejobs.com/videos/keynotes/macworld_2007
- Tatiana Grapsas. (2019). *Conoce la realidad aumentada y las posibilidades de interacción que la hacen sobresalir en el mundo digital*.
<https://rockcontent.com/es/blog/realidad-aumentada/>
- Two Reality. (2021). *GAFAS MICROSOFT HOLOLENS. REALIDAD AUMENTADA SIN LÍMITES*. <https://www.tworeality.com/gafas-virtuales/microsoft-hololens-realidad-aumentada/>
- Vademecum.es. (2011). *APP Vademecum Internacional*.
<https://www.vademecum.es/productos-vademecum-app+vademecum+internacional-49>
- Vuforia. (2021). *Working with Device Databases*.
<https://library.vuforia.com/objects/device-databases>
- Vuforia. (2022). *Image Targets*. <https://library.vuforia.com/objects/image-targets>
- Yúbal Fernandez. (2020). *Qué es el 5G y qué diferencias tiene con el 4G*.
<https://www.xataka.com/basics/que-5g-que-diferencias-tiene-4g>

ANEXO I: Descripción de los casos de uso

En las siguientes tablas se incluyen los casos de uso desarrollados en el apartado de análisis y diseño de software. Se han desarrollado los siguientes casos de uso:

1. Caso de Uso 1: Registro.
2. Caso de uso 2: Registrar alergias.
3. Caso de uso 3: Iniciar sesión.
4. Caso de uso 4: Información del medicamento.
5. Caso de uso 5: Consulta detallada del medicamento.
6. Caso de uso 6: Interacción entre medicamentos.
7. Caso de uso 7: Cerrar Sesión.

CU-01	Registro	
Actores	Todos los usuarios no registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario no registrado quiera registrarse en la aplicación de RA	
Precondición	El usuario no debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario entra en la aplicación, donde lo primero que le saldrá es la pantalla de Iniciar Sesión
	2	El usuario pincha en el botón de "Crear cuenta nueva" lo que lo lleva a la pantalla de Registro
	3	El usuario rellena los campos de nombre, apellidos, teléfono, dirección de correo y contraseña
	4	El sistema comprueba que los datos introducidos no están guardados en la base de datos
	5	Si el correo electrónico ya existe, el sistema imprime un mensaje de error y permite al usuario repetir el intento (pasos 3–4)
6	Si el correo electrónico no existe, el sistema redirige al usuario a la pantalla de selección de alergias	
Postcondición	Con la cuenta validada el usuario puede introducir si quiere las alergias que tenga a ciertos principios activos	
Excepciones	Paso	Acción
	6	El usuario cancela la operación. El caso de uso termina

Tabla AI.1: Caso de uso 1: Registro.

CU-02	Registrar alergias	
Actores	Todos los usuarios no registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario no registrado quiera registrar sus alergias	
Precondición	El usuario debe de completar el formulario de registro	
Secuencia	Pasos	Acción
	1	El usuario termina de completar de manera correcta el formulario de registro y el sistema lo envía a la pantalla de registro de alergias
	2	El usuario elige el principio activo al que tiene alergia y pulsa el botón de "Añadir"
	3	El sistema hace saltar un mensaje al usuario, preguntándole si quiere añadir ese principio activo a la lista de alergias
	4	El usuario pulsa el botón de "Aceptar" y el principio activo se añade a la lista de alergias
	5	El usuario podrá añadir cuantos principios quiera a la lista de alergias (pasos 2-4)
	6	El usuario termina de elegir y pulsa el botón de "Finalizar" y el sistema lo envía a la pantalla principal de la aplicación
Excepciones	Paso	Acción
	4	El usuario cancela la operación. El principio activo no se añade a la lista de alergias
	6	El usuario cancela la operación. El caso de uso termina

Tabla AI.2: Caso de uso 2: Registrar alergias.

