



Universidad de Jaén

Escuela Politécnica Superior de Jaén

PROTOTIPO PARA LA VISUALIZACIÓN DE ENTORNOS NATURALES BASADO EN LA COMBINACIÓN DE LAS TECNOLOGÍAS DE IMPRESIÓN 3D Y REALIDAD AUMENTADA

Autor: Elena Carmona Vallecillo

Grado: Ingeniería Informática

Directores: Juan Roberto Jiménez Pérez

David Jurado Rodríguez

Departamento de los directores: Informática

Fecha: 26/06/2024

Licencia CC



CREA



Universidad de Jaén

Departamento de Informática

Don Juan Roberto Jiménez Pérez y Don David Jurado Rodríguez, tutores del Trabajo Fin de Grado titulado: '**Prototipo para la visualización de entornos naturales basado en las tecnologías de impresión 3D y Realidad Aumentada**', que presenta Doña Elena Carmona Vallecillo, otorgan el visto bueno para su entrega y defensa en la Escuela Politécnica Superior de Jaén.

Jaén, junio de 2024

La alumna:

Elena Carmona Vallecillo

Los tutores:

Juan Roberto Jiménez Pérez
David Jurado Rodríguez

Agradecimientos

Me gustaría aprovechar esta sección para expresar mi más sincero agradecimiento a todas las personas que han hecho posible la realización de este Trabajo de Fin de Grado.

En primer lugar, me gustaría agradecer a mis tutores, D. Juan Roberto Jiménez Pérez y D. David Jurado Rodríguez, por su invaluable guía, apoyo y dedicación a lo largo de todo el proceso. Su conocimiento y orientación han sido fundamentales para el desarrollo de este trabajo.

También quiero extender mi agradecimiento a Daniel y Gonzalo, por su inestimable ayuda con el proceso de impresión 3D. Su experiencia y disposición para ayudarme en cada etapa de este proyecto han sido cruciales para el éxito del mismo.

Agradezco además a mi familia, en especial a mis padres Olga y Fernando, así como a mi hermana Olga, por su constante apoyo emocional y motivación. Su comprensión, escucha y apoyo incondicional desde el inicio han sido esenciales en todo este camino. Gracias por creer en mí y por ser mi pilar fundamental.

Finalmente, a mis compañeros de carrera, en especial a Lydia y Alba, quienes han estado a mi lado durante todos estos años de estudio, compartiendo experiencias, conocimientos y amistad. Gracias por hacer este proceso más ameno.

Muchas gracias a todos.

FICHA DEL TRABAJO FIN DE TÍTULO	
Titulación	Grado en Ingeniería Informática
Modalidad	Proyecto de Ingeniería
Especialidad (solo TFG)	Sin especialidad
Mención (solo TFG)	Sistemas Gráficos
Idioma	Español
Tipo	Específico
TFT en equipo	No
Autor/a	Elena Carmona Vallecillo
Fecha de asignación	22/11/2023
Descripción corta	En el contexto de la visualización 3D inmersiva de escenarios reales o simulados, nos podemos encontrar con dos enfoques diferentes, uno es la realidad virtual, en el que el usuario se introduce e interactúa en un mundo completamente virtual, y otro es la combinación de dos tecnologías que llevan bastante tiempo siendo ampliamente demandadas, la impresión 3D y la realidad aumentada. En este trabajo, se explora este segundo enfoque generando una maqueta 3D que representa un entorno natural, preparado para superponer mediante Realidad Aumentada información relevante del modelo 3D previamente generado.

NORMAS APLICADAS EN ESTE DOCUMENTO

LOCALES	
TFT-UJA:2017	Normativa de Trabajos Fin de Grado, Fin de Máster y otros Trabajos Fin de Título de la Universidad de Jaén (Normativa marco UJA aprobada en Consejo de Gobierno)
TFT-EPSJ:2017	Normativa sobre Trabajos Fin de Grado y Fin de Máster en la Escuela Politécnica Superior de Jaén (Normativa EPSJ aprobada en Junta de Escuela)
TFT-EPSJ	Criterios de evaluación y normas de estilo para TFG y TFM de la Escuela Politécnica Superior de Jaén
NACIONALES E INTERNACIONALES	
ISO 2145:1978	Documentación - Numeración de divisiones y subdivisiones en documentos escritos
UNE 50132:1994	Traducción de la ISO 2145
APA 6ª edición	Estilo de referencias y citas de APA (American Psychological Association)

NORMAS UTILIZADAS COMO BASE O REFERENCIA

NACIONALES	
UNE 157001:2014	Criterios generales para la elaboración formal de los documentos que constituyen un proyecto técnico
UNE 157801:2007	Criterios generales para la elaboración de proyectos de sistemas de información
<i>Estas normas se han utilizado como base o referencia para la inclusión de algunos contenidos y definiciones sobre elaboración de proyectos, entendiendo como proyecto la documentación consensuada entre una empresa y un cliente, que da lugar al perfeccionamiento de un contrato para la elaboración de una obra o la prestación de un servicio. Por consiguiente, no debe esperarse la aplicación de estas normas en cuanto a la completitud de los contenidos ni a la organización de los mismos.</i>	

Contenido

RESUMEN	14
ABSTRACT.....	15
1 INTRODUCCIÓN	16
1.1 Objetivos del trabajo	19
1.2 Antecedentes y estado del arte	20
1.3 Planteamiento del problema	24
1.3.1 Identificación y justificación del problema	24
1.3.2 Contexto del problema	25
1.4 Requisitos iniciales.....	26
1.4.1 Requisitos de facilidad de uso.....	27
1.4.2 Requisitos funcionales.....	28
1.4.3 Requisitos no funcionales.....	28
1.4.4 Requisitos de modelado e impresión 3D.....	29
1.5 Análisis de riesgos	30
1.5.1 Matriz de riesgos	30
1.6 Alcance.....	37
1.7 Hipótesis y restricciones	38
2 TECNOLOGÍAS Y HERRAMIENTAS	39
2.1 Software para Realidad aumentada	39
2.1.1 Unity.....	39
2.1.2 Vuforia Engine	42
2.1.3 Microsoft Visual Studio	45
2.2 Software para Modelado e Impresión 3D	46
2.2.1 Blender	47
2.2.2 Meshmixer	48
2.2.3 GIMP.....	49
2.2.4 PrusaSlicer	50
3 PLANIFICACIÓN	52
3.1 Metodología de desarrollo de software.....	52
3.1.1 Adaptación de la metodología incremental al proyecto	61
3.2 Planificación temporal.....	65
3.3 Presupuesto	69
3.3.1 Costes del software	70
3.3.2 Costes del hardware.....	70
3.3.3 Costes del personal.....	72
3.3.4 Coste total.....	73
4 DISEÑO INICIAL	74
4.1 Especificaciones del sistema	74
4.2 Análisis y diseño del sistema	76
4.2.1 Diseño arquitectónico del sistema.....	77
4.2.2 Diagramas de clases	77

4.2.3	Diagramas de casos de uso	78
4.2.4	Diagramas de secuencia	79
4.2.5	Diseño de la interfaz	85
5	DESARROLLO	89
5.1	Primera iteración	89
5.1.1	Modelado del Puente del Obispo	90
5.1.2	Modelado de la torre	92
5.1.3	Modelado del puente	93
5.1.4	Aplicación de texturas	94
5.1.5	Revisión y ajuste de los modelos	96
5.2	Segunda iteración	97
5.2.1	Configuración impresión 3D	98
5.2.2	Impresión 3D, pruebas y ajustes	105
5.3	Tercera iteración	106
5.3.1	Configuración del proyecto en Unity	107
5.3.2	Integración de Vuforia	108
5.3.3	Primeras pruebas de Vuforia	109
5.3.4	Diseño e implementación de la interfaz de usuario	111
5.4	Cuarta iteración	111
5.4.1	Generación de modelos objetivos	112
5.4.2	Pruebas con los modelos objetivos	118
5.4.3	Integración de los modelos en Unity	121
5.5	Quinta iteración	122
5.5.1	Diseño imágenes de los marcadores	123
5.5.2	Interacción usando marcadores	125
5.5.3	Interacción entre varios marcadores	127
5.5.4	Interacción con modelos 3D	129
5.5.5	Interacción entre marcadores y modelos 3D	133
5.6	Sexta iteración	135
5.6.1	Pruebas de usabilidad y funcionalidad	135
5.6.2	Diseño del logo de la aplicación	141
5.6.3	Implementación cambio de idioma	142
5.6.4	Implementación de mejoras	149
6	RESULTADOS OBTENIDOS	151
7	CONCLUSIONES Y TRABAJOS FUTUROS	159
8	APÉNDICES	162
8.1	Guía original del Trabajo Fin de Título	162
8.2	Instalación y configuración del sistema	164
8.3	Manual de usuario	164
9	DEFINICIONES Y ABREVIATURAS	168
9.1	Definiciones	168
9.2	Abreviaturas	169
10	BIBLIOGRAFÍA	171

Índice de ilustraciones

Ilustración 1: Gafas de Realidad Aumentada de Google	16
Ilustración 2: Tipos de materiales en la impresión 3D	17
Ilustración 3: Sensorama, primer simulador de realidad virtual de la historia	21
Ilustración 4: Uso de tabletas en la Realidad Aumentada	22
Ilustración 5: Charles W. Hull, inventor de la estereolitografía	22
Ilustración 6: Evolución y crecimiento de la impresión 3D	23
Ilustración 7: Tamaño del mercado global de plástico para impresión 3D	25
Ilustración 8: Matriz de riesgos [21]	30
Ilustración 9: Unity: Plataforma de desarrollo de videojuegos [7]	40
Ilustración 10: Uso de AR Foundation en Unity [26]	41
Ilustración 11: Vuforia Engine [22]	43
Ilustración 12: Uso de Vuforia Engine en Unity	44
Ilustración 13: Aplicación Vuforia Creator para iOS	44
Ilustración 14: Model Targets Test App para Android	45
Ilustración 15: Microsoft Visual Studio 2022	46
Ilustración 16: Herramienta de modelado Blender [31]	48
Ilustración 17: Autodesk Meshmixer	49
Ilustración 18: Programa de edición de imágenes GIMP	50
Ilustración 19: Herramienta de impresión 3D PrusaSlicer	51
Ilustración 20: Modelo del ciclo de vida en Cascada	53
Ilustración 21: Modelo del ciclo de vida del Prototipado	54
Ilustración 22: Modelo del ciclo de vida en Espiral	55
Ilustración 23: Modelo del ciclo de vida Incremental	56
Ilustración 24: Modelo del ciclo de vida RAD	57
Ilustración 25: Metodología XP o Programación Extrema	58
Ilustración 26: Proceso y roles de Scrum	59
Ilustración 27: Tablero Kanban	59
Ilustración 28: Diagrama de Gantt	69
Ilustración 29: Conjunto de clases	78
Ilustración 30: Diagrama de casos de uso	79
Ilustración 31: Diagrama de secuencia Visualizar Instrucciones	80
Ilustración 32: Diagrama de secuencia Seleccionar modelos 3D	81
Ilustración 33: Diagrama de secuencia Interacción con marcadores	82
Ilustración 34: Diagrama de secuencia Visualización de información adicional	83
Ilustración 35: Diagrama de secuencia Interacción entre modelos y marcadores	84
Ilustración 36: Diagrama de secuencia Cambiar idioma de la aplicación	85
Ilustración 37: Interfaz menú principal	86
Ilustración 38: Pantalla para cambiar el idioma	86
Ilustración 39: Interfaz de las instrucciones	87
Ilustración 40: Interfaz principal de la aplicación	87
Ilustración 41: Interfaz principal con panel informativo	88
Ilustración 42: Modelo original Puente del Obispo	90

Ilustración 43: Fachada de la torre del Puente del Obispo	91
Ilustración 44: Modelo 3D de la torre por delante	92
Ilustración 45: Modelo 3D de la torre por detrás	93
Ilustración 46: Modelo 3D recortado del puente.....	93
Ilustración 47: Modelo Puente del Obispo sin texturas	94
Ilustración 48: Modelo Puente del Obispo con texturas	95
Ilustración 49: Textura en GIMP sin escalar	95
Ilustración 50: Textura en GIMP escalada	96
Ilustración 51: Modelo del Puente del Obispo en formato GLB	97
Ilustración 52: Modelo del Puente del Obispo en formato STL	98
Ilustración 53: Modelo Puente del Obispo en PrusaSlicer	98
Ilustración 54: Configuración Capas y perímetros de PrusaSlicer	100
Ilustración 55: Configuración Relleno de PrusaSlicer	101
Ilustración 56: Configuración Material de soporte de PrusaSlicer.....	102
Ilustración 57: Configuración Velocidad de PrusaSlicer.....	103
Ilustración 58: Configuración del filamento de PrusaSlicer	104
Ilustración 59: Configuración de impresión PrusaSlicer.....	104
Ilustración 60: Laminación y tiempos de impresión en PrusaSlicer	105
Ilustración 61: Primera prueba de impresión.....	106
Ilustración 62: Captura de Unity Hub	107
Ilustración 63: Licencias de Vuforia.....	108
Ilustración 64: Configuración de Vuforia en Unity	109
Ilustración 65: Captura del componente Image Target	110
Ilustración 66: Ejemplo de un cubo 3D sobre un Image Target	111
Ilustración 67: Pestaña Vector hacia arriba del modelo de MTG	113
Ilustración 68: Pestaña Unidades del modelo de MTG	114
Ilustración 69: Pestaña Esquema de color de MTG	114
Ilustración 70: Pestaña Complejidad.....	115
Ilustración 71: Pestaña Optimizar seguimiento en MTG	116
Ilustración 72: Pestaña Vistas de guía de MTG	116
Ilustración 73: Pestaña Vistas de guía avanzadas de MTG.....	117
Ilustración 74: Vista de guía de cúpula 360º	117
Ilustración 75: Archivos base de datos de MTG.....	118
Ilustración 76: Objeto detectado en Model Targets Test App	119
Ilustración 77: Captura de la aplicación Model Targets Test	119
Ilustración 78: Menú de opciones de Model Targets Test App	120
Ilustración 79: Captura de las bases de datos de Vuforia.....	121
Ilustración 80: Prueba de visualización de un modelo objetivo	121
Ilustración 81: Model Target con modelo 3D original	122
Ilustración 82: Código QR que actúa de marcador	123
Ilustración 83: Imagen del marcador de la torre	124
Ilustración 84: Imagen del marcador del puente	124
Ilustración 85: Imágenes objetivo en Vuforia	125
Ilustración 86: Marcador y modelo de la torre	126
Ilustración 87: Marcador y modelo del puente	126
Ilustración 88: Ejemplo interacción con dos marcadores	128

Ilustración 89: Box Colliders asociados al modelo del Puente del Obispo	129
Ilustración 90: Panel informativo del Puente del Obispo	130
Ilustración 91: Panel informativo sobre la torre del Puente del Obispo.....	130
Ilustración 92: Panel informativo del entorno del Puente del Obispo	131
Ilustración 93: Modelo 3D de un barco.....	132
Ilustración 94: Modelo del barco azul	132
Ilustración 95: Captura botones de los modelos	133
Ilustración 96: Efecto río sobre el puente	134
Ilustración 97: Modelo del barco rosa	135
Ilustración 98: Porcentaje de la edad de los usuarios	139
Ilustración 99: Porcentaje del nivel de conocimientos informáticos	140
Ilustración 100: Porcentaje de la calificación de facilidad de uso de la interfaz	140
Ilustración 101: Porcentaje de la satisfacción general de la aplicación	141
Ilustración 102: Imagen del logotipo de la aplicación	141
Ilustración 103: Imagen del logotipo y nombre de la aplicación	142
Ilustración 104: Ajustes Player del proyecto de Unity	142
Ilustración 105: Ejemplo tabla de localización tipo String	143
Ilustración 106: Panel de cambio de idioma.....	149
Ilustración 107: Modelo con Canvas integrados	149
Ilustración 108: Instrucciones para interactuar con los modelos de los marcadores	150
Ilustración 109: Primer modelo impreso del Puente del Obispo	151
Ilustración 110: Modelo impreso del barco.....	152
Ilustración 111: Segundo modelo impreso del Puente del Obispo.....	152
Ilustración 112: Marcador y modelo 3D de la torre	153
Ilustración 113: Marcador y modelo 3D del puente aislado	153
Ilustración 114: Modelo obtenido al interactuar con dos marcadores	154
Ilustración 115: Maqueta y modelo 3D del Puente del Obispo reducido.....	154
Ilustración 116: Modelo reducido con efecto de agua.....	155
Ilustración 117: Interacción con el modelo del barco	155
Ilustración 118: Interacción entre el barco y marcador	156
Ilustración 119: Maqueta y modelo 3D del Puente del Obispo mejorado	156
Ilustración 120: Panel informativo con datos del Puente del Obispo	157
Ilustración 121: Panel informativo con datos sobre el entorno del Puente del Obispo	157
Ilustración 122: Panel informativo con datos sobre la torre del Puente del Obispo	158
Ilustración 123: Propuesta original del TFG	163
Ilustración 124: Pantalla del menú de la aplicación	165
Ilustración 125: Pantalla cambio de idioma de la aplicación	165
Ilustración 126: Pantalla instrucciones de uso de la aplicación	166
Ilustración 127: Pantalla principal de la aplicación	166
Ilustración 128: Pantalla principal con menú de modelos	167

Índice de tablas

Tabla 1: Tabla de posibles problemas y sus riesgos	35
Tabla 2: Riesgos y planes de contingencia.	37
Tabla 3: Comparación entre metodologías ágiles y tradicionales [41]	60
Tabla 4: Tabla de costes software	70
Tabla 5: Tabla de costes hardware	71
Tabla 6: Tabla de costes del personal	73
Tabla 7: Tabla de costes finales.....	73
Tabla 8: Respuestas de la encuesta del primer usuario	137
Tabla 9: Respuestas de la encuesta del segundo usuario.....	137
Tabla 10: Respuestas de la encuesta del tercer usuario	138
Tabla 11: Respuestas de la encuesta del cuarto usuario	139
Tabla 12: Tabla de traducciones	148

RESUMEN

En el contexto de la Ingeniería Informática y la visualización 3D inmersiva de escenarios, en el presente Trabajo de Fin de Grado se propone investigar y desarrollar un enfoque que resulta de la combinación de dos tecnologías muy demandadas actualmente, la impresión 3D y la Realidad Aumentada. Mientras que la realidad virtual permite al usuario sumergirse de forma completa en un entorno virtual, este trabajo se centra en la materialización física de modelos 3D mediante el desarrollo de una maqueta tridimensional que representa un entorno natural.

Unos de los principales motivos de combinar la Realidad Aumentada con la impresión 3D es poder explorar las posibilidades de enriquecer la experiencia del usuario al superponer sobre la maqueta 3D información relevante a través de la Realidad Aumentada. Así, la maqueta se convierte en un punto de convergencia entre lo digital y lo tangible, ofreciendo una representación física del entorno natural a la vez que permite la integración de datos adicionales de una manera inmersiva.

El principal objetivo de este proyecto consiste en obtener un prototipo que permita visualizar los detalles de un entorno 3D mediante la combinación de una maqueta de dicho entorno junto con una herramienta de Realidad Aumentada.

Este trabajo abarca desde un repaso bibliográfico detallando tanto los procesos de impresión 3D como los de Realidad Aumentada, pasando por la selección adecuada de la impresora 3D, así como de la herramienta de desarrollo de Realidad Aumentada, hasta el análisis y especificación de los requisitos del sistema. También se destacan otras etapas importantes para el desarrollo de este proyecto como la planificación y presupuesto, junto con el diseño del sistema, implementación y pruebas.

En conclusión, este Trabajo de Fin de Grado busca la combinación de la Realidad Aumentada junto con la impresión 3D para crear una maqueta 3D de un entorno natural que pueda ser ampliada con información adicional de manera virtual. De esta manera, se espera proporcionar a los usuarios una visión más detallada de entornos rurales, permitiendo así su exposición e interacción de una manera más participativa e inmersiva.

ABSTRACT

In the context of Computer Engineering and immersive 3D visualization of scenarios, this Final Degree Project proposes to investigate and develop an approach that results from the combination of two highly demanded technologies today, 3D printing and augmented reality. While virtual reality allows the user to fully immerse themselves in a virtual environment, this work focuses on the physical materialization of 3D models through the development of a three-dimensional model that represents a natural environment.

One of the main reasons for combining augmented reality with 3D printing is to explore the possibilities of enriching the user experience by overlaying relevant information on the 3D model through augmented reality. Thus, the model becomes a convergence point between the digital and the tangible, offering a physical representation of the natural environment while allowing the integration of additional data in an immersive way.

The main objective of this project is to obtain a prototype that allows visualizing the details of a 3D environment by combining a model of that environment with an augmented reality tool.

This work ranges from a bibliographic review detailing both the 3D printing and augmented reality processes, through to the appropriate selection of the 3D printer, as well as the augmented reality development tool, to the analysis and specification of the system requirements. Other important stages in the project's development, such as planning and budgeting, along with system design, implementation, and testing, are also highlighted.

In conclusion, this Final Degree Project aims to combine augmented reality with 3D printing to create a 3D model of a natural environment that can be augmented with additional virtual information. In this way, it is expected to provide users with a more detailed view of rural environments, allowing their exposure and interaction in a more participative and immersive way.

1 INTRODUCCIÓN

Hoy en día, el avance tecnológico ha propiciado la creación de diferentes experiencias inmersivas que superan las limitaciones de la realidad física y tangible. Dentro de este aspecto, la visualización 3D surge como un campo bastante prometedor a la vez que fascinante, ofreciendo la posibilidad de modelar y explorar diversos escenarios tanto reales como simulados de una forma totalmente realista. En este contexto, destacan dos enfoques principales, por un lado, la realidad virtual, que ofrece la posibilidad de que los usuarios puedan sumergirse en mundos completamente virtuales y, por otro lado, la combinación de dos tecnologías que cada vez son más demandadas y despiertan un gran interés actualmente: la Realidad Aumentada y la impresión 3D.

La Realidad Aumentada (RA) es el término que se utiliza para describir al conjunto de tecnologías que permiten que un usuario visualice parte del mundo real a través de un dispositivo tecnológico con información gráfica añadida por este [1]. Como su propio nombre indica, la Realidad Aumentada [2] amplía o “aumenta” la percepción del mundo real al superponer información digital en el entorno real o físico. Esto se consigue a través de dispositivos electrónicos como dispositivos móviles, tabletas o gafas inteligentes (Ilustración 1) entre otros, que permiten a los usuarios visualizar elementos virtuales integrados en el mundo que les rodea.



Ilustración 1: Gafas de Realidad Aumentada de Google

El funcionamiento básico de la Realidad Aumentada implica la utilización de sensores y cámaras para poder capturar la imagen del entorno físico en tiempo real.

A continuación, se procesa esta información visual y se superponen elementos virtuales adicionales, como texto, gráficos o animaciones, que se integran con el entorno físico real. Como resultado se obtiene una experiencia combinada que ofrece a los usuarios una visión mejorada y contextualizada del mundo que les rodea.

La Realidad Aumentada tiene diversas aplicaciones en diferentes áreas, como en el campo de la medicina, en el diseño y producción, en el campo del entretenimiento, en el ámbito educativo, entre otros [3].

Por otro lado, la impresión 3D, también conocida como fabricación aditiva, es un avance muy importante hoy en día donde un objeto tridimensional es creado mediante la superposición de capas sucesivas de material a partir de un modelo digital [4]. Inicialmente pensada como una herramienta para el prototipado rápido en la industria, la impresión 3D ha experimentado una evolución exponencial, extendiéndose a una gran variedad de sectores y distintas aplicaciones.

La impresión 3D es muy versátil gracias a su capacidad de trabajar con distintos tipos de materiales, desde plásticos y metales hasta cerámicas y resinas, por lo que es una tecnología que se adapta a múltiples necesidades (Ilustración 2).



Ilustración 2: Tipos de materiales en la impresión 3D

Las impresoras 3D se pueden clasificar en domésticas e industriales según su uso y prestaciones, siendo versátiles en aplicaciones como la confección de joyas, vestimenta, muebles y prótesis. Existe una amplia gama de sectores en los que se

aplica la impresión 3D, como en el diseño, la medicina, la arquitectura o la industria en general. Además, la tecnología de impresión 3D permite la fabricación de productos personalizados que se ajustan a las distintas necesidades de los usuarios [5].

Trabajar con procesos de impresión 3D y Realidad Aumentada demuestra cómo la tecnología está evolucionando a un ritmo acelerado [6]. Uniendo la Realidad Aumentada y la impresión 3D, logramos crear una interacción entre el mundo físico y el digital, es decir, conseguimos mezclar lo que vemos y tocamos con elementos que desarrollamos en un ordenador. Esto nos permite superponer información digital generada por computadora sobre objetos físicos y reales creados mediante impresión 3D, abriendo nuevas posibilidades en diversos campos.

En este Trabajo de Fin de Grado se aplica la combinación de estas dos tecnologías, Realidad Aumentada e impresión 3D, con el motivo de desarrollar un prototipo que permita visualizar los detalles de un entorno natural en 3D. Para conseguir esto, se va a modelar y diseñar una maqueta 3D del terreno del entorno natural elegido para después imprimirlo en una impresora 3D, a su vez, se va a desarrollar una aplicación de Realidad Aumentada a través de la herramienta Unity [7] que va a permitir añadir detalles e información adicional de forma digital a dicha maqueta 3D. Para probar el prototipo y la aplicación de Realidad Aumentada con usuarios reales para evaluar su usabilidad se debe utilizar un dispositivo electrónico como un teléfono móvil o una tableta.

En este informe se describe el procedimiento seguido para desarrollar el prototipo. En primer lugar, se realiza una introducción en la que se tratan los objetivos, los requisitos iniciales, los riesgos, el alcance y las restricciones del proyecto ([Sección 1](#)). A continuación, se desarrollan las tecnologías y herramientas utilizadas en el desarrollo del trabajo ([Sección 2](#)). Además, se lleva a cabo una planificación del proyecto ([Sección 3](#)), que incluye la elección de la metodología de desarrollo y una estimación de recursos y presupuesto. Posteriormente, se realiza el análisis y diseño de los elementos involucrados tanto en el proceso de impresión 3D como en el de Realidad Aumentada ([Sección 4](#)). Después, se expone el desarrollo del prototipo ([Sección 5](#)), se realizan las pruebas con los usuarios y, por último, se analizan los resultados obtenidos ([Sección 6](#)) y se obtienen las conclusiones oportunas ([Sección 7](#)).

1.1 Objetivos del trabajo

El objetivo principal de este Trabajo de Fin de Grado es la obtención de un prototipo que permita una visualización detallada de un entorno natural tridimensional mediante la combinación de una maqueta física de dicho entorno, generada con una impresora 3D, y una herramienta de Realidad Aumentada. Con esto se busca proporcionar una experiencia inmersiva y enriquecida al usuario, superponiendo información digital relevante sobre la maqueta física, lo que contribuirá a la comprensión y exploración más profunda del entorno representado.

Además del objetivo principal, es necesario definir unos objetivos específicos que se quieren alcanzar en este proyecto y que derivan del objetivo principal ya detallado. Estos objetivos específicos son los siguientes:

- **Identificar las características necesarias de la impresora 3D y selección de la más adecuada para la maqueta:** Este objetivo implica analizar y seleccionar las características esenciales que debe tener la impresora 3D necesaria para la creación de la maqueta tridimensional. Se buscará la impresora 3D más adecuada y que cumpla con los requisitos específicos del proyecto.
- **Preparar los modelos 3D del terreno para que puedan ser impresos:** Se llevará a cabo la preparación de los modelos 3D del terreno, asegurándose que estén listos para su impresión. Este paso implica la optimización y adaptación de los modelos digitales para su posterior materialización en la maqueta física.
- **Preparar un modelo 3D de la zona seleccionando la información que se va a proporcionar mediante Realidad Aumentada:** La tarea de selección y preparación de un modelo 3D de la zona incluye la identificación y elección de la información que será proporcionada mediante Realidad Aumentada. Esto garantiza la relevancia y utilidad de los detalles superpuestos en la experiencia de usuario.
- **Desarrollar una herramienta de Realidad Aumentada que permita visualizar la maqueta 3D con información enriquecida procedente del modelo 3D:** Se procederá al desarrollo de una herramienta de Realidad Aumentada que permita la visualización de la maqueta 3D con

información adicional e importante del modelo 3D. Esto involucra la programación y diseño de la interfaz para garantizar una integración efectiva y una experiencia de usuario fluida.

- **Probar el prototipo con usuarios reales que permita evaluar su usabilidad:** Este último objetivo implica la evaluación práctica del prototipo con usuarios reales. Se llevarán a cabo distintas pruebas que permitan analizar la usabilidad del sistema, recopilando comentarios y datos que orienten posibles mejoras y ajustes en el diseño y la funcionalidad del prototipo.

En conjunto, estos objetivos se alinean para lograr la creación exitosa de un prototipo que integre la impresión 3D y la Realidad Aumentada, proporcionando una herramienta innovadora y funcional que no solo visualice, sino también enriquezca la experiencia de explorar un entorno natural tridimensional.

1.2 Antecedentes y estado del arte

En esta sección, se lleva a cabo un análisis de los elementos significativos que han ocurrido en el pasado y del conocimiento previo existente que tienen influencia en este trabajo. Este apartado también resume la historia y el contexto del tema tratado para conseguir una comprensión más profunda del mismo.

Además, se proporcionará información sobre el estado del arte de las disciplinas implicadas, haciendo referencia a los hechos más relevantes de las alternativas estudiadas.

El origen de la Realidad Aumentada (RA) se remonta a 1901 [8] cuando el escritor Frank Baum imaginó por primera vez unas gafas electrónicas para mostrar información adicional sobre las personas. La implementación tecnológica inicial de la RA ocurrió en 1957 con Morton Heilig y su Sensorama (Ilustración 3) [9], que ofrecía una experiencia multisensorial con elementos visuales, sonoros y olfativos. En 1973, Myron W. Krueger creó la primera instalación de RA utilizando cámaras de vídeo y proyección. Durante los años 90, aplicaciones de RA comenzaron a abordar problemas en la industria y el diseño, siendo Tom Caudell quien dio forma al término “Realidad Aumentada”.



Ilustración 3: Sensorama, primer simulador de realidad virtual de la historia

En los últimos años, la Realidad Aumentada ha experimentado un notable crecimiento, impulsado por el crecimiento de smartphones y tabletas que facilitan la experiencia de RA en entornos móviles [10]. Aunque las expectativas generadas en el Hype Cycle de Gartner [11] han superado los desarrollos actuales, la capacidad de acceder a información contextualizada, de manera móvil y con baja intrusión en las tareas de los usuarios, asegura el éxito de la RA en diversos sectores. Este éxito se debe, en parte, a la posibilidad de que los usuarios interactúen en tiempo real desde sus lugares de trabajo con información relevante para sus tareas, como órdenes de trabajo, planificación, diseño o revisión de mantenimiento.

A pesar de su concepción hace décadas, la RA sigue siendo una tecnología emergente, evolucionando con avances en computación y visualización. Empresas especializadas en Realidad Aumentada aprovechan su enorme potencial en distintos sectores. El uso de tabletas en la Realidad Aumentada explora la viabilidad de emplear RA y tabletas digitales táctiles (Ilustración 4) para manipular modelos digitales tridimensionales, simulando la experiencia de trabajar con modelos físicos. El objetivo es crear un entorno que fomente la comprensión del espacio tridimensional [12].



Ilustración 4: Uso de tabletas en la Realidad Aumentada

Por otro lado, la historia de la impresión 3D tiene sus raíces en 1976 con la invención de la impresora de inyección de tinta. En 1984, se realizaron adaptaciones para transformar la tecnología hacia la impresión con materiales. Durante las últimas décadas, la impresión 3D ha experimentado una gran variedad de aplicaciones en distintas industrias. En 1981, en Japón, el Dr. Hideo Kodama presentó la primera solicitud de patente para un dispositivo de impresión 3D, utilizando métodos de fabricación aditiva con resina fotosensible. El cambio significativo ocurrió en 1984 cuando Charles Hull [13] inventó la estereolitografía (Ilustración 5), una técnica que permite la creación de objetos tridimensionales a partir de datos digitales, revolucionando la impresión 3D al ofrecer la capacidad de probar diseños antes de la fabricación definitiva [14].

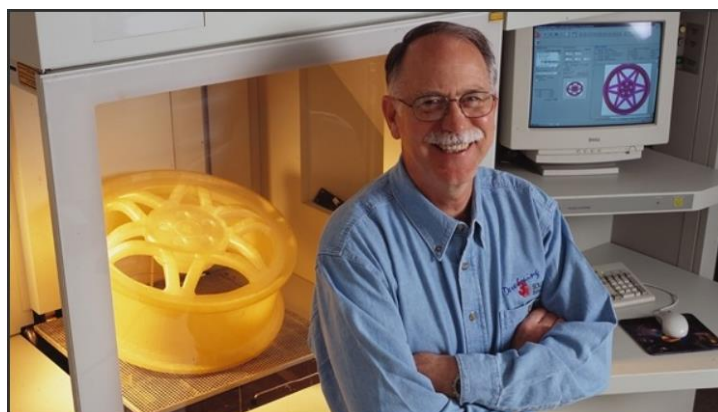


Ilustración 5: Charles W. Hull, inventor de la estereolitografía

En las últimas tres décadas, las impresoras 3D han avanzado significativamente, expandiéndose a diversas áreas de investigación. Inicialmente limitada a plásticos simples, la tecnología FDM (Fused Deposition Modeling) se ha diversificado para permitir el uso de materiales complejos y la combinación de diferentes polímeros. La impresión 3D ha encontrado aplicaciones en muchos sectores como en la industria textil, en la ingeniería o en la electrónica, entre muchos [15].

Actualmente, el mercado de la impresión 3D experimenta un crecimiento notable y atraviesa una fase de transición histórica. En abril de 2012, la fusión de Stratasys y Object [16] creó la principal empresa mundial de impresión 3D. Los procesos de impresión se han perfeccionado, la variedad de materiales disponibles ha aumentado y los precios de las máquinas han disminuido significativamente [17]. La impresión 3D ha experimentado un crecimiento exponencial (Ilustración 6), pasando de ser utilizado en el ámbito industrial y prototipado rápido principalmente a una herramienta accesible para consumidores y pequeñas empresas.

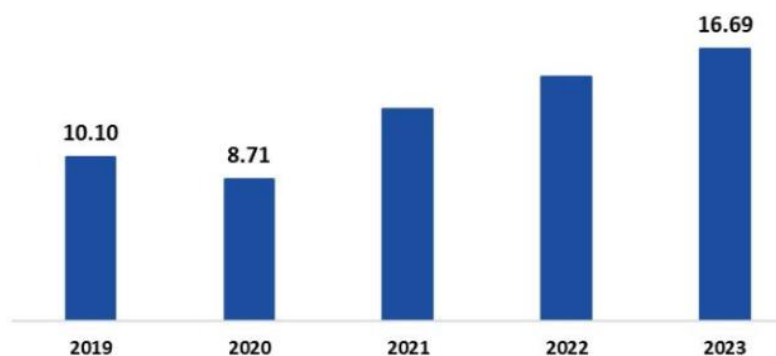


Ilustración 6: Evolución y crecimiento de la impresión 3D

En la convergencia de la Realidad Aumentada y la impresión 3D, proyectos anteriores han explorado la visualización 3D inmersiva, pero pocos han abordado específicamente la combinación de una maqueta física con Realidad Aumentada. Investigaciones en campos como la arquitectura, la educación y la medicina han demostrado el potencial de esta combinación para mejorar la comprensión, la interactividad y la toma de decisiones.

Entender la evolución y las tendencias actuales en estas disciplinas es esencial para contextualizar y justificar la elección de enfoques y tecnologías en el desarrollo

del este trabajo. El análisis de antecedentes y estado del arte proporciona una base sólida para la construcción de un prototipo innovador y relevante en el contexto actual de la visualización 3D inmersiva.

1.3 Planteamiento del problema

En esta sección se desarrolla el planteamiento del problema, centrándose principalmente en su identificación y justificación, así como el contexto en el que se sitúa.

1.3.1 Identificación y justificación del problema

El principal problema que se busca resolver en este trabajo es la ausencia de herramientas efectivas que combinen tanto la Realidad Aumentada como la impresión 3D, de manera que la experiencia del usuario en la visualización y exploración de entornos naturales se vea enriquecida significativamente.

Hoy en día, a pesar de los múltiples avances tecnológicos que existen en Realidad Aumentada e impresión 3D, son aún limitadas las soluciones que integran ambas tecnologías y que permiten ofrecer una experiencia interactiva y tangible a los usuarios.

Son varias las razones que justifican la resolución de este problema. En primer lugar, al combinar estas dos tecnologías se consigue una gran mejora de la interactividad. La integración de modelos impresos en 3D con Realidad Aumentada permite una interacción mucho más enriquecedora y significativa con los contenidos, de forma que los usuarios puedan visualizar datos en tiempo real superpuestos sobre objetos físicos del mundo real.

Por otro lado, este prototipo puede ser de gran utilidad en el ámbito de la educación. En el sector educativo, estas dos tecnologías pueden ser de gran ayuda para los estudiantes a la hora de adquirir nuevos conocimientos. La combinación de ambas tecnologías provee un aprendizaje mucho más entretenido y efectivo al permitir una interacción y visualización de los contenidos digitalmente.

Otros ámbitos en los que se puede aplicar la combinación de Realidad Aumentada e impresión 3D son la arquitectura y medicina. Los arquitectos se pueden beneficiar de la visualización de modelo en 3D y su interacción, de forma que puedan diseñar maquetas y posibles reconstrucciones de edificaciones. En cuanto a la medicina, se mejora la precisión y la seguridad en el trabajo gracias a la posibilidad que se les ofrece a los profesionales de poder explorar estructuras anatómicas complejas, lo que permite una mejor enseñanza médica.

1.3.2 Contexto del problema

El crecimiento de las tecnologías de impresión 3D y Realidad Aumentada es contante. Según estudios actuales, se estima que el tamaño del mercado de Realidad Aumentada llegue a 42,48 mil millones de dólares en 2024, y se espera que supere los 248 mil millones de dólares en 2029 [18]. Este aumento refleja el creciente interés y la inversión en tecnologías que mejoran la interacción con el entorno digital.

A su vez, el tamaño del mercado de impresión 3D (Ilustración 7) se estima en 2,13 mil millones de dólares en 2024 y se prevé que alcance los 10,10 mil millones de dólares en 2029, creciendo a una tasa anual de 36,49% [19]. Por otro lado, se estima que el tamaño del mercado mundial de plástico para impresión 3D llegue a los 4060,9 millones de dólares en 2031. A continuación, se muestra una gráfica del tamaño del mercado de plástico en la impresión 3D desde 2020 hasta 2031 [20].

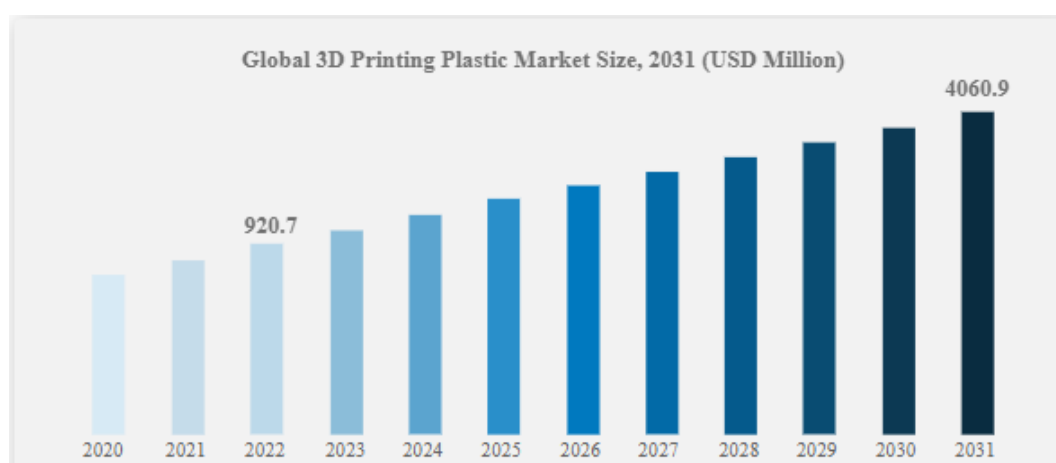


Ilustración 7: Tamaño del mercado global de plástico para impresión 3D

Por lo tanto, estas tendencias indican un contexto bastante favorable para el desarrollo de soluciones que combinan tanto la Realidad Aumentada como la impresión 3D.

Como ya se ha comentado en la sección anterior, las aplicaciones de esta solución son amplias y variadas.

En la educación, los modelos impresos en 3D combinados con Realidad Aumentada pueden mejorar significativamente la enseñanza de materias como geografía, biología o historia, ofreciendo a los alumnos la posibilidad de interactuar con objetos físicos aumentados con información digital.

En medicina, la integración de Realidad Aumentada e impresión 3D puede ser útil a la hora de crear modelos anatómicos detallados que los médicos pueden estudiar y manipular, mejorando la formación y la planificación de procedimientos quirúrgicos.

En áreas como la ciencia de datos, la capacidad de imprimir modelos en tres dimensiones y superponer información en tiempo real, puede suponer nuevas formas de visualizar y comprender grandes conjuntos de datos.

En cuanto a la arquitectura y diseño, los arquitectos y diseñadores pueden crear maquetas físicas de sus proyectos y usar la Realidad Aumentada para visualizar diferentes aspectos y modificaciones en tiempo real, mejorando la comunicación con los clientes y la precisión en la planificación.

De esta manera, el prototipo de aplicación de Realidad Aumentada y los modelos 3D que se desarrollan en este proyecto pueden llegar a tener diversas aplicaciones y usos diferentes, centrándonos principalmente en el uso educativo y arquitectónico.

1.4 Requisitos iniciales

En este apartado se indican los requisitos principales a alto nivel, sin llegar a especificar en detalle el análisis de requisitos. Por tanto, la descripción completa de los requisitos con un mayor nivel de abstracción y detalle se realizará en apartados posteriores. En este punto, y según la norma UNE 157801, los requisitos principales

se deben expresar seguidos de la palabra “debe”, mientras que las sugerencias o propuestas no obligatorias se deben expresar mediante la utilización del término “debería”.

Los requisitos del prototipo de aplicación se pueden dividir en tres categorías, cada una formando un grupo específico para el sistema.

