



Sede “Julio Antonio Mella”

Facultad de Ingeniería en Telecomunicaciones,
Informática y Biomédica

Trabajo de Diploma

En opción al título de Ingeniero en Informática

Título:

“ARShop, Aplicación para dispositivos móviles
con tecnología de Realidad Aumentada para la
visualización de muebles”

Autor: Yilbert Benitez Barrio

Tutor: Dr.C. Dionis López Ramos

Santiago de Cuba, 2024

Resumen

En la actualidad, comprar muebles para el hogar es más cómoda y accesible porque los compradores con un dispositivo móvil pueden acceder a las plataformas de compras en línea: visualizando o comprando los diferentes muebles; sin tener que desplazarse físicamente a múltiples tiendas. Esto permite tomar decisiones de compra informadas y eficientes. Sin embargo, la dificultad de no poder visualizar cómo se verá un mueble en nuestro entorno antes de la compra puede llevar a adquirir productos inadecuados que no se ajustan a nuestras necesidades ni al tamaño de nuestro hogar. Además, la falta de percepción de cómo combinar colores y estilos también puede resultar en decisiones de compra erróneas. En esta investigación se diseña e implementa un prototipo para desplegarse en dispositivos móviles utilizando Flutter, como marco de trabajo de desarrollo, y la tecnología de **realidad aumentada** que permite la visualización del posible mueble a comprar. Se crea una plataforma web para la publicación de diferentes tipos de muebles a ofertar por diferentes tiendas y el prototipo antes mencionado permite la visualización en los espacios del hogar. La persistencia de la información se logra con el sistema gestor de bases de datos MySQL y se utilizan servicios web para el consumo de la información, entre dispositivos, con una arquitectura cliente servidor.

Palabras claves: realidad aumentada, teléfonos inteligentes, comercio digital, flutter.

Abstract

Currently, buying furniture for the home is more convenient and accessible because shoppers with a mobile device (smartphone or tablet) can access online shopping platforms: viewing, comparing, or purchasing different furniture without having to physically visit multiple stores. This allows for informed and efficient purchasing decisions. However, the difficulty of not being able to visualize how a piece of furniture will look in our environment before purchase can lead to acquiring unsuitable products that do not fit our needs or the size of our home. Additionally, the lack of perception of how to combine colors and styles can also result in poor purchasing decisions. In this research, a prototype is designed and implemented to be deployed on mobile devices (e.g., smartphones and tablets) using Flutter as the development framework and **augmented reality** technology that allows the visualization of the possible furniture to be purchased. A digital platform is created for the publication of different types of furniture offered by various stores, and the aforementioned prototype allows visualization in home spaces. The persistence of information is achieved with the MySQL database management system, and web services are used to consume the information with a client-server architecture.

Keywords: augmented reality, smartphones, digital commerce, flutter.

Índice

Introducción.....	1
Capítulo 1. Marco Teórico.....	6
1.1 Estado del arte de las aplicaciones móviles que utilizan realidad aumentada para la visualización de muebles:.....	6
1.2 Términos y definiciones.....	9
1.3 Herramientas, lenguajes y tecnologías utilizadas	10
1.3.1 Herramientas.....	10
1.3.2 Lenguajes	14
1.3.3 Tecnologías.....	15
1.4 Arquitectura MVC (Modelo-Vista-Controlador)	17
1.5 Metodología de desarrollo de software XP	18
1.6 Conclusiones del Capítulo.....	21
Capítulo 2. Organización y Diseño.	22
2.1 Propuesta del Sistema.....	22
2.2 Objetivo a alcanzar	22
2.3 Flujo del Sistema	22
2.3.1 Flujo del Sistema para la visualización de los muebles con RA:	23
2.4 Usuarios del sistema	24
2.5 Requerimientos para el despliegue de la App	25
2.6 Historias de Usuarios	25
2.7 Diagrama entidad relación para la persistencia en la base de datos	28
2.8 Diagramas de Secuencia.....	29
2.9 Diagrama de Actividades	30
2.10 Conclusiones del capítulo.....	31
Capítulo 3. Implementación y Prueba.....	32

3.2 Diagrama de clases	33
3.3 Algoritmos importantes.....	35
3.3.1 Algoritmo para mostrar el catálogo de productos.....	35
3.3.2 Algoritmo para la visualización del producto en 3D.....	35
3.4 Pruebas al sistema	36
3.4.1 Pruebas a la aplicación para dispositivos móviles	37
3.4.2 Pruebas al servidor	38
3.5 Análisis económico del costo de producción del sistema	40
3.5.1 Estimación de costo y tiempo.....	40
3.6 Conclusiones del Capítulo.	41
Conclusiones	42
Recomendaciones.....	43
Referencias Bibliográficas	44

Introducción

En la actualidad, la compra de muebles del hogar como sillas, mesas, sillones, etc. (ver una lista de los muebles del hogar probados en esta investigación en el [anexo 7](#)) se ha convertido en una tarea complicada para los usuarios debido a la insuficiencia en la visualización de los mismos. Los usuarios se enfrentan a un problema al momento de seleccionar muebles, ya que la falta de información sobre cómo lucirían en su espacio y cómo se adaptarían al resto de la decoración dificulta la toma de decisiones. Por esta razón, es necesario buscar alternativas tecnológicas que permitan mejorar la experiencia de compra de los usuarios.

Además en la industria y comercialización de mobiliarios de madera se ha hecho uso de diferentes tipos de catálogos para promocionar y posicionar el producto frente al consumidor, mediante los cuales el cliente puede observar ya sea en hojas impresas o en imágenes digitales el producto que ha sido realizado con anterioridad; el problema se presenta cuando el cliente quiere saber si el mobiliario escogido es apropiado al estilo de su vivienda, ya sea por el color o el tipo de material con el que es fabricado el mobiliario (Bolaños, 2019).

Bolaños (2019) señala que, con el paso del tiempo se ha mejorado la manera de presentar al cliente los diferentes tipos de mobiliario, de tal forma se ha hecho uso de bosquejos a mano alzada y de herramientas informáticas especializadas para dicho diseño. Sin embargo, en el primer caso, el cliente no puede visualizar claramente los detalles finales y el tipo de acabado del mueble; en el segundo caso se requiere de un mayor tiempo para entregar el diseño con detalles más claros, lo que provoca que el cliente pierda el interés y muchas veces desista de adquirir el o los productos.

Es por eso que las empresas están cada vez más conscientes de la importancia que la tecnología representa en la vida de las personas y, por esa razón, muchas están buscando nuevas alternativas para llamar la atención de su audiencia mediante procesos de transformación digital. En este contexto, la Realidad

Aumentada (RA) puede ser de gran ayuda, ya que además de ser extremadamente eficiente, abre infinitas posibilidades en comparación con otras tecnologías. La versatilidad de este recurso permite implementarlo en diferentes ocasiones, con la finalidad de alcanzar distintos objetivos («Realidad aumentada», 2019).

Una de las herramientas que ayuda a la visualización de objetos como son los muebles es la realidad aumentada. Esta se ha convertido en una tecnología emergente que permite la integración de elementos virtuales en el entorno real, brindando a los usuarios una experiencia más interactiva y enriquecedora (Azuma, 1997). En el campo de las aplicaciones para dispositivos móviles (ej.: teléfonos inteligentes y tablets), la RA ha encontrado diversas aplicaciones, entre ellas, la visualización de productos, que permite a los usuarios apreciar los objetos en un contexto real antes de realizar la compra (Carmigniani & Furht, 2011).

La RA puede definirse como una combinación visual de elementos reales y virtuales que interaccionan entre ellos. Brindando una experiencia visual y sensorial enriquecida. En el ámbito de la compra de muebles, esta tecnología puede jugar un papel fundamental al permitir a los usuarios visualizar y probar virtualmente los muebles antes de realizar una compra.

Al contrario que en la realidad virtual, en la RA la persona no está inmersa dentro del entorno, el dispositivo suele ser un teléfono móvil o una tableta desde la que vemos nuestro entorno real a través de la cámara y sobre ella se muestran elementos inexistentes que corresponden al entorno virtual (Ramírez A, *et al.* 2018).

Sin embargo, se ha identificado que existe una insuficiencia en la visualización de muebles, antes de la compra, a través de las opciones de comercio digital existentes en Cuba (ed., sitios o aplicaciones web de tiendas virtuales de venta de muebles), lo que dificulta la correcta selección de los mismos por parte de los usuarios.

Esta limitación puede llevar a compras erróneas, devoluciones frecuentes y una experiencia de compra insatisfactoria.

Teniendo en cuenta lo expuesto anteriormente se determina como **problema científico**: Insuficiencia en la visualización de muebles a la hora de la compra por parte de los usuarios lo que atenta a la correcta selección de los mismos.

El **objeto de esta investigación** se centra en los procesos de visualización de muebles a través de aplicaciones informáticas que utilizan tecnologías de realidad aumentada. Se busca analizar los principales problemas y deficiencias presentes en las aplicaciones actuales, así como explorar nuevas soluciones que permitan mejorar la perspectiva de compra de los usuarios.

El **objetivo general** de este estudio es desarrollar una aplicación para dispositivos móviles con tecnología de realidad aumentada, denominada "ARShop", que contribuya a mejorar la experiencia de visualización de muebles por parte de los usuarios. Esta aplicación buscará ofrecer una representación precisa y detallada de los muebles en el entorno real, permitiendo a los usuarios obtener una perspectiva realista antes de tomar una decisión de compra.

Para alcanzar este objetivo, se realizó una revisión exhaustiva de la literatura relacionada con la realidad aumentada y su aplicación en el ámbito de la visualización de muebles. Se analizaron estudios previos que aborden los problemas y limitaciones existentes, así como aquellos que propongan soluciones innovadoras. Además, se utilizaron técnicas de diseño y desarrollo de aplicaciones para dispositivos móviles para crear y evaluar la efectividad de la aplicación "ARShop".

En este sentido, el **campo de acción de la investigación** está dirigido a: Las aplicaciones para dispositivos móviles para la visualización de muebles con tecnología de realidad aumentada.

La **hipótesis** de la investigación plantea que: Si se desarrolla una aplicación para dispositivos móviles con tecnología de realidad aumentada donde se puedan visualizar los muebles antes de comprarlos esta permitiría a los usuarios hacer una mejor compra debido a que podrían ver desde un principio como se verán estos en sus hogares logrando así una mejor selección del mueble a comprar.

Para orientar el curso de la investigación y dar cumplimiento al objetivo y a la hipótesis planteada, se enuncian los siguientes **objetivos específicos**:

1. Sistematizar los principales referentes teóricos sobre el proceso de visualización a través de aplicaciones móviles con tecnología realidad aumentada.

2. Identificar metodologías, requisitos funcionales y no funcionales de la aplicación para dispositivos móviles para la visualización a la hora de compra de muebles.
3. Diseñar la aplicación para dispositivos móviles y otras herramientas necesarias.
4. Implementar las aplicaciones necesarias para lograr el objetivo de la investigación.
5. Realizar las pruebas de las aplicaciones, corregir errores y sugerencias de los usuarios.

El desarrollo de la investigación demandó la utilización de los siguientes métodos y técnicas de investigación.

Los métodos del **nivel teórico** utilizados fueron:

- ✓ Histórico-lógico: Se aplicó durante la revisión y análisis de documentos y sistemas asociados al proceso de gestión de la información referente a los sistemas informáticos relacionados con la visualización de muebles con tecnología de realidad aumentada.
- ✓ Inductivo-deductivo: Utilizado en la extracción de las ideas fundamentales para la construcción y fundamentación teórica del trabajo de diploma. Además de la realización de razonamientos sobre los temas investigados y así obtener conocimientos generales y particulares sobre el objeto de la investigación.
- ✓ Analítico-sintético: para la fundamentación teórica del objeto y el campo de investigación, así como la interpretación de los datos que aportan los métodos empíricos utilizados en toda la lógica del proceso investigativo lo que contribuyó a arribar a generalizaciones teóricas. En la etapa del desarrollo del sistema se utilizó para el desglose de los requisitos funcionales, en la identificación de los patrones de diseño y el análisis de los elementos del software para aplicar los patrones utilizados.
- ✓ Modelación: Se empleó para realizar los diagramas necesarios para documentar el software.

Dentro de los métodos y técnicas del **nivel empírico** se encuentran:

- ✓ Análisis de documentos: Posibilitó el estudio del tema investigado, particularmente relacionados con la carrera Ingeniería en informática para dar tratamiento a la problemática.

- ✓ Entrevistas: Se les aplicó a trabajadores del Centro de Desarrollo de software.

También demandó la aplicación de los **métodos matemático-estadísticos**: la técnica del análisis porcentual posibilitó el procesamiento de los datos obtenidos, también se empleó el trabajo con tablas para exponer la información de los principales aportes de la investigación.

Estructura del Trabajo:

El presente trabajo se estructura en introducción, desarrollo que incluye tres capítulos en los que se abordan las características más relevantes del tema de la investigación, además de ofrecer conclusiones, recomendaciones, referencias bibliográficas y anexos como medios de apoyo del estudio que se presenta.

Capítulo I: Fundamentación Teórica. En este capítulo se hace un análisis de los principales conceptos y términos que se utilizan y regulan en la realización de la venta de muebles mediado por aportes informáticos, otro elemento que aparece implícito es el estudio de la metodología de desarrollo de aplicaciones para dispositivos móviles y las herramientas y tecnologías que se tuvieron en cuenta para la realización de la solución propuesta.

Capítulo II: Planificación y Diseño. Este capítulo contiene la propuesta del trabajo de investigación, en la cual se detalla el diseño de la aplicación tomando como base la metodología de desarrollo estudiada en el capítulo I.

Capítulo III: Implementación y prueba de la aplicación para dispositivos móviles. Se describe de manera clara y concisa la implementación del proyecto en su totalidad, la elaboración de los marcadores para cada mobiliario específico, el software para la elaboración de las aplicaciones, se detalla las características técnicas de los dispositivos móviles en los que va a funcionar la aplicación y las pruebas de funcionamiento en ambientes reales, lo que permite adecuar los procesos de desarrollo y lograr una buena calidad con el fin del producto.