CU-03	Iniciar sesión	
Actores	Todos los usuarios registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario registrado quiera iniciar sesión en la aplicación de RA	
Precondición	El usuario debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario entra en la aplicación, donde lo primero que le saldrá es la pantalla de Iniciar Sesión
	2	El sistema solicita al usuario su correo electrónico y su contraseña
	3	El usuario proporciona al sistema su correo electrónico y su contraseña
	4	El sistema comprueba si el correo electrónico y la contraseña son correctos
	5	Si el correo electrónico y la contraseña no son correctos, el sistema imprime un mensaje de error y permite al usuario repetir el intento (pasos 2-4)
	6	Si el correo electrónico y la contraseña son correctos, el sistema permite el acceso al usuario y lo redirige a la pantalla principal, la de información sobre un medicamento

Tabla AI.3: Caso de uso 3: Iniciar sesión.

CU-04	Información del medicamento	
Actores	Todos los usuarios registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario registrado quiera saber la información de un medicamento al apuntar al medicamento con la cámara de la aplicación	
Precondición	El usuario debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario al iniciar sesión o al registrarse sera redirigido a la pantalla principal donde podra consultar la información de un medicamento
	2	El usuario apunta con la cámara al embalaje de un medicamento
	3	El sistema realizara OCR al embalaje sacando de el código nacional, el cual usara para hacer una consulta medicante Unity Rest a la web de CIMA
	4	Si no se encuentra el medicamento, el sistema imprime un mensaje en la pantalla y permite al usuario repetir el intento (pasos 2-4)
5	Si se encuentra el medicamento, el sistema una ventana con el nombre y un boton para ver más información detallada del medicamento	
Postcondición	Se consulta la base de datos para las interacciones con alimentos y para ver si el usuario tiene alergia a ese medicamento	

Tabla AI.4: Caso de uso 4: Información del medicamento.

CU-05	Consulta detallada del medicamento	
Actores	Todos los usuarios registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario registrado quiera saber la información detallada de un medicamento	
Precondición	El usuario debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario pulsa el boton de ver más información que aparece cuando se apunta a un medicamento
2	El sistema muestra toda la información posible del medicamento al usuario, por ejemplo, si se comercializa, su ficha tecnica, sus pricipios activos, si tiene alergia al medicamento o las interacciones con alimentos que tenga el medicamento	

Tabla AI.5: Caso de uso 5: Consulta detallada del medicamento.

CU-06	Interacción entre medicamentos	
Actores	Todos los usuarios registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario registrado quiera saber las interacciones que tengan 2 medicamentos al apuntar la cámara a los 2 medicamentos	
Precondición	El usuario debe de estar registrado	
Secuencia	Pasos	Acción
	1	El usuario pulsa el boton de Interacciones que hay en la pantalla principal para ser redirigido a la pantalla de interacciones
	2	El usuario apunta con la cámara al embalaje del primer medicamento
	3	El usuario apunta con la cámara al embalaje del segundo medicamento
	4	El sistema realizara OCR al embalaje de los 2 medicamentos sacando de el código nacional, el cual usara para hacer una consulta medicante Unity Rest a la web de CIMA
	5	El sistema saca los principios activos de cada medicamento y hace una consulta a la base de datos para ver si hay interacciones entre los principios de los 2 medicamentos
	6	Si no hay interacciones, el sistema imprime un mensaje en la pantalla
	7	Si hay interacciones, el sistema muestra una venta donde aparece el nombre de los 2 medicamentos y la descripción de las interacciones que hay entre los 2 medicamentos

Tabla AI.6: Caso de uso 6: Interacción entre medicamentos.

CU-07	Cerrar Sesión	
Actores	Todos los usuarios registrados	
Descripción	El sistema deberá comportarse tal y como se describe en el siguiente caso de uso cuando un usuario registrado quiera cerrar sesión.	
Precondición	El usuario debe haber iniciado sesión	
Secuencia	1	El usuario entra en su perfil
	2	El usuario selecciona el apartado "Cerrar Sesión"
	3	El sistema lleva al usuario a la pantalla de Iniciar Sesión

Tabla AI.7: Caso de uso 7: Cerrar Sesión.