1.4.1 Requisitos de facilidad de uso

Estos requisitos aseguran que los usuarios puedan usar el producto de manera sencilla y eficiente. Los principales requisitos de uso que se han identificado en este proyecto son:

- **Interfaz clara:** La interfaz del prototipo de aplicación debe ser clara y sin muchos elementos que distraigan la atención del usuario, de manera que este pueda centrarse en sus tareas. La interfaz debe seguir los estándares de usabilidad para ser accesible a todos.
- **Comunicación constante:** La aplicación debe mantener una comunicación continua con el usuario mediante acciones y mensajes que informen sobre el estado de la aplicación, evitando así posibles confusiones.
- **Interacción de Realidad Aumentada:** La aplicación debe proporcionar una experiencia de AR intuitiva, permitiendo a los usuarios visualizar e interactuar con modelos 3D de manera sencilla.
- **Adaptación multidispositivo:** La aplicación debería estar adaptada para su uso en diversos dispositivos (smartphones, tablets, etc.), facilitando de esta manera el acceso desde cualquier lugar y en cualquier momento.
- **Instrucciones claras:** La aplicación debe incluir unas instrucciones claras y accesibles que guíen al usuario sobre cómo utilizar las distintas funcionalidades, garantizando que se pueda aprovechar al máximo las capacidades de la aplicación.

1.4.2 Requisitos funcionales

Estos requisitos definen las funcionalidades internas de un sistema informático, en este caso de un prototipo de aplicación de Realidad Aumentada. Los requisitos funcionales de este sistema son:

- **Visualización de modelos 3D en Realidad Aumentada:** La aplicación debe permitir cargar y visualizar modelos 3D en un entorno de realidad aumentada, integrándolos adecuadamente en el mundo real del usuario.
- **Interacción con modelos 3D:** La aplicación debe permitir a los usuarios interactuar con los distintos modelos 3D, incluyendo la capacidad de rotarlos, escalarlos y moverlos dentro de espacio de Realidad Aumentada.
- **Interacción con marcadores:** La aplicación debe reconocer y responder a marcadores específicos, permitiendo a los usuarios interactuar tanto con los modelos 3D como con los marcadores.
- **Información adicional:** Al interactuar con los modelos 3D, la aplicación debe proporcionar información adicional relevante sobre dichos modelos, mejorando así la experiencia del usuario.

1.4.3 Requisitos no funcionales

Los requisitos no funcionales describen características del sistema que aseguran su correcto funcionamiento y calidad del mismo. Los principales requisitos no funcionales de nuestro sistema son los siguientes:

- **Rendimiento y eficiencia:** El sistema debe manejar y procesar modelos 3D rápida y eficientemente, minimizando los tiempos de carga y asegurando una experiencia fluida en Realidad Aumentada.
- **Optimización de recursos:** La aplicación debe optimizar el uso de recursos del dispositivo (CPU, memoria, batería, etc.) para asegurar una experiencia de Realidad Aumentada sostenible en el tiempo.

- **Fiabilidad y tolerancia a fallos:** El sistema debe gestionar fallos inesperados, proporcionando respuestas al usuario o continuando su funcionamiento sin que afecte significativamente la experiencia del usuario.
- **Compatibilidad de formatos:** La aplicación debería ser compatible con los formatos estándar de archivos para modelos 3D, facilitando la interoperabilidad con otros sistemas y herramientas de diseño e impresión 3D.

1.4.4 Requisitos de modelado e impresión 3D

Por otro lado, en este apartado también se definen los requisitos clave en cuanto al modelado e impresión de modelos 3D. Estos requisitos definen los procesos de modelado e impresión 3D que se realizarán aparte del prototipo de aplicación:

- **Preparación de modelos 3D:** Los modelos 3D deben prepararse con un nivel de detalle adecuado para su correcta impresión en impresoras 3D, asegurando su compatibilidad y preparación para la impresión.
- **Precisión de modelado:** Los modelos 3D deben ser diseñados con alta precisión para garantizar que las dimensiones y características físicas se representen fielmente en el objeto impreso.
- **Compatibilidad de formatos de modelado:** Los modelos 3D deben ser compatibles con los formatos estándar utilizados en el software de diseño 3D y las impresoras 3D, facilitando su importación y exportación.
- **Optimización para impresión 3D:** Los modelos 3D deben ser optimizados para su posterior impresión, teniendo en cuenta factores como la reducción de material, el tiempo de impresión y la estabilidad estructural de los objetos impresos.
- **Pruebas de impresión:** Antes de la producción final, se deben realizar pruebas de impresión para verificar la calidad y precisión de los modelos 3D, asegurando que cumplan con los requisitos especificados.

1.5 Análisis de riesgos

El objetivo de este apartado es identificar los riesgos potenciales que pueden afectar a este trabajo, tanto en su planteamiento como en su desarrollo.

En esta sección se provee una lista de riesgos clasificada con una evaluación de sus impactos y los planes de contingencia asociados a cada uno de ellos. Se identifican las incertidumbres, se otorga una probabilidad de ocurrencia, se evalúa el impacto (en tiempo y esfuerzo) y se presenta un plan alternativo.

1.5.1 Matriz de riesgos

Para realizar el análisis de riesgos, se ha empleado una matriz de riesgos (Ilustración 8) que combina el impacto del riesgo con su probabilidad de ocurrencia para determinar el nivel del riesgo existente. A continuación, se puede observar la matriz de riesgos que se va a utilizar en este análisis [21].

		Impact						
		N (Negligible)	Mi (Minor)	Mo (Moderate)	S (Serious)	C (Critical)		
Probability (%)	86-100	Low	Medium	High	High	High		High
	61-85	Low	Medium	Medium	High	High		Medium
	41-60	Low	Low	Medium	Medium	High		High
	21-40	Low	Low	Low	Medium	Medium		Low
	0-20	Low	Low	Low	Low	Medium		Low

Ilustración 8: Matriz de riesgos [21]

Como se puede observar en la matriz de riesgos, las probabilidades de que los riesgos ocurran se dividen en cinco niveles:

- **Very low (muy baja):** 0-20%
- **Low (baja):** 21-40%
- **Medium (media):** 41-60%

- **High (alta):** 61-85%
- **Very high (muy alta):** 86-100%

Además, podemos clasificar los tipos de impactos o riesgos en cinco categorías:

- **Negligible (insignificante):** El impacto es mínimamente perceptible y no tiene repercusiones significativas en el proyecto.
- **Minor (menor):** El impacto es leve y puede influir ligeramente en algunos aspectos secundarios del proyecto, pero no en los principales.
- **Moderate (moderado):** El impacto es considerable y puede afectar principalmente el cumplimiento de los objetivos secundarios del proyecto.
- **Serious (serio):** El impacto es notable y afecta significativamente el logro de los objetivos primarios.
- **Critical (crítico):** El impacto es máximo y pone en riesgo la consecución de los objetivos fundamentales del proyecto y su viabilidad en general.

A partir de estos dos criterios podemos clasificar los riesgos en bajos, medios o altos. Los principales riesgos que pueden afectar a este proyecto son los siguientes:

1. Dependencias de librerías y componentes externos:

- Descripción: Uso de librerías de Realidad Aumentada o de impresión 3D que podrían no cumplir con las expectativas.
- Probabilidad: Media (41-60%).
- Impacto: Serio.
- Riesgo: Medio.

- Plan de contingencia: Investigar y seleccionar librerías alternativas desde el inicio del proyecto, realizar pruebas preliminares antes de integrarlas completamente en el sistema.

2. Compatibilidad de formatos 3D:

- Descripción: Problemas de compatibilidad con los formatos estándar de archivos para modelos 3D.
- Probabilidad: Baja (21-40%).
- Impacto: Moderado.
- Riesgo: Bajo.
- Plan de contingencia: Implementar un conversor de formatos y realizar pruebas exhaustivas con diferentes tipos de archivos desde las primeras fases del proyecto.

3. Interfaz de usuario poco usable:

- Descripción: La interfaz de usuario de la aplicación de Realidad Aumentada podría no ser intuitiva para los usuarios finales.
- Probabilidad: Baja (21-40%).
- Impacto: Moderado.
- Riesgo: Bajo.
- Plan de contingencia: Realizar pruebas de usabilidad con usuarios finales durante el desarrollo, recoger el feedback de los usuarios e iterar sobre el diseño de la interfaz basándose en sus comentarios.

4. Precisión de modelado 3D:

- Descripción: Los modelos 3D podrían no ser lo suficientemente precisos para una correcta impresión.

- Probabilidad: Media (41-60%).
- Impacto: Serio.
- Riesgo: Medio.
- Plan de contingencia: Asegurarse de que los modelos 3D se diseñen con una alta precisión desde el inicio, realizar múltiples pruebas de impresión para verificar su calidad y ajustar y mejorar los modelos en base a los resultados de dichas pruebas.

5. Problemas técnicos con dispositivos de Realidad Aumentada:

- Descripción: Problemas con la compatibilidad y el rendimiento de los dispositivos de Realidad Aumentada utilizados.
- Probabilidad: Media (41-60%).
- Impacto: Crítico.
- Riesgo: Alto.
- Plan de contingencia: Investigar y realizar pruebas con diferentes dispositivos de Realidad Aumentada disponibles en el mercado y seleccionar aquellos que mejor se adapten a los requisitos.

6. Retraso en los incrementos:

- Descripción: Problemas con el retraso en la entrega de alguno de los incrementos previstos en los que se divide el proyecto.
- Probabilidad: Media (41-60%).
- Impacto: Moderado.
- Riesgo: Medio.

- Plan de contingencia: Realizar una buena planificación temporal del proyecto que se adapte perfectamente a los plazos definidos. Organizar las tareas de cada incremento en función del tiempo preestablecido.

7. Problemas técnicos con la impresora 3D:

- Descripción: Problema técnico o fallos hardware que afectan a la impresora 3D y a su correcto funcionamiento.
- Probabilidad: Baja (21-40%).
- Impacto: Moderado.
- Riesgo: Bajo.
- Plan de contingencia: Esperar hasta que estos fallos sean reparados lo antes posible y la impresora 3D vuelva a funcionar correctamente.

A continuación, se muestran estos problemas o riesgos que pueden afectar al proyecto, junto a su probabilidad de ocurrencia, impacto y riesgo asociados en forma de tabla:

Problema	Probabilidad	Impacto	Riesgo
Dependencia de librerías y componentes	Media	Serio	Medio
Compatibilidad de formatos 3D	Baja	Moderado	Bajo
Interfaz de usuario poco usable	Baja	Moderado	Bajo
Precisión de modelado 3D	Media	Serio	Medio
Problemas técnicos con dispositivos AR	Media	Crítico	Alto
Retraso en los incrementos	Media	Moderado	Medio

Problemas técnicos con la impresora 3D	Baja	Moderado	Bajo
--	------	----------	------

Tabla 1: Tabla de posibles problemas y sus riesgos

En cuanto a los planes de contingencia, existen cuatro tipos de estrategia que se pueden aplicar a cada riesgo en función de sus características:

- **Evitar:** Esta estrategia se centra en eliminar la fuente del riesgo por completo. Esto puede implicar cambiar el alcance del proyecto, seleccionar diferentes métodos o tecnologías, o incluso cancelar el proyecto completamente si el riesgo es demasiado alto.
- **Transferir:** Con esta estrategia se busca transferir la responsabilidad o el impacto del riesgo a una tercera parte. Esto se puede conseguir mediante la contratación de una empresa externa, la adquisición de un seguro, etcétera.
- **Mitigar:** Esta estrategia se enfoca en reducir la probabilidad de que un riesgo o su impacto negativo ocurra, en el caso de que este llegue a materializarse. Esto implica implementar medidas preventivas, establecer planes de contingencia y realizar mejoras en los procesos.
- **Aceptar:** Esta estrategia implica reconocer que el riesgo existe y que no se puede evitar ni transferir completamente. En este caso, se deben desarrollar planes para responder al riesgo si este llega a ocurrir, y asumir las posibles consecuencias negativas.

Una vez vistos los tipos de respuestas ante los distintos riesgos que pueden afectar a nuestro proyecto, podemos clasificar los planes de contingencia asociados a cada uno de ellos en función de estos cuatro tipos de estrategias:

Riesgo	Plan de contingencia	Tipo
Dependencias de librerías y componentes externos	Investigar y seleccionar librerías alternativas desde el inicio del proyecto, realizar pruebas preliminares antes de integrarlas completamente en el sistema.	Mitigar
Compatibilidad de formatos 3D	Implementar un conversor de formatos y realizar pruebas exhaustivas con diferentes tipos de archivos desde las primeras fases del proyecto.	Mitigar
Interfaz de usuario poco usable	Realizar pruebas de usabilidad con usuarios finales durante el desarrollo, recoger el feedback de los usuarios e iterar sobre el diseño de la interfaz basándose en sus comentarios.	Mitigar
Precisión de modelado 3D	Asegurarse de que los modelos 3D se diseñen con una alta precisión desde el inicio, realizar múltiples pruebas de impresión para verificar su calidad y ajustar y mejorar los modelos en base a los resultados de dichas pruebas.	Mitigar
Problemas técnicos con dispositivos de Realidad Aumentada	Investigar y realizar pruebas con diferentes dispositivos de Realidad Aumentada disponibles en el mercado y seleccionar aquellos que mejor se adapten a los requisitos.	Mitigar
Retraso en los incrementos	Realizar una buena planificación temporal del proyecto que se adapte perfectamente a los plazos definidos.	Evitar

	Organizar las tareas de cada incremento en función del tiempo preestablecido.	
Problemas técnicos con la impresora 3D	Esperar hasta que estos fallos sean reparados lo antes posible y la impresora 3D vuelva a funcionar correctamente.	Aceptar

Tabla 2: Riesgos y planes de contingencia.

1.6 Alcance

En este apartado se define hasta dónde llega el trabajo. Se enumeran todos los entregables junto con su contenido. En la enumeración aparecen tanto los entregables que hacen referencia al producto final, como los que hacen referencia a la gestión y control de la ejecución del proyecto.

Los entregables de este trabajo son los siguientes:

1. Documentación del proyecto final:

- **Contenido:** Informe completo del Trabajo de Fin de Grado (TFG) que incluye entre otras, secciones como la introducción, objetivos, metodología, planificación, herramientas, diseño, desarrollo, pruebas, conclusiones, anexos, etc.

En este informe también se incluye documentación técnica de la aplicación como manual de usuario y de instalación. Además, también contiene los resultados de las pruebas realizadas, así como los apartados necesarios que especifican la gestión y control de proyecto (planificación cronograma, etc.).

- **Características:** Debe cumplir con los estándares establecidos y seguir las normas de la institución.

2. Prototipo de aplicación de Realidad Aumentada:

- **Contenido:** Aplicación funcional que permita la visualización de modelos 3D en un entorno de Realidad Aumentada.
- **Características:** Debe ser compatible con dispositivos que soporten Realidad Aumentada y debe permitir interacciones básicas con los modelos 3D.

3. Modelos 3D para la impresión:

- **Contenido:** Conjunto de modelos 3D preparados para la impresión, incluyendo archivos en formato STL o similar, así como los modelos necesarios ya impresos.
- **Características:** Los modelos deben estar optimizados para la impresión 3D, con detalles precisos y sin errores.

1.7 Hipótesis y restricciones

En esta sección se identifican las diversas hipótesis y restricciones que son necesarias en la elaboración del trabajo.

En cuanto a las hipótesis, se considera que las tecnologías necesarias para el desarrollo del prototipo y los modelos 3D estarán disponibles durante todo el proyecto. Se dispone de acceso a dispositivos compatibles como smartphones o tablets con la aplicación de Realidad Aumentada. Además, se espera que los usuarios finales puedan utilizar la aplicación sin ningún tipo de problema, de manera que se asegure una usabilidad máxima.

Por otra parte, la principal restricción de este proyecto es la limitación de la duración del mismo. La duración total del proyecto está limitada a 300 horas, dado que el TFG se define como una asignatura de 12 créditos. Este tiempo incluye todas las etapas del ciclo de vida del proyecto, desde la investigación inicial hasta la entrega final, exceptuando el mantenimiento.

2 TECNOLOGÍAS Y HERRAMIENTAS

En esta sección se describen las tecnologías y herramientas utilizadas para el desarrollo de todo el proyecto, divididas en dos grupos. Por un lado, se encuentra el software utilizado para la implementación del prototipo de aplicación de Realidad Aumentada y, por otro lado, el software usado el modelado e impresión 3D.

2.1 Software para Realidad aumentada

Para el desarrollo del prototipo de la aplicación de Realidad Aumentada son necesarias varias herramientas software. La principal herramienta necesaria es el motor de desarrollo de la aplicación, en este caso, se ha elegido la plataforma de desarrollo en tiempo real de Unity [7]. Además, como herramienta de soporte y apoyo para el desarrollo de Realidad Aumentada, se ha optado por utilizar Vuforia Engine [22].

En cuanto al entorno de desarrollo integrado (IDE) y lenguaje de programación empleados en el desarrollo de la aplicación, se ha optado por la herramienta Microsoft Visual Studio [23] y el lenguaje de programación C# respectivamente.

A continuación, se detalla en profundidad cada herramienta y software utilizados en el desarrollo del prototipo de aplicación de Realidad Aumentada.

2.1.1 Unity

Unity es un motor de desarrollo de videojuegos multiplataforma creado por Unity Technologies en 2005 [24]. Unity es una plataforma de desarrollo de aplicaciones 2D y 3D en tiempo real. Esta herramienta está disponible como plataforma de desarrollo para Microsoft Windows, Mac OS y Linux, y cuenta con tres versiones: Unity Professional (Pro), Unity Personal y Unity Enterprise.

Unity (Ilustración 9) se integra con múltiples herramientas de modelado como Blender, lo que lo hace realmente útil para el desarrollo de este proyecto. Para la representación gráfica utiliza tecnologías como OpenGL y Direct3D, permitiendo así efectos como sombras dinámicas y efectos de post-procesamiento. También ofrece soporte para varios lenguajes de programación, como C#, UnityScript y Boo, mediante

el uso de Mono. Además, cuenta con características adicionales como control de versiones, sistema de audio basado en FMOD y motor de terreno [24].



Ilustración 9: Unity: Plataforma de desarrollo de videojuegos [7]

Este motor de desarrollo software incluye varias iniciativas para ayudar a sus usuarios en el desarrollo de aplicaciones y juegos, como por ejemplo la iniciativa de Asset Store [25]. El Unity Asset Store, lanzado en noviembre de 2010, es un recurso disponible dentro del editor de Unity. Ofrece acceso a una amplia colección de paquetes de Assets en diversas categorías, que incluyen modelos 3D, tutoriales, texturas, sistemas de partículas, scripts, música, efectos de sonido, extensiones para el editor, y muchos más [24].

Unity también cuenta con otros recursos que permiten integrar Realidad Aumentada en el desarrollo de las aplicaciones. Unity ofrece varias herramientas en función de la plataforma utilizada: ARKit para dispositivos Apple, ARCore para dispositivos Android y Mixed Reality Toolkit para dispositivos Windows Mixed Reality.

ARKit y ARCore están englobadas en el conjunto de herramientas AR Foundation (Ilustración 10) [26], que es un marco de trabajo de Unity desarrollado especialmente para la integración de Realidad Aumentada en las propias aplicaciones o videojuegos. Incluye funciones importantes de cada plataforma, así como funciones únicas de Unity, como optimizaciones para los dispositivos, física y renderizado fotorrealista entre otras [27].



Ilustración 10: Uso de AR Foundation en Unity [26]

Sin embargo, para el desarrollo de esta aplicación de Realidad Aumentada, se ha optado por el uso de Vuforia Engine, en lugar de AR Foundation, debido al uso de modelos 3D como marcadores y modelos objetivos. En la siguiente sección se profundizará más sobre esta herramienta.

Unity, en su versión gratuita, ha sido elegida como la plataforma ideal para desarrollar la aplicación de Realidad Aumentada por varias razones:

- **Versatilidad y potencia:** Unity es una de las plataformas más versátiles y potentes para el desarrollo de aplicaciones de Realidad Aumentada. Es ideal para este tipo de proyectos debido a su gran capacidad de manejar gráficos 3D avanzados y su arquitectura robusta de desarrollo.
- **Compatibilidad multiplataforma:** Unity permite desarrollar aplicaciones que pueden ejecutarse en múltiples plataformas, como iOS, Android y Windows, entre otras. Esta característica garantiza que la aplicación pueda ser utilizada en una amplia gama de dispositivos, aumentando así su alcance y usabilidad potencial.
- **Soporte para Realidad Aumentada:** Unity ofrece integraciones robustas con herramientas y frameworks de Realidad Aumentada como ARCore, ARKit y Vuforia, facilitando la implementación eficiente de funcionalidades avanzadas de Realidad Aumentada.

- **Comunidad y recursos:** La comunidad de desarrolladores de Unity es bastante extensa y cuenta con una gran cantidad de recursos educativos, foros de ayuda y tutoriales. Esto simplifica la resolución de problemas y la adquisición de los conocimientos necesarios durante el proceso de desarrollo.
- **Facilidad de uso:** A pesar de su potencia, Unity es relativamente fácil de utilizar gracias a su interfaz intuitiva y a sus herramientas de desarrollo documentadas adecuadamente. Posee una curva de aprendizaje bastante sencilla. Esto agiliza el proceso de desarrollo, especialmente considerando el tiempo limitado del proyecto.
- **Integración con modelado 3D:** Unity permite la importación y manipulación eficiente de modelos 3D, aspecto crucial en el desarrollo de este trabajo, ya que se combina la Realidad Aumentada con modelos 3D impresos. De esta manera se garantiza que los modelos se representen correctamente y sean interactivos.
- **Rendimiento:** Las aplicaciones desarrolladas en Unity suelen destacarse por su buen rendimiento en términos de velocidad y estabilidad. Esto es clave para ofrecer una experiencia de usuario fluida, especialmente en una aplicación de Realidad Aumentada donde el rendimiento gráfico es crucial.
- **Flexibilidad de desarrollo:** Unity ofrece una gran flexibilidad en términos de programación, con soporte para lenguajes como C#, lo que permite adaptar fácilmente el entorno de desarrollo a las necesidades específicas del proyecto.

2.1.2 Vuforia Engine

Vuforia Engine [22] es una plataforma para el desarrollo de aplicaciones de Realidad Aumentada (AR) y Realidad Mixta (MR) compatible con múltiples plataformas. Ofrece un seguimiento robusto y un rendimiento eficaz en diversos dispositivos, incluyendo móviles y visores de realidad mixta, como el Microsoft HoloLens. Su integración con Unity permite crear juegos y aplicaciones visuales tanto

para Android como para iOS a través de un flujo de trabajo sencillo basado en arrastrar y soltar. En la Asset Store de Unity se puede encontrar un paquete de muestras Vuforia AR + VR, que incluye ejemplos prácticos que muestran las características principales de la plataforma [28].

Vuforia (Ilustración 11) es compatible con numerosos dispositivos de terceros, como gafas de AR/MR y dispositivos de realidad virtual como cámaras traseras, como el Gear VR.



Ilustración 11: Vuforia Engine [22]

Vuforia destaca por una serie de características principales:

- 1) Reconocimiento de objetos:** Vuforia es capaz de reconocer y rastrear tanto imágenes como objetos en 3D.
- 2) Facilidad de integración:** Se integra fácilmente con Unity, lo que permite a los desarrolladores añadir capacidades de Realidad Aumentada sin complicaciones.
- 3) Herramientas de desarrollo:** Vuforia incluye una gran variedad de herramientas para la creación y prueba de Realidad Aumentada, como Vuforia Model Target Generator, Vuforia Model Target CLI, Vuforia Area Target Generator, Vuforia Creator, Vuforia VuMark Designer o UWP Camera Bridge.

Se ha elegido la utilización de Vuforia Engine en este proyecto ya que permite superponer información digital de manera aumentada fácilmente, además de su fácil integración con Unity (Ilustración 12). La principal razón por la que se ha optado usar esta herramienta frente a otras proporcionadas por Unity como ARKit o ARCore, es la capacidad de rastrear no solo imágenes, sino también modelos u objetos 3D. Esto es

totalmente necesario para este trabajo ya que será necesario que la aplicación sea capaz de detectar los modelos impresos en 3D.

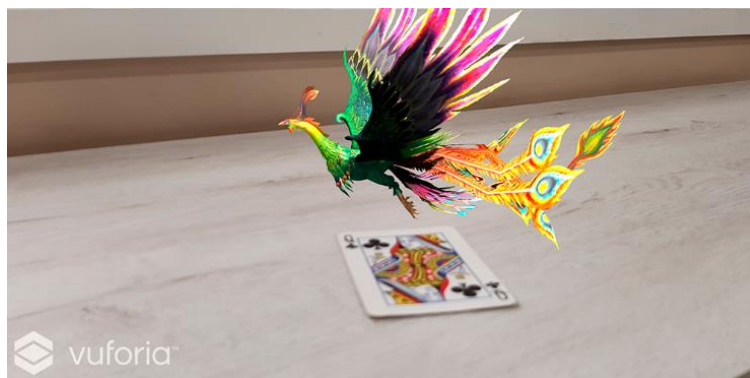


Ilustración 12: Uso de Vuforia Engine en Unity

Para poder realizar este rastreo de modelos 3D en Realidad Aumentada son necesarias varias de las herramientas de proporciona Vuforia. En primer lugar, se ha utilizado la herramienta Model Target Generator [29] que permite convertir rápidamente un modelo 3D existente en un conjunto de datos que podrá ser usado por Vuforia, de manera que este objeto pueda ser detectado y rastreado por la aplicación. Este conjunto de datos puede ser utilizado para crear un Objetivo de Modelo (Model Target) de Vuforia en Unity.

Además, esta aplicación de escritorio permite verificar si las características de los modelos son utilizables. El generador de objetivos de modelos es compatible con formatos como OBJ, FBX o STL entre otros.

En relación con esta herramienta existen otras dos herramientas que permiten realizar pruebas con los objetivos de modelo generados, una compatible con dispositivos iOS, Vuforia Creator (Ilustración 13), y otra para Android, Model Targets Test App (Ilustración 14).

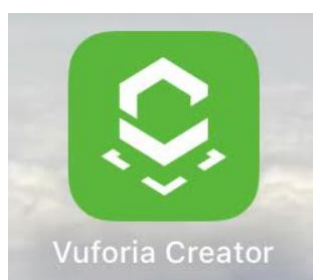


Ilustración 13: Aplicación Vuforia Creator para iOS

Ambas aplicaciones proporcionan una forma sencilla de validar los objetivos, basta con cargar los objetivos de modelos y probar que sean detectados correctamente.

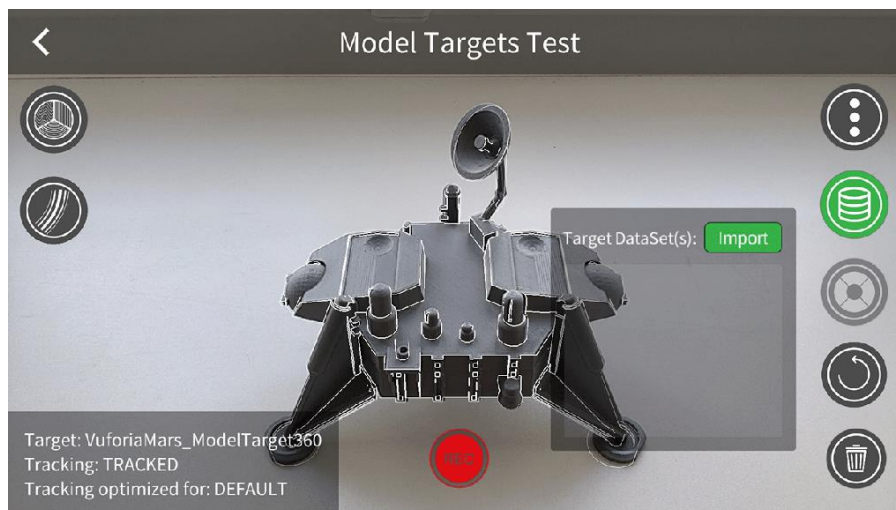


Ilustración 14: Model Targets Test App para Android

2.1.3 Microsoft Visual Studio

Microsoft Visual Studio [23] es un entorno de desarrollo integrado (IDE) para Windows y macOS. Es compatible con diversos lenguajes de programación, como C++, C#, Java, Python, etcétera [30]. También ofrece herramientas avanzadas de depuración y análisis de código y permite instalar extensiones que aumentan su funcionalidad.

Algunas de las ventajas de uso que ofrece esta herramienta son las siguientes:

- **Entorno completo de desarrollo:** Proporciona todas las herramientas necesarias para el desarrollo, desde la codificación hasta la depuración y el despliegue.
- **Integración con Unity:** Se integra perfectamente con Unity, lo que permite un flujo de trabajo bastante eficiente.
- **Facilidad de uso:** Su interfaz de usuario es personalizable e intuitiva, lo que facilita el trabajo del desarrollador.

Se ha elegido este IDE como plataforma para el desarrollo de la programación de la aplicación, usando el lenguaje de programación C#, ya que esta herramienta está directamente vinculada con Unity, lo que facilita la integración, escritura y depuración del código. En este caso se utiliza la versión gratuita Visual Studio Community 2022 (Ilustración 15).



Ilustración 15: Microsoft Visual Studio 2022

2.2 Software para Modelado e Impresión 3D

Para el modelado e impresión de los modelos 3D que serán utilizados en la aplicación de Realidad Aumentada son necesarias distintas herramientas software especializadas. La principal herramienta utilizada para el modelado 3D es Blender [31], una herramienta de modelado 3D gratuita y de código abierto.

En cuanto a la edición y optimización de modelos 3D, se ha optado por la herramienta Meshmixer [32], un software diseñado para modificar y reparar modelos para la impresión 3D.

En el contexto de la creación de texturas y su aplicación en los modelos 3D, se ha utilizado la herramienta software GIMP [33], un editor de imágenes gratuito y de código abierto. También se ha utilizado esta herramienta para el diseño y creación de las imágenes que actúan como marcadores.

Finalmente, para convertir los modelos 3D en instrucciones específicas para la impresión 3D, se ha elegido la herramienta PrusaSlicer [34], un software que toma los modelos 3D y los convierte en archivos G-code, que contienen las instrucciones necesarias para que la impresora 3D pueda imprimir los objetos físicos.

A continuación, se detalla en profundidad cada herramienta y software usados en el desarrollo de los modelos 3D.

2.2.1 Blender

Blender [31] es un programa informático multiplataforma, especializado en el modelado, renderizado, iluminación, animación y creación de gráficos tridimensionales. Además, permite la composición digital mediante nodos, la escultura, la pintura digital y la edición de vídeo. Es un software libre compatible con todas las versiones de Windows, macOS, GNU/Linux, Android, Solaris, FreeBSD e IRIX [35].

Algunas de las características principales de esta herramienta son las siguientes:

- 1) **Modelado y esculpido:** Herramientas avanzadas para modelado poligonal y esculpido digital.
- 2) **Animación y rigging:** Capacidades completas para la animación de personajes y objetos.
- 3) **Renderizado:** Motor de renderizado Cycles y soporte para renderizado en tiempo real con Eevee.
- 4) **Simulaciones:** Simulación de fluidos, humo, partículas, etcétera.

En este proyecto se ha utilizado la herramienta Blender (Ilustración 16) para realizar todo el proceso de modelado 3D de las maquetas de los entornos naturales y objetos que serán impresos en 3D. Una de las principales razones por la que se ha optado por utilizar este programa de modelado 3D es por su gran capacidad para manejar todas las etapas del modelado 3D dentro de una única herramienta, además de su potencia y flexibilidad.



Ilustración 16: Herramienta de modelado Blender [31]

Además, Blender cuenta con muchas ventajas de uso, entre las cuales destacan las siguientes:

- **Gratuito y Open Source:** Es una herramienta accesible para todos los desarrolladores sin ningún tipo de coste.
- **Versatilidad:** Blender soporta una amplia gama de tareas de modelado y animación.
- **Comunicación activa:** Existe una gran comunidad de usuarios que proporciona tutoriales, scripts y complementos.

2.2.2 Meshmixer

Meshmixer [32] es un software 3D gratuito ofrecido por Autodesk, disponible para Windows y Mac OS, que permite editar y optimizar modelos 3D para la impresión. Es especialmente útil para modificar y preparar modelos, asegurando que estén listos para su impresión 3D sin errores.

Meshmixer (Ilustración 17) permite la edición de modelos 3D importando archivos en varios formatos como STL, OBJ, PLY y AMF. Una vez importado, el programa permite modificar, medir, redimensionar o reparar el modelo usando mallas triangulares compuestas por vértices, aristas y caras [36].



Ilustración 17: Autodesk Meshmixer

Se ha elegido esta herramienta de modelado 3D principalmente para reparar los modelos tras su creación en Blender, sobre todo para cerrar, suavizar y reducir la malla de polígonos de los modelos.

2.2.3 GIMP

GIMP (GNU Image Manipulation Program) [33] es un software para la edición de imágenes digitales en formato de mapa de bits, que incluye tanto dibujos como fotografías. Es un programa libre y gratuito, perteneciente al proyecto GNU. Es compatible con la mayoría de los sistemas operativos como Unix, GNU/Linux, Microsoft Windows y macOS, entre otros [37].

Algunas de las características esenciales de este programa son las siguientes:

- 1) **Edición de imágenes:** Herramientas avanzadas para el retoque, composición y creación de imágenes.
- 2) **Soporte de capas:** Permite trabajar con múltiples capas para una edición más precisa.
- 3) **Filtros y efectos:** Amplia gama de filtros y efectos para la mejora y modificación de imágenes.
- 4) **Personalización:** Interfaz altamente personalizable con soporte para complementos y scripts.

GIMP (Ilustración 18) se ha utilizado principalmente para la edición de las imágenes de las texturas correspondientes a cada modelo 3D, así como la creación

de las diferentes imágenes que actúan como marcadores en la aplicación de Realidad Aumentada.



Ilustración 18: Programa de edición de imágenes GIMP

2.2.4 PrusaSlicer

PrusaSlicer [34] es un software abierto y libre que permite preparar los modelos para su impresión 3D. Convierte los modelos 3D en instrucciones detalladas (G-code) que las impresoras usan para crear los objetos físicos.

Esta herramienta tiene las siguientes características principales [34]:

- 1) **Interfaz intuitiva:** En las últimas versiones se ha rediseñado la interfaz de usuario para que sea más clara y sencilla de usar.
- 2) **Soportes personalizados:** Permite pintar soportes personalizados directamente en el modelo o bloquear los soportes automáticos en áreas específicas.
- 3) **Análisis de tiempo/características de impresión:** Contiene una vista previa que muestra un desglose del tiempo de impresión por característica.
- 4) **Envío de código G a través de la red:** Permite enviar el código G-code a la impresora directamente, compatible con Octoprint, Repetier Host y tarjetas SD FlashAir, entre otros.

PrusaSlicer (Ilustración 19) es la herramienta que se usará en este caso para preparar los modelos para su impresión y enviar el código G-code directamente a la impresora.



Ilustración 19: Herramienta de impresión 3D PrusaSlicer

3 PLANIFICACIÓN

En este apartado se tratarán todos los elementos que están relacionados con la planificación del proyecto, incluyendo la metodología de desarrollo de software, la planificación temporal y el presupuesto, teniendo en cuenta los costes del software, del hardware y del personal.

3.1 Metodología de desarrollo de software

En este apartado se expone la metodología de desarrollo software seleccionada para llevar a cabo el proyecto, así como las razones que fundamentan dicha elección. Dado que la metodología de desarrollo afecta a todo el proyecto, es fundamental elegir la más adecuada para garantizar el éxito de este trabajo.

Para elegir la mejor metodología de desarrollo software para el proyecto en cuestión, primero se va a realizar un análisis tanto de las metodologías tradicionales como de las metodologías ágiles. A continuación, se explicará cada una de ellas junto con sus tipos y se proporcionará una comparativa para justificar la metodología elegida finalmente para este trabajo.

Por un lado, las metodologías tradicionales establecen un enfoque estructurado para el desarrollo de software, donde se prioriza la disciplina en el proceso. Esto implica una planificación exhaustiva antes de iniciar el ciclo de desarrollo del producto, con el objetivo de lograr una mayor eficiencia en el software final [38].

Según Pressman (2013), las metodologías tradicionales, también conocidas como modelos de proceso prescriptivo, surgieron en la década de los 60 para organizar el caos en el desarrollo de software. Estos modelos proporcionaron una estructura útil para la ingeniería de software y siguen siendo eficaces para los equipos de desarrollo. En estas metodologías, el proyecto se concibe como uno grande y bien definido, con un proceso secuencial sin retroceso, rigidez en los cambios y acuerdos de requisitos que se mantienen durante todo el proyecto. Esto implica una extensa planificación inicial y poca comunicación con el cliente una vez que esta planificación se ha completado [39].

A continuación, se exponen algunos de los diferentes enfoques utilizados en la metodología tradicional o clásica, detallando sus principales características, ventajas e inconvenientes [40].

El **modelo en Cascada (Waterfall)** es un enfoque lineal y secuencial para el desarrollo de software, con fases perfectamente definidas que fluyen de manera descendente como una cascada. Estas etapas incluyen la definición de requisitos, diseño, implementación, verificación y mantenimiento (Ilustración 20).

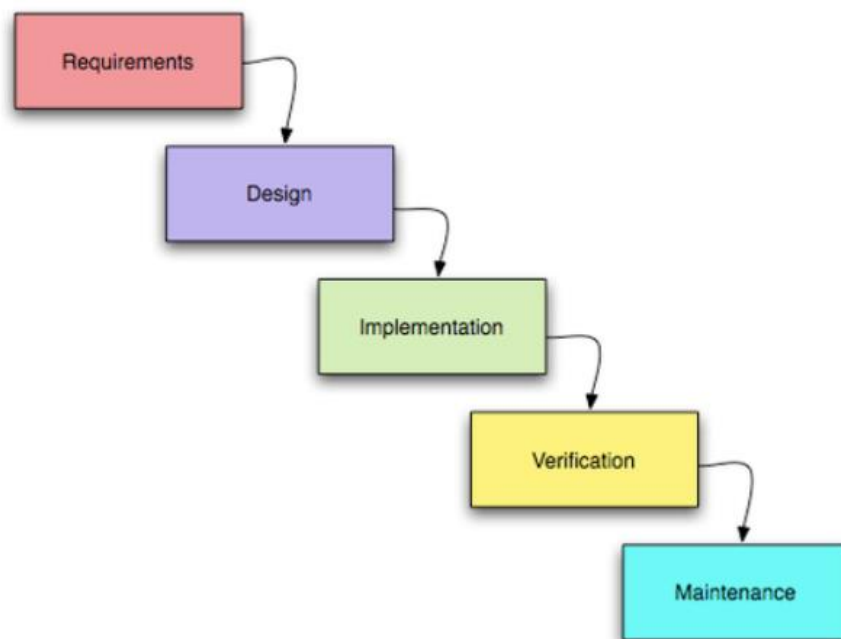


Ilustración 20: Modelo del ciclo de vida en Cascada

Las principales ventajas de este modelo son una estructura clara y fácil de entender para el desarrollo del proyecto, es adecuada para proyectos con requisitos estables y bien definidos, es simple, fácil de implementar y facilita la detección de errores. También tiene desventajas como poca flexibilidad para adaptarse a cambios en los requisitos durante el proceso de desarrollo y dificultad de corregir errores en etapas ya avanzadas del proyecto al no poder retroceder a una etapa anterior.

El **modelo basado en Prototipos** se centra en la construcción rápida de versiones preliminares del producto final para su validación y refinamiento de los requisitos del proyecto. Los prototipos son utilizados para obtener una

retroalimentación temprana por parte de los usuarios y clientes finales, lo que permite realizar cambios antes de la implementación completa (Ilustración 21).



Ilustración 21: Modelo del ciclo de vida del Prototipado

Este modelo tiene como ventajas una mayor participación por parte del cliente, una mejor comprensión de los requisitos, facilita la detección de problemas y riesgos y permite probar el modelo, arquitectura o tecnología que se está desarrollando. Sin embargo, puede resultar costoso de desarrollar y mantener múltiples prototipos, puede ser lento y puede generar una falsa ilusión al cliente o usuario final.

El **modelo en Espiral** combina elementos de enfoques iterativos y secuenciales, dividiendo el proyecto en ciclos de planificación, análisis de riesgos, desarrollo y evaluación. En cada ciclo se obtiene una versión más refinada del producto, permitiendo una adaptación continua conforme se van identificando los riesgos (Ilustración 22).

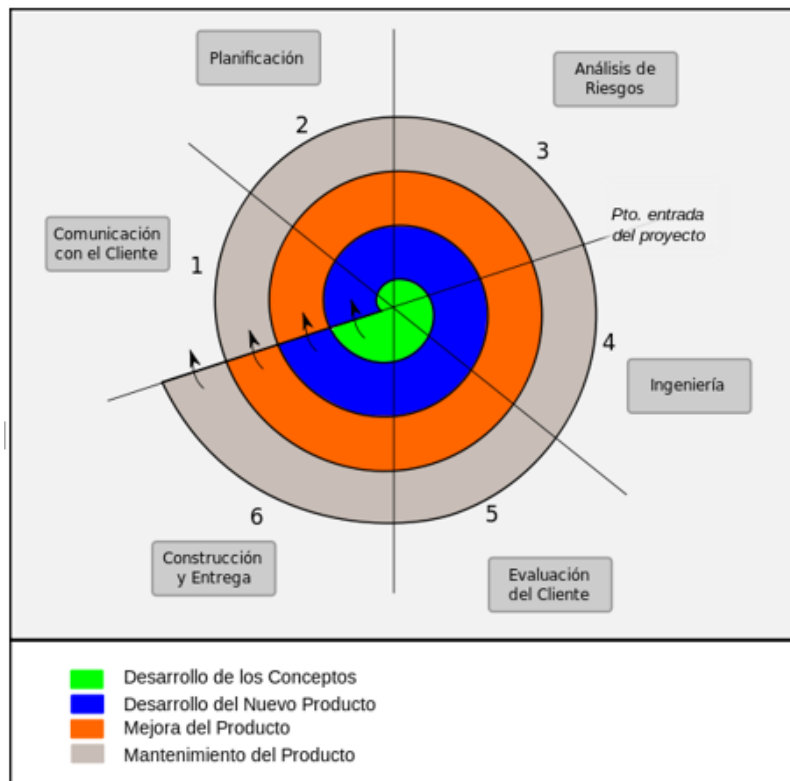


Ilustración 22: Modelo del ciclo de vida en Espiral

Este modelo integra las ventajas del modelo de desarrollo en cascada y del modelo basado en prototipos, incorporando el análisis de riesgos. Las ventajas de este modelo son un enfoque en la gestión proactiva de riesgos, permite añadir nuevos cambios durante el desarrollo y proporciona una estructura adaptable y flexible. También tiene algunos inconvenientes, ya que puede resultar compleja de gestionar y controlar, requiere un equipo altamente experimentado y no es recomendable para proyectos pequeños.

El **modelo Incremental** se basa en construir y mejorar un producto mediante un conjunto de versiones incrementales. Se divide el proyecto en distintas partes o módulos y se implementan y desarrollan de forma incremental y progresiva. En cada iteración se añaden al producto final nuevas funcionalidades (Ilustración 23).