Aportes de la investigación:

- ✓ Desarrollo de una aplicación para dispositivos móviles con tecnología de realidad aumentada, denominada "ARShop", que contribuye a mejorar la experiencia de

visualización de muebles por parte de los usuarios de plataformas de venta de muebles en línea.

- ✓ Incorporación de un algoritmo que emplea la realidad aumentada para la visualización de muebles.

Capítulo 1. Marco Teórico

En este capítulo se abordan los conceptos relacionados con la investigación, los cuales constituyen el conocimiento sobre los procesos y tecnologías asociadas al desarrollo de la solución planteada. Se exponen y explican los lenguajes de programación, metodologías, herramientas y tecnologías a utilizar en el desarrollo de la aplicación, justificando el motivo de su selección.

1.1 Estado del arte de las aplicaciones móviles que utilizan realidad aumentada para la visualización de muebles:

El uso y aplicaciones de la RA ha ido en crecimiento en los últimos años, el objetivo principal de esta tecnología es obtener una combinación entre el mundo real y el mundo virtual proporcionando una fascinante experiencia visual que facilita y mejora la calidad de comunicación. Actualmente la realidad aumentada es usada para distintos tipos de aplicaciones tales como: Catálogos de Artículos, obras en tres dimensiones (3D), Videojuegos, etc. Como lo detallan Jorge Caiza y Santiago Tapia en (Caiza & Tapia, 2017).

En el campo de las aplicaciones móviles, la RA ha encontrado diversas aplicaciones, entre ellas, la visualización de productos que permite a los usuarios apreciar los objetos en un contexto real antes de realizar la compra (Carmigniani & Furht, 2011).

Algunas aplicaciones móviles que utilizan realidad aumentada para la visualización de muebles incluyen:

Houzz¹: Es una aplicación (app) de decoración disponible para Android y para iOS que, además de ofrecer multitud de ideas para decorar en forma de fotos, permite

¹ <https://www.houzz.com/>

poner los objetos virtualmente en casa. Posee desde una galería muy completa con fotos que sirven como ejemplo de decoración. La aplicación permite observar virtualmente cómo quedan muchos de los artículos que se encuentran en las distintas categorías. (*Houzz, la aplicación de decoración con la que renovar tu casa*, s. f.)

IKEA Place² (Ikea): Permite a los usuarios visualizar muebles de IKEA en sus propios espacios, utilizando el teléfono inteligente. La aplicación superpone la imagen virtual del mueble seleccionado en el lugar donde se quiere colocar. De esta manera, los usuarios pueden ver cómo quedaría el mueble en su hogar, comprobar las medidas y los colores, y decidir si el mueble se ajusta a sus necesidades (Ikea ,2017)

La App IKEA Place es una herramienta muy útil para los usuarios que quieren comprar muebles de IKEA sin tener que arriesgarse a que el producto no quede bien en su hogar. La aplicación permite ahorrar tiempo y dinero al evitar devoluciones y cambios de muebles. Además, ofrece una experiencia de compra más personalizada y adaptada a las necesidades de cada usuario (*Ikea Place, su aplicación de realidad aumentada para decorar tu casa llega a los móviles Android con ARCore*, s. f.).

Wayfair View in Room ³(Wayfair): Aplicación que permite a los usuarios visualizar muebles de wayfair en sus hogares a través de RA (Wayfair, 2019). Es una aplicación web y móvil a la que puedes acceder desde cualquier parte del mundo. Al igual que otras aplicaciones de compra, también tiene una potente barra de búsqueda donde debe colocar el nombre del producto para obtener una lista de artículos relacionados. Wayfair ofrece una solución ingeniosa: el uso de realidad aumentada (AR) para colocar estos objetos en nuestro hogar. Así descubriremos si encajan con nuestro estilo actual o si tienen las medidas idóneas (SL, s. f.).

Esta aplicación, que en estos momentos está disponible en EE.UU., Reino Unido, Alemania y Canadá, permite ver en la pantalla de un teléfono inteligente o tablet

² <https://www.ikea.com/us/en/customer-service/mobile-apps/>

³ <https://www.wayfair.com/the-wayfair-app>

cómo quedaría cada pieza del mobiliario que vende esta compañía, en cualquier rincón de la casa, de una forma tan sencilla (*La aplicación móvil Wayfair, un ejemplo de como compraremos muebles mañana*, s. f.).

Nuestro país con el aumento del acceso a internet de la población y el uso de la tecnología hace que el acceso a marcas, productos y servicios sea mucho más fácil y continuo. El acceso a esta tecnología permite la comodidad de realizar una compra desde el hogar y poder, incluso, pagar sin necesidad de desplazarse del lugar y usando los canales electrónicos (*Comercio electrónico en Cuba: una realidad creciente*, s. f.).

Es por ello que muchas empresas cubanas han desarrollado sitios web como “Muebles cubanos⁴”, “Leo y Miguel⁵” y “Madesa” con el fin de hacer las compras más cómodas y fáciles a los usuarios, aun así sigue siendo una tarea difícil comprar un mueble, por lo que se plantea como problema de investigación en este trabajo la insuficiencia en la visualización de muebles a la hora de la compra (no es posible tener una correspondencia entre el espacio del hogar, su decoración y las dimensiones del mueble a comprar).

No obstante, se encontraron algunas empresas en el ámbito internacional que desarrollaron aplicaciones que ayudan a hacer compras más confiables y seguras, pero algunas de ellas tienen desventajas en nuestro medio porque son de pago y no están soportadas para todos los países como la app propuesta por wayfair.

Otra deficiencia es la forma de publicar los posibles muebles, diseñados para poder ser visualizados en el dispositivo móvil. La mayoría de las apps de las tiendas de venta de muebles en línea internacionales analizadas en la investigación, no permiten agregar muebles de otras tiendas o buscar entre ellos para compararlos o visualizarlos en el hogar.

En nuestro país hacer compras de muebles desde otros países se hace difícil a los usuarios cubanos por problemas de comunicación, uso de diferentes monedas, entre otros y muchas de las aplicaciones son para visualizar solo muebles vendidos

⁴ <https://www.mueblescubanos.com/home-mobile/>

⁵ <https://www.leoymigue.com/>

por una empresa (p.ej., IKEA, Wayfair) y estas aplicaciones no son accesibles desde Cuba.

Todos los problemas expuestos anteriormente sugieren tomar la decisión de crear una aplicación para dispositivos móviles en la cual los usuarios cubanos tengan la posibilidad de poder visualizar los muebles en venta por las diferentes tiendas o empresas que pueden ofertar estos productos en sus hogares antes de realizar la compra y así hacer una mejor elección y tener una experiencia positiva.

1.2 Términos y definiciones.

A continuación, se representa un conjunto de términos y definiciones de los principales conceptos involucrados en el problema de investigación.

Tecnología móvil: La tecnología móvil se refiere a los dispositivos, redes y servicios que permiten la comunicación y el acceso a la información de manera inalámbrica y portátil, como teléfonos celulares, smartphones, tablets y otros dispositivos inteligentes.

Las redes de comunicación que conectan estos dispositivos se denominan tecnologías inalámbricas. Permiten que los dispositivos móviles compartan llamadas, datos y aplicaciones (aplicaciones móviles) (Organización Internacional de Normalización (ISO), 2013).

Sistema Operativo Android: Android es un sistema operativo móvil basado en el kernel de Linux y diseñado principalmente para dispositivos móviles con pantalla táctil, como teléfonos inteligentes y tabletas. Android es desarrollado y mantenido por Google, y es el sistema operativo móvil más utilizado en el mundo. (Cheng & Liu, 2018).

Realidad Aumentada: La Realidad Aumentada se refiere a la tecnología que combina elementos del mundo real con elementos virtuales para crear una experiencia mejorada y ampliada. Esta tecnología permite superponer gráficos, imágenes, videos y otros contenidos digitales en el entorno real. Uno de los primeros autores en mencionar el concepto de Realidad Aumentada fue Tom Caudell en 1990.

Dentro de los distintos tipos de realidad aumentada se destacan los siguientes:

- ✓ Realidad aumentada sobre marcadores: es aquella que usa marcadores visuales. Entre ellos se destacan los códigos QR o las imágenes. Se ejecuta a partir de la superposición de elementos virtuales.
- ✓ Realidad aumentada geolocalizada: es utilizada principalmente para la geolocalización de lugares específicos. Se ve en aplicaciones de turismo o entornos de navegación.
- ✓ Realidad aumentada holográfica: se hacen proyecciones tridimensionales de un objeto virtual en un espacio real. Así, se genera una experiencia inmersiva y realista. Es muy común en sectores como la arquitectura y la medicina.

Dentro de las distintas ventajas de la realidad aumentada se destaca el poder ofrecer una experiencia que sea totalmente inmersiva. Gracias a ello, los clientes pueden interactuar tanto con los productos como servicios de la empresa. Esto puede generar un mayor compromiso y fidelidad hacia la marca, así como aumentar las ventas. Otra ventaja clave es la posibilidad de personalizar las experiencias de los clientes. Las empresas pueden adaptar los productos o servicios a las necesidades individuales de cada cliente, lo que crea un vínculo más estrecho y fortalece la relación comercial. También puede mejorar la eficiencia y la productividad interna de las empresas.

(¿Qué es la realidad aumentada (AR)? | SAP Insights, s. f.)

1.3 Herramientas, lenguajes y tecnologías utilizadas

Para dar cumplimiento al objetivo propuesto en nuestra investigación se hizo necesario el empleo de un conjunto de herramientas y tecnologías informáticas que hiciera posible la implementación de un sistema que cumpla con los requerimientos del cliente y arroje resultados satisfactorios con las pruebas a las que será sometido. Estas herramientas se describen a continuación.

1.3.1 Herramientas

Según Sánchez-Rivas et al. (2019), las herramientas se refieren a "todo instrumento, aplicación o recurso que facilita la realización de tareas y actividades

dentro de un determinado contexto, ya sea en el ámbito personal, académico o profesional". Las herramientas pueden ser tanto físicas como digitales y su uso permite mejorar la eficiencia y productividad en diversas áreas. Para dar cumplimiento al objetivo propuesto en nuestra investigación se utilizaron las siguientes herramientas:

Visual Studio Code (VSC): 1.73.0 Según Hemon y Lutellier (2020), VSCode es "un editor de texto gratuito y de código abierto que se destaca por su facilidad de uso, extensibilidad y soporte para una amplia gama de lenguajes de programación".

Este se basa en Electron, un *framework* que se utiliza para implementar Chromium y Node.js como aplicaciones para escritorio, que se ejecuta en el motor de diseño Blink. Aunque utiliza el *framework* Electron, el software no usa Atom y en su lugar emplea el mismo componente editor (Monaco) utilizado en Visual Studio Team Services (anteriormente llamado Visual Studio Online).

Además, es una herramienta que permite a los desarrolladores escribir y editar código fuente para sitios web y aplicaciones. Ofrece funciones como resaltado de sintaxis y autocompletado del código para facilitar la programación. VSC es altamente personalizable a través de extensiones que permiten adaptarlo a diferentes tareas, tecnologías y lenguajes de programación. Los desarrolladores pueden trabajar en diversos lenguajes como PHP, Python, Java, entre otros, gracias a la amplia gama de extensiones disponibles.

(*Visual Studio Code - Wikipedia, la enciclopedia libre*, s. f.).

Android Studio: 17.0.10 es el entorno de desarrollo integrado (IDE) oficial para la plataforma Android. Según Gu *et al.* (2019), Android Studio proporciona un conjunto de herramientas y funcionalidades que facilitan el desarrollo de aplicaciones móviles para el sistema operativo Android.

Android Studio ofrece aún más funciones que mejoran la productividad cuando se compilan ~~compilan~~ apps para Android, como las siguientes:

- ✓ Un sistema de compilación flexible basado en Gradle.
- ✓ Un emulador rápido y cargado de funciones.

- ✓ Un entorno unificado donde se pueden desarrollar app para todos los dispositivos Android.
- ✓ Ediciones en vivo para actualizar elementos componibles en emuladores y dispositivos físicos, en tiempo real.
- ✓ Integración con GitHub y plantillas de código para ayudar a compilar funciones de apps comunes y también importar código de muestra.

Android Studio usa Gradle como base del sistema de compilación, y el complemento de Android para [HYPERLINK](#), Gradle proporciona capacidades específicas de Android. Este sistema de compilación se ejecuta en una herramienta integrada desde el menú de Android Studio, y lo hace independientemente de la línea de comandos (*Introducción a Android Studio*, s. f.).

MySQL: 8.4.0 es un sistema de gestión de bases de datos relacional de código abierto, desarrollado y mantenido por Oracle Corporation. Es uno de los sistemas de gestión de bases de datos más populares y ampliamente utilizados en la industria del software (Verma, 2019).

MySQL es una base de datos altamente valorada por su facilidad de uso, permitiendo a los desarrolladores instalarla rápidamente y administrarla sin complicaciones. Además, se destaca por su fiabilidad, siendo una de las bases de datos más maduras y utilizadas en el mercado, probada en diversos escenarios durante más de 25 años. Su escalabilidad es otro punto fuerte, ya que puede satisfacer las demandas de las aplicaciones más accedidas, con la capacidad de escalar para soportar grandes cantidades de usuarios gracias a su arquitectura de replicación nativa (*What Is MySQL? | Oracle India*, s. f.) .

En el desarrollo del sistema, se optó por utilizar MySQL como solución de base de datos debido a su robustez, flexibilidad y amplia comunidad de soporte.

Git: 2.43.0 es un sistema de control de versiones distribuido de código abierto diseñado para manejar desde proyectos pequeños hasta muy grandes, con velocidad y eficiencia (Umali, 2020).

Los objetivos de Git incluyen velocidad, integridad de datos y soporte para flujos de trabajo distribuidos y no lineales. Git fue escrito originalmente por Linus Torvalds en 2005 para el desarrollo del kernel de Linux, y otros desarrolladores del kernel contribuyeron a su desarrollo inicial.

El diseño de Git es una síntesis de la experiencia de Torvalds con Linux en el mantenimiento de un gran proyecto de desarrollo distribuido, junto con su profundo conocimiento del rendimiento del sistema de archivos obtenido del mismo proyecto y la urgente necesidad de producir un sistema que funcione en poco tiempo.

