

**DESARROLLO DE UNA APLICACION DE REALIDAD AUMENTADA PARA  
DISPOSITIVOS MOVILES ANDROID, QUE REFUERCE EL APRENDIZAJE DEL  
IDIOMA INGLES NIVEL A1 EN NIÑOS DE 10 A 12 AÑOS DE EDAD**

**Trabajo de grado para optar por el título de Ingeniero de Sistemas**

**WILLIAM GABRIEL SANCHEZ ACOSTA**

**UNIVERSIDAD DE CUNDINAMARCA EXTENSIÓN CHIA**

**PROGRAMA DE INGENIERIA DE SISTEMAS**

**FACULTAD DE INGENIERÍA**

**2019**

**DESARROLLO DE UNA APLICACION DE REALIDAD AUMENTADA PARA  
DISPOSITIVOS MOVILES ANDROID, QUE REFUERCE EL APRENDIZAJE DEL  
IDIOMA INGLES NIVEL A1 EN NIÑOS DE 10 A 12 AÑOS DE EDAD**

**WILLIAM GABRIEL SANCHEZ ACOSTA**

**561212233**

**DIRECTOR**

**ESP. ING. EDISON GUSTAVO CAÑON VARELA**

**UNIVERSIDAD DE CUNDINAMARCA EXTENSIÓN CHIA**

**PROGRAMA DE INGENIERÍA DE SISTEMAS**

**FACULTAD DE INGENIERIA**

**2019**

## **AGRADECIMIENTOS**

Agradezco a la Universidad de Cundinamarca por haberme brindado la oportunidad de hacer parte de ella y formarme como un profesional, a todos los docentes de la facultad de ingeniería de la universidad por proveerme sus conocimientos y ayudarme a llegar al punto en el que me encuentro que es el de culminar mi carrera profesional. También agradezco al Ingeniero Edison Gustavo Cañón Varela por todo el apoyo y las asesorías brindadas durante el periodo de desarrollo de este proyecto, de igual manera, a los profesores Jorge Paramo Fonseca & Misael Perilla Benítez por sus colaboraciones y críticas constructivas respecto al desarrollo del proyecto.

## **DEDICATORIA**

Dedico este proyecto primeramente a Dios y a mis padres quienes estuvieron presentes en los momentos buenos como también en los momentos difíciles, quienes me brindaron su apoyo incondicional, sus consejos y su comprensión durante toda mi carrera. También le dedico este proyecto a mi pareja, la cual siempre estuvo a mi lado brindándome su amor y apoyo durante todo este ciclo estudiantil. Por último y no menos importante, le dedico este trabajo de grado a mis compañeros de estudio y familiares, quienes de una u otra manera me brindaron buenas energías y ayudas que en dado momento fueron de gran utilidad durante el lapso de mi carrera.

## RESUMEN

En el presente documento se explica el desarrollo de la aplicación English AR, una aplicación diseñada para niños de 10 a 12 años la cual permite la enseñanza y aprendizaje del vocabulario del inglés nivel A1 utilizando la realidad aumentada como instrumento de apoyo.

La realidad aumentada es una tecnología que tiene una gran aceptación por parte de los usuarios al día de hoy, su aplicación ha sido exitosa en diversos campos como es el caso de la educación, gracias a que ésta permite crear nuevas experiencias interactivas mediante la combinación del mundo real con elementos virtuales, enriqueciendo la experiencia visual y mejorando la calidad de comunicación con la información que se presenta.

Para la elaboración de este proyecto fue fundamental la implementación de una metodología de desarrollo ágil como lo es la metodología Mobile-D, una metodología que destaca por su facilidad, agilizando la elaboración de los diferentes procesos o tareas propuestas en el proyecto. A través del uso de esta metodología se logró estructurar todo el desarrollo del proyecto de una manera más organizada, cumpliendo con cada entregable propuesto en cada una de las fases que la componen.

