

**Universidad Internacional de La Rioja (UNIR)**

**ESIT**

**Máster Universitario en Industria 4.0**

# Realidad aumentada para entrenamiento de operadores de producción y montaje

**Trabajo Fin de Máster**

**presentado por:** Arbós Gómez, Luis

**Director/a:** Bueno Manglano, Jorge Juan

## Resumen

Este trabajo final de máster tiene por objeto plantear y desarrollar un software de realidad aumentada aplicado al ámbito industrial, concretamente al entrenamiento de operadores de producción y montaje.

Para cumplir con este objetivo se estudiarán cuáles son las metodologías más utilizadas para poder plantear el desarrollo en cuestión. Si es más apropiado desarrollar la aplicación para Android, iOS, Windows, Linux, Mac, o alguna otra opción, como implementarlo sobre un dispositivo inteligente. Posteriormente, se planteará y llevará a cabo un desarrollo del sistema, un primer prototipo, ya sea simulado o físico, y posteriormente se evaluará, se modificará, se expondrán todos los problemas que hayan surgido en la realización del primer prototipo.

Finalmente se analizarán las ventajas y desventajas, así como los costes y rentabilidad que supondría llevar la aplicación al mercado industrial, ofreciéndola como un PaaS, una plataforma como servicio.

**Palabras clave:** Realidad aumentada, entrenamiento de operarios, Platform-as-a-Service (PaaS), aplicación para Android

## Abstract

This master's final project aims to purpose and develop augmented reality software applied to the industrial field, specifically to the training of production and assembly operators.

To reach the objective, the most used methodologies will be studied to be able to propose the development. It is mandatory to know if is more appropriate to develop the application for Android, iOS, Windows, Linux, Mac, or some other option, such as deploying it on a smart device. Subsequently, a development of the system will be proposed and carried out, getting ready the first prototype, whether simulated or physical, and later it'll be evaluated, modified, and all the problems that have appeared in the realization of the first prototype will be exposed.

Finally, the advantages and disadvantages will be analyzed, as well as the costs and the profitability of bringing the app to the industrial market, offering it as a PaaS, a Platform-as-a-Service.

**Keywords:** Augmented reality, workers training, Platform-as-a-Service (PaaS), Android app

## Índice de contenidos

1. Introducción .....	9
1.1. Motivación .....	9
1.2. Planteamiento del trabajo .....	10
1.3. Estructura de capítulos .....	10
2. Contexto y estado del arte .....	12
2.1. Descripción general del contexto del proyecto .....	12
2.2. Proyectos relacionados con el tema del TFM .....	13
2.2.1. Realidad virtual y realidad aumentada: desarrollo de aplicaciones. ....	13
2.2.2. Diseño de la interfaz de una aplicación móvil para realizar visitas turísticas con realidad aumentada (RA) en Teruel. ....	15
2.2.3. La realidad aumentada en la formación del profesorado. Una experiencia en las prácticas del Máster de Profesorado de Enseñanza Secundaria. ....	18
2.2.4. Blended learning y realidad aumentada: experiencias de diseño docente .....	19
2.2.5. La innovación en el aula universitaria a través de la realidad aumentada. Análisis desde la perspectiva del estudiantado español y latinoamericano. ....	21
2.3. Tecnologías relacionadas con el tema del TFM .....	22
2.3.1. Realidad aumentada (RA) .....	22
2.3.2. Industrial Internet of Things (IIoT) .....	25
2.3.3. Cloud Computing .....	26
2.4. Conclusiones sobre el estado del arte .....	28
3. Descripción general de la contribución del TFM .....	30
3.1. Objetivos .....	30
3.2. Metodología del trabajo .....	30
3.3. Descripción general de las partes o componentes de la propuesta .....	31
4. Desarrollo específico de la contribución .....	35
4.1. Diseño de baja fidelidad .....	36
4.1.1. Elección de sistema operativo .....	36

4.1.2.	Elección del software de desarrollo .....	38
4.1.3.	Tecnologías implicadas: Diagrama .....	40
4.1.4.	Diseño de baja fidelidad .....	42
4.2.	Diseño del software .....	48
4.2.1.	Creación de proyecto en Android Studio.....	48
4.2.2.	Creación y configuración de las distintas pantallas .....	49
4.2.3.	Primeras pruebas de simulación del proyecto en Android Studio.....	51
4.2.4.	Incorporación del plugin de RA, configuración de la realidad aumentada y pruebas de funcionamiento.....	57
4.2.5.	Dificultades encontradas en el desarrollo .....	59
4.3.	Adaptación a PaaS.....	61
4.3.1.	¿Qué ventajas ofrece la salida a mercado de la aplicación como PaaS? .....	62
4.3.2.	KPIs y resultados esperados .....	63
4.3.3.	Costes y retorno de la inversión .....	71
5.	Conclusiones y trabajos futuros .....	74
5.1.1.	Líneas de trabajo futuras .....	75
	Referencias bibliográficas .....	77
Anexo A.	Código implementado XML.....	80
Anexo B.	Código implementado JAVA .....	91

## Índice de figuras

Figura 1: Diferencias entre Realidad Virtual, Realidad Aumentada y Realidad Mixta (Cámara de Valencia, 2021).....	14
Figura 2: Prototipo de baja fidelidad (Martínez Olivar, 2020).....	17
Figura 3: Primer prototipado desarrollado (Martínez Olivar, 2020) .....	18
Figura 4: Objetos en Realidad Aumentada para la formación en medicina (Cabero Almenara et al., 2018).....	20
Figura 5: David Mizell con el primer prototipo de realidad aumentada de Boening (Schmalstieg et al., 2016).....	23
Figura 6: Google Glass Enterprise Edition 2 (Google LLC, 2022) .....	24
Figura 7: Microsoft HoloLens 2 (Microsoft Corporation, 2022) .....	24
Figura 8: Diferencia entre IIoT y IoT (Bin Zikria et al., 2019) .....	26
Figura 9: Modelos de servicios cloud (Stackscale, 2021) .....	27
Figura 10: Metodología SCRUM (The Drew Company, 2019).....	31
Figura 11: Android e iOS, un duopolio (Statista, 2021) .....	37
Figura 12: Diagrama de tecnologías implicadas (Elaboración Propia) .....	41
Figura 13: Diseño de baja fidelidad. Pantalla de inicio de sesión (Elaboración propia) .....	43
Figura 14: Diseño de baja fidelidad. Pantalla principal (Elaboración Propia).....	44
Figura 15: Diseño de baja fidelidad. Menú Ajustes (Elaboración Propia) .....	45
Figura 16: Diseño de baja fidelidad. Menú Escanear (Elaboración Propia) .....	46
Figura 17: Diseño de baja fidelidad. Menú Cerrar Sesión (Elaboración Propia) .....	47
Figura 18: Pantalla de creación de proyecto Android Studio (Elaboración propia) .....	49
Figura 19: Simulación de la pantalla de acceso (Elaboración propia) .....	52
Figura 20: Pantalla de acceso con credenciales erróneas (Elaboración propia).....	53
Figura 21: Pantalla principal (Elaboración propia).....	54
Figura 22: Pantalla Ajuste. (Elaboración propia) .....	55
Figura 23: Pantalla principal. Menú QR. (Elaboración propia).....	56
Figura 24: Pantalla principal. Menú Cerrar Sesión. (Elaboración propia) .....	57

Figura 25: Simulador con cámara (Elaboración propia) .....	58
Figura 26: Simulación de panel CNC en aplicación (Elaboración propia) (Passmore, 2020) .....	59
Figura 27: Problemas de otras librerías (Elaboración propia).....	60
Figura 28: Cuadro de Mando Integral (Elaboración propia).....	64
Figura 29: Indicador con datos más recientes (Elaboración propia) .....	65
Figura 30: Gastos en formación histórico (Elaboración propia) .....	67
Figura 31: Productividad histórico (Elaboración propia) .....	67
Figura 32: Tasa de Accidentes histórico (Elaboración propia).....	68
Figura 33: Tasa de Gravedad de Accidentes histórico (Elaboración propia) .....	68
Figura 34: Tasa de Errores de la Plataforma histórico (Elaboración propia).....	69
Figura 35: Tasa de Concurrencia de Usuarios histórico (Elaboración propia) .....	69
Figura 36: Latencia histórico (Elaboración propia) .....	69

## Índice de tablas

Tabla 1. Ventajas y desventajas de desarrollo para Android y iOS .....	37
--	----



# 1. Introducción

## 1.1. Motivación

¿Es realmente necesario que los empleados tengan una buena formación? ¿Son suficientes algunas nociones básicas para realizar correctamente su trabajo o se debería invertir más en formación? ¿Los nuevos empleados disponen de suficiente conocimiento con las formaciones que se le imparten? La lista de preguntas podría alargarse de forma innecesaria, pero la cuestión es que la formación de los empleados es una de las claves del buen funcionamiento de una empresa. Los empleados eran, son y serán el corazón de la empresa, la clave de la misma, su formación, sus conocimientos y competencias son los que harán que una empresa crezca, se mantenga o desaparezca. Con el objetivo de facilitar, o complementar, la formación que se imparte en las empresas del sector industrial, se ha planteado la idea que da origen a este trabajo de fin de máster, implementar el uso de una de las tecnologías de la industria 4.0, la Realidad Aumentada, buscando así ofrecer un mayor rendimiento en la formación de los empleados.

En el contexto actual, en el año 2022, hay tecnología con la que se convive con normalidad, que forman parte del pan de cada día de la mayoría de las personas. Esta tecnología son teléfonos móviles, tablets, ordenadores, y una amplia variedad de dispositivos inteligentes. ¿Entonces, por qué no incorporarlos dentro de la formación del personal de una empresa?

Con el objetivo de conseguir una formación más eficiente, más rápida e incluso más completa, se plantea incorporar una tecnología puntera, como es la Realidad Aumentada, a la forma en la que se imparte el conocimiento a los nuevos empleados en el entorno industrial.

Expuesto lo anterior se plantea la siguiente pregunta. ¿Se utiliza ya la Realidad Aumentada dentro del entorno industrial? Tanto en fabricación como en uso de algunos dispositivos la respuesta a la pregunta es sí, la Realidad Aumentada hace décadas que existe, y desde el primer momento se incorporó en múltiples sectores industriales, entre ellos el desarrollo de material para uso militar, y el sector aeronáutico. La realidad aumentada, surgió para resolver el problema que tenían los operarios de una compañía en el momento de cablear los tableros de las cabinas de los aviones. A principios de los años 90, se desarrolló un sistema experimental para ayudar a los trabajadores de la empresa Boeing a ensamblar los cables para las aeronaves, pero a día de hoy se puede extrapolar esta tecnología a prácticamente cualquier ámbito, sea o no industrial.

## 1.2. Planteamiento del trabajo

Con la finalidad de poder implementar el uso de la Realidad Aumentada en la formación, que en este caso concreto es de los futuros empleados de línea de producción y montaje, se propone, en el presente trabajo de fin de máster, el realizar el diseño de un ecosistema implementado para el usuario final como una aplicación móvil la cual facilite el uso de la distinta maquinaria que el empleado se pueda encontrar en la línea de producción y montaje. Con esto se pretende conseguir una formación más veloz y efectiva, consiguiendo que los empleados obtengan un mayor conocimiento de las distintas máquinas que deben operar en un menor tiempo y además aportando una información más completa, consiguiendo así reducir el número de accidentes laborales en esa zona de la empresa.

El objetivo principal de la formación ha ido evolucionando con los años, pero una buena forma de definirlo es adaptarse a las nuevas formas de trabajo. Hace años la formación se basaba meramente en transmitir una serie de conocimientos de un empleado a otro con menor experiencia. Actualmente, siguen quedando multitud de empleos con trabajos manuales en los que se sigue formando a los empleados de la forma tradicional, porque no se cree necesario cambiarlo. Hay empresas reticentes a realizar inversiones en formación debido a que lo visualizan como un coste y no como una ventaja. Por estas razones se pretende incorporar el uso de una aplicación en la formación, incorporando las nuevas tecnologías sobre un método que en muchos casos sigue siendo tradicional, y que, aunque no lo sea, puede ser un extra a la formación que se imparte. (Observatorio de RRHH, 2020)

## 1.3. Estructura de capítulos

Este trabajo de final de estudios se desglosará en los siguientes capítulos:

- **Capítulo II: Contexto y estado del arte**, donde se comenzará exponiendo en líneas generales el contexto del proyecto en cuestión, la costosa y tradicional formación en relación a lo que puede llegar a ser haciendo uso de tecnologías modernas. Posteriormente se expondrán otros proyectos relacionados con este trabajo, concretamente proyectos de realidad aumentada aplicados a distintas áreas. Como es un tema relativamente nuevo, los proyectos de los que se parte como referencia también lo son, siendo todos del último lustro. Finalmente, en este apartado se expondrán las distintas tecnologías que se pueden utilizar para poder incorporar la realidad aumentada al entorno laboral, y en base a las distintas tecnologías, se escogerá la más apropiada para la aplicación en cuestión.

- **Capítulo III: Descripción general de la contribución del TFM**, apartado en el cual se expondrán los objetivos de este proyecto. Para ello se marcará un objetivo principal, que es el de resolver el problema en cuestión, abordar la formación tradicional que actualmente se tiene y modernizarla para obtener mejores resultados en menos tiempo y haciendo uso de menos recursos. Para lograr este objetivo general, se plantearán objetivos específicos, concretos y alcanzables, para, tal y como se ha expuesto, lograr alcanzar ese objetivo general. Una vez aclarados los objetivos en cuestión, se expondrá la metodología de trabajo, es decir, qué pasos se van a seguir para conseguir dichos objetivos, qué tecnología o tecnologías se utilizarán y como se planteará el desarrollo. Teniendo clara la metodología a utilizar, se pasará a describir de forma general las partes o componentes de la propuesta. En este apartado se marcará el alcance del trabajo y sus limitaciones, quién realizará el trabajo y qué departamentos pueden colaborar (en este caso es un trabajo individual). Se expondrán las tecnologías de la industria 4.0 implicadas y un esquema con la arquitectura, los componentes y la integración de las distintas tecnologías a utilizar, que funciones realizarán estas partes, etc. Finalmente se expondrán los resultados esperados y los obtenidos, para comprobar si realmente difieren y si ha aparecido alguna dificultad en el momento final. Obtenidos los resultados solo quedará exponer el presupuesto de la aplicación en cuestión y el retorno esperado de la inversión, o la amortización, y en caso de llevarlo a la práctica en una empresa real, una aproximación de las distintas fases para su integración en la empresa y los tiempos necesarios, en resumen, una pequeña planificación.
- **Capítulo IV: Desarrollo específico de la contribución**. Dejando ya con esto finalizada la parte teórica, se expone el desarrollo práctico de la aplicación, un sistema de realidad aumentada para entrenamiento de operarios de producción y montaje. En este apartado, se expondrá el desarrollo específico paso por paso hasta el resultado final, para que queda constancia de cómo se ha realizado todo el desarrollo.
- **Capítulo V: Conclusiones y trabajos futuros**. Finalmente, se expondrán las distintas conclusiones para informar del alcance y la relevancia final del trabajo realizado. En resumen, se expondrán las contribuciones principales del TFM, exponiendo cuál es el problema a resolver, como se ha planteado y cómo ha quedado resuelto.

## 2. Contexto y estado del arte

En este apartado se realizará una breve introducción en el campo de la realidad aumentada, también conocida como RA, para poder comprender en qué consiste y cómo puede utilizarse.

### 2.1. Descripción general del contexto del proyecto

Para poder plantear cómo se va a desarrollar el trabajo en cuestión, previamente se debe conocer el entorno en el que se encuentra, las distintas tecnologías y procesos que se van a utilizar para a posteriori realizar el desarrollo del trabajo.

Actualmente, el entorno industrial español sigue unas líneas completamente tradicionales. El uso de tecnologías punteras, como pueden ser las tecnologías de la Industria 4.0, es algo que no todas las empresas contemplan, debido al gasto en servicios e infraestructuras que conlleva este tipo de tecnología. Tecnologías como la realidad aumentada, el Big Data y el análisis de estos datos, el almacenamiento y servicios en la nube, la inteligencia artificial, el uso de IIoT (Industrial Internet of Things), y otras tecnologías similares en el entorno industrial actual es algo que pocas empresas realizan pero que a medida que pasa el tiempo las empresas van incorporando.

En este trabajo en particular, no se va a desarrollar el trabajo enfocado en ninguna empresa en particular, sino que se desarrollará enfocado a un sector, o a un formato de empresa muy común. Este formato de empresa es más bien tradicional, sin prácticamente uso de la tecnología, una empresa que ha tenido que incorporar tecnología de forma forzada debido a la pandemia, dado que han tenido la necesidad de adaptarse antes de caer fuera del mercado. Es por esto que se plantea este proyecto en el contexto de este tipo de empresas, empresas en las que la tecnología más compleja que están utilizando actualmente es el programa Microsoft Excel para el día a día, cuando incorporando tecnologías de la Industria 4.0, podrían realizar las mismas tareas, pero con un gasto menor de tiempo y recursos. (ControlSYS, 2022)

En España actualmente se invierte, de media, menos de la mitad que el resto de países cuando se trata de formación de los empleados, e incluso menos que una tercera parte de lo que se invierte en Estados Unidos. El problema actual reside principalmente en la pequeña y mediana empresa, que es la más abundante en el panorama español, la cual invierte de media una cantidad menor a 350€ por empleado, en comparación con las grandes empresas españolas, que invierten una media de 750€ por empleado anualmente en formación. (Fundación Élogos, 2008)

Este trabajo se va a centrar en proponer una alternativa que pueda ser viable, para implementar en empresas con línea de producción y montaje, para la formación de operarios para esta línea, y para que esta formación sea lo más completa posible y en el menor tiempo posible.

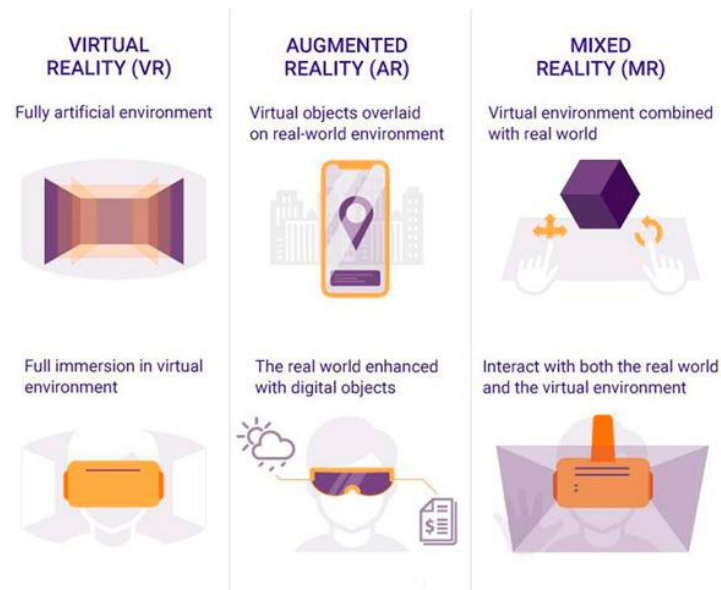
## 2.2. Proyectos relacionados con el tema del TFM

Este TFM tiene por antecedentes distintos trabajos, artículos y libros relacionados con el tema en cuestión, la incorporación de la realidad aumentada para mejorar un campo determinado dentro de un sector. Los proyectos relacionados con el tema principal del trabajo que se han tomado como referencia son los siguientes:

### 2.2.1. Realidad virtual y realidad aumentada: desarrollo de aplicaciones.

Este libro recoge los resultados de un trabajo de investigación sobre el desarrollo de una aplicación de realidad virtual y realidad aumentada básica.

Primeramente, comienza recorriendo los orígenes y la evolución tanto de la Realidad Virtual como de la Realidad Aumentada (RA). Luego, se presentan ambas tecnologías dentro del desarrollo de aplicaciones. Realiza una exposición de la realidad virtual, la cual define como un entorno que puede ser de apariencia real o no, que da la sensación al usuario de estar inmerso en él. La realidad aumentada, realidad en la que no se está inmerso, sino que el dispositivo, ya sea un teléfono móvil, un Tablet o cualquier otro dispositivo con cámara, hará que veamos el entorno real a través de dicha cámara, y sobre este entorno mostrará elementos inexistentes que formarán esa parte virtual, por lo que una definición apropiada sería una combinación visual de elementos reales y virtuales que interaccionan. La realidad mixta, otra de las realidades que menciona, esta es más compleja de definir, ya que comprende la parte de inmersión total de la realidad virtual y además la realidad aumentada, combinando elementos virtuales con el mundo real.



*Figura 1: Diferencias entre Realidad Virtual, Realidad Aumentada y Realidad Mixta (Cámara de Valencia, 2021)*

El libro también expone los distintos habilitadores de la Industria 4.0 que utiliza. Primeramente, el IoT (o Internet of Things) junto con las Tecnologías de visión por ordenador. Ambas tecnologías forman la base de las dos aplicaciones que se desarrollan en este libro, utilizándolas para poder dotar de una visión artificial al equipo, que en este caso es un dispositivo móvil. También utiliza el Cloud Computing y la Ciberseguridad en el desarrollo de la aplicación para poder acceder de forma segura y ágil directamente a información almacenada en la nube. Adicionalmente, explica otros habilitadores de la Industria 4.0 pero los anteriores son lo que utilizan para el desarrollo en cuestión.

Posteriormente expone un conjunto de ejemplos prácticos de las empresas más destacadas dentro del sector de desarrollo de aplicaciones las cuales aplican la realidad virtual, aumentada o mixta, junto con distintas tecnologías de la Industria 4.0 para desarrollar sus aplicaciones. Entre estos ejemplos se encuentran aplicaciones de Realidad Virtual totalmente inmersivas como “Realidad Virtual para análisis de conducta humana en situaciones de emergencia” de la empresa 6D Lab. También exponen ejemplos de aplicaciones de Realidad Aumentada como el juego educativo “Enigma Galdiano” de PadaOne Games o “Museo Soroll” de ARS VIVA, o un ejemplo que está más relacionado con este trabajo como es la implantación de las herramientas “Space 1” de la compañía “OverIT” realizadas por EPTISA para el mantenimiento de líneas de producción, o inspección en remoto.

A sabiendas de todo lo anterior, comienzan el desarrollo del trabajo de investigación en cuestión, la creación de una aplicación de realidad virtual y realidad aumentada básica. El objetivo que plantean es llegar a conseguir generar dos aplicaciones para Android para poder probar en el teléfono que efectivamente funcionan.