Ilustración 23: Modelo del ciclo de vida Incremental

Algunas de sus principales ventajas son una entrega temprana de funcionalidades básicas, permite la retroalimentación del cliente a lo largo del desarrollo, reduce los riesgos y facilita la gestión del cambio. En cambio, puede ser difícil mantener el enfoque en los requisitos finales y requiere una buena coordinación y planificación para asegurar la integración adecuada de los módulos.

Por último, **la metodología RAD** (Desarrollo Rápido de Aplicaciones) se centra en el desarrollo rápido y la entrega temprana de software funcional, utilizando técnicas como la reutilización de componentes, la prototipación y la iteración rápida. Su objetivo es reducir el tiempo de entrega del producto final y aumentar la participación de los usuarios en el proceso de desarrollo. Las fases de este modelo son: modelado de gestión, modelado de datos, modelado de proceso y generación de aplicaciones (Ilustración 24).

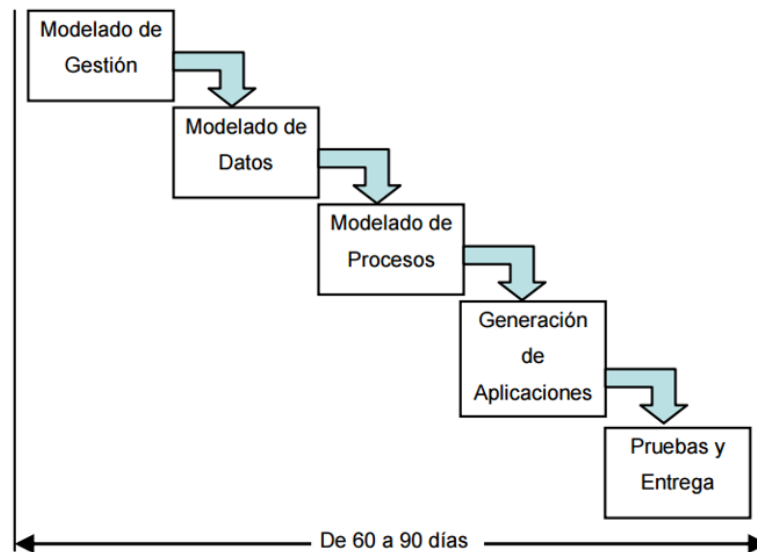


Ilustración 24: Modelo del ciclo de vida RAD

Este modelo tiene como ventajas la reducción del tiempo de entrega del producto final, una mayor participación por parte del usuario, una mayor flexibilidad y la identificación temprana de problemas. Algunas de las desventajas que presenta son el difícil mantenimiento de la calidad del software con un enfoque tan rápido, requiere un equipo altamente experimentado y puede ser menos eficiente.

Por otro lado, la metodología ágil es una de las distintas metodologías de desarrollo de software que se basan en el desarrollo iterativo e incremental, a diferencia de las metodologías tradicionales de desarrollo de software lineal o cascada [38].

En la actualidad, las empresas necesitan adaptarse rápidamente a los cambios en el mercado global. Como el software es esencial para muchas operaciones empresariales, es importante desarrollarlo de forma flexible y rápida para satisfacer las necesidades del cliente. Las metodologías ágiles destacan por su capacidad de adaptación y flexibilidad y son excelentes para esto porque dividen los proyectos en partes más pequeñas, mantienen una comunicación constante con los usuarios y se ajustan fácilmente a los cambios. Además, permiten revisar y mejorar continuamente el trabajo en progreso [39].

A continuación, se describen algunos de los enfoques más comunes dentro de la metodología ágil, destacando sus características principales, ventajas e

inconvenientes. Aunque existen diferentes enfoques dentro de la metodología ágil, voy a exponer los tres más comunes [40].

La **Programación Extrema** o Extreme Programming (XP) es una de las metodologías ágiles más conocidas. Fue creada por Kent Beck a finales de la década de 1990 como un enfoque para mejorar la calidad del software y la satisfacción del cliente. Utiliza prácticas como la programación en parejas, la integración continua y las pruebas automatizadas (Ilustración 25).

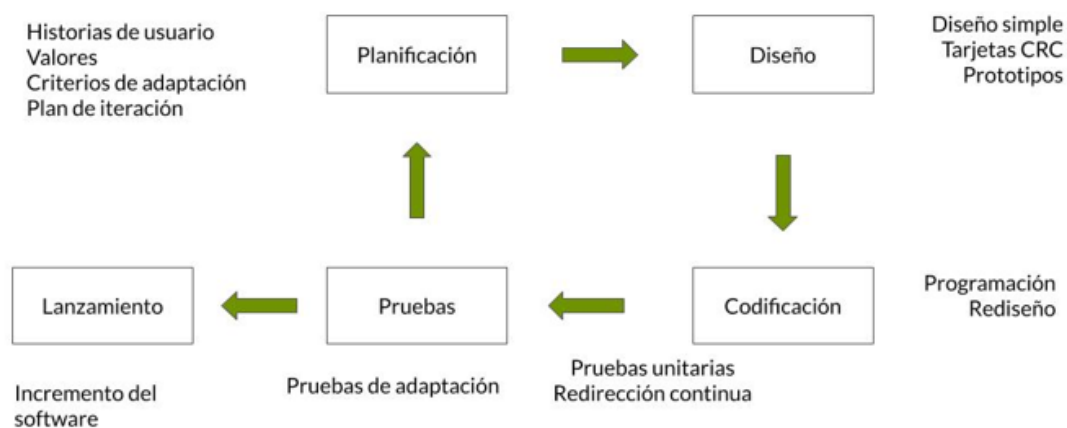


Ilustración 25: Metodología XP o Programación Extrema

Esta metodología tiene como ventajas una mejora de la calidad del software y la productividad del equipo y permite una adaptación rápida a los cambios en los requisitos. En cambio, tiene ciertas desventajas al requerir un compromiso total del equipo y puede ser desafiante para proyectos grandes o complejos.

La metodología **Scrum** es un marco de trabajo ágil que organiza el trabajo en sprints, ciclos de desarrollo cortos que generalmente duran de una a cuatro semanas. Se basa en roles definidos, eventos y artefactos para facilitar la colaboración y la entrega continua de valor. Algunos de los roles utilizados son Scrum Master, Product Owner y el Equipo de Desarrollo (Ilustración 26).

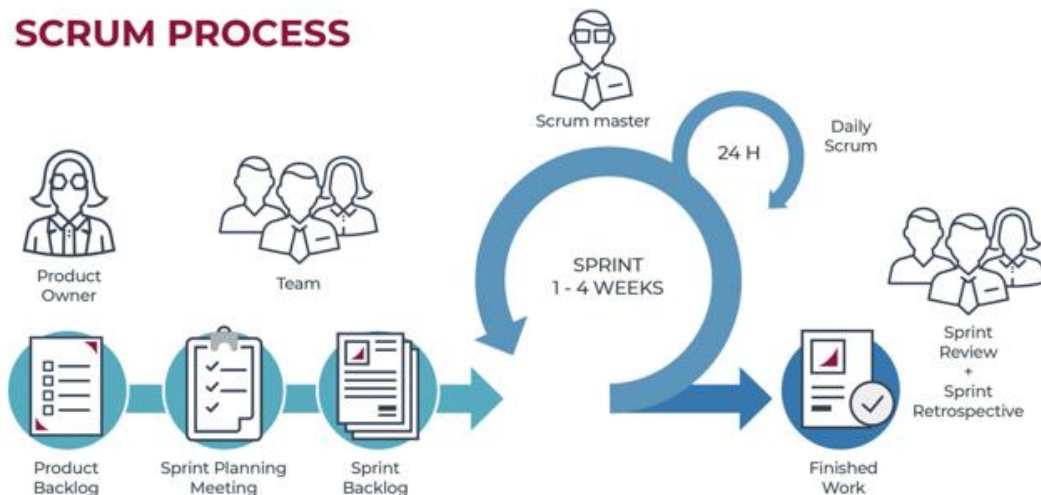


Ilustración 26: Proceso y roles de Scrum

Entre sus principales ventajas destacan la transparencia y la colaboración dentro del equipo, además de una adaptación rápida a los cambios en los requisitos y la posibilidad de entregas parciales a corto plazo. Algunos de sus inconvenientes son que puede resultar desafiante para equipos grandes o distribuidos y requiere una comprensión sólida de los roles y procesos de Scrum.

La metodología **Kanban** se basa en los sistemas de producción Lean y se adaptó al desarrollo de software como un enfoque para gestionar el flujo de trabajo. Utiliza un tablero visual para visualizar el flujo de trabajo y limitar el trabajo en curso. Se centra en la mejora continua y la eliminación de desperdicios (Ilustración 27).

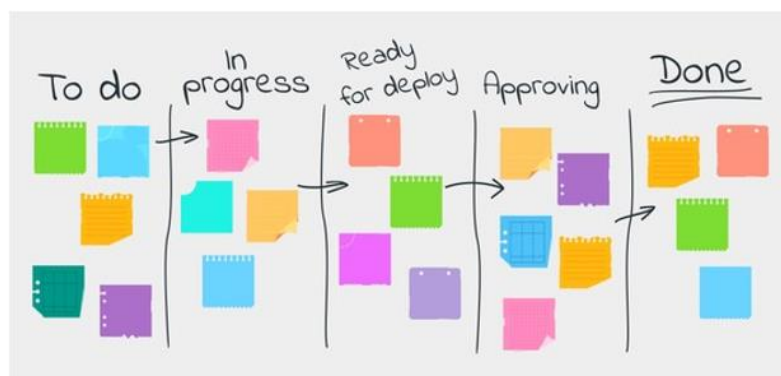


Ilustración 27: Tablero Kanban

Las principales ventajas de esta metodología son una mejora de la visibilidad y la transparencia del proceso de desarrollo, permite una rápida adaptación a los

cambios en la demanda y los requisitos. Sin embargo, puede resultar menos estructurado que otras metodologías ágiles, requiere una gestión cuidadosa del tablero Kanban y los límites de trabajo en curso.

A continuación, se muestra una tabla comparativa [40] entre estas dos metodologías desarrolladas anteriormente:

Metodologías ágiles	Metodologías tradicionales
Basadas en heurísticas provenientes de producción de código	Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo
Preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente por el equipo	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso muy controlado, numerosas normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado
Es cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajan en el mismo lugar	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menor énfasis en la arquitectura del software	La arquitectura del software es esencial
Poca documentación	Documentación exhaustiva
Muchos ciclos de entrega	Pocos ciclos de entrega

Tabla 3: Comparación entre metodologías ágiles y tradicionales [41]

Una vez vistas estas dos metodologías junto con sus enfoques más utilizados, he optado por utilizar el modelo incremental para este proyecto por varias razones.

En primer lugar, tanto el modelado de la maqueta en 3D como la implementación de la aplicación de Realidad Aumentada requieren de la realización de distintas pruebas intermedias durante su desarrollo. La metodología incremental permite adaptarse a los diversos cambios y nuevas ideas que puedan surgir a lo largo del desarrollo del proyecto.

Este enfoque también permite la división del trabajo en etapas más pequeñas y manejables, de manera que se pueden ir entregando e implementando nuevas funcionalidades poco a poco y gestionar los desafíos que pueden ir surgiendo durante el desarrollo del prototipo.

Además, esta metodología posibilita la incorporación de retroalimentación de los usuarios, muy útil ya que uno de los objetivos principales de este trabajo es probar el prototipo con usuarios reales para evaluar su usabilidad, lo que garantiza que se cumplan las necesidades y expectativas del cliente.

La naturaleza de este proyecto, donde parte de los requisitos finales no están completamente definidos desde un principio, y donde la utilización de incrementos o iteraciones favorece la implementación gradual de las diferentes funcionalidades de la aplicación, hacen que la metodología de desarrollo software incremental sea la más adecuada para el desarrollo de este trabajo.

3.1.1 Adaptación de la metodología incremental al proyecto

En este apartado se detalla la planificación seguida para el desarrollo de este proyecto. Teniendo en cuenta que el trabajo tiene una limitación de 300 horas y que la metodología seguida es incremental, se dividirá el trabajo en una serie de incrementos o iteraciones que se detallarán a continuación. Cada incremento se define por un conjunto de tareas y actividades que debe completar, y unos resultados esperados que se deben obtener al final de cada incremento.

Previamente, se ha realizado un análisis y planteamiento de los distintos objetivos del proyecto, para facilitar la posterior división en iteraciones.

Iteración 1: Diseño de Modelos 3D

El objetivo de este incremento es diseñar y crear los modelos 3D necesarios para el proyecto.

- Tareas:

- Diseño inicial de los modelos 3D utilizando la herramienta Blender.
- Aplicación de texturas a los modelos mediante la herramienta GIMP.
- Revisión y ajuste de los modelos para asegurar la calidad y detalle necesarios.

- Resultados esperados:

- Modelos 3D detallados y texturizados con formato GLB.
- Archivos de Blender y GIMP documentados.

Iteración 2: Preparación e Impresión 3D

El objetivo de esta iteración consiste en optimizar y preparar los modelos 3D ya diseñados para su correcta impresión en 3D.

- Tareas:

- Uso de Meshmixer para realizar ajustes y optimizar los modelos 3D.
- Reparación de errores en la geometría de los modelos.
- Preparación de los archivos STL para la impresión, configuración de la impresora 3D y uso de PrusaSlicer para generar los archivos G-code.
- Proceso de impresión 3D y ajustes de los modelos impresos.

- Resultados esperados:

- Modelos 3D optimizados y correctamente impresos.
- Archivos STL y G-code documentados.

Iteración 3: Diseño e implementación de la aplicación móvil de AR

En esta iteración se busca desarrollar la base principal de la aplicación de Realidad Aumentada.

- Tareas:

- Configuración del entorno de desarrollo en Unity.
- Integración básica de Vuforia para la detección de marcadores y modelos.
- Prueba integración de Vuforia.
- Diseño e implementación de la interfaz de usuario.

- Resultados esperados:

- Prototipo básico de la aplicación de Realidad Aumentada.
- Código fuente inicial documentado.

Iteración 4: Integración de los modelos 3D

En este incremento se pretende conseguir la integración de los modelos 3D impresos en la aplicación de Realidad Aumentada.

- Tareas:

- Escaneado y generación de modelos objetivos usando la herramienta de Vuforia, Model Target Generator.
- Ajustes en Unity para la correcta detección y visualización de los modelos 3D.
- Pruebas iniciales de integración.

- Resultados esperados:

- Modelos 3D correctamente integrados en la aplicación de Realidad Aumentada.
- Documentación de pruebas de integración.

Iteración 5: Interacciones con marcadores y modelos

El objetivo de esta iteración es desarrollar funcionalidades avanzadas de interacción entre los usuarios, los marcadores y los modelos 3D.

- Tareas:

- Diseño de las imágenes de los marcadores utilizando GIMP.
- Implementación de interacciones con marcadores en Unity usando Vuforia (rotación, zoom, traslación).
- Implementación de interacciones con modelos 3D, con superposición de información digital sobre los modelos impresos.
- Implementación de interacciones avanzadas entre varios marcadores.
- Implementación de interacciones avanzadas entre marcadores y modelos.
- Pruebas de usabilidad y ajustes necesarios.

- Resultados esperados:

- Prototipo avanzado con interacciones funcionales entre los usuarios, los marcadores y los modelos 3D.
- Documentación de pruebas de usabilidad y ajustes realizados.

Iteración 6: Solución de errores y refinamiento

En esta iteración se busca identificar y solucionar errores presentes en la aplicación y en los modelos 3D. Otro de los objetivos de este incremento es realizar refinamientos finales para mejorar la funcionalidad y usabilidad del sistema.

- Tareas:

- Pruebas exhaustivas para identificar y solucionar errores.
- Diseño del logo de la aplicación.
- Implementar compatibilidad con varios idiomas.
- Implementación de mejoras basadas en la retroalimentación obtenida.

- Resultados esperados:

- Prototipo final sin errores críticos y con funcionalidades refinadas.
- Documentación final del proceso de desarrollo y de los resultados obtenidos.

La adaptación de la metodología incremental a este proyecto permite una gestión estructurada y eficiente del desarrollo, de manera que cada fase aporte valor y permita realizar ajustes iterativos. Esta aproximación facilita la identificación y resolución temprana de errores, optimizando así el proceso de desarrollo.

3.2 Planificación temporal

En esta sección se desarrolla de manera detallada la planificación temporal del proyecto. Se detalla un cronograma que incluye los resultados parciales, hitos intermedios y la duración del trabajo a partir de la fecha de inicio del mismo.

Teniendo en cuenta que la duración del proyecto total es de 300 horas, se ha repartido el trabajo a lo largo de las semanas dedicando aproximadamente 20 horas por semana, lo que da lugar a un total de 15 semanas de trabajo. El trabajo se empezó la semana del 4 de marzo, por lo que, según la planificación la fecha límite para finalizarlo sería la semana del 10 de junio, dejando una semana previa para la organización y planificación anterior del proyecto.

A continuación, se detalla más en profundidad el tiempo dedicado a cada tarea en cada iteración. El proyecto se llevará a cabo en un plazo de 15 semanas (75 días),

distribuyendo las 300 horas disponibles de la siguiente manera (20 horas por semana, 4 horas por día, 5 días a la semana):

- **Preparación previa:** 5 días.
- **Iteración 1: Diseño de Modelos 3D** - 15 días, 60 horas.
 - Diseño inicial de los modelos 3D utilizando la herramienta Blender (12 días).
 - Aplicación de texturas a los modelos mediante la herramienta GIMP (1 día).
 - Revisión y ajuste de los modelos para asegurar la calidad y detalle necesarios (2 días).
- **Iteración 2: Preparación e Impresión 3D** - 8 días, 32 horas.
 - Uso de Meshmixer para realizar ajustes y optimizar los modelos 3D (2 días).
 - Reparación de errores en la geometría de los modelos (1 día).
 - Preparación de los archivos STL para la impresión, configuración de la impresora 3D y uso de PrusaSlicer para generar los archivos G-code (1 día).
 - Proceso de impresión 3D y ajustes de los modelos impresos (4 días).
- **Iteración 3: Diseño e implementación de la aplicación móvil de AR** - 10 días, 40 horas.
 - Configuración del entorno de desarrollo en Unity (2 días).
 - Integración básica de Vuforia para la detección de marcadores y modelos (2 días).
 - Prueba integración de Vuforia (1 día).

- Diseño e implementación de la interfaz de usuario (5 días).
- **Iteración 4: Integración de los modelos 3D - 8 días, 32 horas.**
 - Escaneado y generación de modelos objetivos usando la herramienta de Vuforia, Model Target Generator (4 días).
 - Ajustes en Unity para la correcta detección y visualización de los modelos 3D (3 días).
 - Pruebas iniciales de integración (1 día).
- **Iteración 5: Interacciones con marcadores y modelos - 20 días, 80 horas**
 - Diseño de las imágenes de los marcadores utilizando GIMP (2 días).
 - Implementación de interacciones con marcadores en Unity usando Vuforia (rotación, zoom, traslación) (5 días).
 - Implementación de interacciones con modelos 3D, con superposición de información digital sobre los modelos impresos (4 días).
 - Implementación de interacciones avanzadas entre varios marcadores (3 días).
 - Implementación de interacciones avanzadas entre marcadores y modelos (4 días).
 - Pruebas de usabilidad y ajustes necesarios (2 días).
- **Iteración 6: Solución de errores y refinamiento - 9 días, 36 horas**
 - Pruebas exhaustivas para identificar y solucionar errores (3 días).
 - Diseño del logo de la aplicación (1 día).
 - Implementar compatibilidad con varios idiomas (2 días).

- Implementación de mejoras basadas en la retroalimentación obtenida (3 días).

- **Memoria del proyecto - 75 días**

La redacción de la memoria y documentación del trabajo se irá desarrollando de manera simultánea al desarrollo de todo el proyecto, es decir, a lo largo de las 300 horas establecidas.

A continuación, se muestra el diagrama de Gantt (Ilustración 28) que representa la planificación temporal completa de todo el proyecto, dividido en iteraciones que a su vez están compuestas por distintas tareas:

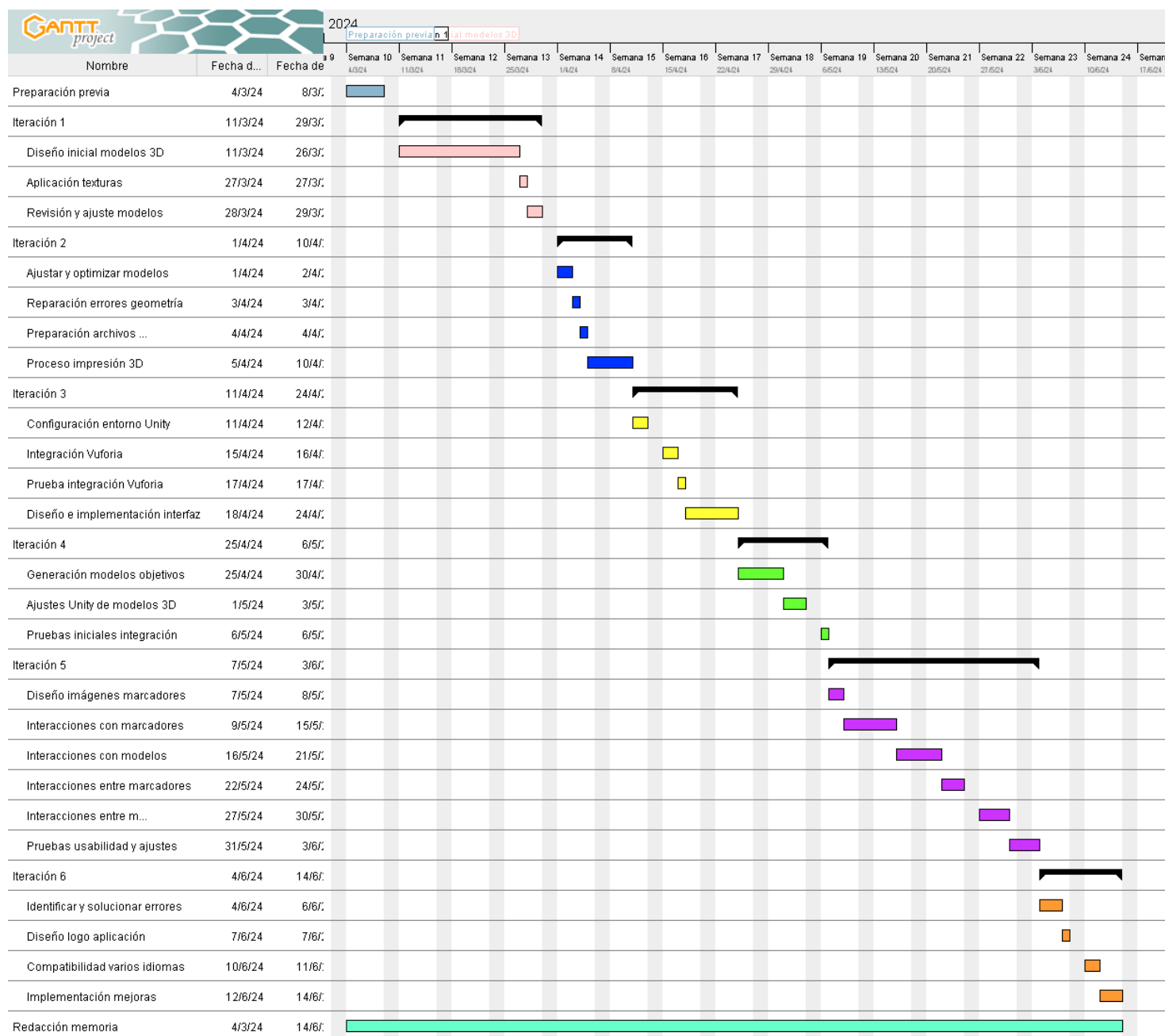


Ilustración 28: Diagrama de Gantt

3.3 Presupuesto

El presupuesto tiene como objetivo determinar y justificar el coste económico del desarrollo del trabajo. Para calcular el coste total de un proyecto software hay que tener en cuenta tanto los costes asociados al software, al hardware y a los recursos humanos. Al ser parte de un TFG no implica un costo económico real, por lo que se hace una estimación.

A continuación, se realiza un desglose de los costes asociados al proyecto en función del software, del hardware y del personal.

3.3.1 Costes del software

En esta sección se determinan las herramientas software utilizadas en el desarrollo del proyecto y su coste asociado. Ninguna de estas herramientas ha supuesto coste adicional, ya que la mayoría ofrecen una versión gratuita y libre o han sido usadas bajo una licencia de estudiante.

En la siguiente tabla se muestra cada una de estas junto con su coste:

Herramienta	Descripción	Coste
Unity	Plataforma de desarrollo de videojuegos y aplicaciones (Licencia de estudiante)	0,00 €
Vuforia Engine	Herramienta para desarrollo de aplicaciones de Realidad Aumentada	0,00 €
Visual Studio	IDE utilizado para la programación	0,00 €
Blender	Herramienta de modelado 3D	0,00 €
Meshmixer	Herramienta de modelado 3D	0,00 €
GIMP	Programa de edición de imágenes	0,00 €
PrusaSlicer	Software de impresión 3D	0,00 €
Coste total:		0,00 €

Tabla 4: Tabla de costes software

3.3.2 Costes del hardware

En este apartado se especifican los recursos materiales y hardware necesario para la realización del trabajo. En este caso han sido necesarios el equipo informático

donde se desarrolla el trabajo, la impresora 3D y material de impresión, y el dispositivo para ejecutar la aplicación de Realidad Aumentada.

El equipo informático usado es un portátil de la marca HP con Intel Core i7 y 16 GB de RAM, con un coste aproximado de 750€.

En cuanto a los recursos hardware necesarios para la impresión 3D, se ha necesitado una impresora 3D y el material plástico necesario para la impresión de los modelos. La impresora utilizada es la Prusa i3 MK3S con multimaterial integrado, con un precio de unos 800€. El material utilizado para la impresión ha sido PLA de 1.75 mm con un coste estimado de 15€/kg. En este caso se ha usado alrededor de unos 500 gramos.

Por otro lado, para la ejecución de la aplicación de Realidad Aumentada es necesario un dispositivo compatible con esta tecnología y con una versión de Android 8.0 o superior. El dispositivo que se ha adquirido en este caso ha sido la Tablet Galaxy Tab s6 lite de la marca Samsung, adquirida en noviembre de 2023 y con un precio de 290€.

Hay que tener en cuenta que estos no son los costes finales ya que hay que amortizar estos dispositivos en el tiempo. A continuación, se muestra una tabla con el desglose de los costes según su amortización, teniendo en cuenta que el uso de estos dispositivos para el desarrollo de este trabajo ha sido de 4 meses:

Componente hardware	Vida útil / unidades	Coste
Portátil HP	5 años (60 meses)	$750\text{€} \rightarrow 750/60 * 4 = 50,00\text{€}$
Impresora Prusa i3 MK3S	5 años (60 meses)	$800\text{€} \rightarrow 800/60 * 4 = 53,33\text{€}$
PLA 1.75mm	500 g	$500\text{g} * 15\text{€} / 1000 \text{ g} = 7,50\text{€}$
Samsung Galaxy Tab s6 lite	4 años (48 meses)	$290\text{€} \rightarrow 290/48 * 4 = 24,16\text{€}$
Coste total:		131,99€

Tabla 5: Tabla de costes hardware

3.3.3 Costes del personal

En esta sección se describen los costes asociados a los recursos humanos necesarios para desarrollo del proyecto. En este caso, todos los puestos de trabajo necesarios son desarrollados por el alumno, que asume los diferentes roles necesarios.

Para este proyecto se necesitan los siguientes puestos o roles de trabajo:

- **Analista Programador:** Se encarga del análisis de requisitos y la planificación del sistema.
- **Diseñador gráfico:** Responsable del diseño de la interfaz de usuario, de los marcadores y de los modelos 3D.
- **Programador:** Se encarga de la implementación del sistema, en este caso la aplicación.
- **Gestor de proyectos:** Supervisa el proyecto y elabora la documentación necesaria.

Según el “XVIII Convenio colectivo estatal de empresas de consultoría, tecnologías de la información y estudios de mercado y de la opinión pública” [42], el salario base de un Analista Programador es de 26.534,12€, el de un Diseñador Gráfico 18.451,80€, el de un Programador graduado 17.079,16€ y el de un Gestor de Proyectos 28.850,23€. Además, es necesario restar un 28,30% al sueldo mensual de los trabajadores para cubrir los gastos de Seguridad Social de la empresa [43].

A continuación, se muestra una tabla con los costes estimados de cada uno de los puestos de trabajo, en función del tiempo, en este caso 4 meses:

Puesto	Salario Base	Tiempo (meses)	Retención anual	Coste
Analista Programador	26.534,12€	4	7.509,16€	6.341,65€

Diseñador gráfico	18.451,80€	4	5.221,86€	4.409,98€
Programador	17.079,16€	4	4.833,40€	7.081,92€
Gestor de proyectos	28.850,23€	4	8.164,62€	6.895,20€
Coste total:				24.728,75€

Tabla 6: Tabla de costes del personal

3.3.4 Coste total

Para determinar el coste final de todo el proyecto, basta con sumar los costes asociados a las distintas categorías: hardware, software y personal:

Recurso	Coste
Software	0,00€
Hardware	131,99€
Personal	24.728,75€
Total:	24.860,74€

Tabla 7: Tabla de costes finales

En conclusión, el presupuesto total del proyecto es de 24.860,74 € para cubrir todos los gastos detallados anteriormente.

4 DISEÑO INICIAL

El objetivo de esta sección consiste en establecer cómo está estructurado el software o sistema propuesto, incluyendo su arquitectura y los modelos de diseño relacionados con su funcionalidad, interfaz y manejo de datos. Este diseño inicial es fundamental en el proceso de desarrollo, ya que será la base sobre la cual se construirá la solución final.

Se consideran aspectos como el análisis de casos de uso, diseño de alto nivel, diseño detallado, etc. También se incluye una descripción del diseño general previo a la primera iteración. El desarrollo de las distintas iteraciones se realizará en los sucesivos apartados.

4.1 Especificaciones del sistema

En este apartado se detalladamente las especificaciones del sistema, basándose en los requisitos funcionales y no funcionales ya definidos anteriormente. Para la elaboración de las especificaciones del sistema, se ha optado por utilizar historias de usuario, ya que permiten una descripción concisa y clara de las necesidades y expectativas del usuario.

A continuación, se presentan algunas de las principales historias de usuario basadas en los requisitos preestablecidos:

Historia de usuario 1:

- **Como:** Usuario del sistema.
- **Quiero:** Visualizar un entorno natural en 3D mediante una maqueta física.
- **Para:** Obtener una representación tangible del entorno que pueda ser enriquecida con información digital a través de Realidad Aumentada.

Historia de usuario 2:

- **Como:** Usuario del sistema.

- **Quiero:** Utilizar una aplicación de Realidad Aumentada que superponga información digital sobre una maqueta 3D.
- **Para:** Mejorar mi comprensión del entorno natural mediante la integración de datos visuales adicionales.

Historia de usuario 3:

- **Como:** Encargado del proyecto.
- **Quiero:** Asegurarme de que el sistema sea fácil de usar y eficiente en la superposición de información.
- **Para:** Garantizar que los usuarios puedan interactuar con el entorno 3D sin ninguna dificultad.

Los casos de uso detallan la interacción entre los usuarios y el sistema en diversas situaciones. Los actores son las personas que usan el sistema para realizar distintas actividades. A continuación, se describen los principales casos de usos del sistema:

Visualizar instrucciones de funcionamiento:

- **Descripción:** Permite a los usuarios acceder a instrucciones detalladas sobre cómo usar la aplicación y sus funcionalidades.
- **Actor:** Usuario.

Seleccionar modelos 3D:

- **Descripción:** Permite a los usuarios seleccionar y cargar modelos 3D en la aplicación para su visualización y manipulación.
- **Actor:** Usuario.

Interacción con marcadores:

- **Descripción:** Permite a los usuarios interactuar con marcadores físicos para superponer información adicional en el entorno e interactuar con los modelos 3D.
- **Actor:** Usuario.

Visualización de información adicional:

- **Descripción:** Permite a los usuarios interactuar con los modelos 3D y mostrar información adicional en pantalla, proporcionando detalles y datos relevantes.
- **Actor:** Usuario.

Interacción entre marcadores y modelos:

- **Descripción:** Permite a los usuarios interactuar simultáneamente con marcadores físicos y modelos 3D.
- **Actor:** Usuario.

Cambiar el idioma de la aplicación:

- **Descripción:** Permite a los usuarios cambiar el idioma de la interfaz de la aplicación.
- **Actor:** Usuario.

Los diagramas de cada caso de uso serán representados de manera gráfica en la [Sección 4.2.3](#).

4.2 Análisis y diseño del sistema

En esta sección se realiza el análisis y diseño del sistema en distintas subsecciones, donde se detallan en profundidad la arquitectura del sistema, los

diagramas de clases, de casos de uso y de secuencia, y el diseño de la interfaz y storyboards.

4.2.1 Diseño arquitectónico del sistema

La arquitectura de la aplicación de Realidad Aumentada, desarrollada en Unity utilizando Vuforia, está estructurada en tres capas de manera que se garantice la modularidad y eficiencia en la interacción con los modelos 3D y la visualización de información adicional. Las capas son las siguientes:

- **Capa de presentación:** Esta capa proporciona la interfaz de usuario y permite la interacción con la aplicación. En ella se incluye la visualización de las instrucciones de uso y la interacción con los modelos 3D.
- **Capa de lógica de negocio:** En esta capa se gestionan las operaciones principales de la aplicación, donde se incluyen la carga de modelos 3D, la interacción con marcadoras y la presentación de información adicional.
- **Capa de integración:** Esta capa facilita la comunicación y el intercambio de datos entre las distintas partes del sistema, permitiendo la importación y exportación de datos si es necesario.

4.2.2 Diagramas de clases

Un diagrama de clases es una representación visual de la estructura del sistema, mostrando las clases que lo componen, junto con sus atributos, métodos y relaciones entre ellas. En este caso, cada clase representa un script, junto con sus atributos y funciones. Estas clases no se relacionan entre sí, sino que se relacionan con los distintos componentes de la aplicación de Unity. A continuación, se muestra el conjunto de clases correspondiente a la aplicación de Realidad Aumentada:



Ilustración 29: Conjunto de clases

4.2.3 Diagramas de casos de uso

Los diagramas de casos de uso describen las interacciones entre los usuarios (actores) y el sistema (aplicación) para conseguir un objetivo específico. A continuación, se muestra el diagrama de los casos de uso descritos en secciones anteriores:

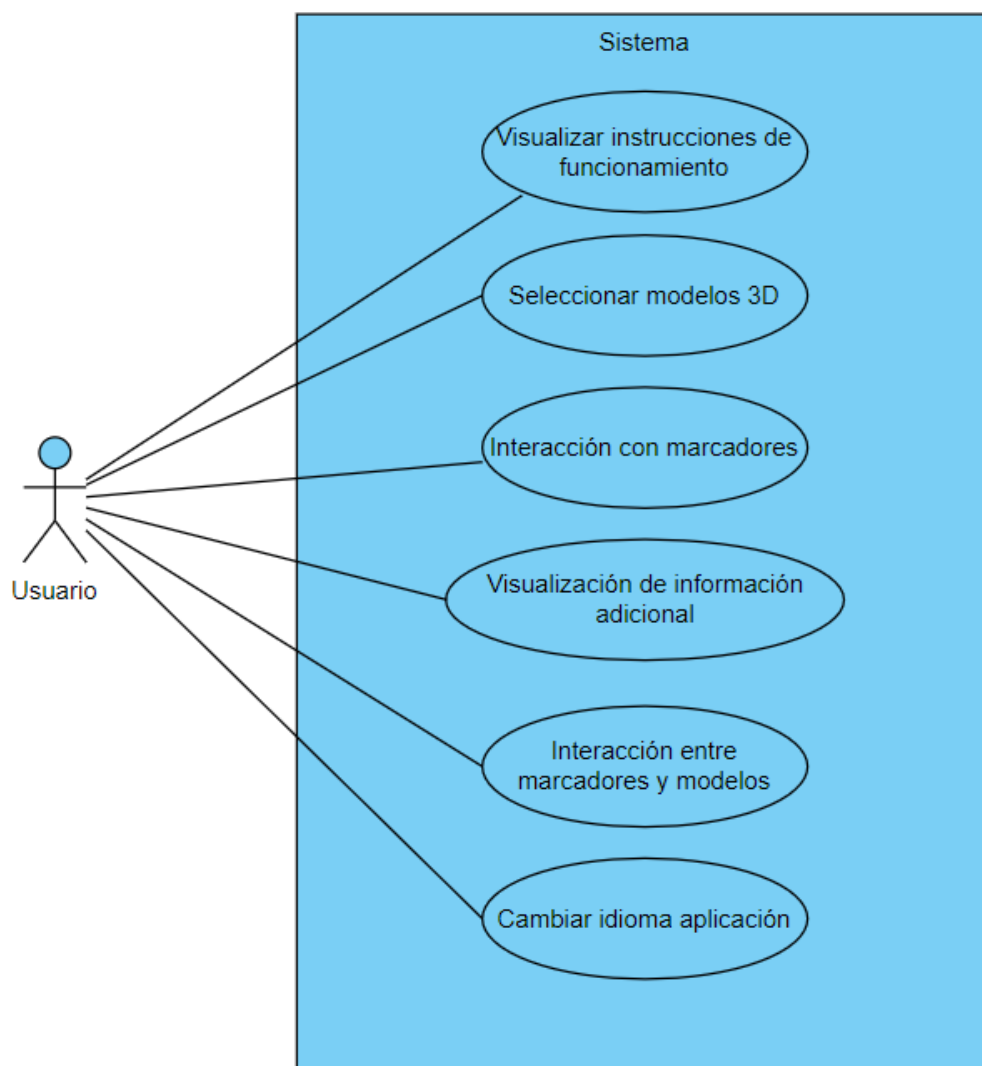


Ilustración 30: Diagrama de casos de uso

4.2.4 Diagramas de secuencia

Los diagramas de secuencia son una herramienta visual utilizada en el desarrollo software para describir la interacción entre los diferentes componentes de un sistema a lo largo del tiempo. Representan la secuencia de mensaje y eventos que se envían entre los objetos o actores.

En este apartado, se presentan los diagramas de secuencia correspondientes a cada uno de los casos de uso de la aplicación, el actor representa a los usuarios y el sistema a la aplicación.

- 1. Visualizar las instrucciones de funcionamiento:** Permite a los usuarios acceder a las instrucciones que detallan cómo usar la aplicación.

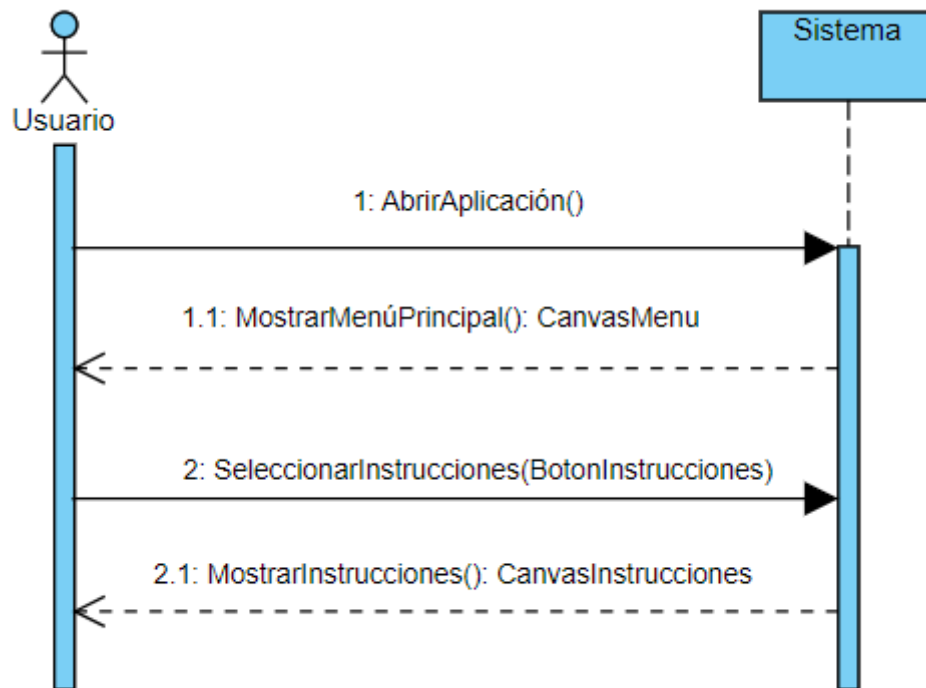


Ilustración 31: Diagrama de secuencia Visualizar Instrucciones

- 2. Seleccionar modelos 3D:** Permite a los usuarios seleccionar y cargar modelos 3D en la aplicación para su visualización y manipulación.

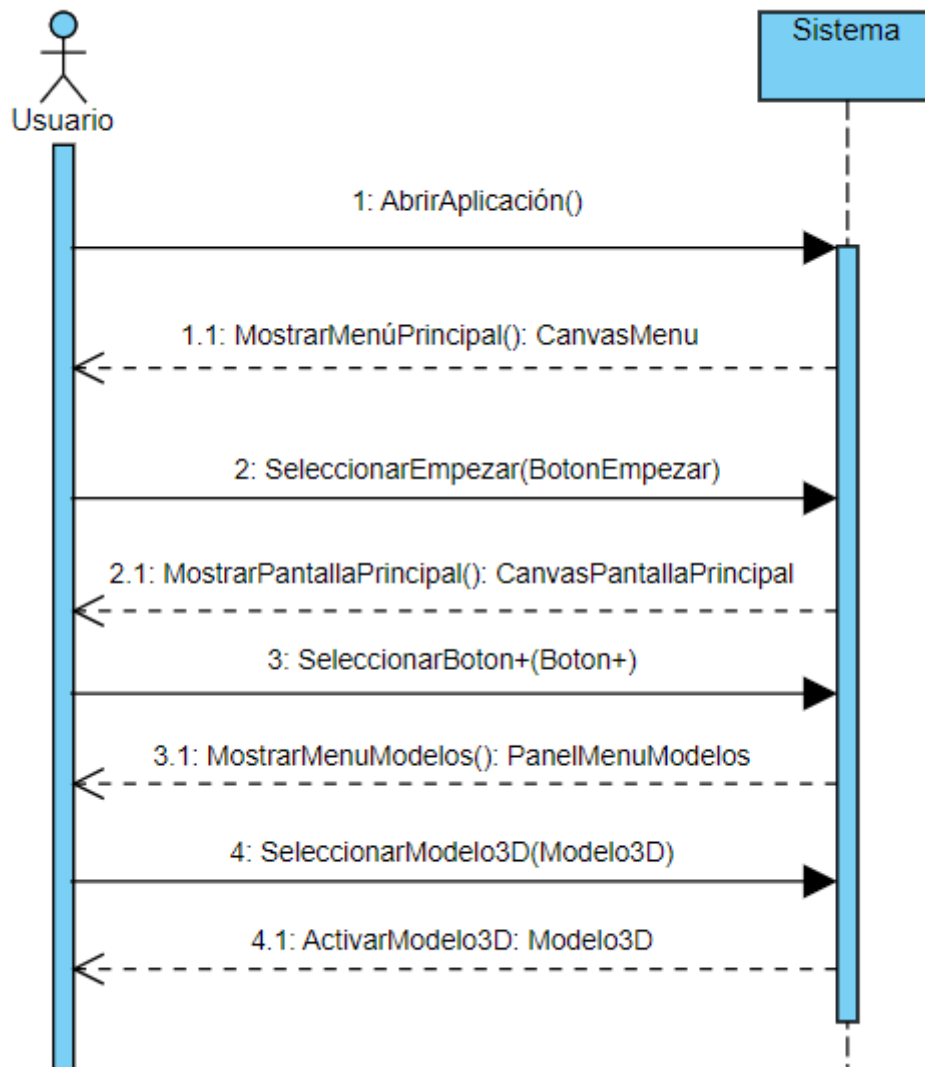


Ilustración 32: Diagrama de secuencia Seleccionar modelos 3D

3. Interacción con marcadores: Permite a los usuarios interactuar con marcadores físicos para superponer información adicional en el entorno con los modelos 3D.

