



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

INTERFAZ DE USUARIO ORIENTADA A PROYECTOS DE REALIDAD
AUMENTADA PARA DISPOSITIVOS MÓVILES

CALVA VICENTE DARWIN ESTEBAN
INGENIERO DE SISTEMAS

MACHALA
2021



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

INTERFAZ DE USUARIO ORIENTADA A PROYECTOS DE
REALIDAD AUMENTADA PARA DISPOSITIVOS MÓVILES

CALVA VICENTE DARWIN ESTEBAN
INGENIERO DE SISTEMAS

MACHALA
2021



UTMACH

FACULTAD DE INGENIERÍA CIVIL

CARRERA DE INGENIERÍA DE SISTEMAS

TRABAJO TITULACIÓN
PROPUESTAS TECNOLÓGICAS

INTERFAZ DE USUARIO ORIENTADA A PROYECTOS DE REALIDAD
AUMENTADA PARA DISPOSITIVOS MÓVILES

CALVA VICENTE DARWIN ESTEBAN
INGENIERO DE SISTEMAS

RIVAS ASANZA WILMER BRAULIO

MACHALA, 24 DE SEPTIEMBRE DE 2021

MACHALA
2021

darwin calva

INFORME DE ORIGINALIDAD

5%

INDICE DE SIMILITUD

5%

FUENTES DE INTERNET

2%

PUBLICACIONES

2%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	documentop.com Fuente de Internet	1%
2	worldwidescience.org Fuente de Internet	1%
3	Submitted to Universidad de Cundinamarca Trabajo del estudiante	<1%
4	Submitted to Universidad Pablo de Olavide Trabajo del estudiante	<1%
5	www.coursehero.com Fuente de Internet	<1%
6	repositorio.ucv.edu.pe Fuente de Internet	<1%
7	dspace.ucuenca.edu.ec Fuente de Internet	<1%
8	MGTI Luz Maria Hernandez Cruz, M. en C. Guadalupe Manuel Estrada Segovia, Mtra. Diana Concepcion Mex Alvarez, Mtro. Jose Ramon Cab Chan et al. "Analysis of the	<1%

CLÁUSULA DE CESIÓN DE DERECHO DE PUBLICACIÓN EN EL REPOSITORIO DIGITAL INSTITUCIONAL

El que suscribe, CALVA VICENTE DARWIN ESTEBAN, en calidad de autor del siguiente trabajo escrito titulado INTERFAZ DE USUARIO ORIENTADA A PROYECTOS DE REALIDAD AUMENTADA PARA DISPOSITIVOS MÓVILES, otorga a la Universidad Técnica de Machala, de forma gratuita y no exclusiva, los derechos de reproducción, distribución y comunicación pública de la obra, que constituye un trabajo de autoría propia, sobre la cual tiene potestad para otorgar los derechos contenidos en esta licencia.

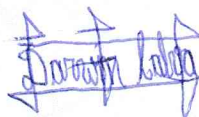
El autor declara que el contenido que se publicará es de carácter académico y se enmarca en las disposiciones definidas por la Universidad Técnica de Machala.

Se autoriza a transformar la obra, únicamente cuando sea necesario, y a realizar las adaptaciones pertinentes para permitir su preservación, distribución y publicación en el Repositorio Digital Institucional de la Universidad Técnica de Machala.

El autor como garante de la autoría de la obra y en relación a la misma, declara que la universidad se encuentra libre de todo tipo de responsabilidad sobre el contenido de la obra y que asume la responsabilidad frente a cualquier reclamo o demanda por parte de terceros de manera exclusiva.

Aceptando esta licencia, se cede a la Universidad Técnica de Machala el derecho exclusivo de archivar, reproducir, convertir, comunicar y/o distribuir la obra mundialmente en formato electrónico y digital a través de su Repositorio Digital Institucional, siempre y cuando no se lo haga para obtener beneficio económico.

Machala, 24 de septiembre de 2021



CALVA VICENTE DARWIN ESTEBAN
1105939282

DEDICATORIA

El presente trabajo va dedicado a mis padres, hermanos profesores y compañeros y otras personas que brindaron su apoyo durante los periodos de clases.

Darwin Esteban Calva Vicente

AGRADECIMIENTO

Agradezco a Dios y todas las personas que con su apoyo fue posible cursar los diez semestres que forman parte de la carrera.

A mis hermanos que siempre estuvieron a mi lado en las situaciones fáciles y difíciles.

A mis profesores quienes redujeron la curva de aprendizaje y dieron su tiempo y esfuerzo para compartir su conocimiento con nosotros.

Agradezco al Ing. Wilmer Rivas por las clases brindadas y orientarme en el desarrollo de este trabajo de titulación.

Darwin Esteban Calva Vicente

RESUMEN

Los dispositivos móviles se han vuelto indispensables en la vida de las personas.

La Realidad Aumentada está siendo utilizada en múltiples disciplinas como: comercio electrónico, medicina, industrias, en la educación.

En el área de comercio electrónico, las ventas han aumentado de forma significativa gracias a esta tecnología, cada vez en más sitios web se ve presente la opción de AR.

Integrar AR en una aplicación móvil es un inconveniente por la cantidad de celulares que existen y la diferencia que estos tienen en sus componentes hardware.

Se ha trabajado en mejorar la tecnología de AR basada en web (Web AR) por las ventajas que presenta en comparación con AR basada en aplicaciones, así se evita en gran parte problemas de compatibilidad en diferentes dispositivos y sistemas operativos, y se llega a la mayor cantidad posible de los mismos

Por estas razones surge la idea de implementar Web AR en una aplicación móvil orientada a ecommerce y desarrollada para dispositivos con sistema operativo Android y iOS.

Se probó el funcionamiento de varias herramientas web AR: Vectary, Model Viewer, A-Frame y Zappar, Onirix.

En Vectary y Onirix se importaron modelos 3D. Para las otras herramientas se creó un servidor usando Node.js y Express para alojar los objetos 3D. Luego se permitió que el usuario acceda a estos recursos mediante su dispositivo móvil y experimente la Realidad Aumentada usando la cámara de su teléfono y un navegador web.

Después se comparó las herramientas web AR para identificar la solución más óptima e integrar en la aplicación móvil.

En la comparación, se identificó las principales características que estas presentan, el soporte que estas tienen para ser utilizadas desde diferentes navegadores, la compatibilidad para trabajar con diferentes formatos de objetos

3D, precio por las suscripciones mensuales que algunas de ellas ofrecen, requisitos de instalación para usar las herramientas, entre otras.

Se evaluó la calidad de las herramientas Web AR usando el modelo iso/iec 25010 con el fin de encontrar la mejor propuesta para integrar en la aplicación. Para valorar el cumplimiento de cada característica que el modelo propone, se usó la escala de Likert con puntajes de calificación de 1 a 5.

Vectary con un puntaje de 4,79 sobre 5, cumple mejor con las características que propone este modelo. Esto lo convierte en la mejor propuesta para trabajar junto a aplicaciones móviles que requieren de un servidor de modelos 3D.

Luego se trabajó en la aplicación móvil, la misma que ha sido desarrollada usando React Native y su entorno Expo.

Se importó modelos en Vectary, a continuación, se usó la librería expo-web-browser que permite acceder a un navegador Web desde la aplicación, se creó un componente que recibe como parámetro la url donde se encuentra el objeto 3D, y lo abre en un navegador.

Se creó una opción para permitir al usuario llamar al componente, así se puede acceder e interactuar con Realidad Aumentada usando la cámara de su dispositivo móvil.

Se probó el funcionamiento de Realidad Aumentada desde la aplicación, para asegurarse que la integración de esta tecnología no presente inconvenientes.

Palabras clave: Realidad Aumentada, Web AR, Visor AR, Vectary, Aplicación móvil orientada a ecommerce, ISO/IEC 25010.

ABSTRACT

Mobile devices have become indispensable in people's lives.

Augmented Reality is being used in multiple disciplines such as: electronic commerce, medicine, industries, in education.

In the area of electronic commerce, sales have increased significantly thanks to this technology, more and more websites are seeing the option of AR.

Integrating AR in a mobile application is a drawback due to the number of cell phones that exist and the difference that these have in their hardware components.

Work has been done to improve web-based AR technology (Web AR) due to the advantages it presents compared to application-based AR, thus largely avoiding compatibility problems in different devices and operating systems, and reaching the as many of them as possible

For these reasons, the idea arose to implement Web AR in a mobile application oriented to ecommerce and developed for devices with Android and iOS operating systems.

Several AR web tools were tested: Vectary, Model Viewer, A-Frame and Zappar, Onirix.

In Vectary and Onirix 3D models were imported. For the other tools a server was created using Node.js and Express to host the 3D objects. The user was then allowed to access these resources through their mobile device and experience Augmented Reality using their phone's camera and a web browser.

The AR web tools were then compared to identify the most optimal solution and integrate it into the mobile application.

In the comparison, the main characteristics they present were identified, the support they have to be used from different browsers, the compatibility to work with different formats of 3D objects, price for the monthly subscriptions that some of them offer, installation requirements to use the tools, among others.

The quality of the Web AR tools was evaluated using the iso / iec 25010 model in order to find the best proposal to integrate into the application.

To assess the fulfillment of each characteristic that the model proposes, the Likert scale was used with qualification scores from 1 to 5.

Vectary with a score of 4.79 out of 5, better complies with the characteristics proposed by this model. This makes it the best proposal to work together with mobile applications that require a 3D model server.

Then we worked on the mobile application, the same one that has been developed using React Native and its Expo environment.

Models were imported into Vectary, then the expo-web-browser library was used that allows access to a Web browser from the application, a component was created that receives as a parameter the url where the 3D object is located, and opens it in a browser.

An option was created to allow the user to call the component, so they can access and interact with Augmented Reality using the camera of their mobile device.

The operation of Augmented Reality was tested from the application, to ensure that the integration of this technology does not present problems.

Keywords: Augmented Reality, Web AR, AR Viewer, Vectary, Mobile application oriented to ecommerce, ISO/IEC 25010.

ÍNDICE DE CONTENIDO

DEDICATORIA.....	1
AGRADECIMIENTO.....	2
RESUMEN	3
ABSTRACT	5
INTRODUCCIÓN	12
1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS	
14	
1.1. Ámbito de Aplicación: descripción del contexto y hechos de interés ..	14
1.2. Establecimiento de requerimientos	15
1.3. Justificación de requerimiento a satisfacer.....	16
2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO	18
2.1. Definición del prototipo tecnológico.....	18
2.2. Fundamentación teórica del prototipo.	21
2.2.1. Herramientas Web AR.....	22
2.2.2. Integración AR en un navegador	24
2.2.3. Integración AR en una aplicación Móvil.....	25
2.2.4. Editor de código.....	25
2.3. Metodología	26
2.4. Objetivos del prototipo.....	27
2.4.1. Objetivo General.....	27
2.4.2. Objetivo Específicos	27
2.5. Diseño del prototipo	27
2.5.1. Requisitos.....	27
2.5.2. Funcionamiento de Herramientas Web AR	28
2.5.3. Comparación y Evaluación de las herramientas.....	39

2.5.4. Integración de Vectary en una aplicación móvil.....	43
3. EVALUACIÓN DEL PROTOTIPO	49
3.1. Plan de evaluación	49
3.1.1. Ejecución de la Evaluación.....	49
3.2. Resultado de evaluación.....	54
3.2. Conclusiones.....	56
3.3. Recomendaciones.....	56
4. BIBLIOGRAFÍA	57
5. ANEXOS	62

ÍNDICE DE FIGURAS

Figura 1: Arquitectura general del desarrollo de la aplicación móvil.....	19
Figura 2: Arquitectura del prototipo	19
Figura 3: Arquitectura del funcionamiento de herramientas Web AR	20
Figura 4: Arquitectura de la integración de AR en la aplicación móvil.....	21
Figura 5: Mapa mental acerca de la Fundamentación Teórica del Prototipo ...	21
Figura 6: Metodología	26
Figura 7: Importación de modelos 3D a Vectary	28
Figura 8: Vista de un modelo importado a Vectary.....	29
Figura 9: Opción para publicar un objeto 3D	29
Figura 10: Publicación de un objeto 3D.....	30
Figura 11: Opción de AR Vectary.....	30
Figura 12: Vista de un objeto 3D en una escena real.....	31
Figura 13: Editor de Model Viewer	31
Figura 14: Estructura proyecto Model Viewer.....	32
Figura 15: Vista HTML para mostrar objeto 3D Model Viewer	32
Figura 16: Código en node.js para levantar un servidor.....	33
Figura 17: Vista Modelo 3D usando Model Viewer.....	33
Figura 18: Modelo 3D en una escena real usando Model Viewer	34
Figura 19: Estructura proyecto A-Frame y Zappar	34
Figura 20: Vista HTML A-Frame y Zappar.....	35
Figura 21: Vista HTML A-Frame y Zappar en el editor Glitch.....	35
Figura 22: Compartir url de proyecto A-Frame y Zappar	36
Figura 23: Objeto 3D en una escena real usando A-Frame y Zappar	36
Figura 24: Creación de un proyecto en Onirix	36
Figura 25: Selección de una escena en Onirix	37
Figura 26: Importación de un modelo a Onirix.....	37
Figura 27: Vista de un objeto 3D en Onirix.....	38
Figura 28: Compartir url para acceder a objeto 3D en Onirix	38
Figura 29: Objeto 3D en una escena real usando Onirix.....	39
Figura 30: Precios de Herramientas Web AR.....	41
Figura 31: Agregación de librería Expo-web-browser al proyecto	43