Comienzan creando una app de Realidad Aumentada muy sencilla, para ello escogen el SDK de Vuforia integrada en Unity 3D. Vuforia es una plataforma de desarrollo de software que permite la creación de aplicaciones de realidad aumentada de una forma simple. Vuforia permite crear aplicaciones tanto para Android como para iOS e incluso para Windows. A lo largo de este apartado exponen como configurar todos los elementos necesarios para poder desarrollar la app de RA simple, que va a operar sobre una imagen como entrada.

Posteriormente exponen la segunda aplicación, una de Realidad Virtual básica. En este caso utilizan el SDK de Google Card Board para Unity 3D. Este lo escogen por la sencillez de uso y configuración, porque está integrado en Unity 3D y porque está desvinculado a hardware específico por lo que se puede utilizar en un smartphone cualquiera.

Para terminar, en el libro se exponen varios capítulos, cada uno dedicado en exclusiva a Vuforia, Arcore, y Microsoft Windows Mixed, todos desde su configuración hasta su uso. (Navarro et al., 2019)

### **2.2.2. Diseño de la interfaz de una aplicación móvil para realizar visitas turísticas con realidad aumentada (RA) en Teruel.**

En este caso, este trabajo es un trabajo de final de máster, del Máster Universitario en Diseño Gráfico Digital de la UNIR. Este trabajo lo que expone es que se podría mejorar la oferta turística actual en la zona de Teruel utilizando tecnologías más punteras como la RA, diversos medios audiovisuales y dispositivos móviles inteligentes (smartphones). El objetivo principal de este TFM es diseñar una UI para móvil que permita a los usuarios viajar por los lugares históricos y los monumentos más característicos utilizando la RA. Con esto se busca difundir y acercar la cultura a una mayor parte de la población, principalmente a las personas que no pueden desplazarse hasta allí.

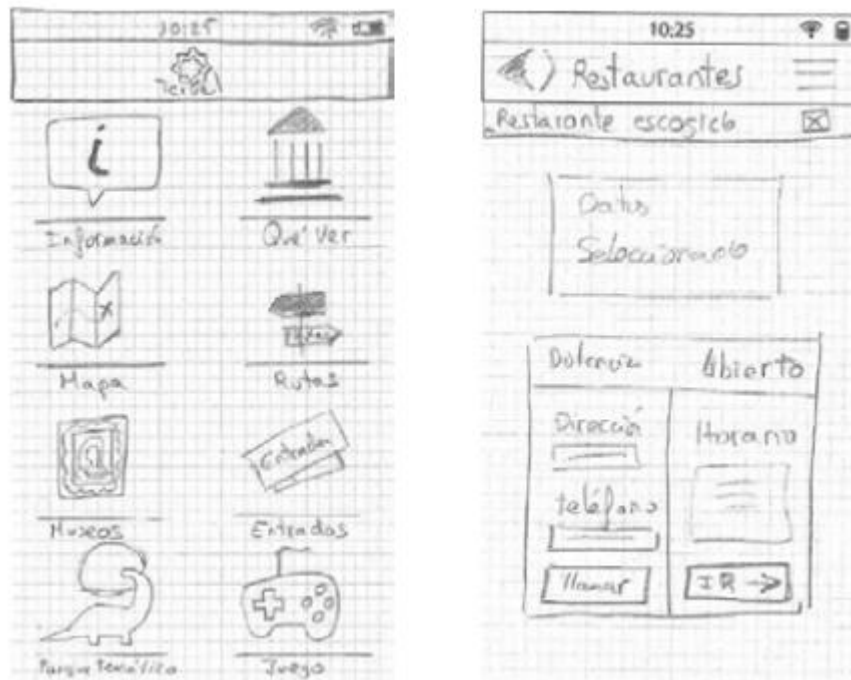
En el trabajo se hace una primera introducción exponiendo la importancia del turismo a nivel nacional y concentrado en esa zona. El objetivo general de este trabajo, tal y como se ha comentado inicialmente, es diseñar la interfaz de una aplicación móvil que permita a los usuarios que visiten una ciudad, poder hacerlo fácilmente y cubriendo todas sus necesidades de información, acercándole y cubriendo todas las necesidades que un usuario pueda tener en las visitas turísticas.

Hecho esto se detalla la metodología de trabajo que se va a seguir, siendo esta una exhaustiva investigación previa tanto del mercado actual y lo que ofrece en las visitas turísticas, así como las aplicaciones con mayor probabilidad de ser encontradas, o que posean alguna característica que las distinga de las demás. Además, se investiga sobre aplicaciones para crear objetos en 3D, y para crear RA. A todo esto, se le suma el hacer encuestas a usuarios de distintos tipos e identificar sus requisitos y necesidades.

Una vez completada la investigación, se establece el objetivo general, el cual es diseñar la interfaz de la aplicación móvil para que los usuarios puedan visitar la ciudad de Teruel y su cultura e historia. Para cumplir el objetivo principal, en el desarrollo se busca que la aplicación sea capaz de adaptarse a las necesidades del usuario, recogiendo y pudiendo ofrecer toda la información que el usuario pueda requerir en una visita turística. Además, pueda ofrecer información personalizada al usuario, y permitir que este se desplace de un punto a otro sin tener la necesidad de estar físicamente ahí.

Sabiendo todos los objetivos, se desarrolla el diseño, se asientan las bases de la interfaz y se comienza una fase vital, el prototipado. Esta fase se subdivide en varios apartados, siendo el primero de todos, la creación de los prototipos de baja fidelidad, es decir se parte de un primer diseño realizado a lápiz y papel. Esta técnica tiene múltiples ventajas a la hora de realizar desarrollo de aplicaciones, como puede ser el bajo coste y la rápida interacción para realizar cambios.





*Figura 2: Prototipo de baja fidelidad (Martínez Olivar, 2020)*

Hecho el primer prototipo pasa a la segunda parte, la creación de prototipos de alta fidelidad, es decir, realizar un diseño visual ahora para ser visualizado en la pantalla de la aplicación, creando y añadiendo colores, imágenes, tipografía, etc. Esta parte tiene mucha importancia porque es donde se definen las distintas pantallas que tendrá la aplicación, cómo serán estas, como se cambiará de una a la otra. En definitiva, es la parte que requiere más trabajo y que establecerá las bases de la aplicación final.

Una vez hecho todo lo anterior, se procede a diseñar los iconos de la aplicación, los distintos botones y pantallas, como la de bienvenida y la de despedida. Seguidamente se realiza una gamificación, también conocida como ludificación, que consiste en añadirle a la aplicación algunas características que llamen la atención de los usuarios. Estas pueden ser, utilizar beneficios, logros, algún tipo de competición, juegos en equipo, etc.

La parte final del prototipado es la del diseño 3D y RA. En esta parte se diseñan todos los elementos necesarios en 3 dimensiones, es decir, todos aquellos que se van a exponer, como pueden ser monumentos o estatuas. Este diseño se realiza asistido con distintos programas para la creación de elementos 3D, y una vez terminados, se utilizan otros programas para la implementación de los diseños en la RA y a su vez en la aplicación.



*Figura 3: Primer prototipado desarrollado (Martínez Olivar, 2020)*

Con esto ya está terminado el prototipo, y la aplicación ya es funcional. En este punto ya solo falta la parte final, la evaluación. Esta evaluación ha sido realizada por distintos usuarios, que posteriormente rellenan una tabla para opinar sobre las distintas partes de la aplicación. Posteriormente, con el recogido de estas encuestas a los usuarios, se ha realizado unas conclusiones y unas modificaciones para ajustarlo a los requisitos de los usuarios. (Martínez Olivar, 2020)

Se han incluido otros artículos, trabajos y referencias que también se utilizan como base. Estos son más escuetos y están más enfocados a la educación, pero se ha considerado que son igualmente extrapolables debido a las tecnologías que se han utilizado.

### **2.2.3. La realidad aumentada en la formación del profesorado. Una experiencia en las prácticas del Máster de Profesorado de Enseñanza Secundaria.**

Este artículo tiene por objeto general diseñar una modalidad de presentación de la memoria de prácticas para simplificar y a su vez mejorar el proceso de evaluación de una asignatura concreta, que en este caso es Prácticum en la educación superior. Para lograr este objetivo, se desglosa en los siguientes subobjetivos, incorporar la RA como recurso TIC básico para

ampliar información y simplificar el proceso de evaluación didáctica. El siguiente, y último, subobjetivo es conocer y manejar RStudio a nivel usuario como medio o recurso para la resolución de tareas académicas o profesionales.

Se realiza el diseño sobre la plataforma RA Layar Creator, así como su aplicación de visualización. El desarrollo en sí será evaluado realizando una prueba sobre una clase de grado o máster, sin importar el número de alumnos que esta tenga.

Se realiza una fase de implementación, donde todo el trabajo radica en dos seminarios, en los que el profesor aprovecha para explicar a su alumnado todo lo que deben saber sobre las plataformas que se van a emplear, RStudio y RA Layar Creator.

Los resultados esperados de la propuesta, es que este formato de entrega de documentación supere al modelo de Prácticum tradicional, el cual está basado en la entrega de una memoria impresa con infinidad de páginas de texto e imágenes. La propuesta de uso de las TIC y de la RA además tiene una ventaja que es poder utilizar material en formato de clips vídeo que pueden incluirse dentro del trabajo que se tenga que entregar.

Vistos los resultados esperados, se exponen los resultados reales. Utilizando las herramientas proporcionadas se ha concluido que este formato de entrega didáctica, utilizando herramientas TIC y de RA, es sumamente superior al formato tradicional. En este caso los alumnos son completamente autónomos por lo que esto afecta a su grado de motivación por aprender. Eso sí, cabe remarcar la desventaja principal que se ha encontrado, que es conseguir que el profesorado actual se adapte al cambio y en todas sus modalidades y entren en una era de máxima digitalización, ya que la formación es una de las ramas que sustenta el desarrollo personal y la formación de nuevos profesionales. (López-García et al., 2018)

#### **2.2.4. Blended learning y realidad aumentada: experiencias de diseño docente**

En este artículo se expone y analiza el auge del uso de dos tecnologías emergentes como son el blended learning, o aprendizaje semipresencial, y la realidad aumentada. El artículo se basa en una exposición de ambos conceptos, teniendo en cuenta que el blended learning realmente es el aprendizaje semipresencial, por lo que hay una parte de este aprendizaje que se realiza haciendo uso de las distintas tecnologías. Lo que pasa actualmente, es que tanto

el profesorado como el estudiantado se ha acostumbrado a hacer uso solo de las tecnologías más simples como puede ser la Web 2.0, pero este artículo, nos hace recordar que se podría implementar hasta RA en la educación semipresencial y se conseguiría fomentar la motivación de los alumnos, y mejorar los resultados.

En el artículo se exponen ejemplos reales como el proyecto RAFODIUN, un proyecto en el que se da uso a estas tecnologías en alumnos de Medicina, pudiendo presentar los distintos órganos y huesos, sobre los cuales se pueden realizar rotaciones, desplazamientos, cambiar las perspectivas, etc. La evaluación del uso de estos objetos dentro de la RA se llevó a cabo en un estudio piloto con alumnos que cursaban Anatomía Humana I. La evaluación se llevó a cabo exponiendo a los alumnos primeramente en qué consistía la RA, posteriormente una demostración de los diferentes objetos producidos, una presentación del lugar en el que podían descargarse las guías e información necesaria, el trabajo individual de cada alumno con su dispositivo móvil y un cuestionario final para evaluar los objetos que se presentaron. Los resultados fueron sorprendentemente positivos, dado que esta tecnología tuvo un alto grado de aceptación.



*Figura 4: Objetos en Realidad Aumentada para la formación en medicina (Cabero Almenara et al., 2018)*

La conclusión final del artículo es que el uso de objetos de RA en acciones de b-learning puede llegar a ser de mucha utilidad para realizar por ejemplo actividades prácticas y ejercitar

sobre contenidos presentados. Además, supone una ventaja en el caso de que haya que realizar prácticas más difíciles de lo normal por ejemplo debido al coste de la instrumentación o de las instalaciones necesarias, ya sea por espacio o tiempo. (Cabero Almenara et al., 2018)

#### **2.2.5. La innovación en el aula universitaria a través de la realidad aumentada. Análisis desde la perspectiva del estudiantado español y latinoamericano.**

Este artículo tiene por objeto analizar las experiencias de innovación universitaria dando pie al uso de tecnologías inmersivas como la realidad aumentada. Para ello, se combina una metodología de análisis de datos con el diseño de un cuestionario ad hoc, para conocer la opinión sobre las ventajas y desventajas de la RA y las aplicaciones que utilizan RA, utilizadas únicamente en el entorno de los seminarios de Tecnologías Emergentes en contextos educativos.

Se expone el uso de RA en el ámbito educativo, un ámbito en el que la incorporación de esta tecnología ha ofrecido nuevos retos y nuevas formas de afrontarlo. Su aplicabilidad en contextos de formación educativa ha resultado más beneficiosa que la metodología tradicional, la cual está completamente sujeta a la transmisión de contenidos, dejando a un lado la comprensión real de los mismos. Esto ha favorecido un cambio didáctico consiguiendo procesos de aprendizaje más dinámicos y significativos, aumentando también el nivel de atención que prestan los alumnos.

La integración de esta tecnología, se ha realizado de forma progresiva, desarrollándola a través de distintos seminarios didácticos de formación en RA, en exclusiva para alumnos del Grado de Educación Social y del Doble Grado de Educación Social y Trabajo Social de la Facultad de Ciencias Sociales de la Universidad Pablo de Olavide, en Sevilla. Una vez impartidos los correspondientes seminarios se procede a analizar los resultados de forma cuantitativa, cualitativa y descriptiva.

Los resultados obtenidos fueron que la aplicación más motivadora, de las cuatro impartidas y utilizadas en los seminarios fue QuiverVision. Esta también fue la más utilizada por los alumnos del seminario. Teniendo esto claro se presentan las ventajas y desventajas del uso de RA en el ámbito educativo universitario. Las ventajas principales son la mejora de las habilidades cognitivas y competenciales, agrupadas como estimulación, concentración, simulación, reflexión y perspectiva. También se valoran como ventajas la interactividad, la

practicidad y dinamicidad. De igual manera nos exponen las siguientes desventajas, la brecha digital que supone y la necesidad de formación del profesorado. El estudiantado ha percibido que el uso de este tipo de tecnologías emergentes como es la RA, posibilita en si una nueva forma de aprender y desarrollar competencias pero que su viabilidad y adopción va a estar condicionada por la necesaria formación del profesorado que, en la mayoría de ocasiones, no se somete a procesos de formativos para favorecer la innovación didáctica.

## 2.3. Tecnologías relacionadas con el tema del TFM

En este apartado se van a exponer las tecnologías habilitadoras de la Industria 4.0 que se van a emplear en el desarrollo del trabajo.

### 2.3.1. Realidad aumentada (RA)

La tecnología principal en la que se centrará el desarrollo de este trabajo, es la Realidad Aumentada (RA), la cual esta considera como una tecnología emergente e innovadora. El concepto de lo que es la RA es muy sencillo, se trata de añadir una capa de virtualidad a la realidad, mediante un dispositivo. Existen multitud de dispositivos entre los que se pueden encontrar aplicaciones de RA, estos pueden ser desde un teléfono móvil o un ordenador hasta gafas como las Google Glass, de Google, o las gafas Hololens de Microsoft entre otros.

Los orígenes de esta tecnología son muy difusos, desde inicio la década de 1960, múltiples aeronaves ya disponían de pantallas que superponían gráficas para ayudar a pilotarlas, eso sí, únicamente se le daba uso en el campo de la tecnología militar. No fue hasta finales de ese siglo, hacia 1990, cuando dos ingenieros de la compañía aeronáutica Boeing diseñaron y crearon un prototipo que consistía en unas gafas transparentes que combinaban la detección de elementos del mundo real y el posicionamiento de la cabeza para proyectar imágenes. Gracias a esta tecnología, los usuarios de las gafas, que en este caso eran los operarios de la empresa, eran capaces de superponer un diagrama del cableado en una posición específica para poder cablear los tableros de instrumentos de los aviones en el mundo real. Cabe remarcar también que el término, Realidad Aumentada, no fue acuñado hasta dos años después, por los mismos ingenieros que había hecho el desarrollo inicial.



*Figura 5: David Mizell con el primer prototipo de realidad aumentada de Boeing  
(Schmalstieg et al., 2016)*

Desde entonces se inició una nueva era en la que multitud de empresas comenzaron a investigar sobre distintos dispositivos para implementar esta tecnología. Esto lo podemos ver reflejado con los primeros ejemplos:

- En 1992, se desarrolló el primer sistema de inmersión total, por lo que casi podría considerarse Realidad Virtual. Este era un exoesqueleto kinestésico desarrollado por Virtual Fixtures.
- En 1994, la compañía Azuma desarrolló el primer equipo de estabilización de movimiento para seguimiento de objetivos, pensado para uso en exteriores.
- En 1996, un equipo del MIT, conocido como The MIT Wearable Computing Group, comenzó a experimentar con gafas inteligentes y dispositivos similares a estos. (Hensen et al., 2020)

Actualmente, podemos encontrar grandes compañías que siguen desarrollando y mejorando dispositivos en los que se incorpora la RA. Entre estas compañías se encuentran Google y Microsoft, ambas cuales tienen un producto de gafas inteligentes de última generación que incorpora el uso de RA:

- Las Google Glass, de Google, fueron el primer dispositivo disponible para su compra para cualquier público. Este fue lanzado en 2012 en exclusiva para desarrolladores, pero actualmente se encuentran disponibles para todo el público. El objetivo principal

era conseguir un dispositivo que pudiera mostrar información para los usuarios que estén habituados a los teléfonos inteligentes, pero con la característica de no tener que utilizar las manos. Este dispositivo posee acceso a internet e incluso comandos por voz, dado que está basado en un software Android.



*Figura 6: Google Glass Enterprise Edition 2 (Google LLC, 2022)*

- Las Microsoft HoloLens, de Microsoft, son la competencia directa de las Google Glass, con la diferencia de que las primeras utilizan Windows 10 como sistema operativo y las segundas tienen un software basado en Android. Se diseñaron a la par que Windows Holographic, la plataforma de realidad tanto virtual como aumentada propiedad de Microsoft, y aunque actualmente parece que Microsoft ha cancelado la salida al mercado de la 3ª generación de este modelo, sigue siendo un dispositivo a tener en cuenta.



*Figura 7: Microsoft HoloLens 2 (Microsoft Corporation, 2022)*

- Existen otras grandes compañías que tienen dispositivos similares como las Smart Glasses de Xiaomi, las Focals de North, o las Echo Frames de Amazon podrían ser



algunos de los ejemplos más actuales. Eso sí, la compañía originaria no se queda atrás, Boeing sigue utilizando la RA para el cableado de los tableros de los aviones, siempre y cuando sea necesario, solo que ahora lo realizan incorporando esta tecnología en unas Google Glass, gracias a los avances de los últimos 30 años. (Navarro et al., 2019)

### **2.3.2. Industrial Internet of Things (IIoT)**

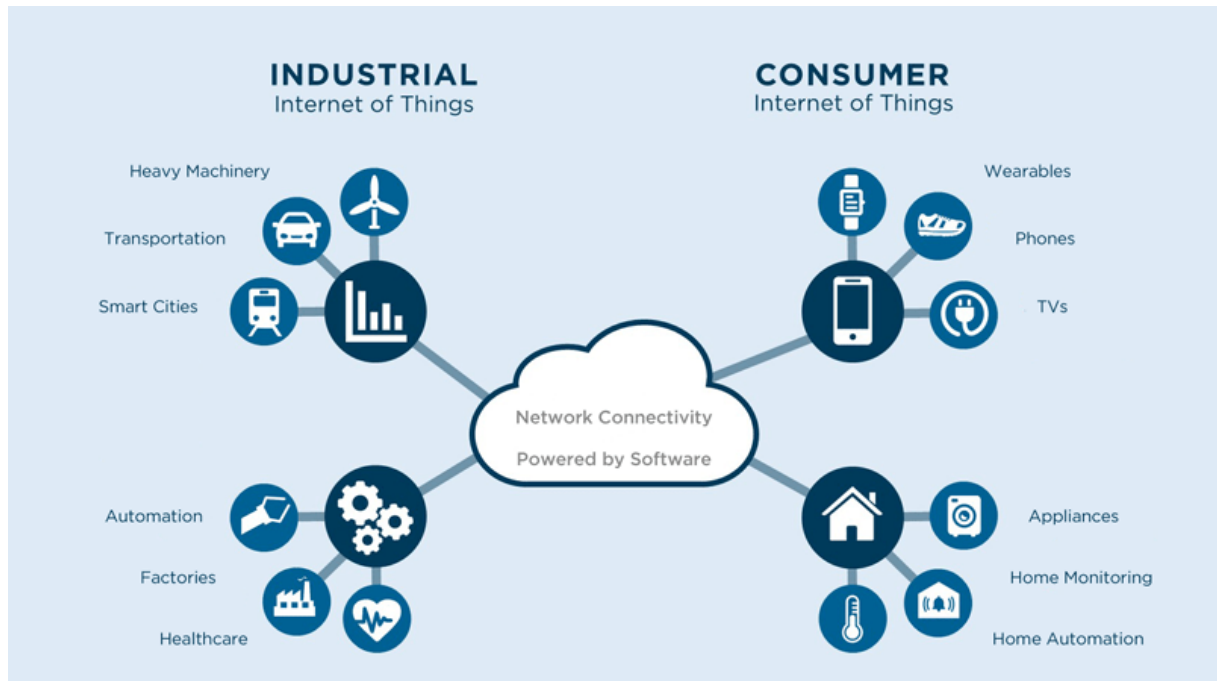
Industrial Internet of Things, también conocido como IIoT o Internet Industrial de las Cosas. Otra tecnología emergente, una de las tecnologías de la Industria 4.0, que como otras ha aportado grandes ventajas al sector industrial. Gracias a esta tecnología se añade un nivel más de inteligencia sobre las máquinas, pudiendo mejorar su control y monitorización.

Esta tecnología consiste en añadir una amplia variedad de sensores de nueva generación, los conocidos como sensores inteligentes, consiguiendo un gran nivel de conectividad, monitorización y control. (CSA GROUP, 2017) Además, es una tecnología basada en la nube, cuyo origen se encuentra en el IoT (el Internet de las Cosas), el cual fue derivando al uso industrial. El IoT surgió a finales del siglo pasado, cuando investigadores del MIT estaban centrados en conseguir información sobre la identificación por radiofrecuencia en red, también conocida como RFID, y distintas tecnologías de sensorización.