La adopción de Git ha crecido rápidamente, convirtiéndose en el sistema de control de versiones distribuido más popular, y casi el 95 % de los desarrolladores lo consideran su principal sistema de control de versiones a partir de 2022. Es la herramienta de gestión de código fuente más utilizada entre los desarrolladores profesionales («Git», 2024).

En el desarrollo del sistema, se adoptó Git como sistema de control de versiones distribuido, lo cual resultó fundamental para la gestión eficiente de los cambios en el código fuente. Git permite a los desarrolladores realizar un seguimiento detallado de los cambios realizados, fusionar modificaciones de manera segura y revertir a versiones anteriores del código si era necesario.

Visual paradigm: 8.0 es una herramienta de modelado de software de código abierto que permite a los desarrolladores y analistas de sistemas crear diagramas UML, diagramas de flujo, diagramas de base de datos y otros tipos de diagramas visuales para guiar el proceso de desarrollo de software. (Lim, 2018).

Visual Paradigm permite a su equipo gestionar la complejidad de la transformación empresarial para hacer frente a la rápida evolución de los mercados, las tecnologías y los requisitos reglamentarios. Es una solución integral ideal para la planificación de la arquitectura empresarial y la transformación del negocio, la gestión de proyectos y el desarrollo ágil de software, para que su empresa pueda mantener el control y fomentar el crecimiento (*Acerca de Visual Paradigm*, s. f.).

Durante el desarrollo del sistema, Visual Paradigm fue seleccionado como herramienta de modelado UML debido a su versatilidad y capacidad para facilitar la

comunicación y comprensión de los requisitos del sistema. Visual Paradigm ofrece una amplia gama de características que no solo ayudan a modelar el sistema o aplicación, sino que también hacen posible modelar cosas de manera más fácil, rápida y precisa. Entre las funciones destacadas se encuentran el soporte para dibujar todo tipos de diagramas UML, incluyendo diagramas de casos de uso, diagramas de clases, secuencia, actividades, entre otros.

1.3.2 Lenguajes

"Un lenguaje de programación es un sistema formal diseñado para comunicar instrucciones a una computadora o dispositivo programable para realizar tareas específicas. Los lenguajes de programación permiten a los desarrolladores expresar algoritmos de una manera que puede ser interpretada y ejecutada por una máquina." (Sommerville, 2018). A continuación, se presentan los lenguajes utilizados en la implementación del sistema.

Dart: 3.3 Según Gupta (2019), "Dart es un lenguaje de programación de código abierto desarrollado por Google con el objetivo de reemplazar JavaScript en la web y como lenguaje para aplicaciones móviles". Además, Firdaus y Aminuddin (2020) afirman que "Dart es un lenguaje de programación de alto nivel que se enfoca en la creación de aplicaciones web, móviles y de servidor de manera eficiente y escalable". Es un lenguaje orientado a objetos y puede ser utilizado para desarrollar aplicaciones web, móviles y de servidor. Tiene herramientas como:

- ✓ **Chromium** es el directorio donde se almacena una compilación especial del navegador web Chromium llamada Dartium porque incluye la máquina virtual de Dart (Dart VM). Actualmente no es posible ejecutar código nativo Dart en ningún navegador más porque ninguno incluye aún la máquina virtual; es posible hacerlo si se convierte Dart a JavaScript.
- ✓ **Dart-sdk** es el kit de desarrollo de software de Dart (SDK). Incluye todas las librerías de Dart como dart:core o dart:html y tiene herramientas de línea de comandos muy útiles como el compilador de Dart-to-JavaScript y la máquina virtual de Dart.
- ✓ **Samples** contiene varios ejemplos de aplicaciones Dart para aprender y profundizar con ejemplos de aplicaciones reales.

Para este proyecto se empleó Dart debido a que es el lenguaje de programación utilizado por Flutter, es rápido, moderno y optimizado para el desarrollo móvil, lo que contribuye a la eficiencia y la capacidad de Flutter para crear aplicaciones de alta calidad.

Python: 3.12 es un lenguaje de programación de alto nivel, interpretado y dinámico, desarrollado por Guido van Rossum. Según Matthes (2019), "Python es un lenguaje de programación de alto nivel, interpretado y orientado a objetos que se utiliza ampliamente en una variedad de aplicaciones, desde desarrollo web hasta análisis de datos y aprendizaje automático".

En el trabajo de Lutz (2020) se afirma que "Python es un lenguaje de programación de propósito general que se ha vuelto cada vez más popular en los últimos años debido a su simplicidad, flexibilidad y potencia" un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning (ML).

Para este proyecto se empleó Python un lenguaje de programación eficiente, versátil y fácil de aprender, se puede ejecutar en muchas plataformas diferentes, conocido por su sintaxis clara y legible, lo que agrega valor al mantenimiento y expansión del código a largo plazo. Además, porque es el lenguaje en que está escrito Django el cual fue donde se desarrolló el servidor.

1.3.3 Tecnologías

Las tecnologías se refieren al conjunto de herramientas, técnicas, métodos y conocimientos aplicados para resolver problemas o satisfacer necesidades. Según Brynjolfsson y McAfee (2014), "las tecnologías son el conjunto de técnicas, habilidades, métodos y procesos utilizados en la producción de bienes o servicios o en el logro de objetivos, como investigación científica". A continuación, se presentan las tecnologías utilizados en la implementación del sistema.

Flutter: 3.19 es un marco de código abierto desarrollado y compatible con Google. Los desarrolladores de front-end y pila completa utilizan Flutter para crear una interfaz de usuario (IU) de aplicación para varias plataformas con un único código base. Cuando Flutter se lanzó, en 2018, era compatible principalmente con el

desarrollo de aplicaciones móviles. Ahora, Flutter es compatible con el desarrollo de aplicaciones en seis plataformas: iOS, Android, web, Windows, MacOS y Linux.

Flutter utiliza el lenguaje de programación de código abierto Dart, que también desarrolló Google. Dart está optimizado para la creación de interfaces de usuario, y muchos de los puntos fuertes de Dart se utilizan en Flutter.

El objetivo de Google es que Flutter sea fácil de usar. Con herramientas como la recarga en caliente, los desarrolladores pueden previsualizar el aspecto de los cambios de código sin perder el estado. Otras herramientas, como el inspector de widgets, facilitan la visualización y la resolución de problemas con los diseños de la interfaz de usuario.

(¿Qué es Flutter?, s. f.)

Se implementó Flutter para el desarrollo de la aplicación para dispositivos móviles porque permite crear interfaces ricas y dinámicas utilizando widgets personalizados y predefinidos, lo que facilita la creación de experiencias de usuario consistentes en múltiples plataformas.

Django: 5.0.6 es un software que puede utilizar para desarrollar aplicaciones web de forma rápida y eficiente. La mayoría de las aplicaciones web tienen varias funciones comunes, como la autenticación, la recuperación de información de una base de datos y la administración de cookies. Los desarrolladores tienen que codificar una funcionalidad similar en cada aplicación web que escriban.

Django facilita su trabajo al agrupar las diferentes funciones en una gran colección de módulos reutilizables, llamada marco de aplicación web. Los desarrolladores utilizan el marco web de Django para organizar y escribir su código de manera más eficiente y reducir significativamente el tiempo de desarrollo web. Hay varios marcos web en el mercado. Django se escribió en el lenguaje Python y es uno de los muchos marcos web de Python (¿Qué es Django?, s. f.).

El desarrollo del servidor se basó en Django, ya que facilita el desarrollo rápido de aplicaciones web complejas. Además, Django se caracteriza por su arquitectura modular, que permite separar la lógica de negocio de la interfaz de usuario, y su

potente ORM (Object-Relational Mapping) que simplifica la interacción con bases de datos SQL.

1.4 Arquitectura MVC (Modelo-Vista-Controlador).

Según Sommerville (2016), "la arquitectura MVC es un patrón de diseño que divide una aplicación en tres capas interconectadas: Modelo, Vista y Controlador, lo que permite una mejor separación de responsabilidades y facilita el mantenimiento y la escalabilidad de la aplicación". Las tres partes del patrón de diseño de software MVC se pueden describir de la siguiente manera:

Modelo

El modelo define qué datos deben contener la aplicación. Si el estado de estos datos cambia, el modelo generalmente notificará a la vista (para que la pantalla pueda cambiar según sea necesario) y, a veces, el controlador (si se necesita una lógica diferente para controlar la vista actualizada). En este sentido, el modelo especificará qué datos deben contener los artículos de la lista y qué artículos de la lista ya están presentes.

Vista

La vista define cómo se deben mostrar los datos de la aplicación. En esta aplicación de lista de muebles del hogar, la vista definiría cómo se presenta la lista al usuario y recibiría los datos para mostrar desde el modelo.

Controlador

El controlador contiene una lógica que actualiza el modelo y/o vista en respuesta a las entradas de los usuarios de la aplicación. En correspondencia con lo anterior, nuestra lista de muebles del hogar podría tener botones que nos permitan visualizar detalles del producto. Estas acciones requieren que se actualice el modelo, por lo que la entrada se envía al controlador, que luego manipula el modelo según corresponda y luego envía datos actualizados a la vista.

Sin embargo, es posible que también se desee actualizar la vista para mostrar los datos en un formato diferente, por ejemplo, cambiar el orden de los artículos de

menor a mayor precio o en orden alfabético. En este caso, el controlador podría manejar esto directamente sin necesidad de actualizar el modelo.

(MVC - Glosario de MDN Web Docs: Definiciones de términos relacionados con la Web | MDN, s. f.)

Se destaca la importancia de la arquitectura MVC (Modelo-Vista-Controlador) debido a su capacidad para separar responsabilidades, facilitando la gestión y el mantenimiento del código. Esta arquitectura permite que diferentes desarrolladores trabajen en paralelo en el modelo, la vista y el controlador, acelerando el desarrollo. Además, promueve la reutilización de componentes y mejora la escalabilidad de las aplicaciones, permitiendo agregar nuevas funcionalidades sin reescribir grandes partes del código. También facilita las pruebas unitarias y de integración, mejorando la calidad del software. Estas características hacen que el patrón MVC sea una elección popular para el desarrollo de aplicaciones robustas y escalables. (Freeman & Robson, 2004)

La elección de la arquitectura MVC (Modelo-Vista-Controlador) en el desarrollo de la aplicación móvil que se conecta a un servidor se fundamenta en varias razones clave. En primer lugar, MVC permite una clara separación de responsabilidades, dividiendo la aplicación en tres componentes distintos. Esta separación facilita el mantenimiento y la escalabilidad del sistema, ya que cada componente puede ser desarrollado, probado y actualizado de manera independiente sin afectar a los demás.

Además, la arquitectura MVC promueve la reutilización de código y la modularidad, lo que resulta en un desarrollo más eficiente y organizado. MVC permite manejar de manera efectiva las solicitudes y respuestas entre el cliente y el servidor, asegurando una experiencia de usuario fluida y una gestión óptima de los datos.

1.5 Metodología de desarrollo de software XP.

Según Duvall y Matyas (2017), "XP es una metodología ágil de desarrollo de software que se basa en valores como la comunicación, la simplicidad, el feedback y el coraje, y que enfatiza prácticas como el desarrollo guiado por pruebas, la programación en parejas y el despliegue frecuente de software funcional".

Características del Extreme Programming XP

- ✓ *Comunicación constante entre el cliente y el equipo de desarrollo.*
- ✓ Respuesta rápida a los cambios constantes.
- ✓ La planificación es abierta con un cronograma de actividades flexible.
- ✓ El software que funciona está por encima de cualquier otra documentación.
- ✓ Los requisitos del cliente y el trabajo del equipo del proyecto son los principales factores de éxito del mismo.

FASES del Extreme Programming XP

FASE 1: PLANIFICACIÓN

Según la identificación de las historias de usuario, se priorizan y se descomponen en mini-versiones. La planificación se va a ir revisando. Cada dos semanas aproximadamente de iteración, se debe obtener un software útil, funcional, listo para probar y lanzar.

FASE 2: DISEÑO

En este paso se intentará trabajar con un código sencillo, haciendo lo mínimo imprescindible para que funcione. Se obtendrá el prototipo. Además, para el diseño del software orientado a objetos, se crearán tarjetas CRC (Clase-Responsabilidad-Colaboración).

FASE 3: CODIFICACIÓN «DE TODOS»

La programación aquí se hace «a dos manos», en parejas en frente del mismo ordenador. Incluso, a veces se intercambian las parejas. De esta forma, se asegura que se realice un código más universal, con el que cualquier otro programador podría trabajar y entender. Y es que debe parecer que ha sido realizado por una única persona. Así se conseguirá una programación organizada y planificada.

FASE 4: PRUEBAS

Se deben realizar pruebas automáticas continuamente. Al tratarse normalmente de proyectos a corto plazo, este testeo automatizado y constante es clave. Además, el

propio cliente puede hacer pruebas, proponer nuevas pruebas e ir validando las mini-versiones.

FASE 5: LANZAMIENTO

Si se ha llegado a este punto, significa que se han probado todas las historias de usuario o mini-versiones con éxito, ajustándose a los requerimientos del cliente. Se dispone de un software útil el cual puede incorporarse al producto.

(Metodología XP o Programación Extrema, s. f.)

Ventajas del Extreme Programming XP

El Extreme Programming XP ofrece varias ventajas que pueden impulsar el éxito del desarrollo de software. A continuación, se presentan algunas de las ventajas más destacadas:

1. Entrega temprana de software funcional.
2. Adaptabilidad a los cambios.
3. Comunicación y colaboración efectivas.
4. Calidad y eficiencia mejoradas.
5. Retroalimentación constante del cliente.
6. Mayor satisfacción del cliente.

Desventajas del Extreme Programming XP

Aunque el Extreme Programming (XP) ofrece numerosas ventajas, también presenta desafíos y desventajas que deben tenerse en cuenta. Estas son algunas de las desventajas más comunes asociadas con el XP:

1. Dependencia de la comunicación constante.
2. Requiere una cultura de colaboración.
3. Requiere una planificación y organización cuidadosas.
4. Dificultad para proyectos grandes y complejos.
5. Falta de documentación exhaustiva.