Mediante el uso de herramientas para el desarrollo de aplicaciones móviles como lo es Unity junto con la SDK de Vuforia, además de incluir modelos en 3D diseñados en Blender, archivos multimedia de imagen creados en GIMP, y archivos de audio elaborados con AWS Polly, se logró construir una aplicación interactiva que hace uso de todos estos recursos para enseñar las palabras de diferentes temáticas del vocabulario inglés a los niños dentro y fuera de las aulas de clase.

**Palabras Clave:** Android, Aplicación móvil, Inglés, Realidad Aumentada, Multimedia, Unity, Vuforia.

## ABSTRACT

This document explains the development of the English AR application, an application designed for children from 10 to 12 years which allows the teaching and learning of English vocabulary level A1 using Augmented Reality as a support tool.

Augmented Reality is a technology that has a great acceptance by users today, its application has been successful in various fields such as education, thanks to it allows creating new interactive experiences by combining the real world with virtual elements, enriching the visual experience and improving the quality of communication with the information presented.

For the elaboration of this project was fundamental the implementation of an agile development methodology such as the Mobile-D methodology, a methodology that stands out for its ease, speeding up the elaboration of the different processes or tasks proposed in the project. Through the use of this methodology it was possible to structure the entire development of the project in a more organized way, complying with each deliverable proposed in each of the phases that compose it.

Through the use of tools for the development of mobile applications such as Unity together with the Vuforia SDK, besides including 3D models designed in Blender, multimedia image files created in GIMP, and audio files elaborated with AWS Polly, it was possible to build an interactive application that makes use of all these resources to teach the words of different topics of the English vocabulary to the children inside and outside the classrooms.

**Keywords:** Android, Augmented Reality, English, Mobile Application, Multimedia, Unity, Vuforia.

## TABLA DE CONTENIDO

<b>Tabla de contenido</b>	<b>Pág.</b>
<b>LISTA DE FIGURAS</b> .....	10
<b>LISTA DE CUADROS</b> .....	14
<b>CAPÍTULO 1</b> .....	15
<b>INTRODUCCIÓN</b> .....	15
<b>1. PROBLEMA</b> .....	16
1.1 Planteamiento del problema .....	16
1.2 Formulación del problema.....	17
<b>2. OBJETIVOS</b> .....	18
2.1 Objetivo general.....	18
2.2 Objetivos específicos .....	18
<b>3. ALCANCES Y LIMITACIONES</b> .....	19
3.1 Alcances .....	19
3.2 Limitaciones .....	20
3.2.1 Técnicas.....	20
3.2.2 Tecnológicas.....	20
<b>4. JUSTIFICACION</b> .....	21
<b>5. LINEA DE INVESTIGACION</b> .....	23
<b>CAPITULO 2</b> .....	24
<b>6. MARCO TEORICO</b> .....	24
6.1 Marco Referencial.....	24
6.1.1 MOW (Matching Objects and Words) .....	24
6.1.2 TeachAR .....	25
6.1.3 Usando la Realidad Aumentada para Enseñar Vocabulario en inglés a Estudiantes de Kindergarten. ....	26
6.1.4 Phonic Tricksters .....	27
6.1.5 Letter Alive.....	28
6.1.7 ReinAR App.....	29
6.2 Marco conceptual.....	31
6.2.1 Animación 3D.....	31
6.2.2 Rigging .....	31

6.2.3	Aplicación móvil.....	32
6.2.4	Aprendizaje.....	33
6.2.5	Banco de imágenes .....	33
6.2.6	Blender .....	33
6.2.7	Android .....	34
6.2.8	Desarrollo web.....	34
6.2.9	Educación moderna.....	34
6.2.10	Herramientas UML .....	34
6.2.11	Interfaz de usuario .....	34
6.2.12	Lenguaje de programación .....	35
6.2.13	Modelado 3D.....	35
6.2.14	Motor gráfico.....	35
6.2.15	Open Source .....	36
6.2.16	Software educativo.....	36
6.2.17	Texturizado.....	37
6.2.18	Tecnologías de la información y comunicación (TIC) .....	37
6.3	Marco ingenieril.....	37
6.3.1	Síntesis de habla .....	37
6.3.2	Amazon Web Services (AWS) Polly .....	38
6.3.3	Android .....	38
6.3.4	Blender .....	38
6.3.5	C#.....	41
6.3.6	GIMP.....	41
6.3.7	HTML .....	42
6.3.8	CSS (Cascading Style Sheets) .....	42
6.3.9	Unity3D.....	42
6.3.10	Visual Studio .....	44
6.3.11	Realidad Aumentada.....	44
6.3.12	Códigos QR.....	47
6.3.13	Vuforia .....	49
6.3.14	SDK para Unity.....	50
<b>CAPITULO 3</b>	.....	<b>51</b>