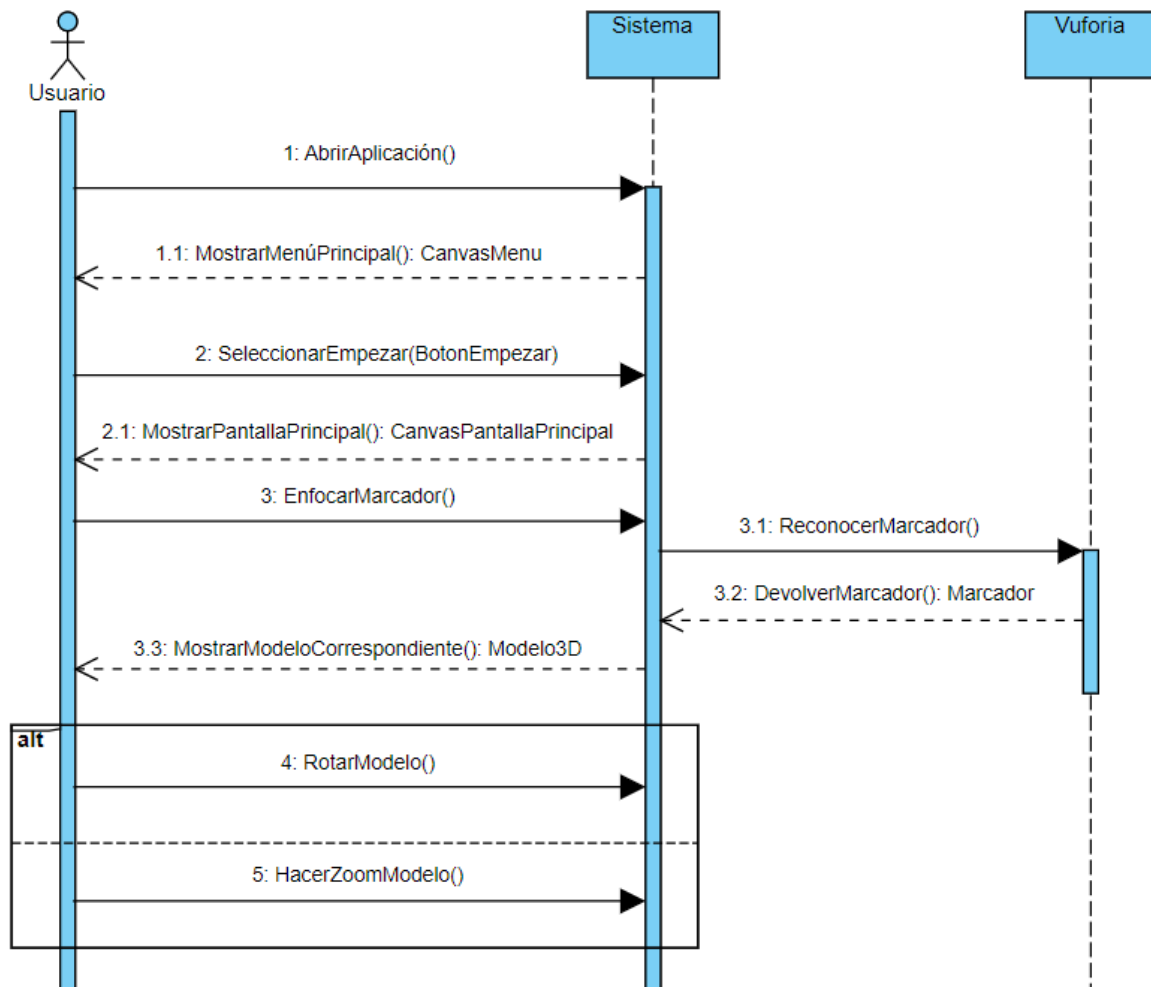


Ilustración 33: Diagrama de secuencia Interacción con marcadores

- 4. Visualización de información adicional:** Permite a los usuarios interactuar con los modelos 3D y mostrar información adicional en pantalla, proporcionando detalles y datos relevantes.

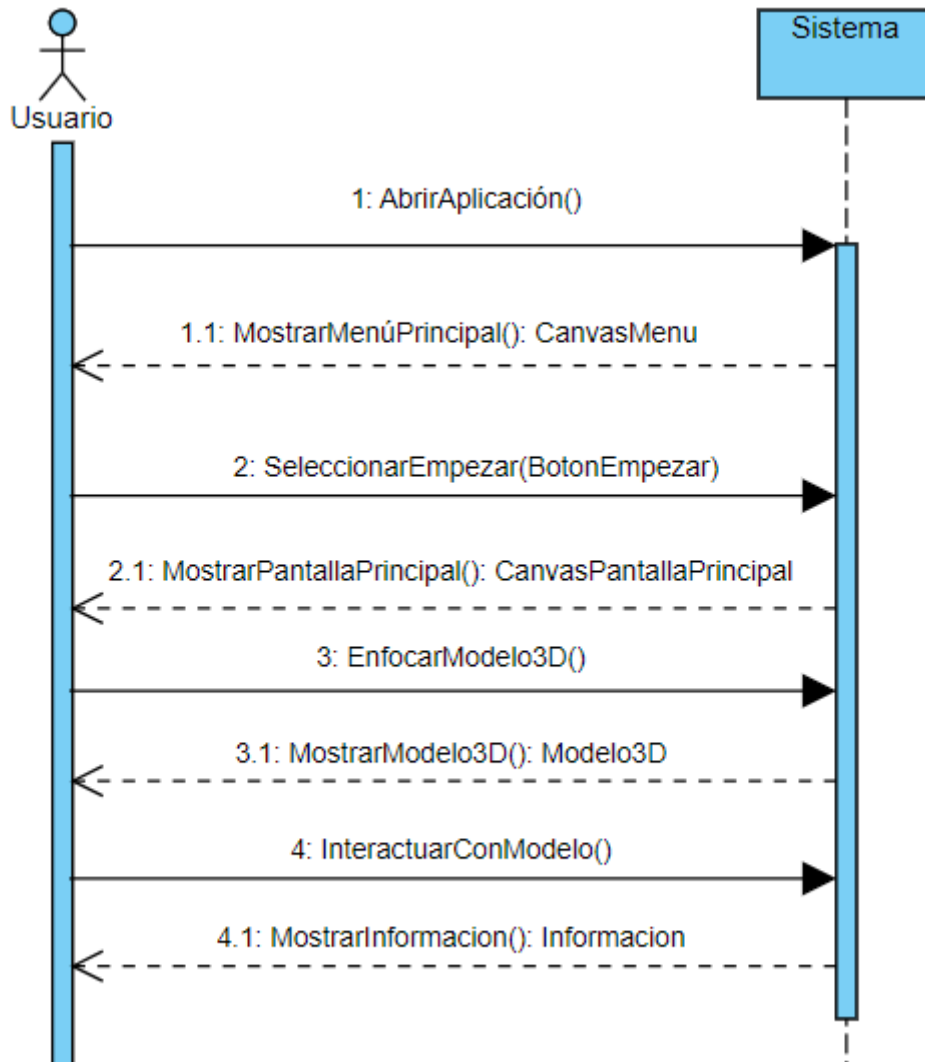


Ilustración 34: Diagrama de secuencia Visualización de información adicional

5. Interacción entre modelos y marcadores: Permite a los usuarios interactuar simultáneamente con marcadores físicos y modelos 3D.

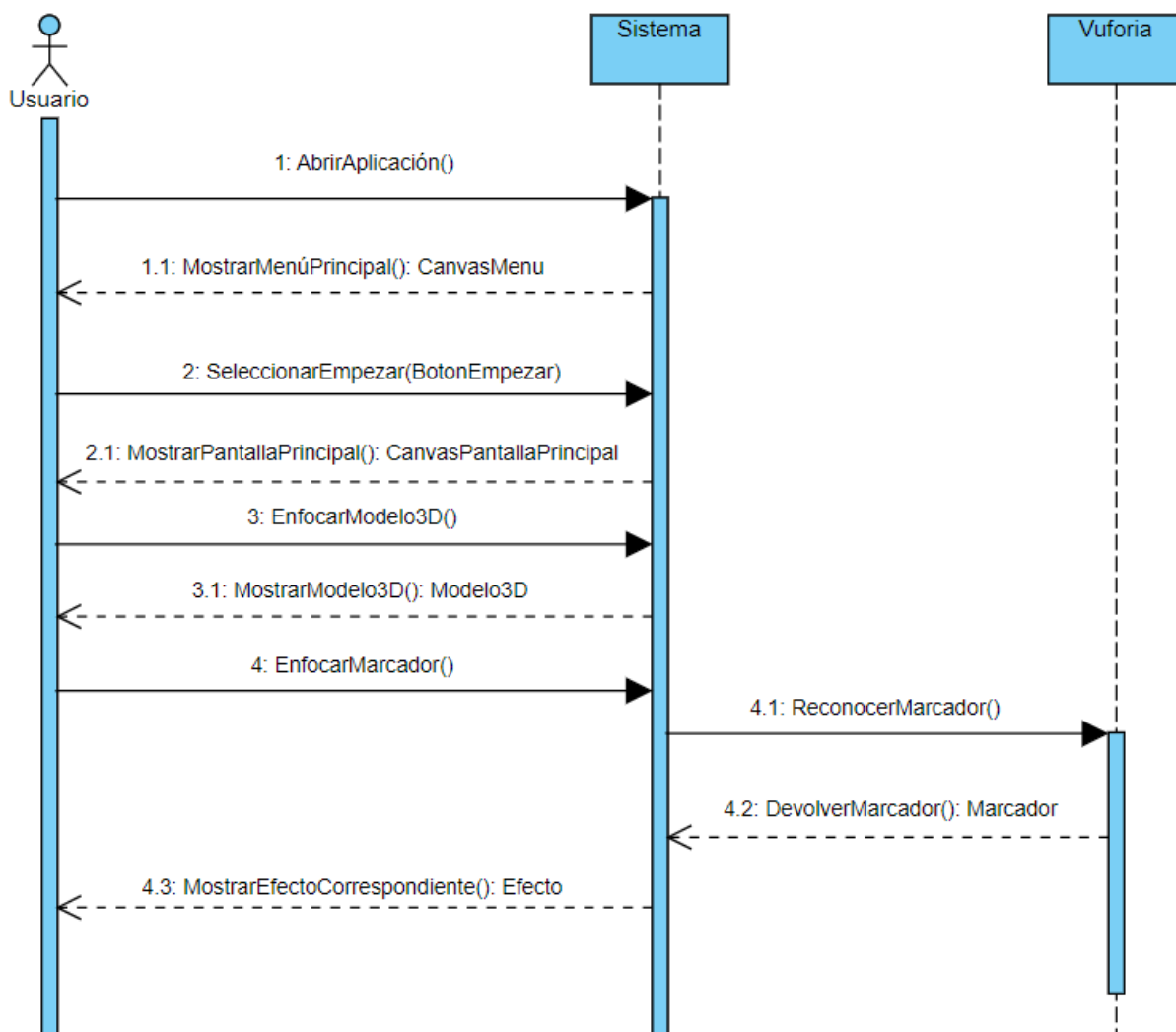


Ilustración 35: Diagrama de secuencia Interacción entre modelos y marcadores

- 6. Cambiar idioma de la aplicación:** Permite a los usuarios cambiar el idioma de la interfaz de la aplicación.

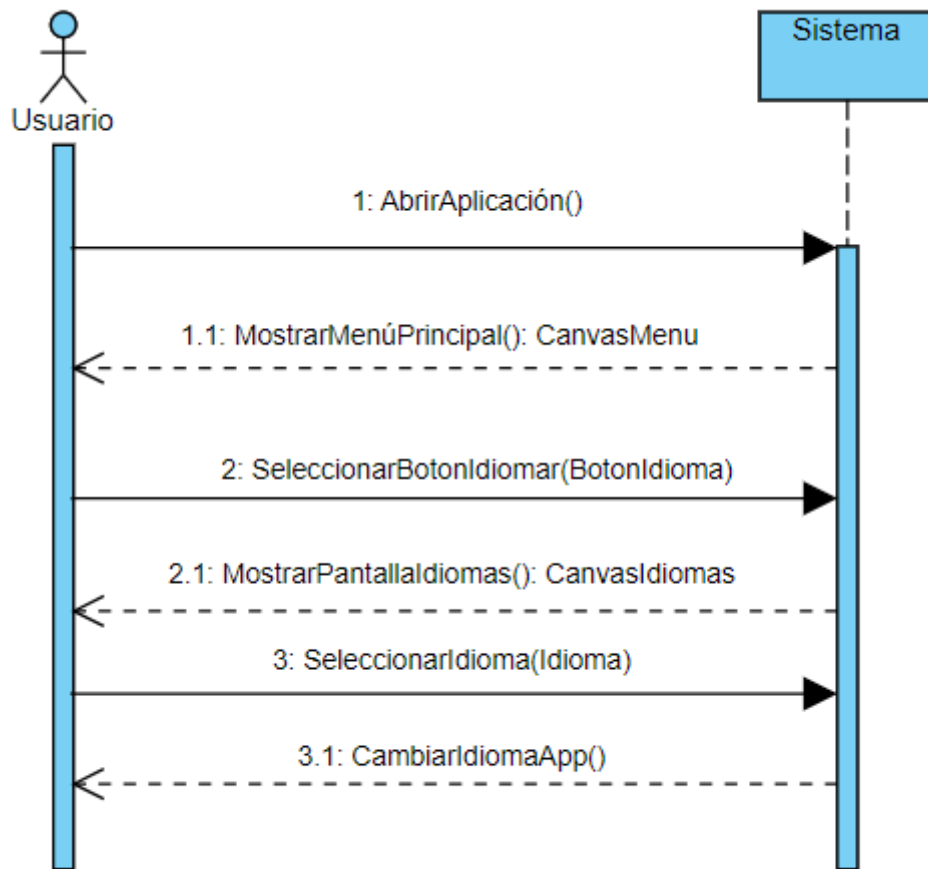


Ilustración 36: Diagrama de secuencia Cambiar idioma de la aplicación

4.2.5 Diseño de la interfaz

El diseño de la interfaz es fundamental para garantizar una experiencia de usuario intuitiva y eficiente. En esta sección se describe la estructura visual y funcional de la aplicación, mediante el uso de bocetos que representan el aspecto visual que tendrá la aplicación.

La interfaz de usuario de la aplicación de Realidad Aumentada contará con las siguientes pantallas:

- Menú principal.
- Pantalla idiomas.

- Pantalla de instrucciones.
- Interfaz principal de la aplicación.

La pantalla del menú principal contará con un botón para ver las instrucciones de uso, otro para cambiar el idioma de la aplicación y otro para comenzar con la aplicación:

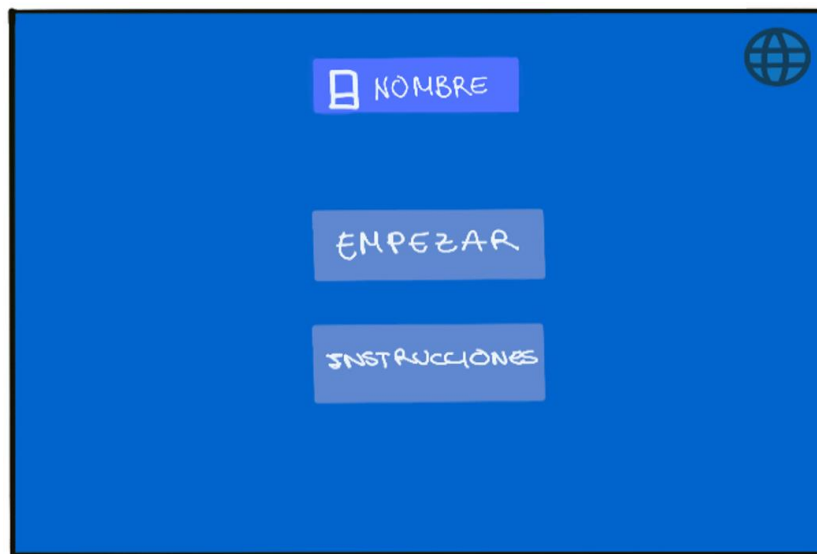


Ilustración 37: Interfaz menú principal

La pantalla para cambiar de idioma tendrá dos botones con las banderas de cada uno y un botón para cerrar esta pantalla:



Ilustración 38: Pantalla para cambiar el idioma

La pantalla con las instrucciones será un panel tipo carrusel en el que deslizando se van mostrando las diferentes instrucciones de uso de la aplicación:

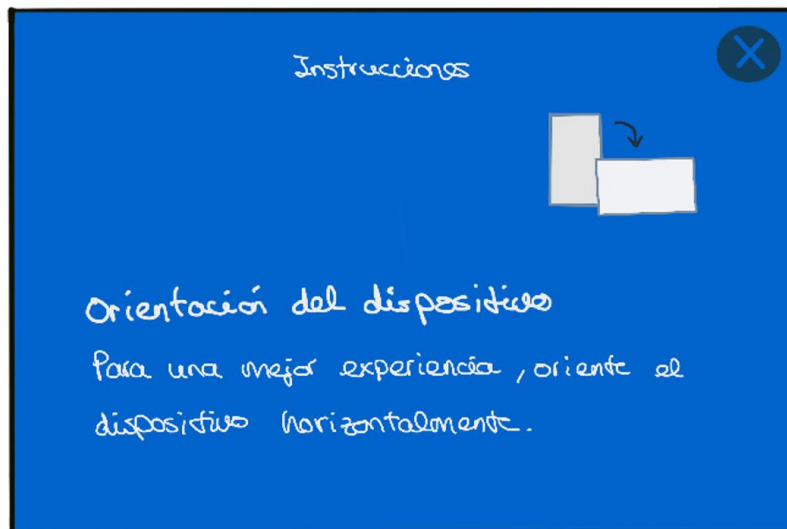


Ilustración 39: Interfaz de las instrucciones

La interfaz principal de la aplicación contará con un botón para volver hacia atrás, otro para mostrar las instrucciones y otro para mostrar los distintos modelos 3D disponibles:

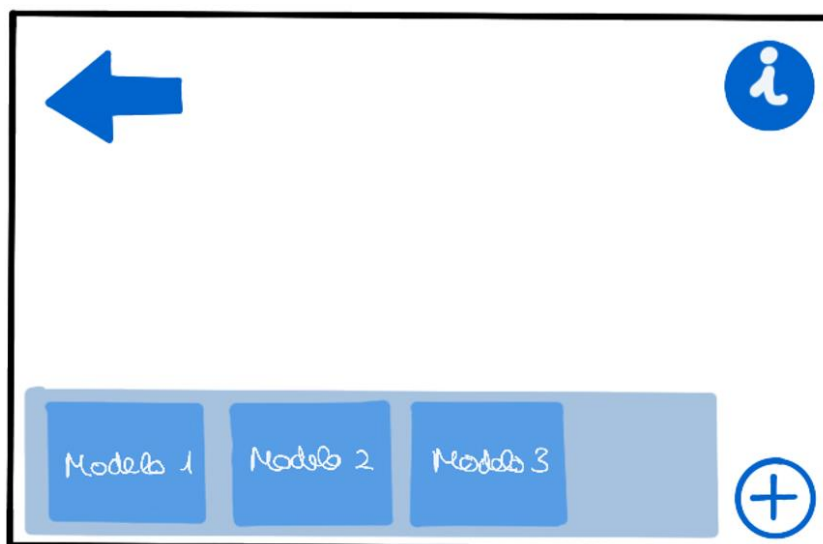


Ilustración 40: Interfaz principal de la aplicación

En esta interfaz también aparecerán paneles en pantalla con información relevante sobre los distintos modelos al pulsar sobre ciertas zonas específicas de los modelos 3D:



Ilustración 41: Interfaz principal con panel informativo

5 DESARROLLO

En esta sección se describe el proceso de desarrollo del prototipo de aplicación de Realidad Aumentada combinada con impresión 3D siguiendo una metodología incremental, basada en iteraciones incrementales. Para cada iteración se documenta con detalle todo el proceso de desarrollo seguido durante dicha iteración para obtener los resultados esperados.

A continuación, se presentan las distintas iteraciones del proyecto, detallando las tareas realizadas y los resultados obtenidos en cada una. Como ya se detalló anteriormente las iteraciones del proyecto son las siguientes:

- **Iteración 1:** Diseño de Modelos 3D
- **Iteración 2:** Preparación e Impresión 3D
- **Iteración 3:** Diseño e implementación de la aplicación móvil de AR
- **Iteración 4:** Integración de los modelos 3D
- **Iteración 5:** Interacciones con marcadores y modelos
- **Iteración 6:** Solución de errores y refinamiento

5.1 Primera Iteración

Esta primera iteración tiene como objetivo diseñar y modelar en 3D el entorno natural escogido para el desarrollo de este proyecto y su utilización en el prototipo de aplicación de Realidad Aumentada. En esta iteración se realiza principalmente el diseño inicial de los modelos, así como añadir las texturas correspondientes y su posterior revisión y ajuste necesarios.

Para este trabajo, se ha elegido el entorno natural del Puente del Obispo, situado sobre el río Guadalquivir, en una pedanía de Baeza. Se ha partido del modelo 3D del entorno de este puente que ha sido escaneado usando drones con sensores LIDAR.

Este modelo 3D original tiene ciertos inconvenientes para ser impreso en 3D directamente, como la existencia de partes voladizas, zonas de mucha densidad como algunos arbustos que dificultan la impresión 3D y algunas partes del modelo quedaban sueltas en el aire (Ilustración 42).

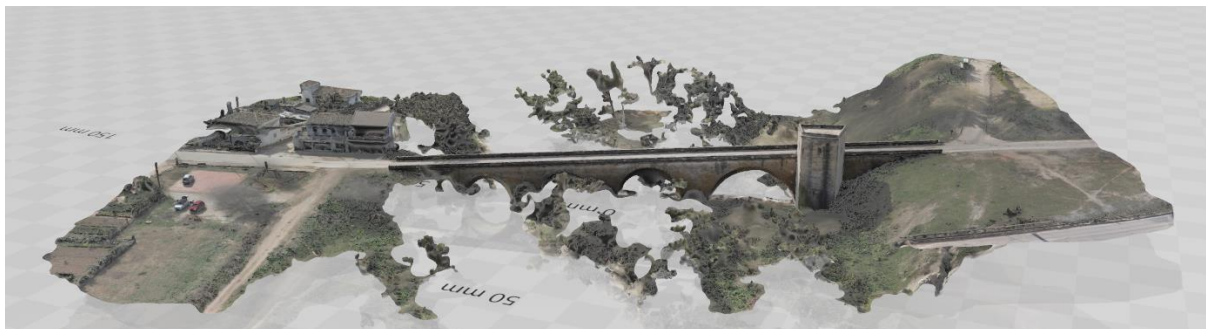


Ilustración 42: Modelo original Puente del Obispo

Por ello, ha sido necesario realizar cambios y ajustes significativos en este modelo original para poder obtener al final un modelo 3D correcto para su impresión. Aparte del modelo del entorno completo del Puente del Obispo, también se han modelado por separado la torre o ermita y el puente solo, ya que estas partes del modelo se usarán en la aplicación para verlas con más detalle.

A continuación, se describe con más detalle cada uno de estos modelados 3D, diseñado y modelados mediante la herramienta Blender.

5.1.1 Modelado del Puente del Obispo

En primer lugar, se ha diseñado y modelado todo el entorno del Puente del Obispo, con el objetivo de obtener un modelo 3D apto para su impresión 3D a la vez que un modelo 3D óptimo para su visualización en la aplicación de Realidad Aumentada.

A continuación, se detalla paso a paso todo el procedimiento desarrollado para la obtención de modelo final.

En primer lugar, se ha eliminado toda la geometría que no es relevante y dificultaba el proceso de impresión, como la zona de arbustos de alrededor del puente. Para eliminar estas partes del modelo, se ha utilizado Blender en el modo de editar, y se han ido seleccionado y eliminado los vértices de estas zonas. Para acelerar y

facilitar este proceso se ha seleccionado el modo de rayos X. También se le ha añadido una base de manera que se pueda colocar en superficies planas y no queden zonas en el aire. Al principio la base es rectangular y se ha ido recortando según la forma del modelo para que quede mejor visualmente.

Posteriormente, utilizando la herramienta para esculpir en Blender, he suavizado algunas zonas del modelo, he mejorado la zona de los arcos y he pulido la zona exterior que une la base con el modelo. Además, he ido añadiendo pequeños detalles que contiene el puente que no habían sido capturados en el escaneo del dron, como por ejemplo algunos pilares que tienen las columnas de los arcos. Para realizar todo esto se ha utilizado tanto la opción de extruir geometría ya existente como la opción de añadir nuevas formas geométricas.

También se han añadido detalles a mano, como los escudos y la puerta de la fachada de la torre del puente. Para añadirlos se han usado imágenes reales de estos elementos y se les han añadido vértices para poder darles forma y tridimensionalidad. En la Ilustración 43 se pueden observar estos elementos con más detalle.



Ilustración 43: Fachada de la torre del Puente del Obispo

Una vez modelado, se ha reducido el número de polígonos utilizando la herramienta de decimar en Blender para reducir el tamaño y peso del modelo. Por último, se ha cerrado la malla por si quedaba algún hueco o agujero mediante la herramienta Meshmixer.

5.1.2 Modelado de la torre

Para obtener un modelo 3D con la torre del puente únicamente, se ha partido del modelo completo del puente y utilizando la herramienta de recorte y bisección que proporciona Blender, se ha ido recortado según la forma de la torre y se ha ido eliminando el resto de la geometría. Una vez recortada, se ha pulido toda la superficie y se han suavizado algunos detalles que sobresalían, mediante la herramienta de esculpir. Para darle un aspecto más visual, se ha optado por darle una forma circular a la base.

En las siguientes ilustraciones, podemos ver cómo queda el modelo 3D de la torre del Puente del Obispo, tanto por delante (Ilustración 44) como por detrás (Ilustración 45)



Ilustración 44: Modelo 3D de la torre por delante



Ilustración 45: Modelo 3D de la torre por detrás

5.1.3 Modelado del puente

Al igual que el modelo que hemos obtenido de la torre, se ha seguido el mismo procedimiento en Blender para recortar únicamente la zona del puente propiamente dicho, eliminado el resto del entorno que le rodea. Como se ha comentado en el apartado anterior, se han utilizado las herramientas de recorte y bisección, y se han ido eliminado las zonas restantes. Al final, se han suavizado y pulido las superficies y parte de la geometría para darle un mejor acabado.

En la Ilustración 46, se puede observar cómo queda el modelo 3D del puente recortado.



Ilustración 46: Modelo 3D recortado del puente

A ambos modelos se les ha aplicado la herramienta de decimar para reducir el número de triángulos de cada malla, de manera que se reduzca su peso sin dañar ningún detalle.

5.1.4 Aplicación de texturas

Una vez han sido diseñado y creados todos los modelos 3D, hay que añadirles sus texturas correspondientes para mejorar su apariencia visual. Para realizar esto se ha usado la herramienta de edición de imágenes GIMP.

Las imágenes de las texturas se han ido añadiendo a cada malla o elemento correspondiente del modelo, en forma de material, mediante el uso de Blender. Para realizar esto, en la pestaña Shading de Blender, cada material tiene asociados unos nodos en los que se asigna cada imagen para la textura y se configuran los parámetros para ajustar valores como la transparencia, el metalizado, la aspereza, etcétera.

Aquí podemos ver un ejemplo del modelo del Puente del Obispo sin ninguna textura (Ilustración 47):

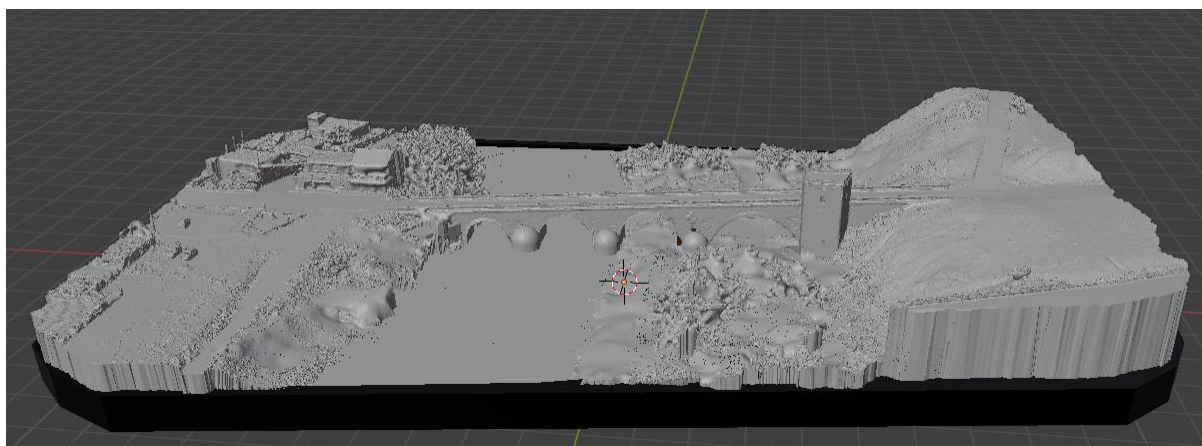


Ilustración 47: Modelo Puente del Obispo sin texturas

Y aquí lo podemos ver con las texturas ya aplicadas (Ilustración 48):

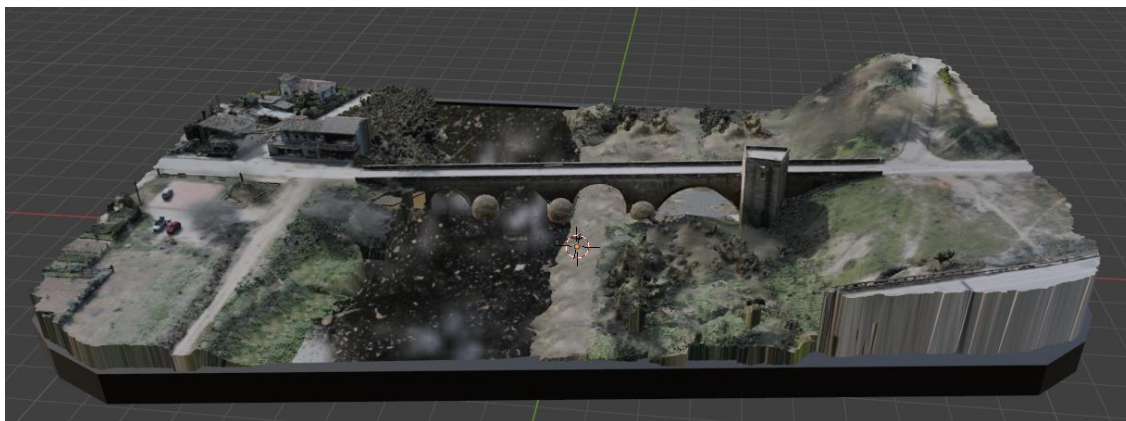


Ilustración 48: Modelo Puente del Obispo con texturas

Los modelos deben estar optimizados y no tener un tamaño muy pesado para no ocasionar problemas al cargarlos en la aplicación de Realidad Aumentada y obtener así un tiempo eficiente. Por ello, para reducir el tamaño de estos modelos, he comprimido y reducido las distintas texturas de cada uno de ellos usando GIMP. Para ello he escalado cada imagen con un tamaño de 1024x1024 píxeles o menor en cada caso, he exportado estas imágenes en formato JPG con una calidad de 90 y he editado el archivo GLTF de cada modelo para sustituir los formatos de cada textura por los nuevos, ya comprimidos.

Aquí podemos ver un ejemplo de la imagen de una textura antes de ser escalada (Ilustración 49):

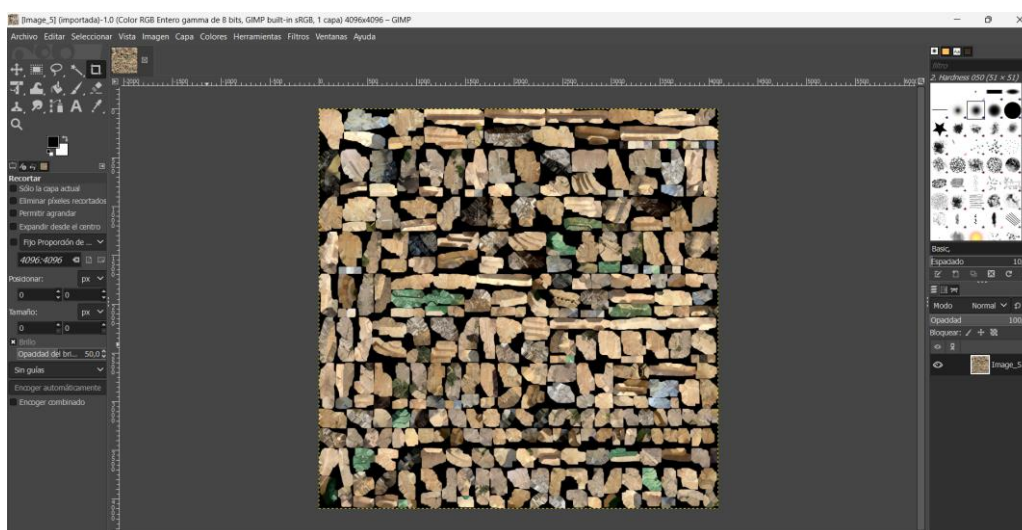


Ilustración 49: Textura en GIMP sin escalar

Y aquí podemos ver la misma imagen escalada a 1024x1024 píxeles (Ilustración 50):

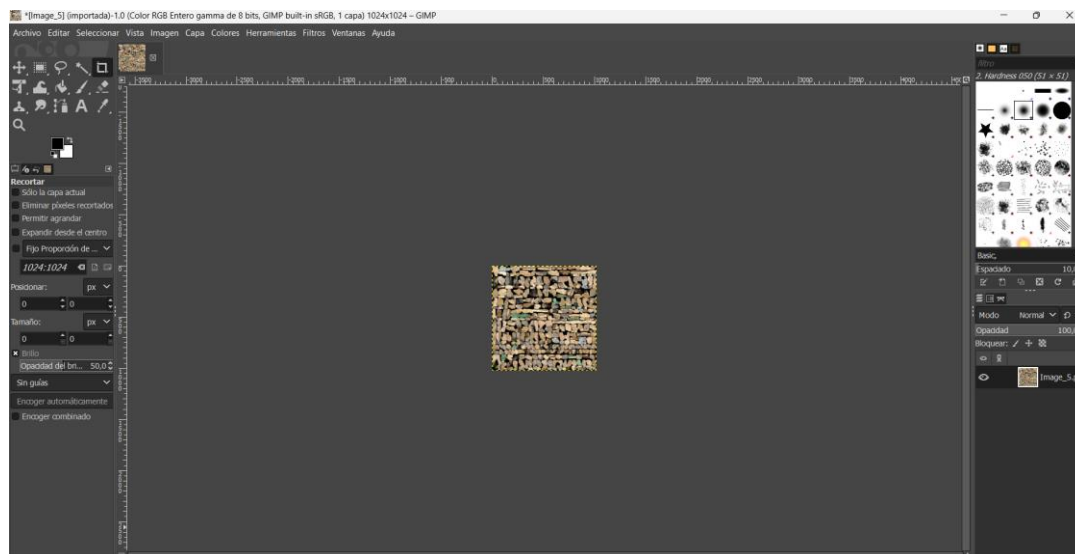


Ilustración 50: Textura en GIMP escalada

Con esto se consigue reducir considerablemente el tamaño de cada imagen, reduciendo así el tamaño final de cada modelo 3D.

5.1.5 Revisión y ajuste de los modelos

Después de diseñar y texturizar los modelos, es importante asegurarse que los detalles de todos los modelos estén correctos y que dichos modelos cumplan con los estándares de calidad y no contengan ningún tipo de error que pueda afectar negativamente a la impresión 3D.

Para ello, se ha llevado a cabo una revisión exhaustiva de cada modelo para detectar y corregir pequeños errores en la geometría. Los principales defectos que se han encontrado en cada modelo han sido pequeños agujeros o aperturas en las mallas de polígonos. Para arreglar estos errores, he utilizado la herramienta Meshmixer que cuenta con una opción para cerrar y corregir las mallas de los modelos 3D automáticamente.

Además, como ya se ha comentado antes, es importante que los modelos estén optimizados y no tengan un peso excesivo para su correcto y eficiente funcionamiento dentro de la aplicación de Realidad Aumentada. Por ello, se ha realizado un proceso

de reducción del número de polígonos de cada modelo mediante la opción de decimar que incluye la herramienta de Blender. De esta manera, se consigue obtener un tamaño y peso adecuado para cada modelo, reduciendo la cantidad de triángulos que forma cada malla, pero sin llegar a perder calidad.

5.2 Segunda iteración

En esta segunda iteración, una vez están completamente diseñados los modelos 3D, se ha realizado la preparación de estos modelos para su correcta impresión 3D.

En primer lugar, se ha comprobado que los modelos no tuviesen ningún tipo de error o defecto. Para ello, se han realizado distintos ajustes para optimizar la geometría de los modelos y se han corregido algunos errores. Como ya se ha desarrollado antes, algunos de los errores más frecuentes que han tenido que ser solucionados son agujeros o huecos en las mallas de los modelos, que han sido cerrados usando Meshmixer. También se han suavizado algunas de las superficies de los modelos para mejorar su aspecto.

Los modelos obtenidos de Blender tienen el formato GLB, ya que este formato permite el uso de texturas (Ilustración 51). Sin embargo, para poder imprimir estos modelos es necesario convertirlos al formato STL, que no permite el uso de texturas (Ilustración 52).

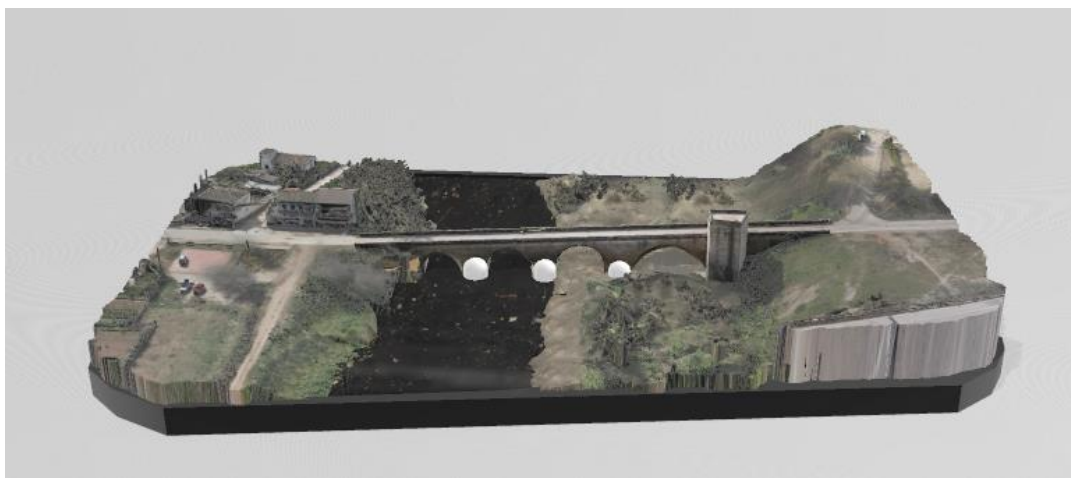


Ilustración 51: Modelo del Puente del Obispo en formato GLB

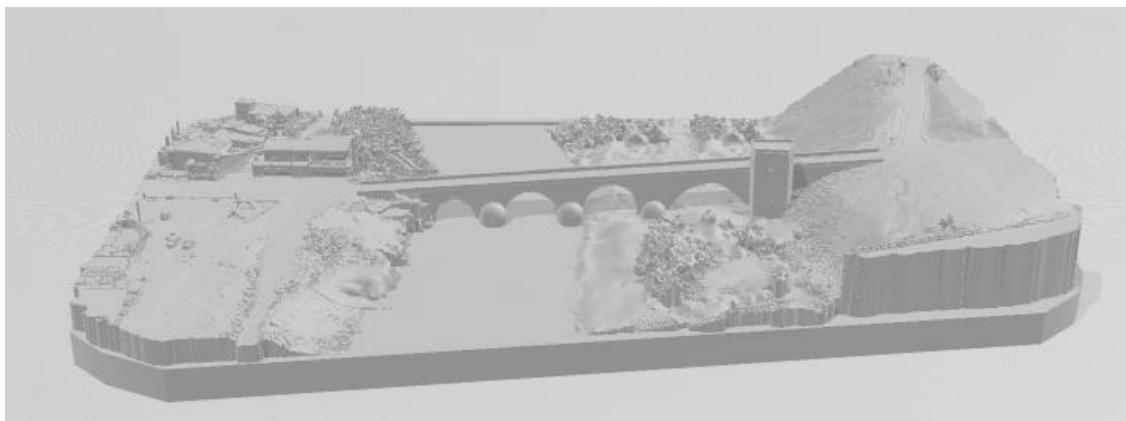


Ilustración 52: Modelo del Puente del Obispo en formato STL

5.2.1 Configuración impresión 3D

Una vez tenemos los modelos en formato STL podemos importarlos directamente en el programa de impresión PrusaSlicer. Aquí podemos ver un ejemplo del modelo del Puente del Obispo en formato STL ya importado en PrusaSlicer (Ilustración 53):

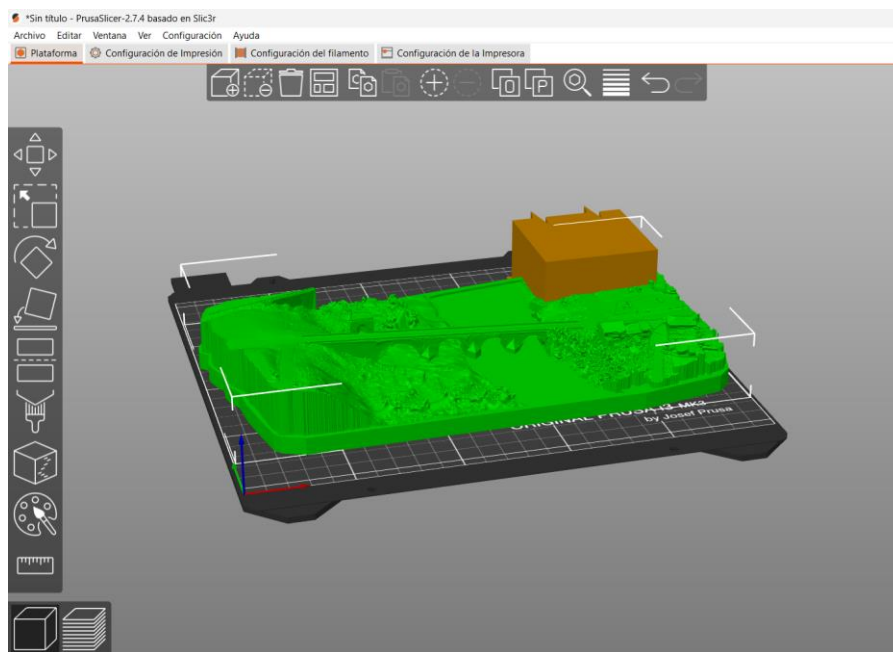


Ilustración 53: Modelo Puente del Obispo en PrusaSlicer

Para garantizar una buena impresión hay que tener en cuenta algunos parámetros de impresión. En el programa PrusaSlicer es necesario configurar los siguientes aspectos dentro del apartado de Configuración de Impresión:

- **Capas y perímetros (Ilustración 54):** Dentro de esta sección podemos configurar los aspectos relacionados con las capas y los perímetros de la impresión [44]. Algunos de los aspectos a destacar son los siguientes:
 - **Altura de la capa:** Define el grosor de cada capa. A mayor altura, menor tiempo de impresión, pero menor detalle. En este caso se ha optado por una altura de 0.2mm que permite obtener un tiempo de impresión razonable sin llegar a perder detalle.
 - **Perímetros:** Define el número mínimo de contornos que forman la pared del modelo. Cuanto mayor sea el número de perímetros mayor será la fuerza del modelo. En este caso se ha establecido en 3.
 - **Capas sólidas:** Define el número de capas sólidas inferior y superior de cada modelo. Cuantas más capas mejor soporte y menor hundimiento. En este caso se han definido tanto 5 capas superiores como inferiores.

