



Instituto Politécnico Nacional  
Escuela Superior de Cómputo

**Augmented Reality Furniture (ARF)**  
TT2018-A002

*Que para cumplir con la opción de titulación curricular en la  
carrera de:*

**Ingeniería en Sistemas Computacionales**

*Presentan*

Cabello Acosta Gerardo Aramis  
Carrillo Mendoza Martín Alejandro  
Del Pilar Morales Saúl

*Directores*

M. en C. Vélez Saldaña Ulises. Director 1

M. en C. José David Ortega Pacheco. Director 2

## **Abstract**

En este reporte se presenta la documentación técnica y marco teórico del Trabajo Terminal 2018-A002 titulado: **Augmented Reality Furniture (ARF)**, cuyo objetivo es desarrollar una aplicación móvil que permita crear entornos virtuales utilizando realidad aumentada en dispositivos móviles para facilitar el diseño de interiores.

**Palabras clave:** Aplicación móvil, realidad aumentada, diseño de interiores.

# Contenido

<b>1</b>	<b>Introducción</b>	<b>1</b>
1.1	Contexto de trabajo . . . . .	1
1.2	Problemática . . . . .	1
1.3	Trabajo previo . . . . .	2
1.4	Solución propuesta . . . . .	2
1.5	Objetivo . . . . .	3
1.6	Justificación . . . . .	3
<b>2</b>	<b>Marco Teórico</b>	<b>6</b>
2.1	Diseño de interiores . . . . .	6
2.1.1	Definicion . . . . .	6
2.1.2	Movimientos en el diseño de interiores . . . . .	6
2.1.3	Fundamentos del diseño de interiores . . . . .	7
2.2	Realidad Aumentada . . . . .	10
2.2.1	Definición . . . . .	10
2.2.2	Tecnología de realidad aumentada . . . . .	10
2.2.3	Tipos de Realidad . . . . .	11
2.2.4	Taxonomía de la realidad aumentada . . . . .	11
2.2.5	Impacto de la realidad aumentada . . . . .	13
2.2.6	Plataformas de realidad aumentada para dispositivos móviles . . . . .	14
<b>3</b>	<b>Alcance de la aplicación</b>	<b>24</b>
3.1	Descripción general de la aplicación . . . . .	24
3.2	Actores . . . . .	25
3.2.1	Usuario final . . . . .	25
3.3	Requerimientos de la aplicación . . . . .	25
3.3.1	Registro de cuenta . . . . .	25
3.3.2	Inicio de sesión . . . . .	25
3.3.3	Recuperar contraseña . . . . .	25
3.3.4	Consultar muebles . . . . .	25
3.3.5	Cambiar de color a mueble . . . . .	25
3.3.6	Crear escenario . . . . .	26
3.3.7	Tomar foto . . . . .	26
3.3.8	Visualizar escenario . . . . .	26
3.4	Casos de uso . . . . .	27

3.4.1	CU1 Acceso . . . . .	28
3.4.2	CU2 Registro . . . . .	28
3.4.3	CU3 Muebles . . . . .	29
3.4.4	CU4 Crear escenario . . . . .	29
3.4.5	CU5 Visualizar escenario . . . . .	30
3.5	Diagramas de secuencia . . . . .	31
3.5.1	Registro . . . . .	31
3.5.2	Login . . . . .	32
3.5.3	Recuperar contraseña . . . . .	33
3.5.4	Gestión de muebles . . . . .	34
3.5.5	Guardar escenario . . . . .	35
3.5.6	Visualizar escenario . . . . .	36
<b>4</b>	<b>Desarrollo</b>	<b>37</b>
4.1	Iteración 0 . . . . .	37
4.1.1	Exploración . . . . .	37
4.1.2	Inicialización . . . . .	48
4.2	Iteración I . . . . .	52
4.2.1	Resumen . . . . .	52
4.2.2	Desarrollo . . . . .	52
4.3	Iteración II . . . . .	52
4.3.1	Resumen . . . . .	52
4.3.2	Desarrollo . . . . .	53
4.4	Iteración III . . . . .	53
4.4.1	Resumen . . . . .	53
4.4.2	Desarrollo . . . . .	53
4.5	Iteración IV . . . . .	57
4.5.1	Resumen . . . . .	57
4.5.2	Desarrollo . . . . .	57
<b>5</b>	<b>Anexos</b>	<b>59</b>

# **Lista de figuras**

1.1	Modelo del proceso de diseño de interiores. . . . .	5
2.1	Esquema monocromático.[13] . . . . .	8
2.2	Esquema complementario.[13] . . . . .	8
2.3	Esquema análogo.[13] . . . . .	9
2.4	Milgram's reality-virtuality continuum.[27] . . . . .	10
2.5	Restricciones de los puntos para el problema de la posición de la cámara respecto al objeto AR.[27] . . . . .	11
2.6	Visión base de la realida aumentada[22] . . . . .	12
2.7	Taxonomía de la realidad aumentada[22] . . . . .	13
3.1	Casos de uso. . . . .	27
3.2	CU1 - Login. . . . .	28
3.3	CU2 - Registrar usuario. . . . .	28
3.4	CU3 - Gestión de muebles. . . . .	29
3.5	CU4 - Crear escenario. . . . .	29
3.6	CU5 - Visualizar escenario. . . . .	30
3.7	Diagrama de secuencia - Registrar usuario. . . . .	31
3.8	Diagrama de secuencia - Registrar Acceso. . . . .	32
3.9	Diagrama de secuencia - Recuperar contraseña. . . . .	33
3.10	Diagrama de secuencia -Gestión de muebles. . . . .	34
3.11	Diagrama de secuencia - Guardar escenario. . . . .	35
3.12	Diagrama de secuencia - Visualizar escenario. . . . .	36
4.1	Posición Norte . . . . .	39
4.2	Posición Sur . . . . .	39
4.3	Posición Este . . . . .	39
4.4	Posición Oeste . . . . .	39
4.5	Objeto visto de cerca . . . . .	40
4.6	Luz alta . . . . .	40
4.7	Luz baja . . . . .	40
4.8	Luz media . . . . .	40
4.9	Objeto colocado en concreto . . . . .	41
4.10	Objeto colocado en superficie blanca . . . . .	41
4.11	Malla de puntos no detectada en superficie negra . . . . .	41
4.12	Malla de puntos no detectada en vidrio . . . . .	41

4.13	Gran cantidad de objetos en escena . . . . .	42
4.14	Objetos puestos a gran distancia . . . . .	42
4.15	Posición Norte . . . . .	43
4.16	Posición Sur . . . . .	43
4.17	Posición Este . . . . .	44
4.18	Posición Oeste . . . . .	44
4.19	Objeto visto de cerca . . . . .	44
4.20	Objeto visto de lejos . . . . .	44
4.21	Luz baja . . . . .	45
4.22	Luz media . . . . .	45
4.23	Luz alta . . . . .	45
4.24	marca de vuforia (thingmark) . . . . .	46
4.25	Arquitectura de Backend de ARF. . . . .	50
4.26	Modelo Entidad-Relación usado para la base de datos de ARF. . . . .	51
4.27	Hola mundo de ARcore . . . . .	51
4.28	Primer modelo 3D implementado en ARcore . . . . .	52
4.29	Mueble modelado en 3D . . . . .	53
4.30	Diagrama de proceso de Login. . . . .	54
4.31	Flujo de información de login exitoso. . . . .	54
4.32	Flujo de información de login fallido. . . . .	55
4.33	Proceso de registro de nuevo usuario. . . . .	55
4.34	Flujo de información de registro de usuario. . . . .	56
4.35	Proceso de guardado de escenario. . . . .	56
4.36	Flujo de información de guardado de escenario. . . . .	57
4.37	Aplicación: modelos en la parte inferior y botón de tomar foto parte superior . . . . .	57
4.38	Pantalla de visualización de fotografías . . . . .	58
4.39	Pantalla de registro nuevo usuario . . . . .	58
5.1	UI1 - Pantalla inicial de autenticación . . . . .	59
5.2	UI2 - Pantalla principal sin autenticación . . . . .	59
5.3	UI3 - Menú desplegable de muebles (catalogo) . . . . .	60
5.4	UI4 - Pantalla principal tras autenticación . . . . .	60
5.5	UI5 - Pantalla de almacenamiento de escenario . . . . .	60
5.6	UI6 - Pantalla de visualización de escenarios . . . . .	60
5.7	UI7 - Pantalla de visualización de fotografías . . . . .	61
5.8	UI8 - Pantalla de registro nuevo usuario . . . . .	61
5.9	UI9 - Pantalla de recuperación de contraseña 1 . . . . .	61
5.10	UI10 - Pantalla de recuperación de contraseña 2 . . . . .	61
5.11	UI11 - Pantalla de actualización de contraseña . . . . .	62
5.12	UI12 - Pantalla de visualización de imágenes de escenario . . . . .	62

# Índice de tablas

2.1	Listado de dispositivos compatibles con Wikitude . . . . .	16
2.2	Listado de dispositivos compatibles con ARKit 2 . . . . .	17
2.3	Listado de dispositivos compatibles con Vuforia . . . . .	18
2.4	Listado de dispositivos compatibles con Vuforia . . . . .	19
2.5	Listado de dispositivos compatibles con Vuforia . . . . .	20
2.6	Listado de dispositivos compatibles con ARCore . . . . .	22
2.7	Listado de dispositivos compatibles con ARCore . . . . .	23
4.1	Especificaciones de prueba Arcore en Moto G6 . . . . .	38
4.2	Especificaciones de prueba Vuforia en Moto G6 . . . . .	42

# **Capítulo 1**

## **Introducción**

En éste capítulo se define el contexto del trabajo terminal, el problema que vamos a abordar y por qué lo vamos a resolver, cuáles son nuestros objetivos y cómo vamos a lograr solucionar el problema planteado. También está descrito el trabajo previo (estado del arte) el cual aborda la problemática que escogimos

### **1.1 Contexto de trabajo**

Actualmente vivimos en un entorno que dominado por la tecnología. Día a día se desarrollan nuevas herramientas con el fin de ayudar al ser humano a realizar tareas de una forma más fácil y eficiente, como los HMD (Head-mounted Display)[15]. Por otro lado también nos encontramos en una época donde el diseño es un área de gran importancia en cualquier sector del mercado, por ejemplo, un buen diseño web en un sitio es fundamental lograr que un producto se logre vender o difundir, un buen diseño gráfico en campañas de marketing asegura más clientes; de igual forma nos encontramos con el diseño de interiores. Para ésta última área se suelen contratar diseñadores de interiores profesionales para lograr que los espacios interiores de un inmueble consigan tal armonía que mejoren la calidad de vida de quienes lo habitan y además generen un impacto en las personas que usan éstas habitaciones.

El proceso del diseño de interiores está integrado por varias etapas, que se pueden observar en la figura 1.1 . Teniendo a la mano una gran diversidad de herramientas tecnológicas, podemos usar estos elementos para lograr que el diseño de interiores sea más sencillo, intuitivo y más rápido, y que cualquier persona pueda realizarlo sin tener todos los conocimientos y habilidades que se requieren para ello.

### **1.2 Problemática**

El diseño de interiores es un proceso que implica muchas etapas, algunas de ellas son de tal complejidad que necesitan de un diseñador de interiores profesional para completarse con éxito. Hay otras que son de gran lentitud como el desarrollo de las propuestas del diseño pues requieren la elaboración de un modelado tridimensional. Lo anterior conlleva a un proceso de diseño de interiores lento y complejo, el cual si no es realizado de forma exitosa, se tiene que repetir hasta conseguir el resultado deseado, lo cual genera pérdidas económicas y de tiempo tanto para el usuario que desea un diseño de interiores como para el profesional encargado de realizar éste diseño.

### 1.3 Trabajo previo

Las aplicaciones y proyectos que abordan el problema anteriormente descrito son:

1. Canvas (iOS).
2. Amazon App
3. Fingo
4. Ikea Place
5. TT 2012-B043

De forma colectiva, en tales aplicaciones pudimos notar las siguientes características:

1. El usuario puede escanear una habitación en formato tridimensional incluyendo los muebles y objetos que haya en ella
2. El usuario puede exportar el escaneo tridimensional de una habitación para usarlo en AutoCAD
3. Mediante realidad aumentada el usuario puede posicionar un objeto a donde enfoque la cámara del celular
4. Existe una posición relativa de los objetos, es decir, si el celular se mueve el objeto permanece en la misma posición
5. Se requiere un hardware especial además del dispositivo móvil

Cabe destacar que Fingo y el TT 2012-B043 utilizan marcadores físicos, colocados en el suelo, sobre los cuales se superponen los objetos tridimensionales, lo cual limita su uso, dado que son dependientes de un elemento externo.

Por otro lado, encontramos características que consideramos importantes para resolver el problema planteado, pero ninguna de las aplicaciones anteriores las posee, como son:

1. No están enfocadas a e-Commerce
2. No existe un gran repertorio de submodelos de objetos
3. No poseen valores agregados en los objetos en general, por ejemplo, que se muestren las propiedades del producto, o que se puedan cambiar colores de productos

En la **Tabla 1** podemos apreciar una comparación de las aplicaciones anteriores y la aplicación que planeamos hacer con base en las características previamente descritas:

### 1.4 Solución propuesta

Para lograr un diseño de interiores óptimo, proponemos desarrollar una aplicación móvil que permita a los usuarios visualizar de forma virtual, muebles y objetos decorativos en una habitación, eliminando la necesidad de tenerlos físicamente en ella.

Características	Canvas	Tango	Fingo	Ikea Place	TT B043	2012-	Nuestra App
Escaneo							
Exportar							
Enfoque							
Posición relativa							
Hardware externo							
e-Commerce							
Variedad							
Diseños realistas de objetos							
Valor agregado							

Tabla 1. Comparación de aplicaciones que abordan la realidad aumentada y el diseño de interiores

## 1.5 Objetivo

Desarrollar una aplicación móvil que permita crear entornos virtuales utilizando realidad aumentada en dispositivos móviles para facilitar el diseño de interiores.

## 1.6 Justificación

Cuando nos cambiamos de hogar, inevitablemente tenemos que afrontarnos con la tarea de decorar las habitaciones que hay en él. En éste punto, hacerlo no resulta tan complicado dado que partimos de una habitación vacía y ésta se convierte en un lienzo en blanco para nuestra imaginación. Al no haber objetos presentes, la percepción espacial de quien decora no se ve afectada, de tal forma que éste escenario facilita el diseño de interiores. Desafortunadamente no siempre tenemos la oportunidad de decorar una habitación cuando ésta se encuentra vacía, pues normalmente ya hay muebles y objetos decorativos en ella, entonces el proceso se resume a agregar nuevos objetos. Si nosotros escogemos un mueble que se ve agradable a simple vista, puede que, al momento de colocarlo en la habitación, no se encuentre en armonía con los demás objetos, lo cual es uno de los objetivos del diseño de interiores [1]. Incluso al no seguir los procesos fundamentales que el diseño de interiores requiere, como la planeación del espacio [2], es posible que se tenga qué reiniciar todo proceso, lo cual es cansado, por el esfuerzo realizado al reorganizar los elementos de la habitación.

Aunado a esto, puede llegar el punto donde quien decora la habitación, al final ya no desee el mueble, y realice un proceso de devolución de producto, si es que la tienda donde lo compró lo permite. Entonces la tienda pasa al domicilio donde se encuentre el producto para recogerlo o el usuario va a la tienda a entregarlo. De cualquier forma, se traduce en una pérdida económica y de tiempo.

Todas éstas consecuencias se podrían evitar si realizamos un diseño de interiores que siga todas las etapas descritas en el diagrama de procesos anteriormente mostrado, pero por otro lado éste proceso es complejo, tardado y costoso.

Este proceso resulta sencillo para un diseñador de interiores titulado y/o certificado [2], pero no para alguien que no tiene esa misma preparación y aquí es donde se pueden provocar pérdidas económicas y de tiempo por parte del cliente que compra un mueble y/o por parte de la tienda si se

efectúa un proceso de devolución de producto dañando el prestigio de la tienda o sucursal asociada a la venta de estos muebles u objetos.

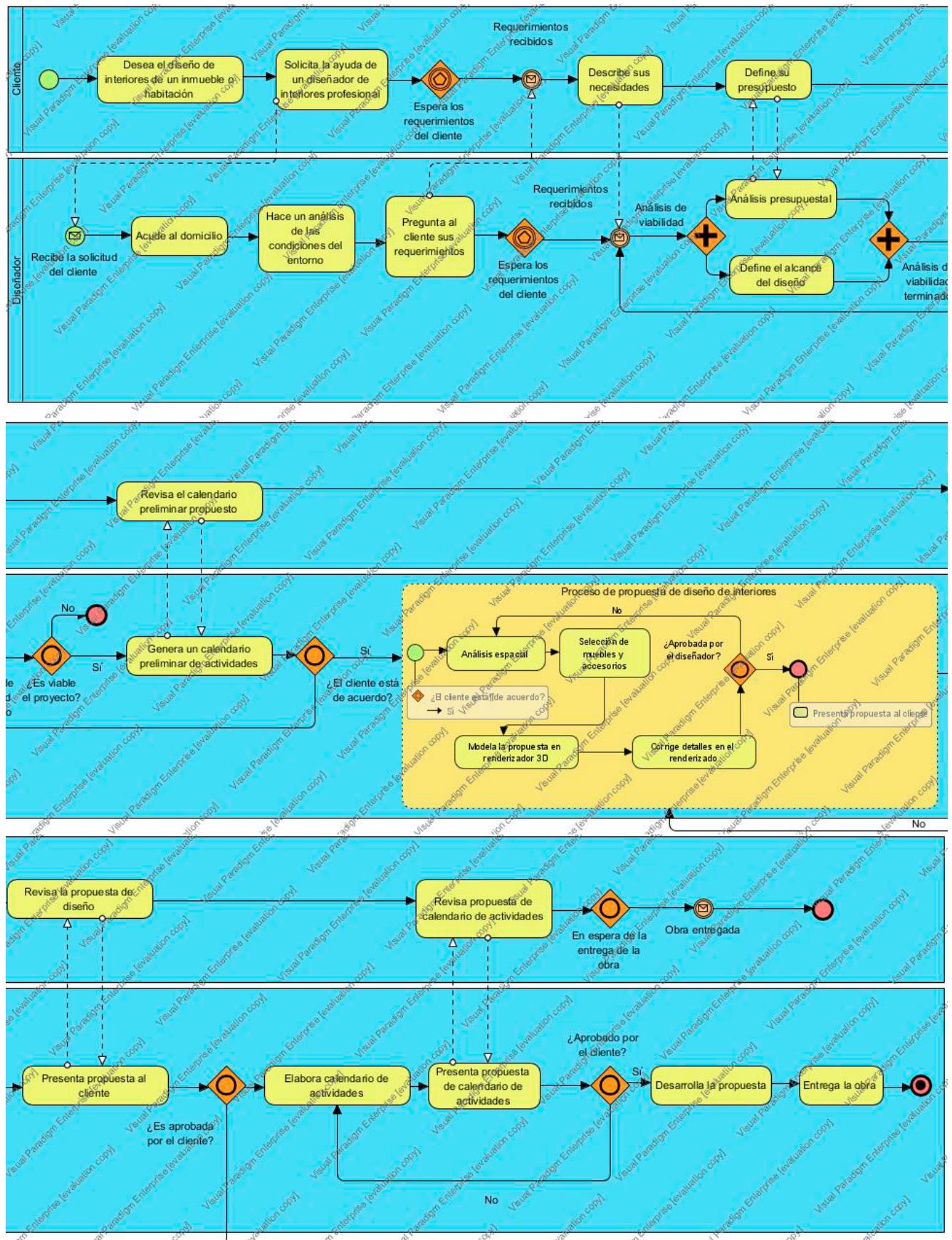


Figura 1.1: Modelo del proceso de diseño de interiores.