Este origen abrió un número casi ilimitado de escenarios en los que se podrían utilizar, siempre y cuando contenga elementos inteligentes y haya que transmitir una cantidad considerablemente grande de información. El IIoT tiene por labor recopilar datos, a partir de sensores y dispositivos inteligentes, consiguiendo una interconexión entre los mismos. Con los datos que recopila brinda la suficiente información para modificar patrones o configuraciones a tiempo real sin paradas de producción ni pérdidas de tiempo innecesarias en todo el sector de manufacturación. (Haghnegahdar et al., 2022)

El IoT ha tenido un crecimiento muy rápido y reciente, ya sea dentro o fuera del mundo industrial, y ha contribuido drásticamente a mejorar la calidad de vida. La diferencia entre el IoT y el IIoT radica meramente en el campo en el que se aplica. El IoT busca mejorar la calidad de vida del consumidor, de la persona de a pie, aportando un nuevo nivel de conectividad entre los distintos dispositivos de los hogares, y del día a día. Por otro lado, el IIoT busca algo similar pero ya enfocado al mundo industrial. Este busca conseguir un mayor nivel de conectividad

entre sensores, máquinas y los sistemas que controlan y automatizan todos los procesos, mejorando así multitud de procesos. (Bin Zikria et al., 2019)



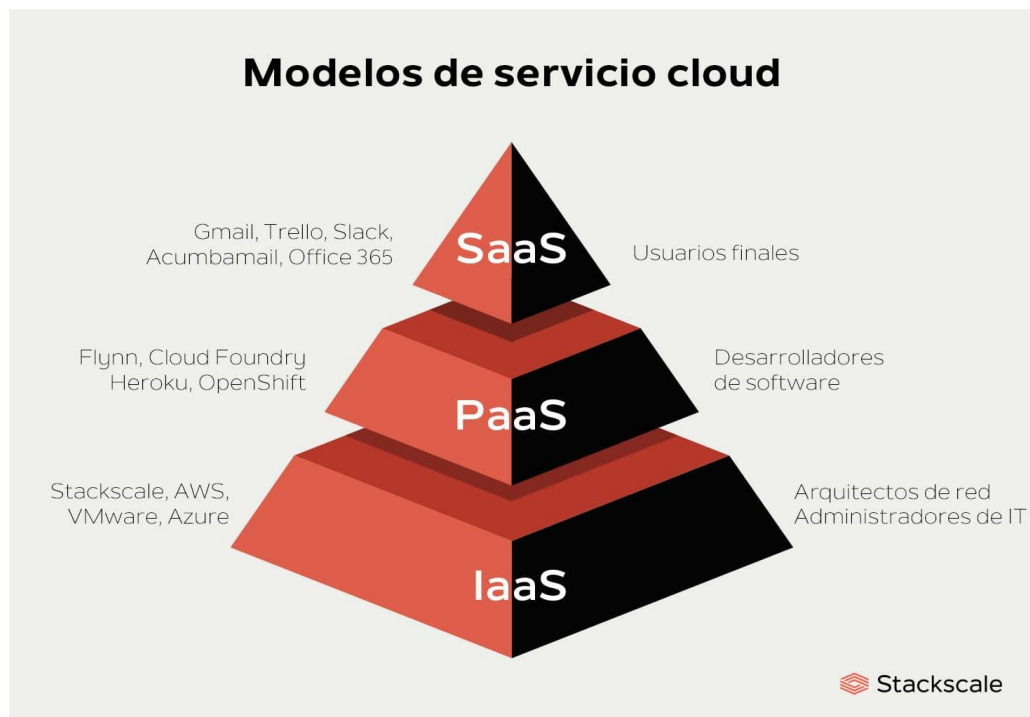
*Figura 8: Diferencia entre IIoT y IoT (Bin Zikria et al., 2019)*

Es por las razones anteriores que se incorporará en este trabajo, debido a que es una tecnología considerada necesaria para crear la comunicación entre el dispositivo que posea la Realidad Aumentada junto con la máquina sobre la que se virtualice información para ayudar en la formación de empleados. Esta podría ser utilizando lectores RFID o QR o similares, para identificación de lo que está leyendo, o lo que tiene que mostrar el sistema de RA.

### **2.3.3. Cloud Computing**

Otra tecnología a utilizar es el Cloud Computing, también conocido como computación en la nube o almacenamiento en la nube. Esta tecnología surgió originalmente para conseguir que dos o más personas utilizaran el mismo equipo de forma simultánea. Una vez conseguido llevar a la práctica la idea anterior surgió otra, la idea de que la información podría ser procesada de forma más eficiente si se centralizaba en grandes granjas de ordenadores y se almacenaba todo en un punto haciendo que fuera accesible vía Internet. Los avances en redes y otras áreas de tecnologías de información similares son los responsables de conseguir que, desde el inicio del siglo XXI, hacia el año 2004-2005, se creas lo que hoy en día conocemos como Cloud Computing.

Esta tecnología surgió inicialmente bajo el nombre de Utility Computing, tecnología donde el hardware y el software se encuentran concentrados en grandes centros aislados. En esta los usuarios pagan mientras consumen recursos, servicios y/o almacenamiento. A posteriori, recibió el nombre de Cloud Computing, popularizándose cuando se enfocó esta hacia grandes compañías tecnológicas, como Amazon, Apple, Google, IBM, etc. Actualmente se conoce como Cloud Computing, y este se puede subdividir en tres modelos de servicios cloud: SaaS (Software as a Service), PaaS (Platform as a Service) e IaaS (Infrastructure as a Service).



*Figura 9: Modelos de servicios cloud (Stackscale, 2021)*

La distinción entre los tres modelos de servicios cloud viene determinada por el servicio que ofrecen y del usuario al que va enfocado, el SaaS ofrece Software como servicio, es decir, permite a usuarios finales utilizar un software que se encuentra en la nube. Gmail, Office365, Google Docs, son solo algunos de los ejemplos más comunes de SaaS.

Tras el SaaS se encuentra el PaaS, el cual ofrece Plataformas como servicio. Este está más enfocado a desarrolladores y personal del mundo técnico, más que usuarios finales, y ofrece plataformas como pueden ser Microsoft Azure App Service, IBM Cloud Foundry, RedHat OpenShift o Zoho Creator, entre muchas otras.

Finalmente, se encuentra el IaaS, el cual ofrece Infraestructura como servicio. Este modelo de servicio está enfocado a arquitectos de red o administradores de IT, ofreciendo infraestructura en la nube como servicio. Los ejemplos más conocidos son, Amazon Web Services, Google Cloud, Microsoft Azure Storage o IBM Cloud Virtual Servers. (Marinescu, 2013)

Para la realización de este trabajo se incorporará el Cloud Computing como tecnología de la Industria 4.0 necesaria para la realización del mismo, dado que el IIoT necesita de conexión a internet y de mínimo un modelo SaaS para poder funcionar de forma remota y descentralizada, sin necesidad de almacenar gran cantidad de información, sea o no relevante, de forma local. Además, esto evita el coste de tener un ordenador o un servidor local, y las desventajas que estos presentan frente a los servicios Cloud.

## 2.4. Conclusiones sobre el estado del arte

De los proyectos expuestos anteriormente, y para concluir este apartado, se ha sacado en claro lo siguiente. En el momento de planificar y plantear un diseño, sea para el tipo de software que sea, o se plantee para la plataforma que se plantee, se debe realizar el diseño de la aplicación en cuestión con un hardware que no sea específico, es decir una app que este desvinculada de un hardware específico y únicamente necesite del sistema operativo para funcionar. Esto brinda una ventaja para implementar, una vez esté desarrollado, y abre mucha mayor oportunidad de mercado.

La aplicación debería de desarrollarse, siempre que fuera posible, siguiendo una metodología ágil, como podría ser Scrum. Comenzando con un primer prototipado de baja fidelidad, un diseño coloquialmente conocido como a lápiz y papel, el cual se pueda modificar todas las veces que sean necesarias sin importar, dado que el diseño del software se realizará una vez este prototipo se tenga claro. En este prototipado de baja fidelidad se debe incluir todo lo que forme la aplicación, aunque sea solo en concepto, qué botones, qué pantallas tendrá, cómo se pasará de una a otra y en caso de que tenga una animación como será, colores, tipografías, etc. En este caso como va a ser una aplicación para uso industrial y formativo, se deja más a un lado la gamificación, o ludificación, de la aplicación.

El personal debe estar completamente formado por lo que se le debe impartir formación utilizando tecnologías de última generación, dado que, según se ha expuesto en los trabajos anteriores, actualmente la brecha digital es lo suficientemente ancha como para que pueda

llegar a ser una complicación implementar esta. También se ha visto que realmente estas tecnologías, aunque inicialmente puedan parecer complejas, con la formación adecuada, cualquier persona podrá tanto impartir como recibir información de la misma, y más si puede llevarla en su teléfono y no necesita de un dispositivo extra para ello.

Para finalizar las conclusiones cabe remarcar que todos los trabajos, artículos, y demás documentos expuestos en los apartados previamente mencionados tienen una cosa en común, todos los diseños se realizan enfocándose en poder utilizarlos en un smartphone, y la mayoría de ellos lo enfocan en smartphones Android, por lo que esto también es un dato a tener en cuenta.

En este caso, el trabajo propuesto tiene algunas similitudes con los demás anteriormente mencionados, pero se distingue de estos en varias razones. Se plantea como un ecosistema que podrá funcionar en distintos sectores de la industria, por lo que, aunque el primer diseño se realiza enfocado a un sector concreto, no se realizará de forma centralizada, sino enfocándolo a poder ofrecerlo como una plataforma adaptable a cada industria.

## 3. Descripción general de la contribución del TFM

### 3.1. Objetivos

#### Objetivo general

El objetivo general de este proyecto es la creación de un ecosistema para la formación de empleados basado en la Realidad Aumentada que además sea escalable y extrapolable a otros entornos del ámbito industrial mediante una plataforma como servicio (PaaS).

#### Objetivos específicos

Para poder conseguir cumplir el objetivo general del proyecto, se debe desglosar en subobjetivos más específicos, pero para ello debemos tener presente el tipo de usuario va a hacer uso del ecosistema que se va a crear y en qué contexto lo va a utilizar. Para cumplir este objetivo general se subdivide en los siguientes objetivos específicos:

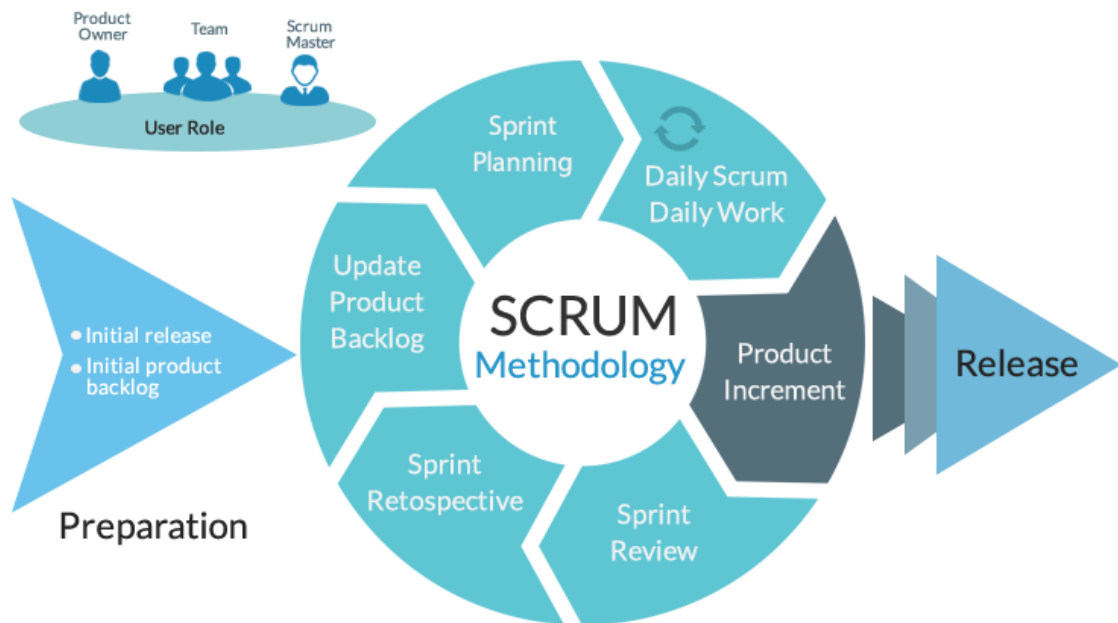
- Ofrecer la suficiente información al usuario final para que este pueda desarrollar el trabajo sin tener que consultar ningún documento ni a ningún compañero.
- Reducir costes económicos y temporales en formación para la empresa.
- Diseñar el sistema para que pueda ser utilizado como un PaaS.

### 3.2. Metodología del trabajo

En este TFM se va a seguir una metodología ágil como es Scrum. Scrum es una de las metodologías de desarrollo ágil de software más reconocida a nivel mundial y que se ha extrapolado a multitud de campos. Se originó en los años 80, y representa un marco de trabajo basado en métodos ágiles, cuyo objetivo es conocer en todo momento el estado actual del software que se está desarrollando. Habitualmente es el cliente el que establece las prioridades y el equipo que utiliza SCRUM se organiza para mejorar los resultados, pero como en este caso la metodología se aplica para el desarrollo individual de este TFM, se modificará ligeramente, manteniendo siempre las bases de esta metodología. (Estrada-Velasco et al., 2021)

Se escoge la metodología Scrum, por varias razones. La primera razón es que Scrum es una metodología en espiral, con ciclo de vida iterativo e incremental, lo que favorece la solución

de errores de manera rápida y eficaz, dado que se van dando versiones de forma periódica y resolviendo errores para dar una mejora de la versión anterior. Esta metodología destaca por la productividad, flexibilidad y competitividad que presenta y, aunque suele plantearse para desarrollos en equipo, en este caso se implementará de forma individual.



*Figura 10: Metodología SCRUM (The Drew Company, 2019)*

El uso de esta metodología se adecua a este proyecto principalmente porque esta metodología sea ágil y cíclica, pudiendo desarrollar varias versiones del mismo software resolviendo fallos del mismo, antes de realizar la entrega al usuario final.

### 3.3. Descripción general de las partes o componentes de la propuesta

El propósito general de este TFM es el desarrollo de un ecosistema escalable y extrapolable a distintas industrias, manteniendo el foco del mismo en una aplicación basada en el uso de la realidad aumentada para facilitar la formación de empleados.

### **Alcance y limitaciones**

El alcance planteado en este TFM es el diseño y desarrollo de un primer prototipo de aplicación móvil basada en la realidad aumentada enfocado al uso para operarios de producción y montaje, por lo que se plantea desarrollar la aplicación y realizar una prueba piloto sobre un caso real.

Dado el hecho de que no se dispone de ninguna industria en la que realizar la prueba piloto, se realizará una simulación a pequeña escala de forma controlada, para comprobar como de eficiente puede ser la aplicación planteada, y corroborar que cumple la función para la que va a ser desarrollada, que es ampliar y fortalecer la formación y el conocimiento de los operarios mencionados.

### **Tecnologías implicadas**

Las tecnologías implicadas en la realización del desarrollo de este TFM son las siguientes. La realidad aumentada, dado que es la base sobre la que parte todo el TFM en cuestión. Se incorporará el uso del IIoT para poder comunicar la maquinaria y sensores, en caso de que los sensores sean necesarios, en la aplicación que se va a desarrollar. Finalmente, el uso del Cloud Computing, o computación en la nube. Esta se utilizará también debido a que se quiere plantear la aplicación escalable y extrapolable, lo que podría entenderse como una PaaS, es decir, una plataforma como servicio a ofrecer a las empresas. Es necesario el uso de esta, ya sea en forma de almacenamiento y/o computación en la nube, para que la aplicación en cuestión no requiera de una gran cantidad de memoria y recursos para poder funcionar con normalidad.

### **Arquitectura, componentes e integración de tecnologías.**

La arquitectura y las tecnologías que forman el desarrollo en cuestión son las siguiente: Realidad aumentada, la cual es la parte central de trabajo. El IIoT, tecnología necesaria para conectar la realidad aumentada, y la aplicación en sí con las redes y el servidor en la nube. Cloud Computing, almacenamiento y transmisión de datos en la nube, donde se alojará toda la información y será transmitida cuando sea necesario. Por último, la ciberseguridad, para asegurar que en todo momento la información está siendo visualizada, utilizada y/o editada por el personal apropiado para ello, sin que intervenga ni pueda acceder nadie ajeno. Para este último punto se utilizará un punto de conexión a una red wifi de la empresa con usuario



y contraseña para asegurar que nadie, ni siquiera los empleados, pueden acceder fuera del trabajo.

### **Resultados esperados**

Los resultados esperados para este TFM son el desarrollo completo de la aplicación de realidad aumentada mencionada, que sea capaz de escanear una máquina que visualice mediante la cámara del dispositivo móvil. Una vez escaneada, acceda a la nube para recibir toda la información de esa máquina y sea capaz de plasmarla sobre la pantalla del teléfono cuando en esta se vea cada una de las partes de la máquina, como el panel de control o los mandos de la misma.

### **Presupuesto y retorno esperado de la inversión**

En este TFM se está planteando una solución de bajo coste y alto retorno que se medirá en función del coste de la aplicación y del ahorro que le plantee a la empresa que la adquiera en forma de horas de formación de empleados evitadas, y reducción del número de accidentes. Para ello se valora que la adquisición de la aplicación, implementada como plataforma, tiene un precio que oscila entre 13.500€ y 73.000€ con un sobrecoste opcional, de entre 500 y 75.000€ anuales, en caso de querer servicio de mantenimiento y averías hasta 24 horas al día los 365 días del año. El coste de adaptar la aplicación por completo a la industria donde se aplique viene incluido en los precios anteriores.

Expuesto esto, se espera un retorno de la inversión de menos de dos años para las empresas con más de 500 empleados y de hasta casi dos décadas para las empresas pequeñas con un número de empleados inferior a 10.

### **Planificación general**

Las principales fases del proyecto se encuentran tras el objetivo principal del proyecto, crear el ecosistema basado en la realidad aumentada. Es por esto que para alcanzar el objetivo se realizarán las siguientes fases generales, en un periodo total aproximado de 20 semanas:

- Búsqueda de información y redacción de documentación y memoria. En este apartado se encuentra la búsqueda de información y posterior redacción del marco teórico, objetivo general y demás apartados previos al desarrollo de este TFM. Se realizará en un plazo de 6 semanas.

- Desarrollo específico de este TFM, el cual se subdividirá en tres partes. La primera será el diseño inicial, elección de sistema operativo, software para desarrollar, y bocetado del software a diseñar. Para esta primera parte se estima un plazo de 5 semanas. Finalizada esta parte, se procederá con la segunda parte del desarrollo, el desarrollo de la aplicación haciendo uso del software escogido en la parte anterior. En esta se incluirá todo lo necesario hasta llegar a las pruebas de funcionamiento de la aplicación y las posibles mejoras a aplicar. Es por esto que para esta segunda parte se cuenta un plazo aproximado de 5 semanas. Finalmente, la tercera y última parte, en la que se buscará adaptar la aplicación diseñada para poder ofrecerla como PaaS en el mercado, para ello se mirará de modificar mínimamente la aplicación hasta conseguir que sea completamente adaptable y ajustable a cada empresa, industria y entorno donde se vaya a aplicar. Para esta última parte se estima un plazo de 2 semanas aproximadamente.
- Última revisión y entrega del ecosistema finalizado. El periodo para esto serán los últimos 7 días del desarrollo del proyecto, donde se asegurará un correcto funcionamiento de todas las características.

## 4. Desarrollo específico de la contribución

Para el desarrollo específico se distinguirán los siguientes subcapítulos.

- Diseño de baja fidelidad. Se realizará un primer diseño hecho a papel y lápiz. Previamente a este primer diseño se debe definir en qué sistema operativo operará la aplicación, con qué software se va a desarrollar y qué tecnologías están implicadas. A sabiendas de lo anterior se dará paso a este primer diseño, o primer prototipo, dónde se recogerá toda la información que se considere necesaria, todas las conexiones que deberá tener, y cómo y en qué parte encaja cada tecnología. Remarcar que en esta parte se realizarán todas las modificaciones que sean necesarias hasta conseguir que tanto el bocetado del prototipo como de todos los elementos contenidos en este, se adapten a lo que se busca con este TFM.
- Diseño del software. En esta parte es donde se transformará este primer prototipo realizado con papel y lápiz, a un diseño realizado con el software escogido en el subcapítulo anterior, desarrollando ya el que será el prototipo final, En este segundo capítulo se realizarán todas las pruebas y correcciones que sean necesarias para alcanzar el resultado deseado.
- Adaptación a PaaS. Finalmente se buscará complementar el apartado anterior buscando que el sistema desarrollado se pueda ofrecer como PaaS, una plataforma ofrecida como servicio. En este servicio, se buscará que la empresa que adquiera el software, pueda modificarlo e implementar la maquinaria o los equipos que ellos deseen, haciéndolo completamente adaptable a cada industria en particular.

## 4.1. DISEÑO DE BAJA FIDELIDAD

En este primer apartado se realizará el diseño de baja fidelidad. Para ello se subdividirá el apartado en los subapartados siguientes:

- Elección de sistema operativo. Donde se escogerá, de forma razonada, el sistema operativo para el que se realiza el diseño.
- Elección del software de desarrollo. Dónde se escogerá el software de desarrollo que se va a utilizar para desarrollar la aplicación en el apartado 4.2.
- Tecnologías implicadas: Diagrama. Dónde se expondrán las tecnologías implicadas y cómo interactúan entre ellas.
- Diseño de baja fidelidad. Dónde se realizará el diseño del primer prototipo del sistema, realizado a papel y lápiz.

### 4.1.1. Elección de sistema operativo

Previamente a comenzar con la etapa de diseño se deben establecer unas bases. La primera de estas es definir para qué sistema operativo se plantea este desarrollo.

En este caso, una de las características que debe tener la aplicación para poder incorporarla es que debe poder llevarse y usarse en el día a día del usuario final, es por esto que los sistemas que se utilizan en grandes equipos que se mantienen estáticos y no son portátiles quedan descartados, con lo que ya se define que los ordenadores de sobremesa o dispositivos similares no son aparatos adecuados para esta aplicación. La opción más viable, y en la que se está enfocando este desarrollo, es realizar el diseño pensando en que el software tiene que poder ser utilizado en un dispositivo pequeño, el cual debe ser portátil y disponer de cámara. Es por esto que se plantean únicamente sistemas operativos que puedan funcionar en dispositivos móviles. Estos sistemas son Android, iOS, Blackberry, Windows Phone entre otros.