<b>7. METODOLOGIA</b> .....	51
<b>8. DESARROLLO DEL PROYECTO</b> .....	53
8.1 Exploración.....	53
8.2 Inicialización .....	64
8.2.1 Cronograma de actividades .....	64
8.2.2 Desarrollo de interfaces.....	65
8.2.3 Adquisición de recursos multimedia.....	66
8.2.4 Modelado y texturizado de los objetos en 3D .....	66
8.2.5 Creación de escenarios .....	67
8.2.6 Desarrollo de código .....	67
8.3 Producción.....	67
8.3.1 Desarrollo iterativo: Etapa 1. Desarrollo de interfaces .....	67
8.3.2 Desarrollo iterativo: Etapa 2. Adquisición de recursos multimedia .....	75
8.3.3 Desarrollo iterativo: Etapa 3. Modelado y texturizado de modelos 3D.....	79
8.3.4 Desarrollo Iterativo: Etapa 4. Creación de escenarios .....	86
8.3.5 Desarrollo Iterativo: Etapa 5. Desarrollo de código .....	92
8.4 Estabilización.....	94
<b>9. TESTER</b> .....	100
9.1 Pruebas de caja blanca .....	100
9.1.1 Prueba de reconocimiento de Códigos QR .....	100
9.1.2 Prueba de interacción del usuario con elementos proyectados .....	101
9.2 Pruebas de caja negra .....	102
9.2.1 Pruebas de funcionamiento .....	102
9.2.2 Pruebas de experiencia de usuario .....	104
9.2.3 Pruebas de rendimiento .....	106
9.2.3 Pruebas de compatibilidad.....	111
<b>CAPITULO 4</b> .....	113
<b>10. CONCLUSIONES</b> .....	113
<b>11. RECOMENDACIONES</b> .....	114
<b>12. PROYECCIONES</b> .....	115
<b>REFERENCIAS</b> .....	116