6. Resistencia al cambio.

(*Extreme Programming XP*, 2023)

1.6 Conclusiones del Capítulo.

La revisión de la literatura especializada en aplicaciones de realidad aumentada para la visualización de muebles en dispositivo móviles permitió establecer un marco teórico sólido para respaldar el diseño y desarrollo de la aplicación. Además, el diagnóstico aplicado reveló la necesidad de desarrollar una aplicación que ofrezca una experiencia inmersiva y realista para los usuarios. En cuanto a las herramientas de desarrollo, Visual Studio Code y Git se presentan como opciones sólidas para la escritura de código y el control de versiones, respectivamente. MySQL se selecciona como la base de datos debido a su robustez, flexibilidad y amplia comunidad de soporte. Para el desarrollo de la aplicación se empleó Flutter debido a que permite crear interfaces ricas y dinámicas; y Django para el desarrollo del servidor por su arquitectura modular, que permite separar la lógica de negocio de la interfaz de usuario, y su potente ORM (Object-Relational Mapping) que simplifica la interacción con la base de datos MySQL. Además, se tomó la arquitectura MVC (Modelo-Vista-Controlador) en el desarrollo de la aplicación móvil que se conecta a un servidor gracias a que permite una clara separación de responsabilidades y promueve la reutilización de código y la modularidad, lo que resulta en un desarrollo más eficiente y organizado. Finalmente, se enfatiza que se utilizó la Metodología de Desarrollo de Software XP por su enfoque en la entrega rápida y de alta calidad, priorizando la comunicación constante y la adaptabilidad.

Capítulo 2. Organización y Diseño.

Este capítulo está orientado a describir los elementos fundamentales del diseño del sistema a desarrollar, tales como: la propuesta del sistema y sus usuarios, roles y responsabilidades identificados. Se declaran las funcionalidades del sistema, así como las historias técnicas y las historias de usuario de las funcionalidades más importantes que se implementan en el software. También se aborda los elementos a tener en cuenta para su implementación. Se presenta el modelo físico de la base de datos y la arquitectura empleada en la construcción del sistema.

2.1 Propuesta del Sistema

Ante el problema de investigación planteado, se propone desarrollar un sistema informático en forma de aplicación para dispositivos móviles que permita al cliente visualizar los muebles antes de comprarlos, sin necesidad de salir de sus casas y no malgastar el dinero en muebles que no se ajustan a sus necesidades.

Además, se desarrolló el servidor donde las empresas que deseen publicar sus productos para la venta pueden agregarlos. La aplicación se conecta a dicho servidor para visualizar todos los productos cargados por las empresas desde el mismo.

2.2 Objetivo a alcanzar

Una vez instalada la aplicación para dispositivos móviles el usuario podrá visualizar los muebles que hay en venta y apreciar información como categorías, precios y nombres. Tendrá además una interfaz con descripción de los mismos y también un botón que será el encargado de mostrarlos modelados 3D, como si fuera el real.

2.3 Flujo del Sistema

El funcionamiento interno de la app se basa en la arquitectura Modelo-Vista-Controlador (MVC). Esta arquitectura separa la lógica de negocio, el modelo y la presentación, por lo que se consigue un mantenimiento sencillo de las aplicaciones. El usuario solicita acceder al sistema. Internamente el procedimiento de enrutamiento determina qué controlador está asociado con la interfaz de la portada. El controlador envía datos al modelo y los obtiene dependiendo de la solicitud del

usuario. Con los datos devueltos por el modelo, el controlador envía a la vista los datos del modelo para mostrarlos al usuario en la interfaz principal. En la figura 2.1 se presenta el funcionamiento del patrón MVC.

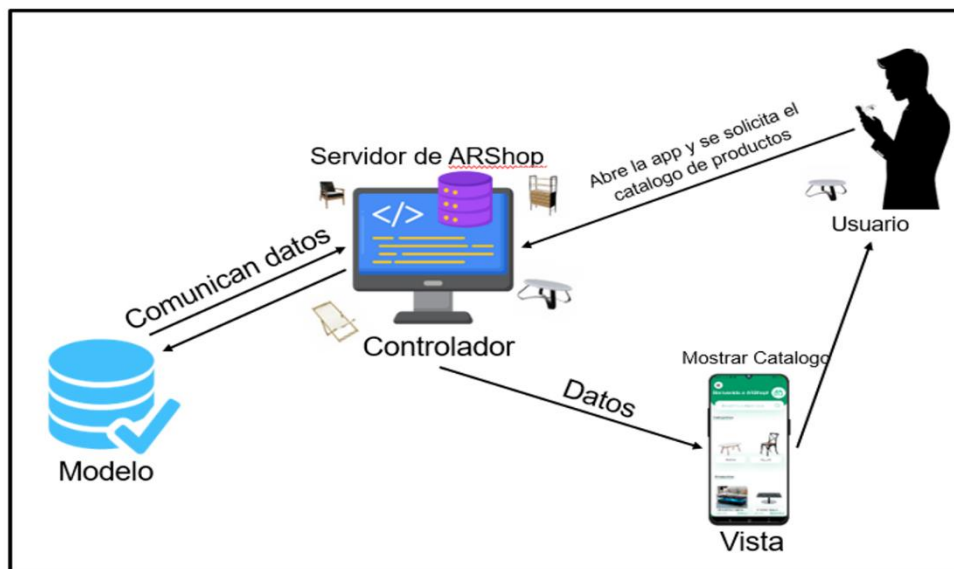


Figura 2.1 Funcionamiento del Patrón MVC.

2.3.1 Flujo del Sistema para la visualización de los muebles con RA:

En la figura 2.2 se muestra el diagrama de flujo para el marco de trabajo que permite visualizar el mueble usando la realidad aumentada. Respecto a la información que se muestra en la figura, el proceso de visualización comienza cuando el usuario entra a la aplicación y ve el catálogo de productos (Paso 3), se selecciona el mueble deseado donde se muestra la información del mismo (Paso 4) y si se quiere visualizar se presiona en el botón que hace esto posible (Paso 5), luego se abre la cámara del dispositivo para mostrar el modelo 3D del mueble seleccionado (Paso 6).

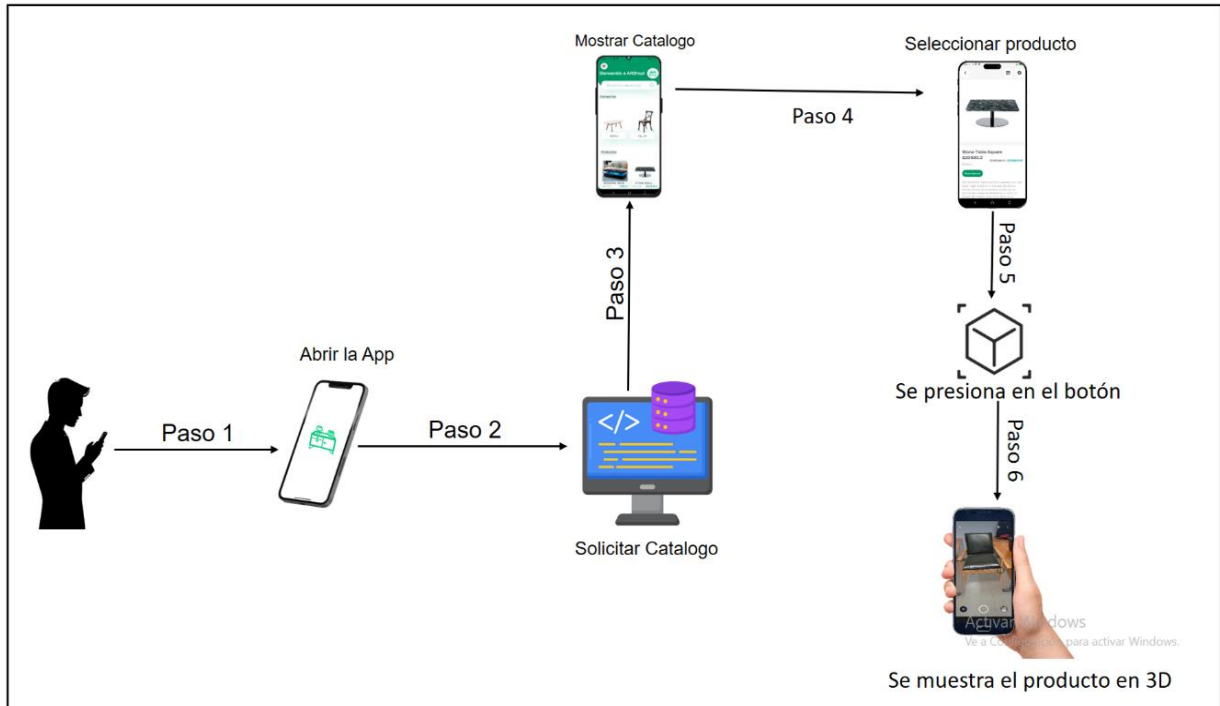


Figura 2.2 Diagrama de flujo para la visualización del producto con RA

2.4 Usuarios del sistema

Los usuarios del sistema constituyen un aspecto crítico del mecanismo de control de acceso a las aplicaciones, especialmente cuando se considera la seguridad y el cumplimiento. La implementación de estrategias adecuadas de control de acceso basadas en roles disminuye los riesgos potenciales de acceso no autorizado y violaciones de datos, garantiza el cumplimiento de las regulaciones pertinentes y, en última instancia, contribuye a la estabilidad y el éxito de la solución de software general (*Roles de usuario | AppMaster, s. f.*).

A continuación, una tabla con los diferentes usuarios del sistema:

Tabla 2.1 Usuarios del sistema

Actor	Tareas que realiza
Cliente	Listar, buscar, visualizar productos 3D.

Administrador	Crear, editar, eliminar productos categorías o entidades.
---------------	---

2.5 Requerimientos para el despliegue de la App

Tanto los requisitos de software como los de hardware desempeñan un papel fundamental en el desarrollo e implementación de aplicaciones para dispositivos móviles Android. Para garantizar el funcionamiento óptimo de la aplicación objeto de esta investigación, es esencial tener en cuenta una serie de especificaciones técnicas que abarcan desde la versión mínima del sistema operativo Android necesarias, hasta las capacidades de memoria RAM, procesador, espacio de almacenamiento y resolución de pantalla requeridas en los dispositivos móviles. Asimismo, se deben considerar las bibliotecas, APIs y otras dependencias específicas de Android que se utilizarán. Este epígrafe se centra en la exposición de los requisitos de software y hardware indispensables para la instalación y funcionamiento exitoso de la aplicación en la amplia variedad de dispositivos móviles Android disponibles en el mercado.

Requisitos de Software

➤ Sistema Operativo: Varía según el dispositivo, pero recomendado Android v9 o superior.

Requisitos de Hardware

Los requerimientos mínimos de hardware para correr la aplicación:

- 1 Gb de memoria RAM.
- Conexión a Internet.
- Buen espacio de almacenamiento libre. (al menos 1 Gb)

2.6 Historias de Usuarios

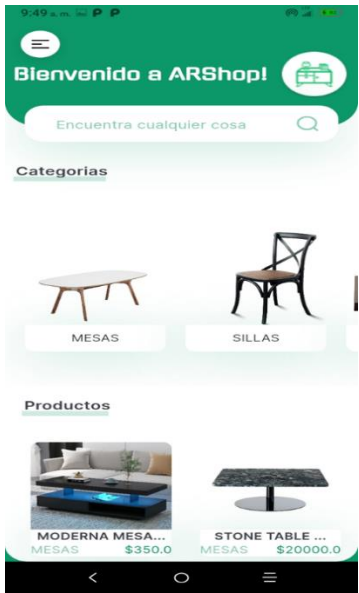
Una historia de usuario es una explicación general e informal de una función de software escrita desde la perspectiva del usuario final o cliente. El propósito de una historia de usuario es articular cómo un elemento de trabajo entregará un valor

particular al cliente. Son además unas pocas frases en lenguaje sencillo que describen el resultado deseado. No entran en detalles, ya que los requisitos se añaden más tarde, una vez acordados por el equipo (Atlassian, s. f.).

A continuación, se muestran de forma más detallada y con su respectivo diseño algunas de las principales historias de usuario.

Interfaz Principal: Aquí podrá ver todas las categorías donde puede acceder a cada una si lo desea para ver los productos de la misma clasificados; además en la interfaz podrá visualizar también todos los productos que se hallan subido al servidor y si presiona en ellos se dirigirá a la interfaz de dicho producto.


Tabla 2.2 Interfaz Principal

Historia de Usuario	
Número: 3	Nombre de HU: Interfaz Principal
Prioridad en negocio: Alta	Riesgo de Desarrollo: Bajo
	

Interfaz Producto: En dicha interfaz se muestra toda la información del producto como nombre, categoría, distribuidor, precio, etc. y además cuenta con tres botones

principales los cuales son el de visualizar en 3D, el que genera un código QR del producto y del distribuidor el cual lo direcciona a una interfaz donde visualiza más información de dicho distribuidor.

Tabla 2.3 Interfaz Producto

Historia de Usuario	
Número: 4	Nombre de HU: Interfaz Producto
Prioridad en negocio: Alta	Riesgo de Desarrollo: Bajo
	

Visualizar Producto mediante la Realidad Aumentada: El usuario podrá visualizar el producto que desee presionando en el botón en forma de cubo el cual es el encargado de abrir la cámara del dispositivo móvil y mostrar el modelo 3D de dicho producto.

Tabla 2.4 Visualizar Producto mediante la Realidad Aumentada

Historia de Usuario	
Número: 5	Nombre de HU: Visualizar Producto mediante la Realidad Aumentada
Prioridad en negocio: Alta	Riesgo de Desarrollo: Bajo
	

2.7 Diagrama entidad-relación para la persistencia en la base de datos

Un diagrama de entidad-relación (ER) es una representación gráfica utilizada en el diseño de bases de datos. Este tipo de diagrama es una herramienta fundamental en el modelado de datos, ya que permite visualizar las entidades relevantes en un sistema, sus atributos y las relaciones entre ellas. Los diagramas de entidad-relación son vitales en el diseño y desarrollo de bases de datos, ya que permiten modelar de manera lógica y estructurada la información, facilitando la identificación de entidades, atributos y relaciones. Además, sirven como guía para diseñar bases de datos eficientes, definiendo tablas, campos y claves, y aseguran la integridad de los datos contribuyendo a la optimización y mantenimiento al identificar y corregir redundancias e incoherencias en el diseño (Besteiro, 2022).