Figura 32: Componente Ar permite el acceso a un navegador Web desde la aplicación	44
Figura 33: Vista producto permite llamar al componente Ar pasando una url ..	45
Figura 34: Vista Producto muestra la opción Ar sólo en productos que tienen un link de acceso al objeto 3D correspondiente	46
Figura 35: Vista de Producto	46
Figura 36: Selección de un navegador para abrir una url	47
Figura 37: Vista de Objeto 3D llamado desde la aplicación	47
Figura 38: Vista de objeto 3D en una escena real	48
Figura 39: Características de calidad según el modelo iso/iec 25010	49

ÍNDICE DE TABLAS

Tabla 1: Comparación de características de herramientas Web AR	39
Tabla 2: Comparación de soporte de herramientas Web AR	40
Tabla 3: Comparación de formatos soportados por las herramientas Web AR	40
Tabla 4: Escala de Likert	49
Tabla 5: Comparación de Adecuación Funcional	50
Tabla 6: Comparación de Eficiencia de desempeño	50
Tabla 7: Comparación de Compatibilidad	51
Tabla 8: Comparación de Usabilidad	51
Tabla 9: Comparación de Fiabilidad	52
Tabla 10: Comparación de Seguridad	52
Tabla 11: Comparación de Mantenibilidad	53
Tabla 12: Comparación de Portabilidad	54
Tabla 13: Resultado de la evaluación	54
Tabla 14: Resultado de la evaluación obteniendo el promedio	55

INTRODUCCIÓN

La Realidad Aumentada (RA) es una tecnología que ha tenido un gran crecimiento en los últimos años y a través de la cual podemos dar respuesta de manera eficaz a los nuevos estilos de aprendizaje requeridos por los alumnos en la sociedad de la información y el conocimiento. [1]

Son un medio de aprendizaje novedoso, divertido, memorable, inclusivo y atractivo, puede complementar la enseñanza y aprendizaje. [2]

Ofrece a los usuarios nuevas formas de interactuar tanto con el mundo físico como con el digital en tiempo real. [3]

La combinación de entornos AR / VR y simulaciones en la educación en ingeniería ha atraído un interés generalizado. [4]

Integrar Realidad Aumentada en una aplicación móvil es un inconveniente por la cantidad de celulares que existen y la diferencia que estos tienen en sus componentes hardware.

En el presente proyecto se tiene como propuesta integrar Realidad Aumentada una aplicación ecommerce destinada para celulares con sistema operativo Android y iOS.

Se utilizará Web AR en vez de AR basada en aplicaciones, para permitir que la aplicación ecommerce sea compatible con más dispositivos móviles y funcione sin problemas en diferentes sistemas operativos.

Se probará diversas herramientas de Web AR, luego se plantea realizar una comparativa entre ellas, evaluarlas usando el modelo de calidad ISO/IEC 25010 y seleccionar la solución más óptima para luego ser integrada en una aplicación ecommerce destinada para celulares con sistema operativo Android y iOS.

La estructura de este documento es la siguiente:

Capítulo 1: Aquí se describe el ámbito de la aplicación, se establecen los requerimientos a satisfacer, y se justifica el uso de Realidad aumentada basada en Web.

Capítulo 2: Se define el prototipo a realizar, la fundamentación teórica del prototipo, se define la metodología a utilizar y se ejecuta según la metodología planteada.

Capítulo 3: Se planifica y ejecuta una evaluación del prototipo, se analiza de los resultados, conclusiones y recomendaciones.

1. CAPÍTULO I. DIAGNÓSTICO DE NECESIDADES Y REQUERIMIENTOS

1.1. Ámbito de Aplicación: descripción del contexto y hechos de interés

Dentro del comercio, la Realidad Aumentada crea una mejor experiencia y valor para los clientes ya que permite visualizar varios ángulos del objeto a comprar. [5]

Esto permite tener mayores ventas promedio, menores costos de ventas, menores retornos de clientes, niveles más altos de venta de artículos relacionados y mejores resultados de soporte al cliente. [6]

Es por ello que se ha planteado integrar Realidad Aumentada en una aplicación ecommerce móvil para permitir al usuario seleccionar un producto, y por medio de la cámara del dispositivo móvil visualizarlo como si fuera un objeto real. Esto le permite tener una mejor percepción del mismo y evitar en gran porcentaje que luego se arrepienta de la compra.

La Realidad Aumentada se ha aplicado a muchas disciplinas que van desde la ingeniería a la ciencia y el turismo a la educación ya que sus oportunidades y beneficios son innumerables. [7]

Con el uso cada vez mayor de dispositivos inteligentes, la tecnología de Realidad Aumentada (AR) se ha generalizado en los dispositivos móviles. [8]

Se usa ampliamente en conducción automatizada, análisis de imágenes médicas, realidad virtual / aumentada, robots de inteligencia artificial y otras áreas. [9]

En la industria, numerosos estudios han demostrado los beneficios de la Realidad Aumentada (AR) para mejorar los procesos de fabricación y control. [10] Tienen un gran potencial cuando se trata de procesos complejos en el campo de las aplicaciones industriales, como la construcción o el mantenimiento en la industria automotriz. [11]

Con su uso se puede acortar el tiempo de implementación de cambios de ingeniería y reducir la probabilidad de errores de planificación costosos. [12]

Se usa ampliamente la Realidad Aumentada para mejorar los procedimientos de trabajo y la toma de decisiones. Los trabajadores pueden recibir instrucciones de reparación a través de gafas de Realidad Aumentada. [13]

AR impulsa el desarrollo de nuevos dispositivos presentando grandes beneficios para el crecimiento de la industria de la carcasa. [14]

Varios automóviles tienen AR integrado en sus sistemas, un ejemplo son las pantallas heads-up GMC que proporcionan al conductor información adicional y aumentan la experiencia de conducción. [15]

Muchas aplicaciones de teléfonos móviles también proporcionan AR. Ejemplo se tiene algunas que indican información sobre varios edificios. [15]

En la medicina, se están utilizando técnicas de neurorehabilitación, como la terapia de realidad virtual (RV) para la recuperación física de pacientes con ictus. [16] [17]

Los médicos pueden acceder a la información más reciente y relevante sobre sus pacientes gracias a la tecnología AR. Además, los pacientes también pueden usar AR para auto educarse y recibir un tratamiento de mayor calidad. [18]

La Realidad Aumentada se utiliza cada vez con mayor frecuencia por medio de teléfonos inteligentes, tabletas y pantallas montadas en la cabeza. En el campo médico se usa a través de la telemedicina y la colaboración remota, y ayudas visuales en los procedimientos. [19]

En este proyecto se tiene como alcance permitir el uso de Realidad Aumentada en una aplicación ecommerce desarrollada para dispositivos iOS y Android.

1.2. Establecimiento de requerimientos

Se puede integrar la Realidad Aumentada en aplicaciones móviles de distintas maneras. Existen algunas herramientas comerciales y otras gratuitas para realizar esta tarea, cada una de ellas tiene sus ventajas y desventajas.

Para este proyecto se siguió la secuencia siguiente.

- Recolección de información, dentro de esta etapa se realizó un estudio sobre diversas herramientas web AR, se encontró 4: Vectary, Zappar que permite trabajar con A-Frame, Model Viewer y Onirix.
- Probar las soluciones, se mostró la interacción con objetos 3D usando herramientas Web AR y un navegador web. Onirix y Vectary ofrecen un servidor donde almacenar los objetos 3D, para el caso de las otras herramientas fue necesario crear uno, para ello se usó Node.js y la librería express.
- Comparar las herramientas, se describió características de cada herramienta. Se buscó información en la página oficial de las mismas.
- Seleccionar la solución más óptima, la herramienta Web AR con mayores ventajas se integra en una aplicación móvil desarrollada para dispositivos móviles que funciona con los sistemas operativos de Apple y Google.
- Verificar que la integración de AR en la aplicación móvil sea correcta, que el usuario interactúe con los objetos 3d por medio de la cámara de un celular compatible con la tecnología ARCORE o ARKit. Se usó un teléfono con sistema operativo Android para esta tarea.

1.3. Justificación de requerimiento a satisfacer

La Realidad Aumentada es una tecnología emergente que combina objetos virtuales tridimensionales con la realidad. [20]

Las presentaciones de productos de prueba virtual basadas en Realidad Aumentada (AR) permiten a los consumidores conocer mejor el producto a comprar, a diferencia de las presentaciones de productos tradicionales basadas en la web. [21]

Por ello se ha agregado esta experiencia en la creación de una aplicación ecommerce destinada a dispositivos con sistema operativo Android y iOS.

Existen algunos conflictos al agregar Realidad Aumentada una aplicación móvil por la cantidad de dispositivos y características de su hardware y software.

Las herramientas Web AR tienen mayor ventaja frente a la Realidad Aumentada orientada a aplicaciones móviles, por la compatibilidad con mayor cantidad de

dispositivos y sistemas operativos. El usuario no necesita descargar programas adicionales para probar Web AR, todos los dispositivos con tecnología Android o IOS tienen instalado al menos un navegador.

En la presente investigación se usará herramientas Web AR para de esta manera mejorar la experiencia del usuario y la aplicación funcione sin ningún inconveniente en celulares con diferentes características.

Luego de implementar esta tecnología en una aplicación móvil orientada a ecommerce, se podrá seleccionar un producto a comprar, usar la cámara del dispositivo, visualizar el producto en 3D junto a un entorno real, e interactuar con el objeto.

La realización del prototipo permite conocer la integración de Realidad Aumentada en una aplicación móvil destinada a dispositivos con sistema operativo Android e IOS.

2. CAPÍTULO II. DESARROLLO DEL PROTOTIPO

2.1. Definición del prototipo tecnológico

La Realidad Aumentada es una escena donde se integra objetos virtuales 3D en un entorno real de forma interactiva. [22] [23]

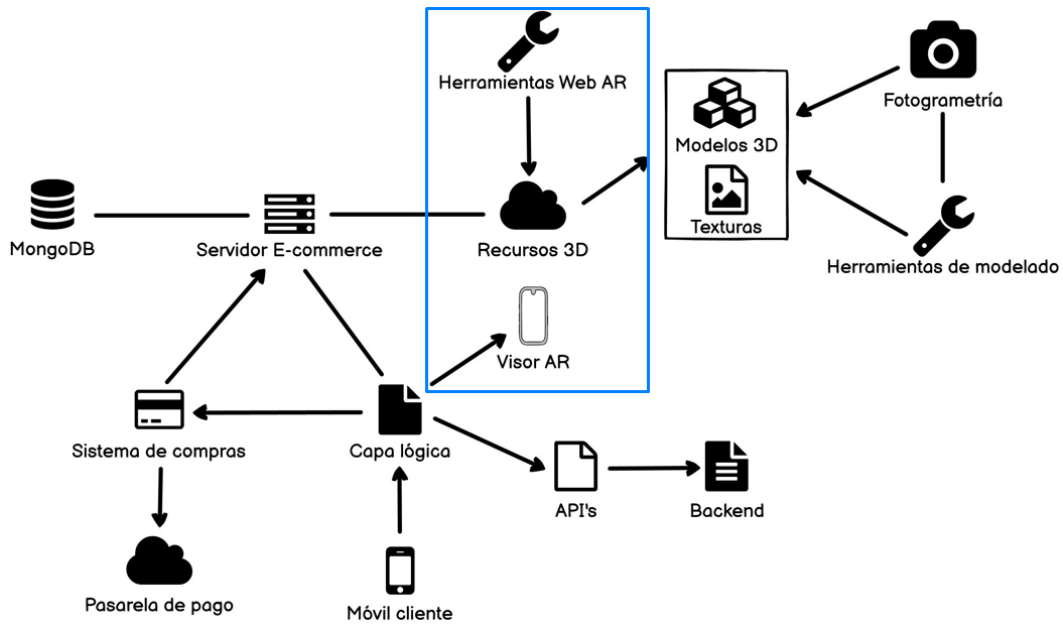
Consiste en utilizar un conjunto de dispositivos tecnológicos que crean una nueva realidad, ya que añaden información virtual a la información física, donde tanto la información real como la virtual desempeñan un papel significativo. [1]

Utiliza ampliamente una variedad de medios técnicos como multimedia, modelado 3D, seguimiento y registro en tiempo real, interacción inteligente y detección para simular la información virtual, como texto generado por computadora, imágenes, modelos 3D, música y video, y luego los aplica al mundo real. [24]

Dentro de una escena visual natural, se puede resaltar un objeto (o su dirección u orientación). El usuario puede recibir información complementaria sobre un objeto o área de interés. [25]

Este trabajo de investigación forma parte de un proyecto integrador el mismo que es el desarrollo de una aplicación móvil orientada a ecommerce aplicando Realidad Aumentada. El aporte de esta investigación es la integración de Realidad Aumentada. Para este proyecto se trabaja con herramientas Web AR, Visor AR y Recursos 3D.

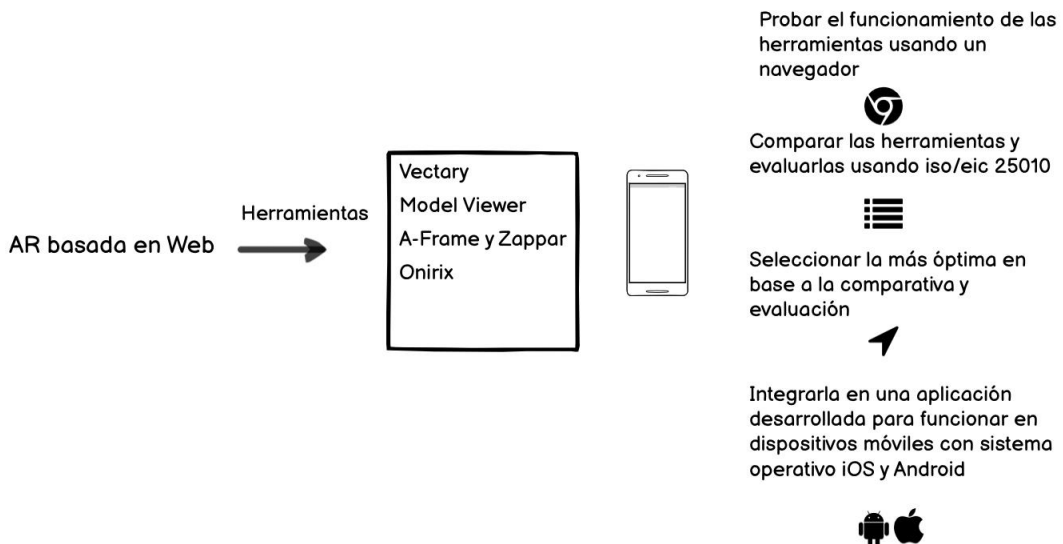
Figura 1: Arquitectura general del desarrollo de la aplicación móvil



Fuente: Elaboración Propia

Se requiere encontrar una herramienta Web AR que funcione de manera óptima y no muestre inconvenientes para trabajar con dispositivos que tienen diferentes hardware y software.