# **Capítulo 2**

## **Marco Teórico**

El presente trabajo pretende analizar y documentar el desarrollo de una aplicación móvil para el diseño de interiores, por ello las definiciones que a continuación se exponen son necesarias para entender objetivo y el funcionamiento del software.

### **2.1 Diseño de interiores**

#### **2.1.1 Definicion**

El diseño de interiores es una profesión en la cual soluciones creativas y técnicas son aplicadas dentro de una estructura para lograr la construcción de un entorno interno determinado. Éstas soluciones son funcionales, mejoran la calidad de vida de los ocupantes y son aestéticamente atractivas. Los diseños deben apegarse al código y normas requeridos, y fomentar los principios de sustentabilidad ambiental definidos por el edificio o empresa. El objetivo del diseño de interiores es lograr una armonía en los espacios que habitamos y dar confort al usuario de dichos espacios[1].

El diseño de interiores sigue una metodología sistemática y coordinada que incluye investigación, análisis e integración de conocimientos dentro de un proceso creativo. Dentro ésta metodología podemos ubicar distintos servicios o etapas, dependiendo de la complejidad del trabajo, en las cuales encontramos: definición de los requerimientos funcionales para los espacios de las habitaciones, planeación de espacios interiores, realización de planos de construcción, definición de especificaciones de ubicación, colores y acabados en piso, paredes, materiales, equipo, mobiliario y muebles, administración de contratos de fabricación o instalación, etc.

En Estados Unidos el diseño de interiores es la única rama del diseño que está sujeta a las regulaciones federales y la ley gubernamental[2].

#### **2.1.2 Movimientos en el diseño de interiores**

##### **Feng shui**

”El Feng Shui es un arte utilizado actualmente para alcanzar la armonización de las energías en las casas y los lugares de trabajo, basado en principios milenarios de la sabiduría china”[26]. Surge de la conjunción de dos ideogramas chinos que significan ”viento” y ”agua”, dos conceptos que para las tradiciones de la antigüedad se relacionaban con el flujo y la circulación de la energía vita. Mediante este arte, nos es posible conocer cuál es la perfecta ubicación para edificar una casa, el

lugar ideal para colocar cada uno de los muebles, como así también la forma de revertir las energías adversas que puedan afectarnos. El Feng Shui estudia la relación del hombre con la naturaleza y brinda la oportunidad de vivir de acuerdo con los principios que la rigen, y de esta manera, aprovechar esas energías que fluyen por todas partes y pueden influir en nuestro bienestar general.

### Deconstructivismo

El desconstructivismo es la “Arquitectura que busca llegar a nuevas formas de expresión al alejarse de las restricciones estructurales y jerarquías funcionales y temáticas, enfocado hacia diseños a menudo no rectangulares, fantásticos y aparentemente inconexos”[24]. Tal trabajo a menudo representa una aplicación de las teorías filosóficas de Jacques Derrida en Francia, que trato de llegar a nuevas ideas en la literatura; esta filosofía se ha aplicado desde finales del siglo 20 a las estructuras arquitectónicas generalmente llamadas arquitecturas deconstructivistas.

La arquitectura deconstructivista surge en una exposición, titulada deconstructivist architecture, que Philip Johnson y Mark Wigley organizaron en el museo de Arte Moderno (MoMa) de Nueva York en 1988.

### Diseño Orgánico

Arquitectura cuyo diseño se establece de acuerdo con los procesos de la naturaleza en lugar de basarse en un diseño ya impuesto. Es una filosofía de diseño propuesta por Frank Lloyd Wright (1867-1959) a comienzos del siglo 20 y en ella afirma que un edificio (y su apariencia) deben de seguir formas que estén en armonía con su entorno natural.[24]

Los materiales utilizados en el exterior deben ser acoplarse con la ubicación del edificio, relacionando así el edificio a su entorno. Por lo tanto, debe hacerse de baja altura, con techos que sobresalgan para proporcionar protección del sol en el verano y para proporcionar alguna protección contra la intemperie en invierno además se debe de hacer un máxima uso de la luz natural.

#### 2.1.3 Fundamentos del diseño de interiores

El diseño de interiores se ve como una actividad que tiene un punto de inicio (cuando el diseñador y el cliente tienen el primer contacto) y otro al final cuando el proyecto se ha ejecutado.

Se debe tomar en cuenta que el diseño de interiores es maleable, es decir, que su realización no está sujeto estrictamente a una serie de reglas. En un caso se puede realizar un determinado proceso, y en otro se puede realizar otro proceso diferente. No existe una solución estandarizada para todos los casos.

Lo más importante es definir el por qué estamos diseñando. Por ejemplo si se está diseñando un armario, se tiene qué saber cuál es el impulso para hacerlo. El diseñador se plantea algunas ideas sobre las funciones que tiene un armario, el uso de la madera, reciclada, el del plástico o el nuevo material, y con base a eso, define el objetivo de diseñar el armario. Otro fundamento importante es la armonía que se busca. Un espacio interior no sólo debe verse bonito, y tener colores agradables a la vista. Cada elemento que compone un espacio debe relacionarse con los demás. Un sillón en una sala de espera debe relacionarse y tener alguna conexión con la mesa de centro. Esta relación puede ser la similitud del acabado de ambos, la ubicación de uno con respecto al otro, etc. Una habitación debe seguir un esquema de colores bien definido. Dentro de estos esquemas tenemos el monocromático, complementario y análogo, y cada uno deriva del círculo cromático.

**Monocromático.**- Es una selección de colores que funcionen bien juntos. Esto es trabajar con un matiz, y la variación de tintes, tonos, y sombras.

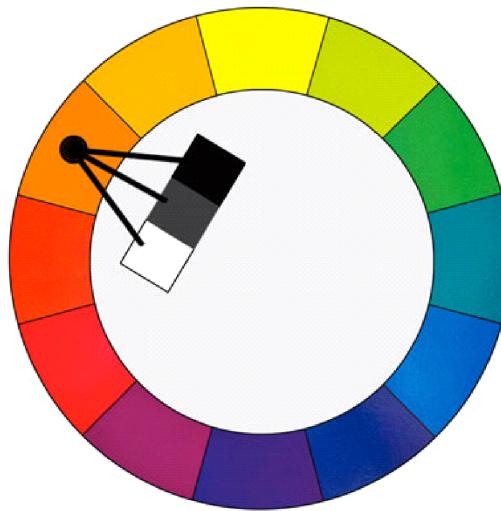


Figura 2.1: Esquema monocromático.[13]

**Complementario.**- Los colores que se encuentran en extremos opuestos del círculo cromático se consideran complementarios. Al combinar estos dos colores, se puede expresar contraste e interés. Son difíciles de usar en grandes cantidades, pero por su contraste son muy buenos para resaltar algo, como un llamado de atención.

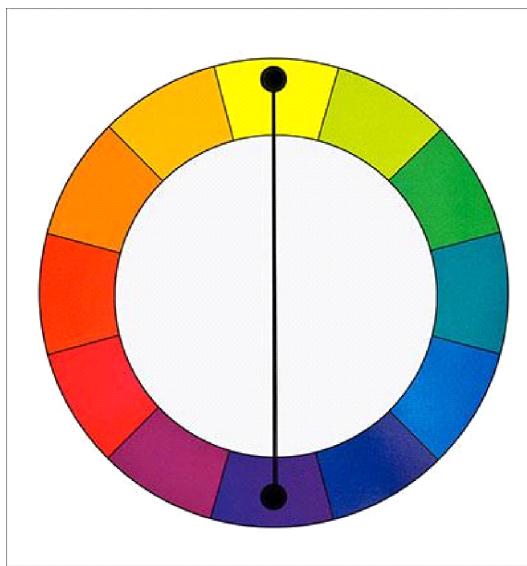


Figura 2.2: Esquema complementario.[13]

**Análogo.**- Los colores que se encuentran al lado en el círculo cromático, son agradables juntos. Son la combinación perfecta, ya que son perfectos para cualquier uso, incluso para resaltar y contrastar un elemento específico sin demasiada interrupción. Como regla general, se debe seleccionar

un color dominante, un segundo color para sustentar, y un tercer color para acentuar.[13]

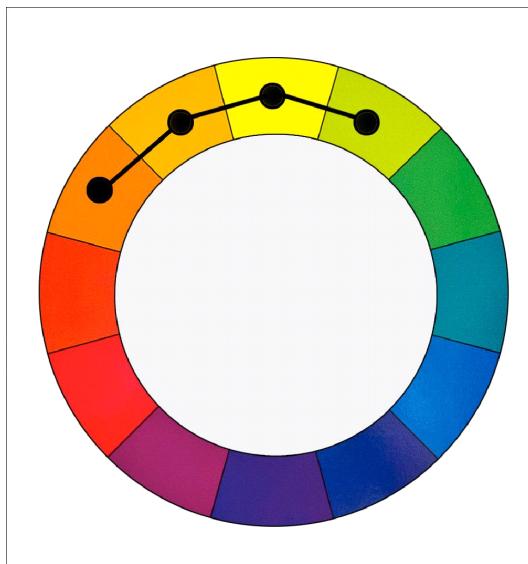


Figura 2.3: Esquema análogo.[13]

## 2.2 Realidad Aumentada

### 2.2.1 Definición

La realidad aumentada (AR) es una aproximación visual interactiva en tiempo real en la cual objetos virtuales son añadidos al entorno real, normalmente en la parte superior de un video, usando gráficos de computadora o móviles.[4]

Básicamente la realidad aumentada se refiere a que imágenes virtuales hechas por computadora sean mezcladas con la vista real para crear una visión con elementos agregados. Su principal fundamento es mezclar la realidad con la realidad virtual. Aunque no sólo se trata de mezclar la realidad virtual, también pueden mezclarse elementos como audio, sensaciones, tacto, olores, y gusto los cuales son superpuestos sobre el mundo real para producir un **entorno de realidad aumentada**.[5] Esta tecnología no es nueva, tuvo sus orígenes en la década de los 90s[4] y su primer aparición a nivel mundial fue en Octubre de 1998 en The First International Workshop on AR (IWAR'98) en San Francisco[5].

La realidad aumentada tiene como objetivo simplificar la vida del usuario al agregar información a su entorno real para su beneficio mediante visión indirecta como la transmisión de video, imágenes, etc. AR mejora la percepción e interacción con el mundo real. Por otro lado la realidad virtual (VR) se sumerge en un entorno puramente virtual sin interactuar con el mundo real.[27]



Figura 2.4: Milgram's reality-virtuality continuum.[27]

### 2.2.2 Tecnología de realidad aumentada

El registro de imágenes de la realidad aumentada utiliza diferentes métodos de visión por computadora, principalmente relacionados con el seguimiento de vídeo. Estos métodos generalmente consisten en dos etapas: reconocimiento y seguimiento/reconstrucción. Primero, se detectan marcadores, imágenes ópticas o puntos de interés en las imágenes de la cámara.[27]

El reconocimiento puede hacer uso de la detección de características, detección de bordes u otros métodos de procesamiento de imágenes con visión artificial, la mayoría de las técnicas de seguimiento disponibles se pueden separar en dos clases[27].

1. Basados en características
2. Basados en modelos

Los métodos basados en características consisten en descubrir la conexión entre las características de imagen en 2D y sus coordenadas del mundo en 3D. [27]

Los métodos basados en modelos utilizan las características de los objetos rastreados, como pueden ser modelos basados en CAD o las plantillas 2D. En el renderizado 3D es posible encontrar la posición de la cámara, basandonos en la proyección de las coordenadas 3D, de igual forma las coordenadas de imagen observada en 2D como se aprecia en la imagen 2.5 [27].

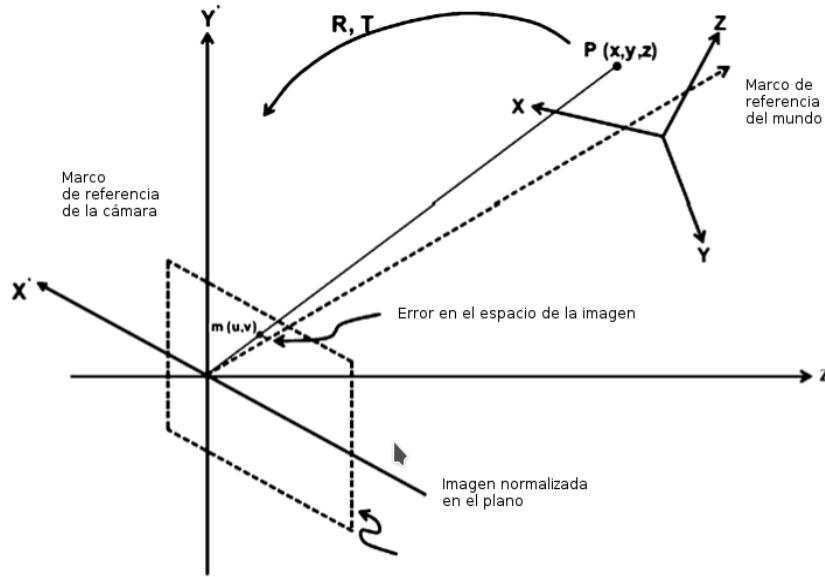


Figura 2.5: Restricciones de los puntos para el problema de la posición de la cámara respecto al objeto AR.[27]

### 2.2.3 Tipos de Realidad

Una de las partes más importantes de la realidad aumentada es la capacidad del usuario para ver su entorno. Sin embargo, el dispositivo de realidad aumentada también tiene que "verlo" y eso implica un sistema de visión basado en computadora [22] .

Todas las pantallas utilizadas en los sistemas de realidad aumentada portátiles (excluidas por dispositivos móviles de definición tales como teléfonos inteligentes, tabletas y portátiles) son normalmente pantallas transparentes que simulan la vision natura[22].

### 2.2.4 Taxonomía de la realidad aumentada

La realidad aumentada es un campo sorprendentemente diverso, robusto y complicado. En un nivel superior se encuentran dos categorías principales: Dispositivos portátiles y dispositivos estacionarios (Móviles y fijos). Los dispositivos móviles incluyen aquellos como auriculares, cascos y

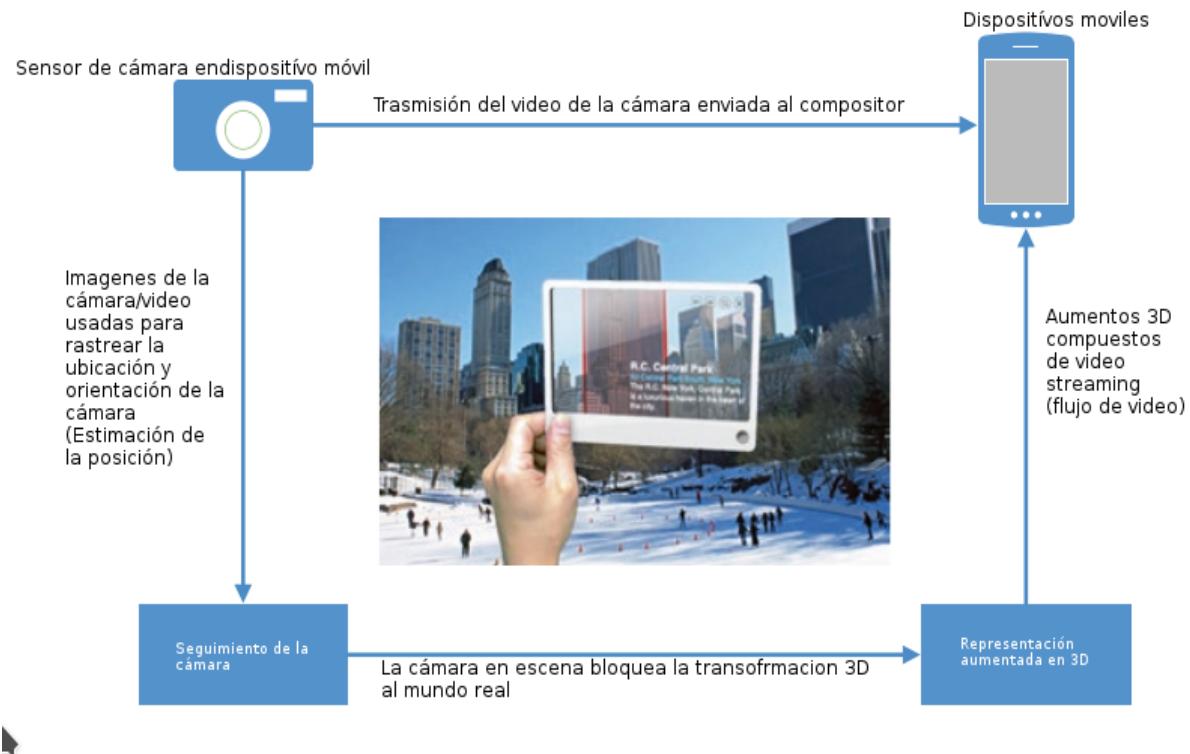


Figura 2.6: Visión base de la realida aumentada[22]

algún día lentes de contacto. Los dispositivos no portátiles (aparatos fijos como televisores, computadoras, reproductores de música, etc.) y dispositivos móviles (Smartphones, tablet's, notebook's), y pantallas de visualización. El siguiente diagrama describe el campo de la realidad aumentada.[22]

La experiencia de la realidad aumentada en dispositivos dedicados o no dedicados como TV's, smartphones, tablet's y PC's. Dentro de los sistemas de realidad aumentada visuales dedicados, hay siete clases:[22]

1. Lentes de contacto
2. Cascos.
3. Display Head-up (HUD)
4. Headset(Gafas inteligentes)
  - a) Integrado (interno, externo)
  - b) Pantallas y sistemas adicionales para gafas convencionales como sol o de seguridad (interno, externo)
5. Proyectores(No HUD)
6. Dedicados y otros (TV, Salud, alarmas, etc)

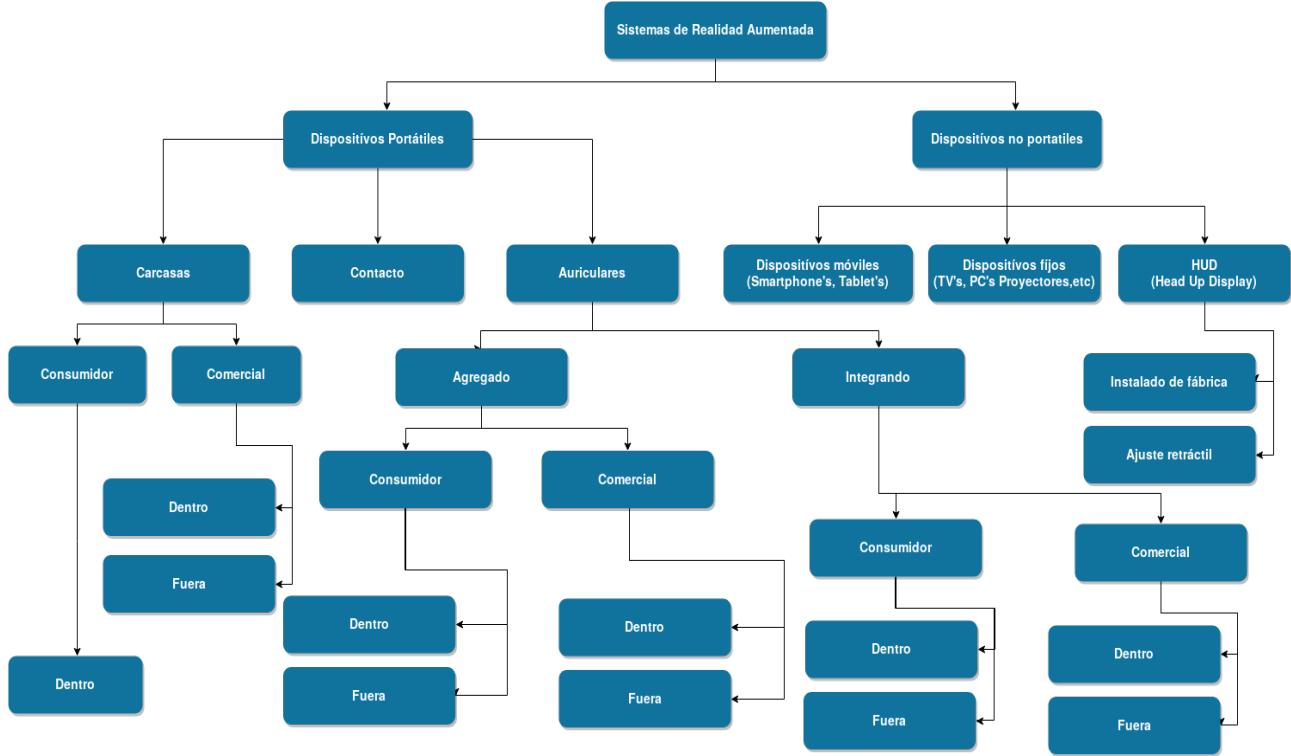


Figura 2.7: Taxonomía de la realidad aumentada[22]

El autor Steve Mann nos dice acerca de la realidad aumentada: "Un mediador ideal de la realidad es tal que es capaz de producir un ilusión de transparencia sobre algunos o todos los campos visuales.[22]

### 2.2.5 Impacto de la realidad aumentada

Desde su aparición en los 90s, la realidad aumentada ha sido una tecnología de punta la cual sólo estaba disponible en algunos laboratorios y de forma remota al público en general.