Partiendo de los sistemas mencionados se debe escoger uno, por lo que se van a ir descartando por las siguientes razones. Dos de estos sistemas operativos son los que se consideran válidos, bajo el pretexto de que son los que tienen a la gran mayoría de los usuarios del mercado. Estas opciones válidas son Android, el cual es un sistema operativo

basado en Linux y demás softwares de código abierto. y iOS, el sistema operativo de la multinacional Apple Inc.

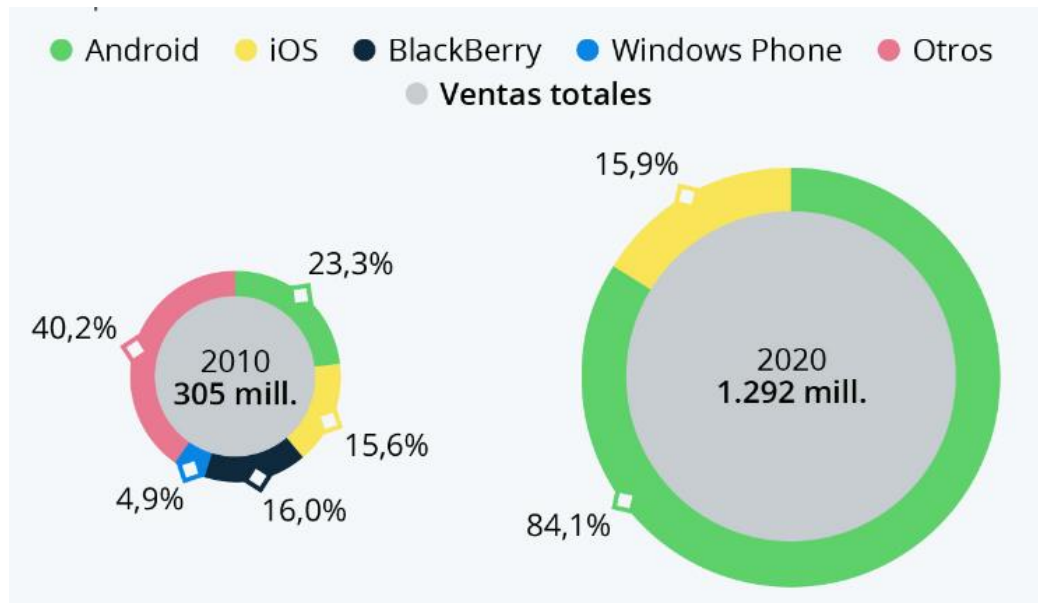


Figura 11: Android e iOS, un duopolio (Statista, 2021)

Según datos de la consultoría tecnológica IDC, se puede apreciar claramente que Android e iOS tienen actualmente lo que se conoce como duopolio, es decir, ambos sistemas operativos controlan el mercado de software para teléfonos, estando muy por encima del resto de sistemas operativos. Tal y como se aprecia en la imagen anterior, Figura 11, desde el año 2010 al año 2020, ha habido un cambio radical tanto en el número de dispositivos móviles como en el porcentaje de sistemas operativos utilizados. Es por estas razones que estos dos son los únicos sistemas que se tienen en cuenta en el momento de la elección del mismo. A continuación, se exponen las ventajas y desventajas de realizar la aplicación en Android o en iOS.

Tabla 1. Ventajas y desventajas de desarrollo para Android y iOS

iOS	Android
PRO	
Generalmente es más sencillo desarrollar aplicaciones en iOS.	Tiene un mayor número de usuarios

Aplicaciones más sencillas de hacer	Mantener una aplicación en Google Store (u otra tienda de apps para Android) es más barato
Aplicaciones más baratas de hacer	Se puede utilizar las aplicaciones en una amplia variedad de dispositivos, incluyendo móviles, dispositivos inteligentes, etc.
Aplicaciones más baratas de mantener	Menores restricciones en el entorno de desarrollo, mayor libertad
Requiere menos trabajo de desarrollo de código debido al uso de Swift como lenguaje	Más variedad de lenguajes de programación, como Java, JavaScript, Kotlin, etc.
<b>CONS</b>	
Mantener una aplicación en Apple Store es más caro	Requiere más tiempo de desarrollo
El número de usuarios de iOS es más bajo	Al haber más variedad de dispositivos, requiere más tiempo de adaptar el desarrollo a la variedad

(Elaboración Propia)

Tras los datos anteriormente expuestos se concluye que el sistema operativo escogido para el desarrollo del diseño de este TFM es Android. La razón principal para escoger Android como sistema operativo es que dispone de un mayor número de dispositivos, más de 1.000 millones de dispositivos, en el año 2020. Además, este tiene grandes ventajas al ser software libre, y dispone de ciertas facilidades principalmente en la incorporación y mantenimiento en el mercado de aplicaciones y software actual.

#### 4.1.2. Elección del software de desarrollo

Partiendo de la conclusión del apartado anterior se procede a escoger con qué software se va a realizar el desarrollo de la aplicación. Como la aplicación se va a desarrollar para Android, y la base, o la parte principal de la aplicación, es el uso de la Realidad Aumentada, se busca realizar el desarrollo en un programa que facilite la implementación de la RA. Para ello, tras realizar una búsqueda previa, se han encontrado multitud de programas y plataformas, entre los que destacan:

- Metaio Creator. Metaio es una empresa privada, centrada en la realidad aumentada y en el desarrollo de software y soluciones basándose en esta realidad aumentada. Esta empresa ha creado un software publicado como Metaio Creator, una herramienta creada para el desarrollo de aplicaciones de Realidad Aumentada. Metaio fue adquirida por

Apple Inc, en el año 2015, y desde ese año cesó la creación de nuevo software y la publicación de nuevos desarrollos y productos.

- Vuforia. Vuforia está considerado un kit de desarrollo de software, también conocido como SDK (software develop kit), que incorpora realidad aumentada, pero a diferencia de Metaio, Vuforia también incorpora visión artificial para reconocer imágenes planas y objetos y formas en 3D. Recientemente este software ha sido integrado dentro de Unity. (Vuforia Developer, 2022)
- ARTool Kit. La compañía china Realmax fundó ARTool Kit, el cual se inició como un proyecto de código abierto hacia el año 2001. Actualmente ARTool es un conjunto de librerías para desarrollo en C/C++, para crear aplicaciones con realidad aumentada. (ARTool Kit X, 2018)
- Google Sceneform Tools / Sceneform Maintained, es un plugin desarrollado por la compañía Google, el cual se puede añadir a softwares de desarrollo como puede ser Android Studio, entre otros. Se considera un kit de desarrollo de software, también conocido como SDK, al igual que lo es Vuforia. (Google LLC, 2020)

Existen otros softwares igualmente utilizados para el desarrollo de aplicaciones de realidad aumentada, pero están enfocados a otros sectores y su uso se enfoca de distinta forma. Uno de estos es Layar, el cuál es un software utilizado para crear imágenes interactivas que utilizan la realidad aumentada para promocionar productos de una marca. Como este tipo de software no es el apropiado para el desarrollo que se persigue, se ha decidido excluir ese tipo de software del listado.

Tras exponer los softwares más utilizados del mercado, se ha determinado que el más apropiado de los anteriores es Sceneform Maintained, y, por ende, este será el escogido para realizar el desarrollo de software. Este es la versión actual del original Sceneform Tools el cuál Google dejó de actualizar y ha quedado incompatible con los softwares de desarrollo actuales.

Sceneform Maintained se implementa como SDK en Android Studio, siendo Android Studio uno de los programas principales para desarrollo en Android en el mercado. Gracias a utilizar este software además del SDK, se tiene todo lo necesario para desarrollar la aplicación y no es necesario añadir ningún otro programa externo, programa con el cual se podrían generar incompatibilidades. Android Studio, es un software dedicado por completo al desarrollo de aplicaciones para Android, por lo que brinda un mayor número de herramientas y opciones

para la aplicación que buscamos. Además, un proyecto realizado en Android Studio se puede exportar directamente sobre un dispositivo móvil, ya sea físico o simulado, es por esto, junto con lo expuesto anteriormente, que se ha decidido desarrollar la aplicación en Android Studio junto con Sceneform Maintained.

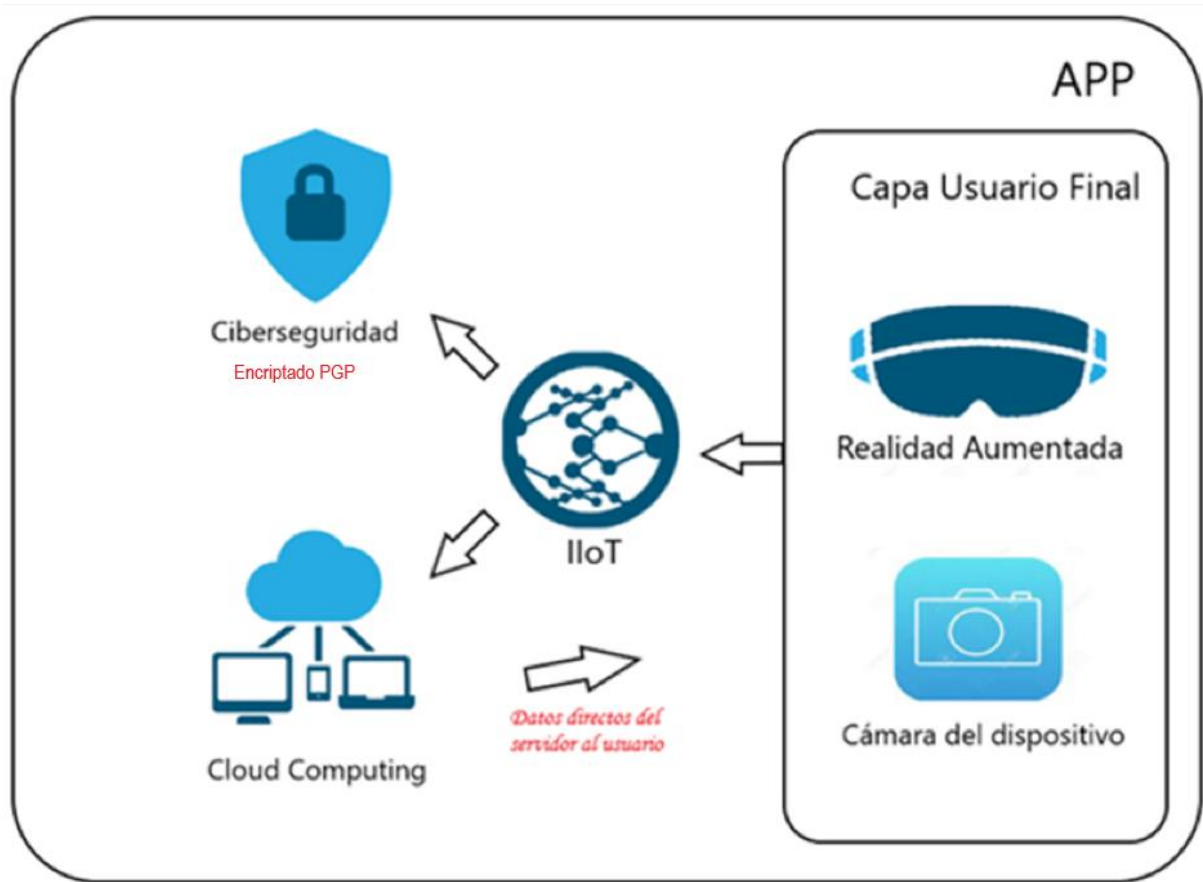
#### **4.1.3. Tecnologías implicadas: Diagrama**

Las tecnologías implicadas en el desarrollo de este TFM son las siguientes:

- Realidad aumentada. Esta es la base de todo el trabajo que está siendo desarrollado, la realidad aumentada.
- IIoT (Industrial Internet of Things o Internet Industrial de las Cosas). Esta tecnología es la que permitirá la conectividad entre la máquina escaneada y el servidor, o almacenamiento, en la nube, donde se encuentra toda la información acerca de los dispositivos.
- Cloud Computing. Esta tecnología es la encargada de almacenar de manera descentralizada de la empresa, la información que sea necesaria, por ejemplo, la información que debe aparecer en pantalla en un momento dado.
- Ciberseguridad. Esta tecnología se debe encargar de salvaguardar y proteger la información, para que solo las personas que deban verla y/o modificarla, puedan tener acceso a esta, y no pueda entrar cualquiera.

Estas son las principales tecnologías implicadas en la aplicación a desarrollar. A continuación, se expone el diagrama de cómo y en qué parte interviene cada una de las implicadas.





*Figura 12: Diagrama de tecnologías implicadas (Elaboración Propia)*

Con el diagrama anterior, Figura 12, se pretende dar una idea básica de las tecnologías que son necesarias para la aplicación y cómo se comunicarán entre ellas. El usuario percibirá la capa usuario final, en la que aparecen la cámara y la realidad aumentada.

Para que todas estas tecnologías y habilitadores puedan funcionar correctamente es necesario tener una conexión a una red wifi que disponga de internet en caso de que el servidor esté alojado en la nube, o en el caso de que una empresa requiera la adaptación a un servidor propio, bastaría con conectarse a la red local. De cualquier modo, es estrictamente necesario tener una conexión a la red en el dispositivo para que la aplicación pueda funcionar con normalidad.

Por otro lado, el uso de Cloud Computing está orientado sobretodo a la adaptación a PaaS, pero en el ecosistema base también es utilizado para almacenamiento y gestión de datos en

la nube, dado que dependerá de lo que decida la empresa y de la cantidad de datos que tengan que ser almacenados.

La Ciberseguridad será la encargada de proteger la información que aparece en el ecosistema, así como a los usuarios. Para ello, utiliza un protocolo PGP (Pretty Good Privacy), el cual tiene por finalidad encriptar y desencriptar toda la información que se distribuye en el ecosistema. PGP utiliza un sistema de encriptado y desencriptado haciendo uso de firmas digitales y claves públicas por lo que configurar la aplicación requerirá de mayor complejidad, pero asegurará una mayor seguridad en su uso, manteniendo la información privada segura. La información dentro de cada ecosistema es propiedad de la empresa que hace uso de este, y únicamente el personal autorizado puede acceder a la información en cuestión. El protocolo de comunicación junto con algunas otras medidas de seguridad podrán ser elegidas por la empresa que utilice la plataforma o el ecosistema.

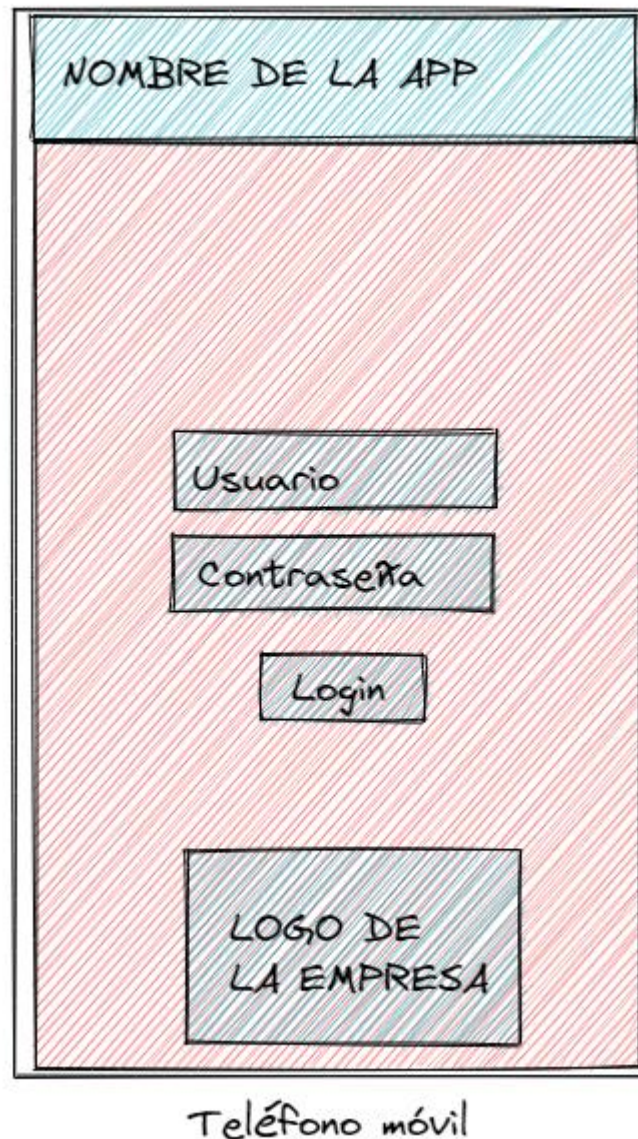
#### **4.1.4. Diseño de baja fidelidad**

Para este primer diseño se ha planteado lo siguiente.

La aplicación requiere de conexión a red y de acceso a la cámara, por lo que, en caso de no disponer de estos accesos, deberá solicitarlos. Para el caso de la red, si durante la ejecución de la aplicación se desconecta la red ya sea de forma manual o por accidente, la aplicación debe seguir funcionando, pero no podrá mostrar información más allá que la que ya estuviera en pantalla.

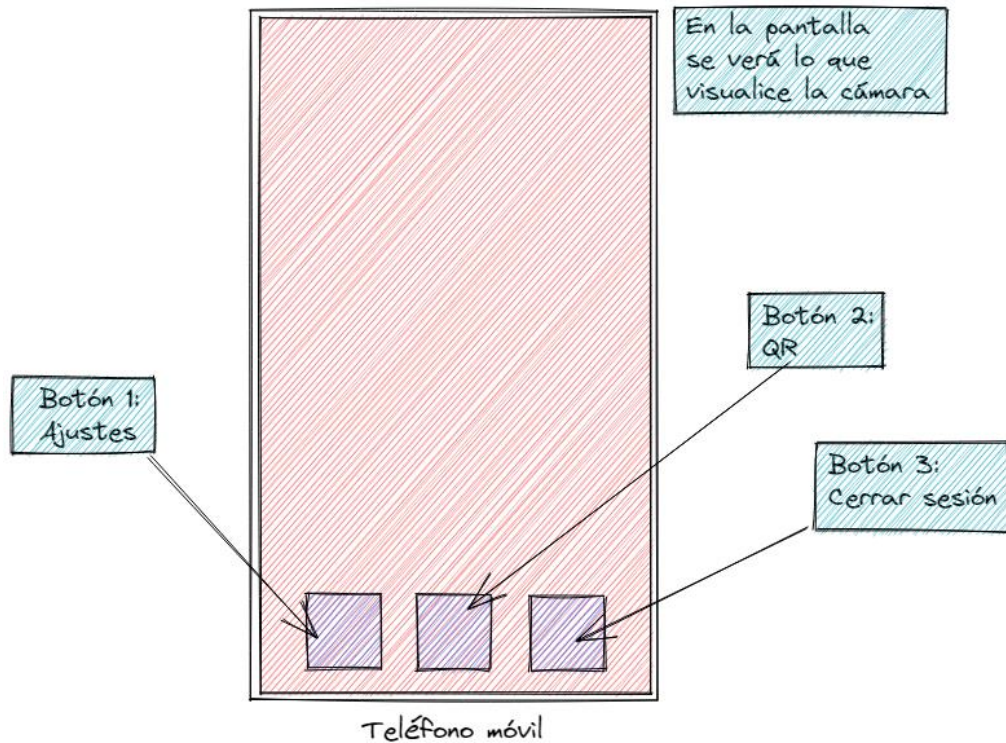
Inicialmente se solicitará unas credenciales, que son usuario y contraseña, para garantizar el acceso. En la primera versión se plantea con un único usuario y una única contraseña. Como se busca que la versión sea escalable, e incluso comercializable como PaaS, más adelante se planteará la alternativa de incorporar una base de datos de usuarios, contraseñas y permisos de los usuarios, para que, en caso de uso en empresas, se pueda filtrar quién accede a la información y a qué información tiene acceso.

Inicialmente se plantea el primer boceto realizado en “papel y lápiz”. La aplicación tendrá una interfaz sencilla, al iniciarla, se abrirá una pantalla de inicio de sesión:



*Figura 13: Diseño de baja fidelidad. Pantalla de inicio de sesión (Elaboración propia)*

Una vez se escriban correctamente las credenciales, usuario y contraseña, y se pulse el botón Login, se accederá a la pantalla principal de la aplicación. Para mantener un diseño minimalista y limpio, desde esta pantalla se debe poder visualizar el máximo de información posible, una vez se haya escaneado la máquina o elemento que pase por la realidad aumentada. Es por esto que se realizará un diseño con el mínimo número de iconos, o botones, posible. A continuación, se expone el boceto inicial de la pantalla principal.



*Figura 14: Diseño de baja fidelidad. Pantalla principal (Elaboración Propia)*

En este primer boceto se plantea que la aplicación tenga únicamente tres botones, uno para ajustes, donde se desplegará una ventana flotante en la que se permita cambiar el tamaño de los iconos, de la letra, el color de relleno de las ventanas de información, el idioma y comprobar si hay alguna actualización disponible. Este tipo de ajustes se podrán modificar en caso de que el cliente así lo requiera, pudiendo añadir o quitar algunas de las opciones.

A continuación, se expone el diseño del menú de ajustes.