La estructura de la base de datos del servidor, que se conecta a la aplicación móvil se ha diseñado con solo tres tablas: categoría, producto y entidad. Esta decisión se

basa en varios factores clave. En primer lugar, la simplicidad y eficiencia de una base de datos con menos tablas facilita su mantenimiento y gestión, mejorando así la eficiencia operativa. Las tablas de categoría, producto y entidad contienen toda la información esencial que la aplicación móvil necesita para funcionar correctamente, organizando los productos, almacenando sus detalles y gestionando la información de las entidades, respectivamente.

Finalmente, una base de datos bien definida con relaciones claras entre las tablas asegura la integridad de los datos, facilitando el desarrollo y la implementación de nuevas características en la aplicación móvil. Adicionalmente, se garantiza la persistencia de datos asegurando que la información se mantenga disponible y consistente a lo largo del tiempo, incluso ante posibles fallos del sistema o interrupciones en el servicio. A continuación, dicho diseño de base de datos:

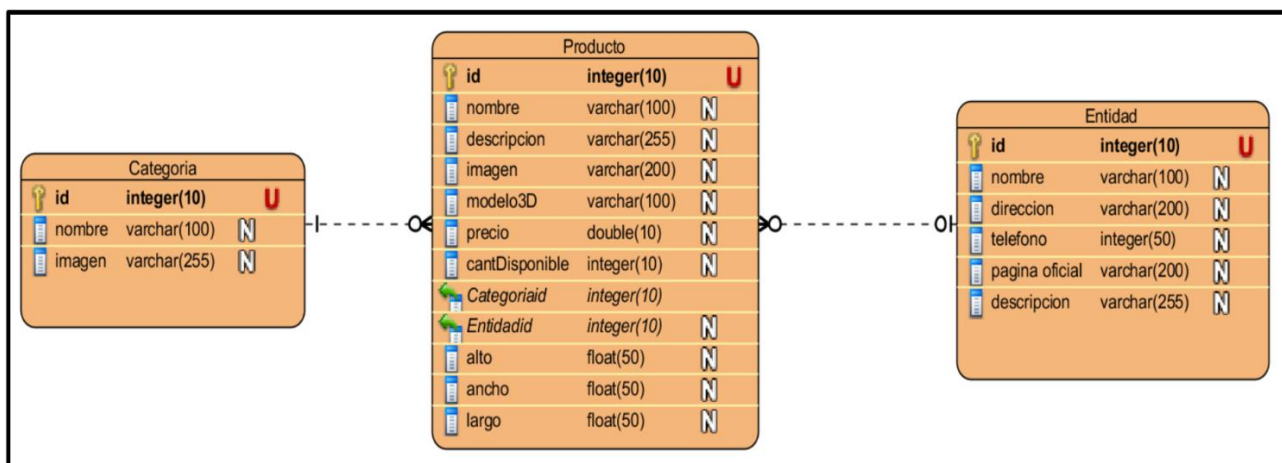


Figura 2.3 Diagrama de base de datos

2.8 Diagrama de Secuencia

Aunque la metodología de Programación Extrema (XP) no requiere explícitamente el uso de diagramas UML, la implementación de un diagrama de secuencia puede justificarse por varias razones. Los diagramas de secuencia ayudan a visualizar y entender las interacciones entre los objetos en un proceso específico, lo que mejora la comunicación y la claridad entre los miembros del equipo y los clientes. Además, estos diagramas pueden ser útiles durante la fase de diseño inicial para asegurar

una comprensión del flujo del sistema. A continuación, el diagrama de secuencia para la visualización de muebles con RA:

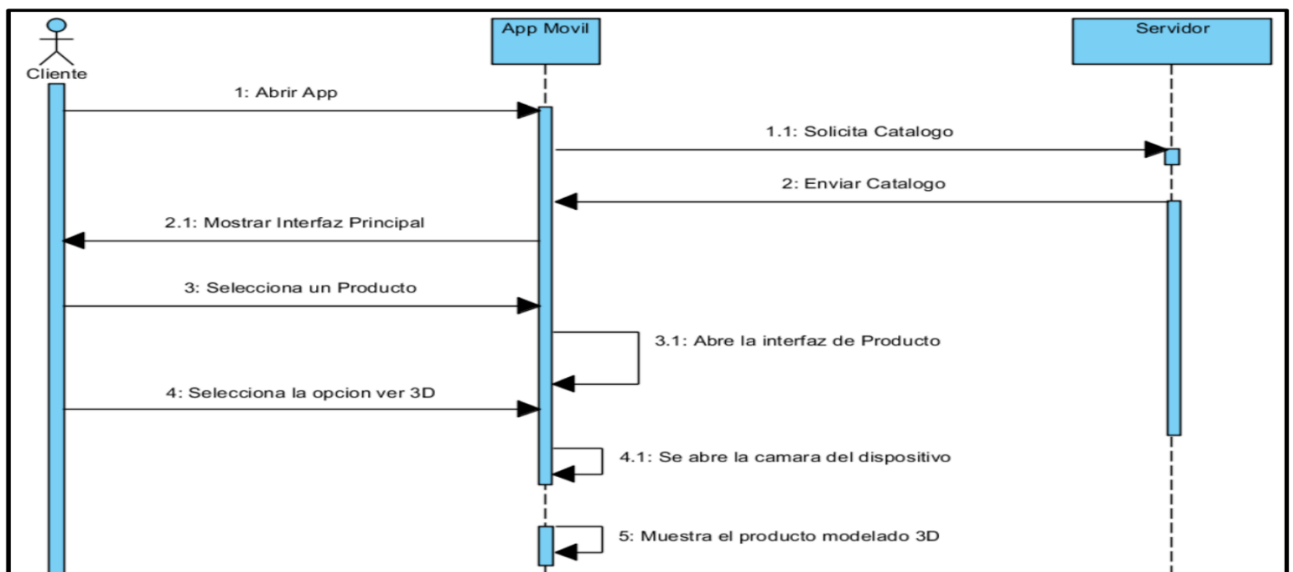


Figura 2.4 Diagrama de secuencia para visualizar muebles con RA

2.9 Diagrama de Actividades

En el contexto de la metodología XP (Extreme Programming), el uso de un diagrama de actividades puede justificarse por varias razones. Primero, proporciona claridad en el flujo de trabajo y las interacciones entre diferentes partes del sistema. Además, mejora la comunicación, un valor fundamental en XP, al servir como una herramienta visual que facilita la comprensión y discusión de los procesos entre los desarrolladores y otros interesados. También permite identificar “cuellos de botella” o áreas de mejora en el proceso, alineándose con el enfoque de mejora continua de XP. Finalmente, aunque XP prioriza el software funcional sobre la documentación extensa, un diagrama de actividades ofrece una documentación visual mínima pero suficiente, que puede ser muy útil sin ser excesiva. A continuación, el diagrama de actividades para visualización de muebles con RA.

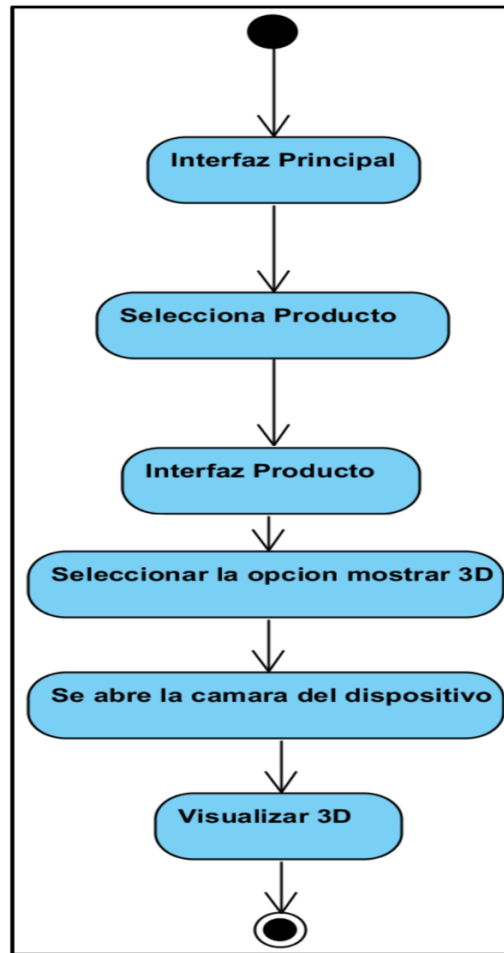


Figura 2.5 Diagrama de actividades para visualizar muebles con RA

2.10 Conclusiones del capítulo

En este capítulo, se muestra el diseño de la aplicación para dispositivos móviles, utilizando una arquitectura MVC dividida en capas para facilitar la integración de nuevos algoritmos y mejoras constantes. Se presentaron historias de usuario e ilustraciones de diseño para proporcionar una idea clara de los resultados deseados. También se incluyeron los requisitos técnicos, el diseño de la base de datos y varios diagramas (flujo, secuencia y actividades) para documentar adecuadamente el proyecto. Lo que proporciona a los clientes una experiencia de compra más conveniente y atractiva, fomentando una mayor interacción lo que podría traducirse en un aumento de las ventas y una mayor satisfacción del cliente.

Capítulo 3. Implementación y Prueba.

Este capítulo contiene los detalles más importantes del análisis de la construcción de la solución propuesta, donde se presentarán las funcionalidades del sistema diseñado y detalles de su implementación. Se describe como se aplicó la filosofía del Flutter al sistema utilizando las herramientas del software y evaluando su correcto funcionamiento. También se procede en la traducción del diseño al código fuente, necesaria para el funcionamiento de la aplicación. Con el objetivo de comprobar el cumplimiento de los requisitos funcionales iniciales se realizaron las pruebas al software, permitiendo comprobar la calidad del producto propuesto y se expondrán los resultados de los casos de prueba que se llevaron a cabo una vez terminado el proyecto.

3.1 Dependencias utilizadas.

model_viewer_plus: Este es un widget de Flutter para renderizar modelos 3D interactivos en los formatos glTF y GLB. El widget incrusta el componente web de Google <model-viewer> en un WebView. Compatible con Android, iOS, Web, con una versión reciente del navegador del sistema. Tiene funciones como:

- Renderiza modelos glTF y GLB.
- Soporta modelos animados, con una configuración configurable de reproducción automática.
- Opcionalmente, admite el lanzamiento del modelo en un visor de realidad aumentada.

(*Model_viewer_plus* | *Flutter Package*, s. f.)

Rive: Es una biblioteca en tiempo de ejecución para Rive, que es una herramienta de animación y diseño interactivo en tiempo real. Esta biblioteca le permite controlar completamente los archivos Rive con una API de alto nivel para interacciones y animaciones simples, así como una API de bajo nivel para crear bucles de renderizado personalizados para múltiples mesas de trabajo, animaciones y máquinas de estado en un solo lienzo. Además, ayuda a los equipos a crear y ejecutar animaciones interactivas para aplicaciones, juegos y sitios web. Los

diseñadores y desarrolladores pueden usar el editor colaborativo para crear gráficos en movimiento que respondan a diferentes estados y entradas del usuario, y luego usar bibliotecas livianas en tiempo de ejecución de código abierto, como Rive Flutter, para cargar sus animaciones en sus proyectos (*Rive | Flutter Package*, s. f.).

hexcolor: El paquete Hexcolor es una herramienta simple pero poderosa que te ayuda a aprovechar las ventajas de los colores hexadecimales mientras desarrollas aplicaciones Flutter. Con el complemento Hexcolor para Flutter, puede usar directamente cadenas de colores hexadecimales en su aplicación, sin preocuparse por la conversión ni por la codificación adicional para la personalización.

El paquete Hexcolor es una gran ayuda para los desarrolladores que usan Flutter porque incorpora la práctica generalizada de definir colores usando cadenas hexadecimales a su flujo de trabajo de Flutter.

(*Mastering Flutter Hex Color*, s. f.).

http: Una biblioteca componible basada en Future para realizar solicitudes HTTP. Este paquete contiene un conjunto de funciones y clases de alto nivel que facilitan el consumo de recursos HTTP. Es multiplataforma (móvil, escritorio y navegador) y admite múltiples implementaciones. La forma más sencilla de utilizar esta biblioteca es a través de las funciones de nivel superior. Le permiten realizar solicitudes HTTP individuales con mínimas molestias.

(*Http | Dart Package*, s. f.).

3.2 Diagrama de clases

El Diagrama de Clases es una representación visual de la estructura estática de un sistema de software. Es un tipo de diagrama perteneciente a la UML (Unified Modeling Language) y se utiliza para modelar los elementos que componen un sistema, como las clases, interfaces, asociaciones, herencias, entre otros. El uso del Diagrama de Clases en la ingeniería de software proporciona diversos beneficios, entre los que destacan:

1. Comunicación efectiva: el diagrama permite una comunicación clara y concisa entre los miembros del equipo de desarrollo, ya que brinda una representación visual del sistema y facilita la comprensión de su estructura.

2. Diseño estructurado: el diagrama ayuda a organizar y estructurar el diseño del sistema, permitiendo identificar y definir las clases, atributos y métodos necesarios para su implementación.

3. Identificación de problemas: al analizar el diagrama, es posible identificar posibles problemas o incoherencias en el diseño del sistema, lo que facilita su corrección antes de la implementación.

(Descubre La Importancia Del Diagrama De Clases En Ingeniería De Software | 2024, s. f.)

A continuación, en la figura 3.1 se representa el diagrama de clases del sistema para el REN y la generación de datos de entrenamiento. Donde se puede apreciar las clases Categorías, Productos y Entidad. Las relaciones que existen entre ellos son:

Relaciones:

- Categoría - Producto: Donde una Categoría puede contener uno o más productos, pero un producto solo está contenido en una categoría.
- Entidad - Producto: Donde una entidad puede distribuir uno o más productos. Además, es una composición debido a que si no hay distribuidores entonces no puede haber productos en venta.