## LISTA DE FIGURAS

<b>Lista de figuras</b>	<b>Pág.</b>
Figura 1. Captura de pantalla del funcionamiento del videojuego MOW .....	25
Figura 2. Capturas de pantalla del funcionamiento de TeachAR .....	26
Figura 3. Captura de pantalla de la aplicación en funcionamiento. ....	27
Figura 4. Interfaz gráfica de la aplicación Phonic Tricksters .....	27
Figura 5. Logo de la compañía Alive Studios .....	28
Figura 6. software diseñado por los estudiantes de la Universidad de Nariño .....	29
Figura 7. Aplicación desarrollada por estudiantes de la Universidad Popular del Cesar .....	30
Figura 8. aplicación desarrollada por estudiante de la Universidad de Medellín .....	31
Figura 9. Ejemplo de un sistema de huesos en la herramienta Blender .....	32
Figura 10. Captura de pantalla del panel transform y edit en Blender. Elaboración propia. ....	39
Figura 11. Panel Material de Blender. Elaboración propia .....	40
Figura 12. Panel Texture de Blender. Elaboración propia .....	40
Figura 13. Código de barras y Código QR. ....	45
Figura 14. Ejemplo de marcadores nivel 1 de Realidad aumentada. ....	45
Figura 15. Ejemplo de nivel 2 de Realidad aumentada. ....	46
Figura 16. Esquema de funcionamiento de la realidad aumentada. ....	47
Figura 17. Comparación de un código QR y un código de barras. ....	48
Figura 18. Partes de un código QR. ....	48
Figura 19. Arquitectura de Vuforia. ....	49
Figura 20. Uso de Vuforia junto a Unity en un marcador 2D. ....	50
Figura 21. Tipos de marcadores (Triggers) reconocidos por Vuforia. ....	50
Figura 22. Etapas de desarrollo Mobile-D. ....	52
Figura 23. Requisitos funcionales. Elaboración propia .....	53
Figura 24. Requisitos funcionales: acceso a la aplicación por primera vez. Elaboración propia ..	54
Figura 25. Requisitos Funcionales: Menú principal. Elaboración propia .....	55
Figura 26. Requisitos Funcionales: Acceso a lecciones. Elaboración propia. ....	56
Figura 27. Requisitos Funcionales: Acceso a exámenes de lección. Elaboración propia. ....	57
Figura 28. Requisitos no funcionales. Elaboración propia. ....	57
Figura 29. Reglas del negocio. Elaboración propia .....	58
Figura 30. Restricciones. Elaboración propia .....	58
Figura 31. Diagrama de casos de uso. Elaboración propia .....	59
Figura 32 Cronograma de actividades. Elaboración propia. ....	64
Figura 33. Wireframe de la aplicación. Elaboración propia. ....	66
Figura 34. Captura de pantalla de prototipo de interfaz de pantalla inicial. Elaboración propia. .	67
Figura 35. Proceso secuencial de la creación de un perfil de usuario. Elaboración propia. ....	68
Figura 36. Desarrollo de interfaz del menú principal. Elaboración propia. ....	69
Figura 37 adición de los botones ayuda, perfil y salir en el menú principal. Elaboración propia. 69	
Figura 38 Captura de pantalla de interfaz de menú perfil de usuario. Elaboración propia. ....	70
Figura 39 Desarrollo de interfaz de menú ayuda. Elaboración propia. ....	70
Figura 40 Interfaz de escenario de lecciones y exámenes. Elaboración propia. ....	71

Figura 41 Menú de pausa con las interfaces de submenús de opciones y ayuda. Elaboración propia.	72
Figura 42 Secuencia de menú informativo e instructivo de lecciones y exámenes. Elaboración propia.	73
Figura 43 Diversos mensajes de advertencia en la aplicación. Elaboración propia.	73
Figura 44 Pagina web FLATICON	74
Figura 45 Diseño final de imagen de fondo en GIMP. Elaboración propia.	74
Figura 47 Recopilación del resultado final de las principales interfaces de usuario. Elaboración propia.	75
Figura 48 Banco de imágenes Pixabay.com.	76
Figura 50 Cuadro comparativo de los diferentes software y servicios web de síntesis de habla disponibles en el mercado. Elaboración propia.	77
Figura 52 Captura de pantalla de herramienta de síntesis de habla Polly.	78
Figura 53 Muestra del proceso de modelado en Blender	80
Figura 54 Elaboración de las letras y los números en 3D en Blender. Elaboración propia.	80
Figura 55 Textura base del calendario 3D. Elaboración propia.	81
Figura 56 Proceso de texturizado usando la herramienta de mapeo y despliegue UV en Blender. Elaboración propia.	81
Figura 57 Reloj digital hecho en Blender. Elaboración propia.	82
Figura 58. Captura de pantalla del software Make Human. Elaboración propia.	82
Figura 60 Proceso de modelado, creación de texturas y texturizado de comidas en 3D. Elaboración propia.	83
Figura 61 Prendas de vestir disponibles en el software Make Human. Elaboración propia.	84
Figura 62 Modelado de prendas de vestir adicionales hechas en Blender. Elaboración propia.	84
Figura 65 Creación de palabras del vocabulario ingles en 3D hechas en Blender. Elaboración propia.	85
Figura 66 Modelado de interior y exterior de una casa en Blender. Elaboración propia.	86
Figura 67 Pagina web QRCODEMONKEY.	86
Figura 68 Códigos QR: Principal, Imagen y Audio. Elaboración propia.	87
Figura 69 Prototipo de escenario de lecciones. Elaboración propia.	88
Figura 70 Escenario de lección: El Calendario. Elaboración propia.	89
Figura 71 Proceso de elaboración del escenario de Los Números. Elaboración propia.	89
Figura 72 Elaboración del escenario de Las Horas. Elaboración propia.	90
Figura 73 Construcción de escenario y mecánicas de las prendas de vestir y partes del cuerpo. Elaboración propia.	91
Figura 74 Creación del escenario de las partes de la casa. Elaboración propia.	91
Figura 75 Prototipo de exámenes de lección. Elaboración propia.	92
Figura 76 captura de pantalla de fragmento de código para reproducir audio. Elaboración propia.	92
Figura 77 captura de pantalla de fragmento de código para mostrar u ocultar elementos. Elaboración propia.	93
Figura 78 Captura de pantalla de fragmento de código para reproducir animaciones. Elaboración propia.	93