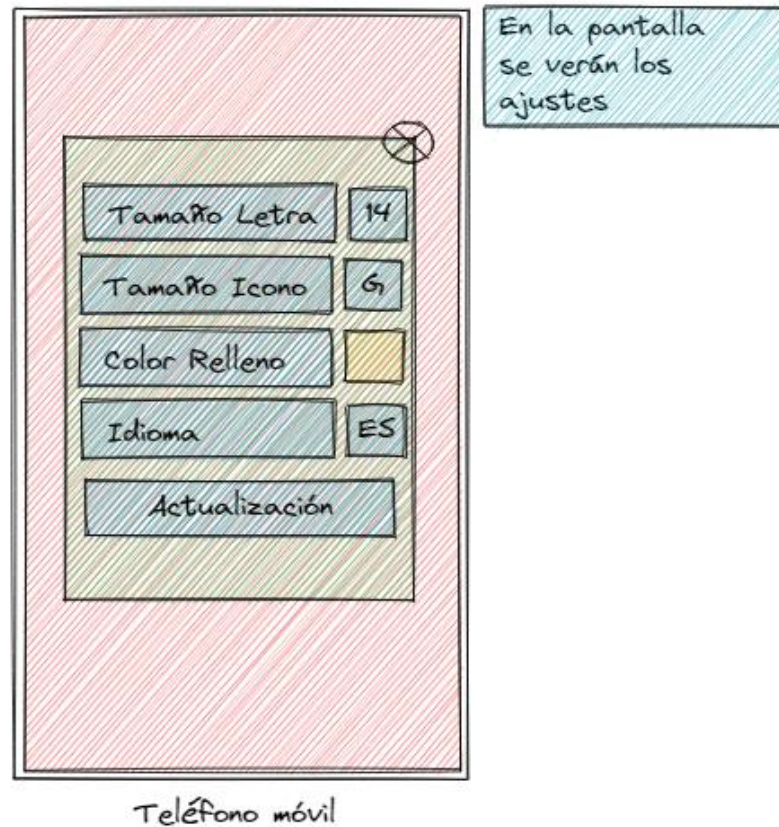
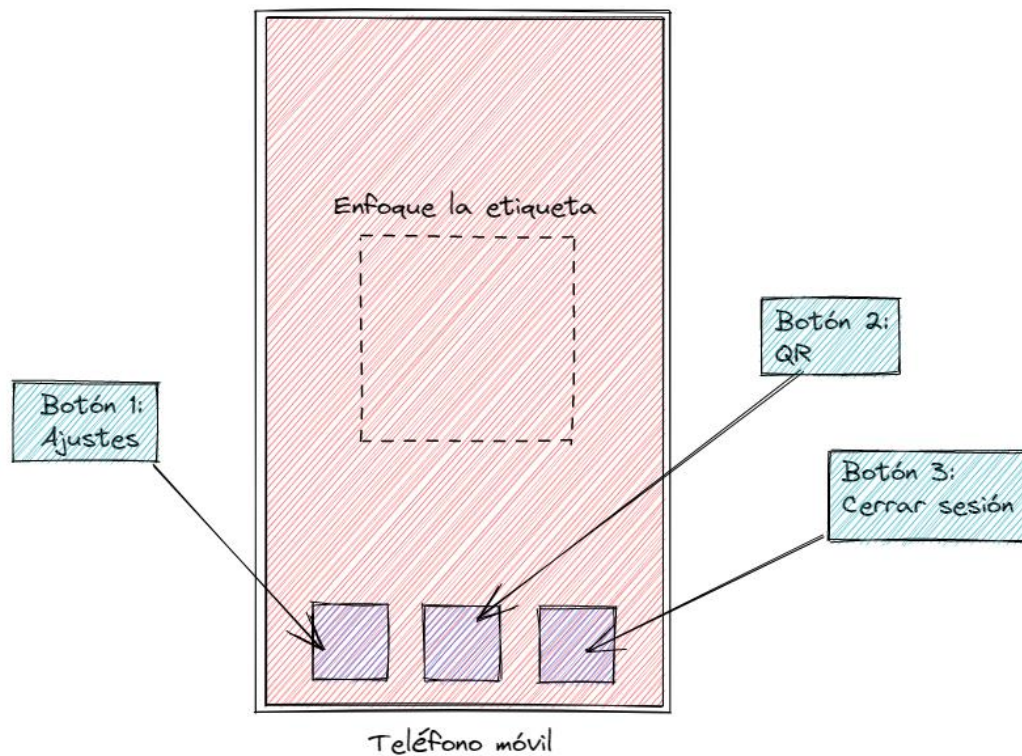


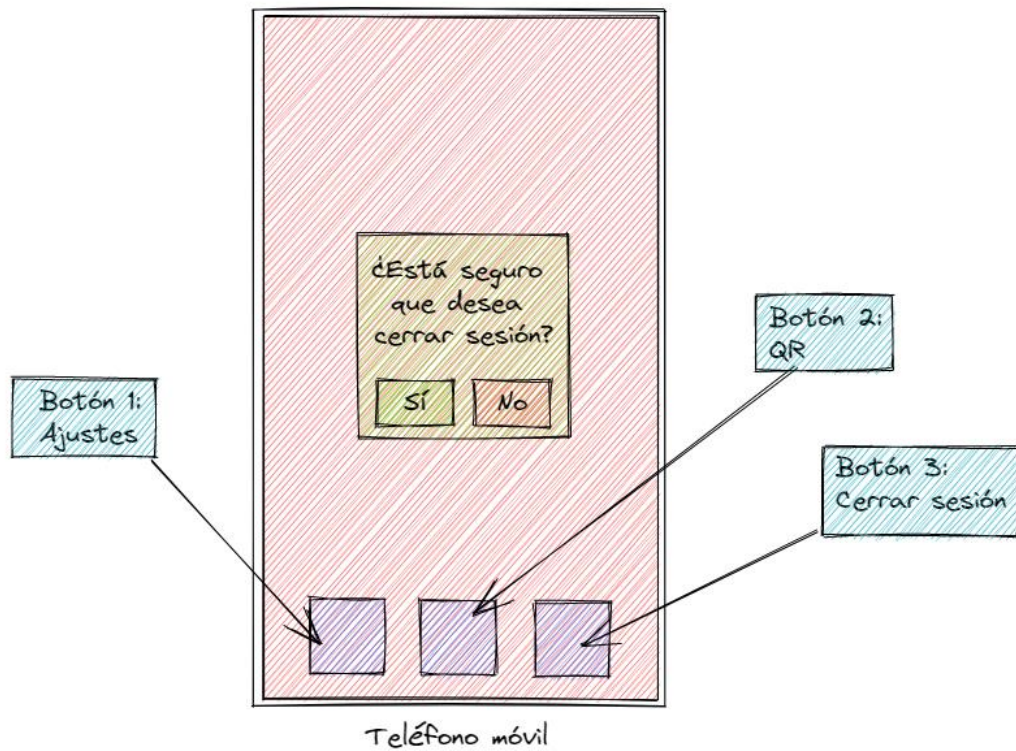
Figura 15: Diseño de baja fidelidad. Menú Ajustes (Elaboración Propia)

Siguiendo el orden de los botones, está el denominado Botón 2: QR. Este es el botón central, y su función será activar el escáner auxiliar, por si la aplicación no detectase correctamente la máquina, el usuario disponga de la opción de escanear una etiqueta o un código QR. Además de eso, si el usuario ya ha escaneado una, y quiere escanear otra, tendrá que utilizar el botón. Este botón también puede ser utilizado para obtener información adicional como pueden ser manuales o guías de funcionamiento, y poder disponer de un link directo a esta información, siempre que se esté enfocando a la máquina y haya sido correctamente escaneada.



*Figura 16: Diseño de baja fidelidad. Menú Escanear (Elaboración Propia)*

Finalmente, el Botón 3: Cerrar sesión, situado en la esquina inferior derecha, cuya función es cerrar sesión o permitir un cambio de usuario, por si es necesario realizar alguna de estas acciones en un momento dado. Una vez el usuario pulse, le aparecerá un mensaje en pantalla pidiendo confirmar que desea salir de la cuenta, y lo mandará a la pantalla de la Figura 13, para que pueda introducir otras credenciales.



*Figura 17: Diseño de baja fidelidad. Menú Cerrar Sesión (Elaboración Propia)*

El diseño de la aplicación busca ser minimalista y limpio dejando que la función principal, que es dar información sobre lo que el usuario escanee, ocupe el máximo posible. Por ello se dejan los mínimos botones posibles, para que lo que vea la cámara sea lo que se muestre principalmente en pantalla, sin restricciones de espacio.

## 4.2. DISEÑO DEL SOFTWARE

En este capítulo se procede a transformar el boceto en papel, expuesto en el apartado 4.1.4, a un software que pueda ser ejecutado en un teléfono, o cualquier otro dispositivo, con sistema operativo Android y que a su vez disponga de cámara.

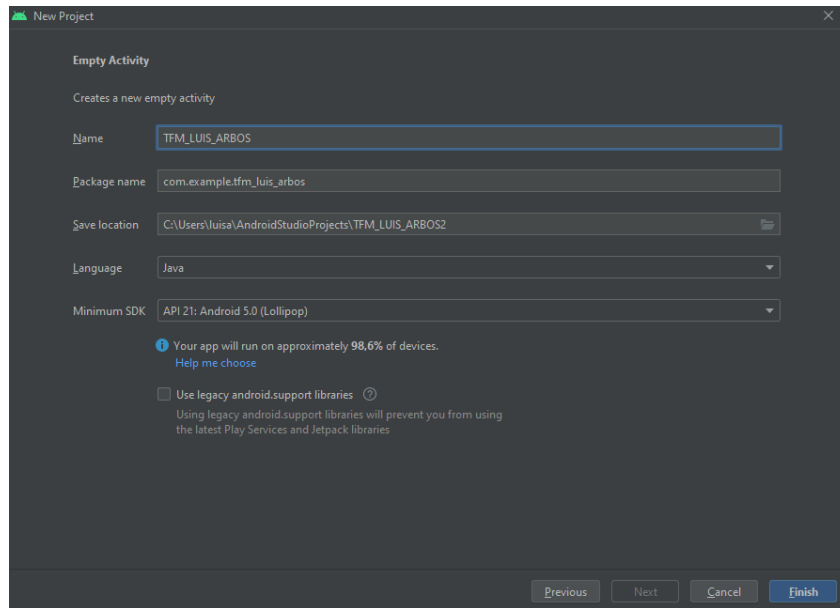
Este apartado se estructurará de la siguiente forma:

- Creación de proyecto en Android Studio
- Creación y configuración de las distintas pantallas
- Primeras pruebas de simulación del proyecto en Android Studio
- Incorporación del plugin de RA, configuración de la realidad aumentada y pruebas de funcionamiento
- Dificultades encontradas en el desarrollo

### 4.2.1. Creación de proyecto en Android Studio

Se comienza creando un proyecto en Android Studio. En este caso, el proyecto, o la aplicación, recibe el nombre de TFM\_LUIS\_ARBOS.





*Figura 18: Pantalla de creación de proyecto Android Studio (Elaboración propia)*

En la figura anterior se aprecia la configuración que se utiliza, la aplicación está programada en Java, y el sistema operativo mínimo que requiere para funcionar es Android 5.0 Lollipop. Esta versión quedó atrás hace años, pero es la primera en la que esta aplicación podría funcionar, lo que significa, que actualmente podría ejecutarse esta aplicación en al menos un 98,6% de dispositivos con Android, lo que supone un número de dispositivos aptos de más de 1.200 millones.

#### **4.2.2. Creación y configuración de las distintas pantallas**

Hecho esto se proceden a crear todas las pantallas necesarias para este programa. Para este caso con dos pantallas es suficiente para cubrir todas las opciones de la aplicación.

- Pantalla de acceso
- Pantalla principal

Para el desarrollo visual, Android Studio ofrece dos opciones, una pestaña Diseño, en la que se puede añadir cada elemento de forma sencilla e intuitiva, y posteriormente ajustar las características en una tabla, o, por otro lado, utilizar la página de código e incorporar todo utilizando programación en XML. La opción de hacerlo usando código en XML es menos intuitiva para el usuario medio, pero es un desarrollo similar al CSS que se usa en el desarrollo web.

#### 4.2.2.1. Pantalla de acceso

Para la pantalla de acceso, siguiendo el boceto del diseño de baja fidelidad, en la Figura 13, se ve que hay una barra superior con el nombre de la aplicación. En la parte central se encuentran dos espacios para colocar el usuario y la contraseña y bajo estos se encuentra el botón de acceso, botón de “login”. Además de lo anterior, se le añade un campo de texto oculto bajo el botón de acceso, para que, en caso de colocar mal las credenciales, devuelva un mensaje de error. En la parte inferior, se encuentra el logo de la empresa, que para esta aplicación se ha utilizado el logo de UNIR.

Con todas las características anteriores ya estaría la primera pantalla visual completada, a falta de añadir detalles como cambios de color, y ajuste de tamaños. Se procede con la parte de programación. Android Studio es un software sumamente completo en el que cada proyecto tiene un código principal, el conocido como código Main, y se puede realizar todo en este directamente, o ir dividiéndolo en subcódigos para ayudar a entender mejor la programación del mismo.

Para el desarrollo de esta aplicación se realiza una parte de la programación, que corresponde a la pantalla de acceso, sobre el archivo Main, y se genera un segundo archivo para la pantalla principal, en el cual se realizará la mayoría de la programación.

#### 4.2.2.2. Pantalla principal

Para la pantalla principal, siguiendo el boceto del diseño de baja fidelidad, en la Figura 14, se ve que la barra superior con el nombre de la aplicación no está. Esto se ha hecho así para dejar mayor espacio de visualización a lo que vea la cámara, facilitando el uso de la aplicación. Lo único que aparece en esta pantalla son los tres iconos en la parte inferior. Una vez añadidos ya estaría completada la base de esta pantalla.

Para esta pantalla concreta, cada botón deberá desplegar la ventana correspondiente. El primer botón, el de más a la izquierda, despliega la ventana de ajustes que corresponde a la Figura 15. La misma ventana de ajustes se encontrará añadida sobre la pantalla principal, pero con la visualización anulada, para que el usuario no pueda verla a menos que pulse el botón de ajustes.

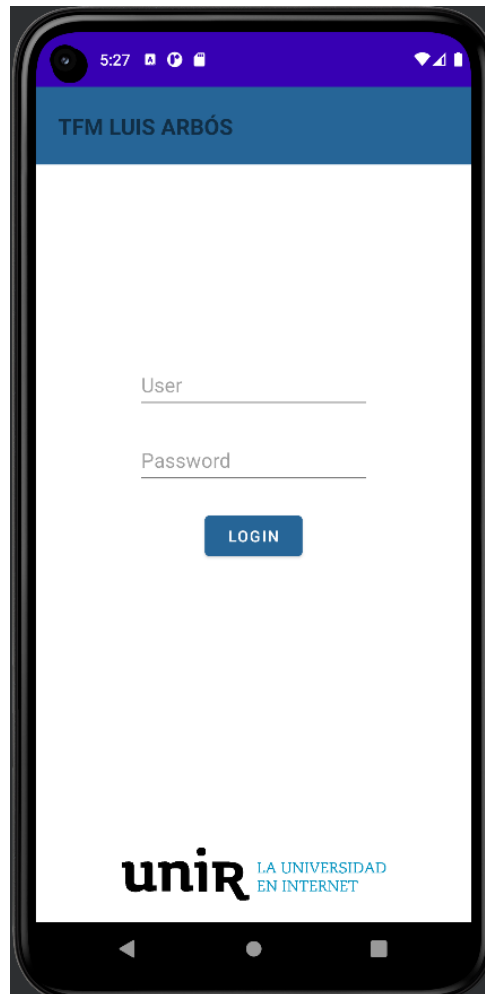
Una vez pulsado el botón central, que es el botón QR, aparecerá en el centro un recuadro y un texto, para indicar que está en modo lectura y que el usuario proceda a enfocar el QR deseado, como se muestra en la Figura 16. Esta característica también se diseña sobre esta pantalla, con la peculiaridad de que se programa el cuadrado para que visualmente aparezca y desaparezca dependiendo del botón QR.

Lo mismo sucede para el tercer botón, el de cerrar sesión, el cuál al ser pulsado hace aparecer un mensaje como muestra la Figura 17. En el momento de diseñar todos estos elementos se ha tenido en cuenta que deben estar presentes, aunque posteriormente, mediante la programación en Java, se activan o desactivan para que el usuario los vea y pueda interactuar con ellos.

#### **4.2.3. Primeras pruebas de simulación del proyecto en Android Studio**

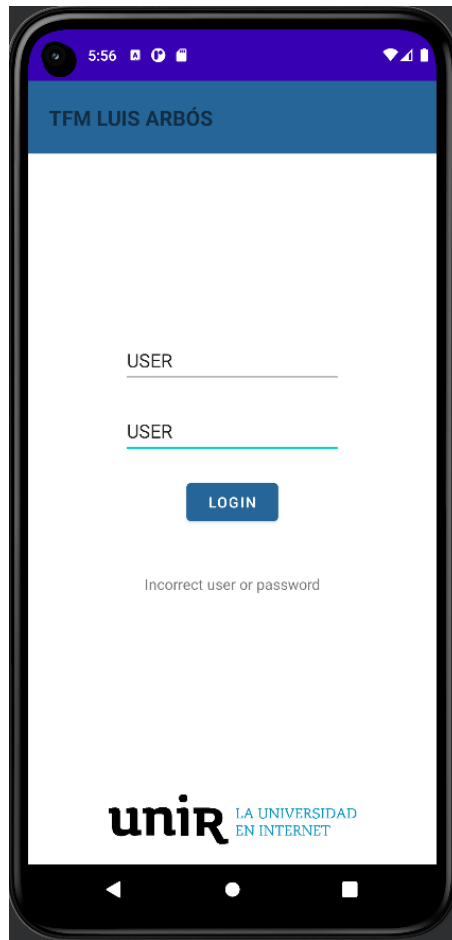
La plataforma Android Studio brinda distintas opciones para poder realizar pruebas del funcionamiento de la aplicación. Dispone de una opción para cargar la aplicación sobre un dispositivo Android. La ventaja que ofrece Android Studio es que no es necesario que este sea un dispositivo físico, un dispositivo real, sino que el programa permite la simulación de un dispositivo para posteriormente emular en este la aplicación. Para este caso, se utiliza un dispositivo Google PIXEL 5, con un sistema operativo Android 11.0, una versión bastante más actualizada que sobre la que estamos trabajando.

Se crea y configura el dispositivo en predeterminado, sin modificar ninguna opción. Una vez configurado el dispositivo, se procede a simular una por una las pantallas y ventanas anteriores para comprobar cómo se vería en el dispositivo del usuario final, la aplicación.



*Figura 19: Simulación de la pantalla de acceso (Elaboración propia)*

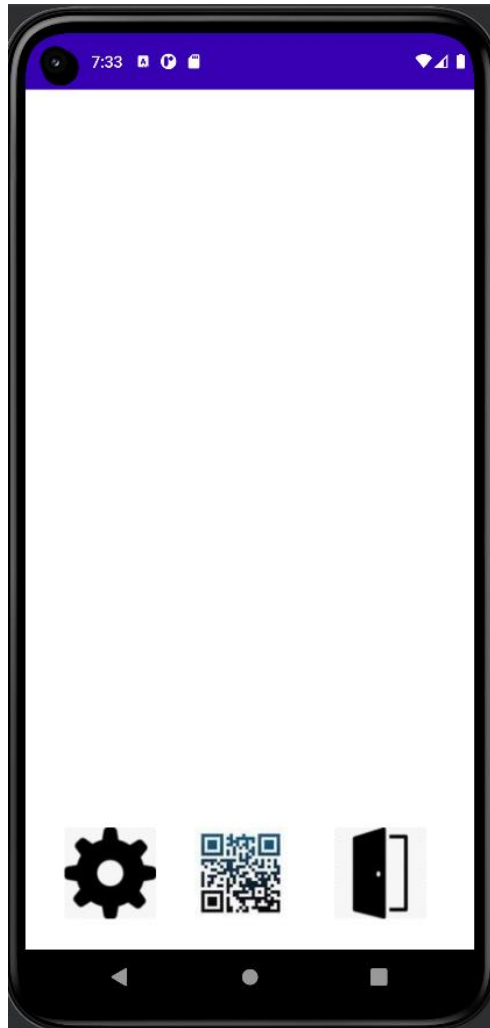
Esta primera pantalla se ve bastante similar al diseño inicial, el boceto de baja fidelidad. Se procede a realizar una comprobación poniendo incorrectamente las credenciales para ver si devuelve algún mensaje.



*Figura 20: Pantalla de acceso con credenciales erróneas (Elaboración propia)*

Al utilizar un usuario y/o una contraseña incorrectos devuelve un mensaje, bajo el botón de “Login”, que indica que el usuario o la contraseña son incorrectos.

Una vez los campos de usuario y contraseña son los correctos, y el usuario pulsa el botón de “Login”, la aplicación cambia a la pantalla principal.



*Figura 21: Pantalla principal (Elaboración propia)*

La primera vez que el usuario acceda a la aplicación y entre a la pantalla principal, se requerirá permisos para utilizar la cámara.

La pantalla principal sigue el diseño del boceto de baja fidelidad, únicamente una pantalla que en la simulación aparece en blanco pero que posteriormente, cuando se le añada el plugin de realidad aumentada, toda la zona blanca quedará cubierta por todo lo que visualice la cámara.

Llegado a este punto están los tres botones, si el usuario pulsa el primero, el icono del engranaje, se despliega el menú de opciones. Este está diseñado como una ventana flotante, por lo que el fondo que se ve alrededor sigue siendo el mismo.



*Figura 22: Pantalla Ajuste. (Elaboración propia)*

En este caso también se ha seguido el diseño del boceto de baja fidelidad, se han mantenido las opciones de configuración, y su diseño no ha cambiado. La mayoría de opciones están hechas mediante un desplegable con distintas opciones, para evitar que el usuario final cometa errores en el caso de que tuviera que introducirlo mediante texto.

Además de esto, se puede cambiar el idioma de la aplicación, el cual cambia el texto de la pantalla de ajustes, de la pantalla de acceso y los textos que vayan apareciendo en la principal.

Para salir de esta ventana, el usuario puede volver a pulsar el icono del engranaje o la cruz roja de la parte superior derecha.

Volviendo a la pantalla principal está el segundo botón, el icono del QR en el centro. Al pulsar este icono se despliega sobre la misma pantalla un texto con un recuadro que indica que se enfoque hacia la etiqueta para escanearla.



*Figura 23: Pantalla principal. Menú QR. (Elaboración propia)*

Finalmente, el tercer botón de la pantalla principal, el icono de una puerta, es para cerrar sesión. Al pulsarlo, le aparecerá al usuario final un desplegable siguiendo el diseño de baja fidelidad. En este el usuario debe seleccionar si quiere cerrar sesión o no, y caso de pulsar sí será redirigido a la pantalla de acceso, y si selecciona el no, se quedará en la misma pantalla.





*Figura 24: Pantalla principal. Menú Cerrar Sesión. (Elaboración propia)*

#### **4.2.4. Incorporación del plugin de RA, configuración de la realidad aumentada y pruebas de funcionamiento**

Se añade el plugin de realidad aumentada para que la aplicación sea funcional. El plugin que se utiliza es Sceneform Tools Maintained. Gracias a este se podrá añadir uno o varios modelos en 3D, en los que se podrá ver la información.

Previamente a incorporar el modelo se solicitan permisos para activar la cámara, y se hace una prueba con la cámara encendida. El fondo que se simula es el predeterminado que tiene Android Studio, debido a que se está utilizando el simulador que posee.



*Figura 25: Simulador con cámara (Elaboración propia)*

Comprobado el correcto funcionamiento Para ello lo que se hace es añadir el plugin, y configurar un primer modelo de panel de control cuya información aparecerá al escanear el QR del panel. Para el ejemplo simulado, dado que no se dispone de ningún panel de control de ninguna máquina, se simula un panel de control de una CNC en 3D. (Passmore, 2020)



*Figura 26: Simulación de panel CNC en aplicación (Elaboración propia) (Passmore, 2020)*

Llegado a este punto ya puede generar modelos, por lo que se le añaden mensajes sobre los modelos y se ajusta la aplicación para que el modelo se genere una vez se escanee el QR.