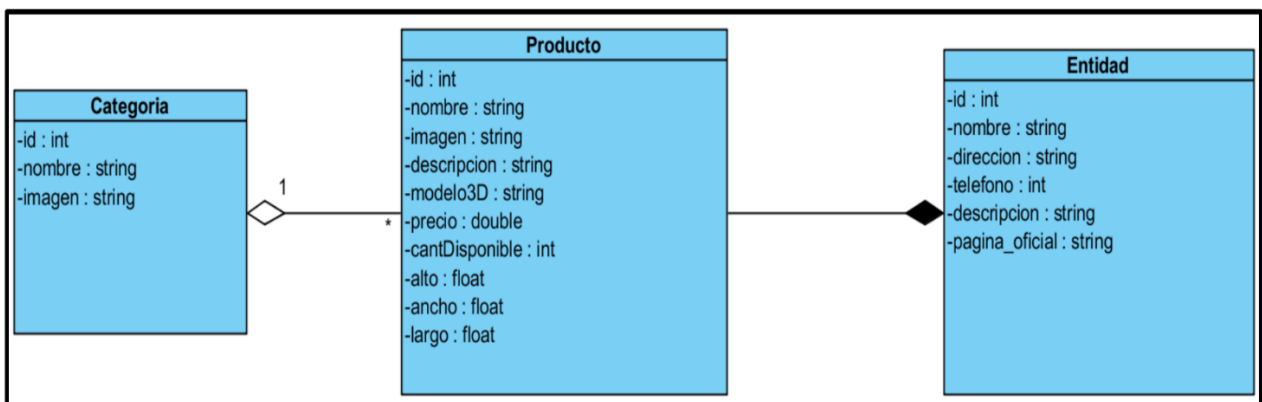


Figura 3.1 Diagrama de clases

3.3 Algoritmos importantes.

En el siguiente epígrafe se realizará una explicación de algunos de los algoritmos más importantes de la aplicación.

3.3.1 Algoritmo para mostrar el catálogo de productos

El ejemplo de código siguiente muestra como la aplicación para dispositivos móviles recibe desde el servidor el catálogo de productos donde se define la *url* a la que se va a conectar mediante la variable *uri* después se recibe la respuesta en formato *json* y se convierte al modelo *Producto* para después mostrarlo en la interfaz.

```
var productoEndpoint = Uri.parse('$baseUrl/producto');

// Mostrar Todos los Productos
Future<List<Producto>> getListOfProducto() async {
  var uri = Uri.parse('$productoEndpoint/getListOfProducto/');

  var res = await http.get(uri);

  if (res.statusCode == 200) {
    List<Producto> productos = (json.decode(res.body) as List)
      .map((data) => Producto.fromJson(data))
      .toList();

    for (var producto in productos) {
      producto.nombre = utf8.decode(producto.nombre.codeUnits);
      producto.descripcion = utf8.decode(producto.descripcion.codeUnits);
    }

    return productos;
  } else {
    throw Exception('Fallo al cargar');
  }
}
```

Figura 3.2 Algoritmo para mostrar el catálogo de productos

3.3.2 Algoritmo para la visualización del producto en 3D

El ejemplo de código siguiente muestra la clase encargada para mostrar el modelo 3D del mueble elegido mediante la cámara del dispositivo dicha clase es *ModelViewer* a la cual se le ponen los parámetros que se muestran, pero los

fundamentales son la *url* del producto y activar la función *ar* en *true* la cual hace todo el proceso.

```
class _ARModelViewerState extends State<ARModelViewer> {  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        centerTitle: true,  
        title: Text(  
          widget.producto.nombre,  
        ), // Text  
      ), // AppBar  
      body: ModelViewer(  
        src: widget.producto.modelo3dUrl,  
        alt: "A 3D model of an astronaut",  
        ar: true,  
        autoRotate: true,  
        cameraControls: true,  
        // arScale: ,  
      ), // ModelViewer  
    ); // Scaffold  
  }  
}
```

Figura 3.3 Algoritmo para visualizar el producto con RA

3.4 Pruebas al sistema

Las pruebas de software son un conjunto de procesos con los que se pretende probar un sistema o aplicación en diferentes momentos para comprobar su correcto funcionamiento. Este tipo de pruebas abarca cualquier estadio del desarrollo del sistema, desde su creación hasta su puesta en producción. Lo interesante de las pruebas es que se puedan ejecutar de manera automática, para determinar en cualquier momento si tenemos una aplicación estable o si, por el contrario, un cambio en una parte ha afectado a otras partes sin que nos demos cuenta (Turrado, 2020).

3.4.1 Pruebas a la aplicación para dispositivos móviles.

Casos de prueba ejecutados:

Tabla 3.1 Caso de Prueba “Buscar un producto por el nombre del mismo”

CP	Entrada	Resultado esperado	Condición
CP1	Nombre del producto correcto	El sistema muestra el producto buscado	El nombre del producto es correcto
CP2	Nombre del producto erróneo	El sistema muestra un mensaje de error	El nombre del producto es incorrecto o el producto no está disponible

Pruebas en entornos reales

Otras de las pruebas realizadas al sistema fueron en dispositivos reales. Ellas son posiblemente las más importantes debido a que se evalúa el comportamiento del sistema en entornos reales y se verifica la aceptación por parte de los usuarios. La aplicación se probó en una serie de teléfonos inteligentes con resultados exitosos. Estas pruebas evidenciaron que para poder visualizar los muebles en 3D con la realidad aumentada se debe tener integrado el servicio de Google Play para RA como fue en el caso del Samsung S21; sin embargo, el Redmi 10S no tenía este servicio integrado pero la aplicación le redirigió a Play Store para instalar este servicio en forma de aplicación bajo el nombre “servicios de Google Play para RA”. Además, se probó que en otros dispositivos como el Alcatel3X y el TecnoBF7 este servicio tampoco estuvo integrado y además tampoco fue soportada la app de servicios de Google Play para RA como se muestra en la figura 3.4 de los anexos por lo que no pudo ser posible usar esta función.

No obstante, se pasaron todas las pruebas de aceptación, el sistema mantuvo buen desempeño y funcionamiento. Demostró ser estable y fácil de utilizar. Todos los dueños de los teléfonos en los que se probó la aplicación manifestaron su conformidad. A continuación (ver la siguiente tabla) se muestran las características de los teléfonos en los que se hicieron las pruebas.

Tabla 3.2 Listado de dispositivos en las pruebas en entorno reales.

Dispositivo	Sistema Operativo	Memoria (RAM)
Alcatel 3X	9	4 GB
Redmi 10C	11	4 GB
Samsung S21	13	4 GB
Tecno BF7	12	4 GB
Redmi 10S	12	4 GB

3.4.2 Pruebas al servidor

Al servidor también se le hicieron debido a que este es el encargado de añadir los productos y los datos de las entidades al sistema para luego visualizarlos en la app; a continuación, los casos de pruebas ejecutados:

Casos de prueba ejecutados:

Tabla 3.3 Caso de Prueba “Inicio de Sesión”

CP	Entrada	Resultado esperado	Condición
CP1	Usuario y contraseñas válidos.	El sistema redirecciona a la interfaz principal del servidor.	El usuario y la contraseña son correctos

CP2	Usuario o contraseñas inválidos.	El sistema muestra un mensaje de error	Por favor introduzca el nombre y la clave correctos para una cuneta de personal. Observe que ambos son sensibles a mayúsculas
-----	----------------------------------	--	---

Tabla 3.3 Caso de Prueba “Ingresar Producto”

CP	Entrada	Resultado esperado	Condición
CP1	Datos y archivos del producto validos	El sistema agrega correctamente el producto y muestra un mensaje de éxito	Se presiona el botón guardar con todos los datos correctos
CP2	Falta algún dato o archivo para agregar el producto	El sistema muestra un mensaje de error	No dejar ningún campo vacío
CP3	El formato de archivo del modelo 3D es erróneo	El sistema muestra un mensaje de error.	El formato de archivo debe ser glb o gltf

Tabla 3.4 Caso de Prueba “Ingresar Datos del Distribuidor”

CP	Entrada	Resultado esperado	Condición
CP1	Datos y archivos del distribuidor validos	El sistema agrega correctamente al distribuidor y muestra un mensaje de éxito	Se presiona el botón guardar con todos los datos correctos
CP2	Falta algún dato o archivo para agregar a la entidad	El sistema muestra un mensaje de error	No dejar ningún campo vacío

3.5 Análisis económico del costo de producción del sistema

En la realización de un proyecto se hace necesaria la planificación y el control del esfuerzo, costo y tiempo que tomará llevarlo a cabo. Con la utilización de métodos de estimación de costos, se puede determinar una aproximación de los recursos necesarios, así como el total de tiempo que gastaría una persona o un equipo, en el desarrollo de un producto de software específico. Además, se determina la viabilidad económica, ambiental, técnica y de mercado. A continuación, se realiza un análisis de costos para el sistema.

3.5.1 Estimación de costo y tiempo

En el desarrollo de una aplicación para dispositivos móviles con tecnología de realidad aumentada, la planificación y el control del esfuerzo, costo y tiempo son fundamentales.

El método de punto de casos de uso es una técnica utilizada en la gestión de proyectos de desarrollo de software para estimar el esfuerzo necesario y los recursos requeridos para llevar a cabo el proyecto. Consiste en cuatro etapas que involucran el cálculo de varios factores, incluyendo los puntos de historias de usuarios sin ajustar (UUCP), el factor de complejidad técnica (TCF), y el factor de ambiente (EF). En la primera etapa, se calculan los puntos de historias de usuarios sin ajustar considerando el peso de los actores. Luego,

se ajustan estos puntos considerando el TCF y el EF para obtener los puntos de historias de usuarios ajustadas (UCP).

A través de los UCP, se estima el esfuerzo en horas hombre para el proyecto. Finalmente, se distribuye este esfuerzo en distintas etapas del proyecto, y se calcula el costo total del proyecto en función de las tarifas horarias promedio y otros costos asociados. Este método permite una estimación detallada y precisa del esfuerzo y el costo involucrado en el desarrollo de un proyecto de software, lo que facilita la planificación y gestión efectiva del mismo.

En la realización del análisis económico ([ver Anexo 2](#)) se llegó a la conclusión que el costo total del proyecto sería aproximadamente de **\$ 225 511.36** en CUP.

En este análisis no se tuvo en cuenta los costos por electricidad e internet necesarios para el desarrollo del sistema desde el hogar.

3.6 Conclusiones del Capítulo.

En este capítulo, se concretó e implementó la aplicación móvil para la visualización de muebles y el servidor al que se conecta la app, incluyendo elementos fundamentales como listar los productos en venta, mostrar información de los distribuidores y visualizar los productos en 3D por parte de la aplicación; y otros elementos como agregar, editar y eliminar tanto productos como categorías por parte del servidor los cuales fueron mostrados correctamente desde la app. Además, las pruebas llevadas a cabo demostraron la capacidad del sistema para abordar diversas situaciones y verificar el rendimiento de las funcionalidades. Por último, se efectuó un análisis económico para calcular el costo del proyecto el cual fue de \$ 225 511.36 pesos cubanos, teniendo en cuenta aspectos como la complejidad técnica y los recursos humanos requeridos.

Conclusiones

En el transcurso de la investigación se dio solución a los objetivos específicos del proyecto, obteniéndose los siguientes resultados:

- ✓ Los referentes teóricos y el diagnóstico fáctico posibilitaron profundizar lo relacionado con las aplicaciones para dispositivos móviles con tecnología de Realidad Aumentada para la visualización de muebles, garantizando así la comprensión de las tecnologías y metodologías empleadas.
- ✓ Se realizó la correcta selección de las herramientas y metodologías adecuadas para el desarrollo del proyecto de investigación, proporcionando que la solución propuesta atienda las necesidades de los usuarios y el desarrollo exitoso de la aplicación para dispositivos móviles.
- ✓ Se diseñó e implementó una aplicación para dispositivos móviles que permite visualizar muebles con la tecnología de Realidad Aumentada. Además, se desarrolló un servidor a donde se conecta la app donde las empresas pueden subir sus productos para que el cliente los vea desde la app además de otra información de las mismas.
- ✓ Las pruebas tanto de la app como del servidor han validado la viabilidad de la solución propuesta, demostrando su capacidad para cargar productos al servidor y visualizarlos desde la app además de poder visualizarlos en 3D mediante la Realidad Aumentada.

Recomendaciones

En el desarrollo del proyecto se identificaron oportunidades para futuras investigaciones y mejoras en la aplicación como:

- ✓ La realización de evaluaciones periódicas de la aplicación para identificar problemas en su funcionamiento.
- ✓ La realización de un seguimiento y análisis del impacto social de la aplicación en la provincia de Santiago de Cuba para medir su contribución al desarrollo regional.
- ✓ Además de un apartado de valoración hacia la aplicación móvil para continuar perfeccionando el sistema, teniendo en cuenta las opiniones del cliente, a través de la integración de las nuevas tecnologías que permitan un funcionamiento rápido y seguro del mismo, así como una adecuada gestión de la información que se maneja.
- ✓ Implementar un sistema de compra online dentro de la aplicación móvil para que los usuarios puedan adquirir productos directamente desde la app, sin necesidad de recurrir a terceros. Esto no solo mejorará la experiencia del usuario al simplificar el proceso de compra, sino que también permitirá a la empresa tener un mayor control sobre las ventas y la satisfacción del cliente.
- ✓ Además, se puede aprovechar la tecnología de realidad aumentada de la aplicación para expandir su uso más allá de los muebles de hogar. En el futuro, la app podría mostrar otros tipos de muebles y productos, como artículos de oficina, decoración, electrodomésticos, y más. Esto diversificará la oferta y atraerá a un público más amplio, aumentando así las oportunidades de venta.