ANEXO II: Uso de patrones Vuforia

En este punto se describe paso a paso la definición de patrones con Vuforia y su integración en el motor gráfico Unity.

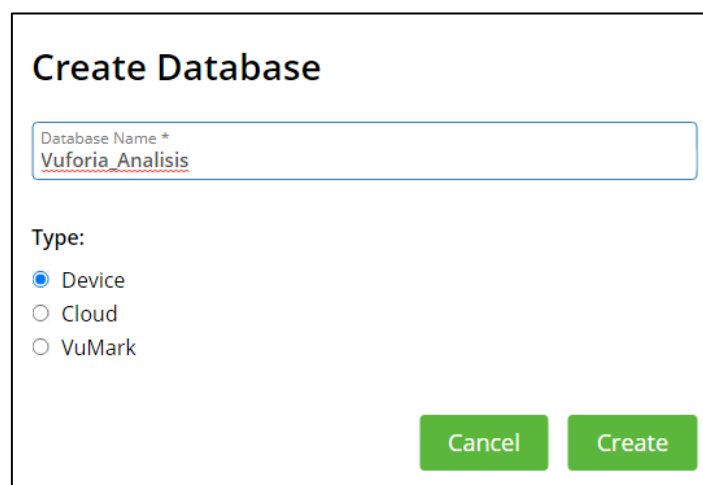
All.1.- Creación de los Image Targets en Vuforia

Como ya hemos visto, para el reconocimiento de objetos, Vuforia utiliza los llamados “*Image Targets*” los cuales representan imágenes que el motor de Vuforia puede detectar y rastrear.

Los *Image Targets* pueden utilizar cualquier imagen plana que proporcione suficiente detalle para ser detectada por el motor de Vuforia y para crear los *Image Targets* debemos crear una base de datos en el Target Manager de Vuforia. Esta última, es una herramienta basada en la web que nos permite crear y gestionar bases de datos de objetivos en el portal para desarrolladores de Vuforia. A la hora de crear la base de datos (ver Figura All.1), debemos elegir entre 3 tipos de bases de datos:

- **Device:** La base de datos, es una base de datos local de imágenes, la cual se almacena en nuestro dispositivo.
- **Cloud:** La base de datos, es una base de datos en la nube de imágenes, a la que demos de conectarnos a internet para hacer consultas.
- **VuMark:** La base de datos, es una base de datos de VuMark, el cual es un código visual que indica al usuario que una experiencia de RA está disponible, esta base de datos se almacena en nuestro dispositivo.

En nuestro caso elegiremos la de Device, para así poder descargar la base de datos e indexarla a Unity.



The image shows a web-based form titled "Create Database". At the top, there is a text input field labeled "Database Name *" with the text "Vuforia Analisis" entered. Below this, there is a section labeled "Type:" with three radio button options: "Device" (which is selected), "Cloud", and "VuMark". At the bottom right of the form, there are two buttons: "Cancel" and "Create".

Figura All.1: Creación de la BBDD en el Target Manager

Después de crear la base de datos, debemos crear los *Image Targets*, para ello, debemos añadir imágenes a la base de datos que hemos creado en Vuforia para

posteriormente utilizarlas como marcadores. Como vemos en la Figura AII.2, tenemos una especie de formulario para rellenar, mediante el cual podremos crear un *Image Target* en la base de datos de Vuforia. Lo primero que vemos es el tipo de imágenes de objetos que el Target Manager soporta:

- **Single Image:** utiliza imágenes para la creación de *Image Targets*
- **Cuboid:** sirve para crear un *Multi Target*, la cual es una colección de múltiples *Image Targets* combinados en una disposición geométrica definida, como cajas.
- **Cylinder:** sirve para crear un *Cylinder Target*, el cual nos permiten detectar y rastrear imágenes envueltas en formas cilíndricas y cónicas.
- **3D Object:** nos sirve para crear *Model Targets*, permite a Vuforia reconocer y rastrear objetos particulares en el mundo real basándose en la forma del objeto.