Figura 2: Arquitectura del prototipo

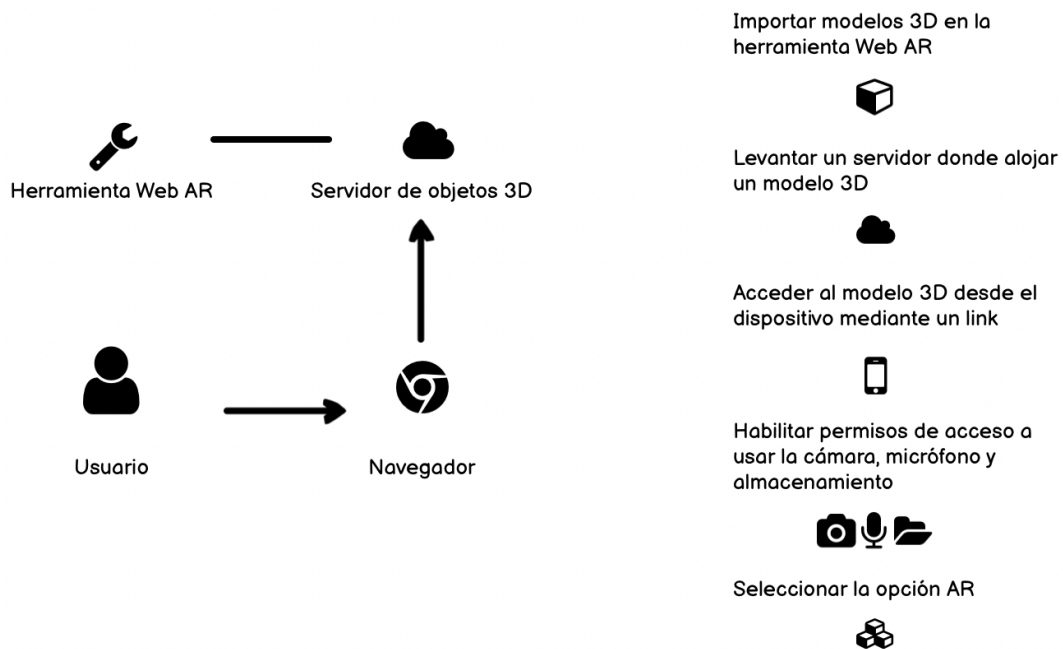


Fuente: Elaboración Propia

Para probar el funcionamiento de las herramientas Web AR con algunas herramientas, es necesario levantar un servidor donde se alojan los modelos 3D. Con todas las herramientas Web AR a utilizar en este trabajo, se prepara el

entorno para que el usuario pueda acceder a objetos de Realidad Aumentada desde un navegador web, en su dispositivo móvil.

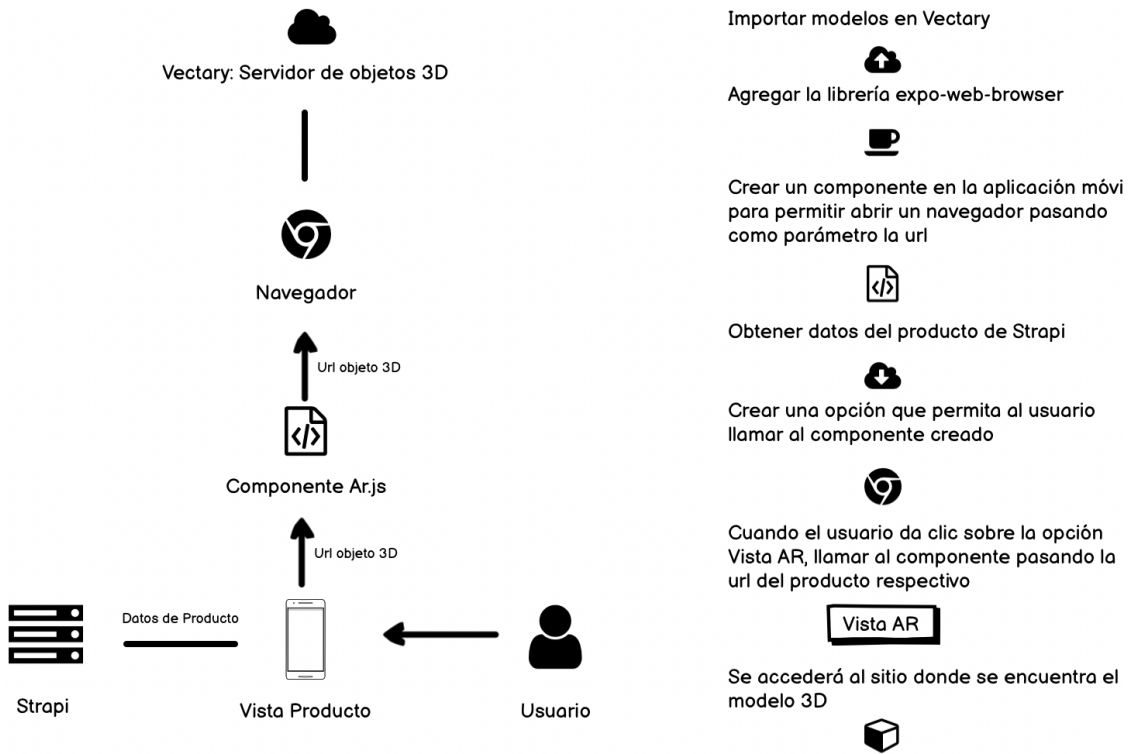
Figura 3: Arquitectura del funcionamiento de herramientas Web AR



Fuente: Elaboración Propia

En el proceso de integrar AR en la aplicación, se añaden componentes para permitir al usuario tener acceso a los objetos 3D e interactuar con ellos por medio de un navegador, esto se realiza de la siguiente manera:

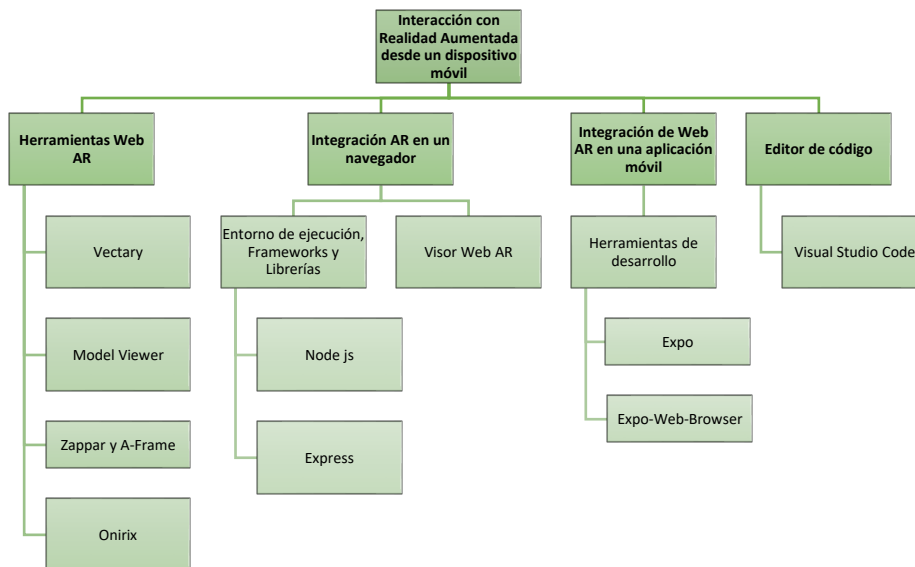
Figura 4: Arquitectura de la integración de AR en la aplicación móvil



Fuente: Elaboración Propia

2.2. Fundamentación teórica del prototipo.

Figura 5: Mapa mental acerca de la Fundamentación Teórica del Prototipo



Fuente: Elaboración Propia

2.2.1. Herramientas Web AR

2.2.1.1. Vectary

El primer CMS (Sistema de gestión de contenidos) de Realidad Aumentada y 3D para la web. Vectary es una aplicación versátil, potente, maneja la mayoría de cosas que los diseñadores necesitan para crear contenido 3D visualmente atractivo. [26]

Según [27] ,la creación de contenido 3D animado personalizado es un tema importante en entornos de RV / RA basados en la web que involucran a usuarios con diversos requisitos para el contenido, por ejemplo, en educación, capacitación, turismo, patrimonio cultural y comercio electrónico.

Vectary permite crear e importar objetos 3D en varios formatos: fxb, glTf, obj, stl.

Ofrece un visor web 3D para cada objeto, este visor se puede usar en un sitio web.

Se puede usar modelos siguientes los pasos que se detallan a continuación:

- Importar o crear un objeto 3D.
- Generar un link para acceder al modelo.
- Acceder al link desde un dispositivo móvil.
- El visor muestra la opción de AR, donde se puede visualizar el modelo.

2.2.1.2. Model Viewer

Muestra fácilmente modelos 3D interactivos en la web y en Realidad Aumentada, los modelos que se pueden usar deben ser en formato GLB. [28]

Esta herramienta es desarrollada por Google. Posee un editor de modelos, donde el usuario puede subir sus modelos 3D en formato GLB.

Este servicio se puede probar de la siguiente manera:

- Seleccionar o importar un modelo en formato GLB.
- Realizar los ajustes necesarios al modelo, como intensidad de luz, sombra, dar color al modelo entre otros.
- La aplicación muestra la opción de ver el modelo desde un dispositivo móvil, para ello genera un código QR.

- En el dispositivo móvil se lee el código QR y se accede a la URL que se muestra.
- Se muestra el modelo y la opción de AR para visualizarlo usando la cámara del dispositivo.

2.2.1.3. Zappar

Es una plataforma que permite a usuarios y empresas desarrollar AR y Web AR.

Ofrece servicios de capacitación, que permiten mejorar el diseño de objetos 3D y desarrollo de la Realidad Aumentada. [29]

2.2.1.4. A-Frame

Es un marco web basado en HTML, permite crear experiencias de realidad virtual de forma fácil. Utiliza etiquetas entidad-componente, su estructura es declarativa, extensible y componible a three.js. [30]

Para trabajar junto con Zappar se puede seguir la secuencia siguiente:

Importar las librerías Zappar y A-Frame.

- Crear una escena de a-frame y pedir permisos de acceso a la cámara
- Dentro de la escena importar el modelo 3D
- Levantar un servidor con protocolo https, para alojar el sitio web
- Acceder al URL del sitio web desde un dispositivo móvil
- Dar permisos para usar la cámara del dispositivo
- El usuario ya puede interactuar con el objeto 3D

2.2.1.5. Onirix

Aplicación web que permite crear Realidad Aumentada sin la necesidad de programar, ofrece la posibilidad de crear escenas de RA sofisticadas y poder visualizarlas en dispositivos iOS y Android de forma sencilla. [31]

Permite acceder a los objetos 3D a través de un enlace o código QR, para que el usuario visualice una escena AR en un entorno real.

Para trabajar con esta herramienta los pasos a seguir son:

- Crear un proyecto
- Crear una escena

- Crear o importar los objetos 3D
- Compartir el proyecto mediante enlace o código QR
- En el dispositivo móvil se accede al link y se otorgan permisos para acceder a la cámara y de esta forma presentar el objeto 3D en una escena real.

2.2.2. Integración AR en un navegador

2.2.2.1. Entorno de ejecución, Frameworks y Librerías

2.2.2.1.1. Node.js

Entorno que permite ejecutar código de JavaScript (El lenguaje de programación más popular en la web [32]) del lado del servidor [33]

Se usa para crear aplicaciones network escalables con de forma sencilla. Puede atender a múltiples conexiones simultáneamente. [34]

Permite a los desarrolladores crear toda clase de herramientas de lado servidor y aplicaciones en JavaScript. [35]

El uso de JavaScript del lado del servidor en combinación con el marco Node.js para implementar aplicaciones web se está volviendo cada vez más común en la práctica. [36]

Los proveedores de navegadores web invierten mucho en optimizar el tiempo de ejecución subyacente. [37]

2.2.2.1.2. Express

Express es el framework web más popular de Node, y es la librería subyacente para un gran número de otros frameworks web de Node populares. [38]

Express permite crear un servidor que aloja las páginas web para apoyar la comunicación en tiempo real entre el back-end y el front-end. [39]

2.2.2.2 Visor Web AR

Vista que permite interactuar con objetos 3D alojados en un servidor web, aprovecha los desarrollos recientes en la tecnología web, a saber, la adopción de WebGL (Web Graphics Library) por parte de los navegadores web actuales.

[40]

Debido a que, los dispositivos móviles son una interfaz importante para la interacción humana con la web [41], desde estos se puede acceder a un visor de este tipo y aprovechar la tecnología de AR en la web.