Más allá de ser una útil técnica de visualización, actualmente es usada en muchos sectores como la ingeniería, la medicina, la robótica, la milicia, la educación, entretenimiento, etc.

La realidad aumentada se ha vuelto una tecnología bastante común, y esto es debido principalmente a la popularidad que tienen los dispositivos inteligentes portátiles (como smartphones y tablets), que ya tienen integradas características como cámaras de video, resolución de alta definición y gran poder computacional que se adecúa al procesamiento de datos relacionados a la realidad aumentada.

Hoy, con el software apropiado, la mayoría de los usuarios de dispositivos inteligentes pueden usar ésta tecnología para múltiples propósitos sin tener que comprar otros dispositivos de costos elevados como los HDM (Head Mount Displays). Adicionalmente con el anuncio de Google Glasses (AR Glasses), se puede inferir que el siguiente nivel de la realidad aumentada será la implementación de la realidad aumentada en pupilentes. Sin embargo puede asumirse que los dispositivos inteligentes seguirán siendo la herramienta dominante para el propósito general de la realidad aumentada en el futuro.

Como una poderosa herramienta de visualización, la realidad aumentada es aplicable en muchas áreas del diseño, las propuestas de diseño pueden ser examinadas, naturalmente la realidad aumentada mezcla elementos virtuales previamente renderizados con el entorno real, lo cual la convierte particularmente útil para las áreas del diseño que involucran construcción de entornos tales como el diseño urbano, la arquitectura y el diseño de interiores. Al reemplazar una porción del entorno físico con algún modelo tridimensional, la diferencia entre el estatus actual del entorno y el estatus simulado se vuelve evidente, logrando que sea posible previsualizar un diseño, sin necesidad de llevarlo a cabo.

En el campo de la investigación del diseño de interiores, la realidad aumentada ha sido usada para simular objetos dentro de un espacio arquitectónico determinado. Estos objetos a menudo son muebles, accesorios o aparatos. Con el entorno físico capturado por un dispositivo en tiempo real y procesado de forma visual, el renderizado de objetos 3D es superpuesto.[15]

### **2.2.6 Plataformas de realidad aumentada para dispositivos móviles**

El impacto y la popularidad de la realidad aumentada han sido tan grandes, que actualmente existen varias plataformas para desarrollar tecnologías que usen realidad aumentada, ya sea para dispositivos móviles como smartphones y tablets, o para dispositivos integradores como los cascos de realidad virtual. Dentro de las plataformas más usadas podemos encontrar:

- Wikitude
- ARKit
- Vuforia
- ARToolKit
- ARCore

## Wikitude

Wikitude es una plataforma de desarrollo de aplicaciones móviles, teniendo como principales dispositivos los smartphones, tabletas y gafas inteligentes. Esta plataforma les otorga a los desarrolladores una combinación única de un mundo físico y virtual. Wikitude se apoya de potentes funciones en combinación con la realidad aumentada (AR), actualmente disponibles para los sistemas Android, iOS, Google Glass, Epson Moverio, Vuzix M-100, Optinvent ORA1, PhoneGap, Titanium y Xamarin. Algunas de estas funciones son:[9]

- **Escaneo de objetos para su reconocimiento.**- Wikitude proporciona un escaneo de objetos como muebles a través de la cámara de un dispositivo móvil. Las últimas versiones de Wikitude se apoya de **SLAM Simultaneous Location and Mapping** que permite un escaneo mas amplio como habitaciones.
- **Seguimiento instantáneo.**- La tecnología Instant Tracking hace posible que las aplicaciones de realidad aumentada trabajen sin necesidad de un marcador, Wikitude utiliza SLAM (**Simultaneous Localization and Mapping**) esta tecnología mapea las imágenes obtenidas por la cámara de un dispositivo compatible y generar marcas o puntos característicos.
- **Reconocimiento 2D.**- El reconocimiento de imágenes por parte de Wikitude proporciona a los desarrolladores el cambiar los ángulos, dimensiones y ubicación de las imágenes virtuales dentro del mundo real obtenida por medio de la cámara del dispositivo en tiempo real.
- **Servicios de geolocalización.**- Esta tecnología nos permite combinar la realidad aumentada (AR) combinando datos georreferenciados. Dependiendo del uso que se le de a la aplicación, la ubicación de algo en particular se apoya de la tecnología (GPS sistema de posicionamiento global). Un ejemplo de este tipo de aplicaciones es Pokemon GO.
- **Reconocimiento multi-imagen.**- Esta función de Wikitude permite el reconocimiento de varias imágenes simultáneamente adquiridas por la cámara del dispositivo. Una vez que se reconocen las imágenes, los desarrolladores pueden manipular modelos 3D y combinarlos con el mundo real, este tipo de reconocimiento múltiple de imágenes se puede usar para brindar interactividad a muchas aplicaciones.
- **Seguimiento extendido.**- Extended Tracking permite a los desarrolladores ir más allá con la manipulación de imágenes y objetos virtuales, los usuarios mejoran su experiencia de realidad aumentada (AR) moviendo libremente sus dispositivos sin la necesidad de mantener un marcador a la vista de la cámara. Esta función de igual forma se apoya de **SLAM**.
- **Reconocimiento en la nube.**- El servicio de reconocimiento en la nube de Wikitude permite a los desarrolladores trabajar con miles de imágenes alojadas en la nube y trabaja con un tiempo de respuesta muy rápido.
- **Aumentado en 3D.**- Esta tecnología carga y renderiza modelos en 3D sobre la escena proporcionada por la cámara del dispositivo, importando los objetos virtuales desde herramientas de renderizado como **Autodesk, Maya o Blender 3D**, Wikitude trabaja conjuntamente con Unity3D para integrar animaciones por computadora.[16]

Wikitude nos ofrece muchas vertientes para el desarrollo de realidad aumentada (AR), actualmente la convierte en una de las mejores tecnologías posicionadas en el mercado. Esta plataforma se puede instalar desde su pagina oficial, existe una versión de prueba y se puede adquirir la licencia desde 2490 euros.[16]

Algunos de los dispositivos compatibles con Wikitude se muestran en la siguiente tabla.

Marca	Modelo
Google	Nexus 4+
	Nexus 5+
	Nexus 6P
	Nexus 10+
	Google Glass
Epson	Epson Moverio BT-200
Apple	iPhone 4+
	Pad2X
	iPod Touch 5th gen
Vuzix	Vuzix M100
Samsung	Galaxy S2+
	Galaxy S7, Galaxy S7 edge
	Galaxy S8, Galaxy S8+
	Galaxy S9, Galaxy S9+
	Galaxy Tab S4
Sony	Xperia XZ Premium
	Xperia XZ1, Xperia XZ1 Compact
	Xperia XZ2, Xperia XZ2 Compact, Xperia XZ2 Premium

Tabla 2.1: Listado de dispositivos compatibles con Wikitude

## ARKit

ARKit 2 es la plataforma nativa de los sistemas **iOS** para el desarrollo de realidad aumentada (AR). Esta API nos proporciona una experiencia única en los objetos 3D incluyendo reflejos de entorno del mundo real en objetos virtuales brillantes, incluso la capacidad de interactuar con varios usuarios en tiempo real. Actualmente ARKit se encuentra en su versión 2 y con soporte exclusivo para los sistemas **iOS 12** y posteriores.[20]

- **Experiencias de AR compartidas.**-Las aplicaciones de AR ya no están limitadas a una sola persona o dispositivo, actualmente ARKit proporciona experiencias multiusuario como juegos donde las experiencias de realidad aumentada se extienden hasta poder ser participe como un espectador.[20]
- **Experiencias de AR persistentes.**- ARKit proporcionar experiencias de AR persistentes mediante una sesión y sea capaz de reanudarla más adelante, por ejemplo los usuarios pueden iniciar un rompecabezas AR en una mesa y volver a él en el mismo estado que se dejó.[20]
- **Detección y Seguimiento de Objetos.**- Desde su versión 1.5 ARKit agregó soporte para la detección de imágenes 2D, lo que le permite desencadenar una experiencia de realidad

aumentada basada en imágenes 2D y aplicarlas en carteles, obras de art, etc. ARKit proporciona un seguimiento dinámico total de imágenes en 2D, de modo que objetos virtuales móviles se podrán observar en nuestro mundo real. ARKit también agrega la capacidad de detectar objetos 3D dentro de un catalogo, como esculturas, juguetes o muebles.[20]

- **Detección y Seguimiento de Objetos.**- Desde su versión 1.5 ARKit agregó soporte para la detección de imágenes 2D, lo que le permite desencadenar una experiencia de realidad aumentada basada en imágenes 2D y aplicarlas en carteles, obras de art, etc. ARKit 2 proporciona un seguimiento dinámico total de imágenes en 2D, de modo que objetos virtuales móviles se podrán observar en nuestro mundo real. ARKit también agrega la capacidad de detectar objetos 3D dentro de un catalogo, como esculturas, juguetes o muebles.[20]
- **Quick Look en objetos 3D y AR.**- Las experiencias de objetos 3D y realidad aumentada (AR) de forma nativa para **iOS** ofrece una optimización de almacenamiento, usted puede compartir estas experiencias en mensajes, correos, noticias, etc. Con la tecnología **Quick Look** los usuarios ven experiencias detalladas e increíbles, incluyendo reflejos del entorno del mundo real en objetos virtuales.[20]

Marca	Modelo
Apple	iPhone 6
	iPhone 6 Plus
	iPhone 6S
	iPhone 6S Plus
	iPhone 7
	iPhone 7 Plus
	iPhone 8
	iPhone 8 Plus
	iPhone X
	iPad Pro 12.9
	iPad Pro 12.9 (2nd Gen)
	iPad Pro 10.5
	iPad Pro 9.7
	iPad (3rd Gen)
	iPad (4th Gen)
	iPad Air

Tabla 2.2: Listado de dispositivos compatibles con ARKit 2

## Vuforia

Vuforia es una API que integra un motor Unity3D para su desarrollo con realidad aumentada (AR), sus aplicaciones están escritas en C sin embargo es multiplataforma.

- **Objetivos.**- Un Target es un patrón de características naturales predefinidas por el desarrollador o usuario. El buscador de Vuforia asocia una determinada a una imagen u objeto, a su vez el software necesita saber qué patrones buscar. Para este propósito, los desarrolladores

pueden usar el administrador de objetivos de Vuforia, de igual forma ofrece a los desarrolladores un servicio web para crear objetivos a partir de imágenes 2D, cuboides, cilindros y objetos tridimensionales.

- **Rastreo Extendido.**- El rastreo es una función que permite a Vuforia encontrar objetivos utilizando las características específicas y se puede construir un mapa de características que rodean al objetivo a escanear y mantener la detección aunque no esté dentro de la vista de la cámara,
- **Reconocimiento de imágenes.**- Vuforia contiene un administrador de objetivos, se puede definir un objetivo de imagen utilizando cualquier patrón, la composición y la cantidad de características son los factores decisivos en la facilidad con que Vuforia puede detectar un objetivo de imagen.
- **Reconocimiento de Objetos.**- Si bien el reconocimiento de objetos de Vuforia utiliza el seguimiento de funciones naturales en su núcleo, el proceso de creación de objetos objetivos es bastante diferente en comparación con la creación de objetos de imágenes virtuales. Una imagen contiene características en 2D mediante el análisis de los pixeles del archivo de imagen proporcionado al administrador de destino. Sin embargo, un objeto 3D tiene un tercer eje a considerar. Los objetivos de objeto permiten a los usuarios ver objetos aumentados desde varios ángulos mientras el objeto mantiene su aumento. Antes de aumentar objetos 3D, Vuforia ha desarrollado una aplicación de escaneo que es capaz de obtener los objetos relativamente pequeños usando un cierto "plano de escaneo" sobre el cual se coloca el objeto a escanear, el objeto a ser escaneado sobre el objetivo de la imagen, el escáner de Vuforia puede determinar la coordenada XYZ, representada por un punto verde.

Vuforia se apoya de un patron o marca como un objetivo detectable por la aplicación. Vuforia trabaja con la posición y orientación, como marco de referencia utiliza un sistema de coordenadas 2D. Algunas de las funciones mas importantes para entender el alcance de esta librería. [12] Algunos de los dispositivos compatibles con Vuforia se muestran a continuación.

Marca	Modelo
Asus	Asus ZenFone AR
Xiaomi	Xiaomi Redmi 3S
	Xiaomi Redmi Note 3
	Xiaomi Redmi Note 3 Snapdragon
	Huawei Mate 10
Huawei	Huawei Mate 10 Pro
	Huawei P10
	Huawei P10 Lite
	Huawei P20 Lite
	Huawei P20
	Huawei P20 Pro

Tabla 2.3: Listado de dispositivos compatibles con Vuforia

Marca	Modelo
Huawei	Huawei Mate 10
	Huawei Mate 10 Pro
	Huawei P10
	Huawei P10 Lite
	Huawei P20 Lite
	Huawei P20
	Huawei P20 Pro
Motorola	Motorola Moto G4
LG	LG V30
	LG G6
	LG G7
	LG G8
OPPO	OPPO R11
	OPPO R11t
Sony	Xperia Z5
	Xperia XZ
Apple	iPad Air 2
	iPad Mini
	iPad Mini 2
	iPad Mini 3
	iPad Mini 4
	iPhone 5
	iPhone 5c
	iPhone 5s
	iPhone SE
	iPhone 6
	iPhone 6 Plus
	iPhone 6S
	iPhone 6S Plus
	iPhone 7
	iPhone 7 Plus
	iPhone 8
	iPhone 8 Plus
	iPhone X
	iPad Pro 12.9
	iPad Pro 12.9 (2nd Gen)
	iPad Pro 10.5
	iPad Pro 9.7
	iPad (3rd Gen)
	iPad (4th Gen)
	iPad (5th Gen)
	iPad Air
	iPod Touch (5th Gen)
	iPod Touch (6th Gen)

Tabla 2.4: Listado de dispositivos compatibles con Vuforia

Marca	Modelo
Samsung	Samsung Galaxy J5 Pro
	Samsung Galaxy S6
	Samsung Galaxy S6 Edge
	Samsung Galaxy S6 Edge+
	Samsung Galaxy A7
	Samsung Galaxy J7 Pro
	Samsung Galaxy S7
	Samsung Galaxy S7 Edge - Exynos
	Samsung Galaxy S7 Edge - Snapdragon
	Samsung Galaxy A8+
	Samsung Galaxy S8
	Samsung Galaxy S8+
	Samsung Galaxy S9
	Samsung Galaxy S9+
	Samsung Galaxy Note 5 – Exynos
	Samsung Galaxy Note 8
	Samsung Galaxy Note 9
	Galaxy Tab S3 9.7
Google	Google Pixel
	Google Pixel XL
	Google Pixel 2
	Google Pixel 2 XL
	Google Nexus 6P

Tabla 2.5: Listado de dispositivos compatibles con Vuforia

## ARCore

ARCore es la plataforma de Google para construir experiencias de realidad aumentada. A través de diferentes APIs, ARCore le da la habilidad a un dispositivo móvil para percibir su entorno, entender el mundo e interactuar con la información. Algunas de nuestras APIs están disponibles a través de Android y iOS para crear experiencias de realidad aumentada compartidas [14].

ARCore tiene nueve pilares básicos que permiten integrar contenido virtual en el entorno real, por medio de la cámara de un dispositivo móvil:

- **Rastreo de movimiento.**- Cuando el celular se mueve a través del mundo real, ARCore usa un proceso llamado **Odometría concurrente y mapeo** o simplemente **COM**, para entender en qué posición se encuentra con respecto al entorno. ARCore detecta visualmente distintas características en las imágenes capturadas por la cámara para definir puntos llamados **puntos característicos**, a partir de los cuales virtualmente genera una maya de puntos, y los usa para computar el cambio físico de su posición. Esto se combina con mediciones realizadas por el teléfono para determinar la posición y la orientación de la cámara relativa al mundo en tiempo real.
- **Entendimiento ambiental.**- ARCore trata de identificar una maya de puntos coincidentes dentro de una misma superficies, como una mesa, el suelo o un muro, y hace que cada una

de estas superficies detectadas esté disponible como un **plano**. ARCore también es capaz de detectar los límites de cada plano y convertir esto en información útil para las aplicaciones, de tal forma que sea posible posicionar objetos virtualmente sobre cada uno de estos planos. Debido a que ARCore usa puntos característicos para detectar planos, superficies sin texturas como paredes blancas pueden no ser detectadas apropiadamente.