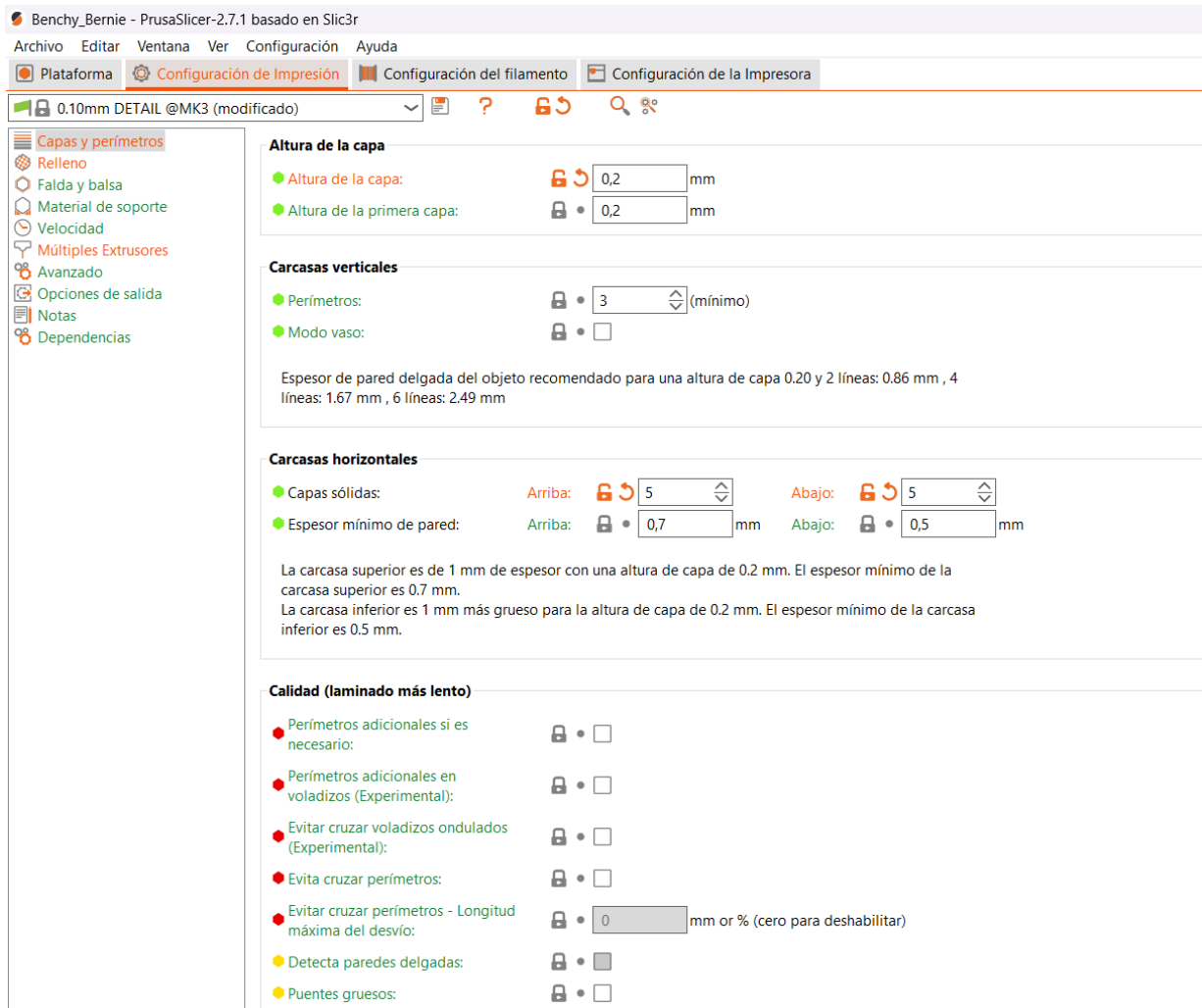


Ilustración 54: Configuración Capas y perímetros de PrusaSlicer

- **Relleno (Ilustración 55):** En esta sección podemos especificar la configuración del relleno del modelo [45]. Los elementos más importantes de esta sección son los siguientes:
 - **Densidad de relleno:** En este caso se ha establecido la densidad del relleno en el 10%, ya que no se busca obtener un modelo pesado, si no uno ligero. Además, de esta manera se consigue reducir también el tiempo de impresión.
 - **Patrón de relleno:** Define la forma en la que se realiza el relleno. En este caso se ha optado por el patrón de panal de abeja ya que permite una mayor resistencia mecánica.

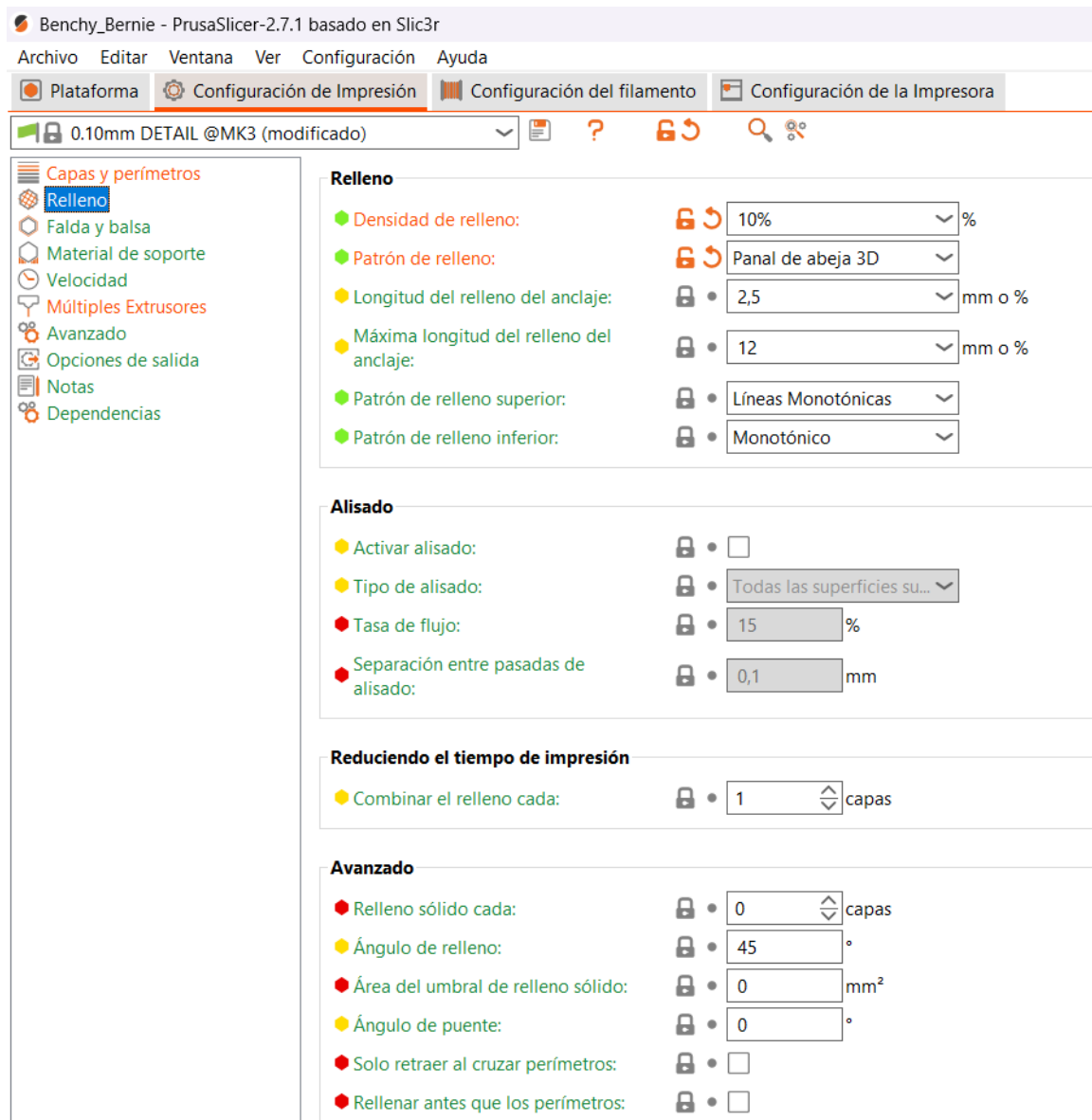


Ilustración 55: Configuración Relleno de PrusaSlicer

- **Material de soporte (Ilustración 56):** Este apartado permite configurar todo lo relacionado con los soportes que son necesarios en algunas zonas del modelo [46]. En el caso del modelo del Puente del Obispo serán necesarios soportes en las zonas de los arcos del puente para evitar hundimientos y roturas en esas partes. Estos son los parámetros más importantes de los soportes:
 - **Generar material de soporte y Soportes generados automáticamente:** Es importante que estas dos opciones estén activadas para poder generar los soportes necesarios.

- **Umbral de voladizos:** Establece el ángulo más horizontal que se puede imprimir sin usar material de soporte. Cuanto menor sea el ángulo, menos soportes se generan. En este caso se ha establecido en 70°.
- **Estilo:** Define la forma de los soportes. En este caso se ha optado por el estilo ajustado, ya que mantiene la forma del voladizo, lo que hace que no se filtren a las paredes.

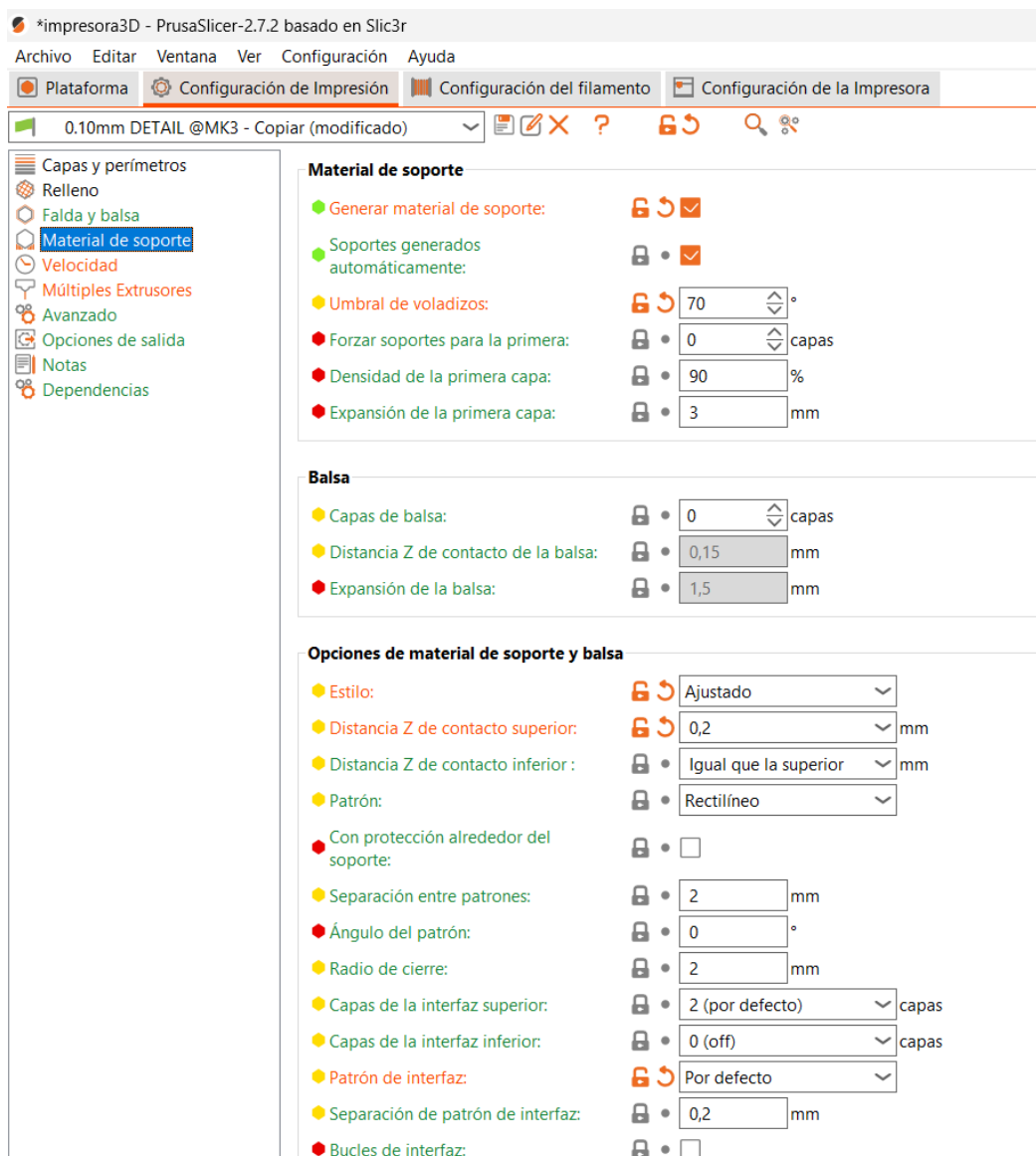


Ilustración 56: Configuración Material de soporte de PrusaSlicer

- **Velocidad (Ilustración 57):** Se han cambiado algunos de los parámetros de la velocidad para acelerar el tiempo de impresión.

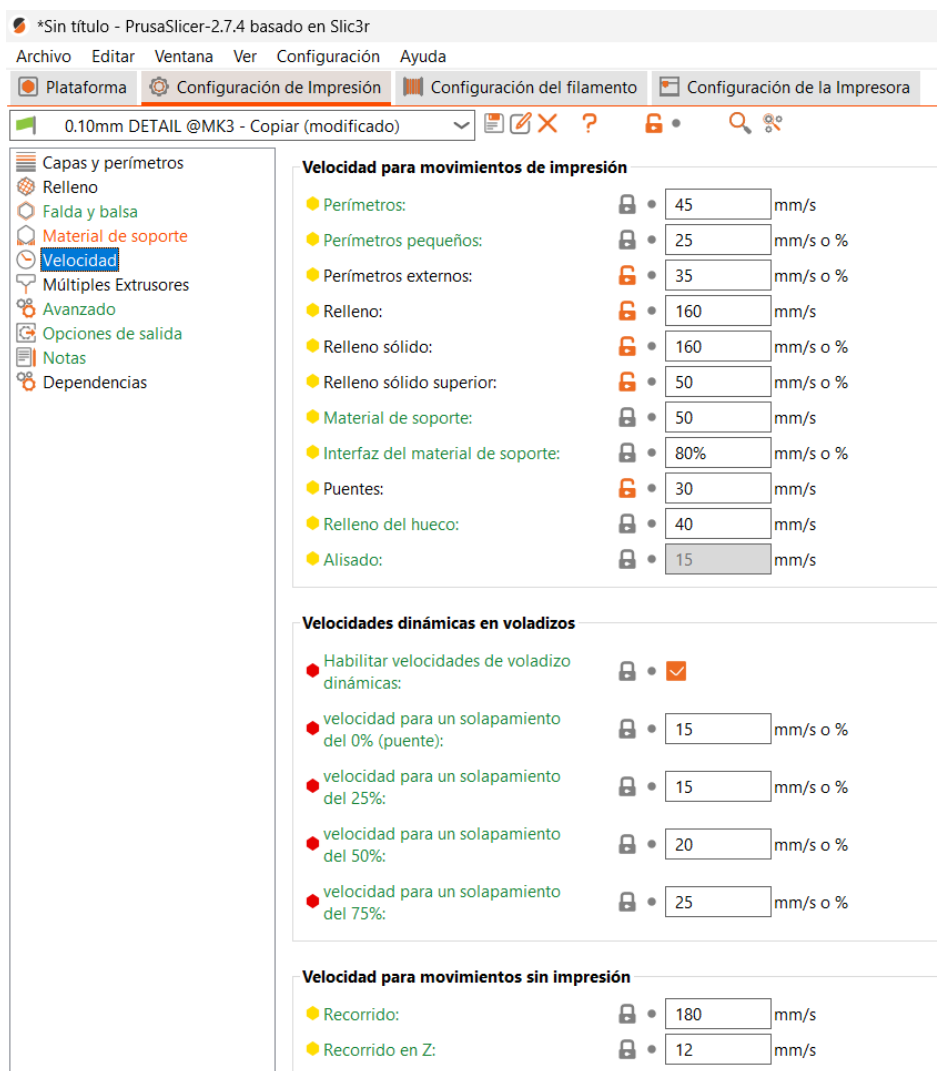


Ilustración 57: Configuración Velocidad de PrusaSlicer

Por otro lado, dentro de la sección Configuración del filamento (Ilustración 58), hay que establecer el diámetro del filamento en 1.75mm, ya que éste es el grosor del material que se usa para la impresión.

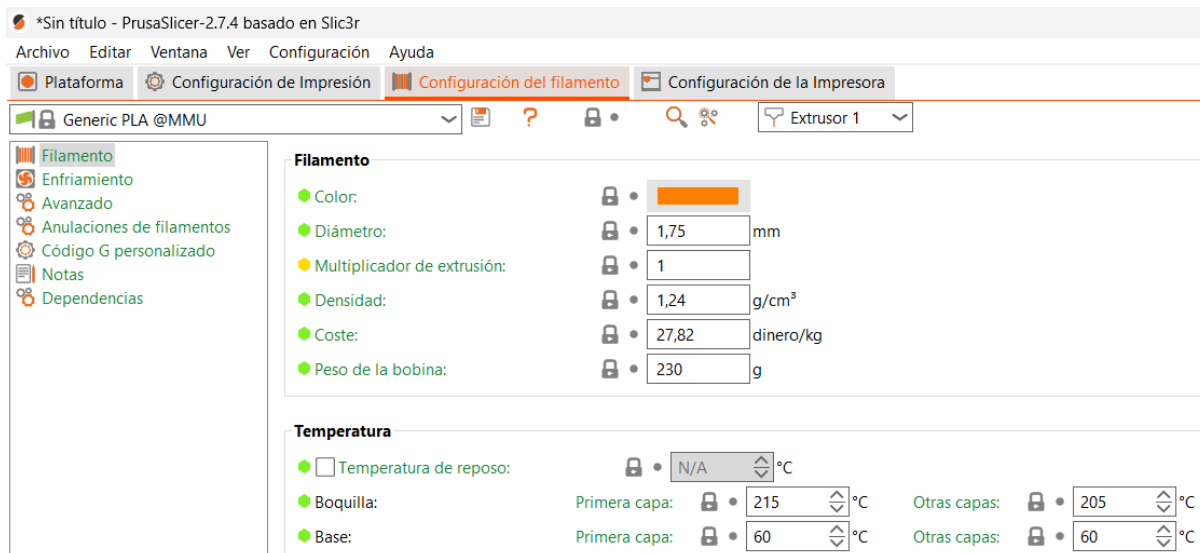


Ilustración 58: Configuración del filamento de PrusaSlicer

Por último, hay que elegir el material Generic PLA para el filamento y seleccionar el extrusor por defecto para el modelo a imprimir (Ilustración 59).

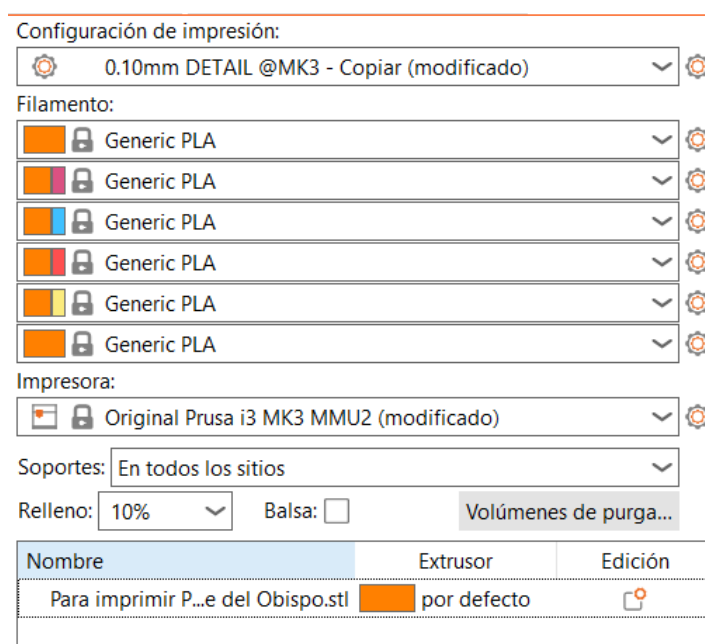


Ilustración 59: Configuración de impresión PrusaSlicer

Una vez configurados todos los parámetros necesarios, podemos laminar el modelo para previsualizar su impresión, así como los distintos tiempos de impresión de cada parte (Ilustración 60). Después, mediante PrusaSlicer, hay que generar los archivos G-code necesarios para imprimir los modelos. Estos archivos se envían directamente a la impresora.

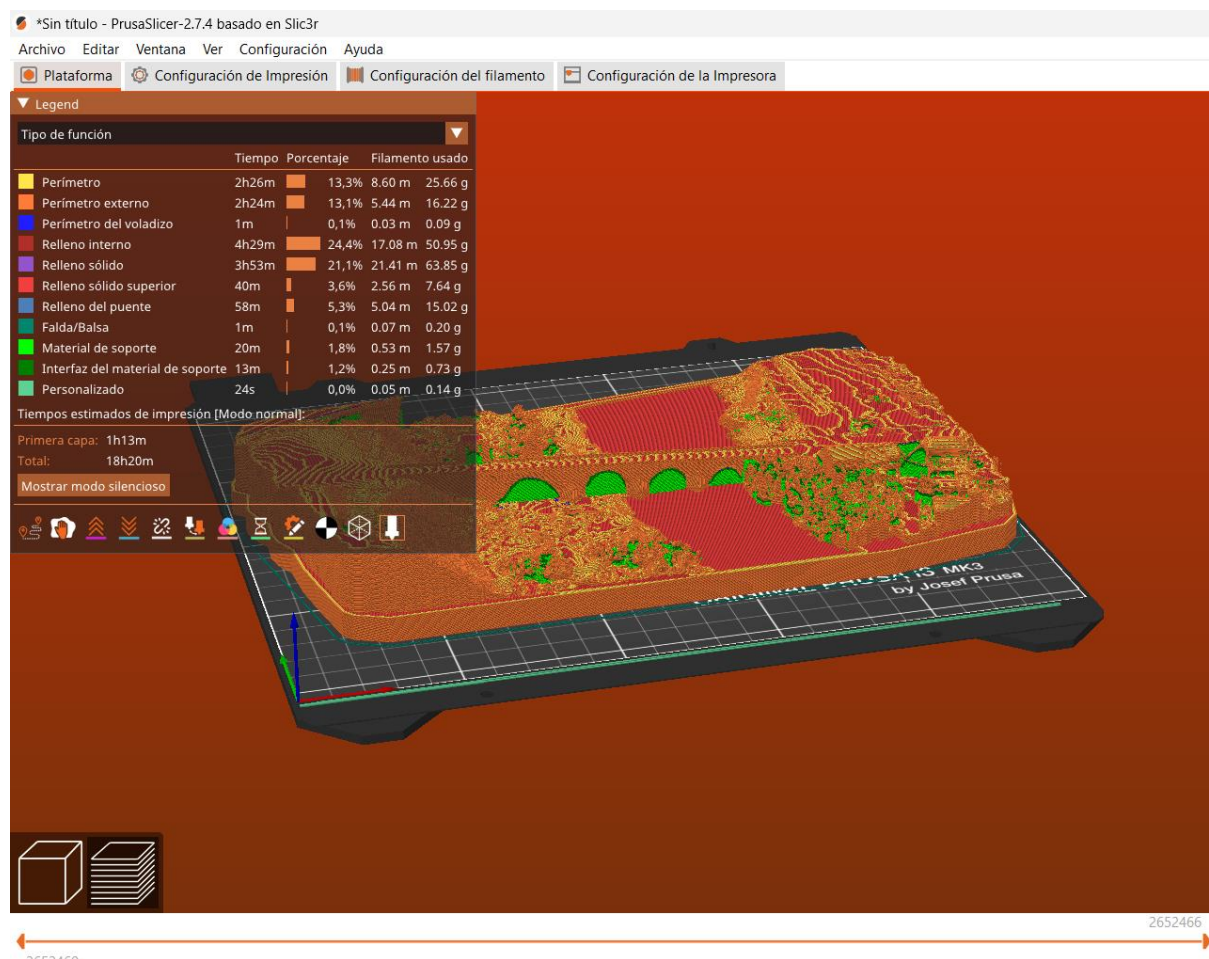


Ilustración 60: Laminación y tiempos de impresión en PrusaSlicer

5.2.2 Impresión 3D, pruebas y ajustes

Una vez realizada toda la preparación y configuración, se comenzó imprimiendo un primer modelo de prueba, para comprobar que se imprimiese correctamente y sin ningún tipo de fallo.

Aquí podemos ver la primera prueba de impresión realizada (Ilustración 61). Esta fue impresa en un tamaño inferior al deseado solo para realizar la prueba y no desperdiciar demasiado material:

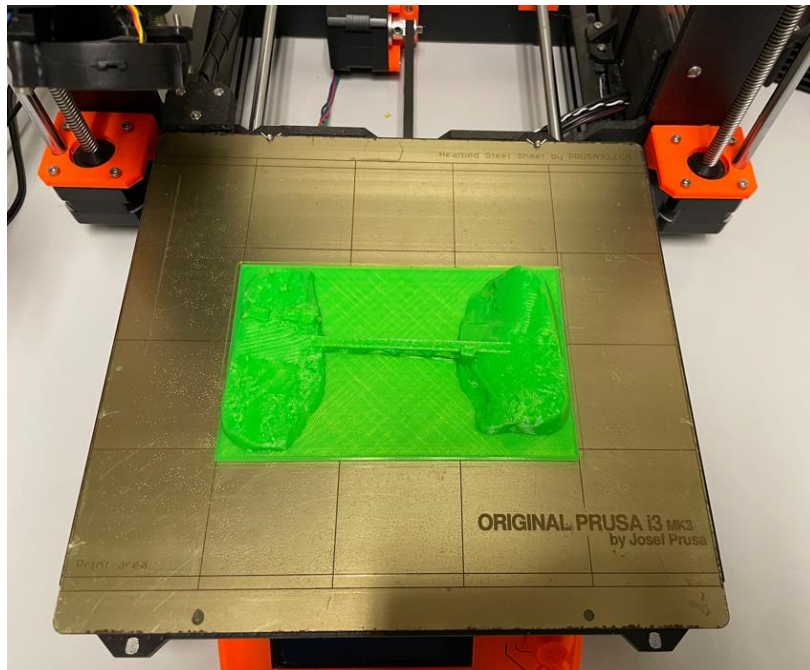


Ilustración 61: Primera prueba de impresión

Al imprimirla, podemos darnos cuenta de varios detalles a mejorar, como recortar la base siguiendo la forma del modelo para mejorar el aspecto visual o aumentar el número de capas de la parte superior para que no se perciba el relleno de la base.

Una vez impreso, hay que realizar algunos ajustes sobre este modelo, como extraer los soportes o pulir ciertas superficies que queden más afiladas.

5.3 Tercera iteración

Esta iteración está centrada en el desarrollo inicial del prototipo de aplicación de Realidad Aumentada. Los principales objetivos de esta iteración son obtener una buena base para la aplicación, realizar toda la parte de configuración previa del proyecto para garantizar su correcto funcionamiento, integrar Vuforia sin ningún tipo de problema (realizando las pruebas oportunas) y la implementación inicial e interfaz de usuario.

5.3.1 Configuración del proyecto en Unity

El primer paso necesario es la correcta instalación de la herramienta Unity Hub [47], que permite la instalación de múltiples versiones del Editor de Unity y manejar distintas extensiones y módulos (Ilustración 62).

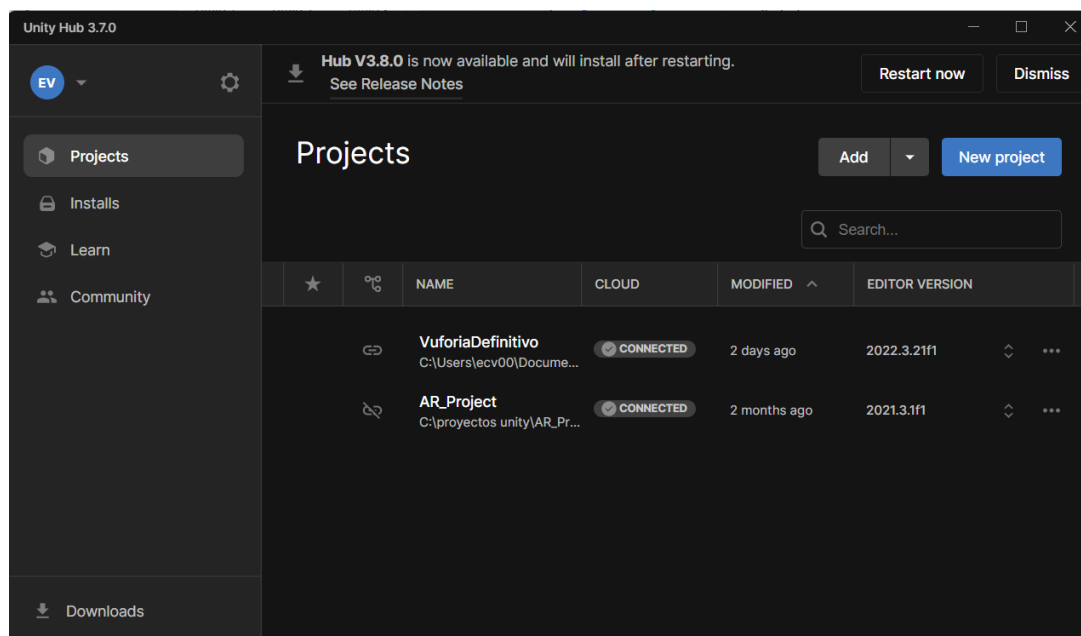


Ilustración 62: Captura de Unity Hub

Una vez instalado, dentro de Unity Hub se ha descargado la versión 2022.3.21f1 del Editor, junto con los módulos correspondientes de Windows y Android Build Support, de manera que la aplicación sea compatible con ambos sistemas.

En primer lugar, se ha creado un proyecto usando la plantilla 3D Core que incluye el propio Unity. Dentro del proyecto, para poder implementar correctamente las funcionalidades de Realidad Aumentada hay que configurar algunos aspectos previos:

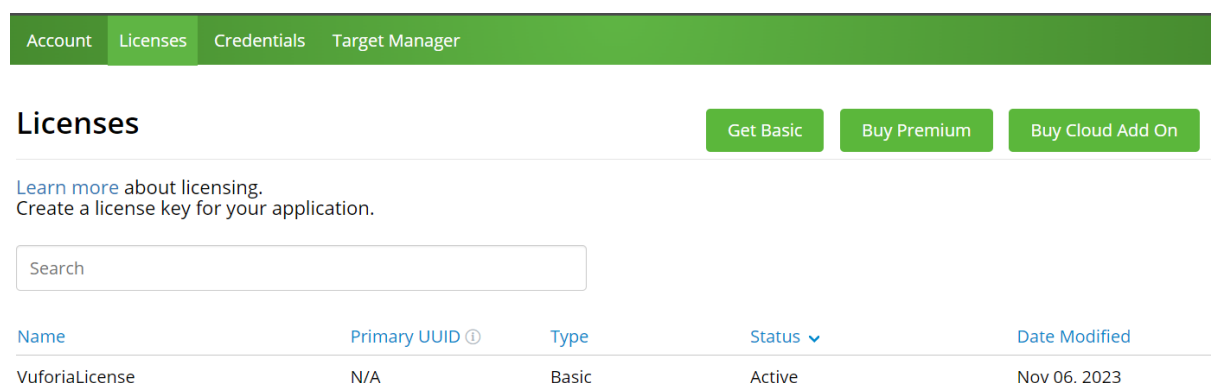
- **ARCore:** Añadir el Plug-in de ARCore en el proyecto ya que permite el desarrollo y uso de Realidad Aumentada en plataformas como Android.
- **Mínimo nivel de API:** Es necesario establecer el nivel mínimo de API en Android 8.0 'Oreo' ya que es el mínimo nivel que soporta el uso de Vuforia en Android.

- **Arquitectura:** Para poder desarrollar correctamente la aplicación tanto en Windows como en Android, es necesario establecer la arquitectura de 64 bits (ARM64).

Además, dado que los modelos 3D tienen el formato GLB, ha sido necesaria la instalación en el proyecto de Unity del paquete glTFast para poder trabajar con este tipo de formato [48].

5.3.2 Integración de Vuforia

Una vez creado y configurado el proyecto en Unity, se procede a la integración de Vuforia en el mismo [49]. Previamente es necesario crear una cuenta en Vuforia y obtener una licencia básica (Ilustración 63).



Name	Primary UUID	Type	Status	Date Modified
VuforiaLicense	N/A	Basic	Active	Nov 06, 2023

Ilustración 63: Licencias de Vuforia

Para integrar Vuforia en Unity, primero hay que descargar el SDK correspondiente [50] e importar el paquete que se ha descargado directamente en el proyecto.

Para comprobar que la instalación e integración se han realizado correctamente, es necesario realizar una serie de pruebas que incluyen la creación de una escena de prueba junto con la verificación de la detección de marcadores y la superposición de modelos 3D. En primer lugar, se ha añadido el componente AR Camera, que permite abrir la configuración de Vuforia y añadir nuestra clave de licencia (Ilustración 64). Si al ejecutar el proyecto se abre la cámara de nuestro dispositivo y no aparece ningún error, podemos afirmar que la integración de Vuforia se ha realizado con éxito.

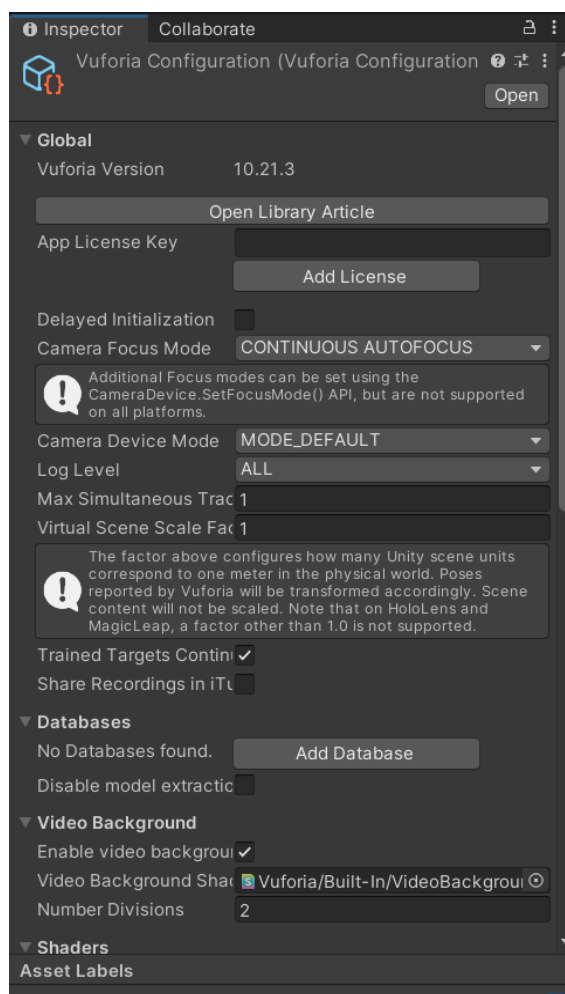


Ilustración 64: Configuración de Vuforia en Unity

5.3.3 Primeras pruebas de Vuforia

Cuando ya se ha asegurado la correcta implantación de Vuforia en el proyecto de Unity, lo siguiente es realizar unas primeras pruebas para probar y comprobar las funcionalidades básicas.

Primero se ha añadido el componente Image Target (imagen objetivo), que representa imágenes que Vuforia Engine puede reconocer y rastrear [51]. En el inspector de Unity de este componente añadimos una imagen propia que actuará como marcador de este componente, es decir, esta será la imagen que será detectada por Vuforia (Ilustración 65).

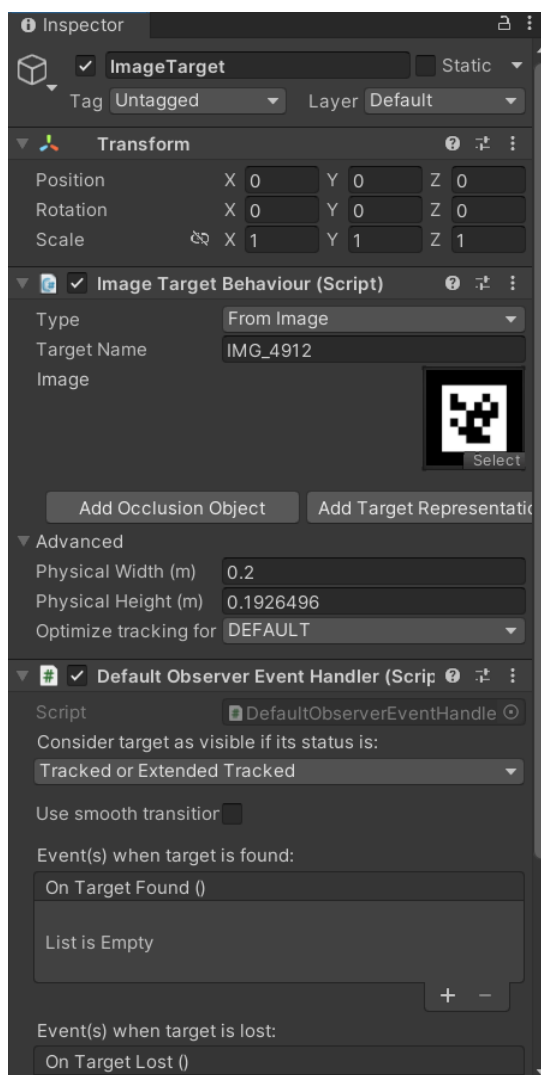


Ilustración 65: Captura del componente Image Target

Si añadimos a la escena un objeto 3D, un cubo, por ejemplo, lo escalamos y trasladamos de forma que esté colocado justo encima de la imagen del Image Target (Ilustración 66) y lo establecemos como hijo de este componente, podemos comprobar como al ejecutar la aplicación y enfocar a la imagen que actúa como marcador, aparece el cubo justo encima de la imagen. Si dejamos de enfocar la imagen, el cubo desaparece.

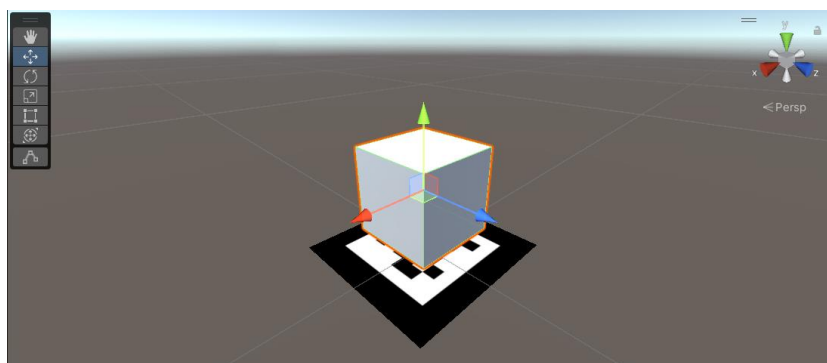


Ilustración 66: Ejemplo de un cubo 3D sobre un Image Target

Una vez que comprobamos que funciona correctamente al ejecutarlo en el ordenador, lo siguiente es asegurarnos que también funcione en el dispositivo Android. Para ello, debemos cambiar la configuración de compilación de Unity y cambiar la plataforma de Windows a Unity. Seguidamente, conectamos el dispositivo al ordenador, ejecutamos la aplicación, y comprobamos que efectivamente también funciona perfectamente en la tablet y se muestra el cubo sobre el marcador.

Con estas pruebas podemos concluir que la integración de Vuforia ha sido exitosa y podemos comenzar con el desarrollo e implementación de la aplicación de Realidad Aumentada propiamente dicha.