2.2.3. Integración AR en una aplicación Móvil

2.2.3.1. Herramientas de desarrollo

2.2.3.1.1. Expo

Es un conjunto de herramientas y servicios que ayudan a desarrollar, construir, implementar e iterar rápidamente en iOS, Android y aplicaciones web usando React Native y plataformas nativas. [42]

2.2.3.1.2. Expo-Web-Browser

Proporciona acceso al navegador web del sistema y admite el manejo de redireccionamientos. En iOS, usa `SFSafariViewController` o `SFAuthenticationSession`, según el método al que llame, y en Android usa `ChromeCustomTabs`. [43]

2.2.4. Editor de código

2.2.4.1. Visual Studio Code

Es un editor de código fuente ejecutable como programa de escritorio, se encuentra disponible para Windows, macOS y Linux. Tiene soporte para desarrollar JavaScript, TypeScript y Node.js además se pueden descargar extensiones para otros lenguajes como C++, C#, Java, Python, PHP, Go. [44]

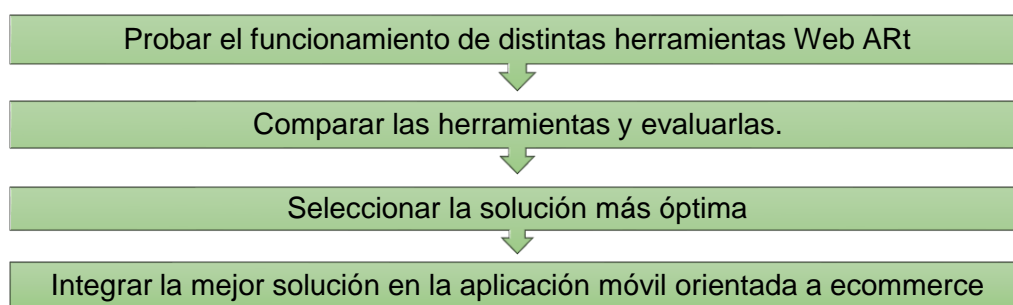
2.3. Metodología

Luego de realizar un estudio exploratorio se encontró 4 herramientas Web AR. En el presente proyecto se tiene como objetivo probar varias herramientas Web AR, realizar una comparativa entre ellas, usar el modelo de calidad iso/iec 25010 para evaluarlas y seleccionar la solución más óptima para luego ser integrada en una aplicación ecommerce destinada para celulares con sistema operativo Android y iOS.

La secuencia de pasos a seguir fué la siguiente:

- Se probó el funcionamiento de las herramientas web AR desde un dispositivo móvil compatible con la tecnología ARCORE, por medio de un navegador web.
- Se comparó las distintas herramientas, en base a las características de las mismas, soporte para ser utilizadas en diferentes navegadores, requisitos de instalación, compatibilidad para trabajar con diferentes formatos de modelos 3D y suscripciones que ofrecen.
- Se evaluó las herramientas usando el modelo de calidad iso/iec 25010
- Se selecciona la herramienta más óptima, y es implementada en una aplicación móvil orientada a ecommerce.

Figura 6: Metodología



Fuente: Elaboración Propia

2.4. Objetivos del prototipo

2.4.1. Objetivo General

Comparar diversas herramientas de Web AR, mediante el análisis de sus principales características y el uso del modelo de calidad iso/iec 25010 para la identificación de la solución más óptima e integración de la misma en una aplicación móvil orientada a ecommerce.

2.4.2. Objetivo Específicos

- Investigar en artículos científicos y fuentes confiables, información sobre el funcionamiento de herramientas Web AR.
- Realizar una comparativa de las herramientas Web AR.
- Seleccionar la herramienta más óptima e integrar en una aplicación móvil.
- Evaluar las herramientas usando el modelo de calidad iso/iec 25010.

2.5. Diseño del prototipo

El prototipo está formado por dos partes:

En la primera parte se prueba el funcionamiento de herramientas Web AR para permitir que el usuario experimente Realidad Aumentada desde su celular por medio de un navegador web.

Luego se comparan las herramientas y la opción más óptima es implementada en una aplicación móvil orientada a ecommerce.

2.5.1. Requisitos

Los recursos a utilizar se detallan a continuación:

Características del Hardware:

- Computador:
 - Marca: MacBook Pro
 - Procesador: Apple M1
 - Memoria: 8 RAM
 - Sistema Operativo: MacOS Big Sur
- Teléfono
 - Marca: Samsung
 - Nombre del modelo: Galaxy A30

- Versión de Android: 11

Programas, librerías y entornos de ejecución:

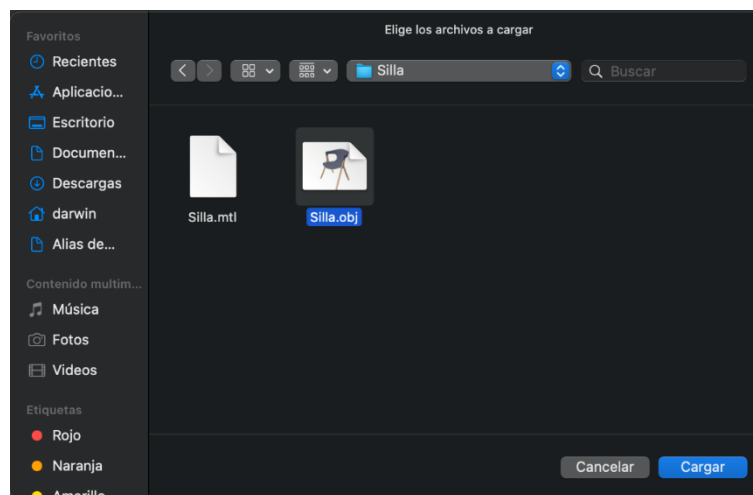
- React Native
- React Native Expo
- Expo-Web-Browser
- Vectary
- Model Viewer
- A-Frame y Zappar
- Onirix
- Node.js
- Express
- Visual Code Studio
- Navegador Web

2.5.2. Funcionamiento de Herramientas Web AR

2.5.2.1. Vectary

Vectary ofrece varias herramientas que permiten la creación de objetos 3D. Este software permite importar archivos en los siguientes formatos: fbx, glTf, obj, stl, dae, hdr, jpg, png, svg.

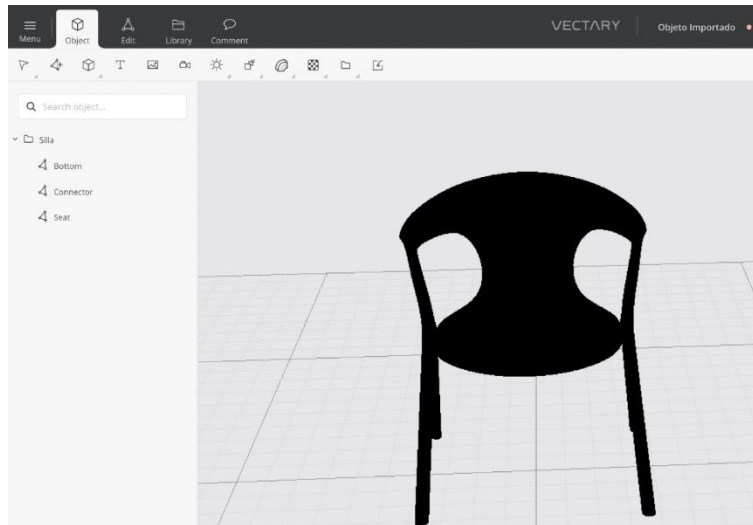
Figura 7: Importación de modelos 3D a Vectary



Fuente: Elaboración Propia

En este ejemplo se importó un objeto en formato obj.

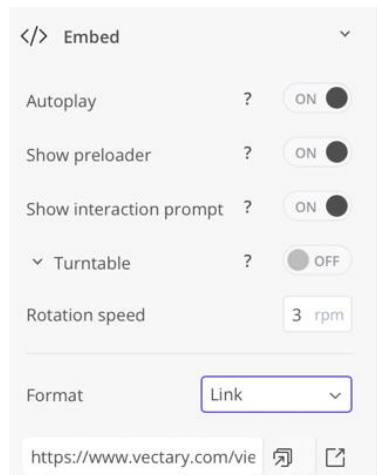
Figura 8: Vista de un modelo importado a Vectary



Fuente: Elaboración Propia

Luego de importar el objeto se puede modificar diversos parámetros del mismo y agregar contenido adicional al como texto, imágenes, botones entre otros. Vectary permite tener acceso a los modelos mediante un link o colocar en sitios web por medio de la etiqueta HTML iframe.

Figura 9: Opción para publicar un objeto 3D



Fuente: Elaboración Propia

Si se accede al link desde un computador, muestra la opción AR.

Figura 10: Publicación de un objeto 3D



Fuente: Elaboración Propia

Al dar clic sobre esta opción aparece un código QR que debe ser escaneado desde el dispositivo móvil.

Figura 11: Opción de AR Vectary



Fuente: Elaboración Propia

Luego de escáner el código QR se puede ver el objeto 3D en un lugar real, se puede interactuar con el objeto.

Figura 12: Vista de un objeto 3D en una escena real



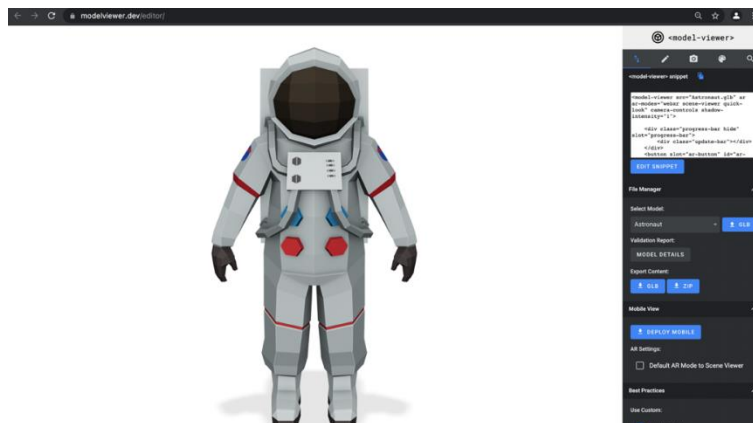
Fuente: Elaboración Propia

2.5.2.2. Model Viewer

El editor de Model Viewer permite la importación y exportación de modelos sólo en formato GLB. Las opciones para editar modelos son muy limitadas como intensidad de sombra, colocar etiquetas en puntos específicos del modelo, dar colores al modelo, entre otros.

Se usó el editor de model viewer y se seleccionó un modelo, luego se descargó el mismo en formato GLB.

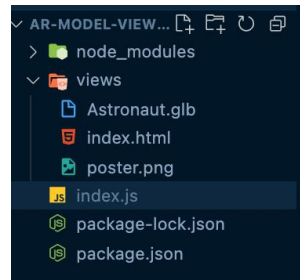
Figura 13: Editor de Model Viewer



Fuente: Elaboración Propia

Para poder visualizar el modelo en un sitio web se creó un proyecto con la siguiente estructura.

Figura 14: Estructura proyecto Model Viewer



Fuente: Elaboración Propia

El archivo index.html es la vista donde se mostrará el modelo 3D.

Figura 15: Vista HTML para mostrar objeto 3D Model Viewer

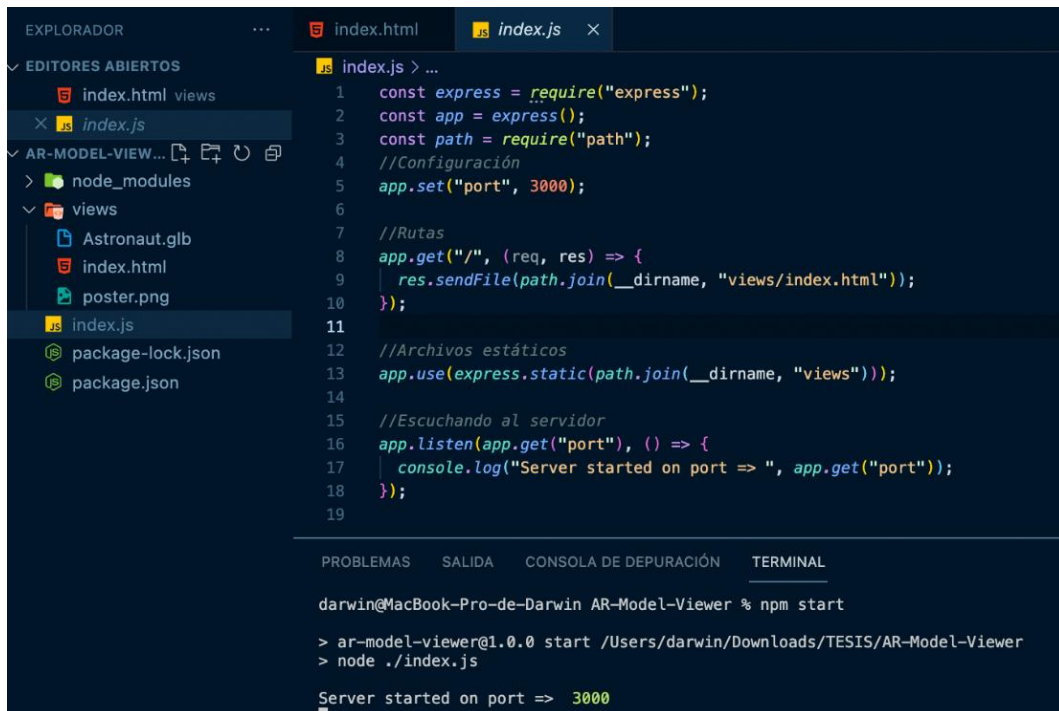


Fuente: Elaboración Propia

Una imagen, esta se mostrará antes de que el modelo 3D sea cargado por completo.

El archivo index.js contiene el código necesario para levantar un servidor usando Node.js y Express.