- **Estimación de luz.**- ARCore puede detectar información sobre la iluminación del entorno y proveer a la cámara de dispositivo móvil de una corrección de color y gamma, para lograr una imagen óptima. Esta información permite variar la iluminación de una habitación y ver cómo la iluminación también varía en los elementos virtuales, aumentando la sensación de realismo.
- **Interacción con el usuario.**- Una vez que se ha definido la maya de puntos, ARCore permite que el usuario interactúe con el entorno visualizado en la cámara del dispositivo. El usuario puede hacer tap, o realizar gestos con los dedos sobre la pantalla del móvil, y ARCore tiene la capacidad de interpretar y transportar éstas acciones al entorno virtual, por ejemplo, el usuario puede agregar un objeto en alguna superficie al tocar la pantalla del dispositivo con el dedo, tras esto se hace una cálculo de las coordenadas relativas X, Y del plano donde se va a agregar el objeto, y finalmente el objeto es mostrado a través de la cámara.
- **Puntos de orientación.**- Los puntos de orientación permiten colocar objetos virtuales en superficies inclinadas. Cuando se está detectando la maya de puntos y se detectan superficies cercanas con diferente ángulo, ARCore se encarga de calcular el ángulo de inclinación de tal superficie.
- **Anclas y rastreables.**- La posición y orientación puede cambiar conforme ARCore mejora su entendimiento del entorno. Cuando se quiere posicionar un objeto virtual, se requiere definir una ancla para asegurar que su posición sea rastreada en tiempo real. Los planos y puntos son un tipo de objeto especial denominado **rastreable**. Como su nombre sugiere, son objetos que ARCore estará rastreando a lo largo del tiempo. Es posible anclar objetos a rastreables, de tal forma que estos objetos virtuales van a conservar su posición relativa en el mundo, no importando si la cámara se mueve e inclusive si estos salen del foco de visualización.
- **Imágenes aumentadas.**- Las imágenes aumentadas permiten desarrollar aplicaciones de realidad aumentada que puedan responder a imágenes en 2D específicas como paquetes o posters de películas. Los usuarios pueden tener experiencias de realidad aumentada cuando ellos enfocan la cámara del celular a uno de estos elementos, por ejemplo, al enfocar la cámara el póster de una película se puede hacer que un personaje aparezca dentro del entorno virtual, y desaparezca cuando la cámara deje de enfocar a la imagen.
- **Compartir.- Cloud Anchors API**) permite crear experiencias de realidad aumentada colaborativas con otras personas. Con ésta API, un dispositivo puede enviar un ancla y puntos característicos a la nube. Éstas anclas pueden ser compartidas con otros usuarios ya sea de Android o iOS. Esto habilita a las apps para renderizar los mismos objetos 3D asociados a tales anclas, permitiendo a los usuarios tener la misma experiencia de realidad aumentada de forma simultánea

Dado que ARCore es una tecnología nueva (su primer publicación oficial fue en febrero de 2018 y su primera versión estable fue en agosto de 2018) la lista de dispositivos compatibles con ella no es tan

amplia. En la *Tabla 2* podemos observar una lista con los dispositivos que actualmente soportan ARCore.

Marca	Modelo
ASUS	Zenfone AR
	Zenfone ARES
Google	Nexus 5X
	Nexus 6P
	Pixel, Pixel XL
	Pixel 2, Pixel 2 XL
LG	G6
	G7 ThinQ
	V30, V30+, V30+ JOJO
	V35 ThinQ
HMD Global	Nokia 6
	Nokia 6.1 Plus
	Nokia 7 Plus
	Nokia 8
	Nokia 8 Sirocco

Tabla 2.6: Listado de dispositivos compatibles con ARCore

Marca	Modelo
Huawei	Honor 10
	nova 3, nova 3i
	P20, P20 Pro
	Porsche Design Mate RS
Motorola	Moto GS5 Plus
	Moto G6
	Moto G6 Plus
	Moto X4
	Moto Z2 Force
	Moto Z3
OnePlus	Moto Z3 Play
	OnePlus 3T
	OnePlus 5
	OnePlus 5T
	OnePlus 6
Samsung	Galaxy A5
	Galaxy A6
	Galaxy A7
	Galaxy A8, Galaxy A8+
	Galaxy Note8
	Galaxy Note9
	Galaxy S7, Galaxy S7 edge
	Galaxy S8, Galaxy S8+
	Galaxy S9, Galaxy S9+
	Galaxy Tab S4
Sony	Xperia XZ Premium
	Xperia XZ1, Xperia XZ1 Compact
	Xperia XZ2, Xperia XZ2 Compact, Xperia XZ2 Premium

Tabla 2.7: Listado de dispositivos compatibles con ARCore

# Capítulo 3

## Alcance de la aplicación

En la siguiente sección describe los requerimientos y diagramas de la aplicación. Además, se detallan los diferentes escenarios posibles.

### 3.1 Descripción general de la aplicación

Este sistema móvil sera capaz de brindarle al usuario final una herramienta que le permita observar muebles virtuales y al mismo tiempo de percibir la realidad obtenida por una cámara,

Se observará un vídeo en tiempo real de la información virtual sobrepuerta dentro del entorno real por medio de la cámara. La idea es darle una perspectiva más exacta en forma, color, dimensión nuestro mueble u objeto de agrado. Los resultados de obtener esta realidad mezclada es generar acierto, certidumbre y seguridad a las necesidades de un usuario con pretensiones de adquirirlo mediante una compra. Esta aplicación será enfocada al e-Commerce y el comprador potencial serán mueblerías y establecimientos de venta de artículos de decoración y hogar. Para concluir enlistaremos las principales características que tendrá este TT.

1. La aplicación debe ser un sistema móvil capaz de comunicarse con los servicios de AWS (Amazon Web Services). Esta plataforma nos ofrece todo lo necesario para la interacción y persistencia de la información. AWS será el encargado de alojar la información dentro de una instancia de RDS, ahí se encontrara MySQL donde se encontrarán los usuarios y la información de los muebles. Por otro lado AWS nos ofrece S3 para el alojamiento de las imágenes por medio de S3 Bucket.
2. Los datos registrados por cada usuario dentro de la aplicación será un nombre, correo y contraseña.
3. Cada imagen será alojada en formato base64. Por otro lado el tamaño en bytes de cada objeto dependerá de la complejidad, textura, colores, dimensiones y detalles que influyan significativamente en el proceso de renderización en un dispositivo anfitrión.
4. Por otro lado, cada dispositivo anfitrión deberá contar con un mínimo de hardware y software definido en la sección técnica dentro de este documento, los usuarios que interactúen con la aplicación deberá contar con los permisos necesarios para gestionar la interacción con hardware o software disponible en el dispositivo.

## 3.2 Actores

### 3.2.1 Usuario final

**Nombre del actor:** *Cliente (Usuario)*

**Definición :** Usuario final y el principal interesado en usar nuestra aplicación. Tendrá los permisos de un usuario estándar, este actor sera el único considerado hasta este momento ya que en iteraciones posteriores se contempla el aumento de actores. A continuación se describen las funciones donde interviene la aplicación este usuario.

## 3.3 Requerimientos de la aplicación

La siguiente sección describimos los módulos, detallando la funcionalidad de cada uno y su flujo con sus posibles escenarios.

### 3.3.1 Registro de cuenta

**Actor :** *Cliente*

**Descripción :** Los usuarios deberán registrarse si desean interactuar por completo con todas las funciones disponibles de la aplicación. Este registro se basa en crear una cuenta con tu nombre, correo electrónico y una contraseña.

### 3.3.2 Inicio de sesión

**Actor :** *Cliente*

**Descripción :** Los usuarios deberán acceder con sus datos de registro para tener acceso a toda la funcionalidad de la aplicación.

### 3.3.3 Recuperar contraseña

**Actor :** *Cliente*

**Descripción :** Los usuarios que olviden su contraseña de acceso deberán solicitarla y posteriormente enviada a su correo electrónico

### 3.3.4 Consultar muebles

**Actor :** *Cliente*

**Descripción :** Los usuarios podrán consultar los muebles disponibles dentro de un catalogo, donde sera el modelo visualizado en la realidad aumentada.

### 3.3.5 Cambiar de color a mueble

**Actor :** *Cliente*

**Descripción :** Los usuarios podran cambiarle el color a un mueble.

### 3.3.6 Crear escenario

**Actor :** *Cliente*

**Descripción :** Los usuarios podrán crear un escenario, consiste en una galería de fotos de un mueble dentro del interior de un espacio tomado por el smartphone.

### 3.3.7 Tomar foto

**Actor :** *Cliente*

**Descripción :** Los usuarios podrán registrar capturas de su experiencia de la realidad aumentada de la aplicación dentro de un escenario creado.

### 3.3.8 Visualizar escenario

**Actor :** *Cliente*

**Descripción :** Los usuarios podrán ver un escenario tomado, donde observaran las imágenes guardadas dentro de su smartphone.

### 3.4 Casos de uso

En la siguiente sección se define y describen los Actores, funciones y escenarios, describiendo su funcionalidad e interacción dentro de la aplicación.

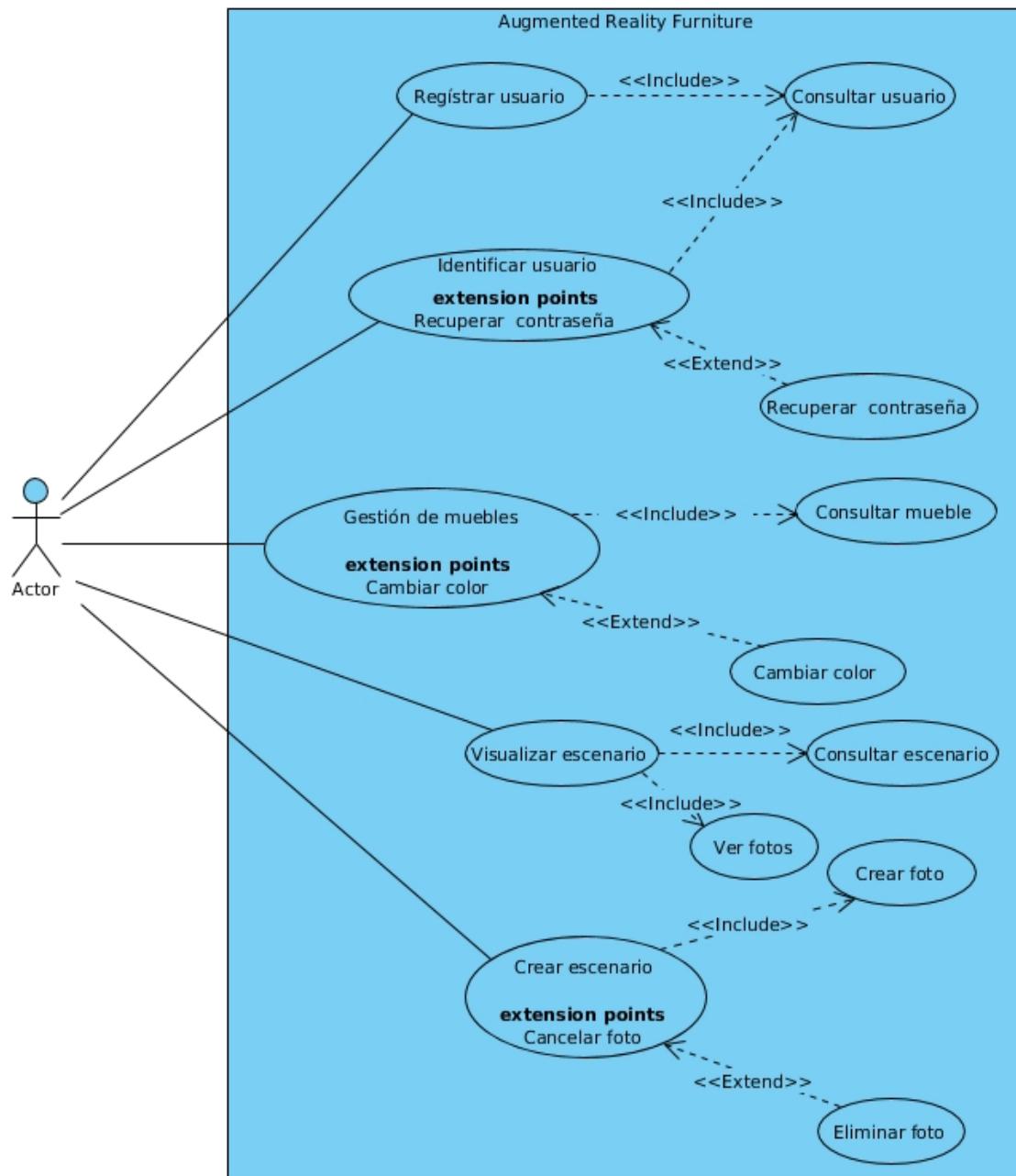


Figura 3.1: Casos de uso.

### 3.4.1 CU1 Acceso

Escenario de autenticación y acceso a la aplicación. Las entradas son un correo y una contraseña, ambos validados en la base de datos. Incluye una función que recupera la contraseña de los usuarios mediante el envío de un email.

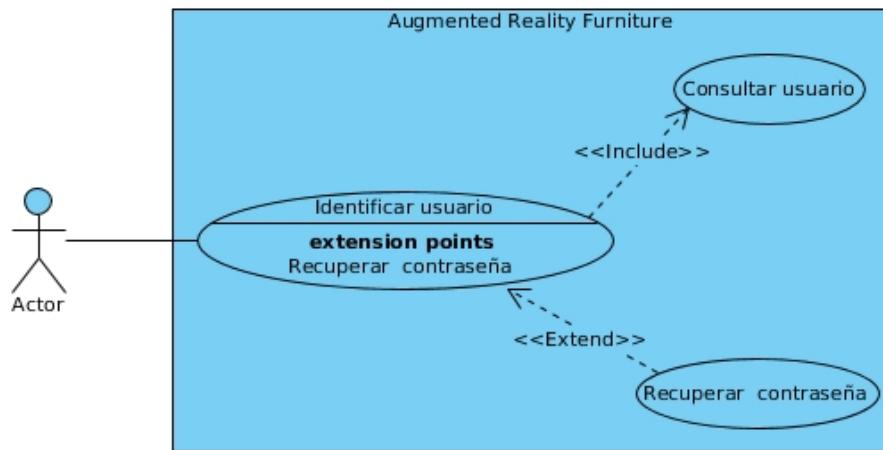


Figura 3.2: CU1 - Login.

### 3.4.2 CU2 Registro

El registro de la información de un usuario, los datos registrados son el nombre, correo electrónico y una contraseña. El pre-registro con la información sera consultada para que no se registren usuarios idénticos.

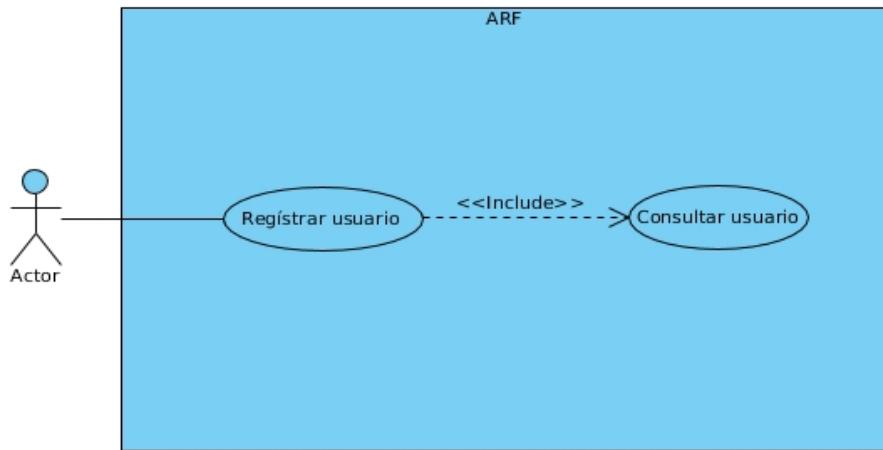


Figura 3.3: CU2 - Registrar usuario.

### 3.4.3 CU3 Muebles

Escenario donde un usuario consulta, cambia de color y selecciona un mueble para visualizarlo con realidad aumentada.

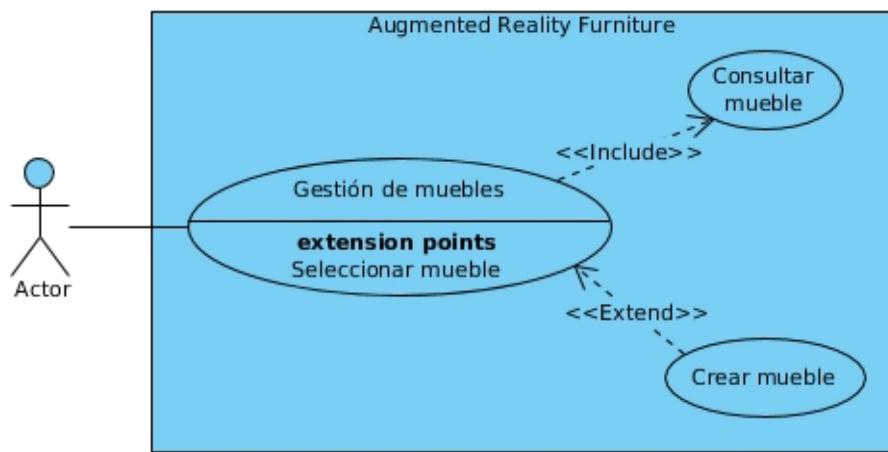


Figura 3.4: CU3 - Gestión de muebles.

### 3.4.4 CU4 Crear escenario

El actor podrá crear escenarios, cada escenario es una galería de captura de imágenes generadas con un mueble virtual y la realidad registrada por la cámara, incluye la función de eliminar imágenes antes de guardar el escenario.

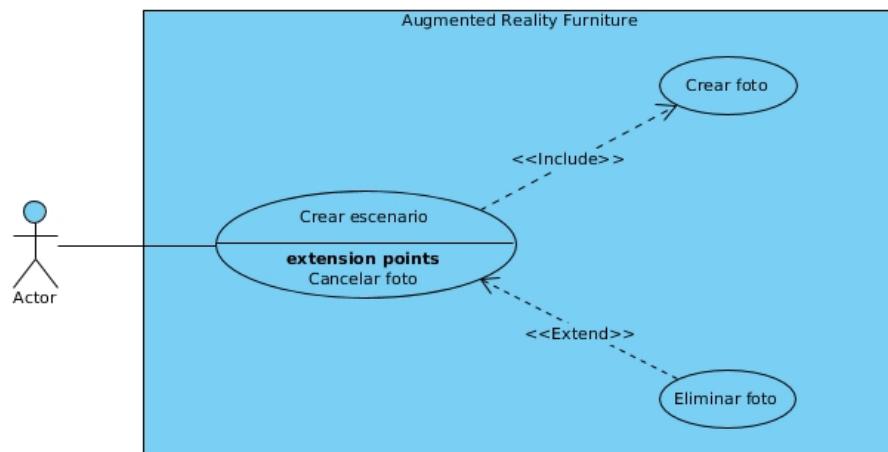


Figura 3.5: CU4 - Crear escenario.

### 3.4.5 CU5 Visualizar escenario

En esta parte el usuario podrá ver las imágenes de un escenario generado con anterioridad, podrá seleccionarlo desde una lista con sus escenarios disponibles

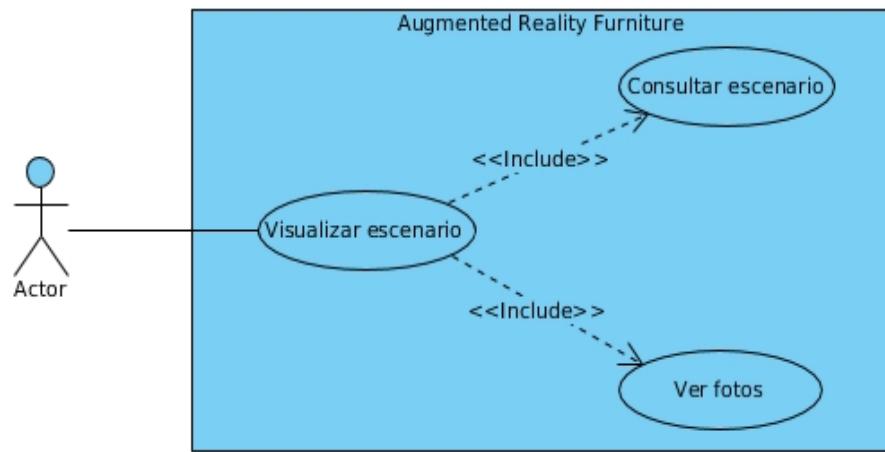


Figura 3.6: CU5 - Visualizar escenario.