Figura 79 Captura de pantalla de fragmento de código para cambio y visualización de imágenes. Elaboración propia. ....	94
Figura 80 adición de modelos 3D a diversos prototipo de escenarios. Elaboración propia. ....	94
Figura 81 adición de imágenes 3D a diversos prototipo de escenarios. Elaboración propia. ....	95
Figura 82 Ejemplos de mecánicas programadas en funcionamiento en cada escenario. Elaboración propia. ....	95
Figura 83 Conexión entre: menú principal, escenarios de lecciones y escenarios de exámenes. Elaboración propia. ....	96
Figura 84 Proceso de compilación en Unity. Elaboración propia. ....	97
Figura 85 Información de identificación de la app. Elaboración propia. ....	97
Figura 86 Configuración de orientación de la aplicación. Elaboración propia. ....	98
Figura 87 Icono de la aplicación con diferente resolución de pixeles. Elaboración propia. ....	99
Figura 88 Configuraciones de publicación. Elaboración propia. ....	99
Figura 89. Reconocimiento de Códigos QR. Elaboración propia. ....	100
Figura 91. Resultado de ajustes a la distancia de recortes de plano. Elaboración propia. ....	100
Figura 92. Fragmento de código de detección de toques en pantalla. Elaboración propia. ....	101
Figura 93. Prueba de funcionamiento de script que detecta toques en pantalla. Elaboración propia. ....	102
Figura 94. Secuencia de acciones al seleccionar una lección. Elaboración propia. ....	103
Figura 95. apertura de pantalla de ayuda al pulsar el botón de ayuda. Elaboración propia. ....	103
Figura 97. Flujo de navegación entre lecciones de menú principal. Elaboración propia. ....	103
Figura 98. Interfaz de bienvenida al acceder por primera vez a la aplicación. Elaboración propia. ....	104
Figura 99. Interfaz de usuario de escenario de lecciones y exámenes de lección. Elaboración propia. ....	105
Figura 100. Secuencia de ejecución de examen de lección. Elaboración propia. ....	105
Figura 101. Consumo de memoria RAM telefono móvil Huawei. Elaboración propia. ....	107
Figura 102. Consumo de memoria RAM teléfono móvil LEECO LePro3. Elaboración propia. ....	108
Figura 103. Consumo de memoria RAM teléfono móvil LEECO LeMax2. Elaboración propia. ....	108
Figura 104. Consumo y temperatura de CPU en teléfono móvil Huawei Honor 4X. Elaboración propia. ....	109
Figura 105. Consumo y temperatura de CPU en teléfono móvil LEECO LePro3. Elaboración propia. ....	109
Figura 106. Consumo y temperatura de CPU en teléfono móvil LEECO LeMax2. Elaboración Propia. ....	110
Figura 107. Prueba de compatibilidad App en dispositivo Huawei. Elaboración propia. ....	111
Figura 108. Prueba de incompatibilidad App en dispositivo Samsung S5. Elaboración propia. ....	112

**LISTA DE TABLAS**

Tabla 1. Tiempos de instalación de la aplicación. Elaboración propia. ....	106
Tabla 2. Tiempos de desinstalación de la aplicación. Elaboración propia. ....	106
Tabla 3. Tiempos de carga de escenas en los 3 dispositivos de prueba. Elaboración propia. ....	107
Tabla 4. Promedio de consumo de memoria RAM antes y después de ejecución Huawei Honor 4X. Elaboración propia. ....	108
Tabla 5. Promedio de consumo de memoria RAM antes y después de ejecución LEECO LePro3. Elaboración propia. ....	108
Tabla 6. Promedio de consumo de memoria RAM antes y después de ejecución LEECO LeMax2. Elaboración propia. ....	109
Tabla 7. Promedio de consumo de batería. Elaboración Propia. ....	110