Elegiremos el de tipo “Single Image”, luego debemos elegir como archivo una imagen ya sea JPG o PNG de tamaño que no sea mayor a 2MB. Después, debemos elegir el tamaño y el nombre que tendrá la imagen de objeto, además, este proceso se tiene que ir repitiendo cada vez que vallamos a crear un *Image Target*.

Add Target

Type:

Single Image Cuboid Cylinder 3D Object

File:

Choose File Browse...

.jpg or .png (max file 2mb)

Width:

Enter the width of your target in scene units. The size of the target should be on the same scale as your augmented virtual content. Vuforia uses meters as the default unit scale. The target's height will be calculated when you upload your image.

Name:

Name must be unique to a database. When a target is detected in your application, this will be reported in the API.

Cancel Add

Figura AII.2: Creación de un Image Target en la BBDD

Como vemos en la Figura AII.3, después de crear los *Image Targets*, estos aparecen en la base de datos, donde se nos indica el nombre, el tipo, la calificación, el estatus y la fecha de modificación para cada *Image Target*.



Add Target		Download Database (All)			
<input type="checkbox"/> Target Name	Type	Rating [ⓘ]	Status [▼]	Date Modified	
<input type="checkbox"/>  paracetamol_normon	Single Image	★★★★☆	Active	Mar 19, 2022 17:11	
<input type="checkbox"/>  fluimucil-forte	Single Image	★★★★☆	Active	Mar 19, 2022 16:52	

Figura All.3: Gestión de una BBDD en el Target Manager de Vuforia

Si elegimos uno de los *Image Target*, por ejemplo, el de paracetamol_normon podremos editar su nombre, eliminarlo, actualizar el objetivo y, como se ve en la Figura All.4, mostrar las características. Una vez ya creados todos los *Image Targets* podremos descargar la base de datos, para luego indexarla en Unity como veremos más adelante. La base de datos se puede descargar para la plataforma de Unity, pero también se puede descargar para Android Studio, Xcode o Visual Studio.



Figura All.4: Características en la imagen de un medicamento

All.2.- Implementación de Vuforia en Unity

Para poder utilizar Vuforia, debemos de tener una Clave de Licencia, la cual podremos obtener en en “License Manager”, en nuestro caso elegiremos la opción Básica. Una vez creado el proyecto de Unity, tenemos que importar Vuforia Engine para poder trabajar con el dentro de Unity, para ello debemos de descargar Vuforia y luego dentro de Unity ir a Assets → Import Package → Custom Package, seguidamente debemos elegir la versión de Vuforia que hemos descargado, en nuestro caso la versión 10.5.5, y finalmente importamos todo el paquete de Vuforia.

Una de los aspectos más importantes de una experiencia de RA, es la gestión de cámaras de la escena, Unity por defecto trae una cámara llamada “Main Camera”, en cambio Vuforia trae la llamada “ARCamera”, la cual se trata de una cámara que tiene de base las funcionalidades de la Main Camera de Unity, pero incluye el VuforiaBehaviour para añadir soporte para la RA para dispositivos móviles o gafas de RA.

Por ello, en nuestra escena debemos de eliminar la Main Camera e incluir la ARCamera de Vuforia. Una vez incluida esta última, debemos de ir al panel de

configuración de Vuforia y añadimos la Clave de Licencia que hemos creado anterior mente, como se ve en la Figura AII.5, pues si no le añadilos la clave la cámara no funcionará.