## 3.5 Diagramas de secuencia

A continuación se describe la interacción de nuestras entidades de negocio y el ciclo de vida que tendrán en la aplicación. También se detalla la interacción, mensajes y la lógica implementada por cada uno de los diferentes escenarios.

### 3.5.1 Registro

El registro de un usuario nuevo consiste en capturar el correo, nombre y una contraseña. Esta captura sera tratada como un JSON el cual viajara atravez de un método POST. Gateway se encargara de redirigir esta petición al servicio de registro. Este servicio viajara hacia RDS donde se enviara un INSERT para registrar los datos.

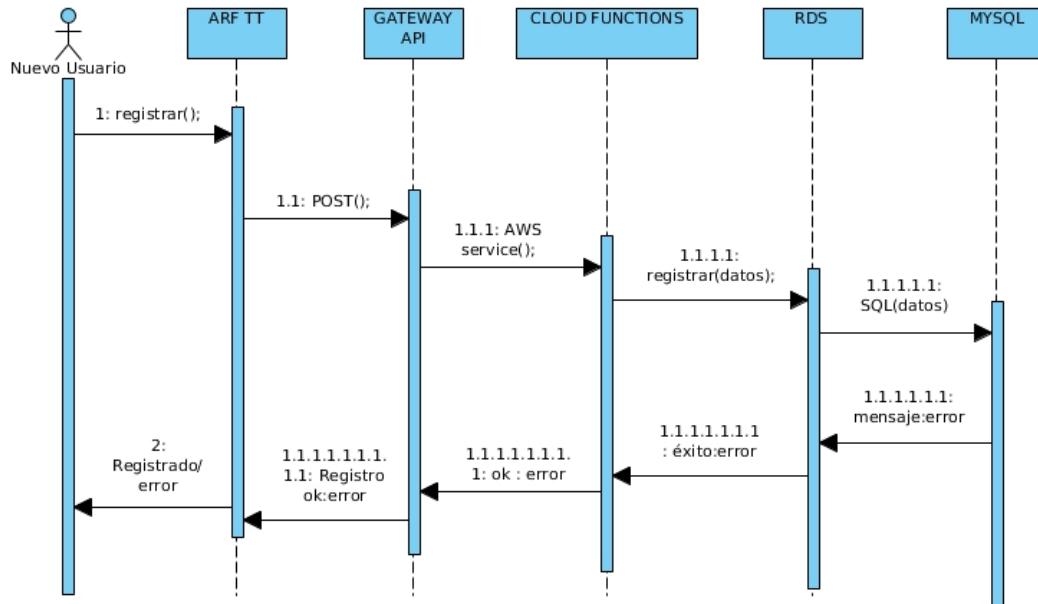


Figura 3.7: Diagrama de secuencia - Registrar usuario.

Si el usuario y la contraseña se encuentra repetido, se enviara un mensaje de error, como se muestra en la figura 3.7

### 3.5.2 Login

La validación de credenciales comienza con capturar el correo y contraseña de un usuario registrado. Esta información sera tratada como un objeto JSON, el cual se mandara por una método POST. La API Gateway direccionara la petición a un Servicio AWS y comparara la información en la instancia creada en RDS de un motor de base de datos MySQL. RDS se encargara de generar la conexión y consulta a la base de datos.

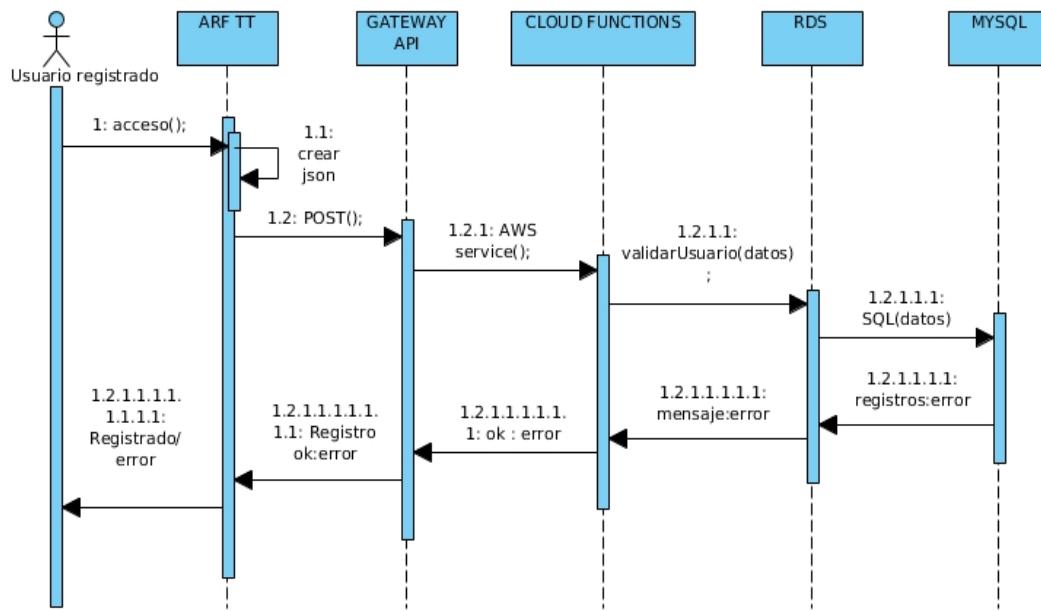


Figura 3.8: Diagrama de secuencia - Registrar Acceso.

La petición regresara con un mensaje si hubo coincidencia, de lo contrario se enviara un error a la interfaz de usuario.

### 3.5.3 Recuperar contraseña

La recuperación de credenciales inicia con la función recuperar, en ella se captura el correo de un usuario y posteriormente lo procesa a un objeto JSON. Esta información se mandara en un método POST. La API Gateway envía la petición a un Servicio AWS y en la instancia creada en RDS procesa un script que comparara el correo con la información de MySQL, buscando alguna coincidencia. Si existe una coincidencia exacta, MySQL enviará el registro en un mensaje con la contraseña.

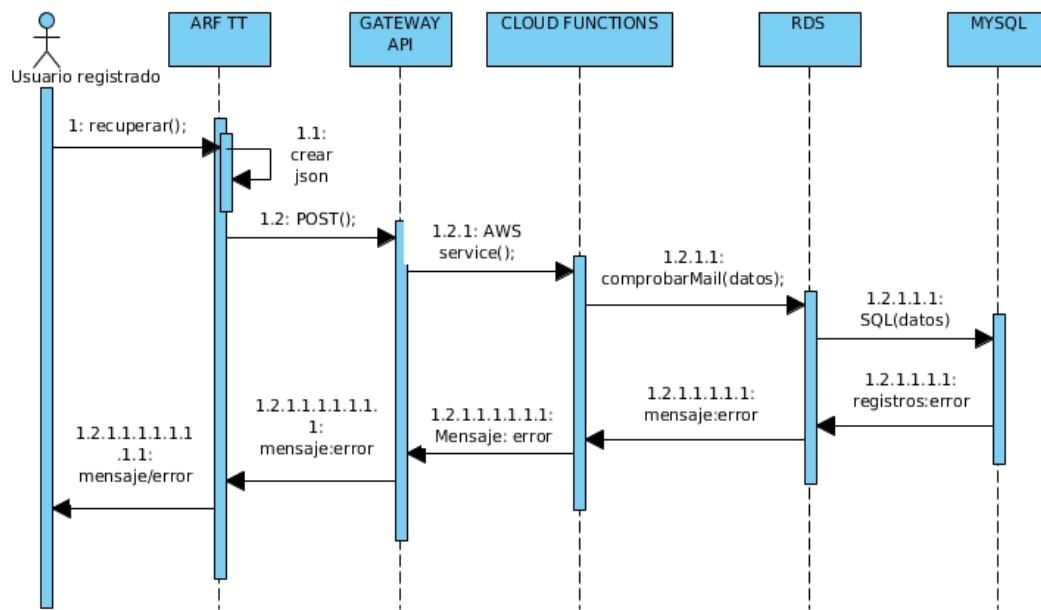


Figura 3.9: Diagrama de secuencia - Recuperar contraseña.

Si no existe coincidencia se manda un mensaje de error. RDS se encargara de generar la conexión y consulta con el servicio de Amazon Web Services y nuestra instancia de MySQL en RDS.

### 3.5.4 Gestión de muebles

La visualización y edición del color de un mueble inicia desde un menú con los muebles disponibles, cuando un actor selecciona un mueble este dispara una acción donde opcionalmente podrá editar el color. El actor después de seleccionar un color valido enviara la información a los servicios AWS donde se tienen almacenados los muebles, esta información será enviada de regreso para visualizar este objeto con la realidad aumentada de la aplicación.

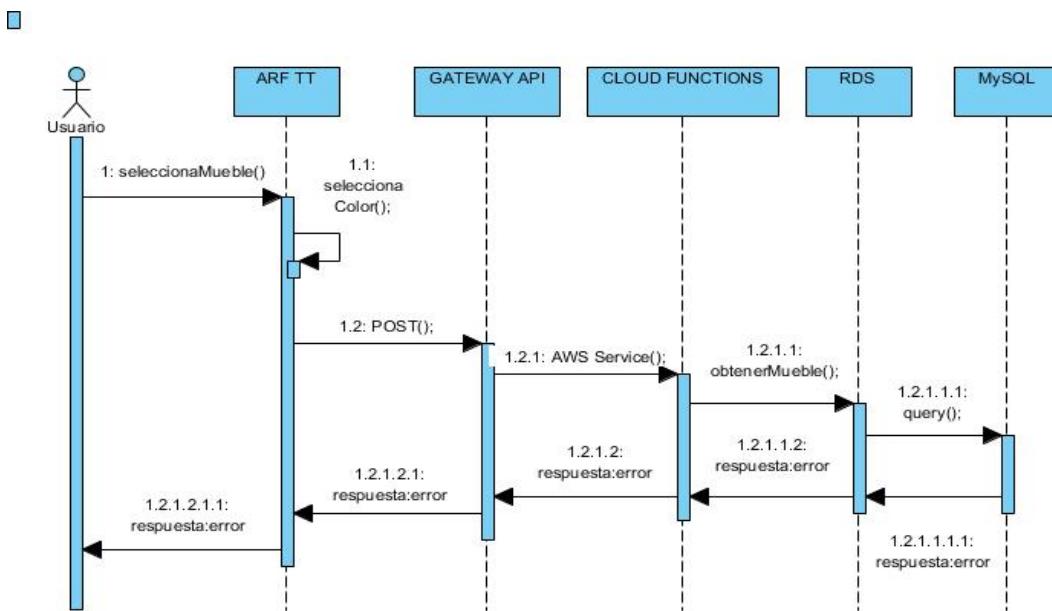


Figura 3.10: Diagrama de secuencia -Gestión de muebles.

Si ocurre un error desde el servidor se envía un código de error donde se notificara al usuario que no se encuentra disponible.

### 3.5.5 Guardar escenario

La creación y guardado de un escenario inicia cuando se comienza a capturar la realidad aumentada de un mueble, la acción que dispara este proceso será la de guardar escenario, la función tiene como parámetros el identificador del usuario, el nombre del escenario y una colección de imágenes en formato Base64. Esta función procesa las imágenes a binario para su posterior persistencia en S3 Bucket (Simple Storage Service).

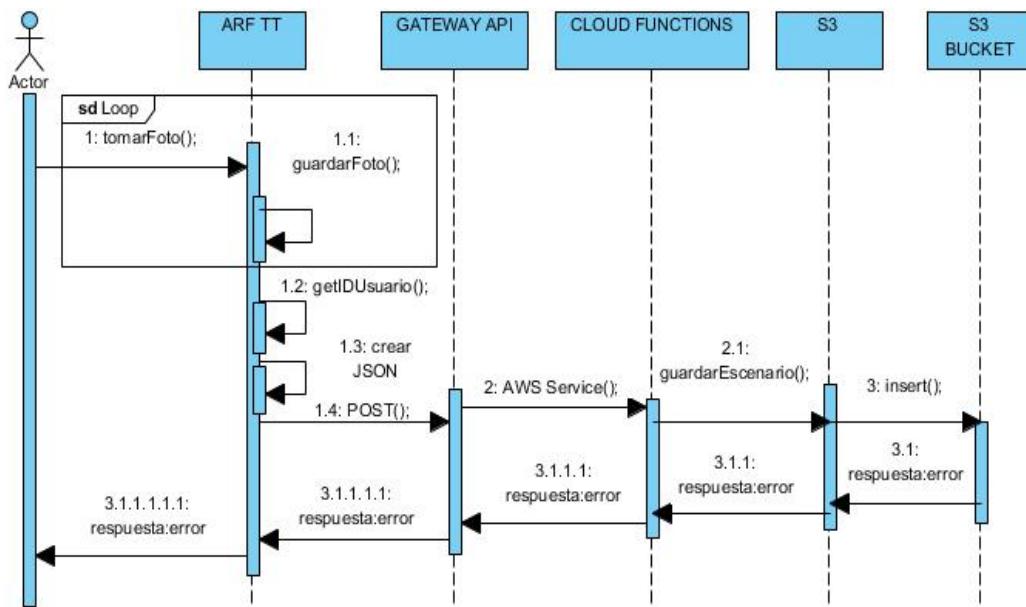


Figura 3.11: Diagrama de secuencia - Guardar escenario.

El S3 envia un mensaje de éxito o error de la persistencia, el usuario observara la notificación de guardado satisfactorio. En dado caso que la petición sea incorrecta se enviara una notificación de guardado fallido.

### 3.5.6 Visualizar escenario

La visualización de un escenario se dispara cuando el actor selecciona la opción visualizar escenario. La petición se envía por los servicios de AWS que consulta el bucket S3, este a su vez hace la consulta de la información con MySQL, si encuentra una coincidencia se obtiene los nombres y las URL de las imágenes guardadas. Esta información viaja de regreso para que sea visualizada en la pantalla de la aplicación.

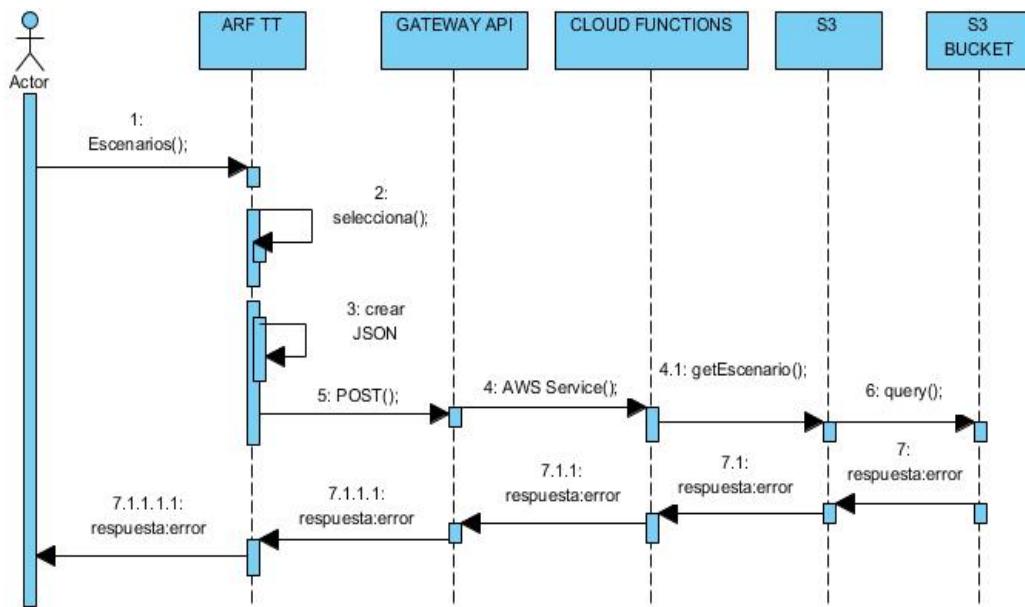


Figura 3.12: Diagrama de secuencia - Visualizar escenario.

Si la petición no encuentra alguna coincidencia dentro de S3 se envía un código de error, informando que no existen escenarios para visualizar.

# **Capítulo 4**

## **Desarrollo**

En éste capítulo se describirá el proceso de desarrollo del trabajo terminal de acuerdo a la metodología Mobile-D, de tal forma que estará divido en capítulos según las iteraciones que se vayan realizando.

Mobile-D es una metodología ágil iterativa basada en Extreme Programming. Consta de cinco etapas básicas:

- Exploración
- Inicialización
- Producción
- Estabilización
- Pruebas

Solamente durante la primera iteración se realizarán las fases "Exploración" e "Inicialización", que sirven para definir la dirección del proyecto. Posteriormente "Producción" se realiza de forma iterativa, llevándose a cabo una y otra vez hasta terminar de construir o desarrollar el sistema. Las iteraciones y el entregable de cada una serán definidos durante la iteración inicial. Finalmente en la fase "Estabilización" se integran los componentes que se hayan desarrollado y se documenta el proceso realizado en las iteraciones anteriores, para culminar con una última fase de "Pruebas" donde se asegura la calidad y funcionamiento del sistema para poder ser liberado, y en caso de que se requiera, se hagan correcciones.

### **4.1 Iteración 0**

#### **4.1.1 Exploración**

En éste apartado se definirá el alcance del proyecto, los recursos u herramientas a usar, y las personas involucradas en su desarrollo, así como las pruebas de contexto realizadas en las tecnologías que usaremos.

### Pruebas de contexto

Con el fin de conocer las capacidades de las plataforma ARCore y Vuforia, excluyendo ARKit debido a que su desarrollo es enfocado a sistemas iOS, se realizaron pruebas de contexto definiendo las siguientes especificaciones :

- Nombre del dispositivo en el que se realizó
- Fecha de prueba
- Versión de la tecnología SDK
- Versión de Android

Por otro lado cada prueba se realizó tomando en cuenta las siguientes características:

- **Posición Cardinal.**- Define si un objeto virtual puede visualizarse desde los cuatro puntos cardinales (norte, sur, este, oeste) y desde la parte superior del mismo, cuando la cámara gira alrededor del objeto mientras lo enfoca.
- **Tamaño relativo.**- Define si un objeto cambia su tamaño en el entorno virtual dependiendo de la distancia a la que se acerque o se aleje la cámara. Cuando la cámara se acerca, el objeto deberá aumentar su tamaño, y viceversa, como si se tratase de un objeto real.
- **Luminosidad.**- Define el grado de oscuridad de un objeto a determinada luz, es decir, cuando la luz en el ambiente real es alta, el objeto se verá iluminado, en caso contrario cuando haya escasa luz, el objeto se oscurecerá.
- **Superficie.**- Describe en qué superficies el objeto virtual es puesto, si fue posible su superposición en este material y qué comportamiento tiene en cada una de estas.
- **Memoria de objetos.**- Define si los objetos virtuales se conservan en la memoria cuando la cámara pierde su enfoque en ellos y los vuelve a enfocar. También describe si los objetos conservaron su posición tras el re-enfoque.
- **Capacidad máxima de objetos.**- Define el número de objetos virtuales que pueden ser mostrados en escena sin que el rendimiento de la aplicación caiga considerablemente.
- **Distancia.**- Define la distancia a la que se encuentra la cámara de un objeto virtual sin que éste desaparezca o sin que su resolución baje considerablemente.