Figura 16: Código en node.js para levantar un servidor



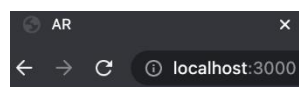
```
EXPLORADOR
... index.html index.js x
EDITORES ABIERTOS
index.html views
index.js
AR-MODEL-VIEW...
node_modules
views
Astronaut.glb
index.html
poster.png
index.js
package-lock.json
package.json
index.js > ...
1 const express = require("express");
2 const app = express();
3 const path = require("path");
4 //Configuración
5 app.set("port", 3000);
6
7 //Rutas
8 app.get("/", (req, res) => {
9   | res.sendFile(path.join(__dirname, "views/index.html"));
10  });
11
12 //Archivos estáticos
13 app.use(express.static(path.join(__dirname, "views")));
14
15 //Escuchando al servidor
16 app.listen(app.get("port"), () => {
17   | console.log("Server started on port => ", app.get("port"));
18  });
19
PROBLEMAS SALIDA CONSOLA DE DEPURACIÓN TERMINAL
darwin@MacBook-Pro-de-Darwin AR-Model-Viewer % npm start
> ar-model-viewer@1.0.0 start /Users/darwin/Downloads/TESIS/AR-Model-Viewer
> node ./index.js
Server started on port => 3000
```

Fuente: Elaboración Propia

Al acceder al sitio web se puede visualizar el modelo en 3D.

Para probar el funcionamiento de AR, es necesario acceder al sitio web desde un dispositivo compatible con ARCORE.

Figura 17: Vista Modelo 3D usando Model Viewer



Fuente: Elaboración Propia

En el dispositivo móvil al dar clic sobre el botón que aparece junto al modelo, se activa la cámara del dispositivo y se puede ver el modelo en una escena real.

Figura 18: Modelo 3D en una escena real usando Model Viewer



Fuente: Elaboración Propia

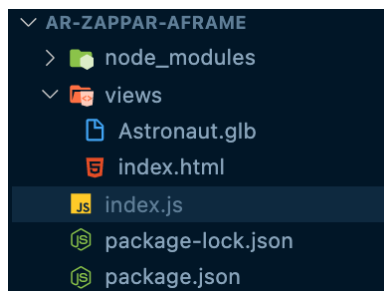
2.5.2.3. A-Frame y Zappar

A-Frame es un framework del lenguaje JavaScript, permite desarrollar experiencias en realidad virtual.

Zappar es una empresa que permite trabajar con Web AR de forma simple.

Para trabajar con Realidad Aumentada usando estas dos tecnologías se creó un proyecto que tiene la siguiente estructura:

Figura 19: Estructura proyecto A-Frame y Zappar



Fuente: Elaboración Propia

El archivo index.js almacena el código que permite levantar un servidor donde se mostrará la página web para interactuar con Realidad Aumentada.

En el archivo index.html se importan las librerías de a-frame y zappar y se trabaja con los objetos necesarios que permiten mostrar experiencia de Realidad Aumentada.

Figura 20: Vista HTML A-Frame y Zappar

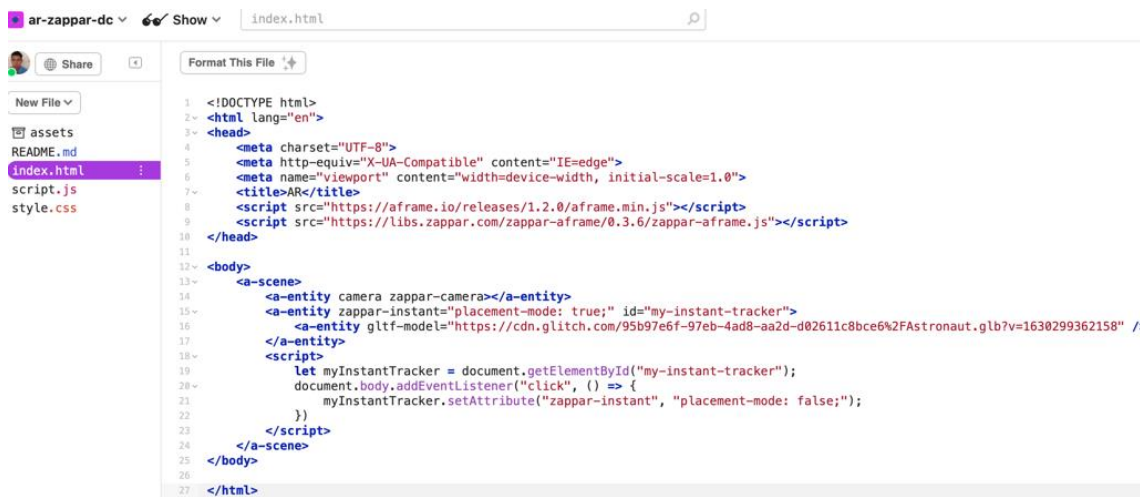
```
views > index.html > ...
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>AR</title>
8   <script src="https://aframe.io/releases/1.2.0/aframe.min.js"></script>
9   <script src="https://libs.zappar.com/zappar-aframe/0.3.6/zappar-aframe.js"></script>
10 </head>
11 <body>
12   <a-scene>
13     <a-entity camera zappar-camera></a-entity>
14     <a-entity zappar-instant="placement-mode: true;" id="my-instant-tracker">
15       <a-entity gltf-model=".Astronaut.glb" />
16     </a-entity>
17     <script>
18       let myInstantTracker = document.getElementById("my-instant-tracker");
19       document.body.addEventListener("click", () => {
20         myInstantTracker.setAttribute("zappar-instant", "placement-mode: false;");
21       });
22     </script>
23   </a-scene>
24 </body>
```

Fuente: Elaboración Propia

Si se ejecuta el programa el navegador impide el acceso a la cámara del dispositivo móvil, es necesario que el servidor donde se aloja el proyecto tenga el protocolo https.

Para probar el código se usó glitch que se trata de un sitio web que permite escribir código JavaScript y alojarlo en un servidor con protocolo https.

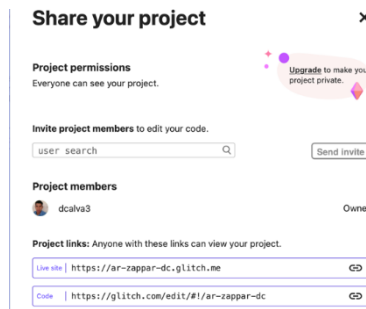
Figura 21: Vista HTML A-Frame y Zappar en el editor Glitch



```
ar-zappar-dc Show index.html
Share
New File
assets
README.md
index.html
script.js
style.css
Format This File
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7   <title>AR</title>
8   <script src="https://aframe.io/releases/1.2.0/aframe.min.js"></script>
9   <script src="https://libs.zappar.com/zappar-aframe/0.3.6/zappar-aframe.js"></script>
10 </head>
11 <body>
12   <a-scene>
13     <a-entity camera zappar-camera></a-entity>
14     <a-entity zappar-instant="placement-mode: true;" id="my-instant-tracker">
15       <a-entity gltf-model="https://cdn.glitch.com/95b97e6f-97eb-4ad8-aa2d-d02611c8bce6%2FAstronaut.glb?v=1630299362158" />
16     </a-entity>
17     <script>
18       let myInstantTracker = document.getElementById("my-instant-tracker");
19       document.body.addEventListener("click", () => {
20         myInstantTracker.setAttribute("zappar-instant", "placement-mode: false;");
21       });
22     </script>
23   </a-scene>
24 </body>
25 </html>
```

Fuente: Elaboración Propia

Figura 22: Compartir url de proyecto A-Frame y Zappar



Fuente: Elaboración Propia

El resultado obtenido es el siguiente:

Figura 23: Objeto 3D en una escena real usando A-Frame y Zappar

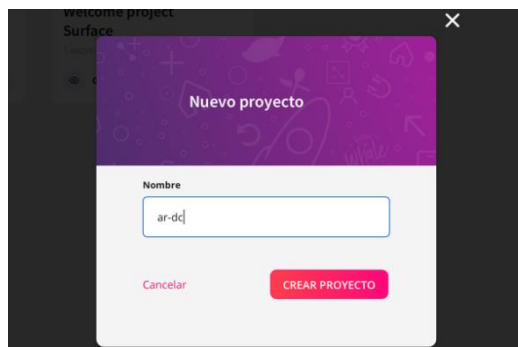


Fuente: Elaboración Propia

2.5.2.4. Onirix

Para trabajar con este programa, se crea un proyecto y se le da un nombre.

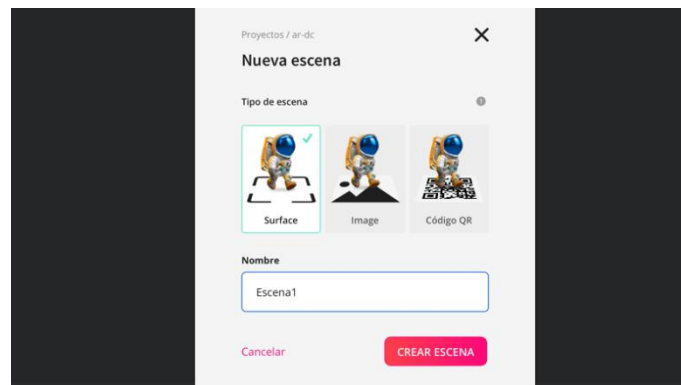
Figura 24: Creación de un proyecto en Onirix



Fuente: Elaboración Propia

Luego se selecciona el tipo de escena, este programa muestra tres opciones, la primera permite poner el objeto 3D en una superficie, la segunda es para usar una imagen como marcador, y la tercera es para usar código QR donde se quiere ver el objeto 3D.

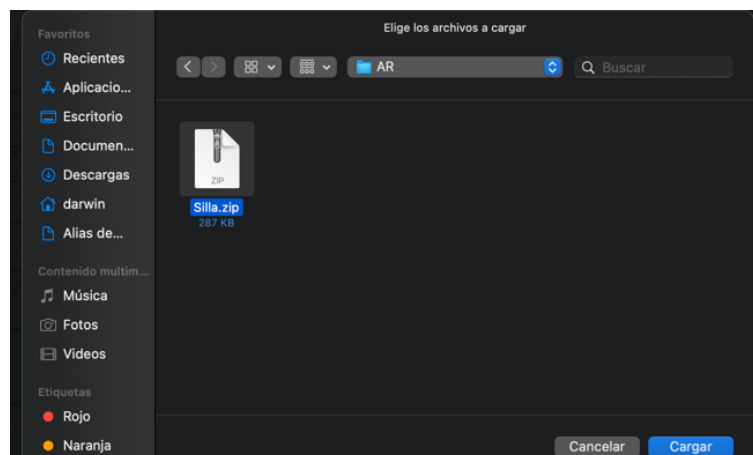
Figura 25: Selección de una escena en Onirix



Fuente: Elaboración Propia

Después de crear la escena aparece el espacio de trabajo, se puede crear objetos o importarlos.

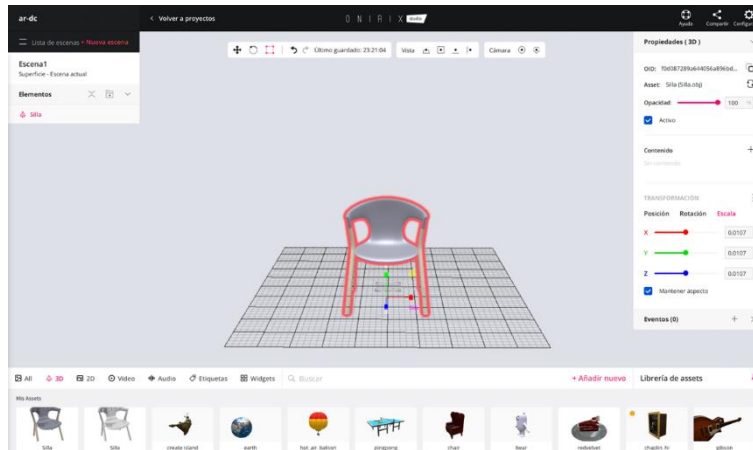
Figura 26: Importación de un modelo a Onirix



Fuente: Elaboración Propia

Luego de importar los objetos, estos se agregan a la escena y se realizan las modificaciones que se requiera o se agregan más objetos.

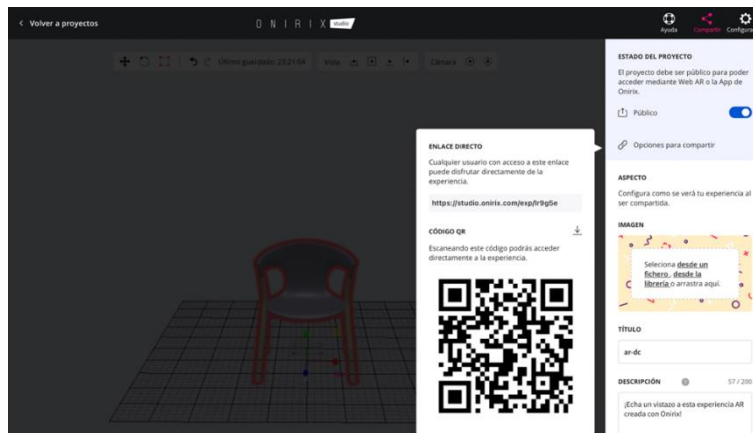
Figura 27: Vista de un objeto 3D en Onirix



Fuente: Elaboración Propia

Para poder visualizar el objeto 3D en un entorno real, se comparte el proyecto, aparece una opción llamada estado de proyecto donde se debe marcar como público, luego el programa provee un enlace directo o código QR para probar desde un dispositivo móvil.