#### **4.2.5. Dificultades encontradas en el desarrollo**

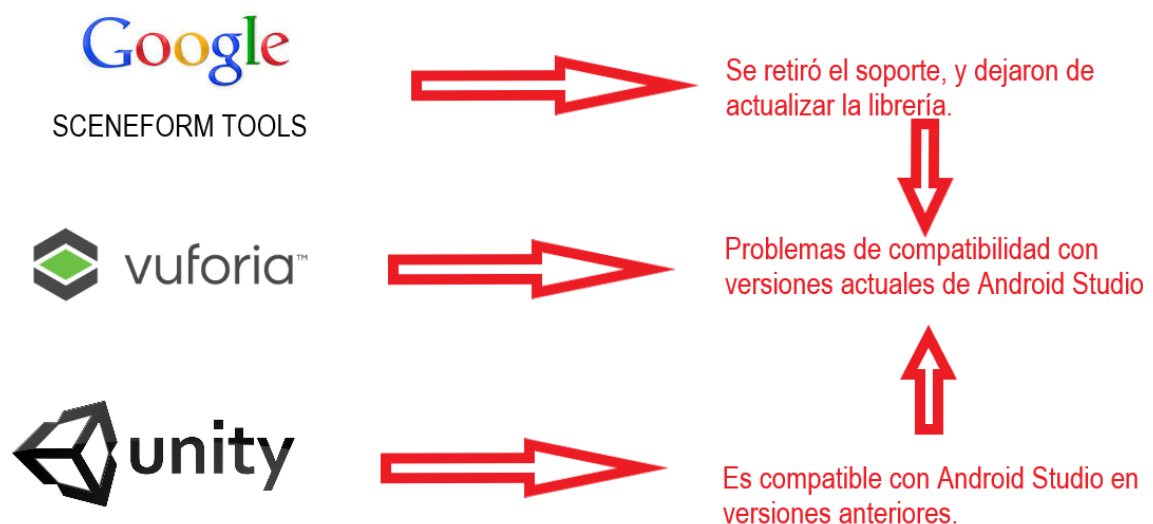
El primero y principal problema del desarrollo realizado ha sido encontrar el software apropiado para realizarlo. La idea inicial era realizarlo utilizando Android Studio y combinándolo con Unity y Vuforia, pero debido a un problema de compatibilidad que no está solventado actualmente, no se ha podido realizar con ello.

En un segundo momento se tomó la decisión de realizar el desarrollo completo en Android Studio, cambiando el resto de software por librerías de Android Studio que realicen la misma función. El número de librerías de realidad aumentada de las que dispone Android Studio es reducido, y se escogió Vuforia debido a que era uno de los softwares a utilizar inicialmente. El

problema surgió porque Vuforia actualmente no da soporte a Android Studio, por lo que su librería dejó de funcionar hace varias versiones.

La tercera idea fue cambiar Vuforia por Sceneform Tools ya que es una librería de Google. (Google LLC, 2020). Ahí surgió el siguiente problema, en el año 2020 Google decidió no mantener la librería e incluso retirarla por lo que dejaba de estar disponible.

Finalmente, se buscó otra alternativa más, que es Sceneform Maintained, que es la antigua librería Sceneform Tools de Google pero mantenida por la comunidad de Github, la cual tiene soporte y sigue funcionando aunque dispone de pocas opciones. A su vez, esta librería también contiene una parte de Google Filament que utiliza como motor de generación de modelos en 3D. (Gorisse, 2022)



*Figura 27: Problemas de otras librerías (Elaboración propia)*

Gracias a esta librería se ha podido realizar el desarrollo de la primera versión de la aplicación. Aun así, para el caso de desarrollar futuras versiones se deberá incorporar otras librerías que den una mayor variedad de opciones para mejorar la versión actual.

### 4.3. ADAPTACIÓN A PAAS

En este capítulo se plantea llevar el desarrollo un paso más allá, enfocándolo al negocio. Para ello se busca modificar lo necesario, dentro de la aplicación, para que esta sea adaptable y ajustable al entorno de cada empresa o industria, pudiendo así, ser incorporado en el mercado como un servicio PaaS, (Platform-as-a-Service o Plataforma como servicio). Para que pueda plantearse como PaaS hay ciertos requisitos recomendados, que son:

La información debe de estar alojada en la nube. No se deben consumir recursos de almacenamiento dentro de los servidores de la empresa, y tiene que poderse acceder a la información desde cualquier punto de cualquier nave de la empresa, por lo que el almacenamiento de la información en la nube y tener conexión a internet para poder acceder a esta, son requisitos mínimos indispensables.

Este apartado se subdividirá en los siguientes apartados:

- ¿Qué ventajas ofrece la salida a mercado de la aplicación como PaaS?
- KPIs y resultados esperados
- Costes y retorno de la inversión

#### **4.3.1. ¿Qué ventajas ofrece la salida a mercado de la aplicación como PaaS?**

El primer, y más importante punto, para poder conocer la viabilidad de esta aplicación en el momento de llevarla al mercado como PaaS es, plantear qué ventajas podría ofrecer. Es por esto que este apartado tiene por objeto exponer las ventajas que supone la comercialización de la aplicación como una PaaS. El uso y la incorporación de la aplicación en cuestión, que se plantea como PaaS, en el entorno laboral presenta tres grandes ventajas principales.

- El ofrecer el ecosistema como servicio PaaS a una empresa tiene la ventaja de poder ejecutar varios ecosistemas sobre la misma plataforma. En una misma empresa se puede utilizar la plataforma para formación de partes completamente opuestas, por ejemplo, dos sedes de la empresa que tengan funciones diferentes, teniendo un ecosistema propio para cada una, pero con una plataforma común.
- La segunda ventaja que presenta es que dispone de mucha facilidad para la migración de los ecosistemas. La reprogramación de la aplicación es mucho más rápida, sencilla y menos costosa. Al ofrecer el ecosistema como servicio PaaS, todos los ecosistemas que tengan una única plataforma común tendrán que partir de un ecosistema base y adaptarse al uso final de este. Gracias a ofrecer el ecosistema como PaaS, la reprogramación, o el ajuste del sistema a cada caso de uso específico es mucho más rápido, más sencillo, y supone un menor desembolso económico a la empresa que lo utilice.
- Solo se necesita un único experto en la información que vaya a contener el ecosistema, por cada uno de los ecosistemas que se tengan dentro de la plataforma. Con un único experto en esa información, junto con el desarrollador de software que tendrá que modificar o adaptar el ecosistema al uso será suficiente para formar a todo empleado que así lo requiera, lo que supone una ventaja y un gran ahorro.

Estas tres ventajas expuestas, son las razones principales de plantear ofrecer la aplicación como PaaS para las empresas. Estas ventajas conllevan ventajas secundarias, como puede ser conseguir un entorno de trabajo más seguro, unos empleados más formados, e incluso atraer un mayor número de clientes gracias a tener un entorno de trabajo más seguro y preparado.

#### **4.3.2. KPIs y resultados esperados**

En este apartado se exponen los KPIs (Key Performance Indicator, cuya traducción al castellano sería indicadores claves para medir el rendimiento o el desempeño). Estos KPIs se utilizan, tal y como su nombre indica, para medir las mejoras que haya podido haber en el rendimiento en un campo determinado. A continuación, se exponen KPIs que se utilizarán para determinar la eficacia del uso tanto del ecosistema como de la plataforma.

Del uso del ecosistema en una empresa se esperan los siguientes resultados:

- Disminución del número de accidentes. El número de accidentes que hay en una empresa de producción debería descender, dado el hecho que los empleados disponen de más información de seguridad, aunque siempre dependa de ellos aplicar los conocimientos de los que se dispone.
- Disminución de la gravedad de los accidentes producidos. La gravedad de los accidentes debe verse reducida. Conseguir alcanzar 0 como número de accidentes en una empresa es complicado, pero se debe insistir siempre en la seguridad, y si ha de haber accidentes, estos tendrían que ser de la menor gravedad posible.
- Mejora en los tiempos de producción. Como los empleados dispondrán de una mayor cantidad y calidad de información, no tendrán la necesidad de avisar a un superior o a un compañero a cada cosa que no sepan o recuerden cómo hacer, por lo que se debe ver una mejora en el tiempo de producción.
- Reducción de gasto en formación. Debido a que la plataforma es un extra en la formación, y en ella se dispone de una amplia cantidad de información, el gasto que una empresa hace en formación adicional y específica a los empleados se debería de ver reducido, dado que la plataforma les debe facilitar toda la información necesaria.

Por otro lado, del uso de la plataforma como PaaS, se espera lo siguiente:

- Baja tasa de errores de la plataforma. Debido a que se ofrece el ecosistema como PaaS, se busca que este tenga una tasa de errores muy baja, lo más cercada a 0 posible.
- Alta tasa de concurrencia de usuarios. Al utilizarlo como PaaS, se monitorizará el uso de la plataforma, y se espera que la tasa de usuarios que utiliza la plataforma sea muy alta, siendo esta de forma simultánea.

- Bajo tiempo de respuesta. La plataforma requiere de una conexión estable y rápida con los usuarios, por lo que se espera tener un bajo tiempo de respuesta, y una baja latencia, para asegurar que la información llega eficaz.

Para poder medir los resultados tanto del ecosistema como de la plataforma, se plantean 7 KPIs principales, uno para cada resultado. Estos KPIs son los que se van a tener en cuenta, y expondrán si el uso de la plataforma da el resultado esperado. Los KPIs principales del ecosistema, se espera que se analicen como indicadores globales, que denoten mejoras a largo plazo. Por otro lado, los KPIs principales de la plataforma se espera que puedan ser analizados a corto-medio plazo. Estos deberán de ser revisado como mínimo de forma trimestral, dado que en caso de revisarlos con mayor frecuencia es difícil ver una mejora sobre los datos.

Para poder revisar estos datos de forma simplificada se exponen sobre un cuadro de mando integral. Este tipo de cuadros expone los datos de forma clara y sencilla desde cuatro puntos de vista distintos que son, el financiero, el del cliente, el interno y el de aprendizaje. Para este caso se ha creado un cuadro de mando adaptado a los KPIs expuestos, en el que se puede apreciar con claridad la mejora de forma tangible dentro de la empresa.

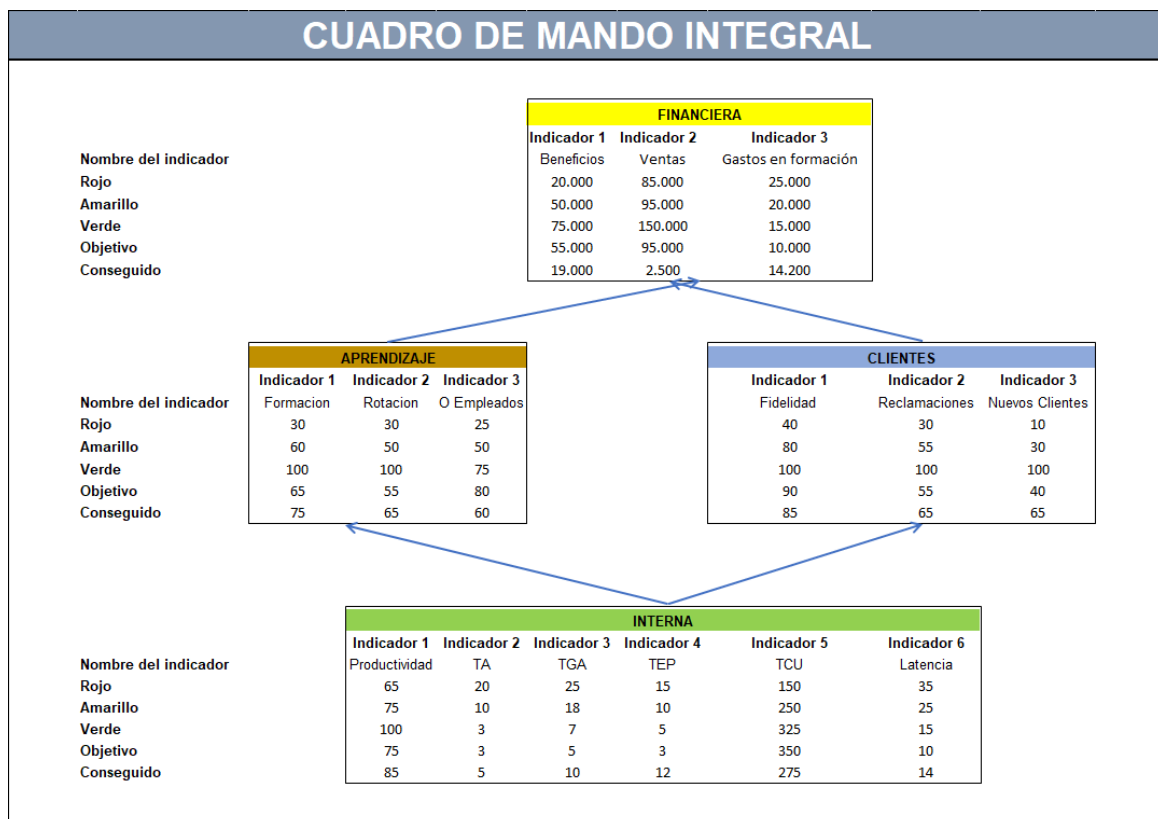
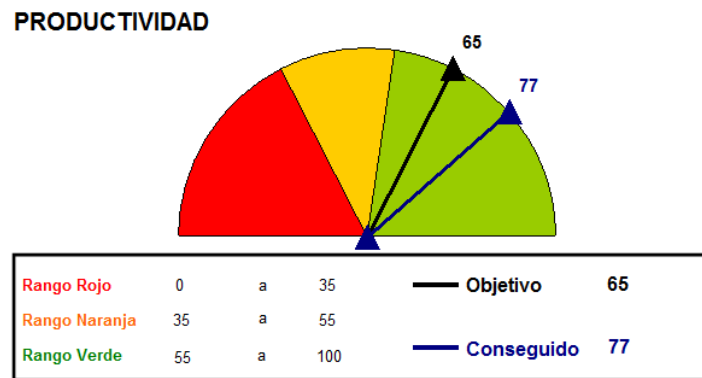


Figura 28: Cuadro de Mando Integral (Elaboración propia)





*Figura 29: Indicador con datos más recientes (Elaboración propia)*

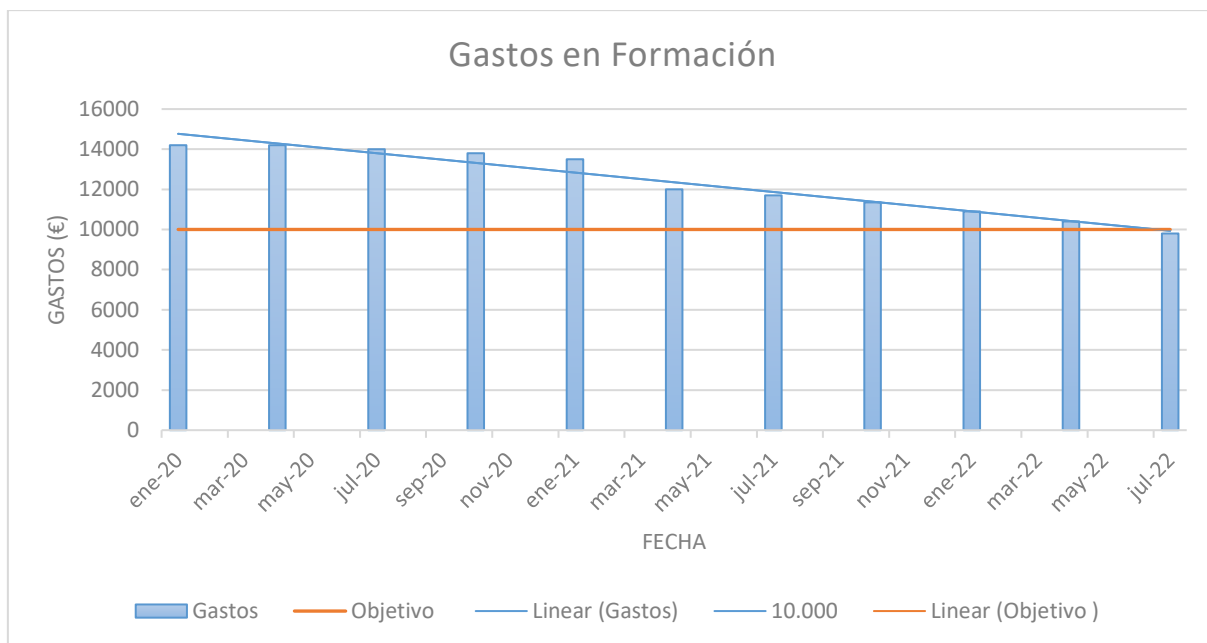
En el anterior cuadro de mando podemos ver que están los valores distinguidos en 4 grupos:

- FINANCIERA
  - Beneficios. Indica los beneficios netos de la empresa por trimestre.
  - Ventas. Indica las ventas de la empresa en € por trimestre
  - Gastos en formación. Indica el gasto en formación de la empresa por trimestre.
- INTERNA
  - Productividad. Indica los niveles de productividad de la empresa en productos hechos por trimestre.
  - Tasa de accidentes. Indica la tasa de accidentes por trimestre.
  - Tasa de gravedad de accidentes Indica la gravedad media de los accidentes por trimestre.
  - Tasa de errores de la plataforma. Indica el número de veces que la plataforma, o alguno de sus ecosistemas falla, medidos en fallos trimestrales.
  - Tasa de concurrencia de usuarios. Indica el número de usuarios máximos que están conectados de forma simultánea en la plataforma, medidos de forma diaria.
  - Latencia y tiempo de respuesta. Indica el tiempo de respuesta y la latencia media en milisegundos que se espera de la plataforma.

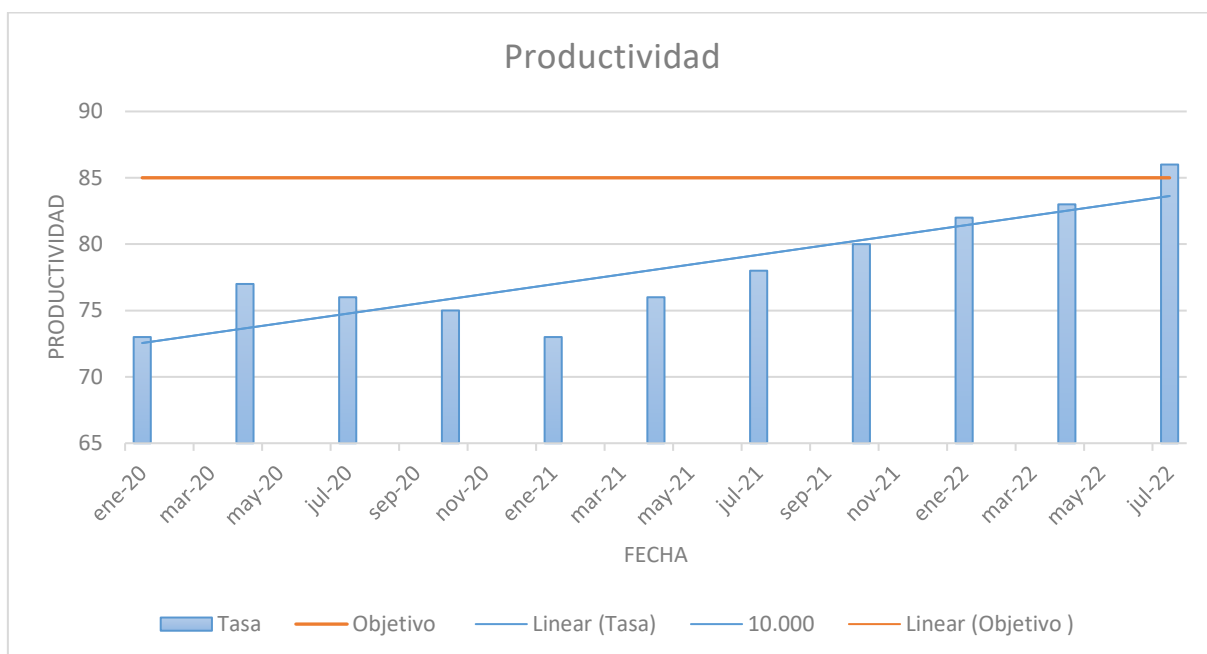
- **CLIENTES**
  - Fidelidad. Indica el nivel de fidelidad de los clientes. Se mide a través de encuestas de satisfacción de forma trimestral.
  - Reclamaciones. Indica el número de reclamaciones por trimestre.
  - Nuevos clientes. Indica el número de nuevos clientes por trimestre
- **APRENDIZAJE**
  - Formación. Indica el nivel de formación de los empleados.
  - Rotación. Índice de rotación de los empleados, indica el número de empleados que son contratados y que se van o son despedidos por trimestre.
  - Opinión de los empleados. Indica el nivel de satisfacción que tienen los empleados, se mide de forma trimestral mediante encuestas de satisfacción.

De todos los indicadores anteriores, los 4 KPIs principales del ecosistema, que se remarcaron previamente, son todos los de la categoría Interna, y además el indicador de gastos en formación de la categoría financiera. Los 3 KPIs principales de la plataforma, también forman parte de la categoría interna. Los demás indicadores pueden servir también para ver una mejora reflejada a nivel global sobre todos los apartados del cuadro de mando integral.