### ARcore

Especificaciones de prueba	
<b>DISPOSITIVO</b>	Moto G6 XT1925
<b>FECHA</b>	2018/08/25
<b>VERSIÓN DE SCENEFORM SDK</b>	V1.4.0
<b>VERSIÓN DE ARCORE SDK</b>	V1.4.0
<b>VERSIÓN DE ANDROID</b>	V8.0.0 (Oreo)

Tabla 4.1: Especificaciones de prueba Arcore en Moto G6

**Posición cardinal**

El objeto virtual se pudo apreciar con claridad desde los cuatro puntos cardinales y la vista superior. El ángulo de visualización del objeto al mover la cámara cambiaba a la perfección, dando una buena percepción de realismo.

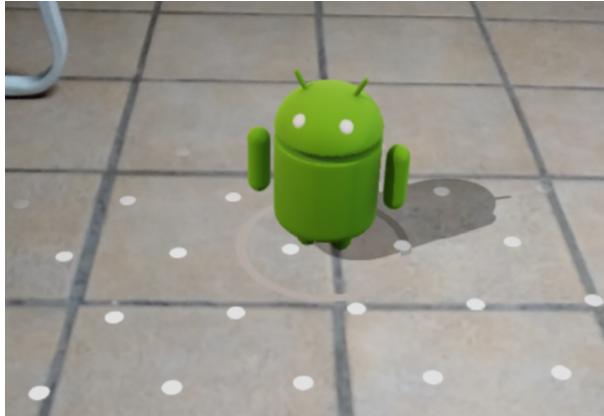


Figura 4.1: Posición Norte



Figura 4.2: Posición Sur



Figura 4.3: Posición Este

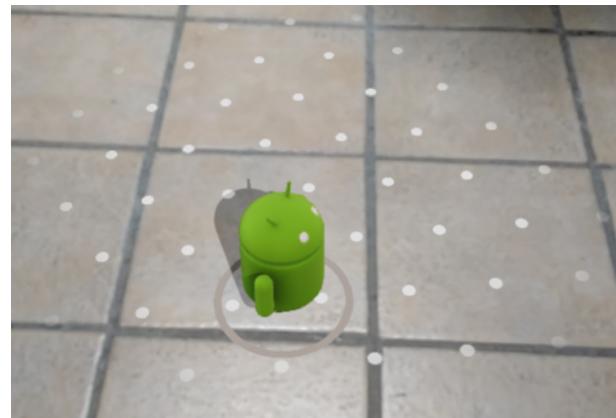


Figura 4.4: Posición Oeste

**Tamaño relativo**

Al acercar o alejar la cámara el objeto virtual variaba su tamaño de forma adecuada, como si el objeto realmente estuviera en la posición donde fue superpuesto.

**Luminosidad**

Al poner el objeto virtual en entornos con diferente cantidad de luz, la cantidad de luz en el



Figura 4.5: Objeto visto de cerca

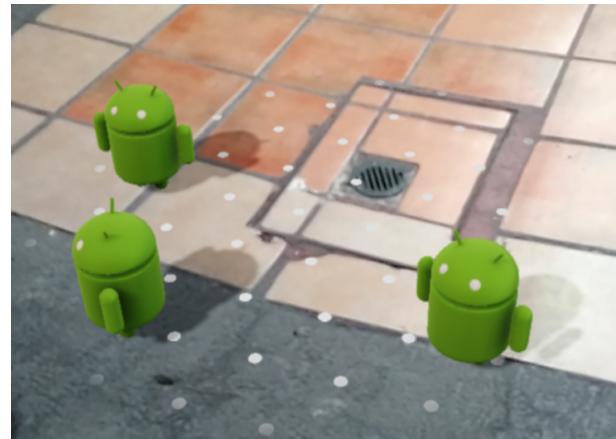


Figura 4.6: Luz alta

objeto virtual también variaba. En un entorno con ausencia casi total de luz el objeto apenas era perceptible, mientras que en un entorno con bastante luz, el objeto se veía altamente iluminado.

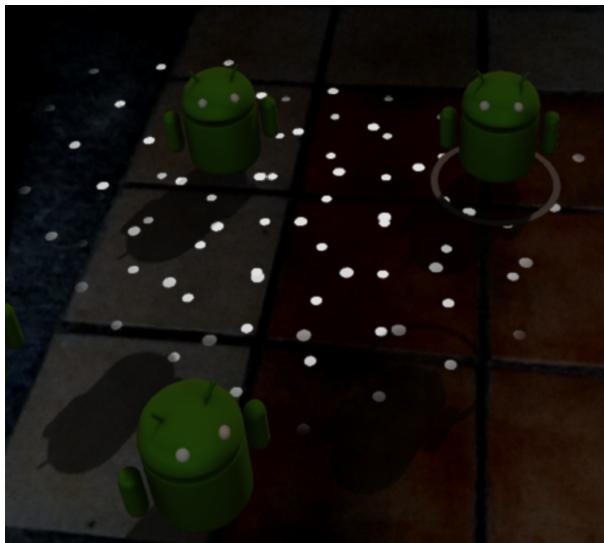


Figura 4.7: Luz baja

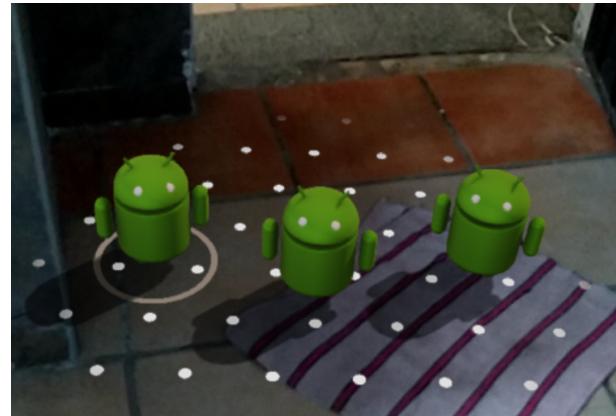


Figura 4.8: Luz media

### Superficie

Se probó posicionar el objeto virtual en cuatro superficies: concreto gris, concreto blanco, azulejo y vidrio.

Concreto gris.- El objeto se pudo posicionar a la perfección.

Concreto blanca.- El objeto no se pudo posicionar. La maya de puntos ni si quiera era detectada

en ésta superficie debido a la ausencia de texturas.

Azulejo.- El objeto se pudo posicionar a la perfección.

Vidrio.- El objeto no pudo ser posicionado en ésta superficie debido a las propiedades reflejantes que posee.

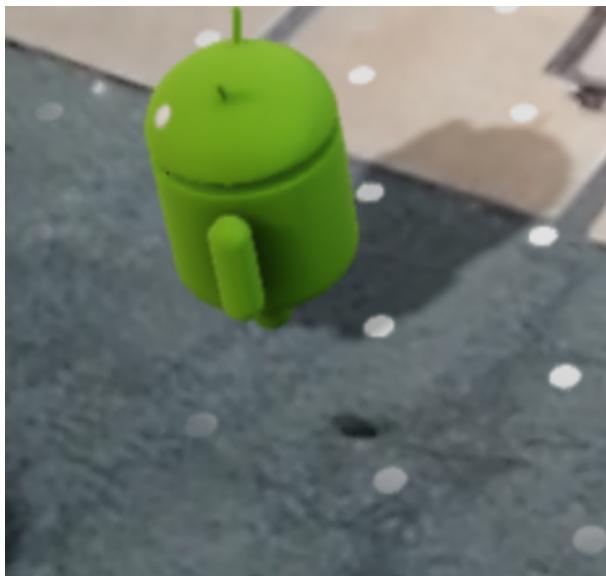


Figura 4.9: Objeto colocado en concreto

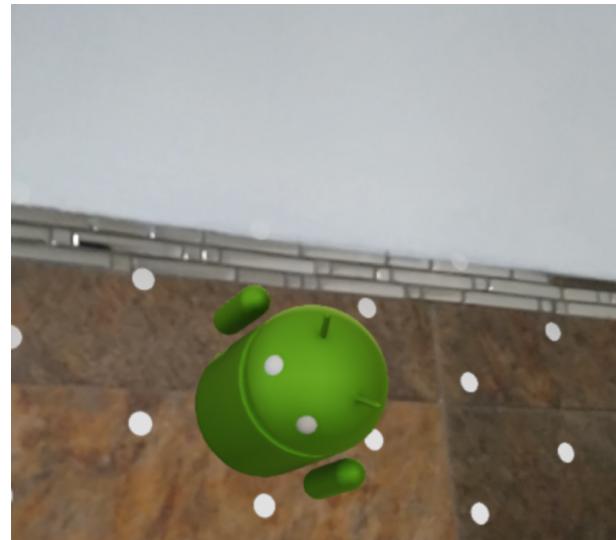


Figura 4.10: Objeto colocado en superficie blanca

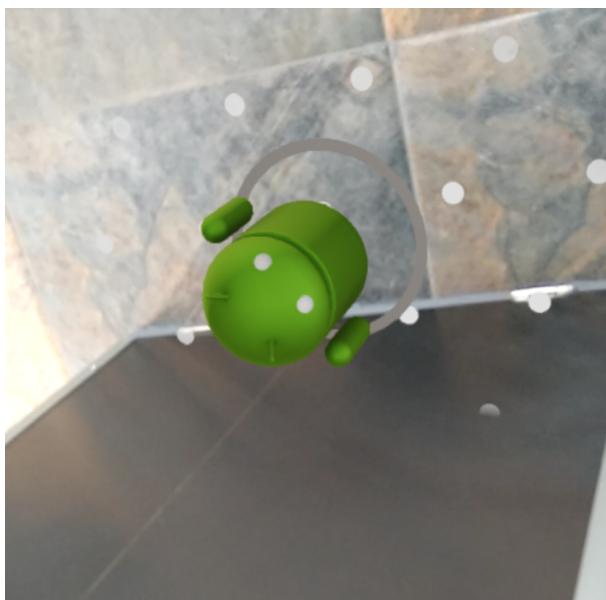


Figura 4.11: Malla de puntos no detectada en superficie negra

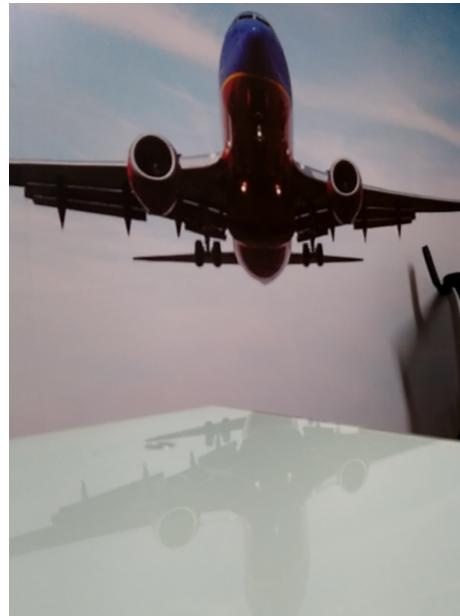


Figura 4.12: Malla de puntos no detectada en vidrio

### Memoria de objetos

Tras perder el enfoque de la cámara, al volverlo a tener, todos los objetos virtuales se volvieron a mostrar en el entorno virtual en la misma posición en la que habían sido puestos.

### Capacidad máxima de objetos

Se colocaron 100 objetos virtuales en escena sin que la aplicación perdiera rendimiento. Todo funcionaba con total fluidez.

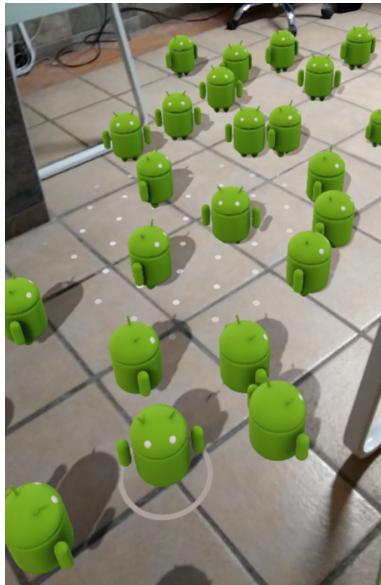


Figura 4.13: Gran cantidad de objetos en escena



Figura 4.14: Objetos puestos a gran distancia

### Distancia

Se colocó un objeto, después la cámara fue alejada hasta una distancia de **11.22m**. A esa distancia los objetos comenzaron a verse pixeleaseados, además comenzaron a desaparecer y reaparecer de forma intermitente.

### Vuforia

Especificaciones de prueba	
<b>DISPOSITIVO</b>	Moto G6 XT1925
<b>FECHA</b>	2018/08/25
<b>VERSIÓN DE SCENEFORM SDK</b>	No aplica
<b>VERSIÓN DE Vuforia SDK</b>	V7.0.50
<b>VERSIÓN DE ANDROID</b>	V8.0.0 (Oreo)

Tabla 4.2: Especificaciones de prueba Vuforia en Moto G6

### Posición cardinal

La aplicación necesita escanear una marca para poder dibujar el objeto, al colocar el objeto virtual se pudo apreciar con claridad desde los cuatro puntos cardinales y la vista superior. El objeto tiende a desaparecer y aparecer al mover la cámara restando realismo.



Figura 4.15: Posición Norte



Figura 4.16: Posición Sur



Figura 4.17: Posición Este

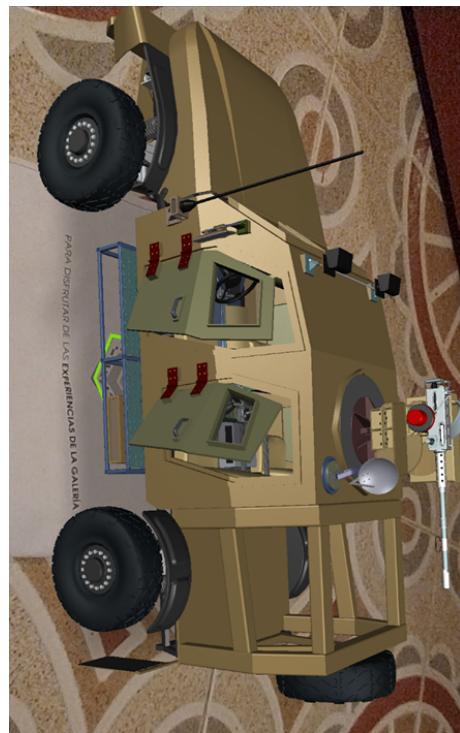


Figura 4.18: Posición Oeste

### Tamaño relativo

Al acercar o alejar la cámara el objeto virtual mantiene su tamaño original.



Figura 4.19: Objeto visto de cerca



Figura 4.20: Objeto visto de lejos

### Luminosidad

Al poner el objeto virtual en entornos con diferente cantidad de luz, el objeto no cambia y en niveles bajos de luz el objeto desaparece.

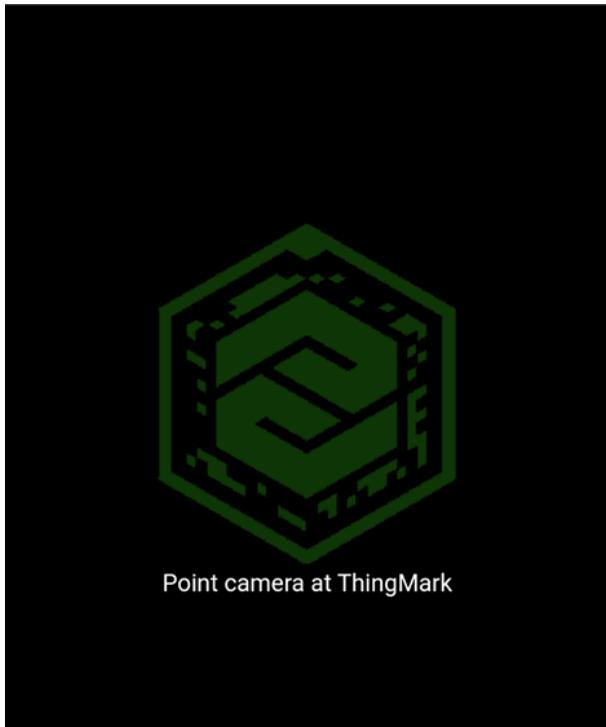


Figura 4.21: Luz baja



Figura 4.22: Luz media



Figura 4.23: Luz alta

### **Superficie**

Esta prueba no aplica en esta tecnología ya que es necesario la marca para que aparezca el objeto virtual



TO ENJOY THE GALLERY EXPERIENCES

Figura 4.24: marca de vuforia (thingmark)

### **Memoria de objetos**

El objeto no guarda su posición, dicha posición depende del lugar en donde se coloque la marca.

### **Capacidad máxima de objetos**

La cantidad de objetos en pantalla depende del número de marcas que se encuentren en el rango de la cámara.

## **Establecimiento de interesados**

### **Desarrolladores**

Para poder construir la aplicación requeriremos desarrolladores que puedan diseñar la arquitectura que tendrá la aplicación y programarla. Los tres desarrolladores que estarán trabajando en el proyecto son Gerardo Aramis Cabello Acosta, Martín Alejandro Carrillo Mendoza y Saúl del Pilar Morales. Su intervención se requerirá en todas las iteraciones del proyecto desde la I a la IX.

### **Directores**

Los directores serán los encargados de supervisar los avances del proyecto en cada iteración. También serán elementos de apoyo a los que los desarrolladores podrán acudir en caso de tener alguna duda o complicación. Su intervención se requerirá durante todo el desarrollo del proyecto desde la iteración I a la iteración IX.

### **Sinodales**

Serán quienes evaluarán el proyecto al terminar los entregables de TT I y TT II. Su intervención se requerirá al finalizar las iteraciones IV y IX.

### **Arquitecto**

Se requiere contactar a un arquitecto que tenga conocimientos en antropometría con el objetivo de poder obtener una retroalimentación del proyecto. Ésta retroalimentación nos servirá para poder mejorar la aplicación modificando, añadiendo o removiendo características. Su intervención

se requerirá en la fase de inicialización y en la fase de planeación de la iteración V, que será la fase inmediata después de presentar TT I.

### Mueblerías

Cuando el proyecto esté terminado, acudiremos a varias mueblerías para presentarlo. De esta forma las tiendas que estén interesadas y deseen que sus muebles aparezcan en el catálogo de la aplicación, se convertirán en nuestros socios de negocio. De ésta forma enfocaremos el proyecto al e-Commerce. Su intervención se requerirá tras finalizar la iteración IX, que será la última fase antes de presentar TT II.

### Alcance

Se desarrollará una aplicación móvil para dispositivos móviles que usen Android como sistema operativo en su versión 7.0 en adelante y que se encuentren dentro de la lista especificada por la *Tabla 2.4*.

El objetivo de la aplicación será lograr que el proceso de diseño de interiores sea más sencillo y rápido, esto a través de la eliminación y reducción de etapas del proceso implicado (véase *Figura 1.1*). La aplicación cumplirá las siguientes funciones:

- Tendrá un login, el cual tras la autenticación, permitirá guardar los escenarios que se vayan generando con sus respectivas fotografías.
- Tendrá una opción alternativa para iniciar la aplicación sin pasar por un proceso de autenticación. En este caso sólo se podrá generar un entorno virtual con AR pero no se podrá guardar el escenario.
- Tendrá un catálogo de muebles modelados y renderizados en 3D
- La aplicación podrá incluir hasta 100 muebles en escena al mismo tiempo.
- Será posible cambiar las texturas de los muebles puestos en escena.
- Los objetos puestos en escena tendrán posición relativa.
- Los objetos puestos en escena tendrán una variación dinámica de luz, esto es, que conforme cambie la luz en el ambiente real, también lo hará de forma virtual en los objetos superpuestos.
- La superposición de objetos no requerirá el uso hardware externo o marcadores físicos.
- Será posible mover los muebles de lugar aún después de ser puestos en escena.
- La aplicación podrá tomar fotografías de los escenarios de realidad aumentada que vaya generando el usuario.
- La aplicación permitirá asociar las fotografías tomadas a escenarios específicos. *v.g. "Diseño de interiores - Casa Loma Bonita"*

El desarrollo del proyecto será finalizado en mayo de 2019.