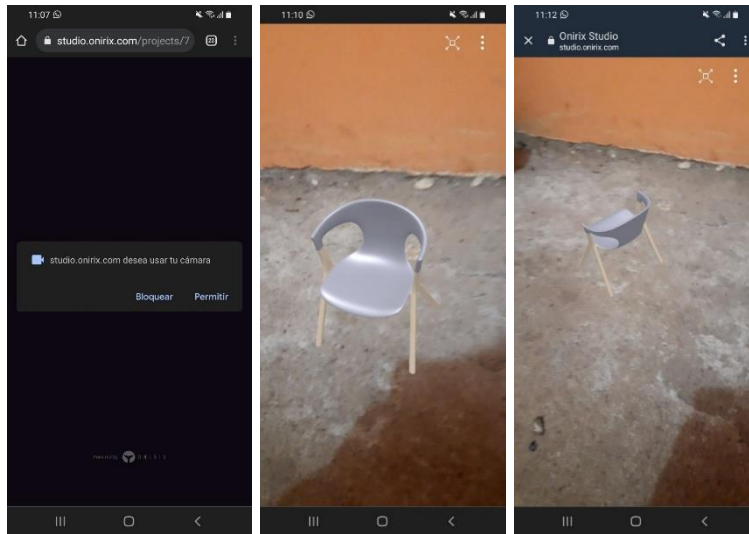
Figura 28: Compartir url para acceder a objeto 3D en Onirix



Fuente: Elaboración Propia

Al acceder al link desde el dispositivo, se permite el acceso a la cámara y se puede interactuar con los objetos 3D.

Figura 29: Objeto 3D en una escena real usando Onirix



Fuente: Elaboración Propia

2.5.3. Comparación y Evaluación de las herramientas

La siguiente tabla compara las características de cada herramienta.

Tabla 1: Comparación de características de herramientas Web AR

Item	Vectary	Model Viewer	A-Frame y Zappar	Onirix
El servicio ofrece un servidor en la nube para almacenar modelos en 3D	X			X
Permite incrustar modelos en sitios web	X	X	X	X
Permite la edición de modelos 3D	X	X		X
Se puede descargar los modelos 3D para ser usados por otro software	X	X		X
La herramienta es sencilla de utilizar	X	X	X	X

Fuente: Elaboración Propia

Según la tabla anterior, las herramientas Vectary y Onirix, presentan más características que las demás.

La siguiente tabla muestra el soporte que tiene cada herramienta, para ser accedida por los navegadores más comunes. Onirix no especifica su compatibilidad con navegadores.

Tabla 2: Comparación de soporte de herramientas Web AR

Soporte	Vectary	Model Viewer	A-Frame y Zappar
Firefox	X		X
Safari	X		X
Google Chrome	X	X	X
Microsoft Edge		X	X

Fuente: Elaboración Propia

En este apartado, A-Frame y Zappar pueden ser utilizados por los navegadores más comunes.

Los formatos de objeto 3D que soporta cada herramienta, se muestran a continuación.

Tabla 3: Comparación de formatos soportados por las herramientas Web AR

Formatos soportados	Vectary	Model Viewer	A-Frame y Zappar	Onirix
dae	X			X
fbx	X			X
glTF	X	X	X	X
glb		X		
obj	X		X	X
stl	X			
usdz	X	X		

Fuente: Elaboración Propia

Vectary es compatible con la mayoría de formatos 3D, seguido por la herramienta Onirix.

2.5.3.1. Precios

Model Viewer, es una herramienta gratuita.

Zappar ofrece una suscripción de 65 dólares mensuales para poder utilizar su plataforma de diseño, almacenar los modelos en sus servidores y trabajar con

un SDK de la empresa ZappWorks, si el usuario no requiere de estas características no necesita suscripción.

Onirix ofrece una prueba gratuita de su plataforma por 15 días, pasado ese tiempo el usuario deberá adquirir una suscripción.

La suscripción más económica es de 99 euros por mes.

Vectary ofrece una prueba gratuita por tiempo ilimitado y su plan más económico es 12 dólares por mes.

En la prueba gratuita permite usar el editor 3D sin restricciones y almacenar 25 proyectos en su servidor, el único inconveniente es que para visualizar el objeto 3D usando AR, se requiere una suscripción.

Figura 30: Precios de Herramientas Web AR

The screenshot shows the Vectary website's pricing page. At the top, there is a navigation menu with the following items: Caracte, VECTARY, Estudio, WebAR, Empresa, Precios, Aprender, and Comunidad. The main content area features three pricing cards:

- PRO:** Priced at 99 € / mes. It is described as being for those who want to start creating WebAR experiences and showing demos to clients. Features include: unlimited RA projects, access to web Studio (CMS), surface and QR tracking, geolocated maps, Web AR SDK, 500 RA visualizations per month, analysis panel, online tutorials and documentation, included hosting, and 1 member.
- Gratis:** Priced at \$ 0. It is a browser-based content creation tool. Features include: 3D editor with all Vectary Studio functions, 25 projects, WebAR preview, 3D export (OBJ, STL), free 3D asset and icon library, community support, Figma and Sketch plugins, and photorealistic representation.
- Prima:** Priced at \$ 12 per editor / mes (billed monthly). It is for designers and small businesses. Features include: 3D editor with all Vectary Studio functions, unlimited projects, WebAR viewer (web and AR), 3D export (USDZ, GLTF, GLB, FBX), photorealistic representation, Figma and Sketch plugins, free 3D asset and icon library, public templates, team sharing, and email support.

At the bottom of the pricing section, there is a note: "Ahorre \$ 45 / año".

Fuente: Elaboración Propia

2.5.3.2. Requisitos de instalación para probar las herramientas desde un navegador web

Vectary, No es necesario instalar debido a que es un programa que se ejecuta en el navegador, sólo se necesita un navegador habilitado para WebGL y una conexión a internet para usarlo.

Model Viewer, Es necesario levantar un servidor para alojar los modelos y poderlos utilizarlos, el código fuente de Model Viewer se inserta en archivos HTML.

A-Frame y Zappar, En este aspecto es similar a Model Viewer, se necesita un servidor donde alojar los modelos, además el sitio debe contar con el protocolo https para permitir el acceso a la cámara del dispositivo móvil, para usar Realidad Aumentada con esta herramienta, se usan etiquetas de marcado en archivos HTML.

Onirix, Es similar a Vectary, no es necesaria realizar ninguna instalación.

Las herramientas Onirix y Vectary presentan una forma más sencilla de ser probadas desde un navegador Web.

Luego de realizar la comparativa de herramientas, se seleccionó Vectary porque permite trabajar con su diseñador 3D sin pagar alguna suscripción, almacena los modelos 3D en su servidor para ser accedido desde diferentes dispositivos, tiene una mayor compatibilidad para trabajar con diferentes formatos de objetos 3D, es soportado por los navegadores más comunes y no requiere ninguna instalación para ser utilizado.

2.5.4. Integración de Vectary en una aplicación móvil

La aplicación móvil orientada a ecommerce a quien se le integrará Realidad Aumentada fue desarrollada con el framework React Native para poder ser ejecutada en dispositivos con sistema operativo iOS o Android.

Se usó el entorno Expo para usar herramientas y servicios construidos alrededor de React Native y plataformas nativas que permiten desarrollar, construir, implementar e iterar rápidamente en iOS, Android y aplicaciones web usando base de código JavaScript / TypeScript. [42]

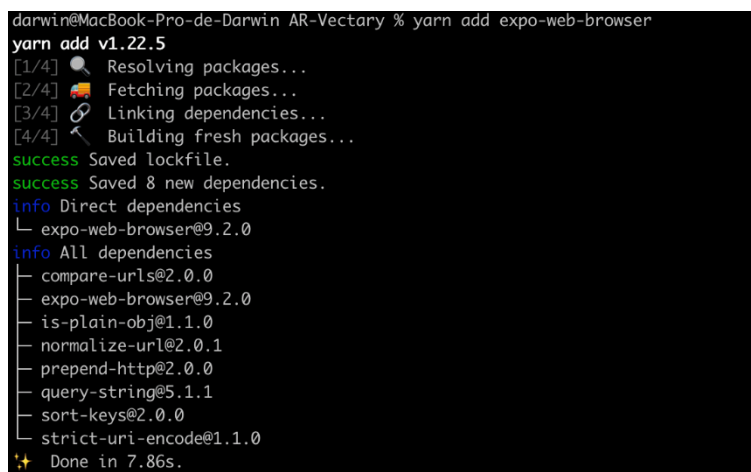
Dentro de la aplicación se implementó una opción que permita acceder al visor web 3D donde se encuentra el modelo de Realidad Aumentada referente al producto que se está ofreciendo en la tienda virtual.

Para trabajar con urls, se descargó la librería y expo-web-browser desde la página oficial de yarn.

Para agregarlas al proyecto se usó el comando:

```
yarn add expo-web-browser
```

Figura 31: Agregación de librería Expo-web-browser al proyecto



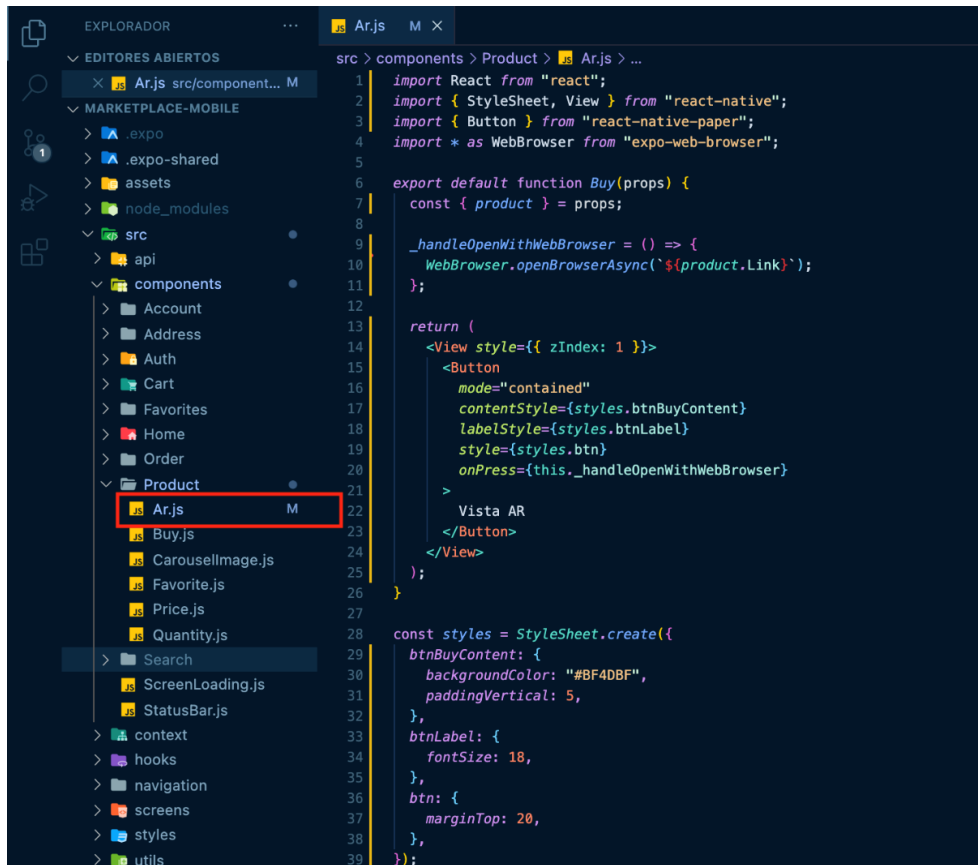
```
darwin@MacBook-Pro-de-Darwin AR-Vectary % yarn add expo-web-browser
yarn add v1.22.5
[1/4] 🔍 Resolving packages...
[2/4] 📦 Fetching packages...
[3/4] 🔗 Linking dependencies...
[4/4] 🔨 Building fresh packages...
success Saved lockfile.
success Saved 8 new dependencies.
info Direct dependencies
└─ expo-web-browser@9.2.0
info All dependencies
├─ compare-urls@2.0.0
├─ expo-web-browser@9.2.0
├─ is-plain-obj@1.1.0
├─ normalize-url@2.0.1
├─ prepend-http@2.0.0
├─ query-string@5.1.1
├─ sort-keys@2.0.0
└─ strict-uri-encode@1.1.0
🌟 Done in 7.86s.
```

Fuente: Elaboración Propia

Luego se creó un componente con el nombre Ar, aquí se importó las dependencias que se descargó en pasos anteriores.

Al ser llamado este componente, en la vista de producto se muestra un botón con el nombre Vista AR, al dar clic sobre el mismo se ejecuta una función que permita abrir un navegador, se le pasa como parámetro la url del visor 3D que provee Vectary.