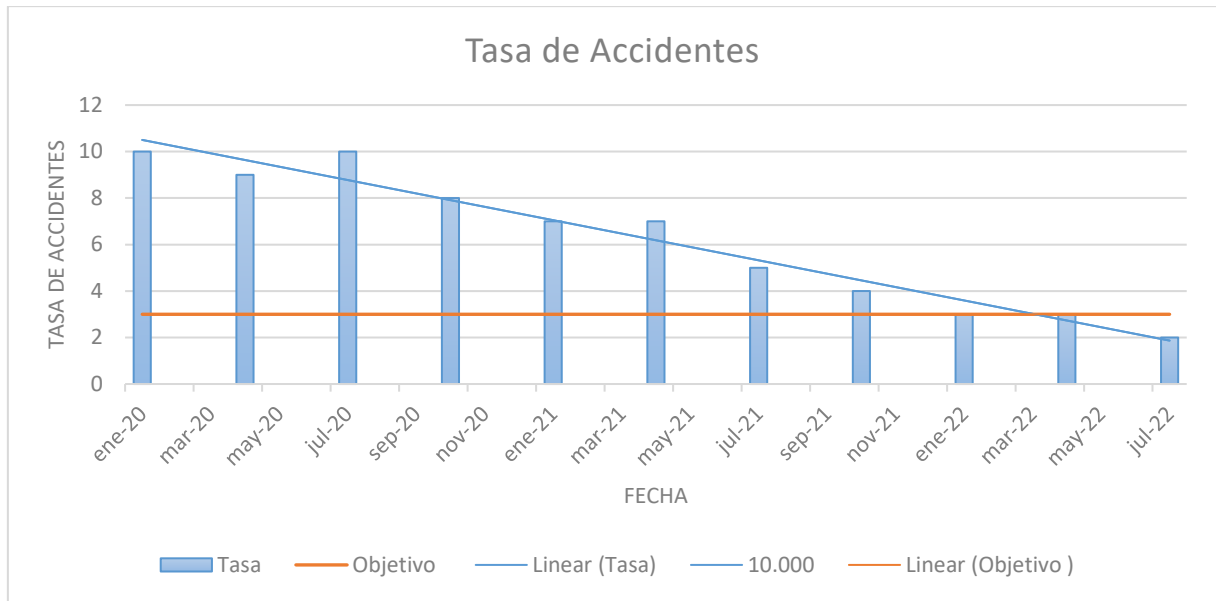
Este cuadro de mando refleja los datos más recientes, sin tener en cuenta el histórico, pudiendo ver si se están cumpliendo los objetivos marcados por la empresa. Para alimentar este cuadro de mando, se dispone de una tabla con la información histórica la cual se refleja en otra gráfica e indica si los indicadores van mejorando, se mantienen o empeoran.



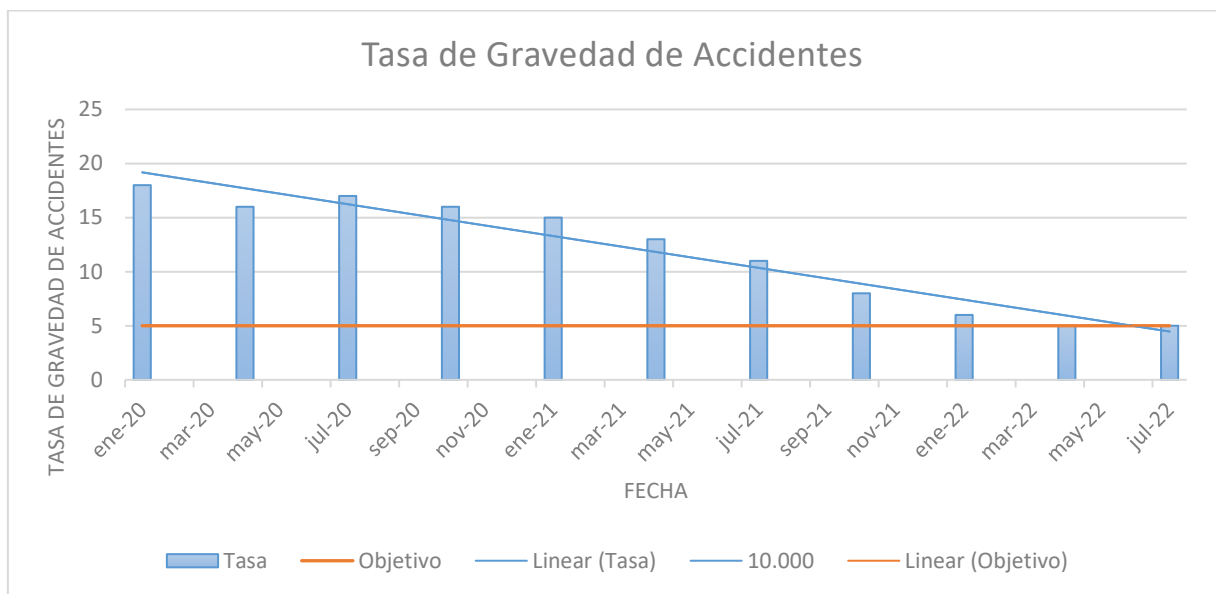
*Figura 30: Gastos en formación histórico (Elaboración propia)*



*Figura 31: Productividad histórico (Elaboración propia)*



*Figura 32: Tasa de Accidentes histórico (Elaboración propia)*



*Figura 33: Tasa de Gravedad de Accidentes histórico (Elaboración propia)*

La figura 30 refleja los gastos frente al tiempo, midiendo estos gastos en € (Euros). La figura 31 refleja la productividad frente al tiempo, midiendo esta productividad en millar de productos fabricados. La figura 32 refleja la tasa de accidentes frente al tiempo, midiendo esta tasa como el número de accidentes por trimestre. La figura 33 refleja la tasa de gravedad de accidentes frente al tiempo, midiendo esta tasa con un nivel de gravedad que oscila entre 0, sin accidente, a 25, accidente mortal.

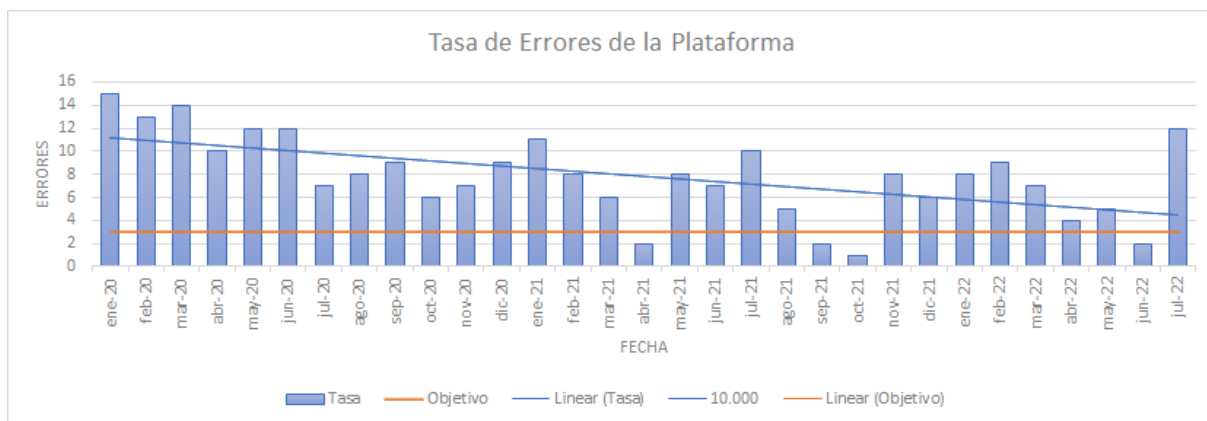


Figura 34: Tasa de Errores de la Plataforma histórico (Elaboración propia)

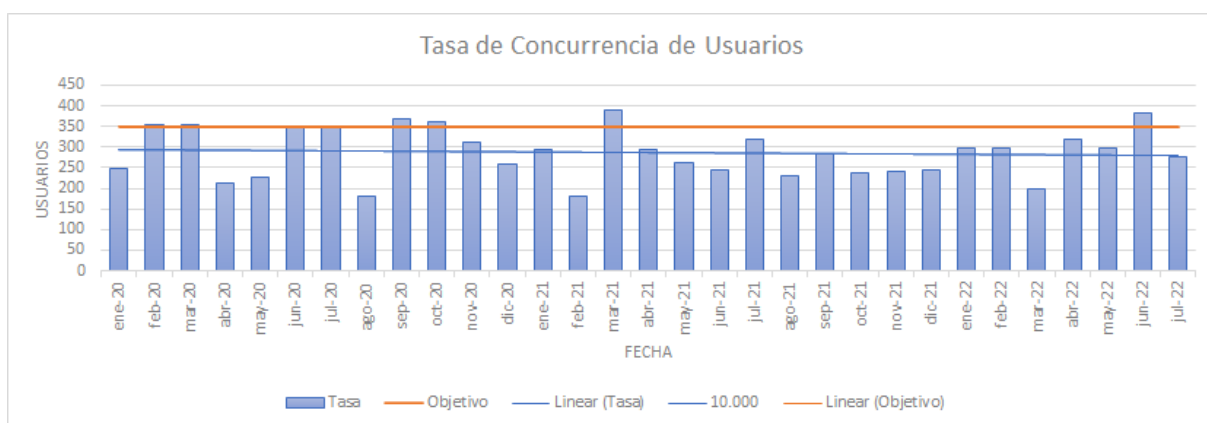


Figura 35: Tasa de Concurrencia de Usuarios histórico (Elaboración propia)

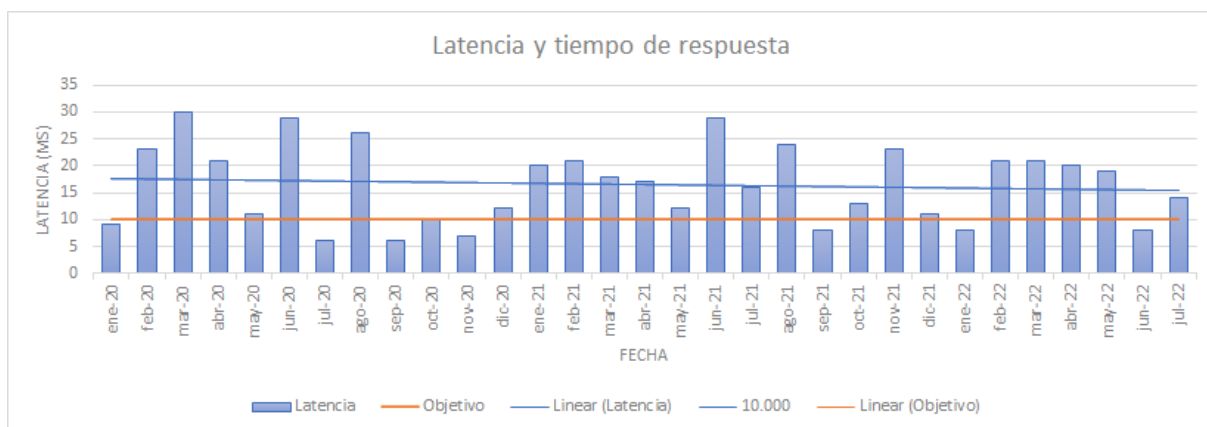


Figura 36: Latencia histórico (Elaboración propia)

La figura 34 refleja la tasa de errores de la plataforma frente al tiempo. La figura 35 refleja la el número máximo de usuarios conectados simultáneamente. La figura 36 representa la latencia en milisegundos.

En las figuras anteriores se muestran todos los datos que se van a extraer de la información histórica. En estas figuras es en las que se refleja si ha habido una mejora, se ha mantenido o ha empeorado cada uno de los indicadores. La barra naranja de todos los gráficos indica el valor objetivo que se busca alcanzar, y en el eje horizontal se encuentra la fecha, por lo que los gráficos podrían alargarse años.

Cada uno de los otros indicadores pueden verse igualmente expuestos en gráficos similares, pero lo fundamental es poder apreciar que la plataforma funciona, y los 4 gráficos anteriores son los que exponen cada uno de los KPIs principales.

#### 4.3.3. Costes y retorno de la inversión

Para poder determinar los costes de tener una plataforma así, junto con el retorno de la inversión esperado se deben determinar los siguientes factores:

- Primeramente, se requiere preferiblemente de un servidor en la nube, aunque puede funcionar en local. Este servidor debe tener capacidad para alojar toda la información de cada una de las empresas que requieran de la plataforma. Cuanta más información requiera la empresa tener colgada en el servidor de la nube, mayor será el coste de este servidor.
- Asociado al anterior, se requiere de un servidor cuyo proveedor pueda asegurar que se cumplen las medidas de ciberseguridad establecidas por la familia de las ISO 27000. Un servidor en la nube, a priori, no tiene mayor riesgo que un servidor local, y aunque se vayan a utilizar protocolos de encriptado y desencriptado, es de vital importancia el mantener alto el nivel de seguridad.
- Además del coste del servidor, el siguiente coste a tener en cuenta son las funcionalidades que requiera la plataforma para cada usuario. La plataforma requiere una adaptación mínima para cada una de las empresas, y a mayor información necesite ser mostrada en el dispositivo del usuario, mayor adaptación requerirá, por lo que tendrá un mayor coste también.
- Otro factor a tener en cuenta es el número de usuarios que se esperan en una misma empresa, y los distintos roles que pueda haber en la misma. Cada uno de los roles irá asociado a un nivel de personalización de la plataforma y a un nivel de permisos, pudiendo ver mayor o menor cantidad de información en función de este.
- A partir de tener los anteriores conceptos claros se requiere un personal que este cualificado y capacitado para llevar a cabo del desarrollo y posterior mantenimiento de una plataforma de tales dimensiones. Aunque sea un proyecto que pueda realizar incluso un equipo reducido, de dos personas, se requiere unos conocimientos que también tienen un coste.
- Además del conocimiento, la cantidad de horas que esas personas deberán de invertir primeramente para llevar a cabo el desarrollo, junto con los equipos que sean necesarios para el desarrollo.

Es por todo lo anterior que estimar un coste para una plataforma de tales dimensiones es complicado, es un coste que puede oscilar enormemente, dependiendo de todos y cada uno de los factores mencionados.

Para este caso, se plantea la plataforma más básica que se podría realizar, con:

- Número de usuarios inferior o igual a 10.
- Dos roles, usuario normal y administrador.
- Alquiler de un sector del servidor de la aplicación, capacidad aproximada de 250 GB, ampliable a 500Gb, capacidad más que suficiente para las pequeñas empresas.
- Coste de la adaptación de la plataforma al cliente.

Una plataforma con las características expuestas tendría un coste aproximado de 13.500€.

Además de este coste, hay un sobrecoste que se mantiene con el tiempo en caso de requerirlo, que no es estrictamente necesario, pero sí recomendado, el tener personal de servicio técnico y/o mantenimiento, para que, en caso de malfuncionamiento, error, problemas de seguridad, se encarguen de arreglarlo. Para las pequeñas empresas supondría un coste demasiado alto el tener gente para urgencias 24hrs, pero sí que podrían abonar cada vez que necesitaran servicios de mantenimiento, lo que supondría un coste aproximado de 500€ anuales.

A continuación, se expone la opción del extremo opuesto:

- Número de usuarios mayor o igual a 500.
- Diez o más roles, administrador, usuario normal, etc.
- Alquiler de un sector del servidor de la aplicación, capacidad aproximada de 2 TB, capacidad ampliable hasta 10TB.
- Coste de la adaptación de la plataforma al cliente.



Una plataforma de las características anteriores tendría un coste aproximado de 73.000€.

Como en el ejemplo anterior, existiría también un sobrecoste, pero para este caso sería necesario tener un servicio de mantenimiento y averías, lo que supondría un sobrecoste anual de 75.000€, tres turnos, una persona siempre fija con un sueldo de 25.000€ anuales brutos.

Para poder calcular el retorno de la inversión en cada uno de los casos se deberían de conocer el ahorro que generaría el uso de la plataforma, ya sea en formación, o en conseguir evitar accidentes. En el apartado 2.1 se expone que, en España de media anual, las pequeñas y medianas empresas invierten unos 350€ por empleado, y las grandes hasta 750€. Suponiendo que con el uso de la plataforma este coste se redujera en un 35%, una empresa pequeña como la del primer ejemplo, con menos de 10 trabajadores, podría tener un ahorro anual de 1.225€ aproximadamente, 122,5€ por trabajador. Por el contrario, una gran empresa como la del segundo ejemplo, podría tener un ahorro anual de 131.250€, lo que suponen 262,5€ por trabajador al año, contando que tenga 500 trabajadores.

Con los datos anteriores se concluye que:

- Para una pequeña empresa, con un número reducido de empleados, y estando esta, realizando gastos como en España, supondría un ahorro muy reducido, por lo que tendría un retorno de la inversión mayor a 19 años.
- Para una gran empresa, con un número de empleados superior a 500, y con unos gastos en formación como en España, supondría un gran ahorro anual, por lo que tendría un retorno de la inversión menor a 2 años.

## 5. Conclusiones y trabajos futuros

Tal y como se recoge en la introducción el principal problema que aborda este trabajo de final de máster es que la formación en las empresas, la cual es una de las claves de su funcionamiento, se sigue impartiendo de formas tradicionales y que han quedado atrás. Además de ello, estos métodos no incluyen tecnologías de última generación que podrían favorecer su impartición, ya que actualmente es muy lenta y costosa.

Gracias al desarrollo de la aplicación y el planteamiento de la aplicación como plataforma como servicio (PaaS) realizado en este TFM, dicho problema quedaría minimizado, consiguiendo reducir los gastos en formación tanto en medida de tiempo empleado para formar, como en materia económica.

Las principales contribuciones obtenidas tras la realización de este proyecto son las siguientes:

- 1.- Revisión conceptual y estudio de los actuales sistemas de realidad aumentada, así como sus usos más habituales, dentro y fuera de la industria.
- 2.- Diseño y creación de la primera versión de un ecosistema basado en la Realidad aumentada. Detallando en múltiples apartados cómo se ha escogido el software para realizar el desarrollo, y cómo, posteriormente se ha realizado. Así como algunos de los problemas que han surgido durante el desarrollo.
- 3.- Planteamiento y exposición de la adaptación de este ecosistema para ofrecerla al mercado como Plataforma como servicio (PaaS). Se han expuesto las ventajas de ofrecer la aplicación como PaaS, así como los KPIs que indicarían su correcto funcionamiento, los resultados y el retorno de la inversión esperado.

Teniendo en cuenta estos resultados, ha quedado demostrado en el desarrollo de este TFM que han sido alcanzados los objetivos específicos planteados ya que:

Ofrecer la suficiente información al usuario final para que pueda desarrollar el trabajo sin tener que consultar ningún documento ni a ningún compañero. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.2 de la memoria donde se refleja el

desarrollo de la aplicación y en el apartado 4.3 donde se expone la adaptación a PaaS, y según los resultados obtenidos, cada empresa que desee utilizar la plataforma podrá cargar tanta información como desee para que cada empleado tenga acceso total a ella.

Reducir costes económicos y temporales en formación para la empresa. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.3.3 de la memoria, y según los resultados obtenidos, tanto para empresas pequeñas como para empresas grandes supone un ahorro económico la incorporación de esta plataforma. Remarcar que, tal y como se expone en el apartado 4.3.3 de la memoria, el ahorro y la amortización es más notable en empresas con un mayor número de empleados y con mayor gasto en formación.

Diseñar el sistema para que pueda ser utilizado como un PaaS. Este objetivo se considera alcanzado ya que, tal y como queda reflejado en el apartado 4.3 de la memoria, y según los resultados obtenidos, ya sea en empresas pequeñas como en grandes es un modelo viable económicamente y supondría una mejora a nivel de producción.

Por consiguiente, se puede concluir que el objetivo principal, crear un ecosistema para la formación de empleados basado en la Realidad Aumentada que además sea escalable y extrapolable a otros entornos del ámbito industrial mediante una plataforma como servicio (PaaS), ha sido alcanzado tras la realización de este TFM.

#### **5.1.1. Líneas de trabajo futuras**

Finalmente, como perspectiva de futuro se podría reenfocar la plataforma combinando la realidad aumentada con la realidad virtual de la siguiente forma. Como concepto preliminar se plantearía adaptar la plataforma para que pudiera funcionar sobre unas gafas de realidad virtual, en las que la inmersión del usuario es completa. A estas gafas se les añadiría dispositivos para las manos, y lo que se buscaría es controlar maquinaria de las distintas líneas de forma remota en un entorno controlado, lo que supondría una reducción de los accidentes a casi el cero absoluto. De esta forma se interrelacionaría el mundo físico con el mundo virtual, consiguiendo una interacción con las distintas máquinas. Esto se podría combinar con un algoritmo de inteligencia artificial que fuera aprendiendo de la experiencia de lo que fueran haciendo los trabajadores, hasta que pudiera realizar funciones repetitivas de forma autónoma.

Con la anterior adaptación tanto el diseño realizado, como la posible futura línea de trabajo se podría incorporar en cualquier sector industrial que contenga operarios en constante trabajo con maquinaria. Estos sectores industriales manufactureros pueden ser, por ejemplo, industria de la automoción, fabricación de productos electrónicos y/o informáticos, industria de productos farmacéuticos, textil, industria química, etc.

## Referencias bibliográficas

- ARTool Kit X. (11 de Abril de 2018). *artoolkitX*. Obtenido de <http://www.artoolkitx.org/>
- Bin Zikria et al. (2019). Internet of Things (IoT) Operating Systems Management: Opportunities, Challenges, and Solution. *National Library of Medicine*, 10.
- Cabero Almenara et al. (2018). Blended learning y realidad aumentada: experiencias de diseño docente. *RIED. Revista Iberoamericana de Educación a Distancia*, 374.
- Cámara de Valencia. (2 de Febrero de 2021). *Realidad Virtual vs Realidad Aumentada: los conceptos clave*. Obtenido de <https://ticnegocios.camaravalencia.com/servicios/tendencias/realidad-virtual-vs-realidad-aumentada-los-conceptos-clave/>
- ControlSYS. (15 de Enero de 2022). *Las empresas españolas gastan en formación la mitad que las europeas*. Obtenido de <https://www.controlsyst.es/conocenos/noticias/114-ultimas-noticias/103-las-empresas-espanolas-gastan-en-formacion-la-mitad-que-las-europeas#:~:text=Ultimas%20noticias-,Las%20empresas%20espa%C3%B1olas%20gastan%20en%20formaci%C3%B3n%20la%20mitad%20que%20las>,
- CSA GROUP`. (3 de Julio de 2017). *REAPING THE BENEFITS OF INDUSTRIAL INTERNET OF THINGS (IIOT)*. Obtenido de <https://www.csagroup.org/>
- Estrada-Velasco et al. (2021). Revisión Sistemática de la Metodología Scrum para el Desarrollo de Software. *DOMINIO DE LAS CIENCIAS*, 14.
- Fundación Élogos. (2008). *Fundación Élogos*. Madrid: Fundación Élogos.
- Google LLC. (7 de Abril de 2020). *Getting started with Sceneform*. Obtenido de <https://developers.google.com/sceneform/develop/getting-started>
- Google LLC. (6 de Mayo de 2020). *Sceneform 1.15.0*. Obtenido de Sceneform overview: <https://developers.google.com/sceneform/develop/getting-started>
- Google LLC. (22 de Abril de 2022). *Glass Google*. Obtenido de <https://www.google.com/glass/start/>
- Gorisse, T. (12 de Mayo de 2022). *Sceneform-Android*. Obtenido de <https://github.com/SceneView/sceneform-android>
- Haghnegahdar et al. (2022). From IoT-based cloud manufacturing approach to intelligent additive manufacturing: Industrial Internet of Things - an overview. *International journal of advanced manufacturing technology*, 119.