## Establecimiento de proyecto

El proyecto será realizado en nueve iteraciones hasta el entregable de TT II. Dado que la metodología es ágil basada en Extreme Programming, dividiremos el desarrollo en dos partes, para los entregables de TT I y TT II. La primer parte comprende desde iteración uno hasta la cuatro, al final de la cual terminaremos el entregable de TT I y realizaremos nuestra presentación. A continuación se encuentran descritas cada una de las iteraciones que involucrarán el desarrollo de TT I.

### Iteración I

En éste punto ya está realizada la primer implementación de AR con la cual se realizaron las pruebas de contexto descritas anteriormente. Por lo tanto, para ésta iteración vamos a implementar un modelo 3D propio. Usaremos Blender para el modelado y renderizado de éste elemento. Al final de ésta iteración tendremos una aplicación en Android que pueda desplegar a través de AR un modelo 3D diseñado por nosotros.

### Iteración II

En ésta iteración vamos a hacer el diseño detallado de un mueble. De igual manera el modelado y renderizado será en Blender. Al final de ésta iteración tendremos una aplicación en Android que pueda desplegar a través de AR un mueble. El entregable de ésta iteración es el equivalente al planeado en el protocolo para TT I.

### Iteración III

En ésta iteración vamos a desarrollar el backend de la aplicación para poder realizar el login, registro de cuenta, recuperación de contraseña y el guardado de los escenarios que se vayan generando en la realidad aumentada. Al final de ésta iteración se tendrá el backend totalmente funcional para conectarlo con la aplicación a través del protocolo HTTP.

### Iteración IV

En ésta iteración vamos a modificar el entregable de la iteración II, de tal forma que sea posible cambiar el color del mueble durante el tiempo de ejecución de la aplicación.

#### 4.1.2 Inicialización

En ésta fase se describirá la línea base sobre la que se desarrollará el proyecto, esto significa, definir las herramientas y dispositivos con los que se realizará el desarrollo, qué tecnologías vamos a usar y el planteamiento general del sistema.

#### Herramientas y Tecnologías:

- Amazon RDS
- Amazon S3
- Android 8.0 Oreo
- Android studio 2.19
- Arcore 1.5
- Blender 2.79
- Debían 9.5 “stretch”

- Git
- Git hub
- Java 7
- JSON
- Windows 10
- XML

**Dispositivos:**

- Laptop acer, procesador Intel core i3 6ta generación, 4GB de ram, 1TB en disco duro, sistema operativo Debián 9.5 “stretch”
- Moto G6
- Moto G6 plus

El backend de la aplicación se encontrará sobre la infraestructura de Amazon Web Services (AWS), debido a la alta escalabilidad que proporciona. La arquitectura contendrá un cluster RDS con MySQL, cinco Lambdas, un Bucket de S3 y una API Gateway (véase Figura 3.15).

Cada Lambda estará enfocada a una funcionalidad principal de la aplicación:

- **Login.**- Encargada de toda la lógica y la seguridad informática relacionada con la autenticación en la aplicación.
- **Registrar cuenta.**- Permitirá registrar una nueva cuenta en el sistema, misma que permitirá guardar escenarios.
- **Recuperar cuenta.**- En caso de que se olvide la contraseña de la cuenta, ésta Lambda contendrá la lógica para recuperar la cuenta
- **Guardar escenario.**- Permitirá almacenar las imágenes que sean enviadas. Estas imágenes serán almacenadas en un bucket de S3 y asociadas a un escenario. Esta información será almacenada en la base de datos de MySQL que está montada sobre el cluster de RDS.
- **Ver escenario.**- Ésta Lambda permitirá recuperar la información e imágenes de un escenario especificado, con el fin de poder visualizarlo desde la ubicación.

**Arquitectura:** La aplicación podrá comunicarse a toda la infraestructura de AWS por medio de la API Gateway, que sirve como puerta de enlace tanto para entrar como para salir de AWS. Ésta comunicación se realizará a través del protocolo HTTPS, dada la facilidad de uso que proporciona, además de la capa de seguridad SSL que ya proporciona AWS durante la comunicación con la API Gateway. En la figura 3.15 puede observarse un diagrama de arquitectura del sistema.

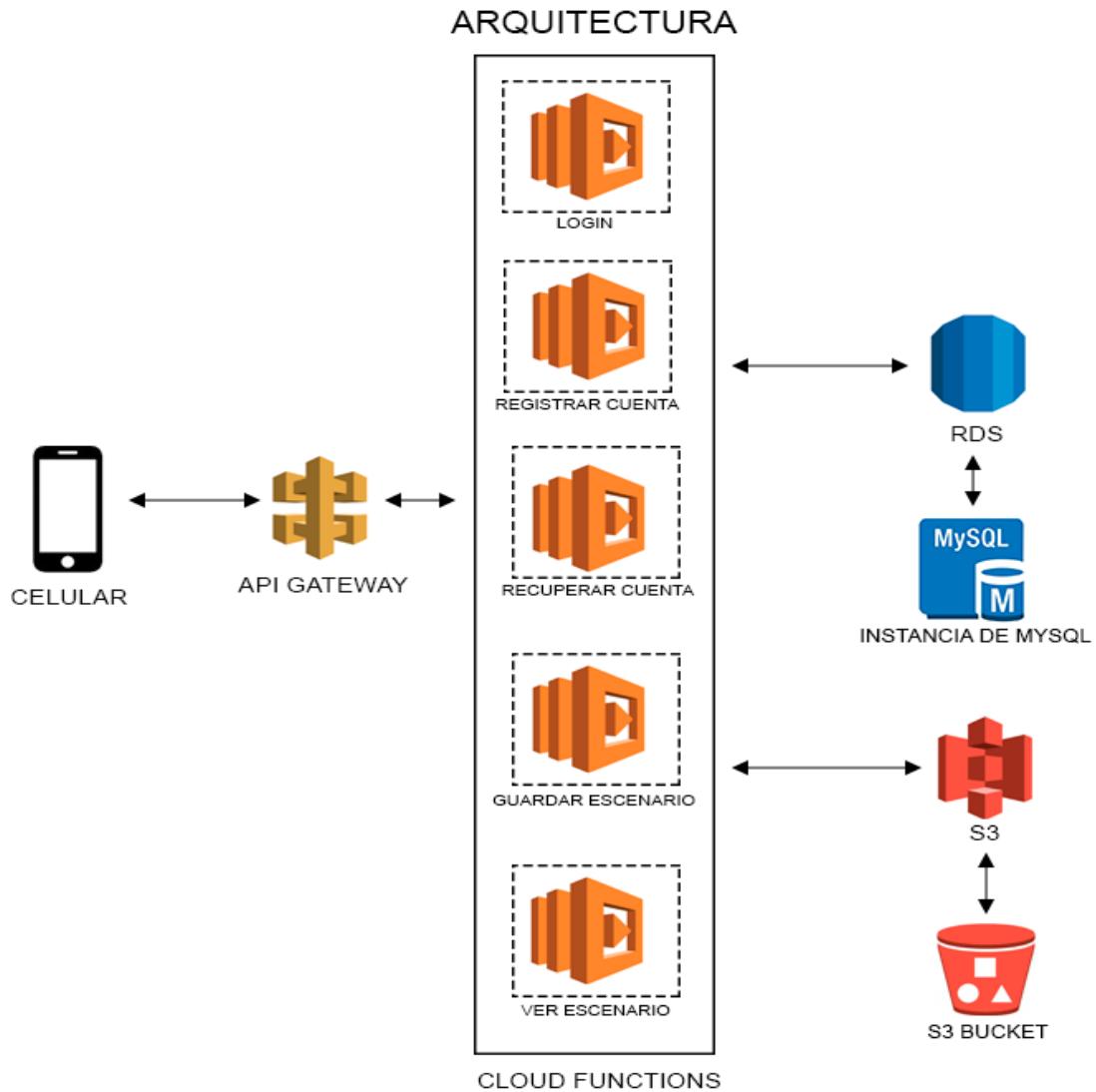


Figura 4.25: Arquitectura de Backend de ARF.

Por otro lado se requerirá una base de datos que pueda almacenar los usuarios registrados y los escenarios que estos vayan creando. Para el tipo y cantidad de información que se requiere almacenar, una base de datos relacional cumple a ésta necesidad. En la figura 3.16 describe el diseño de la base de datos que se va a usar en ARF, misma que se encuentra en el clúster Amazon RDS mostrado en la figura 3.17.

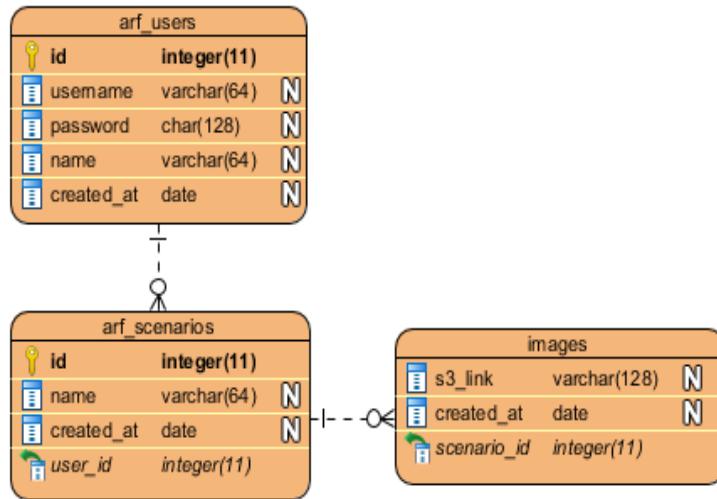


Figura 4.26: Modelo Entidad-Relación usado para la base de datos de ARF.

### Hola mundo! ARcore

Los desarrolladores de ARcore nos proporcionan un quickstart el cual utilizamos para iniciar la curva de aprendizaje de esta herramienta. Este quickstart únicamente nos proporciona una clase principal, la cual contiene la lógica necesaria para mostrar un simple renderizado.

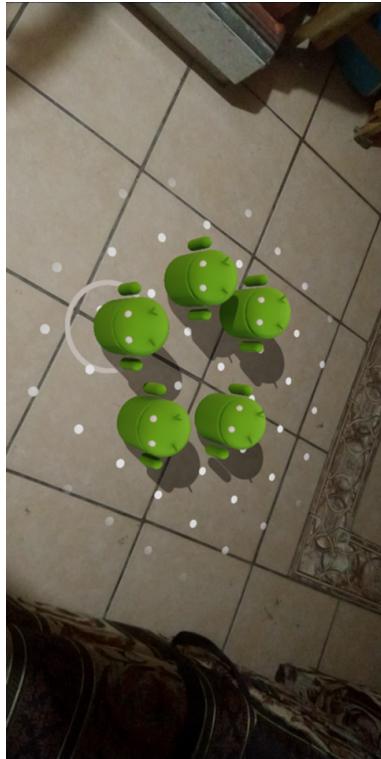


Figura 4.27: Hola mundo de ARcore

## 4.2 Iteración I

### 4.2.1 Resumen

En esta iteración se realizó el renderizado de un modelo 3d propio, así como implementarlo en la aplicación.

### 4.2.2 Desarrollo

El renderizado de un modelo 3d se realizó utilizando la herramienta blender, es dicha herramienta modelamos una simple capsula, que se exportó en dos tipos de archivos .obj y .mtl los cuales convertimos, usando Google Sceneform Tools, en un modelo entendible para ARcore.



Figura 4.28: Primer modelo 3D implementado en ARcore

## 4.3 Iteración II

### 4.3.1 Resumen

En esta iteración se modeló un mueble y se implementó para visualizarse en la aplicación.

### 4.3.2 Desarrollo

Ya entendido el funcionamiento de ARcore y como agregar nuevos modelos, en esta iteración se modelo el primer mueble, al igual que en la iteración pasada se obtuvieron dos tipos de archivos y se renderizaron para mostrarse en la aplicación.



Figura 4.29: Mueble modelado en 3D

## 4.4 Iteración III

### 4.4.1 Resumen

En ésta iteración se realizó el frontend del login de ARF así como el backend para los módulos "LOGIN", "REGISTRAR CUENTA" y "GUARDAR ESCENARIO" de acuerdo a la arquitectura descrita en la figura 3.15.

### 4.4.2 Desarrollo

El desarrollo de los módulos de backend mencionados se realizó sobre Lambdas de AWS. Cada módulo tiene su propia función. El objetivo de hacerlo de esta forma es brindarle escalabilidad al sistema pues el desarrollo se logra de forma modular.

Cada uno de estos módulos es accesible a través del protocolo HTTP gracias a la API Gateway. La API Gateway es un módulo de Amazon Web Services que permite que los recursos de AWS sean accesibles desde el exterior, en éste caso, permite que la aplicación en Android pueda acceder a éstas funciones con una simple petición.

Las Lambdas y la API Gateway están configuradas para que a través del método POST y el formato JSON la aplicación pueda comunicarse y usar las funciones almacenadas en la nube.

El desarrollo de cada función en la nube se realizó con NodeJS v8.10.0. A continuación se describe cada módulo.

## Login

Se encarga de la funcionalidad del login de la aplicación. En la figura 3.16 se puede observar el diagrama del proceso de login.

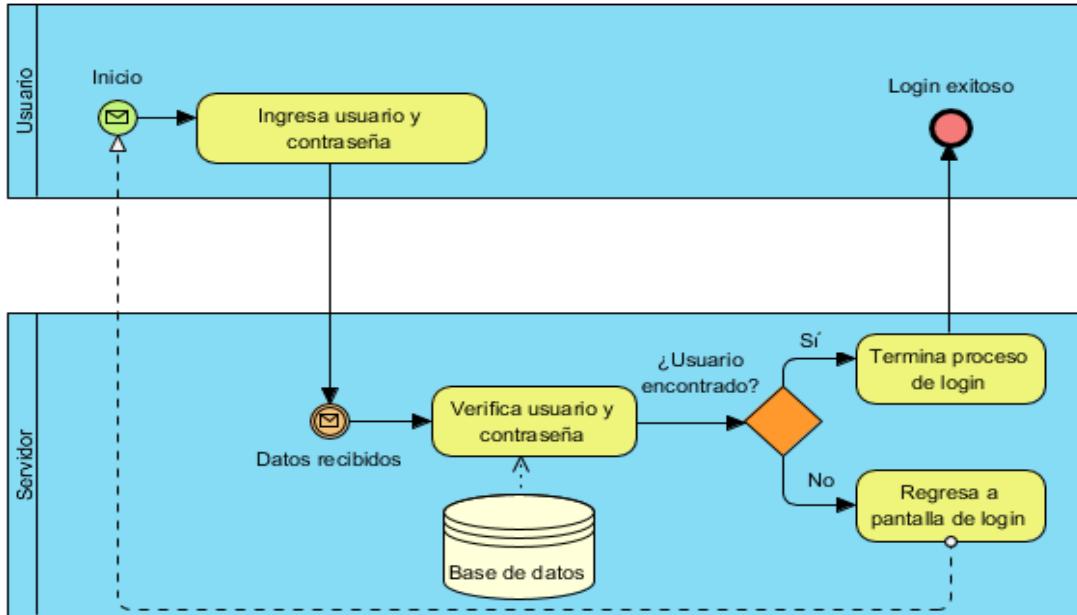


Figura 4.30: Diagrama de proceso de Login.

Ésta función recibe dos parámetros: usuario y contraseña. Posteriormente inicia un primer proceso de validación, el cual consiste en la obtención del hash criptográfico de la contraseña a través de SHA-512. Ese hash junto con el usuario es buscado en una base de datos relacional donde se encuentran los usuarios, esta base de datos está almacenada en Amazon RDS, que tiene un motor MySQL 5.5. Si encuentra alguna coincidencia entonces significa que el usuario existe.

Así, la función regresa en formato JSON el usuario junto a su hash para que la aplicación realice una segunda etapa de verificación (ver figura 3.17).

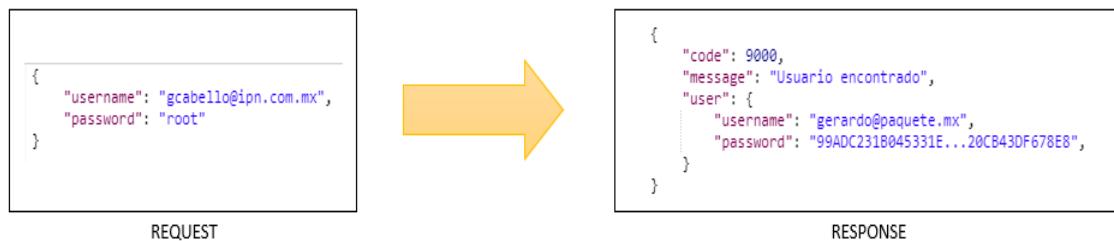


Figura 4.31: Flujo de información de login exitoso.

Por otro lado si los datos ingresados son incorrectos, la función responde con un código de error (-1000), que es el asociado a datos de login incorrectos (ver figura 3.18).

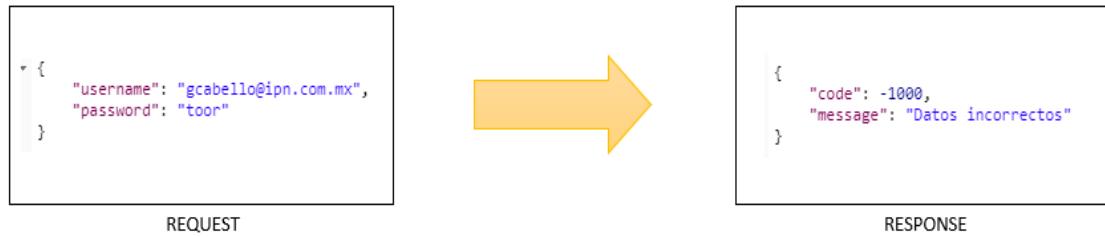


Figura 4.32: Flujo de información de login fallido.

## Registro

Ésta función se encarga de registrar un nuevo usuario en la base de datos para que posteriormente pueda iniciar sesión en la aplicación. El modelado de este proceso se puede apreciar en la figura X.X.

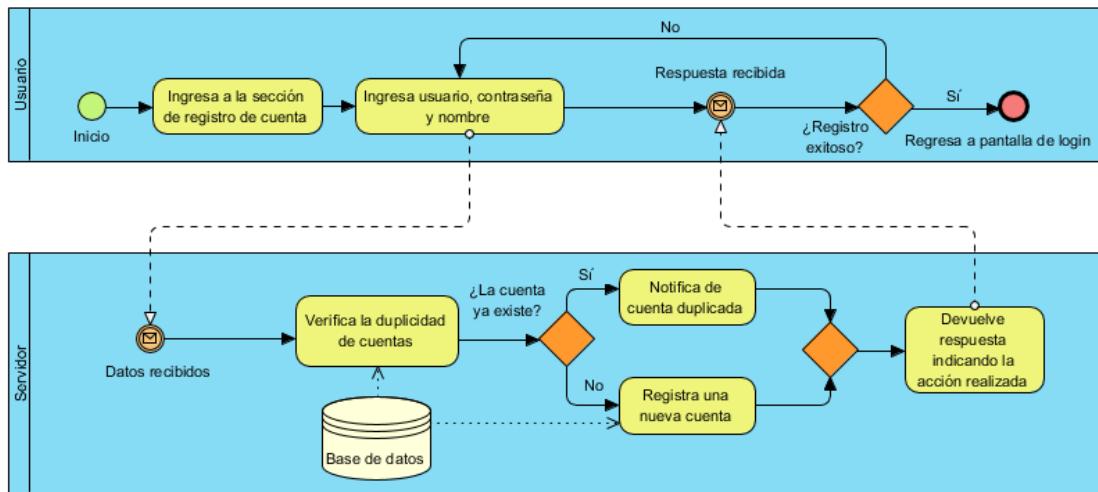


Figura 4.33: Proceso de registro de nuevo usuario.