Figura 32: Componente Ar permite el acceso a un navegador Web desde la aplicación

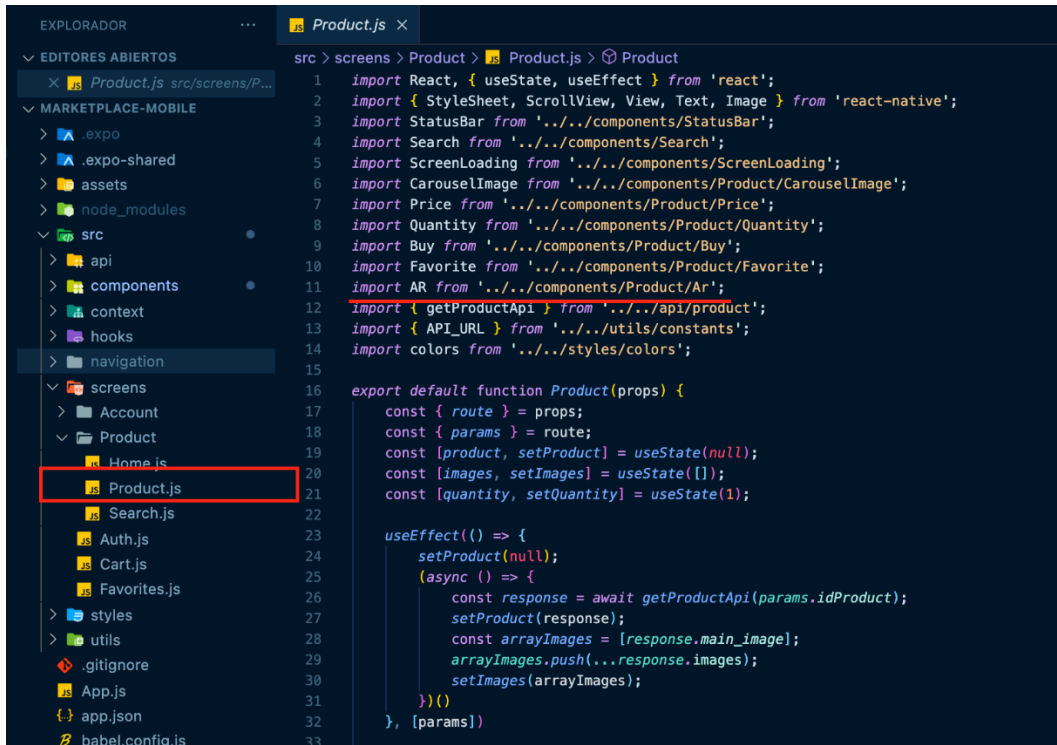


```
1 import React from "react";
2 import { StyleSheet, View } from "react-native";
3 import { Button } from "react-native-paper";
4 import * as WebBrowser from "expo-web-browser";
5
6 export default function Buy(props) {
7   const { product } = props;
8
9   _handleOpenWithWebBrowser = () => {
10     WebBrowser.openBrowserAsync(`${product.Link}`);
11   };
12
13   return (
14     <View style={{ zIndex: 1 }}>
15       <Button
16         mode="contained"
17         contentStyle={styles.btnBuyContent}
18         labelStyle={styles.btnLabel}
19         style={styles.btn}
20         onPress={this._handleOpenWithWebBrowser}
21       >
22         Vista AR
23       </Button>
24     </View>
25   );
26 }
27
28 const styles = StyleSheet.create({
29   btnBuyContent: {
30     backgroundColor: "#BF4DBF",
31     paddingVertical: 5,
32   },
33   btnLabel: {
34     fontSize: 18,
35   },
36   btn: {
37     marginTop: 20,
38   },
39 });
```

Fuente: Elaboración Propia

En la vista que permite mostrar un producto se realiza una importación del componente Ar, antes de mostrar un producto, se realiza una petición al servidor Strapi obteniendo los parámetros del producto como: nombre, imágenes, precio y url del visor web donde se encuentra alojado el objeto 3D correspondiendo al mismo.

Figura 33: Vista producto permite llamar al componente Ar pasando una url

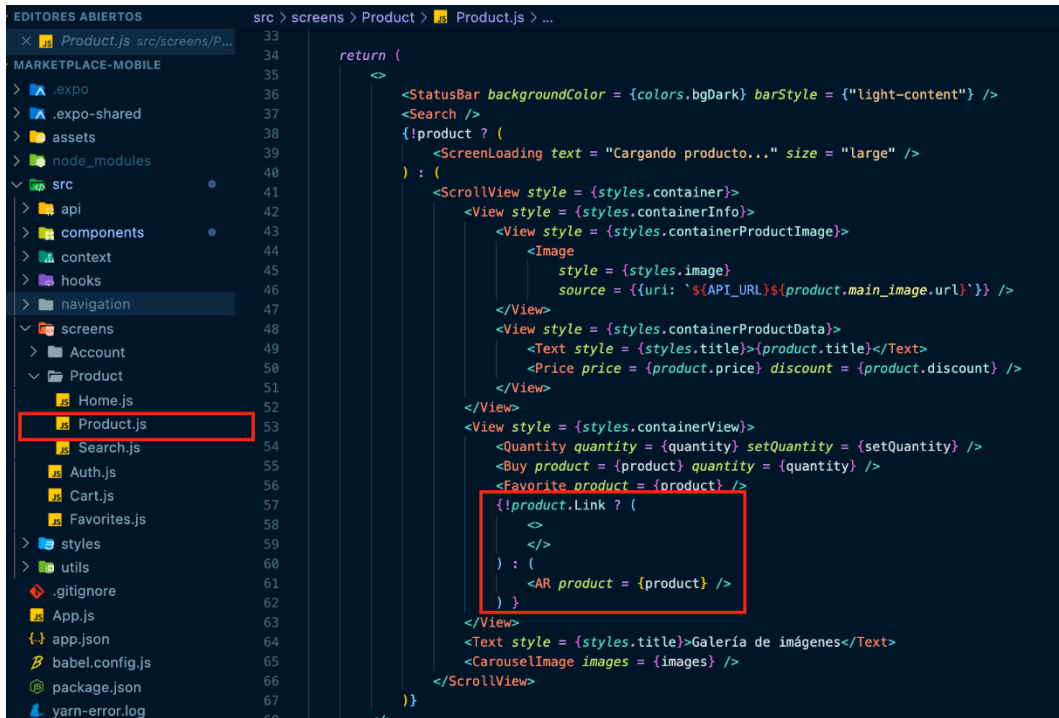


```
EXPLORADOR
... Product.js x
EDITORES ABIERTOS
src > screens > Product > Product.js > Product
Product.js src/screens/P...
MARKETPLACE-MOBILE
.expo
.expo-shared
assets
node_modules
src
  api
  components
  context
  hooks
  navigation
  screens
    Account
    Product
      Home.js
      Product.js
      Search.js
    Auth.js
    Cart.js
    Favorites.js
  styles
  utils
.gitignore
App.js
app.json
babel.config.js
1 import React, { useState, useEffect } from 'react';
2 import { StyleSheet, ScrollView, View, Text, Image } from 'react-native';
3 import StatusBar from '../../components/StatusBar';
4 import Search from '../../components/Search';
5 import ScreenLoading from '../../components/ScreenLoading';
6 import CarouselImage from '../../components/Product/CarouselImage';
7 import Price from '../../components/Product/Price';
8 import Quantity from '../../components/Product/Quantity';
9 import Buy from '../../components/Product/Buy';
10 import Favorite from '../../components/Product/Favorite';
11 import AR from '../../components/Product/Ar';
12 import { getProductApi } from '../../api/product';
13 import { API_URL } from '../../utils/constants';
14 import colors from '../../styles/colors';
15
16 export default function Product(props) {
17   const { route } = props;
18   const { params } = route;
19   const [product, setProduct] = useState(null);
20   const [images, setImages] = useState([]);
21   const [quantity, setQuantity] = useState(1);
22
23   useEffect(() => {
24     setProduct(null);
25     (async () => {
26       const response = await getProductApi(params.idProduct);
27       setProduct(response);
28       const arrayImages = [response.main_image];
29       arrayImages.push(...response.images);
30       setImages(arrayImages);
31     })()
32   }, [params])
33 }
```

Fuente: Elaboración Propia

Sólo, si el producto tiene un link donde se encuentra el objeto 3D correspondiente, se muestra el componente Ar, esto permite que en la aplicación haya productos sin la opción de ser visualizados usando Realidad Aumentada.

Figura 34: Vista Producto muestra la opción Ar sólo en productos que tienen un link de acceso al objeto 3D correspondiente



```
return (  
  <>  
  <StatusBar backgroundColor = {colors.bgDark} barStyle = {"light-content"} />  
  <Search />  
  {!product ? (  
    <ScreenLoading text = "Cargando producto..." size = "large" />  
  ) : (  
    <ScrollView style = {styles.container}>  
      <View style = {styles.containerInfo}>  
        <View style = {styles.containerProductImage}>  
          <Image  
            style = {styles.image}  
            source = {uri: `${API_URL}${product.main_image.url}`} />  
        </View>  
        <View style = {styles.containerProductData}>  
          <Text style = {styles.title}>{product.title}</Text>  
          <Price price = {product.price} discount = {product.discount} />  
        </View>  
      </View>  
      <View style = {styles.containerView}>  
        <Quantity quantity = {quantity} setQuantity = {setQuantity} />  
        <Buy product = {product} quantity = {quantity} />  
        <Favorite product = {product} />  
        {!product.Link ? (  
          <>  
        ) : (  
          <AR product = {product} />  
        ) }  
      </View>  
      <Text style = {styles.title}>Galería de imágenes</Text>  
      <CarouselImage images = {images} />  
    </ScrollView>  
  ) )  
)
```

Fuente: Elaboración Propia

En la aplicación, cuando el usuario seleccione un producto, se verá de la siguiente manera.

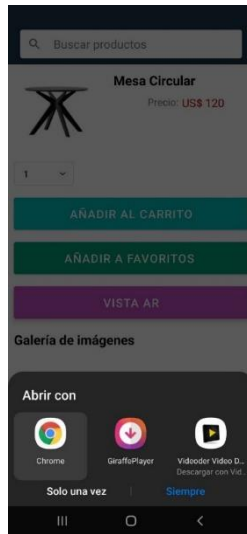
Figura 35: Vista de Producto



Fuente: Elaboración Propia

Al dar clic sobre el botón Vista AR, se muestra la opción de seleccionar el programa con el que se quiere acceder al enlace.

Figura 36: Selección de un navegador para abrir una url



Fuente: Elaboración Propia

El usuario deberá seleccionar un navegador.

En la parte superior derecha se muestra la opción de Realidad Aumentada.

Figura 37: Vista de Objeto 3D llamado desde la aplicación



Fuente: Elaboración Propia

Es necesario otorgar los permisos que solicita el navegador para mostrar el objeto en una escena real, normalmente pide acceso a cámara, micrófono y almacenamiento.

Figura 38: Vista de objeto 3D en una escena real



Fuente: Elaboración Propia

3. EVALUACIÓN DEL PROTOTIPO

3.1. Plan de evaluación

Para evaluar las herramientas, se usó el modelo de calidad iso/iec 25010. En este modelo se determinan las características de calidad que se van a tener en cuenta a la hora de evaluar las propiedades de un producto software determinado. [45]

Este modelo se encuentra compuesto por las ocho características que se muestran en la siguiente figura:

Figura 39: Características de calidad según el modelo iso/iec 25010.



Fuente: [45]

Se usó este modelo con el objetivo de encontrar la herramienta que más ventajas ofrece, para luego integrar en la aplicación orientada al comercio electrónico.

3.1.1. Ejecución de la Evaluación

3.1.1.1. Métricas

Se usó la escala de Likert, para valorar el cumplimiento de cada característica que el modelo propone.

Tabla 4: Escala de Likert

Puntaje	Criterio a Calificar
5	Totalmente de acuerdo
4	De acuerdo
3	Indeciso
2	En desacuerdo
1	Totalmente en desacuerdo

Fuente: Elaboración Propia

Característica: Adecuación Funcional

Tabla 5: Comparación de Adecuación Funcional

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Complejidad	La herramienta cumple con las tareas que están en su documentación	5	5	5	5
Corrección	La herramienta no presenta errores al mostrar los objetos 3D	5	2	2	5
Pertinencia	La herramienta cumple con sus objetivos	5	4	4	5

Fuente: Elaboración Propia

Característica: Eficiencia de desempeño

Tabla 6: Comparación de Eficiencia de desempeño

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Comportamiento temporal	El tiempo de respuesta para mostrar un objeto 3D es óptimo.	5	5	5	5
Utilización de recursos	La herramienta utiliza los recursos de forma adecuada	5	5	5	5
Capacidad	Las herramientas permiten trabajar con modelos 3D en diferentes formatos sin provocar errores	4	2	3	4

Fuente: Elaboración Propia

Característica: Compatibilidad

Tabla 7: Comparación de Compatibilidad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Coexistencia	Los objetos 3D publicados usando la herramienta, pueden ser utilizados desde distintas aplicaciones	5	4	4	5
Interoperabilidad	La aplicación permite intercambiar información con otras sin ningún problema	5	5	5	5

Fuente: Elaboración Propia

Característica: Usabilidad

Tabla 8: Comparación de Usabilidad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Capacidad para reconocer su adecuación	El usuario reconoce con facilidad si la herramienta es adecuada para el	5	5	5	5
Capacidad de aprendizaje	Se puede aprender a usar la herramienta sin complicaciones	5	5	5	5
Capacidad para ser usado	El manejo de la herramienta es fácil	5	4	2	5
Protección contra errores de usuario	La herramienta protege a los usuarios de cometer errores en su manejo	5	2	2	5
Estética de la interfaz de usuario	La interacción con los objetos 3D es agradable para el usuario	5	5	3	4

Accesibilidad	La herramienta puede usarse para usuarios que tienen alguna discapacidad	2	2	2	2
---------------	--	---	---	---	---

Fuente: Elaboración Propia

Característica: Fiabilidad

Tabla 9: Comparación de Fiabilidad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Madurez	Los objetos 3D funcionan correctamente en condiciones normales	5	4	4	5
Disponibilidad	La aplicación no presenta problemas en el funcionamiento cuando el usuario la está utilizando	5	4	4	5
Tolerancia a fallos	Cuando existen fallos de hardware o software el acceso a los objetos, sigue funcionando normalmente	4	4	4	4
Capacidad de recuperación	Si ocurre una interrupción la aplicación recupera los datos afectados	5	5	5	5

Fuente: Elaboración Propia

Característica: Seguridad

Tabla 10: Comparación de Seguridad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Confidencialidad	Se puede impedir el acceso de personal no autorizado	4	1	3	4
Integridad	La aplicación evita que se realicen modificaciones no autorizadas	5	5	5	5