**LISTA DE CUADROS**

Cuadro 1. Caso de uso: Crear usuario. Elaboración propia.....	60
Cuadro 2. Caso de uso: Desarrollar Lección. Elaboración propia.....	62
Cuadro 3. Caso de uso: Ver perfil de usuario. Elaboración propia.....	62
Cuadro 4. Caso de uso: Ver instrucciones de uso. Elaboración propia.....	63

## CAPÍTULO 1

### INTRODUCCIÓN

En Colombia existe un déficit en las instituciones educativas en cuanto a la enseñanza y aprendizaje del idioma inglés en la educación básica y media que va desde el nivel de 6to de bachillerato en adelante, y esto, es a raíz de diversos factores que afectan el desempeño de los jóvenes entre 10 y 12 años en las aulas de clase. Uno de los principales factores es la ausencia del uso de herramientas tecnológicas por parte de los docentes, ya que estos mismos, se limitan al uso de métodos anticuados y desactualizados para impartir la enseñanza de una segunda lengua. Por ello, en el siguiente proyecto se desarrollará una aplicación móvil, que facilite a los jóvenes dentro y fuera de las aulas, el aprendizaje del vocabulario del inglés nivel A1 mediante el uso de la realidad aumentada.

Debido a que la aplicación está diseñada para dispositivos móviles, en el presente trabajo se enfatiza la metodología Mobile-D, una de las más adecuadas para el desarrollo de aplicaciones móviles por ser una metodología de desarrollo ágil. Esta metodología está compuesta por 5 fases, estas fases son: fase de exploración, fase de inicialización, fase de producción, fase de estabilización y fase de pruebas, dichas fases servirán como guía para la elaboración del aplicativo móvil. Durante la realización de las fases, se llevará a cabo la investigación de los componentes esenciales para el desarrollo, elementos que permitan compactar la idea, dar inicio al desarrollo del aplicativo móvil, así como su respectiva documentación y finalizar con la realización de pruebas que permitan dar garantía de su completo funcionamiento y desempeño.

La construcción de la aplicación, se hizo con la finalidad de brindar un apoyo al proceso de aprendizaje del inglés de una forma distinta, haciendo uso de la realidad aumentada como un instrumento para ampliar los conocimientos sobre el vocabulario nivel A1 de dicho idioma, motivando a los niños a profundizar en este tema, interactuando con todo el material de imágenes audio y modelos en 3D que brinda la aplicación. Por último, se consideró que el aplicativo debía ser de fácil acceso, es por esto que se agregó a la tienda de aplicaciones Google Play con la finalidad de que sea más asequible a la población de menores y que de una u otra manera a través de las calificaciones o comentarios obtenidos en la tienda, se pueda apreciar su usabilidad, experiencia y satisfacción.

## 1. PROBLEMA

### 1.1 Planteamiento del problema

Teniendo como base los resultados de las pruebas Saber 11° impartidas por el Instituto Colombiano Para Evaluar la Educación (ICFES), los porcentajes obtenidos a nivel nacional desde años atrás en cuanto al dominio del inglés de los estudiantes de grado 11, denotan un nivel bajo del manejo de esta lengua; porcentajes tales como: 49,31% en el 2012, 54,22% en el 2013 y 39,59% en el año 2014, de estudiantes que no alcanzan el nivel mínimo de inglés A1 según la tabla del Marco Común Europeo de Referencia (MCER) (ICFES Interactivo, s.f.).