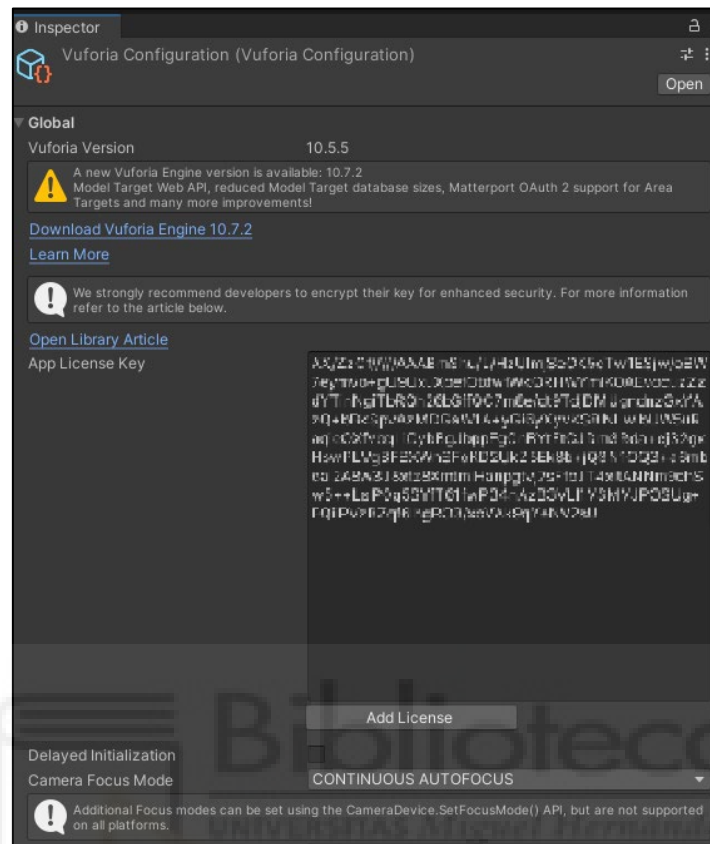


Figura AII.5: Panel de configuración de Vuforia Engine

Despues de añadir la clave en el panel de configuración, debemos de añadir la BBDD de Vuforia que hemos descargado anteriormete, para ello vamos una vez mas a Assets → Import Package → Custom Package y elegimos la base de datos e importamos todo lo que hay en el paquete. Ahora, debemos de crear un *Image Target* en unity, para esto vamos al Hierarchy view lo creamos y le enlazamos la BBDD que hemos descargado y elegimos uno de los *Image Target* que hay en la BBDD, como se puede ver en la Figura AII.6.

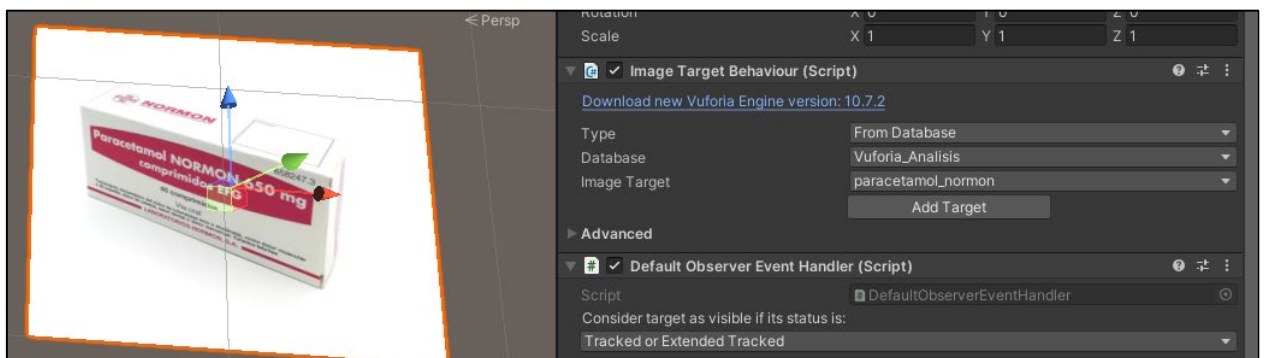


Figura AII.6: Ejemplo de Image Target

Una vez creado el *Image Target* podemos enlazarle un objeto para que a la hora de que apuntemos con la cámara, por ejemplo, a un medicamento nos salga el objeto en pantalla. El objeto que enlazamos al *Image Target*, puede ser un objeto 3D o un Canvas donde aparece texto, en nuestro caso será este último, como podemos ver en la Figura AII.7 a).



Figura AII.7: Ejemplo de canvas y resultado de la RA de Vuforia.

Una vez ya enlazado el objeto que hemos creado al *Image Target* solo nos queda compilar el proyecto y probarlo en un dispositivo móvil, ver Figura AII.7 b).