Referencias Bibliográficas

Acerca de Visual Paradigm. (s. f.). Recuperado 19 de mayo de 2024, a partir de

<https://online.visual-paradigm.com/es/about-us/>

Atlassian. (s. f.). *Historias de usuario | Ejemplos y plantilla.* Atlassian. Recuperado

4 de mayo de 2024, a partir de <https://www.atlassian.com/es/agile/project-management/user-stories>

Besteiro, B. (2022). *Bases de datos relacionales y lenguaje SQL.* Creative

Commons.

Comercio electrónico en Cuba: una realidad creciente. (s. f.). Recuperado 26 de

junio de 2024, a partir de <https://www.cubahora.cu/ciencia-y-tecnologia/comercio-electronico-en-cuba-una-realidad-creciente>

Descubre La Importancia Del Diagrama De Clases En Ingeniería De Software |

2024. (s. f.). Recuperado 4 de mayo de 2024, a partir de

<https://edumasterplus.com/descubre-la-importancia-del-diagrama-de-clases-en-ingenieria-de-software/>

Extreme Programming XP: Ventajas y Desventajas. (2023, 2. julio).

<https://informatecdigital.com/desarrollo/extreme-programming-xp-ventajas-y-desventajas/>

Freeman, E. & Robson, E. (2004). *Head First Design Patterns.* O'Reilly Media.

Git. (2024, 11. abril). En *Wikipedia.*

<https://en.wikipedia.org/w/index.php?title=Git&oldid=1218397028>

Houzz, la aplicación de decoración con la que renovar tu casa. (s. f.). Recuperado 22 de junio de 2024, a partir de

https://www.elespanol.com/elandroidelibre/aplicaciones/20160520/houzz-aplicacion-decoracion-renovar-casa/126237779_0.html

http | Dart package. (s. f.). Dart Packages. Recuperado 17 de mayo de 2024, a partir de <https://pub.dev/packages/http>

Ikea Place, su aplicación de realidad aumentada para decorar tu casa llega a los móviles Android con ARCore. (s. f.). Recuperado 1 de julio de 2024, a partir de <https://www.xatakandroid.com/aplicaciones-android/ikea-place-su-aplicacion-de-realidad-aumentada-para-decorar-tu-casa-llega-a-los-moviles-android-con-arcore>

Introducción a Android Studio. (s. f.). Android Developers. Recuperado 22 de junio de 2024, a partir de <https://developer.android.com/studio/intro?hl=es-419>

La aplicación móvil Wayfair, un ejemplo de cómo compraremos muebles mañana. (s. f.). Recuperado 22 de junio de 2024, a partir de <https://blogs.20minutos.es/un-hogar-con-mucho-oficio/2017/01/17/mira-como-queda-un-mueble-en-tu-casa-con-la-aplicacion-movil-wayfair/>

Mastering Flutter Hex Color: An Essential Guide. (s. f.). Recuperado 17 de mayo de 2024, a partir de <https://www.dhiwise.com/post/flutter-color-magic-exploring-the-power-of-hexcolor-package>

Metodología XP o Programación Extrema: ¿Qué es y cómo aplicarla? (s. f.).

Gestor de proyectos online. Recuperado 4 de mayo de 2024, a partir de

<https://www.sinnaps.com/blog-gestion-proyectos/metodologia-xp>

model_viewer_plus | Flutter package. (s. f.). Dart Packages. Recuperado 25 de

junio de 2024, a partir de https://pub.dev/packages/model_viewer_plus

MVC - Glosario de MDN Web Docs: Definiciones de términos relacionados con la

Web | MDN. (s. f.). Recuperado 22 de junio de 2024, a partir de

<https://developer.mozilla.org/es/docs/Glossary/MVC>

¿Qué es Django? - Explicación del software Django - AWS. (s. f.). Amazon Web

Services, Inc. Recuperado 22 de junio de 2024, a partir de

<https://aws.amazon.com/es/what-is/django/>

¿Qué es Flutter? - Explicación de la aplicación Flutter - AWS. (s. f.). Amazon Web

Services, Inc. Recuperado 22 de junio de 2024, a partir de

<https://aws.amazon.com/es/what-is/flutter/>

¿Qué es la realidad aumentada (AR)? | SAP Insights. (s. f.). Recuperado 22 de

junio de 2024, a partir de

[https://www.sap.com/latinamerica/products/scm/industry-4-0/what-is-](https://www.sap.com/latinamerica/products/scm/industry-4-0/what-is-augmented-reality.html)

[augmented-reality.html](https://www.sap.com/latinamerica/products/scm/industry-4-0/what-is-augmented-reality.html)

Realidad aumentada: ¿qué es, cómo funciona y para qué sirve? (2019, 15.

diciembre). *Rock Content - ES.* [https://rockcontent.com/es/blog/realidad-](https://rockcontent.com/es/blog/realidad-aumentada/)

[aumentada/](https://rockcontent.com/es/blog/realidad-aumentada/)

rive | Flutter package. (s. f.). Dart Packages. Recuperado 17 de mayo de 2024, a partir de <https://pub.dev/packages/rive>

Roles de usuario | AppMaster. (s. f.). Recuperado 4 de mayo de 2024, a partir de <https://appmaster.io/es/glossary/roles-de-usuario>

Turrado, J. (2020, 10. marzo). *Qué son las pruebas de software*. campusMVP.es. <https://www.campusmvp.es/recursos/post/que-son-las-pruebas-de-software.aspx>

Visual Studio Code - Wikipedia, la enciclopedia libre. (s. f.). Recuperado 22 de junio de 2024, a partir de https://es.m.wikipedia.org/wiki/Visual_Studio_Code

What Is MySQL? | Oracle India. (s. f.). Recuperado 22 de junio de 2024, a partir de <https://www.oracle.com/in/mysql/what-is-mysql/>

Anexos

Anexo 1. Imagen de error al utilizar la RA en la app.

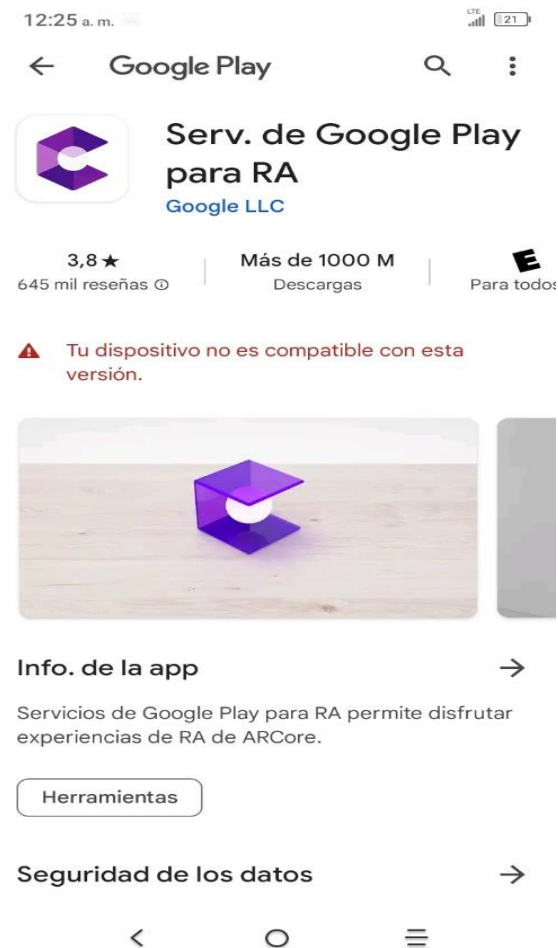


Fig. 3.1 Servicios de Google Play para RA

Anexo 2. Ecuaciones y tablas de la estimación de costo.

Ecuación 3.1: Cálculo de los Puntos de Historias de Usuarios sin ajustar.

$$UUCP = UAW + UUCW$$

Donde:

UUCP: Puntos de Historias de Usuarios sin ajustar.

UAW: Factor de Peso de los Actores sin ajustar.

UUCW: Factor de Peso de Historias de Usuarios sin ajustar.

Factor de Peso de los Actores sin ajustar (UAW)

Este valor se calcula mediante un análisis de la cantidad de Actores presentes en el sistema y la complejidad de cada uno de ellos. En la [tabla 3.3](#) se presenta el Factor de Peso de los Actores sin ajustar.

Tabla 3.3: Factor de peso de los actores sin ajustar

Tipo	Descripción	Peso	Cantidad por peso
Simple	Otro sistema que interactúa mediante una interfaz de programación de aplicaciones. (API)	2	1*2
Medio	Otro sistema que interactúa mediante un protocolo o una persona interactuando con una interfaz basada en texto.	2	0*2
Complejo	Una persona que interactúa con el sistema mediante una interfaz gráfica. (GUI)	3	1*3
Total			2+0+3 = 5

Cálculo de UUCW

Para calcular el UUCW (Unadjusted Use Case Weight) para las historias de usuario, primero se necesita determinar la complejidad de cada una de ellas según el número de transacciones que involucran. Luego se multiplica este valor por el peso asignado a cada tipo de historia de usuario. En la [tabla 3.4](#) se muestra la clasificación de complejidad y cálculo de peso.

Tabla 3.4 Peso de las Historias de Usuario sin Ajustar

Tipo	Descripción	Peso	Cantidad por peso
Simple	La HU contiene de 1 a 3 transacciones.	5	2*5
Medio	La HU contiene de 4 a 7 transacciones.	10	2*10
Complejo	La HU contiene más de 8 transacciones	15	1*15
Total			10+20+15 = 30

Teniendo en cuenta el resultado obtenido en el UAW y UUCW se puede decir que:

$$UUCP = UAW + UUCW$$

$$UUCP = 5 + 30$$

$$UUCP = 35$$

Cálculo de los Puntos de Historias de Usuarios ajustadas.

Una vez que se tienen los Puntos de Historias de Usuarios, se debe ajustar este valor como se muestra en la ecuación 3.2.

Ecuación 3.2: Calculo de los Puntos de Historias de Usuarios ajustadas.

$$UCP = UUCP * TCF * EF$$

Donde:

- **UCP:** Puntos de Historias de Usuarios ajustados.
- **UUCP:** Puntos de Historias de Usuarios sin ajustar.
- **TCF:** Factor de complejidad técnica.
- **EF:** Factor de ambiente.

Factor de complejidad técnica (TCF).

Este coeficiente se calcula mediante la cuantificación de un conjunto de factores que determinan la complejidad técnica del sistema. Cada uno de los factores se cuantifica con un valor de 0 a 5, donde 0 significa un aporte irrelevante o nulo y 5 un aporte muy importante. En la siguiente [tabla 3.5](#) se muestra factor de complejidad técnica con su significado y el peso de cada uno de estos factores.

Tabla 3.5 Factor de complejidad técnica

Factor	Descripción	Peso	Valor	(Peso-i * Valor-i)
T1	Sistema distribuido	2	2	4
T2	Rendimiento o tiempo de respuesta	1	3	3
T3	Eficiencia del usuario final	1	3	3
T4	Procesamiento interno complejo	1	2	2
T5	El código debe ser reutilizable	1	3	3
T6	Facilidad de instalación	0.5	5	2.5
T7	Facilidad de uso	0.5	4	2
T8	Portabilidad	2	5	10
T9	Facilidad de cambio	1	4	4
T10	Concurrencia	1	4	4
T11	Incluye objetivos especiales de seguridad	1	3	3
T12	Acceso directo a terceras partes	1	2	2

T13	Se requieren facilidades especiales de entrenamiento a los usuarios	1	1	1
Total				43.5

Para calcular TCF: Factor de complejidad técnica se muestra la ecuación 3.3 del factor de complejidad técnica.

Ecuación 3.3: Cálculo del Factor de complejidad técnica.

$$\text{TCF} = 0.6 + 0.01 * \Sigma (\text{Peso}_i * \text{Valor}_i)$$

$$\text{TCF} = 0.6 + 0.01 * 43.5$$

$$\text{TCF} = 1.035$$

Factor Ambiente (EF).

El factor ambiente está relacionado con las habilidades y entrenamiento del grupo de desarrollo. Cada factor se cuantifica con un valor desde 0 (aporte irrelevante) hasta 5 (aporte muy relevante). En la siguiente [tabla 3.6](#) se muestra el factor de ambiente con su significado y el peso de cada uno de estos factores.

Tabla 3.6: Factor Ambiente

Factor	Descripción	Peso	Valor	(Peso-i * Valor-i)
E1	Familiaridad con el modelo de proyecto utilizado	1.5	5	7.5
E2	Experiencia en la aplicación	0.5	4	2
E3	Experiencia en orientación a objetos	1	4	4
E4	Capacidad del analista líder	0.5	2	1

E5	Motivación	1	5	5
E6	Estabilidad de los requerimientos	2	4	8
E7	Personal part-time	-1	3	-3
E8	Dificultad del lenguaje de programación	-1	2	-2
Total				22.5

Para Calcular EF: Factor de ambiente se muestra la ecuación 3.4 cálculo del factor ambiente.

Ecuación 3.4: Cálculo del Factor de ambiente.

$$EF = 1.4 - 0.03 * \Sigma (\text{Peso}i * \text{Valor}i)$$

$$EF = 1.4 - 0.03 * 22.5$$

$$EF = 0.725$$

Luego: $UCP = UUCP * TCF * EF$

$$UCP = 35 * 1.035 * 0.725$$

$$UCP = 36.76$$

Estimación de esfuerzo a través de los Puntos de Historias de Usuarios.

Ecuación 3.5: Esfuerzo estimado en horas-hombres.

$$E = UCP * CF$$

Donde:

- **E:** Esfuerzo estimado en horas-hombres.
- **UCP:** Punto de historias de usuarios ajustadas.
- **CF:** Factor de conversión.

Para obtener el factor de conversión (CF) se cuentan cuántos valores de los que afectan el factor ambiente (E1 a E6) están por debajo de la media (< 3), y los que están por encima (> 3) para los restantes (E7 a E8). Si el total (nos da **0**) es 2 ó menos se utiliza el factor de conversión 20 horas-hombre / Punto de historias de usuarios. Si el total es 3 ó 4 se utiliza el factor de conversión 28 horas-hombre / Punto de historias de usuarios. Si el total es mayor o igual que 5 se recomienda efectuar cambios en el proyecto ya que se considera que el riesgo de fracaso es demasiado alto. En este caso:

CF = 20 horas-hombres / Puntos de historias de usuarios.