Funciona de manera similar a la Lambda del Login. Recibe tres parámetros en formato JSON a través de un request con método POST, estos parámetros son: nombre de usuario, contraseña y primer nombre.

Esta función se encarga de tomar estos datos y registrar un nuevo usuario con ellos. La contraseña que se guarda en la base de datos es el hash criptográfico con SHA-512, la cual será usada en el primero proceso de validación del inicio de sesión. En la figura 3.19 se observa el flujo de información para esta función.

## Guardar escenario

El objetivo de ésta función es poder almacenar los escenarios que vayan generando los usuarios para que después puedan ser visualizados desde la aplicación. En la figura X.X se puede apreciar el modelado del proceso de almacenamiento de escenario.

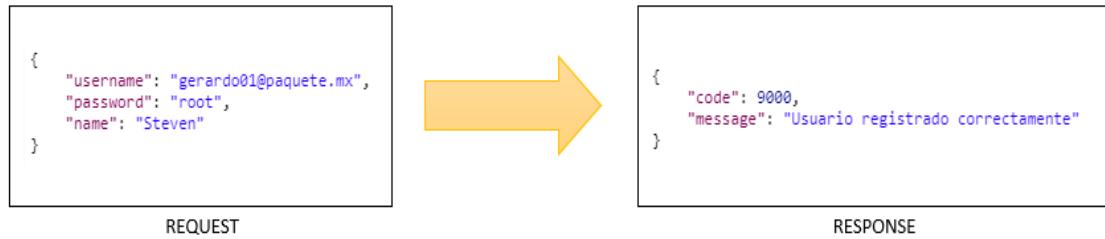


Figura 4.34: Flujo de información de registro de usuario.

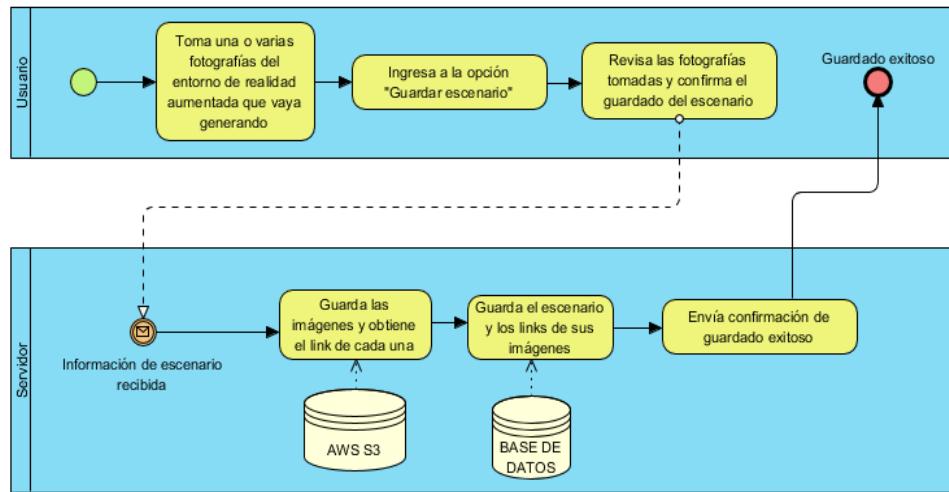


Figura 4.35: Proceso de guardado de escenario.

La función recibe como parámetros el id del usuario al que se va a asociar el escenario, el nombre del escenario y un array de imágenes en formato **Base64**. Una vez recibidos estos atributos, la función se encarga de convertir las imágenes en formato binario para después guardarlas en un Bucket de AWS S3 (Simple Storage Service) que no es más que un contenedor de archivos. Una vez guardadas las imágenes en el Bucket, se obtienen los nombres y las URL de las imágenes guardadas. Los nombres sirven para guardar el escenario en una base de datos, y posteriormente poder recuperarlos del Bucket a través del nombre. Las URL de las imágenes se envían con el objetivo de poder visualizar el escenario guardado en el dispositivo móvil obteniendo las imágenes directamente del almacenamiento en la nube. El proceso de devolver las URL de las imágenes será implementado también en la función de "VER ESCENARIO" con el objetivo de hacer que la respuesta a la petición sea más ligera, y con ello más rápida. El flujo de esta función se puede ver en la figura 3.20.

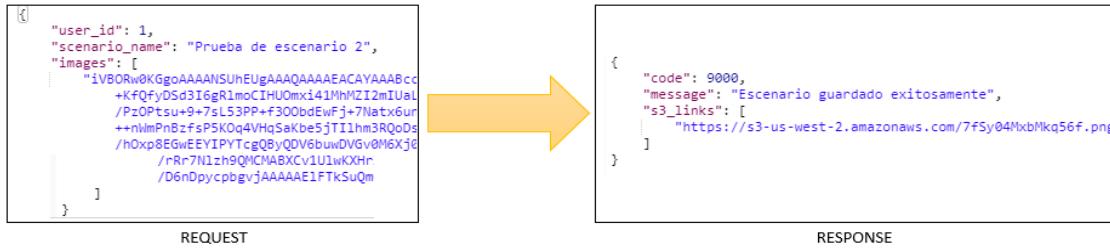


Figura 4.36: Flujo de información de guardado de escenario.

## 4.5 Iteración IV

### 4.5.1 Resumen

En esta iteración se modelaron dos muebles más, tres botones para cambiar de modelo e implementamos un botón para tomar fotografía del escenario.

### 4.5.2 Desarrollo

Se modelaron dos muebles más, utilizando la misma herramienta, obteniendo los mismos archivos y realizando el proceso de las iteraciones I y II.

En la parte inferior de la aplicación se encuentran tres diseños de muebles, que funcionan como botones para seleccionar el objeto 3D que aparecerá en pantalla.

En la parte superior derecha de la aplicación tenemos ubicado un botón el cual toma una screenshot de la escena y la guarda en el dispositivo.



Figura 4.37: Aplicación: modelos en la parte inferior y botón de tomar foto parte superior

Al presionar el botón "tomar fotografía" se guardara una imagen de la escena, en formato jpg y aparecerá un mensaje para abrir la ubicación de la imagen.

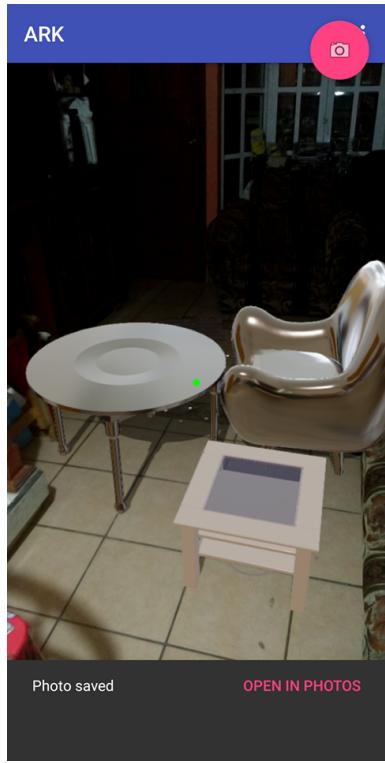


Figura 4.38: Pantalla de visualización de fotografías



Figura 4.39: Pantalla de registro nuevo usuario

Las interfaz que se muestra es temporal, se realizó de esta manera para demostrar requerimientos funcionales, la interfaz final será como se muestra en los Mockup en la sección de Anexos

# Capítulo 5

## Anexos

Dieño de interfaces de usuario



Figura 5.1: UI1 - Pantalla inicial de autenticación



Figura 5.2: UI2 - Pantalla principal sin autenticación

CLASIFICACIÓN A	CLASIFICACIÓN B	MUEBLES
TIPO A1	TIPO B1	
TIPO A2	TIPO B2	
TIPO A3	TIPO B3	
TIPO A4	TIPO B4	
TIPO A5	TIPO B5	
TIPO A6	TIPO B6	
TIPO A6	TIPO B6	
TIPO A6	TIPO B6	

Figura 5.3: UI3 - Menú desplegable de muebles (catalogo)



Figura 5.4: UI4 - Pantalla principal tras autenticación

Guardar escenario

Nombre Escenario

Guardar Cancelar

Figura 5.5: UI5 - Pantalla de almacenamiento de escenario



Figura 5.6: UI6 - Pantalla de visualización de escenarios

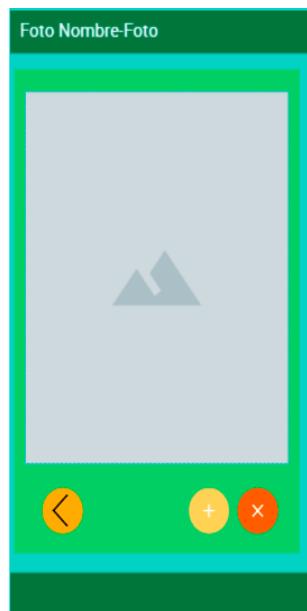


Figura 5.7: UI7 - Pantalla de visualización de fotografías



Figura 5.8: UI8 - Pantalla de registro nuevo usuario

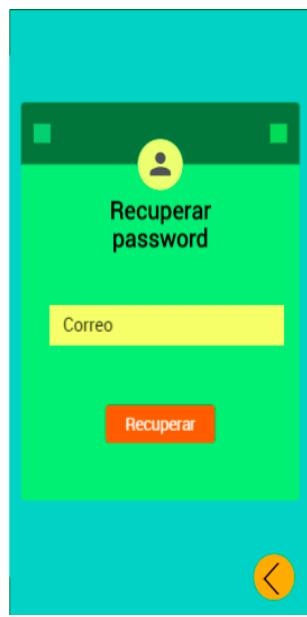


Figura 5.9: UI9 - Pantalla de recuperación de contraseña 1

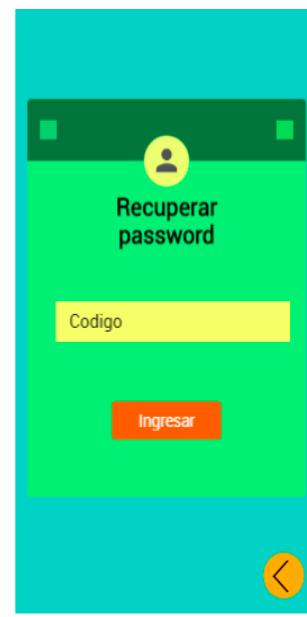


Figura 5.10: UI10 - Pantalla de recuperación de contraseña 2

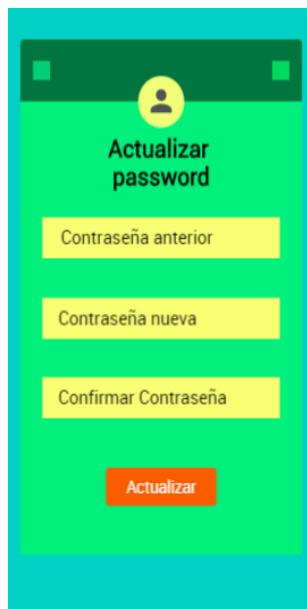


Figura 5.11: UI11 - Pantalla de actualización de contraseña

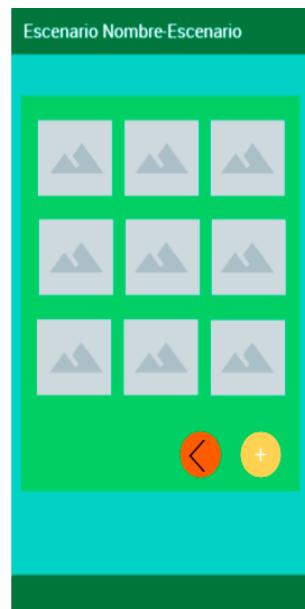


Figura 5.12: UI12 - Pantalla de visualización de imágenes de escenario

# Bibliografía

- [1] MONTES DE OCA, IRINA y RISCO, LUCÍA, "Apuntes de diseño de interiores", *Principios básicos de escalas, espacios, colores y más*, primera edición, ECOE EDICIONES
- [2] "FAQ for designers", *Whats is Interior Design*, Interior Design Legislative Coalition of Pennsylvania (IDLCPA), 2011. Recuperado de: <https://www.idlcpa.org/forms/resources/FAQforDesigners.pdf>
- [3] TAKAHASHI, YOSHIYUKI y MIZUMURA, HIROKO, "Augmented Reality Based Environment Design Support System for Home Renovation", Toyo University, Department of Human Environment Design, Faculty of Human Life Design, Oka 48-1, Asaka-shi, Saitama, 351-8510 Japan
- [4] SILTANEN, SANNI "Diminished reality for augmented reality interior design", Springer-Verlag Berlin Heidelberg 2015. Publicado el: 30 de Noviembre de 2015. DOI: 10.1007/s00371-015-1174-z DOI 10.1007/978-3-319-54502-8
- [5] DONGGANG YU1, JESSE SHENG JINA, "A Useful Visualization Technique: A Literature Review for Augmented Reality and its Application, limitation and future direction", primera edición, School of Design, Communication and Information Technology, The University of Newcastle, Callaghan, NSW 2308, Australia
- [6] BUERLI, MIKE and MISSLINGER, STEFAN, "Introducing ARKit", *Augmented Reality for iOS. Session 602* Recuperado de: [https://devstreaming-cdn.apple.com/videos/wwdc/2017/602pxa6f2vw71ze/602/602\\_introducing\\_arkit\\_augmented\\_reality\\_for\\_ios.pdf?dl=1](https://devstreaming-cdn.apple.com/videos/wwdc/2017/602pxa6f2vw71ze/602/602_introducing_arkit_augmented_reality_for_ios.pdf?dl=1)
- [7] STRANDMARK, PETTER, "Augmented reality with the ARToolKit", Recuperado de: <http://www.maths.lth.se/matematiklth/personal/petter/rapporter/artoolkit4.pdf>
- [8] FUSTER ANDÚJAR, FRANCISCO DE ASÍS, "Aplicación Android de realidad aumentada para mostrar imágenes históricas de lugares turísticos de interés", Tesis. Escola Tècnica Superior d'Enginyeria Informàtica Universitat Politècnica de València, Valencia, España, 2014.
- [9] MONTEIRO, PAULA and NAGELE, ALEKSANDRA, "Wikitude" COMPANY OVERVIEW, Recuperado de: <https://s3-eu-west-1.amazonaws.com/wikitude-web-hosting/static-website/2017/07/01728364/Media+page+presentation+-+Wikitude.pdf>
- [10] ASRUL SANI, NISFU, "Google ARCore", Lab Langit 9 - PENS 20-22 de noviembre de 2017. Recuperado de: <http://dhoto.lecturer.pens.ac.id/training/arcore/SONI-ARCORE.pdf>

- [11] KALEDA, YULIYA, "Add Reality to App with ARCore". Recuperado de: [https://downloads.ctfassets.net/2grufn031spf/7AyZDCxxIs0AoOgwOsy0m4/26f89f9d2f346dd3c6a55f5d3e2de96e/Yuliya\\_Kaleda\\_Add\\_Reality\\_To\\_App\\_with\\_ARCore.pdf](https://downloads.ctfassets.net/2grufn031spf/7AyZDCxxIs0AoOgwOsy0m4/26f89f9d2f346dd3c6a55f5d3e2de96e/Yuliya_Kaleda_Add_Reality_To_App_with_ARCore.pdf)
- [12] GRAHN, IVAR, "The Vuforia SDK and Unity3D Game Engine", *Evaluating performance on Android Devices*. Linköping University — Department of Computer and Information, Science Bachelor thesis, 16 ECTS, Computer Science 2017, LIU-IDA/LITH-EX-G-17/059-SE
- [13] VIOLA, ROMINA (11 de mayo de 2016), "Cómo Elegir la Paleta de Colores", *Parte I: Entender el Color*. Spanish Community Champion — Piktochart. [Online] Recuperado de: <https://piktochart.com/es/blog/como-elegir-la-paleta-de-colores-parte-entender-el-color/>
- [14] "ARCore Overview", 2 de agosto de 2018. Recuperado de: <https://developers.google.com/ar/discover/>
- [15] HSU, PEI-HSIEN, HUANG, SHENG-YANG y LIN, BAO-SHUH "Smart-Device-Based Augmented Reality (SDAR) Models to Support Interior Design: Rethinking "Screen" in Augmented Reality", National Chiao Tung University, Taiwan
- [16] "Wikitude Products", 29 de agosto de 2018. Recuperado de: <https://www.wikitude.com/products/wikitude-sdk/>
- [17] "Artoolkit Documentation", 29 de agosto de 2018. Recuperado de: <https://www.hitl.washington.edu/arToolkit/documentation/devframework.htm>
- [18] "OpenGL Documentation", 30 de agosto de 2018. Recuperado de: <https://www.opengl.org/documentation/>
- [19] "OpenGL Glut", 30 de agosto de 2018. Recuperado de: <https://www.opengl.org/resources/libraries/glut/spec3/spec3.html>
- [20] "ARKit Developer", 2 de Septiembre de 2018. Recuperado <https://developer.apple.com/arkit/>
- [21] "ARKit Documentation", 3 de Septiembre de 2018. Recuperado <https://developer.apple.com/documentation/arkit/>
- [22] PEDDIE, JON "Augmented Reality Where we will all live", Springer International Publishing AG 2017. Publicado el: 19 de Abril del 2017. DOI 10.1007/978-3-319-54502-8
- [23] "Vuforia Supported-devices", 4 de Septiembre de 2018. Recuperado <https://library.vuforia.com/articles/Solution/vuforia-fusion-supported-devices.html>
- [24] Harris, C., (2006), dictionary of architecture and construction (4<sup>a</sup> ed.). New York, EU, McGraw-Hill.
- [25] Medina, V. E., (2003). Forma y composición en la arquitectura desconstructivista (Tesis doctoral). Escuela Técnica Superior de Arquitectura de Madrid, Madrid, Esp.
- [26] Kathryn, T.,(2000), feng shui habitación por habitación (12<sup>a</sup> ed.). ES, Urano.

- [27] FURHT, BORKO "Handbook of Augmented Reality",Department of Computer and Electrical Engineering and Computer Science,Florida Atlantic University (),Glades Road 777 33431 Boca Raton, Florida USA,DOI 10.1007/978-1-4614-0064-6
- [28] "The new power tool for home improvement", 22 de Octubre del 2018. Recuperado <https://canvas.io/>
- [29] "Amazon.com.mx" 22 de Octubre. Recuperado de 2018 <https://www.amazon.com.mx/>
- [30] "Fingo. Furniture. Try before you buy!" 22 de Octubre 2018 <https://itunes.apple.com/us/app/fingo.vybor-mebeli-katalog/id845105741>
- [31] "Amazon.com.mx" 22 de Octubre. Recuperado de 2018 <https://www.amazon.com.mx/>