No repudio	Las acciones realizadas en la herramienta, se pueden deshacer en un futuro	5	5	5	5
Responsabilidad	Se puede rastrear las acciones de un usuario	3	3	3	3
Autenticidad	La aplicación permite el acceso sólo a usuarios autenticados	5	1	3	5

Fuente: Elaboración Propia

Característica: Mantenibilidad

Tabla 11: Comparación de Mantenibilidad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Modularidad	Cuando se realiza un cambio en un componente, este no perjudica a los demás	5	5	5	5
Reusabilidad	La herramienta puede ser reutilizada	5	5	5	5
Analizabilidad	Cuando algo falla en la herramienta, se puede identificar fácilmente el causante del fallo	5	5	5	5
Capacidad de ser modificado	La herramienta permite modificar los modelos sin degradar su desempeño	5	5	5	5
Capacidad de ser probado	Se puede probar fácilmente un modelo 3D	5	5	5	5

Fuente: Elaboración Propia

Característica: Portabilidad

Tabla 12: Comparación de Portabilidad

Sub-característica	Objetivo	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Adaptabilidad	Se puede usar la aplicación desde diferentes sistemas operativos	5	5	5	5
Capacidad para ser instalado	Se puede instalar o desinstalar con facilidad la aplicación	5	4	4	5
Capacidad para ser reemplazado	Se puede usar la aplicación para sustituir otra que cumple la misma tarea	5	5	5	5

Fuente: Elaboración Propia

3.2. Resultado de evaluación

Luego la evaluación aplicada a las diferentes herramientas se obtuvo los siguientes resultados:

Tabla 13: Resultado de la evaluación

Característica	Subcaracterística	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Adecuación Funcional	3	15	11	11	15
Eficiencia de desempeño	3	14	12	13	14
Compatibilidad	2	10	9	9	10
Usabilidad	6	27	23	19	26
Fiabilidad	4	19	17	17	19
Seguridad	5	22	15	19	22
Mantenibilidad	5	25	25	25	25
Portabilidad	3	15	14	14	15

Fuente: Elaboración Propia

Para cada característica, se dividió la calificación que obtuvo la herramienta sobre el número de subcaracterísticas pertenecientes a la misma.

Luego, para cada herramienta se realizó una suma entre los promedios obtenidos por características, y se encontró un promedio por cada suma.

Tabla 14: Resultado de la evaluación obteniendo el promedio

Características	Vectary	Model Viewer	A-Frame y Zappar	Onirix
Adecuación Funcional	5	3,67	3,67	5
Eficiencia de desempeño	4,67	4	4,33	4,67
Compatibilidad	5	4,5	4,5	5
Usabilidad	4,5	3,83	3,16	4,33
Fiabilidad	4,75	4,25	4,25	4,75
Seguridad	4,4	3	3,8	4,4
Mantenibilidad	5	5	5	5
Portabilidad	5	4,67	4,67	5
Promedio	4,79	4,11	4,17	4,77

Fuente: Elaboración Propia

Vectary con un puntaje de 4,79 sobre 5, cumple mejor con las características que propone el modelo de calidad iso/iec 25010. Esto lo convierte en la mejor propuesta para trabajar junto a aplicaciones móviles que requieren de un servidor de modelos 3D.

Se seleccionó Vectary debido a que permite almacenar los modelos en la nube, tiene mayor soporte con navegadores modernos, permite al usuario tener una mejor interacción con los modelos 3D, es compatible con más formatos de modelos 3D y por la calificación obtenida en la evaluación.

3.2. Conclusiones

- Se trabajó con Realidad Aumentada orientada a la Web y se integró esta tecnología en una aplicación desarrollada para dispositivos móviles.
- A través de la investigación realizada, se encontró diversas herramientas Web AR y por medio de la documentación fue posible probar su funcionamiento.
- Se comparó las características de las herramientas, costos para adquirir una suscripción mensual, soporte de los navegadores para trabajar con estas, entre otros aspectos.
- En base a la comparativa realizada y a la evaluación aplicada usando el modelo iso/iec 25010, se definió la herramienta más óptima para integrar en una aplicación móvil.
- Se evaluó las herramientas para verificar el cumplimiento de las características y subcaracterísticas que forman parte del modelo de evaluación iso/iec 25010, esto permitió identificar ventajas de una herramienta frente a otra.

3.3. Recomendaciones

Para aplicar Realidad Aumentada en una aplicación móvil se propone las siguientes recomendaciones:

- Investigar en distintas fuentes bibliográficas los beneficios de la Realidad Aumentada en diferentes ámbitos y formas de hacer llegar esta tecnología a más usuarios.
- Probar las herramientas usando dispositivos Android y iOS, para asegurarse que la experiencia sea adecuada para el usuario.
- Tener en cuenta que el soporte de los celulares para trabajar con esta tecnología no es el más adecuado, por lo tanto, usar herramientas que sean compatibles con la mayor cantidad posible de dispositivos.
- Antes de seleccionar una herramienta e integrar en una aplicación móvil, analizar si es la más adecuada comparando con otras herramientas o usando modelos de calidad.

- Si se requiere implementar Realidad Aumentada en una aplicación móvil, analizar si es más conveniente usar Web AR, o AR nativo según el tipo de dispositivos a quien va dirigida la aplicación.

4. BIBLIOGRAFÍA

- [1] O. M. Obrador, J. C. Almenara y J. Barroso, «Realidad aumentada aplicada a la enseñanza de la medicina,» *Educación Médica*, Julio 2017.
- [2] M. M. Saab, J. Hegarty, D. Murphy y M. Landers, «Incorporating virtual reality in nurse education: A qualitative study of nursing students' perspectives,» *Nurse Education Today*, vol. 105, 2021.
- [3] G. Lampropoulos, E. Keramopoulos y K. Diamantaras, «Enhancing the functionality of augmented reality using deep learning, semantic web and knowledge graphs: A review,» *Visual Informatics*, vol. 4, pp. 32-42, 2020.
- [4] S. Solmaz, J. L. D. Alfaro y P. Santos, «A practical development of engineering simulation-assisted educational AR environments,» *Education for Chemical Engineers*, vol. 35, pp. 81-93, 2021.
- [5] S. Nikhashemi, H. H. Knight, K. Nusair y C. B. Liat, «Augmented reality in smart retailing: A (n) (A) Symmetric Approach to continuous intention to use retail brands' mobile AR apps,» *Journal of Retailing and Consumer Services*, vol. 60, 2021.
- [6] B. Berman y D. Pollack, «Strategies for the successful implementation of augmented reality,» *Business Horizons*, vol. 64, pp. 621-630, 2021.
- [7] C. Duan, «Design of online volleyball remote teaching system based on AR technology,» *Alexandria Engineering Journal*, vol. 60, pp. 4299-4306, 2021.
- [8] M. Yavuz, E. Çorbacioğlu, A. N. Başoğlu, T. U. Daim y A. Shaygan, «Augmented reality technology adoption: Case of a mobile application in Turkey,» *Technology in Society*, vol. 66, 2021.

- [9] S. Qi, X. Ning, G. Yang, L. Zhang y P. Long, «Review of multi-view 3D object recognition methods based on deep learning,» *Displays*, vol. 69, 2021.
- [10] M. Drouot, N. L. Bigot, J. Bolloc'h, E. Bricard, J.-L. d. Bougrenet y V. Nourrit, «The visual impact of augmented reality during an assembly task,» *Displays*, vol. 66, 2021.
- [11] J. Jetter, J. Eimecke y A. Rese, «Augmented reality tools for industrial applications: What are potential key performance indicators and who benefits?,» *Computers in Human Behavior*, vol. 87, pp. 18-33, 2018.
- [12] C. Siedler, M. Glatt, P. Weber, A. Ebert y J. C. Aurich, «Engineering changes in manufacturing systems supported by AR/VR collaboration,» *Procedia CIRP*, vol. 96, pp. 307-312, 2021.
- [13] G. Kayakutlu y Eunika Mercier-Laurent, «Future of Energy,» *Intelligence in Energy*, pp. 153-198, 2017.
- [14] F. Süli, «Market segments,» *Electronic Enclosures, Housings and Packages*, pp. 55-72, 2019.
- [15] D. Galar y U. Kumar, «Maintenance Decision Support Systems,» *eMaintenance*, pp. 371- 474, 2017.
- [16] R. M. Boje y I. Calvo-Muñoz, «Efectos de la terapia de realidad virtual en el miembro superior en pacientes con ictus: revisión sistemática,» *Rehabilitación*, vol. 52, pp. 45-54, 2018.
- [17] M. F. Pereira, C. Prahm, J. Kolbenschlag, E. Oliveira y N. F. Rodrigues, «Application of AR and VR in hand rehabilitation: A systematic review,» *Journal of Biomedical Informatics*, vol. 111, 2020.
- [18] N. Palani, «ONE-GUI Designing for Medical Devices & IoT introduction,» *Trends in Development of Medical Devices*, pp. 17-34, 2020.

- [19] M. B. Privitera, «Special cases: introduction,» *Applied Human Factors in Medical Device Design*, pp. 301-302, 2019.
- [20] M. Alahmari, T. Issa, T. Issa y S. Zaung Nau, «Faculty awareness of the economic and environmental benefits of augmented reality for sustainability in Saudi Arabian universities,» *Journal of Cleaner Production*, vol. 226, pp. 259-269, 2019.
- [21] M. Y.-C. Yim y S.-Y. Park, «“I am not satisfied with my body, so I like augmented reality (AR)”: Consumer responses to AR-based product presentations,» *Journal of Business Research*, vol. 100, pp. 581-589, 2019.
- [22] Mohanty, B. Prasad, Goswami y Laxmi, «Advancements in augmented reality,» *Materials Today: Proceedings*, 2021.
- [23] R. Layona, B. Yulianto y Yovita Tunardi, «Web based Augmented Reality for Human Body Anatomy Learning,» *Procedia Computer Science*, vol. 135, pp. 457-464, 2018.
- [24] J. Buchner, K. Buntins y Michael Kerres, «A systematic map of research characteristics in studies on augmented reality and cognitive load,» *Computers and Education Open*, vol. 2, 2021.
- [25] T. Plewan y B. Mättig, «Exploring the benefits and limitations of augmented reality for palletization,» *Applied Ergonomics*, vol. 90, 2021.
- [26] Vectary, «Vectary,» [En línea]. Available: <https://www.vectary.com/>.
- [27] J. Flotyński, K. Walczak y M. Krzyszkowski, «Composing customized web 3D animations with semantic queries,» *Graphical Models*, vol. 107, 2020.
- [28] Google, «Model Viewer,» [En línea]. Available: <https://modelviewer.dev/>.
- [29] Zappar, «Zappar,» [En línea]. Available: <https://www.zappar.com/>.
- [30] A-Frame, «A-Frame,» [En línea]. Available: <https://aframe.io/docs/1.2.0/introduction/>.

- [31] Onirix, «Onirix,» [En línea]. Available: <https://www.onirix.com/es/creative-studio/>.
- [32] K. Peguero y X. Cheng, «CSRF protection in JavaScript frameworks and the security of JavaScript applications,» *High-Confidence Computing*, vol. 1, 2021.
- [33] Node.js, «Node.js,» [En línea]. Available: <https://nodejs.org/es/about/>.
- [34] N. Lionel y Jang-Jong-Wook, «Performance Evaluation of Server-side JavaScript for Healthcare Hub Server in Remote Healthcare Monitoring System,» *Procedia Computer Science*, vol. 98, pp. 382-387, 2016.
- [35] G. Manduchi, A. Luchetta, G. Moro, A. Rigoni y C. Taliercio, «Web-based streamed waveform display using MDSplus events and Node.js,» *Fusion Engineering and Design*, vol. 157, 2020.
- [36] F. Kaimer y P. Brune, «Return of the JS: Towards a Node.js-Based Software Architecture for Combined CMS/CRM Applications,» *Procedia Computer Science*, vol. 141, pp. 454-459, 2018.
- [37] K.-I. D. Kyriakou, N. D. Tselikas y G. M. Kapitsaki, «Enhancing C/C++ based OSS development and discoverability with CBRJS: A Rust/Node.js/WebAssembly framework for repackaging legacy codebases,» *Journal of Systems and Software*, vol. 157, 2019.
- [38] Mozilla, «MDN Web Docs,» [En línea]. Available: https://developer.mozilla.org/es/docs/Learn/Server-side/Express_Nodejs/Introduction.
- [39] E. Besada-Portas, J. Bermúdez-Ortega, L. d. I. Torre, J. López-Orozco y J. d. I. Cruz, «Lightweight Node.js & EJS-based Web Server for Remote Control Laboratories,» *IFAC-PapersOnLine*, vol. 49, pp. 127-132, 2016.
- [40] F. Galeazzi, M. Callieri, M. Dellepiane, M. Charno, J. Richards y R. Scopigno, «Web-based visualization for 3D data in archaeology: The ADS

3D viewer,» *Journal of Archaeological Science: Reports*, vol. 9, pp. 1-11, 2016.

[41] A. M. Vukicevic, M. Djapan, M. Stefanovic y I. Macuzic, «SafE-Tag mobile: A novel javascript framework for real-time management of unsafe conditions and unsafe acts in SMEs,» *Safety Science*, vol. 120, pp. 507-516, 2019.

[42] Expo, «Expo,» [En línea]. Available: <https://docs.expo.dev/>.

[43] npm, «npm expo-web-browser,» [En línea]. Available: <https://www.npmjs.com/package/expo-web-browser>.

[44] Microsoft, «Visual Code Studio,» Microsoft, [En línea]. Available: <https://code.visualstudio.com/docs>.

[45] ISO 25000, «ISO/IEC 25010,» [En línea]. Available: <https://iso25000.com/index.php/normas-iso-25000/iso-25010>.

5. ANEXOS