Luego

E = 36.76 * 20 horas-hombre

E = 735,2 horas-hombre

E = 735 horas-hombre

En la siguiente tabla del anexo ([ver Tabla 3.7](#)) se muestra distribución del esfuerzo por etapas.

Tabla 3.7 Distribución del esfuerzo por etapas

Actividad	% Esfuerzo	Valor esfuerzo
Planificación	5	60
Diseño	20	265
Codificación	45	320
Prueba	15	45
Sobrecarga	15	45
Total	100	735

Una vez estimado el tiempo de desarrollo del proyecto y conociendo la cantidad de desarrolladores y el pago que recibe cada uno de estos se puede llevar a cabo una estimación del costo total del proyecto referidos a los recursos humanos.

K: Coeficiente que tiene en cuenta los costos indirectos (1,5 y 2,0).

THP: Tarifa Horaria Promedio. El salario promedio mensual de los trabajadores en este caso es de \$12 000 CUP dividido entre 176 h es igual a \$ 67.39 por hora.

176 horas (horas de trabajo para un 1 mes, esto se toma a razón de 24 días, ya que no se cuentan los fines de semana ni sábados cortos).

Tiempo = 735 horas/176 \approx equivalente a 4 meses y 17 días, este es el tiempo que tomaría desarrollar el proyecto empleando una sola persona.

4 meses y 17 días a razón de 24 días laborables por mes, representan 100 días. En esta investigación al emplearse 3 trabajadores para el desarrollo del proyecto, el tiempo para su culminación quedará reducido a 1 meses y 39 días aproximadamente.

Entonces el costo total del proyecto:

$$C = E(\text{Total}) * K * \text{THP}$$

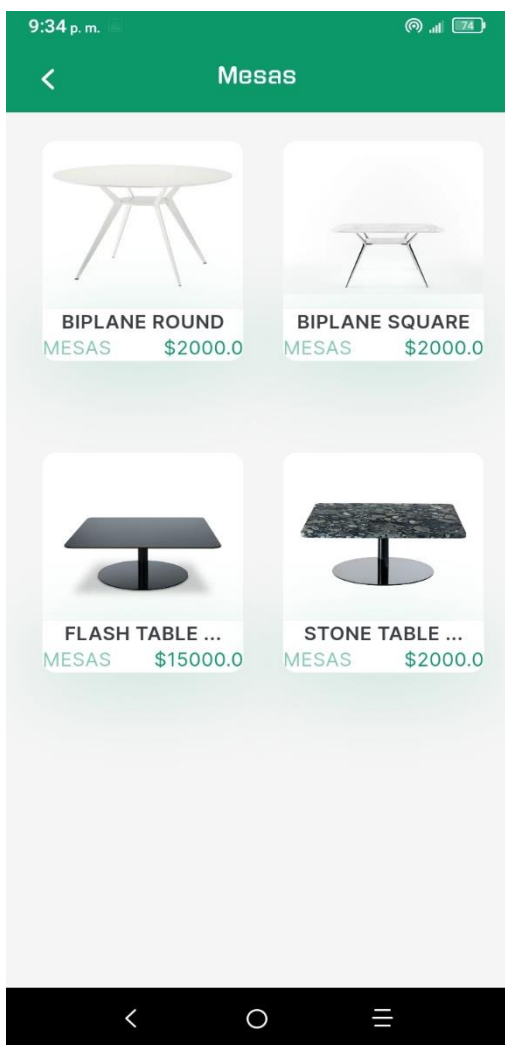
$$C = 735 * 1.5 * [3 * (12\,000/176)] = \underline{\underline{\$ 225\,511.36}}$$

Para el salario a los trabajadores se investigó como varia en diferentes organizaciones o sucursales en Santiago de Cuba, de los sectores estatal y privado, dada las nuevas regulaciones y tasas de cambio. Los datos se obtuvieron a través de trabajadores de las entidades y por anuncios laborales ([ver Tabla 3.8](#)).

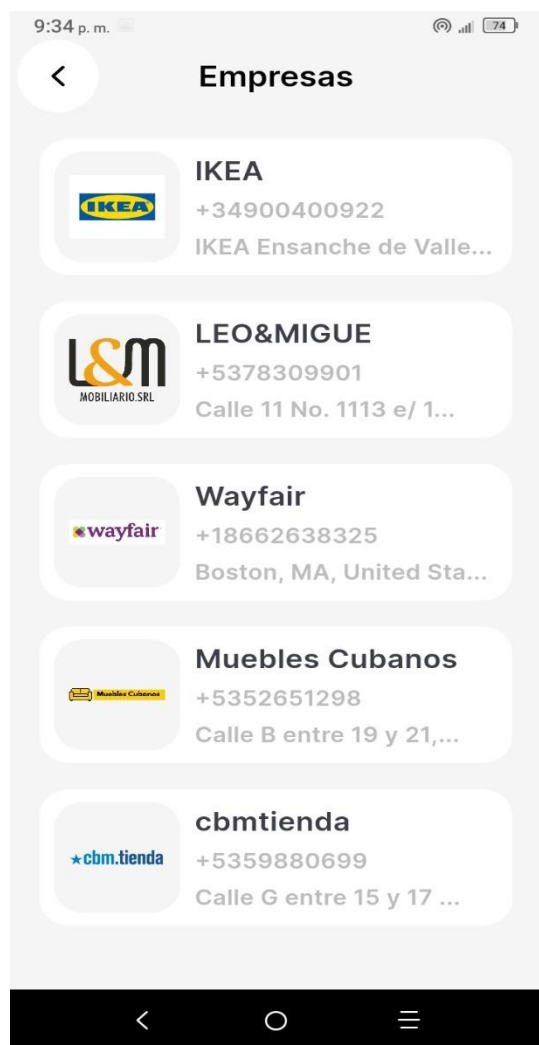
Tabla 3.8 Situación actual de pago

Organización (Sector)	Pago/Mensual (Moneda Nacional)
XETID (Estatad)	9 000 CUP
DATYS (Estatad)	9 000 CUP

DESOFT (Estatel)	9 000 CUP
MYPIMES (Privado)	25 000 CUP
FreeLancer o persona autónoma (privado)	700 MLC a 120 CUP (tasa oficial actual) son 84 000 CUP



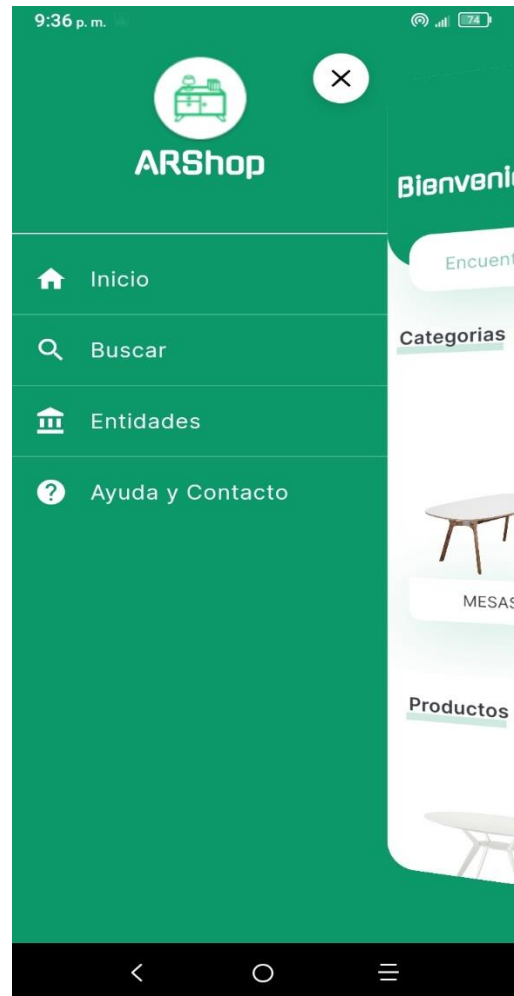
Anexo 3. Listado de productos por categorías.



Anexo 4. Listado de las empresas distribuidoras de productos.



Anexo 5. Interfaz que muestra más información de las empresas.



Anexo 6. Drawer.

Anexo 7 Listado de los muebles del hogar soportados por la App.

Debido a que esta investigación es con tiempo finito se determinó llevar dicho proyecto solamente a muebles del hogar tales como mesas, sillas, sillones, estanterías, armarios, sofás y cómodas. En investigaciones futuras se puede extender la lista de posibles muebles del hogar e incluso incluir otros productos.

Anexo 8 Flujo del Sistema para subir productos al servidor.

En la siguiente [figura](#) se muestra el diagrama de flujo para el marco de trabajo que permite añadir productos al servidor. Haciendo referencia a la información que se muestra en la [figura 4](#), primeramente, las empresas que deseen publicar sus productos a la aplicación deben generar un modelo 3D de los mismos. Para poder agregarlos, se sugiere contratar a un diseñador 3D (Paso 1), quien mediante la fotogrametría hace un escaneo digital del producto (Paso 2) para llevarlo a un modelo tridimensional a escala mediante softwares como “Meshroom⁶” (Paso 3). Una vez obtenido el modelo 3D, se recomienda mejorarlo para que el usuario tenga una mejor experiencia al visualizarlo desde la app (Paso 4). Dicho proceso se puede hacer mediante otros softwares como Blender⁷.

Una vez terminado el modelo 3D con todos sus detalles, este se debe exportar en formato .glb⁸ (Paso 5), que es el formato soportado por el plugin model_viewer_plus de Flutter, el cual es el encargado de mostrar dicho producto en la aplicación para dispositivos móviles. Además, se recomienda comprobar la validez del modelo 3D mediante la página web glTF Validator ([glTF Validator \(khronos.org\)](https://github.com/KhronosGroup/glTF-Validator))⁹.

Una vez obtenido el modelo y comprobado que es válido, este se sube al servidor junto con los demás datos que se deben ingresar del producto (Paso 6 y 7), para que el usuario, cuando acceda a la app, pueda verlo con toda su información y, de la misma manera, visualizarlo en 3D mediante la realidad aumentada.

⁶ <https://github.com/alicevision/Meshroom>

⁷ <https://www.blender.org/>

⁸ <https://filext.com/es/extension-de-archivo/GLB>

⁹ <https://github.com/KhronosGroup/glTF-Validator/>

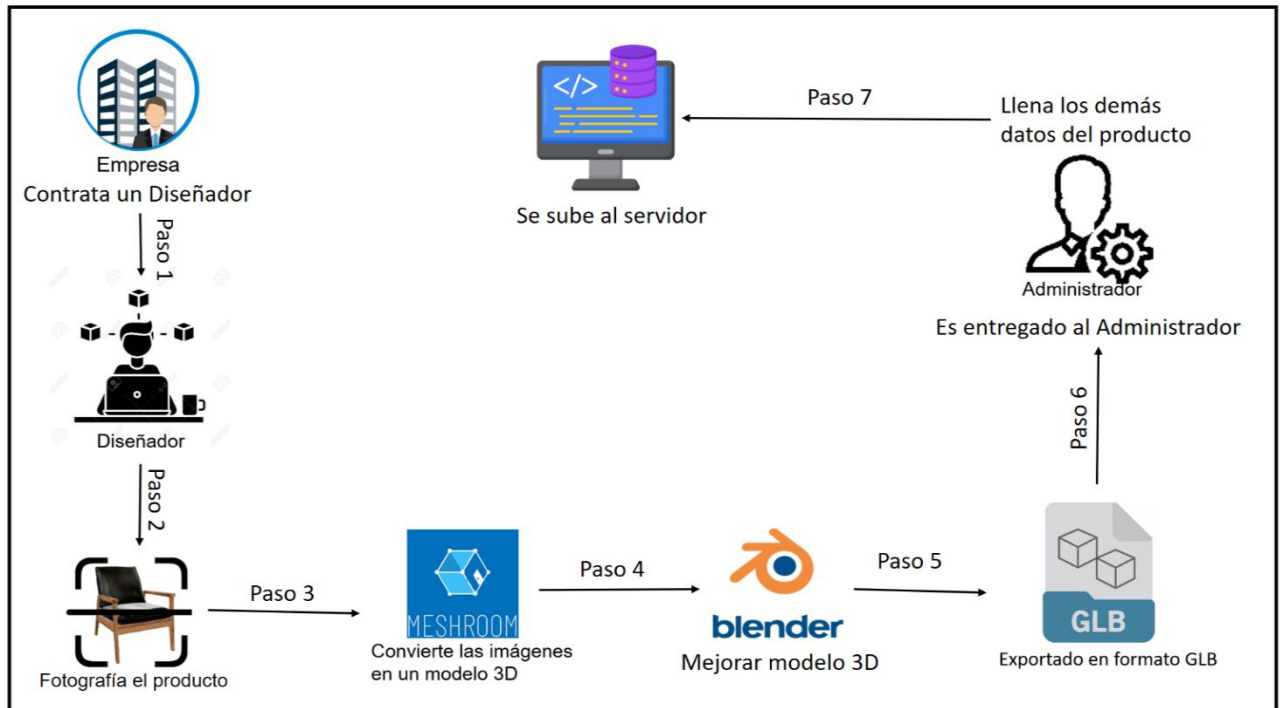


Figura 4 Diagrama de flujo para subir productos al sistema.

Anexo 9 Aval del sistema para la empresa Xetid.

22 de febrero de 2024
"Año del 65 Aniversario de la Revolución"

A quién pueda interesar:

A través de la siguiente carta reconozco la importancia para nuestra institución de la investigación como tesis de pregrado del estudiante de 4to año de Ingeniería Informática Yilbert Benítez Barrio.

La División Territorial de Santiago de Cuba avala esta investigación por el interés en las tecnologías asociadas al marco de trabajo Flutter y las técnicas de Realidad Aumentada.



Liuba María Barroso Revilla

Jefe de Centro de Desarrollo de software



Por este medio avalo que esta carta fue emitida por el Centro de Desarrollo de Software de la División Territorial de Santiago de Cuba de la Empresa de Tecnologías de la Formación para la Defensa (XETID)



Fig. 4.1 Aval del sistema para la empresa Xetid