5.3.4 Diseño e implementación de la interfaz de usuario

Antes de comenzar con la implementación de la aplicación, es de gran utilidad diseñar y planear cómo será su interfaz de usuario, de manera que se diseñe una interfaz intuitiva y fácil de usar.

Este diseño y los bocetos realizados de la interfaz ya han sido expuestos anteriormente en la [Sección 4.2.5](#) de este documento. Sin embargo, los cambios realizados en la interfaz a lo largo del proyecto serán reflejados en las sucesivas iteraciones.

5.4 Cuarta iteración

El objetivo de esta iteración es conseguir integrar los modelos impresos en 3D en la aplicación de Realidad Aumentada, de manera que estas maquetas sean

detectadas por el dispositivo y se superponga información adicional sobre ellas de forma digital, además de la posibilidad de interactuar con estos modelos.

5.4.1 Generación de modelos objetivos

Vuforia ofrece una herramienta Model Target Generator [29] que permite la generación y creación de modelos objetivos (Model Targets [52]) a partir de objetos físicos 3D. Estos Model Targets permiten detectar y reconocer modelos 3D reales al igual que ocurre con las imágenes que actúan como marcadores.

Una vez instalado el programa Model Target Generator (MTG), se ha procedido con la generación de los modelos objetivos correspondientes a los modelos impresos.

Los objetos que van a ser transformados en modelos objetivos deben tener ciertas características para mejorar su detección [53]:

- **Movimiento de los objetos:** La detección de los objetos será mejor si el objeto es estático y no se mueve una vez ha sido reconocido.
- **Detalles geométricos suficientes:** Cuanto mayor sea la complejidad geométrica de los objetos, mejor será su reconocimiento.
- **Rígido y no flexible:** Hay que evitar objetos con partes flexibles que puedan moverse o partes articuladas que dificulten la detección de estos objetos.
- **Modelo CAD correspondiente:** Los modelos 3D digitales deben ser exactamente iguales a los objetos físicos, para que no haya disparidades entre ambos.
- **Superficie coloreada o estampada:** Los objetos que tienen colores o estampados tienen más probabilidad de ser reconocidos más fácilmente.
- **Texturas:** Los modelos 3D con texturas ayudan a mejorar la detección de los objetos, ya que ayudan a la extracción de la información necesaria.

En este caso, los modelos del Puente del Obispo diseñados para este proyecto son objetos estáticos, tienen suficientes detalles geométricos, son rígidos y no contienen partes articuladas, los modelos digitales correspondientes son idénticos a los físicos, los objetos físicos tienen un color y los modelos 3D cuentan con texturas, por lo que cumplen con todas estas características, lo que facilitará su reconocimiento y detección por parte de los dispositivos.

Para crear un modelo objetivo a partir de un modelo 3D, primero hay que importar el archivo original. Esta herramienta admite varios formatos como STL, OBJ, GLB, etc. En este caso importamos los modelos con el formato GLB ya que estos contienen texturas. Una vez importado un modelo, la herramienta nos pide configurar una serie de parámetros:

- **Vector hacia arriba del modelo:** En esta primera pestaña hay que seleccionar el vector hacia arriba que se corresponde al modelo. Como podemos ver en la Ilustración 67 se ha seleccionado el vector hacia arriba en la dirección Z.

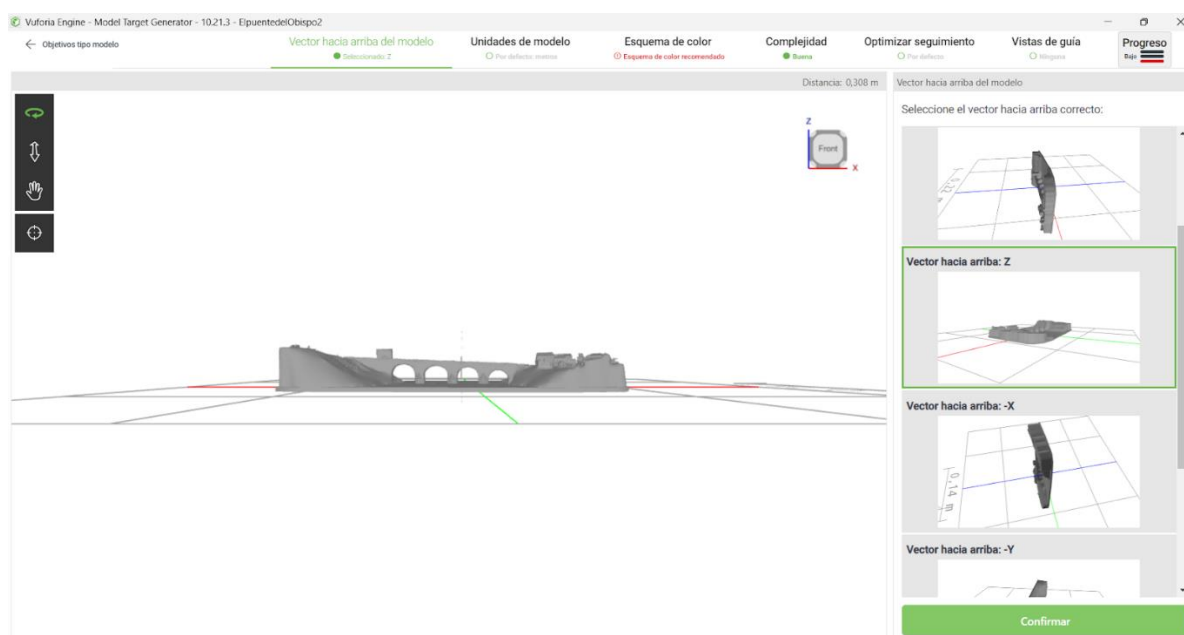


Ilustración 67: Pestaña Vector hacia arriba del modelo de MTG

- **Unidades de modelo:** En esta pestaña hay que elegir las unidades en las que se mide el modelo, así como su tamaño, que debe coincidir con

las medidas reales del objeto. En la Ilustración 68 podemos ver que en este caso se ha elegido el metro como unidad de medida.

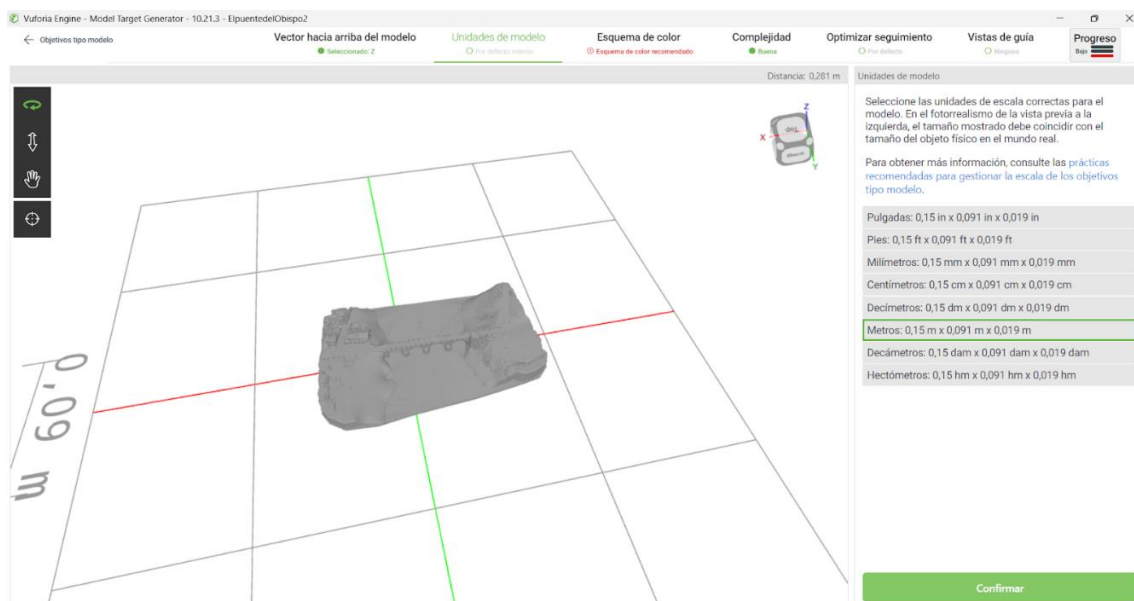


Ilustración 68: Pestaña Unidades del modelo de MTG

- **Esquema de color:** En esta pestaña hay que seleccionar si los colores del objeto físico y el modelo digital coinciden o no son realistas. En este caso se ha elegido el aspecto no realista ya que el objeto impreso tiene un color sólido, mientras que el modelo 3D tiene texturas de distintos colores (Ilustración 69).

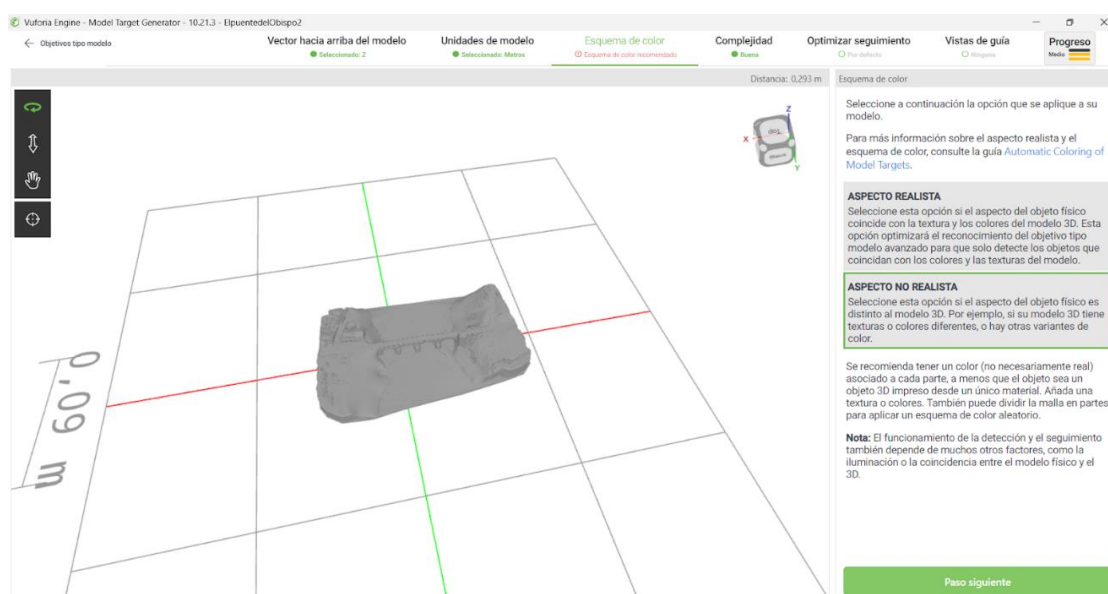


Ilustración 69: Pestaña Esquema de color de MTG

- **Complejidad:** Esta pestaña muestra la complejidad del modelo, incluyendo el número de piezas y de triángulos que contiene (Ilustración 70).

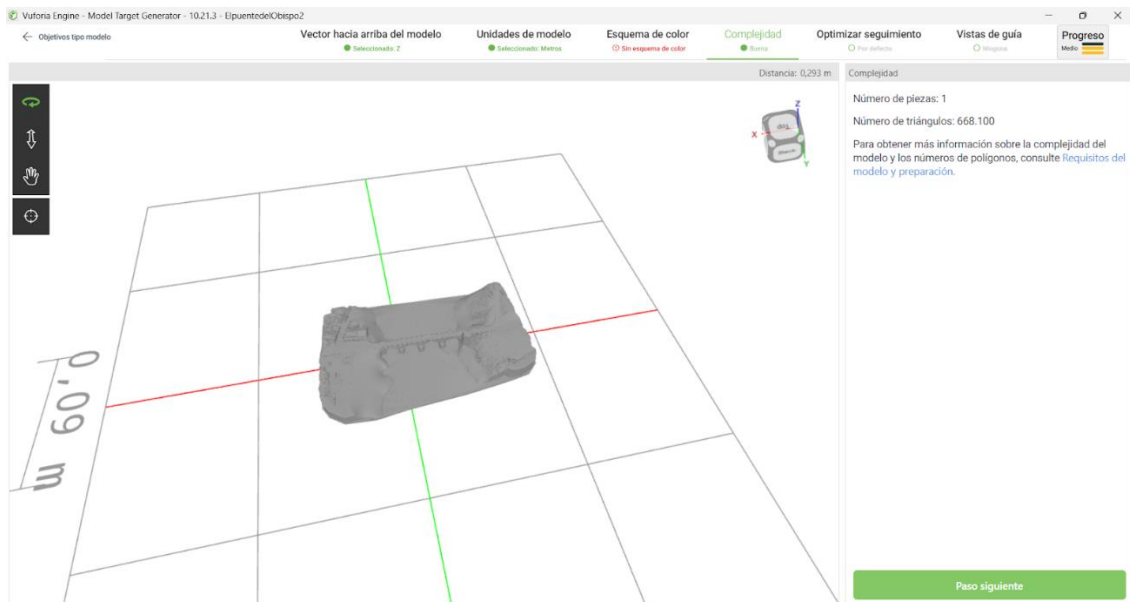


Ilustración 70: Pestaña Complejidad

- **Optimizar seguimiento:** En esta pestaña hay que determinar el modo de optimizar el seguimiento del modelo. Hay dos opciones:
 - **Por defecto:** Este seguimiento es óptimo para la mayoría de objetos.
 - **Objeto con funciones limitadas:** Para objetos con superficies suaves sin texturas, como coches.

En este caso se ha optado por la segunda opción ya que nuestros modelos impresos al tener un color sólido no tienen texturas (Ilustración 71).

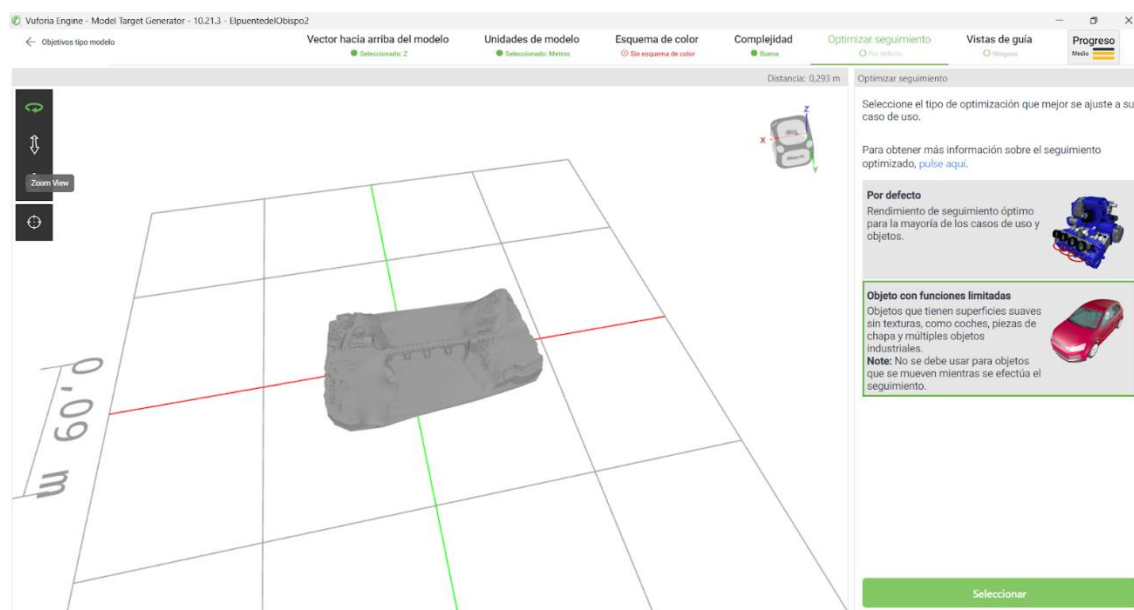


Ilustración 71: Pestaña Optimizar seguimiento en MTG

- Vistas de guía:** Las vistas de guía definen cómo serán reconocidos los modelos, existen las vistas de guía normales o avanzadas. En esta pestaña también se debe elegir el tipo de visión del dispositivo, entre Gafas digitales, Paisaje o Retrato. En este caso se ha elegido el tipo Paisaje, ya que el dispositivo estará orientado horizontalmente en la mayoría de casos (Ilustración 72).

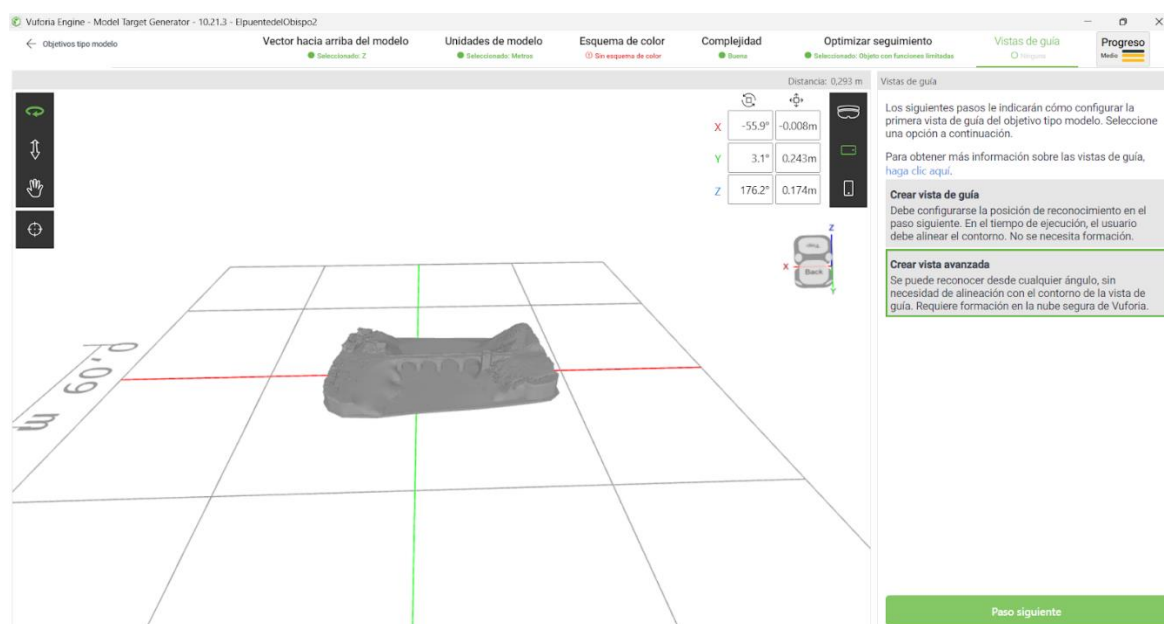


Ilustración 72: Pestaña Vistas de guía de MTG

Dentro de las vistas de guía avanzadas hay varias opciones y, en este caso, se ha optado por la Cúpula 360° (Ilustración 73) que permite reconocer objetos situados en superficies planas desde cualquier ángulo. Este tipo de vista de guía avanzada es perfecta para nuestros modelos ya que estarán colocados en superficies planas (Ilustración 74).

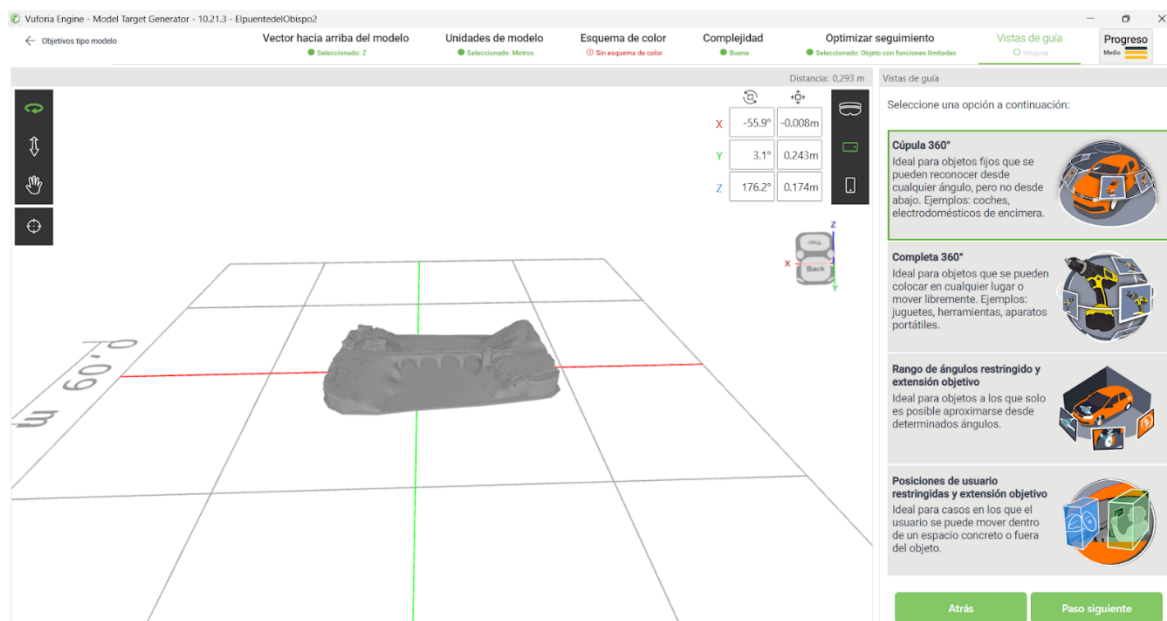


Ilustración 73: Pestaña Vistas de guía avanzadas de MTG

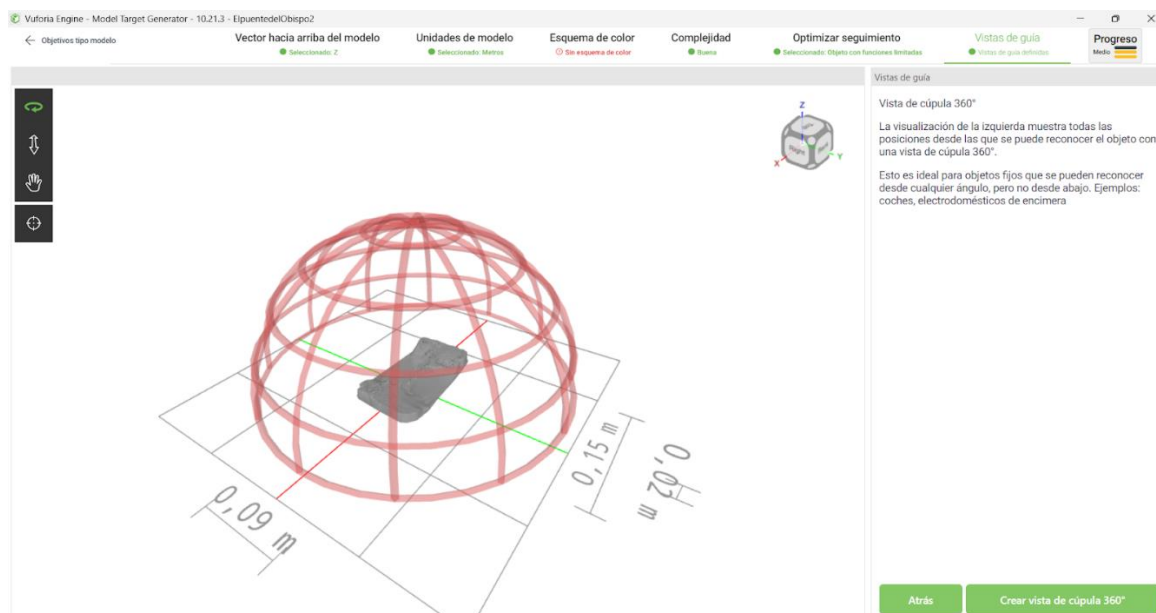


Ilustración 74: Vista de guía de cúpula 360°

Una vez generados los modelos objetivos hay que crear una sesión de entrenamiento con ellos para obtener las bases de datos asociadas a estos modelos, que serán las que se importen en el proyecto de Unity para poder usarlas en los componentes Model Targets de Vuforia.

A continuación, podemos ver los archivos que se obtienen al exportar la base de datos asociada a un modelo (Ilustración 75):

Nombre	Fecha de modificación	Tipo	Tamaño
 ElpuentedelObispo2.dat	16/03/2024 16:25	Media file	9.561 KB
 ElpuentedelObispo2.unitypackage	16/03/2024 16:25	Unity package file	9.365 KB
 ElpuentedelObispo2.xml	16/03/2024 16:25	Documento XML	1 KB
 ElpuentedelObispo2_GuideView_0000_1....	16/03/2024 16:25	Archivo PNG	180 KB
 ElpuentedelObispo2_GuideView_0000_2....	16/03/2024 16:25	Archivo PNG	50 KB

Ilustración 75: Archivos base de datos de MTG

5.4.2 Pruebas con los modelos objetivos

Tras generar los modelos objetivos y obtener su base de datos, es necesario probar su funcionamiento. Para ello existen dos aplicaciones, una para Android y otra para Apple, que ofrece Vuforia para probar los modelos objetivos generados por el programa Model Target Generator. En este caso se ha utilizado la aplicación compatible con Android, Model Targets Test App.

Esta aplicación permite comprobar la función de seguimiento y detección de los modelos objetivos, además de probar las vistas de guía establecidas. Para probar un modelo es necesario importar los archivos DAT y XML obtenidos al exportar la base de datos.

Una vez importados estos dos archivos podemos comprobar que, si el modelo objetivo ha sido generado correctamente, y apuntamos con la cámara al objeto impreso de este modelo, aparecerá un contorno blanco a su alrededor, lo que indica que es detectado con éxito. Podemos comprobar que la vista de guía Cúpula 360° se

ha generado correctamente ya que el objeto es reconocido desde todos sus ángulos (Ilustración 76).



Ilustración 76: Objeto detectado en Model Targets Test App

Esta aplicación cuenta con varias opciones y parámetros de configuración [54]. Como se puede observar en la Ilustración 77, hay dos botones a la izquierda, el primero permite cambiar el color del contorno que aparece alrededor del objeto, y el segundo permite cambiar la anchura o espesor de este contorno.

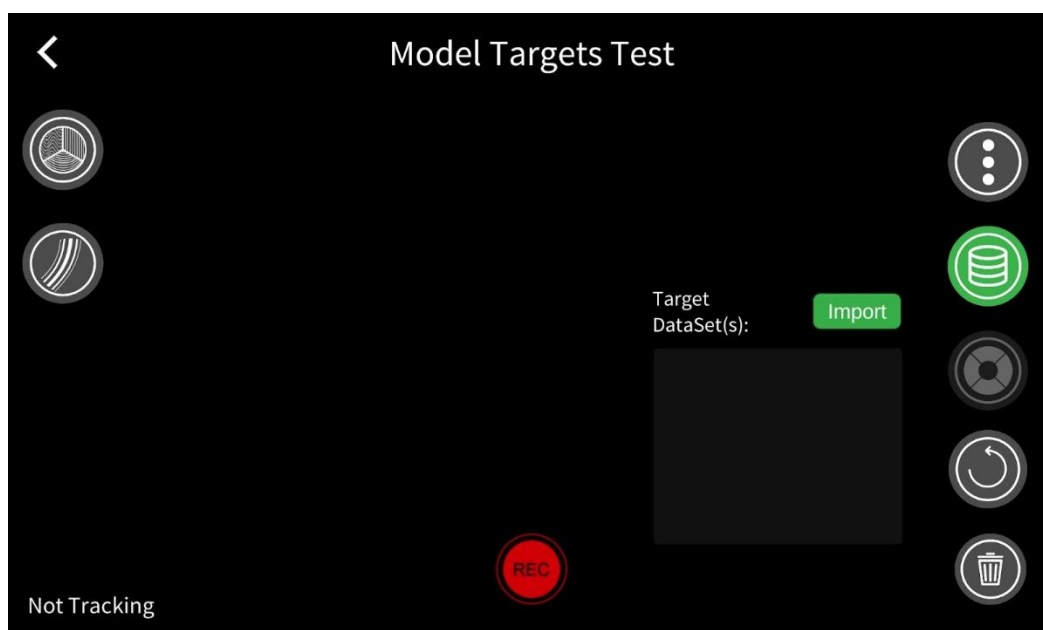


Ilustración 77: Captura de la aplicación Model Targets Test

A la derecha, la aplicación cuenta con cinco botones. El primer botón abre un menú de opciones, el segundo permite importar los archivos de la base de datos, el tercero sirve para recorrer las distintas vistas de guía si existe más de una, el cuarto reestablece el seguimiento y el ultimo lo elimina. El botón de abajo en el centro permite guardar sesiones.

En cuanto al primer botón (Ilustración 78), permite cambiar distintos parámetros de la configuración [54]:

- **Device Observer:** Esta opción permite habilitar o deshabilitar el seguimiento de dispositivos y el seguimiento extendido.
- **Autofocus:** Permite alternar el enfoque automático de la cámara.
- **Optimize tracking for:** Permite elegir entre varios modos de seguimiento.
- **Info panel:** Esta opción muestra información sobre el objeto, su modo de seguimiento, sus estados, etcétera.
- **Static Device Tracker:** Sirve en casos en los que la cámara esta estática.

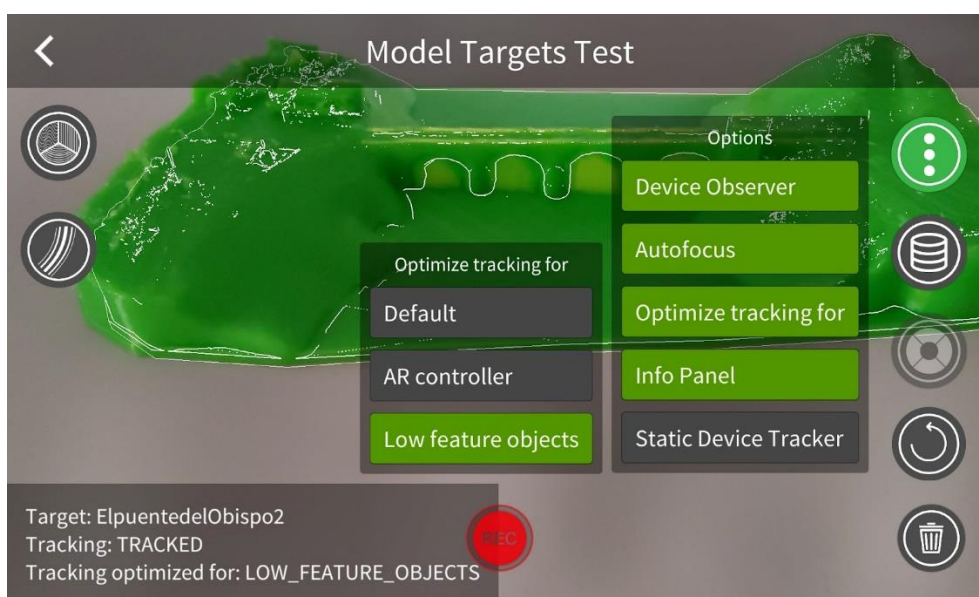


Ilustración 78: Menú de opciones de Model Targets Test App

5.4.3 Integración de los modelos en Unity

Después de comprobar que los modelos se han generado correctamente y son detectado con éxito, se procede a su integración en el proyecto de Unity usando Vuforia.

Antes de nada, es necesario importar el paquete de Unity correspondiente al modelo objetivo que se generó anteriormente. A continuación, se añade un componente Model Target de Vuforia a la escena. Se puede comprobar que se ha importado correctamente si aparece en la base de datos de la configuración de Vuforia (Ilustración 79).

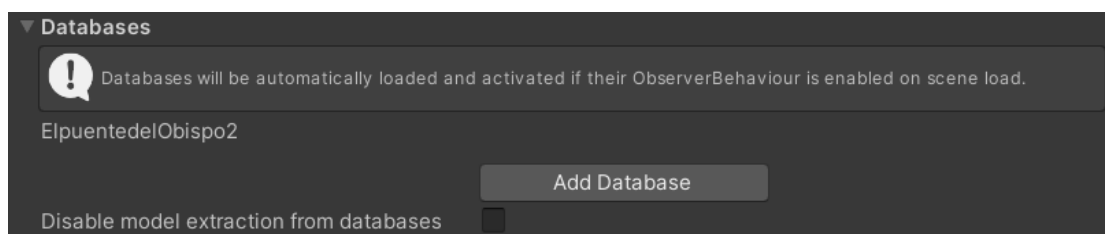


Ilustración 79: Captura de las bases de datos de Vuforia

Una vez importado, podemos seleccionar este modelo en el componente Model Target, en la pestaña del Inspector. Para comprobar si es reconocido en Unity, se ha realizado una primera prueba en la que se ha colocado un cubo 3D justo encima del modelo y se ha establecido como hijo de este componente. Por último, se ha comprobado que al ejecutar la aplicación tanto en el ordenador como en el dispositivo Android, al enfocar el modelo impreso en 3D con la cámara, este es detectado y se visualiza el cubo justo encima (Ilustración 80).



Ilustración 80: Prueba de visualización de un modelo objetivo

Tras comprobar que el reconocimiento y visualización de los modelos objetivos funciona adecuadamente, se ha sustituido el cubo por el modelo 3D original con las texturas correspondientes. Se ha escalado, trasladado y rotado para superponerlo justo encima del Model Target asociado, de manera que al enfocarlos aparezca solapado, simulando el modelo original con texturas (Ilustración 81).

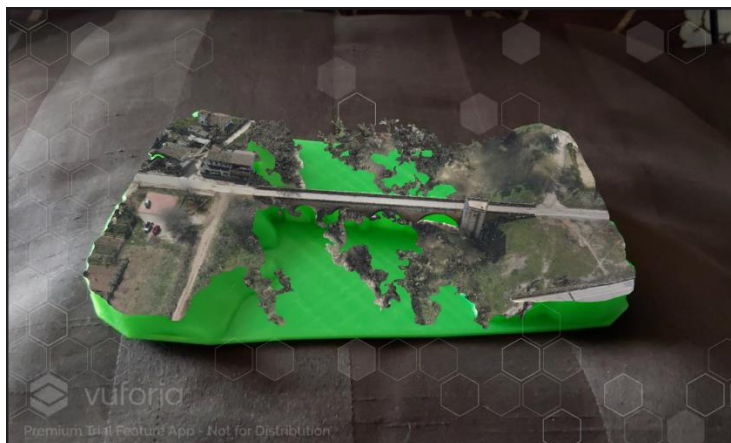


Ilustración 81: Model Target con modelo 3D original

Como se puede apreciar en la imagen, el modelo 3D original no se ajusta totalmente al modelo impreso, por lo que será modificado adecuadamente en iteraciones posteriores para que se adapte y coincida con la forma del modelo impreso.

5.5 Quinta iteración

En esta iteración se han implementado todas las funcionalidades relacionadas con la interacción de los marcadores, de los modelos y de la interacción entre ambos, además del diseño de las imágenes de los marcadores y la superposición de información adicional en pantalla.

A continuación, se describe más detalladamente cada funcionalidad e interacción desarrollada en la aplicación.

5.5.1 Diseño imágenes de los marcadores

Lo primero que se ha realizado en esta quinta iteración ha sido el diseño de las imágenes que actúan como marcadores en la aplicación, es decir, las imágenes asociadas a los componentes Image Targets.

A la hora de diseñar estas imágenes hay que tener en cuenta ciertos aspectos que hacen que sean más fácil de reconocer [55]. Los marcadores deben ser ricos en detalles, lo que facilita su reconocimiento y los diferencia del resto. También deben tener un buen contraste y no tener patrones repetitivos, por ejemplo, los códigos QR cumplen estas características y son rápidos y fáciles de detectar. En cuanto al formato deben ser imágenes JPG o PNG.

En este caso se han diseñado tres marcadores distintos, uno para la interacción entre marcadores y modelos y otros dos para representar tanto la torre como el puente sólo del modelo.

El primer marcador es un código QR, ya que como se ha comentado anteriormente, cumple con los requisitos básicos para tener una fácil detección (Ilustración 82).

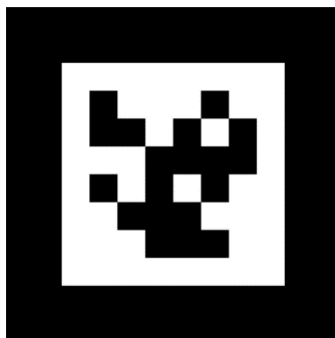


Ilustración 82: Código QR que actúa de marcador

Para el marcador que permite interactuar con la torre o ermita del Puente del Obispo, se ha optado por crear una imagen que represente una torre y a su vez sea reconocible fácilmente, por lo que se le ha añadido cuatro pequeños códigos QR en las esquinas (Ilustración 83).

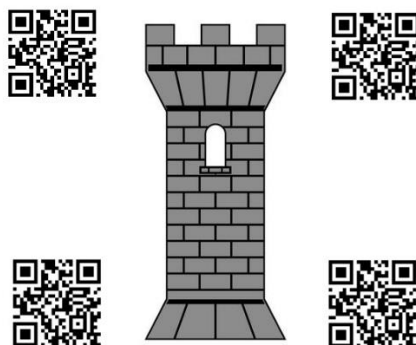


Ilustración 83: Imagen del marcador de la torre

Para el marcador del puente se ha seguido el mismo criterio, se ha diseñado una imagen que represente a un puente y adicionalmente se ha colocado un código QR en el centro (Ilustración 84).

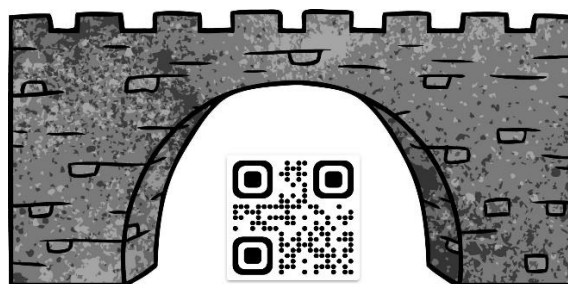


Ilustración 84: Imagen del marcador del puente

Una vez diseñadas las imágenes, en Vuforia, se han creado imágenes objetivos a partir de estas, indicando el tamaño real que tienen estos marcadores, en este caso tienen un tamaño de 5 cm. Estas imágenes objetivo pertenecen a una base de datos de Vuforia que ha sido descargada e importada en el proyecto de Unity.

En Vuforia, al crear estos marcadores indican su valoración en forma de 5 estrellas que indican como de buenos son estos marcadores. Cuantas más estrellas, más reconocibles y más fáciles de detectar serán (Ilustración 85).

Target Manager > database

database [Edit Name](#)**Type:** Device

Targets (6)

Add Target

Download Database (All)

☐	Image	Target Name	Type	Rating ①	Status ▾	Date Modified
☐		puenteQR	Image	★★★★★	Active	Jun 01, 2024
☐		torreQR	Image	★★★★★	Active	Jun 01, 2024
☐		qr	Image	★★★★☆	Active	Apr 15, 2024

Ilustración 85: Imágenes objetivo en Vuforia

5.5.2 Interacción usando marcadores

Tras diseñar y generar las imágenes que actúan como marcadores, podemos pasar a su integración en el proyecto de Unity. Para ello es necesario crear un componente Image Target por cada marcador que se quiera utilizar.

Dentro de la pestaña Inspector de cada Image Target se ha seleccionado la imagen correspondiente de la base de datos que contiene los marcadores diseñados anteriormente.

En la jerarquía de la escena, se han añadido como hijos de cada componente Image Target los modelos 3D de la torre (Ilustración 86) y el puente (Ilustración 87) respectivamente y se han ajustado para situarlos justo encima de las imágenes de los marcadores, de manera que al enfocar estos marcadores aparezcan los modelos sobre ellos.

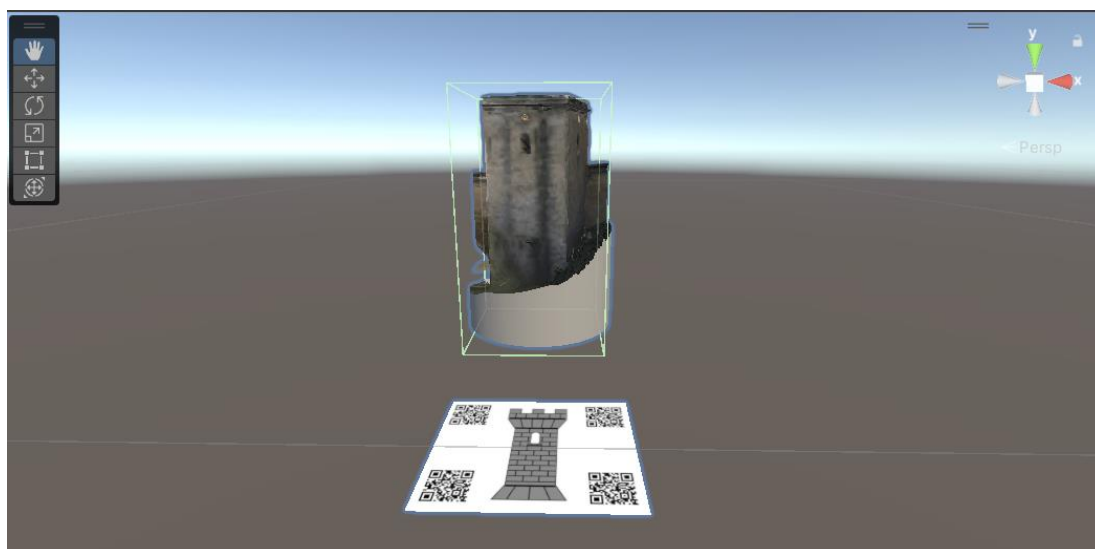


Ilustración 86: Marcador y modelo de la torre

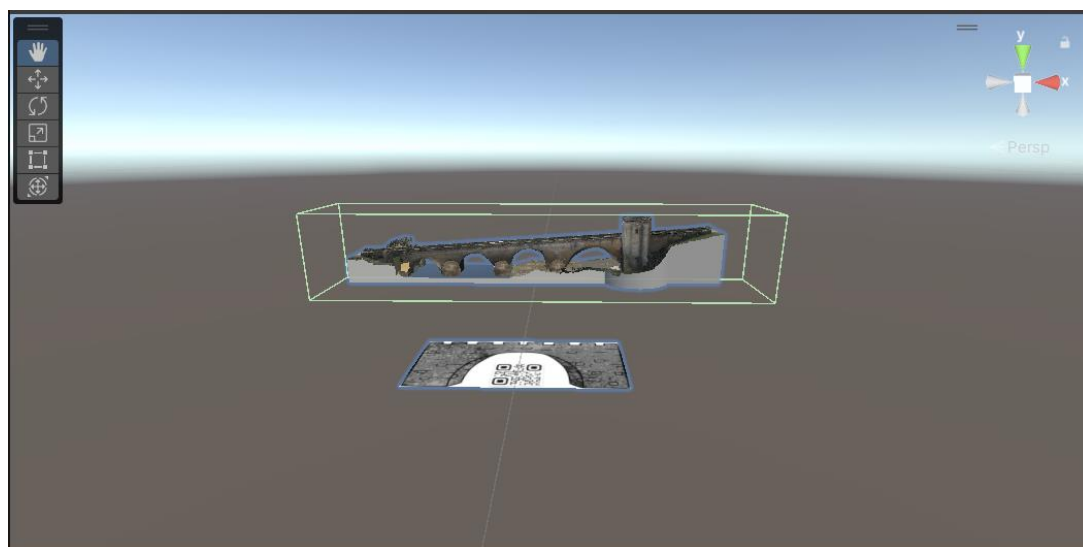


Ilustración 87: Marcador y modelo del puente

Además de poder visualizar estos dos modelos en 3D al enfocar las imágenes, se ha implementado la funcionalidad de poder rotarlos y de poder hacerles zoom deslizando los dedos por la pantalla y pellizcándola respectivamente. Para rotar los modelos en los ejes X e Y basta con deslizar un dedo por la pantalla, mientras que para rotarlos en el eje Z serán necesarios dos dedos.