- Hensen et al. (2020). *The Open Augmented Reality Teaching Book*. Michigan: Create and Code Augmented Reality.
- López-García et al. (2018). La realidad aumentada en la formación del profesorado. Una experiencia en las prácticas del Máster de Profesorado de Enseñanza Secundaria. *Revista Campus Virtuales*, 136.
- Marinescu, D. C. (2013). *Cloud Computing: Theory and Practice*. Florida: Elsevier Science & Technology.
- Martínez Olivar, N. (23 de Julio de 2020). Diseño de la interfaz de una aplicación móvil para realizar visitas turísticas con realidad aumentada (RA) en Teruel. *Diseño de la interfaz de una aplicación móvil para realizar visitas turísticas con realidad aumentada (RA) en Teruel*. Teruel, Teruel, España: Universidad Internacional de la Rioja.
- Microsoft Corporation. (14 de Mayo de 2022). *Microsoft Hololens 2*. Obtenido de <https://www.microsoft.com/en-us/hololens/hardware>
- Navarro et al. (2019). *Realidad virtual y Realidad aumentada. Desarrollo de aplicaciones*. Bogotá: Editorial Ra-ma.
- Observatorio de RRHH. (7 de Julio de 2020). *La importancia que tiene la formación para los empleados*. Obtenido de <https://www.observatoriorh.com/orh-posts/importancia-tiene-formacion-empleados.html#:~:text=Tambi%C3%A9n%20la%20formaci%C3%B3n%20incrementa%20la,de%20permanencia%20en%20la%20empresa>.
- Passmore, V. (7 de Junio de 2020). *CNC Control Panel 3D Model*. Obtenido de <https://sketchfab.com/3d-models/cnc-control-panel-3baed9ac58184563812f896612a45dc0>
- Schmalstieg et al. (2016). *Augmented Reality: Principles and Practice*. Hoboken: Addison-Wesley Professional.
- Stackscale. (4 de Agosto de 2021). *Stackscale*. Obtenido de Principales modelos de servicio cloud: IaaS, PaaS y SaaS: <https://www.stackscale.com/es/blog/modelos-de-servicio-cloud/>
- Statista. (31 de Agosto de 2021). *Android e iOS dominan el mercado de los smartphones*. Obtenido de <https://es.statista.com/grafico/18920/cuota-de-mercado-mundial-de-smartphones-por-sistema-operativo/#:~:text=Seg%C3%BAn%20datos%20de%20la%20consultora,Apple%20casi%20el%2016%25%20restante>.

The Drew Company. (3 de Diciembre de 2019). *Ventajas y desventajas de la metodología Scrum*. Obtenido de <https://blog.wearedrew.co/ventajas-y-desventajas-de-la-metodologia-scrum>

Vuforia Developer. (11 de Mayo de 2022). *Vuforia Developer Portal Engine*. Obtenido de <https://developer.vuforia.com/>

## Anexo A. Código implementado XML

Para la parte que el usuario ve, también conocida como desarrollo front end, de la aplicación desarrollada, se ha implementado utilizando el lenguaje XML. Este compone todo el conjunto de elementos que son visualizados por pantalla, el fondo, los botones, las ventanas desplegables, etc. A continuación, se expone el código XML de la pantalla de acceso, y seguidamente la pantalla principal, en la que se encuentra la mayoría de la programación en este lenguaje.

```
<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/UserText"
        android:layout_width="wrap_content"
        android:layout_height="48dp"
        android:layout_marginTop="175dp"
        android:ems="10"
        android:hint="User"
        android:inputType="textPersonName"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView" />

    <EditText
        android:id="@+id/PassText"
        android:layout_width="wrap_content"
        android:layout_height="48dp"
        android:layout_marginTop="20dp"
        android:ems="10"
        android:hint="Password"
```



```
        android:inputType="textPersonName"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toBottomOf="@+id/UserText" />
<TextView
    android:id="@+id/textView"

    android:layout_width="0dp"

    android:layout_height="70dp"

    android:background="#266597"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent" />
<Button
    android:id="@+id/LoginButton"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginTop="20dp"

    android:backgroundTint="#266597"

    android:text="LOGIN"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toBottomOf="@+id/PassText" />
<ImageView
    android:id="@+id/LogoUnir"

    android:layout_width="239dp"

    android:layout_height="87dp"

    app:layout_constraintBottom_toBottomOf="parent"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:srcCompat="@drawable/unirlogo" />
<TextView
    android:id="@+id/TopText"

    android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"

        android:layout_marginStart="20dp"

        android:text=" TFM LUIS ARBÓS"

        android:textSize="20sp"

        android:textStyle="bold"

        app:layout_constraintBottom_toBottomOf="@+id/textView"

        app:layout_constraintStart_toStartOf="@+id/textView"

        app:layout_constraintTop_toTopOf="parent" />
<TextView

    android:id="@+id/textFail"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginTop="45dp"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toBottomOf="@+id/LoginButton" />
</androidx.constraintlayout.widget.ConstraintLayout>
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:visibility="visible">

    <androidx.fragment.app.FragmentContainerView

        android:id="@+id/arFragment"

        android:name="com.google.ar.sceneform.ux.ArFragment"

        android:layout_width="match_parent"

        android:layout_height="match_parent" />

    <ImageButton

        android:id="@+id/engrButton"

        android:layout_width="80dp"

        android:layout_height="80dp"
```

```
    android:layout_alignParentStart="true"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="42dp"
    android:layout_marginBottom="27dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/qrButton"
    app:layout_constraintStart_toStartOf="parent"
    app:srcCompat="@drawable/engranaje_escalado2"
    tools:ignore="SpeakableTextPresentCheck" />
<ImageButton
    android:id="@+id/qrButton"
    android:layout_width="80dp"
    android:layout_height="80dp"
    android:layout_alignParentBottom="true"
    android:layout_marginStart="34dp"
    android:layout_marginEnd="34dp"
    android:layout_marginBottom="27dp"
    android:layout_toStartOf="@+id/doorButton"
    android:layout_toEndOf="@+id/engrButton"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/doorButton"
    app:srcCompat="@drawable/qr_escalado2"
    tools:ignore="SpeakableTextPresentCheck" />
<ImageButton
    android:id="@+id/doorButton"
    android:layout_width="80dp"
    android:layout_height="80dp"
    android:layout_alignParentEnd="true"
    android:layout_alignParentBottom="true"
    android:layout_marginEnd="42dp"
    android:layout_marginBottom="27dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
```

```
app:srcCompat="@drawable/puerta_escalado2"

tools:ignore="SpeakableTextPresentCheck" />

<FrameLayout

    android:id="@+id/qrFrame"

    android:layout_width="150dp"

    android:layout_height="150dp"

    android:layout_above="@+id/qrButton"

    android:layout_alignParentEnd="true"

    android:layout_marginEnd="121dp"

    android:layout_marginBottom="297dp"

    android:background="#F2000000"

    android:backgroundTint="#000000"

    app:layout_constraintBottom_toTopOf="@+id/qrButton"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    tools:visibility="gone">

    <TextView

        android:id="@+id/qrText"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_marginTop="-50dp"

        android:layout_marginBottom="10dp"

        android:text="Enfoque a la etiqueta para escanear."

        android:visibility="visible"

        app:layout_constraintBottom_toTopOf="@+id/qrFrame"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent" />

    <FrameLayout

        android:id="@+id/qrFrame2"

        android:layout_width="130dp"

        android:layout_height="130dp"

        android:layout_marginStart="10dp"
```

```
        android:layout_marginTop="10dp"

        android:background="#FDFCFD"

        android:visibility="visible"

        app:layout_constraintBottom_toTopOf="@+id/qrButton"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent">

</FrameLayout>

</FrameLayout>

<FrameLayout

    android:id="@+id/logoutFrame"

    android:layout_width="300dp"

    android:layout_height="200dp"

    android:layout_above="@+id/qrButton"

    android:layout_alignParentEnd="true"

    android:layout_marginEnd="46dp"

    android:layout_marginBottom="272dp"

    android:background="#CF9797"

    android:backgroundTint="#F1C2C2"

    app:layout_constraintBottom_toTopOf="@+id/qrButton"

    app:layout_constraintEnd_toEndOf="parent"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    tools:visibility="gone">

    <TextView

        android:id="@+id/logoutText"

        android:layout_width="200dp"

        android:layout_height="wrap_content"

        android:layout_marginStart="50dp"

        android:layout_marginTop="30dp"

        android:layout_marginEnd="80dp"

        android:layout_marginRight="80dp"

        android:text="¿Está seguro que desea cerrar sesión?"
```

```
        android:textAlignment="center"

        android:textSize="16sp"

        android:textStyle="bold"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent" />

<Button

    android:id="@+id/siButton"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginStart="45dp"

    android:layout_marginTop="120dp"

    android:layout_marginBottom="20dp"

    android:text="Si" />

<Button

    android:id="@+id/noButton"

    android:layout_width="wrap_content"

    android:layout_height="wrap_content"

    android:layout_marginStart="160dp"

    android:layout_marginTop="120dp"

    android:layout_marginBottom="20dp"

    android:text="No" />

</FrameLayout>

<FrameLayout

    android:id="@+id/settingsFrame"

    android:layout_width="300dp"

    android:layout_height="500dp"

    android:layout_above="@+id/qrButton"

    android:layout_alignParentEnd="true"

    android:layout_marginEnd="46dp"

    android:layout_marginBottom="122dp"

    android:background="#C9CDCBCB"

    app:layout_constraintBottom_toTopOf="@+id/qrButton"

    app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent"
tools:visibility="gone">
<EditText
    android:id="@+id/buttonColor"
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:layout_marginStart="236dp"
    android:layout_marginTop="224dp"
    android:background="#C13838"
    android:ems="10"
    android:inputType="number|text"
    android:textAlignment="center"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    tools:ignore="SpeakableTextPresentCheck" />
<EditText
    android:id="@+id/buttonIdioma"
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:layout_marginStart="236dp"
    android:layout_marginTop="299dp"
    android:ems="10"
    android:inputType="textShortMessage|number"
    android:text="ES"
    android:textAlignment="center"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
<EditText
    android:id="@+id/buttonIcono"
    android:layout_width="48dp"
    android:layout_height="48dp"
    android:layout_marginStart="236dp"
```

```
        android:layout_marginTop="149dp"

        android:ems="10"

        android:inputType="number|text"

        android:text="G"

        android:textAlignment="center"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent" />

<EditText

    android:id="@+id/buttonLetra"

    android:layout_width="48dp"

    android:layout_height="48dp"

    android:layout_marginStart="236dp"

    android:layout_marginTop="74dp"

    android:ems="10"

    android:inputType="number"

    android:text="14"

    android:textAlignment="center"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent" />

<ImageButton

    android:id="@+id/cerrarButton"

    android:layout_width="50dp"

    android:layout_height="50dp"

    android:layout_marginStart="250dp"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    app:srcCompat="@android:drawable/ic_delete"

    tools:ignore="SpeakableTextPresentCheck" />

<Button

    android:id="@+id/actButton"

    android:layout_width="260dp"

    android:layout_height="60dp"

    android:layout_marginStart="20dp"
```



```
        android:layout_marginTop="400dp"

        android:text="ACTUALIZACIÓN"

        app:layout_constraintBottom_toTopOf="@+id/textLetra"

        app:layout_constraintStart_toStartOf="parent" />
<TextView

        android:id="@+id/textIconos"

        android:layout_width="200dp"

        android:layout_height="45dp"

        android:layout_marginStart="20dp"

        android:layout_marginTop="150dp"

        android:background="#E45E6EC8"

        android:gravity="center"

        android:text="TAMAÑO ICONOS"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent" />
<TextView

        android:id="@+id/textLetra"

        android:layout_width="200dp"

        android:layout_height="45dp"

        android:layout_marginStart="20dp"

        android:layout_marginTop="75dp"

        android:background="#E45E6EC8"

        android:gravity="center"

        android:text="TAMAÑO LETRA"

        android:textAlignment="center"

        android:textAllCaps="false" />
<TextView

        android:id="@+id/textColor"

        android:layout_width="200dp"

        android:layout_height="45dp"

        android:layout_marginStart="20dp"

        android:layout_marginTop="225dp"

        android:background="#E45E6EC8"
```

```
        android:gravity="center"

        android:text="COLOR DE FONDO"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent"

        tools:layout_editor_absoluteY="30dp" />

<TextView

    android:id="@+id/textIdioma"

    android:layout_width="200dp"

    android:layout_height="45dp"

    android:layout_marginStart="20dp"

    android:layout_marginTop="300dp"

    android:background="#E45E6EC8"

    android:gravity="center"

    android:text="IDIOMA"

    app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toTopOf="parent"

    tools:layout_editor_absoluteY="30dp" />

</FrameLayout>

</RelativeLayout>
```

## Anexo B. Código implementado JAVA

Para la parte que el usuario no ve, también conocida como desarrollo back end, de la aplicación desarrollada, se ha implementado utilizando principalmente utilizando JAVA; pero en algunos momentos se ha requerido combinarlo con KOTLIN. En este Anexo se exponen algunas de las partes desarrolladas en lenguaje JAVA. Esta compone las cosas más básicas, las variables para los botones, las funciones que corresponden a cada uno, etc.

```
package com.example.tfm_luis_arbos;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private EditText Name;

    private EditText Password;

    private Button Login;

    private TextView Info;

    private int Counter = 5;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        Name = (EditText) findViewById(R.id.UserText);

        Password = (EditText) findViewById(R.id.PassText);

        Login = (Button) findViewById(R.id.LoginButton);

        Info = (TextView) findViewById(R.id.textFail);

        Info.setText("");

        getSupportActionBar().hide();

        Login.setOnClickListener(new View.OnClickListener() {

            @Override
```

```
        public void onClick(View view) {

            validate_users(Name.getText().toString(), Password.getText().toString());

        }});}

    private void validate_users(String user, String pass) {

        if ((user.equals("user")) && (pass.equals("pass"))) {

            Intent intent = new Intent(MainActivity.this, PantallaPrincipal.class);

            startActivity(intent);}

        else{

            Counter--;

            Info.setText("Incorrect user or password");

            Password.setText("");

            if(Counter ==0){

                Login.setEnabled(false);

                Info.setText("You have exceeded the maximum attempts (5). Try again in a few minutes.");

            }}}

package com.example.tfm_luis_arbos;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.FrameLayout;
import android.widget.ImageButton;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
public class PantallaPrincipal extends AppCompatActivity {
    private ImageButton Engr;
    private ImageButton Qr;
    private ImageButton Logout;
    private FrameLayout SettingsFrame;
    private EditText sizeText;
    private EditText sizeIcons;
    private EditText backColor;
    private EditText language;
    private ImageButton closeButton;
    private Button actButton;
    private TextView textLetra;
    private TextView textIcono;
    private TextView textColor;
    private TextView textIdioma;
    private int SettCounter = 0;
    private FrameLayout QrFrame;
    private FrameLayout QrFrame2;
    private TextView QrText;
    private int QrCounter = 0;
    private FrameLayout LogFrame;
    private TextView LogText;
    private Button LogYes;
    private Button LogNo;
    private int LogCounter = 0;
```

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.pantalla_principal);
    getSupportActionBar().hide();
    //Defining the settings options
    FrameLayout SettingsFrame = (FrameLayout) findViewById(R.id.settingsFrame);
    EditText sizeText = (EditText) findViewById(R.id.buttonLetra);
    EditText sizeIcons = (EditText) findViewById(R.id.buttonIcono);
    EditText backColor = (EditText) findViewById(R.id.buttonColor);
    EditText language = (EditText) findViewById(R.id.buttonIdioma);
    ImageButton closeButton = (ImageButton) findViewById(R.id.cerrarButton);
    Button actButton = (Button) findViewById(R.id.actButton);
    TextView textLetra = (TextView) findViewById(R.id.textLetra);
    TextView textIcono = (TextView) findViewById(R.id.textIconos);
    TextView textColor = (TextView) findViewById(R.id.textColor);
    TextView textIdioma = (TextView) findViewById(R.id.textIdioma);
    FrameLayout QrFrame = (FrameLayout) findViewById(R.id.qrFrame);
    FrameLayout QrFrame2 = (FrameLayout) findViewById(R.id.qrFrame2);
    TextView QrText = (TextView) findViewById(R.id.qrText);
    FrameLayout LogFrame = (FrameLayout) findViewById(R.id.logoutFrame);
    TextView LogText = (TextView) findViewById(R.id.logoutText);
    Button LogYes = (Button) findViewById(R.id.siButton);
    Button LogNo = (Button) findViewById(R.id.noButton);
    //Setting the setting options as invisible
    SettingsFrame.setVisibility(View.GONE);
    //Setting the qr option as invisible
    QrFrame.setVisibility(View.GONE);
    QrFrame2.setVisibility(View.GONE);
    QrText.setVisibility(View.GONE);
    //Setting the Logout option as invisible
    LogFrame.setVisibility(View.GONE);
    LogText.setVisibility(View.GONE);
    LogYes.setVisibility(View.GONE);
    LogNo.setVisibility(View.GONE);
    Engr.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            SettCounter++;
            if(SettCounter == 1){
                SettingsFrame.setVisibility(View.VISIBLE);

                closeButton.setOnClickListener(new View.OnClickListener(){
                    @Override
                    public void onClick(View view){
                        SettingsFrame.setVisibility(View.GONE);
                        SettCounter = 0;
                    }
                });
            }
            else{
                SettingsFrame.setVisibility(View.GONE);
                SettCounter = 0;
            }
        }
    });
    Qr.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            QrCounter++;
            if (QrCounter == 1){
                QrFrame.setVisibility(View.VISIBLE);
                QrFrame2.setVisibility(View.VISIBLE);
                QrText.setVisibility(View.VISIBLE);
            }
            else {
                QrFrame.setVisibility(View.GONE);
                QrFrame2.setVisibility(View.GONE);
                QrText.setVisibility(View.GONE);
                QrCounter = 0;
            }
        }
    });
}
```

```
Logout.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        LogCounter++;
        if(LogCounter == 1) {
            LogFrame.setVisibility(View.VISIBLE);
            LogText.setVisibility(View.VISIBLE);
            LogYes.setVisibility(View.VISIBLE);
            LogNo.setVisibility(View.VISIBLE);
            LogYes.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    Intent intent = new Intent(PantallaPrincipal.this, MainActivity.class);
                    startActivity(intent);
                }
            });
            LogNo.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                    LogFrame.setVisibility(View.GONE);
                    LogText.setVisibility(View.GONE);
                    LogYes.setVisibility(View.GONE);
                    LogNo.setVisibility(View.GONE);
                }
            });
        }
        else{
            LogFrame.setVisibility(View.GONE);
            LogText.setVisibility(View.GONE);
            LogYes.setVisibility(View.GONE);
            LogNo.setVisibility(View.GONE);
            LogCounter = 0;
        }
    }
});
package com.example.tfm_luis_arbos;
import android.net.Uri;
import android.os.Bundle;
import androidx.appcompat.app.AppCompatActivity;
import com.google.ar.sceneform.Node;
import com.google.ar.sceneform.SceneView;
import com.google.ar.sceneform.math.Quaternion;
import com.google.ar.sceneform.math.Vector3;
import com.google.ar.sceneform.rendering.ModelRenderable;
import java.util.concurrent.CompletableFuture;
import java.util.concurrent.ExecutionException;
public class MainActivity extends AppCompatActivity {
    private SceneView backgroundSceneView;
    private SceneView transparentSceneView;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        backgroundSceneView = findViewById(R.id.backgroundSceneView);
        transparentSceneView = findViewById(R.id.transparentSceneView);
        transparentSceneView.setTransparent(true);
        loadModels();
    }
    @Override
    protected void onResume() {
        super.onResume();
        try {
            backgroundSceneView.resume();
            transparentSceneView.resume();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    @Override
    protected void onPause() {
        super.onPause();
        backgroundSceneView.pause();
        transparentSceneView.pause();
    }
}
```

```
public void loadModels() {
    CompletableFuture<ModelRenderable> dragon = ModelRenderable
        .builder()
        .setSource(this
            , Uri.parse("models/dragon.glb"))
        .setIsFilamentGltf(true)
        .setAsyncLoadEnabled(true)
        .build();
    CompletableFuture<ModelRenderable> backdrop = ModelRenderable
        .builder()
        .setSource(this
            , Uri.parse("models/backdrop.glb"))
        .setIsFilamentGltf(true)
        .setAsyncLoadEnabled(true)
        .build();
    CompletableFuture.allOf(dragon, backdrop)
        .handle((ok, ex) -> {
            try {
                Node modelNode1 = new Node();
                modelNode1.setRenderable(dragon.get());
                modelNode1.setLocalScale(new Vector3(0.3f, 0.3f, 0.3f));
                modelNode1.setLocalRotation(Quaternion.multiply(
                    Quaternion.axisAngle(new Vector3(1f, 0f, 0f), 45),
                    Quaternion.axisAngle(new Vector3(0f, 1f, 0f), 75)));
                modelNode1.setLocalPosition(new Vector3(0f, 0f, -1.0f));
                backgroundSceneView.getScene().addChild(modelNode1);

                Node modelNode2 = new Node();
                modelNode2.setRenderable(backdrop.get());
                modelNode2.setLocalScale(new Vector3(0.3f, 0.3f, 0.3f));
                modelNode2.setLocalRotation(Quaternion.multiply(
                    Quaternion.axisAngle(new Vector3(1f, 0f, 0f), 45),
                    Quaternion.axisAngle(new Vector3(0f, 1f, 0f), 75)));
                modelNode2.setLocalPosition(new Vector3(0f, 0f, -1.0f));
                backgroundSceneView.getScene().addChild(modelNode2);

                Node modelNode3 = new Node();
                modelNode3.setRenderable(dragon.get());
                modelNode3.setLocalScale(new Vector3(0.3f, 0.3f, 0.3f));
                modelNode3.setLocalRotation(Quaternion.axisAngle(new Vector3(0f, 1f, 0f), 35));
                modelNode3.setLocalPosition(new Vector3(0f, 0f, -1.0f));
                transparentSceneView.getScene().addChild(modelNode3);

                Node modelNode4 = new Node();
                modelNode4.setRenderable(backdrop.get());
                modelNode4.setLocalScale(new Vector3(0.3f, 0.3f, 0.3f));
                modelNode4.setLocalRotation(Quaternion.axisAngle(new Vector3(0f, 1f, 0f), 35));
                modelNode4.setLocalPosition(new Vector3(0f, 0f, -1.0f));
                transparentSceneView.getScene().addChild(modelNode4);
            } catch (InterruptedException | ExecutionException ignore) {}
            return null;
        });
}
```