Para poder rotar los modelos se ha colocado un objeto vacío en el centro del modelo y se ha desarrollado el script “*objectRotation.cs*”, en el que se instancia una variable para este objeto que hace de rotación, otra variable para establecer

la velocidad de rotación, otra para definir la distancia inicial entre los dedos y otra para almacenar el escalado inicial del objeto.

Este script sólo cuenta con un método en el que se manejan los gestos de rotación y zoom, el método *Update()* de Unity, que se llama una vez por frame. Para realizar la rotación, primero se comprueba si se ha tocado la pantalla, si es así, se obtiene el primer toque y se comprueba si el dedo se ha movido por la pantalla. Si el dedo se ha deslizado, se calcula la rotación en los ejes X e Y. Si se ha tocado la pantalla con más de un dedo se realiza la rotación en el eje Z y si el número de toques en pantalla han sido dos, se realiza el zoom de los objetos. Para realizar el zoom se comprueba si se está iniciando el gesto de pellizco, si es así, se guarda la distancia inicial entre ambos dedos y la escala inicial del modelo. Por el contrario, si los dedos se están moviendo, se calcula un factor de escala que depende de la distancia de los dedos. Finalmente, se aplica este factor a la escala inicial para aumentar o reducir su tamaño adecuadamente.

Sin embargo, cuando se deja de enfocar a estos marcadores y se vuelven a enfocar, los modelos aparecen con el último escalado, en vez de aparecer con su tamaño original. Para solucionar esto, se ha optado por almacenar los valores iniciales de escalado de cada modelo y reestablecerlos cuando se dejan de enfocar las imágenes.

5.5.3 Interacción entre varios marcadores

Para poder detectar varios marcadores a la vez, en la configuración de Vuforia hay que especificar el número máximo de imágenes rastreadas simultáneamente. En el caso de esta aplicación como máximo van a interactuar 2 marcadores de imagen a la vez.

El objetivo es conseguir que cada marcador visualice su propio modelo 3D cuando se enfocan por separado, y cuando se enfoque a ambos a la vez, se visualice otro modelo, en este caso el modelo completo del Puente del Obispo (Ilustración 88).



Ilustración 88: Ejemplo interacción con dos marcadores

Para desarrollar esta funcionalidad se ha implementado el script “*targetInteraction.cs*”, que incluye una variable para cada imagen objetivo, otra para almacenar el modelo completo y otras dos para guardar los modelos originales de cada marcador. En el método *Start()*, llamado automáticamente por Unity cuando se inicia la aplicación, primero se verifica si los objetivos AR han sido asignados y se suscribe al evento *OnTargetStatusChanged* [56] de cada objetivo. Este evento se dispara cada vez que cambia el estado de un objetivo, por ejemplo, cuando se empieza o se deja de rastrear. Además, se almacena una referencia a los modelos originales que se muestran en cada objetivo.

Otro de los métodos de este script es *OnTargetStatusChangedForSwap()*, que se llama cada vez que el estado de un objetivo cambia. Si ambos marcadores están siendo rastreados, hay que cambiar los modelos que se muestran por el modelo completo. En cambio, si uno o ambos de los marcadores dejan de ser enfocados, se vuelve a mostrar los modelos originales y se elimina el modelo actual.

Por otro lado, el método *SwapModel()* es el encargado de desactivar los modelos originales e instancia el nuevo modelo, situándolo en el punto medio de ambos marcadores. Por último, el método *DisableChildObjects()* desactiva los objetos hijos del objeto que recibe como parámetro.

5.5.4 Interacción con modelos 3D

Una de las funcionalidades más importantes que se ha implementado en esta aplicación de Realidad Aumentada es la posibilidad de interactuar con modelos 3D impresos físicamente, mediante la superposición de información adicional sobre ellos.

Para realizar esto, se han generado los modelos objetivos correspondientes a cada modelo impreso y se han integrado en el proyecto como se ha explicado en apartados anteriores. Una vez hecho esto, para interactuar con estos modelos, se ha desarrollado la opción de poder tocar en pantalla en algunas zonas específicas del modelo para obtener información relevante sobre ellas.

En primer lugar, se han añadido cajas de colisión, llamadas Box Collider [57] en Unity, ajustadas a cada una de las partes del modelo de la que queremos obtener información al pulsar sobre ellas. En este caso, se quiere obtener información relevante sobre la zona de la torre (información sobre la ermita del puente), la del puente (información sobre el Puente del Obispo) y la de la casa de al lado del puente (información sobre el entorno del puente). Un Box Collider es una primitiva básica de colisión en forma de cubo. A continuación, podemos ver las tres cajas de colisión asociadas al modelo (Ilustración 89):

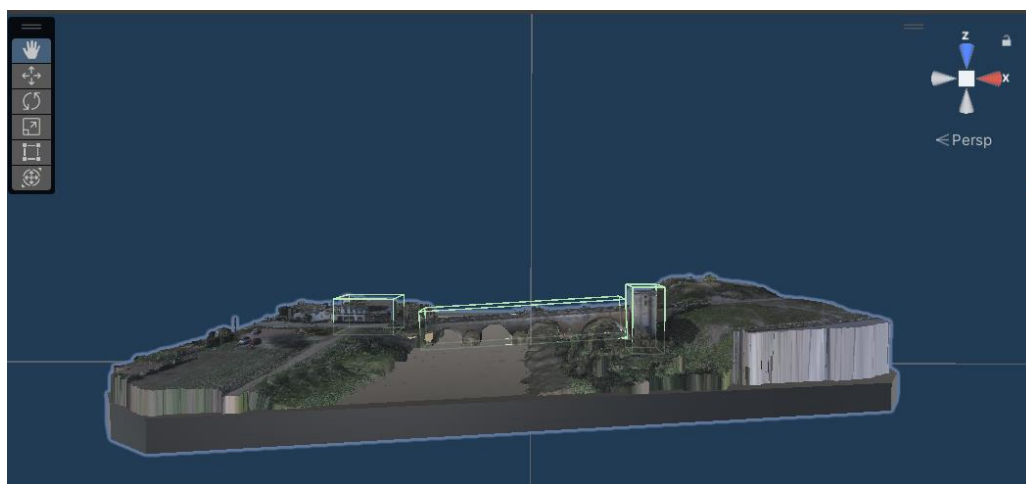


Ilustración 89: Box Colliders asociados al modelo del Puente del Obispo

A cada uno de estos Box Colliders se le ha asignado una etiqueta específica para identificar y diferenciar cada uno de ellos.

Para detectar que el usuario está tocando una de las zonas marcadas del modelo, se ha utilizado la técnica del Raycast [58], que consiste en lanzar un rayo desde la posición donde se ha pulsado en pantalla y comprobar con qué objeto u objetos choca dicho rayo.

En primer lugar, se han diseñado los paneles de la interfaz con la información relevante de cada una de las zonas elegidas del modelo del Puente del Obispo. Se ha diseñado un panel con información sobre el Puente del Obispo (Ilustración 90), otro panel con información sobre la torre o ermita del puente (Ilustración 91), y otro panel con información sobre el entorno del puente (Ilustración 92). Estos tres paneles son deslizantes, es decir, el usuario debe deslizar hacia abajo para seguir leyendo.

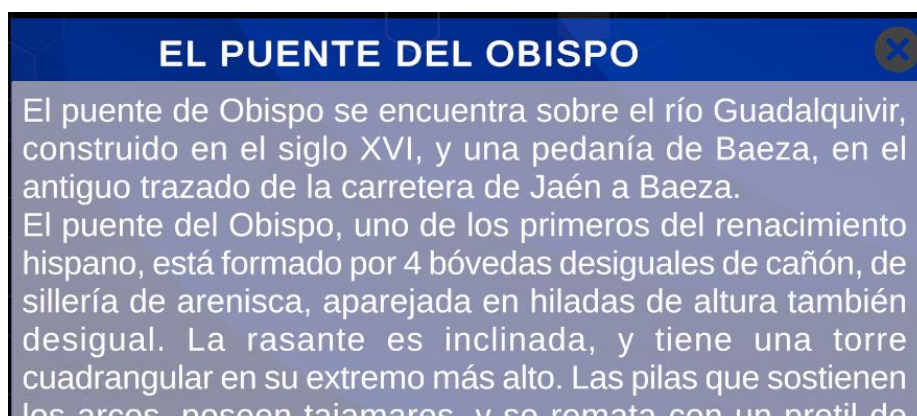


Ilustración 90: Panel informativo del Puente del Obispo

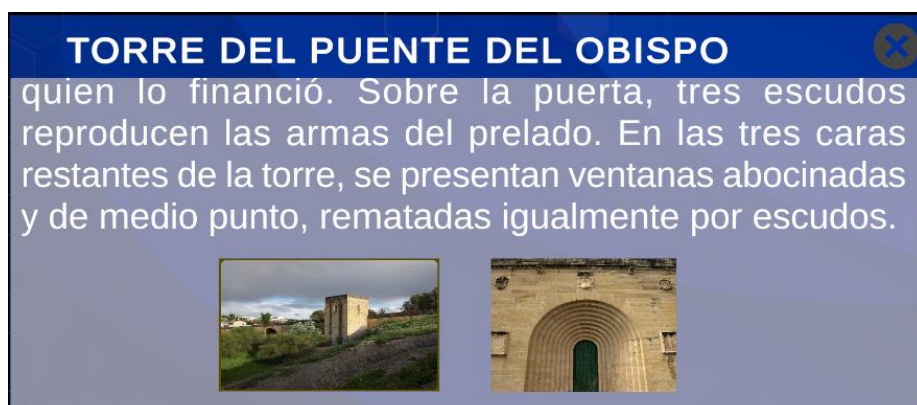


Ilustración 91: Panel informativo sobre la torre del Puente del Obispo

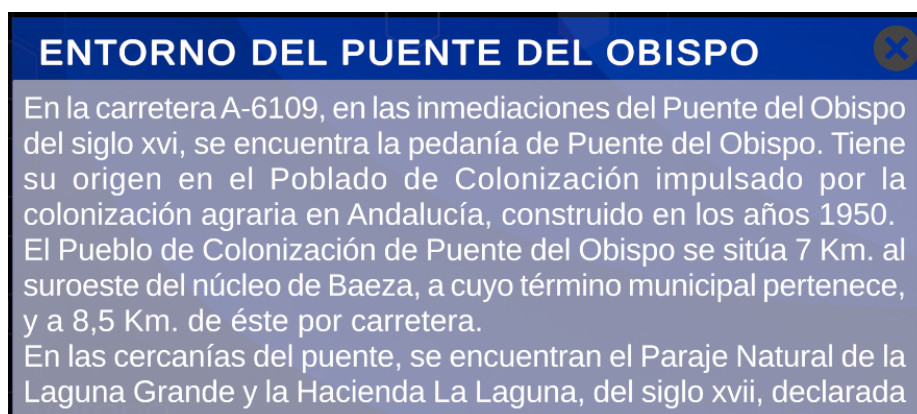


Ilustración 92: Panel informativo del entorno del Puente del Obispo

Tras diseñar y crear los paneles informativos, se ha implementado el script “*objectInteraction.cs*” en el que se instancian como variable los tres paneles asociados a cada zona con la información relevante de cada una de ellas. Este script cuenta con un solo método, el método Update(), en el que primero se comprueba si el usuario ha tocado la pantalla, si es así, se lanza un rayo desde el punto tocado y se comprueba si el rayo intersecta con algún objeto de la escena. Se comprueba si el rayo colisiona con alguna de las tres cajas de colisión según su etiqueta identificativa y en caso de chocar con alguna, activa el panel informativo correspondiente.

Hasta ahora se ha trabajado con un único Model Target o modelo objetivo. Sin embargo, Vuforia solo permite tener activo un Model Target a la vez, por lo que a la hora de añadir nuevos modelos objetivos a la escena tenemos que asegurarnos que solo esté activo uno de ellos. En esta iteración se han añadido nuevos modelos objetivos a la escena, como el modelo 3D de un barco (Ilustración 93), siguiendo el mismo procedimiento que se realizó para añadir el Model Target asociado al modelo del Puente del Obispo.

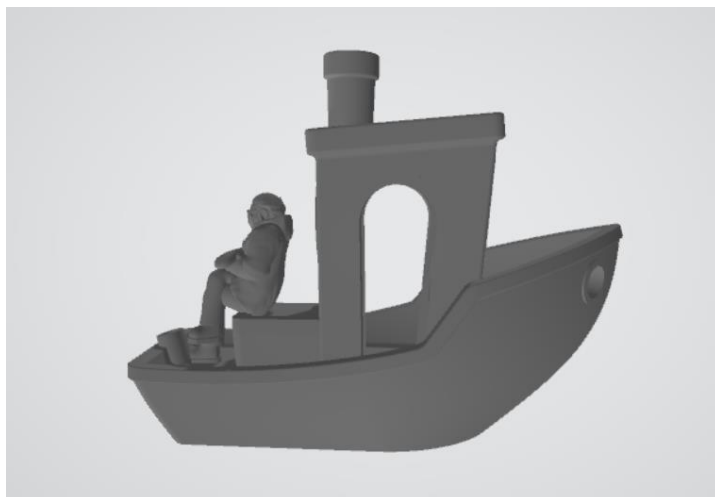


Ilustración 93: Modelo 3D de un barco

Al enfocar al objeto impreso del barco se superpone su modelo 3D asociado con un color específico, en este caso el azul, que se ha añadido al modelo mediante el uso de un material [59] de Unity (Ilustración 94).



Ilustración 94: Modelo del barco azul

Para gestionar los modelos objetivos de esta aplicación se ha desarrollado el script “*ModelTargetManager.cs*” que cuenta con una variable por cada Model Target de la escena y un método para cada uno de ellos. Cada función asociada a cada modelo objetivo, se encarga de desactivar el resto de modelos y activar el modelo correspondiente, de esta manera nos aseguramos tener un único modelo objetivo activo a la vez. Estas funciones serán llamadas desde la interfaz de usuario, en la que se ha implementado un botón para cada uno de los modelos objetivos de la escena

(Ilustración 95). Por ejemplo, al pulsar el botón asociado al modelo del barco, se activará su Model Target asociado y se desactivará el resto.



Ilustración 95: Captura botones de los modelos

5.5.5 Interacción entre marcadores y modelos 3D

Por último, en esta quinta iteración, se ha desarrollado distintas interacciones entre los modelos y los marcadores, de manera que se pueda interactuar con ambos a la vez. De esta manera se pueden conseguir nuevos efectos visuales en los modelos 3D gracias a la interacción con los marcadores.

En este prototipo se ha implementado la opción de simular el agua del río que pasa por el Puente del Obispo, añadiendo unos planos que simulen el efecto del agua sobre el modelo 3D. Estos planos se encuentran desactivados inicialmente y cuando se enfoque al marcador adecuado se activarán y se mostrará este efecto de río sobre el modelo (Ilustración 96).



Ilustración 96: Efecto río sobre el puente

Para realizar esto, se ha desarrollado el script “*modelTargetInteraction1.cs*” en el que se declaran como variables el componente Image Target que actúa como marcador (código QR), el componente Model Target correspondiente al modelo 3D del puente y los objetos de los planos que simulan el agua del río. Este script cuenta con el método *Start()*, llamado automáticamente al inicio de la ejecución de la aplicación, en el que se comprueban si las variables han sido asignadas y se suscribe al evento *OnTargetStatusChanged* tanto del Model Target como del Image Target. El método *OnTargetStatusChangedForSwap()* se llama cuando el estado del Image Target o del Model Target cambia y comprueba si ambos objetivos están siendo rastreados simultáneamente. Si es así, activa los planos correspondientes al agua del río. En cambio, si alguno o ninguno de los dos objetivos está siendo enfocado, se desactivan estos planos.

Por otro lado, también se ha implementado una interacción entre el modelo 3D del barco y el marcador correspondiente a la imagen del código QR. Como ya se ha mencionado anteriormente, cuando se enfoca el modelo impreso del barco, se superpone dicho modelo digitalmente con un material de color azul. Lo que se quiere conseguir con esta interacción es que cuando se esté enfocando al modelo impreso del barco y a la misma vez se enfoque al marcador, cambie el color del barco, por ejemplo, a rosa, es decir, que cambie el material asociado al modelo (Ilustración 97).



Ilustración 97: Modelo del barco rosa

Para implementar esta funcionalidad se ha desarrollado el script “*modelTargetInteraction2.cs*”, similar al anterior. En este script se definen las variables asociadas al Model Target del barco y al Image Target del marcador del código QR. También se declara una variable del modelo 3D del barco, al que se le cambia el color, y dos variables para cada uno de los materiales, uno con el color azul para cuando no se esté enfocando al marcador, y otro con el color rosa para cuando se enfoque al marcador. En el método *Start()* se comprueba que las variables no contengan valores nulos y se suscribe al evento *OnTargetStatusChanged*, de manera que cuando el estado de seguimiento de un objetivo cambia, se llama al método *OnTargetStatusChangedForSwap()*, en el que se comprueba si ambos objetivos están siendo rastreados y asocia al modelo del barco el material correspondiente.

5.6 Sexta iteración

Esta última iteración se ha centrado en la optimización del rendimiento de la aplicación y la preparación para su lanzamiento, incluyendo la realización de pruebas de calidad, diseño del logo de la aplicación, e implementación de mejoras y refinamiento.

5.6.1 Pruebas de usabilidad y funcionalidad

Los objetivos de estas pruebas de usabilidad y funcionalidad consisten en evaluar la facilidad de uso, verificando que los usuarios puedan navegar y utilizar la aplicación sin dificultades; identificar problemas de usabilidad, de manera que se

detecte cualquier aspecto que pueda resultar confuso o complicado para los usuarios; y valide la funcionalidad, asegurándose que todas las funciones de la aplicación funcionan sin ningún tipo de problema.

Para realizar estas pruebas se ha elegido a un número representativo de personas que actuarán como usuarios de la aplicación y evaluarán su funcionalidad realizando una serie de tareas. Las tareas a realizar son las siguientes:

1. Interactuar con un marcador.
2. Interactuar con varios marcadores.
3. Interactuar con los modelos y marcadores.
4. Cambiar de un modelo a otro.
5. Obtener información sobre el modelo del Puente del Obispo

Los datos y comentarios que han ido realizando los usuarios a la hora de realizar estas tareas, se han recogido en forma de encuesta, en la que también se han añadido preguntas para evaluar distintos aspectos de la aplicación. A continuación, se muestran las distintas encuestas realizadas a los usuarios:

Preguntas	Respuestas Usuario 1
Indique su edad	45 o más
Indique su nivel de conocimientos informáticos	Básico
En una escala del 1 al 5, ¿cómo calificarías la facilidad de uso de la interfaz?	4
¿Encontraste algún problema al usar la aplicación? Si es así, descríbelo	Dificultad para mover los objetos

¿Qué te gustó más de la aplicación?	Poder ver las imágenes reales a partir de maquetas en 3D
¿Qué mejorarías en la aplicación?	El movimiento de los objetos que aparecen con los marcadores
En una escala del 1 al 5, ¿cómo calificarías la satisfacción general con la aplicación?	5

Tabla 8: Respuestas de la encuesta del primer usuario

Preguntas	Respuestas Usuario 2
Indique su edad	18-24
Indique su nivel de conocimientos informáticos	Avanzado
En una escala del 1 al 5, ¿cómo calificarías la facilidad de uso de la interfaz?	5
¿Encontraste algún problema al usar la aplicación? Si es así, descríbelo	Ningún problema
¿Qué te gustó más de la aplicación?	Permite interactuar de múltiples formas y obtener información de los objetos
¿Qué mejorarías en la aplicación?	Igual indicaría en el modelo que partes se pueden pulsar
En una escala del 1 al 5, ¿cómo calificarías la satisfacción general con la aplicación?	5

Tabla 9: Respuestas de la encuesta del segundo usuario

Preguntas	Respuestas Usuario 3
Indique su edad	45 o más
Indique su nivel de conocimientos informáticos	Básico
En una escala del 1 al 5, ¿cómo calificarías la facilidad de uso de la interfaz?	3
¿Encontraste algún problema al usar la aplicación? Si es así, descríbelo	Me costó mover los objetos, ya que se movían muy rápido
¿Qué te gustó más de la aplicación?	Me gustó mucho todas las posibilidades que ofrece y poder verlo en Realidad Aumentada
¿Qué mejorarías en la aplicación?	Poner en las instrucciones cómo usar los dedos para mover los objetos
En una escala del 1 al 5, ¿cómo calificarías la satisfacción general con la aplicación?	4

Tabla 10: Respuestas de la encuesta del tercer usuario

Preguntas	Respuestas Usuario 4
Indique su edad	18-24
Indique su nivel de conocimientos informáticos	Intermedio
En una escala del 1 al 5, ¿cómo calificarías la facilidad de uso de la interfaz?	4

¿Encontraste algún problema al usar la aplicación? Si es así, descríbelo	No queda muy claro en qué partes se puede pulsar para obtener información
¿Qué te gustó más de la aplicación?	Poder ver los objetos en 3D a la vez que se ve el espacio en el que me encuentro
¿Qué mejorarías en la aplicación?	Indicar las partes que se pueden pulsar
En una escala del 1 al 5, ¿cómo calificarías la satisfacción general con la aplicación?	5

Tabla 11: Respuestas de la encuesta del cuarto usuario

A continuación, podemos ver distintas gráficas que resumen algunas de las preguntas de esta encuesta. En primer lugar, se puede observar el porcentaje de la edad de las personas que han realizado la encuesta (Ilustración 98):

Indique su edad
4 respuestas

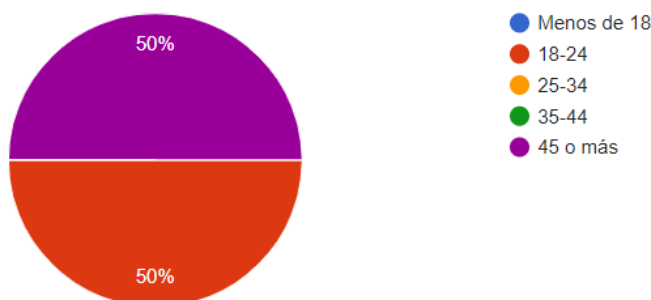


Ilustración 98: Porcentaje de la edad de los usuarios

Por otro lado, también podemos ver el porcentaje del nivel de conocimientos informáticos de cada usuario que ha realizado la encuesta (Ilustración 99):

Indique su nivel de conocimientos informáticos

4 respuestas

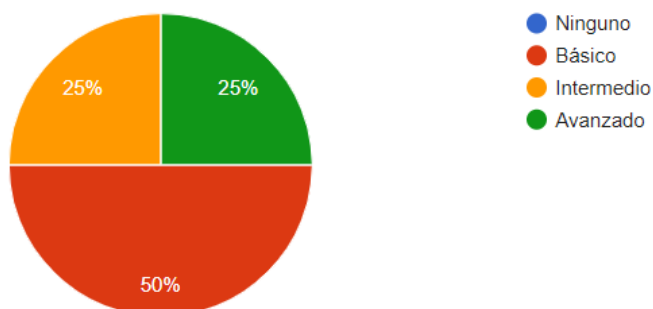


Ilustración 99: Porcentaje del nivel de conocimientos informáticos

También, se puede observar el porcentaje de la calificación que cada usuario proporciona para determinar la facilidad de uso de la interfaz (Ilustración 100):

En una escala del 1 al 5, ¿cómo calificarías la facilidad de uso de la interfaz?

4 respuestas

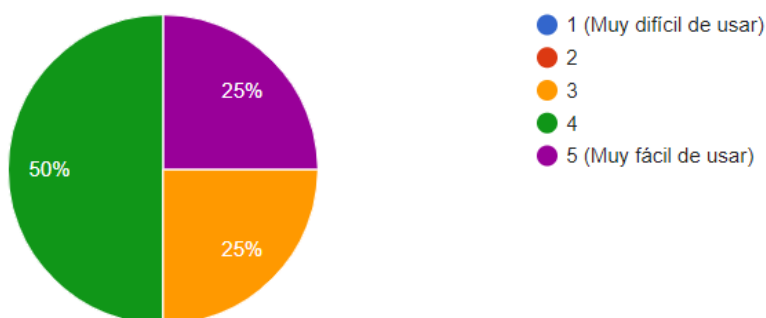


Ilustración 100: Porcentaje de la calificación de facilidad de uso de la interfaz

Por último, el porcentaje de la satisfacción general que tiene los usuarios de la aplicación se ve reflejado de la siguiente manera (Ilustración 101):

En una escala del 1 al 5, ¿cómo calificarías la satisfacción general con la aplicación?

4 respuestas

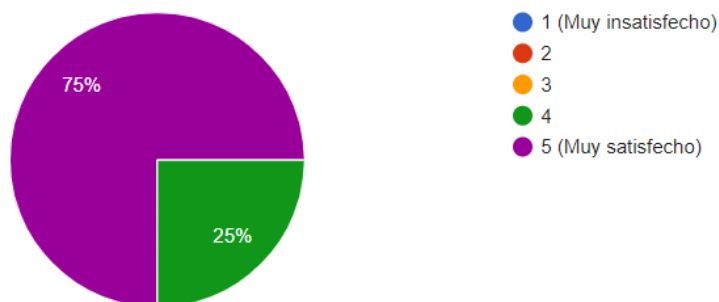


Ilustración 101: Porcentaje de la satisfacción general de la aplicación

Todos estos comentarios se han tenido en cuenta y se han utilizado para realizar las mejoras y ajustes correspondientes en la aplicación de Realidad Aumentada para mejorar la usabilidad y funcionalidad de esta. Estos cambios se desarrollan más detalladamente en la [Sección 5.6.4](#).

5.6.2 Diseño del logo de la aplicación

En esta última iteración también se ha realizado el diseño del logotipo de la aplicación. El logo se ha diseñado usando la herramienta Canva. A continuación, podemos ver el logotipo (Ilustración 102) y el logo junto con el nombre de la aplicación ARScape (Ilustración 103).

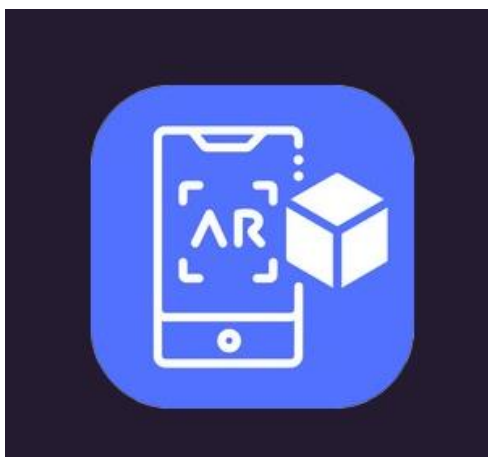


Ilustración 102: Imagen del logotipo de la aplicación



Ilustración 103: Imagen del logotipo y nombre de la aplicación

Para que aparezca este logo en la pantalla inicial de la aplicación, es necesario añadir la imagen en la configuración del proyecto de Unity, dentro de la sección Player, en el apartado de Default Icon (Ilustración 104).

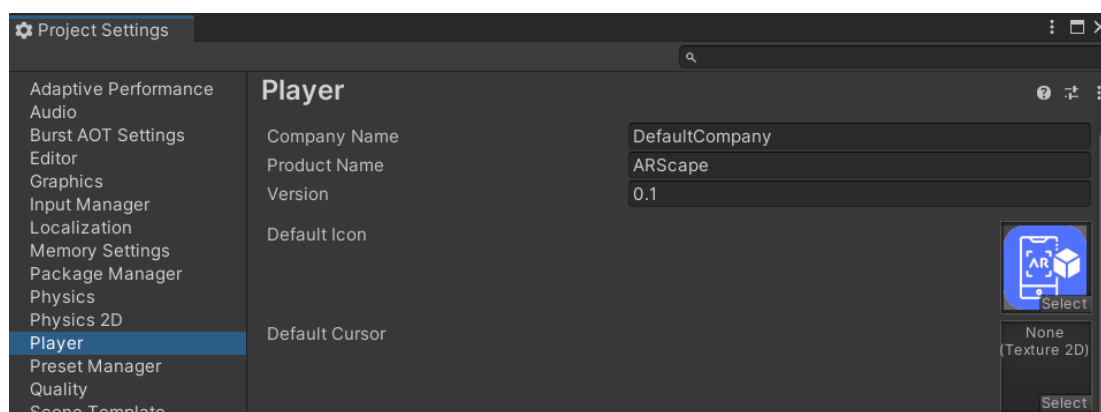


Ilustración 104: Ajustes Player del proyecto de Unity

5.6.3 Implementación cambio de idioma

Otra de las funcionalidades que se ha implementado en esta última iteración ha sido la posibilidad de cambiar el idioma de la aplicación entre español e inglés. Para realizar esto se ha instalado el paquete Localization de Unity directamente desde los paquetes que incluye Unity en su registro.

Una vez instalado, en los ajustes correspondientes a la localización, se han seleccionado los dos idiomas que se usan en la aplicación, el español y el inglés. Seguidamente se ha creado una tabla de localización de tipo String en la que se ha creado una entrada por cada texto existente en la aplicación. En cada entrada se establece el identificador o clave de dicho texto y la traducción correspondiente en cada idioma, es decir, existe una columna para la traducción en español y otra para su traducción en inglés (Ilustración 105).

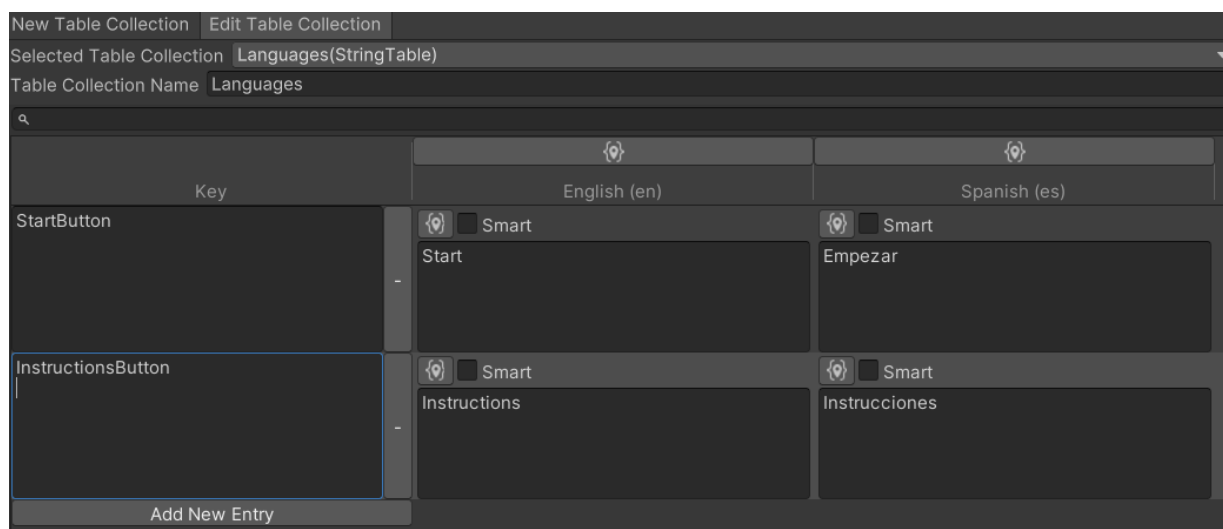


Ilustración 105: Ejemplo tabla de localización tipo String

A continuación, se muestra en una tabla todos los elementos que han sido traducidos a ambos idiomas. Esta tabla es similar a la tabla de localización de tipo String creada en el proyecto.

Clave	Inglés	Español
StartButton	START	EMPEZAR
InstructionsButton	INSTRUCTIONS	INSTRUCCIONES
Instructions	Instructions	Instrucciones
Title1Instructions	Device orientation	Orientación del dispositivo
Description1Instructions	For a better experience, please orient your device horizontally.	Para una mejor experiencia, oriente el dispositivo horizontalmente.
Title2Instructions	Model detection	Detección de modelos

Description2Instructions	To interact with a model, tap on the + button, select the desired model, and focus the model with the device's camera.	Para interactuar con un modelo, pulse en el botón +, seleccione el modelo que desee y enfoque el modelo con la cámara del dispositivo.
Title2.1Instructions	Interaction with marker	Interacción con marcador
Description2.1Instructions	To interact with the models of a marker, you need to swipe with one finger to rotate them or pinch the screen with two fingers to zoom.	Para interactuar con los modelos de los marcadores, hay que deslizar un dedo para rotarlos o pellizcar la pantalla con dos dedos para hacer zoom.
Title3Instructions	Marker detection	Detección de marcadores
Description3Instructions	To interact with a marker, focus the marker with the device's camera.	Para interactuar con un marcador, enfoque el marcador con la cámara del dispositivo.
Title4Instructions	Interaction between markers	Interacción entre marcadores
Description4Instructions	To interact with multiple markers, gather the markers and focus them with the device's camera.	Para interactuar con varios marcadores, junte los marcadores y enfóquelos con la cámara del dispositivo.

Title5Instructions	Interaction between models and markers	Interacción entre modelos y marcadores
Description5Instructions	To interact between models and markers, select the desired model and simultaneously focus on both the model and the corresponding marker.	Para interactuar entre modelos y marcadores, seleccione el modelo que desee y enfoque a la vez al modelo y al marcador correspondiente.
Title6Instructions	Interaction with models	Interacción con modelos
Description6Instructions	To get more information about a model, tap on the indicated areas of the model.	Para obtener más información sobre un modelo, pulse sobre las zonas indicadas del modelo.
SmallBridgeButton	Small bridge	Puente pequeño
ShipButton	Ship	Barco
BigBridgeButton	Big bridge	Puente grande
BridgeTitle	THE BISHOP'S BRIDGE	EL PUENTE DEL OBISPO
BridgeDescription1	Bishop's Bridge spans the Guadalquivir River, built in the 16th century, located in a village near Baeza, along the old	El puente de Obispo se encuentra sobre el río Guadalquivir, construido en el siglo XVI, y una pedanía de Baeza, en el

	route from Jaén to Baeza. The Bishop's Bridge, one of the early examples of Spanish Renaissance architecture, consists of four uneven barrel vaults made of sandstone ashlar, laid in rows of varying heights. Its profile is inclined, and it features a square tower at its highest point. The piers supporting the arches have cutwaters, and it is topped with a parapet wall.	antiguo trazado de la carretera de Jaén a Baeza. El puente del Obispo, uno de los primeros del renacimiento hispano, está formado por 4 bóvedas desiguales de cañón, de sillería de arenisca, aparejada en hiladas de altura también desigual. La rasante es inclinada, y tiene una torre cuadrangular en su extremo más alto. Las pilas que sostienen los arcos, poseen tajamares, y se remata con un pretil de fábrica.
BridgeDescription2	To view the bridge in more detail, search for and focus on the following marker:	Para ver en más detalle el puente, busque y enfoque el siguiente marcador:
TowerTitle	BISHOP'S BRIDGE TOWER	TORRE DEL PUENTE DEL OBISPO
TowerDescription1	The tower houses a chapel inside, which is accessed from the bridge's roadway through a door with archivolt. On both sides of the	La torre posee en su interior una capilla, que tiene acceso desde la calzada del puente, mediante una puerta con arquivoltas, situándose a

	door, there are inscribed plaques alluding to the Bishop of Jaén, Alonso Suárez de la Fuente del Sauce, who financed the bridge. Above the door, three shields reproduce the bishop's coat of arms. On the remaining three sides of the tower, there are flared and round-arched windows, also topped with shields.	ambos lados de la misma sendas lápidas con inscripciones alusivas al obispo de Jaén, D. Alonso Suárez de la Fuente del Sauce, que fue quien lo financió. Sobre la puerta, tres escudos reproducen las armas del prelado. En las tres caras restantes de la torre, se presentan ventanas abocinadas y de medio punto, rematadas igualmente por escudos.
TowerDescription2	To view the tower in more detail, search for and focus on the following marker:	Para ver en más detalle la torre, busque y enfoque el siguiente marcador:
SurroundingsTitle	SURROUNDINGS OF THE BISHOP'S BRIDGE	ENTORNO DEL PUENTE DEL OBISPO
SurroundingsDescription	On the A-6109 road, near the 16th-century Bishop's Bridge, lies the hamlet of Puente del Obispo. Its origins trace back to the Settlement of Colonization promoted by agrarian colonization	En la carretera A-6109, en las inmediaciones del Puente del Obispo del siglo xvi, se encuentra la pedanía de Puente del Obispo. Tiene su origen en el Poblado de Colonización impulsado por la colonización

	in Andalusia, built in the 1950s. The Puente del Obispo Settlement of Colonization is located 7 km southwest of the core of Baeza, to which it belongs administratively, and 8.5 km by road. Near the bridge, you'll find the Natural Area of Laguna Grande and the Hacienda La Laguna, dating back to the 17th century, declared a Cultural Interest Site (BIC).	agraria en Andalucía, construido en los años 1950. El Pueblo de Colonización de Puente del Obispo se sitúa 7 Km. al suroeste del núcleo de Baeza, a cuyo término municipal pertenece, y a 8,5 Km. de éste por carretera. En las cercanías del puente, se encuentran el Paraje Natural de la Laguna Grande y la Hacienda La Laguna, del siglo xvii, declarada BIC (Bien de Interés Cultural).
--	--	---

Tabla 12: Tabla de traducciones

Tras definir una entrada para cada texto junto con sus traducciones, se ha asociado cada una de estas entradas a los textos correspondientes que son utilizados dentro de la aplicación, sobretodo en la interfaz de usuario.

Para poder cambiar de idioma desde la interfaz de usuario, se ha añadido un botón en la pantalla inicial que permite abrir un panel en el que se han añadido dos botones con la bandera de cada idioma para poder cambiar entre un idioma y otro pulsando estos botones (Ilustración 106).



Ilustración 106: Panel de cambio de idioma

5.6.4 Implementación de mejoras

A partir de los comentarios de los usuarios al probar las distintas funcionalidades de la aplicación, se pueden aplicar ciertas mejoras en los aspectos en los que los usuarios han encontrado más dificultades.

Varios usuarios han comentado el problema de no saber en qué partes del modelo pulsar para obtener información. Para solucionar esto, se ha añadido a cada una de las zonas que contienen información un Canvas [60] con coordenadas del mundo, para poner una imagen identificativa que permita a los usuarios saber qué partes del modelo se pueden pulsar y obtener información asociada. Un Canvas es el área donde deben colocarse todos los elementos pertenecientes a la interfaz de usuario. Al ser de tipo World Space, también conocido como interfaz diagética, se integra con los modelos y objetos en la escena (Ilustración 107).

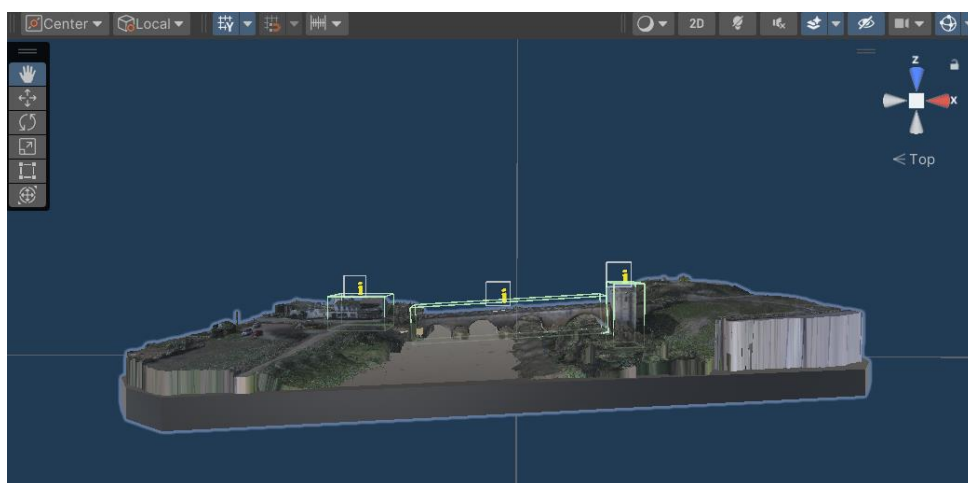


Ilustración 107: Modelo con Canvas integrados

Otros usuarios han comentado la dificultad a la hora de rotar y hacer zoom a los objetos, por lo que como solución se ha reducido la velocidad con la que se mueven y se ha añadido a las instrucciones una breve explicación de cómo interactuar con estos objetos (Ilustración 108).

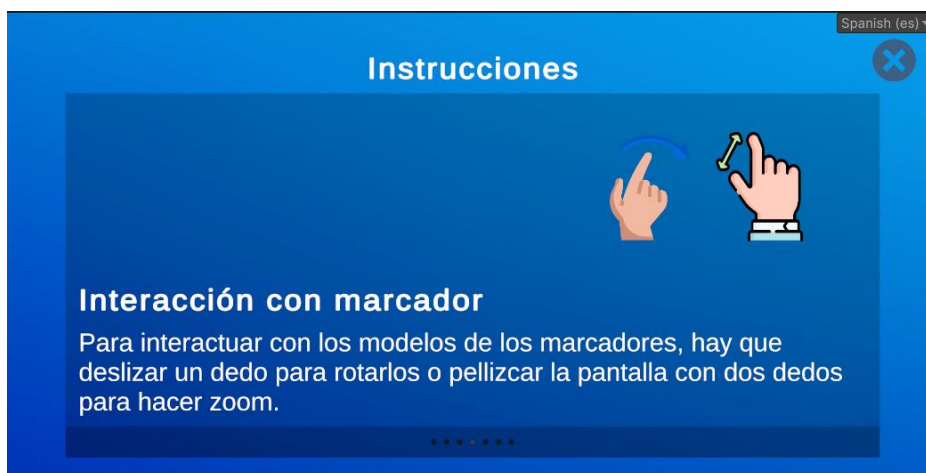


Ilustración 108: Instrucciones para interactuar con los modelos de los marcadores

Por último, también se ha realizado una mejora para optimizar el rendimiento y carga de la aplicación, de manera que no tuviese un tiempo elevado a la hora de abrirla y ejecutarla. Para ello se ha reducido el tamaño y el peso de los modelos 3D, reduciendo el número de polígonos que contiene cada malla, pero sin perder ningún tipo de detalle.

6 RESULTADOS OBTENIDOS

En esta sección se presentan todos los resultados finales obtenidos en el desarrollo de este trabajo. Se describen los resultados obtenidos tanto del proceso de impresión 3D como los resultados del prototipo de aplicación de Realidad Aumentada.

En cuanto a los resultados que se han obtenido respecto a la impresión 3D, son tres modelos o maquetas en 3D. El primer modelo es una representación del Puente del Obispo con unas dimensiones de 150 x 91,13 x 18,58 mm (Ilustración 109). Este objeto impreso es usado posteriormente como un modelo objetivo en la aplicación.



Ilustración 109: Primer modelo impreso del Puente del Obispo

El segundo objeto impreso que se ha obtenido como resultado del proceso de impresión 3D es la maqueta de un barco (Ilustración 110). Este objeto también será usado como modelo objetivo en la aplicación para reflejar distintas interacciones.



Ilustración 110: Modelo impreso del barco

Por último, se ha obtenido otra maqueta del Puente del Obispo, de mayores dimensiones y mejorada, es decir, con más detalles. Este objeto impreso en 3D tiene un tamaño de 236,37 x 146,67 x 32,24 mm (Ilustración 111). Al igual que los otros dos objetos impresos, este también será utilizado como modelo objetivo en la aplicación.

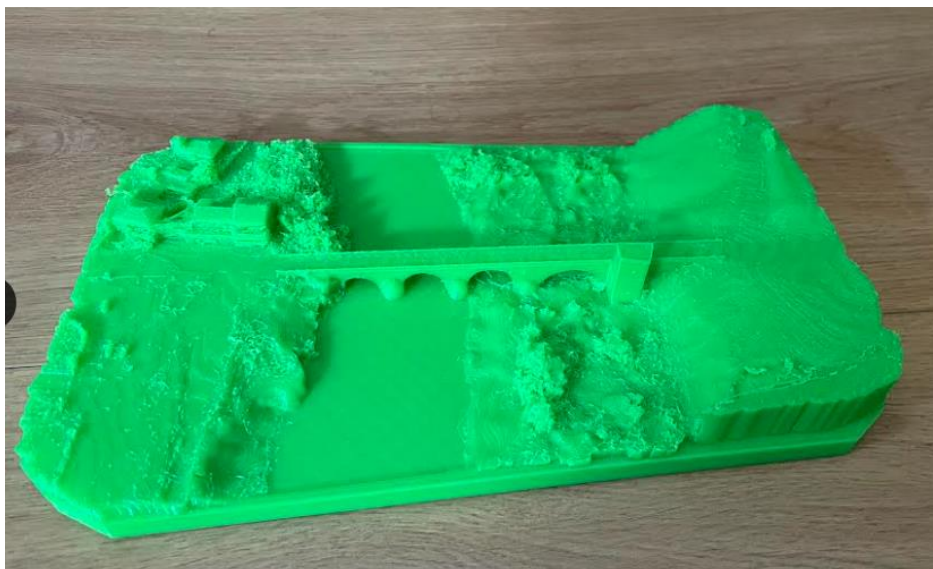


Ilustración 111: Segundo modelo impreso del Puente del Obispo

En lo que se refiere a la aplicación de Realidad Aumentada, se han obtenido diversos resultados y diferentes formas de interactuar utilizando imágenes que actúan como marcadores y maquetas u objetos 3D que actúan como modelos objetivos.

En relación a los marcadores, podemos diferenciar tres resultados diferentes. En primer lugar, al enfocar el marcador que representa una torre, aparece en pantalla un modelo tridimensional de la torre del Puente del Obispo. Esto permite a los usuarios ver el modelo con más detalle, permitiéndoles hacer zoom en dicho modelo y rotarlo en sus tres direcciones (Ilustración 112).



Ilustración 112: Marcador y modelo 3D de la torre

De la misma manera, también obtenemos un resultado similar al enfocar el marcador que representa el puente aislado, sin tener en cuenta el entorno. Al igual que en ocurre en el caso anterior, se visualiza en pantalla el modelo del puente en 3D con la posibilidad de rotarlo y hacerle zoom (Ilustración 113).



Ilustración 113: Marcador y modelo 3D del puente aislado

El tercer resultado que se obtiene resulta de la interacción de ambos marcadores. Cuando se enfocan los dos marcadores a la vez, aparece en pantalla el modelo original del Puente del Obispo (Ilustración 114).



Ilustración 114: Modelo obtenido al interactuar con dos marcadores

Siguiendo con los resultados obtenidos de la aplicación, se obtienen dos resultados a partir de la primera maqueta 3D. Al enfocar esta maqueta con la cámara, se superpone el modelo original sobre este objeto (Ilustración 115). Con esto se quiere representar la posibilidad de superponer modelos 3D digitales sobre objetos físicos reales, lo que permite infinitas posibilidades.



Ilustración 115: Maqueta y modelo 3D del Puente del Obispo reducido

El segundo resultado que obtenemos proviene de la interacción de esta maqueta con el marcador del código QR. Al enfocar esta maqueta y al marcador

simultáneamente, aparece sobre la maqueta un efecto que simula el agua del río que pasa bajo el puente (Ilustración 116).



Ilustración 116: Modelo reducido con efecto de agua

De forma similar se obtienen dos resultados a partir del modelo del barco. El primer resultado que obtenemos surge al enfocar el objeto impreso del barco, donde se superpone el modelo digitalmente del barco con un color determinado, en este caso, el azul (Ilustración 117).



Ilustración 117: Interacción con el modelo del barco

Por otro lado, al interactuar con este objeto y el marcador del código QR, se obtiene resultado como el anterior, pero cambiando el color del barco a rosa (Ilustración 118).

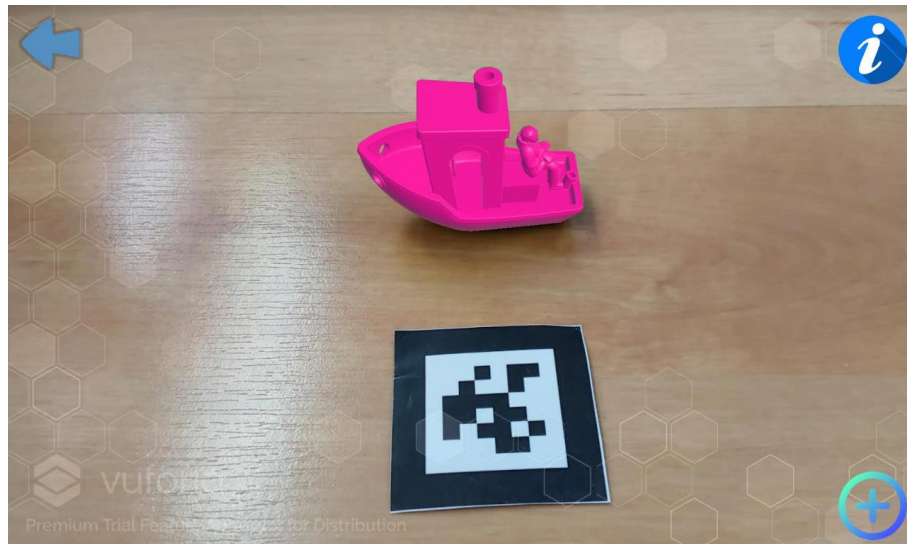


Ilustración 118: Interacción entre el barco y marcador

Por último, se encuentran los resultados obtenidos de la interacción con el modelo mejorado y de mayor tamaño del Puente del Obispo. De la misma manera que ocurre con el modelo reducido, al enfocar la maqueta impresa aparece el modelo superpuesto en pantalla. Podemos ver en la Ilustración 119 como encajan el modelo y la maqueta a la perfección.



Ilustración 119: Maqueta y modelo 3D del Puente del Obispo mejorado

Al pulsar sobre las zonas del modelo indicadas visualmente con una señal, se muestran en pantalla una serie de paneles informativos con datos relevantes sobre

dichas zonas. Podemos obtener información sobre el Puente del Obispo al pulsar sobre la zona del puente (Ilustración 120).

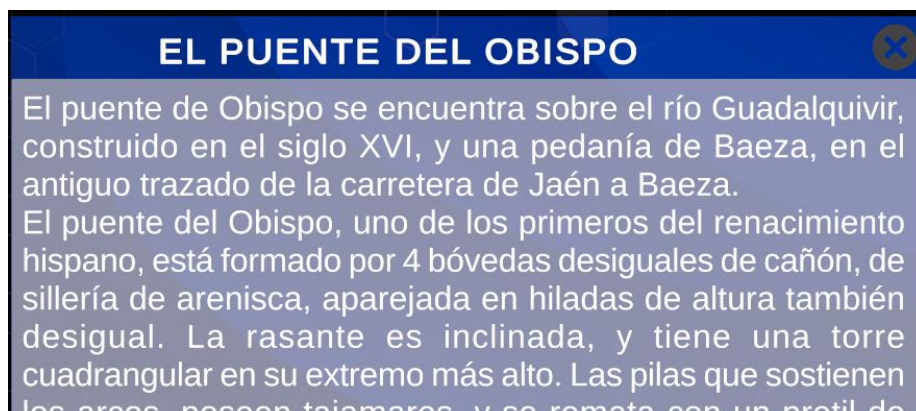


Ilustración 120: Panel informativo con datos del Puente del Obispo

También podemos obtener información sobre el entorno de esta zona rural al pulsar sobre una de las casas (Ilustración 121).

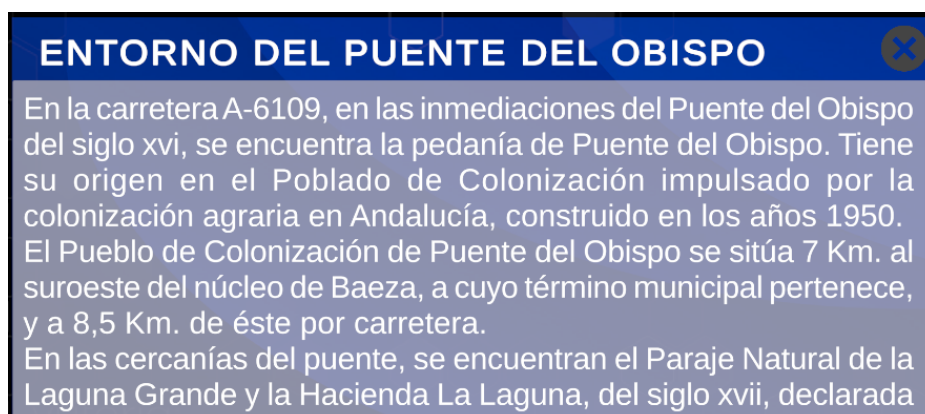


Ilustración 121: Panel informativo con datos sobre el entorno del Puente del Obispo

Por último, al pulsar sobre la torre del modelo se muestran algunos de los datos más importante de esta torre (Ilustración 122).

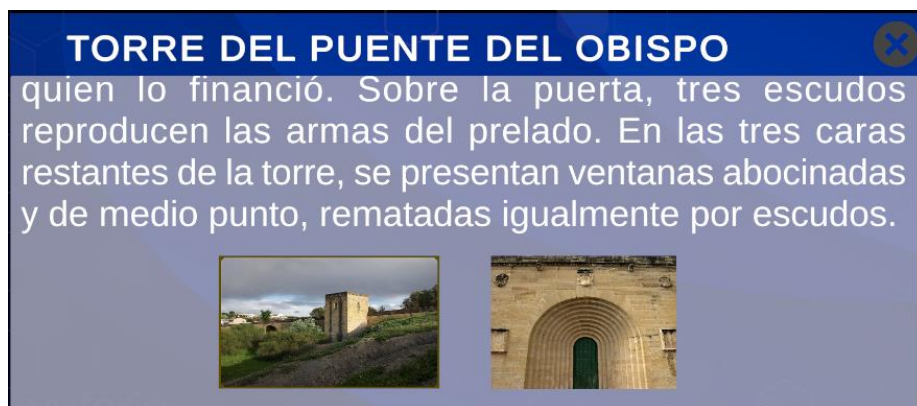


Ilustración 122: Panel informativo con datos sobre la torre del Puente del Obispo

7 CONCLUSIONES Y TRABAJOS FUTUROS

En este Trabajo de Fin de Grado (TFG) se ha desarrollado un prototipo de aplicación que combina modelos 3D impresos con Realidad Aumentada para ofrecer una experiencia interactiva y enriquecedora.

De acuerdo a los objetivos definidos al inicio de este proyecto, en la [Sección 1.1](#), podemos concluir que se han cumplido satisfactoriamente tanto el objetivo principal del proyecto, como los objetivos específicos:

- **Identificar las características necesarias de la impresora 3D y selección de la más adecuada para la maqueta:** Como se ha visto en secciones anteriores, se ha descrito detalladamente las características de la impresora, así como la configuración de la herramienta de impresión utilizadas para la impresión de las maquetas del trabajo.
- **Preparar los modelos 3D del terreno para que puedan ser impresos:** Los modelos 3D del terreno han sido modelados y preparados para su impresión, optimizándolos y adaptándolos adecuadamente, con la ayuda de las herramientas de modelado utilizadas.
- **Preparar un modelo 3D de la zona seleccionando la información que se va a proporcionar mediante Realidad Aumentada:** He seleccionado la información que he considerado más relevante sobre algunas de las zonas del entorno que se ha trabajado en este proyecto. Esta información ha sido incluida en la aplicación para ser superpuesta de forma virtual.
- **Desarrollar una herramienta de Realidad Aumentada que permita visualizar la maqueta 3D con información enriquecida procedente del modelo 3D:** En este caso se ha desarrollado una aplicación en Unity que permite la visualización de varias maquetas 3D con información adicional superpuesta, además de la posibilidad de interactuar con los modelos y marcadores.

- **Probar el prototipo con usuarios reales que permita evaluar su usabilidad:** Como se ha descrito en secciones anteriores, se ha probado la aplicación con distintos usuarios y se ha evaluado su usabilidad y funcionalidad mediante la realización de encuestas. Además, estas pruebas han permitido realizar mejoras en la aplicación.

Durante el desarrollo de este trabajo he encontrado algunas dificultades o limitaciones que he ido resolviendo a lo largo del proyecto. He encontrado problemas con la precisión del modelado 3D, por lo que he realizado múltiples pruebas de impresión para asegurar la calidad de estos modelos. Otra de las dificultades encontradas ha sido la compatibilidad de dispositivos con Realidad Aumentada, ya que solo algunos dispositivos son compatibles con esta tecnología, por lo que se ha tenido que elegir el más adecuado. Además, se experimentaron algunos fallos técnicos con la impresora 3D que retrasaron el inicio del proceso de impresión.

La aplicación desarrollada, al ser un prototipo, permite muchas posibilidades de ampliación e implementaciones futuras. La combinación con la impresión 3D permite poder integrar y cambiar los modelos 3D al gusto del usuario y de una manera sencilla. Se puede ampliar la funcionalidad de la aplicación para incluir más detalles sobre otros entornos o estructuras, como se mencionó con la posibilidad de modelar olivos y experimentar con dichos entornos gracias a las infinitas posibilidades que ofrece la interacción de estos modelos y marcadores en Realidad Aumentada.

Las interacciones entre marcadores y modelos pueden ser ampliamente mejoradas integrando nuevos efectos, animaciones, nuevos modelos, etcétera, lo que permite nuevas formas de interactuar y nuevos resultados. Se pueden diseñar tantos marcadores como se deseen, cada uno con una funcionalidad específica.

Todo esto puede ser aplicado en muchos campos como en la agricultura, la educación, la arquitectura e incluso la medicina. Las posibilidades son infinitas y permiten una personalización y diseño bastante personal y creativo.

En conclusión, los resultados obtenidos en este proyecto demuestran el potencial de combinar tecnologías de impresión 3D y Realidad Aumentada para crear experiencias educativas y visuales interactivas. Se han logrado maquetas impresas en 3D y un prototipo de aplicación que permite visualizar información adicional

superpuesta en estas maquetas, enriqueciendo la comprensión del entorno representado. El proyecto puede extenderse incluyendo más funcionalidades interactivas, mejorando la precisión de los modelos 3D y ampliando la compatibilidad con diferentes dispositivos de Realidad Aumentada. Este trabajo puede servir como base para futuros desarrollos que puedan explorar más a fondo las posibilidades de estas tecnologías combinadas.

8 APÉNDICES

En esta sección se incluyen aquellos aparatos que son de utilidad para comprender el proyecto, pero no encajaban adecuadamente en secciones anteriores. Se detalla la guía o propuesta original de este trabajo, la instalación y configuración del sistema, así como el manual de usuario.

8.1 Guía original del Trabajo Fin de Título

A continuación, se incluye la propuesta original de este Trabajo de Fin de Grado, publicada en la web de la Escuela Politécnica Superior de Jaén (Ilustración 123). Esta guía no ha sido modificada desde su publicación.

(Cód.: 21/22-4208) Prototipo para la visualización de entornos naturales basado en la combinación de las tecnologías de impresión 3D y realidad aumentada

Tutor del TFG: **JUAN ROBERTO JIMENEZ PEREZ**

Modalidad: Proyecto de Ingeniería | Tipo: TFG Específico

Número máximo de estudiantes: 1 (1 asignados)

Idioma: Castellano

Asignado al alumno con DNI:78162613B

Segundo tutor del TFG: **DAVID JURADO RODRIGUEZ**

En el contexto de la visualización 3D inmersiva de escenarios reales o simulados, nos podemos encontrar con dos enfoques diferentes, uno es la realidad virtual, en el que el usuario se introduce e interactúa en un mundo completamente virtual, y otro es la combinación de dos tecnologías que llevan bastante tiempo siendo ampliamente demandadas, la impresión 3D + realidad aumentada.

En este trabajo, se propone explorar este segundo enfoque generando una maqueta 3D que represente un entorno natural, preparado para superponer mediante Realidad Aumentada información relevante del modelo 3D previamente generado.

Conocimientos Previos

Objetivos del TFG

Objetivo principal:

- Obtener un prototipo que permita visualizar los detalles de un entorno 3D combinando una maqueta de dicho entorno y una herramienta de realidad aumentada.

Objetivos específicos:

- Identificar las características necesarias de la impresora 3D y selección de la más adecuada para la maqueta.
- Preparar los modelos 3D del terreno para que puedan ser impresos.
- Preparar un modelo 3D de la zona seleccionando la información que se va a proporcionar mediante realidad aumentada.
- Desarrollar una herramienta de realidad aumentada que permita visualizar la maqueta 3D con información enriquecida procedente del modelo 3D.
- Probar el prototipo con usuarios reales que permita evaluar su usabilidad.

Metodología a Desarrollar

- Repaso bibliográfico de los procesos de impresión 3D y de realidad aumentada. En particular, en aquellos proyectos en los que se utilizan estas tecnologías de manera conjunta.
- Selección de impresora 3D.
- Selección de herramienta para el desarrollo de aplicaciones de realidad aumentada.
- Análisis y especificación de requisitos del sistema
- Planificación y presupuesto.
- Diseño del sistema
- Implementación y pruebas.
- Elaboración de la memoria del trabajo

Documentos y Formatos de Entrega

- Memoria del TFG. (PDF)
- Código fuente y ejecutables.
- Video demostrativo del prototipo.
- Anexos para la presentación del TFG:
 - Informe favorable del tutor.
 - Autorización de publicación.

Ilustración 123: Propuesta original del TFG

8.2 Instalación y configuración del sistema

Para imprimir los modelos o maquetas 3D, es necesario instalar una herramienta de impresión 3D compatible con la impresora que se utilice. En este caso se ha utilizado PrusaSlicer. También son necesario los modelos en formato STL o el código G-code para imprimirlos directamente. Estos archivos están accesibles en el siguiente enlace:

https://drive.google.com/drive/folders/16KNXXuleNs_R7dDDdbRo_UPmC6gmaqD6?usp=sharing

En cuanto a la aplicación de Realidad Aumentada, para instalarla es necesario contar con un dispositivo compatible con ARCore. Google ofrece una lista en su sitio web (<https://developers.google.com/ar/devices?hl=es-419>) para conocer qué dispositivos son compatibles con esta tecnología. En este caso se ha utilizado el dispositivo Samsung Galaxy Tab S6 Lite. Además, es necesario que el dispositivo tenga una versión de Android correcta, en este caso superior o igual a la versión 8.0. Para instalar la aplicación basta con descargar el APK disponible en el siguiente enlace:

<https://drive.google.com/file/d/1jzIWPykdqzgwd56ORxbrpfCd8w3accl0/view?usp=sharing>

El proyecto de Unity y el código fuente de la aplicación se encuentra en el siguiente enlace:

<https://drive.google.com/file/d/1ngWldywua07jwCSVZtyBDnATjqsHv-LI/view?usp=sharing>

8.3 Manual de usuario

En este apartado se especifican las instrucciones de uso de la aplicación para facilitar su utilización, de manera que se puedan probar todas las funcionalidades implementadas sin ningún tipo de problema.

En primer lugar, al abrir la aplicación aparece la pantalla con el menú principal (Ilustración 124), que permite tres opciones: empezar con la aplicación en sí, mostrar las instrucciones o cambiar el idioma de la aplicación.



Ilustración 124: Pantalla del menú de la aplicación

Al pulsar en el botón de arriba a la derecha, se abre la pantalla para cambiar el idioma de la aplicación (Ilustración 125). En esta pantalla aparece la bandera de cada idioma y al clicar sobre ellas se cambia al idioma seleccionado. También cuenta con un botón en la esquina superior derecha para cerrar esta pantalla.



Ilustración 125: Pantalla cambio de idioma de la aplicación

Por otro lado, si pulsamos sobre el botón de las instrucciones, se abre otra pantalla que contiene un carrusel con las distintas instrucciones de uso de la aplicación (Ilustración 126). Esta pantalla también cuenta con un botón para cerrarla.

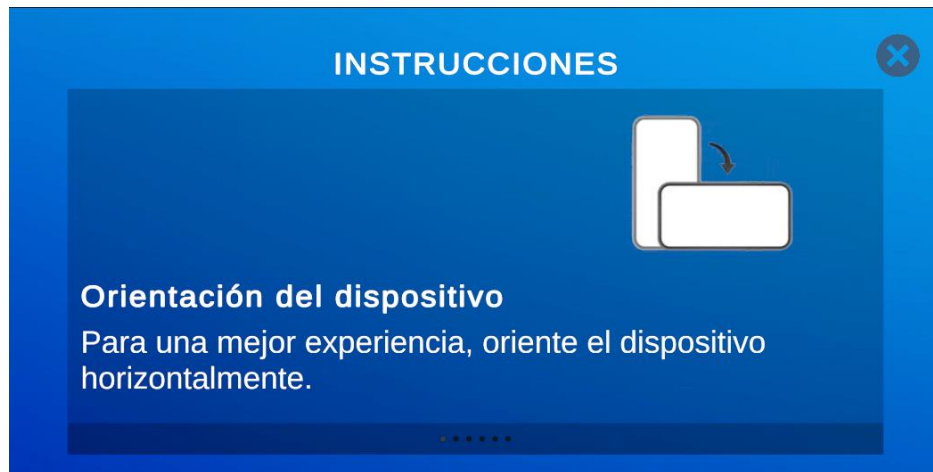


Ilustración 126: Pantalla instrucciones de uso de la aplicación

Y si clicamos sobre el botón “EMPEZAR” se muestra la pantalla principal de la aplicación (Ilustración 127). En la interfaz de esta pantalla se encuentran varios componentes. Arriba a la derecha se encuentra un botón que abre la pantalla de las instrucciones y a la izquierda hay una flecha que permite volver a la pantalla anterior.



Ilustración 127: Pantalla principal de la aplicación

Si se pulsa sobre el botón de la esquina inferior derecha, se abre un pequeño panel en el que se muestran los distintos modelos disponibles y al pulsar en cada uno de estos botones, se activa el modelo correspondiente (Ilustración 128).

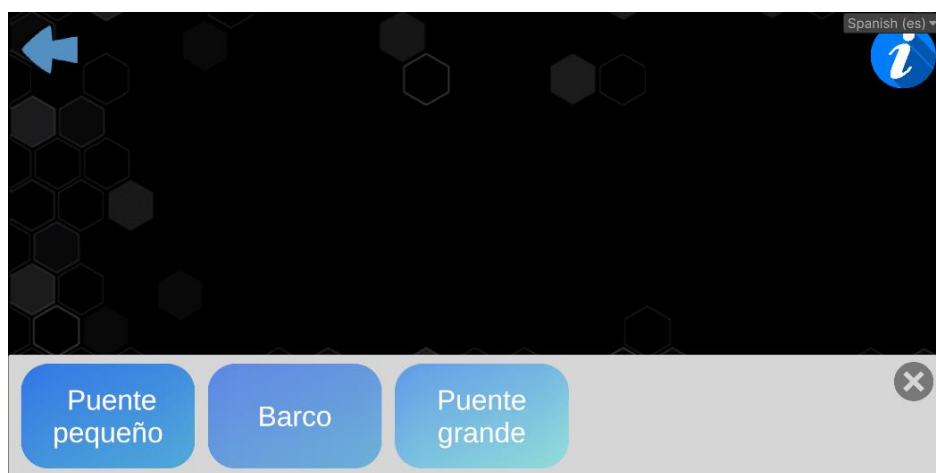


Ilustración 128: Pantalla principal con menú de modelos

Como ya se ha comentado en secciones anteriores, en esta pantalla principal tienen lugar todas las interacciones posibles con los modelos y marcadores. Si se apunta con la cámara a un marcador, se puede interactuar con el modelo que aparece sobre este marcador utilizando los dedos, deslizando un dedo para rotar el modelo y pellizcando la pantalla con dos dedos para acercar o alejar el modelo. Además, al juntar dos marcadores se pueden observar modelos diferentes a los individuales.

Por otra parte, si se enfoca alguno de los modelos también se puede interactuar con ellos y con los marcadores a la vez. Si enfocamos el modelo del puente pequeño, se puede clicar sobre las zonas indicadas del modelo para obtener información sobre ellas. También se puede enfocar este modelo y al marcador del código QR a la vez para visualizar un efecto que simula el agua del río. Con el modelo del barco también podemos interactuar simultáneamente con el marcador del código QR para cambiar el color del barco. Por último, al enfocar con la cámara del dispositivo al modelo del puente de mayor tamaño, también aparece el modelo en 3D con indicadores sobre las zonas que pueden ser pulsadas. Al clicar sobre ellas aparecen paneles con información relevante de cada parte del modelo.

En el siguiente enlace se puede encontrar un vídeo en el que se muestra el funcionamiento de la aplicación:

<https://drive.google.com/file/d/1ihK4FTSILOBeWQzyxYN-dl2kcSJbnT3/view?usp=sharing>

9 DEFINICIONES Y ABREVIATURAS.

9.1 Definiciones

- **Modelado 3D:** Proceso de crear una representación tridimensional de cualquier objeto mediante software especializado. Este proceso se utiliza en áreas como la animación, los videojuegos, la ingeniería y la Realidad Aumentada, permitiendo la visualización y manipulación de objetos en un espacio tridimensional.
- **Impresión 3D:** Tecnología de fabricación aditiva que permite la creación de objetos tridimensionales a través de la superposición de capas sucesivas de material. Esta técnica es utilizada para prototipos rápidos, producción de piezas finales y en campos como la medicina y la arquitectura.
- **Realidad Aumentada:** Tecnología que permite superponer elementos virtuales sobre el entorno real, a través de dispositivos como smartphones o gafas especiales, para proporcionar información adicional o mejorar la experiencia del usuario.
- **Realidad Virtual:** Entorno de escenas y objetos de apariencia real generado mediante tecnología informática que simula una experiencia de inmersión completa para el usuario. Este entorno es interactivo y responde a los movimientos y acciones del usuario.
- **Realidad Mixta:** Combina aspectos de la Realidad Aumentada y la Realidad Virtual para crear un entorno donde objetos físicos y digitales coexisten e interactúan en tiempo real, proporcionando una experiencia más integrada y cohesiva.
- **Renderizado:** Proceso de generar una imagen o modelo visual a partir de un modelo 3D utilizando software de computadora. El renderizado puede ser en tiempo real o prerenderizado, dependiendo de la aplicación y los requisitos del proyecto.
- **Prototipado Rápido:** Técnica de fabricación que permite la creación rápida de un modelo físico a partir de datos tridimensionales. Se utiliza

para evaluar el diseño, la forma y la funcionalidad de un objeto antes de su producción en masa.

- **Tracking:** Proceso mediante el cual se determina la posición y orientación de un dispositivo en el espacio. En aplicaciones de Realidad Aumentada, el tracking es esencial para superponer correctamente los elementos virtuales en el entorno real.
- **Oclusión:** Técnica en Realidad Aumentada que permite que los objetos virtuales interactúen de manera más realista con el entorno físico, ocultándose detrás de objetos reales cuando es necesario.
- **Shader:** Programa que se ejecuta en la GPU y que se utiliza para calcular los efectos de iluminación y color en los gráficos 3D. Los shaders son fundamentales para lograr realismo en la representación de modelos tridimensionales.
- **Fotogrametría:** Técnica de medición que permite obtener información precisa sobre las propiedades geométricas de los objetos y el entorno a partir de fotografías.

9.2 Abreviaturas

- **RA/AR (Realidad Aumentada/Augmented Reality):** Tecnología que permite superponer elementos virtuales e información digital sobre el entorno real.
- **RAD (Desarrollo Rápido de Aplicaciones):** Metodología que se centra en el desarrollo de aplicaciones rápidamente mediante iteraciones frecuentes.
- **MR (Mixed Reality):** La realidad mixta es la combinación de Realidad Aumentada y Realidad Virtual.
- **VR (Virtual Reality):** La Realidad Virtual es un entorno de escenas y objetos de apariencia real, generado mediante tecnología informática.

- **IDE (Entorno de Desarrollo Integrado):** Es una aplicación de software que ayuda a los programadores a desarrollar código de software de manera eficiente.
- **GLB:** Es la representación en formato de archivo binario de modelos 3D con formato de transmisión GL (glTF).
- **STL (Standard Triangle Language):** Es un formato de archivo informático de diseño asistido por computadora (CAD) que define la geometría de objetos 3D.
- **MTG (Model Target Generator):** Herramienta de Vuforia que permite la generación de modelos objetivos a partir de objetos físicos y modelos 3D.

10 BIBLIOGRAFÍA

- [1] Wikipedia, «Realidad aumentada - Wikipedia, la enciclopedia libre,» 2023. [En línea]. Available: https://es.wikipedia.org/wiki/Realidad_aumentada. [Último acceso: 15 03 2024].
- [2] F. y. Borko, Handbook of augmented reality, Springer Science & Business Media, 2011.
- [3] R. Gallego Delgado, N. Saura Parra y P. M. Núñez Trujillo, AR-Learning: Libro interactivo basado en realidad aumentada con aplicación a la enseñanza, Universidad de Extremadura, 2012.
- [4] Wikipedia, «Impresión 3D- Wikipedia, la enciclopedia libre,» 2023. [En línea]. Available: https://es.wikipedia.org/wiki/Impresi%C3%B3n_3D. [Último acceso: 17 03 2024].
- [5] F. R. Alfredo Bordignon, Á. Hahn y A. A. Iglesias, Diseño e impresión de objetos 3D: Una guía de apoyo a escuelas, UNIPE: Editorial Universitaria, 2018.
- [6] V. Lara Prieto, E. Bravo Quirino, M. Á. Rivera Campa y J. E. Gutiérrez Arredondo, «An Innovative Self-learning Approach to 3D Printing Using Multimedia and Augmented Reality on Mobile Devices,» vol. 75, Procedia Computer Science, 2015, pp. 59-65.
- [7] Unity Technologies, «Plataforma de desarrollo en tiempo real de Unity | Visualizaciones de VR y AR en 2D y 3D,» [En línea]. Available: <https://unity.com/es>. [Último acceso: 05 06 2024].
- [8] Innovae, «Realidad Aumentada,» 2023. [En línea]. Available: <https://www.innovae.com/la-realidad-aumentada/>. [Último acceso: 21 03 2024].
- [9] Wikipedia, «Sensorama - Wikipedia, la enciclopedia libre,» 2023. [En línea]. Available: <https://es.wikipedia.org/wiki/Sensorama>. [Último acceso: 23 03 2024].
- [10] J. C. Vázquez Paulino, M. Á. Backhoff Pohls, E. M. Morales Bautista y J. O. González Moreno, Estado del arte de la realidad aumentada y su aplicación al transporte carretero, Sanfandila, Qro, 2022.
- [11] J. Pérez y V. Kreinovich, Gartner's Hype Cycle: A Simple Explanation, Departmental Technical Reports (CS), 2018.
- [12] R. D. Buitrago Pulido, Estado del arte: Realidad aumentada con fines educativos, Escuela Colombiana de Carreras Industriales (ECCI), 2013.

- [13] Wikipedia, «Chuck Hull - Wikipedia, la enciclopedia libre,» 2023. [En línea]. Available: https://es.wikipedia.org/wiki/Chuck_Hull. [Último acceso: 24 03 2024].
- [14] Impresoras3D, «Breve Historia de la impresión 3D,» 2023. [En línea]. Available: <https://www.impresoras3d.com/breve-historia-de-la-impresion-3d/>. [Último acceso: 26 03 2024].
- [15] M. L. Quemada Villagómez, L. A. Alcaraz Caracheo, G. Ruíz Mondragón y R. Rodríguez Castro, Revisión del estado de arte de la fabricación de multimateriales por medio de impresión 3D, Pistas Educativas, 2017.
- [16] Stratasys, «Make additive work for you,» 2024. [En línea]. Available: <https://www.stratasys.com/en/>. [Último acceso: 27 03 2024].
- [17] M. Berchon y B. Luyt, Impresión 3D: Guía definitiva para makers, diseñadores, estudiantes, profesionales, artistas y manitas en general, GG, 2014.
- [18] Mordor Intelligence, «Informe del mercado de realidad aumentada,» 2024. [En línea]. Available: <https://www.mordorintelligence.com/es/industry-reports/augmented-reality-market>. [Último acceso: 21 03 2024].
- [19] Mordor Intelligence, «Informe del mercado de impresión 3D,» 2024. [En línea]. Available: <https://www.mordorintelligence.com/es/industry-reports/automated-3d-printing-market>. [Último acceso: 21 03 2024].
- [20] Business Research, «Descripción general del informe de mercado de plástico de impresión 3D,» 2023. [En línea]. Available: <https://www.businessresearchinsights.com/es/market-reports/3d-printing-plastic-market-106580>. [Último acceso: 23 03 2024].
- [21] L. Xiaosong, L. Shushi, C. Wenjun y F. Songjiang, «The Application of Risk Matrix to Software Project Risk Management,» de *International Forum on Information Technology and Applications*, 2009.
- [22] Vuforia Engine, «Vuforia Engine Developer Portal,» [En línea]. Available: <https://developer.vuforia.com/>. [Último acceso: 05 06 2024].
- [23] Microsoft, «Microsoft Visual Studio,» [En línea]. Available: <https://visualstudio.microsoft.com/es/#vs-section>. [Último acceso: 25 05 2024].
- [24] Wikipedia, «Unity (motor de videojuegos) - Wikipedia, la enciclopedia libre,» 2024. [En línea]. Available: [https://es.wikipedia.org/wiki/Unity_\(motor_de_videojuego\)](https://es.wikipedia.org/wiki/Unity_(motor_de_videojuego)). [Último acceso: 28 04 2024].
- [25] Asset Store, «Unity Asset Store,» [En línea]. Available: <https://assetstore.unity.com/>. [Último acceso: 06 06 2024].

- [26] AR Foundation, «Unity, AR Foundation,» [En línea]. Available: <https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@6.0/manual/index.html>. [Último acceso: 07 06 2024].
- [27] Realidad aumentada, «Unity, Realidad Aumentada,» [En línea]. Available: <https://unity.com/es/unity/features/ar>. [Último acceso: 04 06 2024].
- [28] Vuforia, «Vuforia, Unity Documentation,» [En línea]. Available: <https://docs.unity3d.com/es/2018.4/Manual/vuforia-sdk-overview.html>. [Último acceso: 03 06 2024].
- [29] Model Target Generator, «Vuforia, Model Target Generator User Guide,» [En línea]. Available: <https://developer.vuforia.com/library/objects/model-target-generator-user-guide>. [Último acceso: 02 06 2024].
- [30] Wikipedia, «Microsoft Visual Studio - Wikipedia, la enciclopedia libre,» [En línea]. Available: https://es.wikipedia.org/wiki/Microsoft_Visual_Studio. [Último acceso: 25 05 2024].
- [31] Blender Foundation, «Blender - Home of the Blender project - Free and Open 3D Creation Software,» [En línea]. Available: <https://www.blender.org/>. [Último acceso: 29 05 2024].
- [32] Meshmixer, «Meshmixer - Powerful, free 3D software,» [En línea]. Available: <https://meshmixer.com/>. [Último acceso: 30 05 2024].
- [33] GIMP, «GIMP - GNU Image Manipulation Program,» [En línea]. Available: <https://www.gimp.org/>. [Último acceso: 30 05 2024].
- [34] Prusa Research, «PrusaSlicer,» [En línea]. Available: https://www.prusa3d.com/es/pagina/prusaslicer_424/. [Último acceso: 01 06 2024].
- [35] Wikipedia, «Blender - Wikipedia, la enciclopedia libre,» [En línea]. Available: <https://es.wikipedia.org/wiki/Blender>. [Último acceso: 27 05 2024].
- [36] 3Dnatives, el sitio web de la impresión 3D, «MeshMixer, el software gratuito de modelado para superficies 3D,» [En línea]. Available: <https://www.3dnatives.com/es/meshmixer-software-modelado-261020202/>. [Último acceso: 02 06 2024].
- [37] Wikipedia, «GIMP - Wikipedia, la enciclopedia libre,» [En línea]. Available: <https://es.wikipedia.org/wiki/GIMP>. [Último acceso: 17 04 2024].
- [38] Wikipedia, «Metodología de desarrollo de software - Wikipedia, la enciclopedia libre,» 2024. [En línea]. Available: https://es.wikipedia.org/wiki/Metodolog%C3%ADa_de_desarrollo_de_software. [Último acceso: 25 03 2024].
- [39] B. Molina Montero, H. Vite Cevallos y J. Dávila Cuesta, Metodologías ágiles frente a las tradicionales en el proceso de desarrollo de software, Espirales revista multidisciplinaria de investigación, 2018.

- [40] E. G. Maida y J. Pacienza, Metodologías de desarrollo de software. Tesis Final de Licenciatura en Sistemas y Computación, UCA, 2015.
- [41] O. Tinoco Gómez, P. P. Rosales López y J. Salas Bacalla, «Criterios de selección de metodologías de desarrollo software,» vol. 13, Industrial Data, 2010, pp. 70-74.
- [42] Boletín Oficial del Estado (BOE), «Resolución de 27 de marzo de 2024, de la Dirección General de Trabajo, por la que se registra y publica el XVIII Convenio colectivo estatal de empresas de consultoría, tecnologías de la información y estudios de mercado y de la opinión pública,» [En línea]. Available: <https://www.boe.es/buscar/doc.php?id=BOE-A-2024-7019>. [Último acceso: 04 06 2024].
- [43] Seguridad Social, «Seguridad Social: Cotización / Recaudación de Trabajadores,» [En línea]. Available: seg-social.es/wps/portal/wss/internet/Trabajadores/CotizacionRecaudacionTrabajadores. [Último acceso: 01 06 2024].
- [44] Prusa Research by Josef prusa, «Capas y perímetros | Prusa Knowledge Base,» [En línea]. Available: https://help.prusa3d.com/es/article/capas-y-perimetros-_1748. [Último acceso: 13 05 2024].
- [45] Prusa Research by Josef Prusa, «Relleno | Prusa Knowledge Base,» [En línea]. Available: https://help.prusa3d.com/es/article/relleno_42. [Último acceso: 13 05 2024].
- [46] Prusa Research by Josef Prusa, «Material de soporte | Prusa Knowledge Base,» [En línea]. Available: https://help.prusa3d.com/es/article/material-de-soporte_1698. [Último acceso: 13 05 2024].
- [47] Unity Technologies, «Download the Unity Hub to start your creative projects,» [En línea]. Available: <https://unity.com/es/download>. [Último acceso: 22 03 2024].
- [48] Unity, «Unity Manual, Installation,» [En línea]. Available: <https://docs.unity3d.com/Packages/com.unity.cloud.gltfast@6.1/manual/installation.html>. [Último acceso: 27 04 2024].
- [49] Vuforia Engine, «Getting Started with Vuforia Engine,» [En línea]. Available: <https://developer.vuforia.com/library/getting-started/getting-started-vuforia-engine-unity>. [Último acceso: 06 04 2024].
- [50] Vuforia Engine, «Downloads. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/downloads/sdk>. [Último acceso: 27 05 2024].
- [51] Vuforia Engine, «Image Targets. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/library/objects/image-targets>. [Último acceso: 17 05 2024].

- [52] Vuforia Engine, «Model Targets. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/library/objects/model-targets>. [Último acceso: 21 05 2024].
- [53] Vuforia Engine, «Model Targets supported objects CAD model best practices. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/library/model-targets/model-targets-supported-objects-cad-model-best-practices>. [Último acceso: 22 05 2024].
- [54] Vuforia Engine, «Model Target App User Guide. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/library/model-targets/model-target-test-app-user-guide>. [Último acceso: 09 05 2024].
- [55] Vuforia Engine, «Best practices designing and developing image based targets. Vuforia Engine, developer portal,» [En línea]. Available: <https://developer.vuforia.com/library/objects/best-practices-designing-and-developing-image-based-targets>. [Último acceso: 08 06 2024].
- [56] Vuforia API Reference, «Vuforia Engine, ObserverBehaviour, OnTargetStatusChanged,» [En línea]. Available: https://developer.vuforia.com/library/sites/default/files/references/unity/class_Vuforia_1_1ObserverBehaviour.html#aaccfb9680d9dca08eb26830207d0a1b9. [Último acceso: 07 06 2024].
- [57] Unity Documentation, «Unity Manual, Box Collider,» [En línea]. Available: <https://docs.unity3d.com/es/2018.4/Manual/class-BoxCollider.html>. [Último acceso: 13 05 2024].
- [58] Unity Documentation, «Unity Manual, Physics.Raycast,» [En línea]. Available: <https://docs.unity3d.com/ScriptReference/Physics.Raycast.html>. [Último acceso: 22 05 2024].
- [59] Unity Documentation, «Unity Manual, Material,» [En línea]. Available: <https://docs.unity3d.com/es/2018.4/Manual/class-Material.html>. [Último acceso: 01 05 2024].
- [60] Unity Documentation, «Unity Manual, Canvas,» [En línea]. Available: <https://docs.unity3d.com/es/2019.4/Manual/UICanvas.html>. [Último acceso: 17 05 2024].