Todo lo anteriormente descrito, es consecuencia de malas prácticas y poco nivel académico en la educación básica y media que se imparte en las instituciones educativas desde el nivel de 6to de bachillerato en donde la población de jóvenes entre 10 y 12 años en Colombia, recibe una educación de idiomas extranjeros muy limitada, esta problemática nace principalmente en las aulas de los colegios, y esto se debe a diversos factores que dificultan el aprendizaje en los niños, uno de ellos, es la carencia de docentes de educación básica primaria que posean las habilidades en inglés necesarias para poder impartir una enseñanza de calidad (Moncada, 2015). Puesto que, las políticas educativas, erróneamente han asignado responsabilidad a los docentes de enseñar inglés a niños de primaria, sin que necesariamente estén certificados como profesores de inglés (Olarte, 2016). Otro aspecto que alimenta este problema, es el uso de métodos anticuados que utilizan los docentes para la enseñanza, ya que los profesores se limitan a utilizar herramientas desactualizadas y no optan por implementar nuevas tecnologías en sus clases; “Esto puede estar asociado al hecho empírico de qué a medida que los docentes presenten más edad van a usar menos las TIC, originado probablemente porque no están familiarizados con su uso y prefieren continuar con los métodos convencionales de enseñanza” (Lemoine, y otros, 2015).

De dicha problemática la población afectada son los niños, y generalmente los que tienen edades que van desde los 10 a los 12 años, rango de edades en la cual un niño promedio está finalizando la educación básica primaria y empezando la educación media tal como lo especifica el Sistema de Educación Colombiano dado por el Ministerio de Educación Nacional (MEN) (Ministerio de Educación Nacional, 2014), esta misma población de menores concuerda con la

propuesta curricular del MEN de impartir clases de inglés nivel A1 a partir de 6to grado en adelante (Equipo Colombia Bilingüe, 2016).

## **1.2 Formulación del problema**

¿Cómo desarrollar una herramienta tipo aplicación móvil para sistema operativo Android que utilice la realidad aumentada como instrumento de enseñanza y aprendizaje del idioma inglés nivel A1 en niños de 10 a 12 años de edad?

## **2. OBJETIVOS**

### **2.1 Objetivo general**

Desarrollar una aplicación para dispositivos móviles con sistema operativo Android, que refuerce del aprendizaje del idioma inglés de nivel A1 en niños de 10 a 12 años de edad mediante el uso en realidad aumentada.

### **2.2 Objetivos específicos**

- Establecer los requisitos funcionales y no funcionales necesarios para el desarrollo del proyecto, teniendo en cuenta las indicaciones pedagógicas dadas por los asesores.
- Diseñar la aplicación a partir de la funcionalidad.
- Desarrollar la aplicación móvil con las herramientas previamente seleccionadas y conforme a las necesidades establecidas.
- Elaborar la documentación del proyecto, tales como los manuales técnicos y de usuario, de acuerdo a las necesidades emergentes del proyecto.
- Realizar las pruebas de funcionalidad del software, así como de cumplimiento sobre los requisitos establecidos.

### 3. ALCANCES Y LIMITACIONES

#### 3.1 Alcances

Se desarrolló el aplicativo para ser implementado en dispositivos móviles con sistema operativo Android 5.0 o superior debido a que, a partir de esta versión, los dispositivos cuentan con un mejor hardware. Estos dispositivos cuentan con capacidad de almacenamiento de 8GB como mínimo, memoria RAM a partir de los 2GB en adelante, cámaras de al menos 8 Megapíxeles y procesadores de 4 o más núcleos, haciendo posible el uso de la realidad aumentada, efecto que no pasa en las versiones anteriores.

El método que utiliza la aplicación para proyectar todo el contenido en pantalla es el reconocimiento basado en marcadores, con esto se hace referencia al nivel 1 de la realidad aumentada, dichos marcadores son imágenes impresas en blanco y negro que cuando son escaneados por la cámara del dispositivo, muestra sobre ellos, imágenes u objetos en tercera dimensión. En este punto hay que hacer una breve aclaración, ya que dichos marcadores que se crearon y usaron tienen la misma forma y diseño que un código QR (nivel 0 de realidad aumentada), pero funcionan de manera distinta, esto con la finalidad de que fueran más fáciles de reconocer por los dispositivos gracias a la estructura que poseen dichos códigos.

Teniendo en cuenta que el nivel A1 del idioma inglés es muy extenso, y generalmente está dividido en lecciones, se desarrolló una aplicación que solo contiene el vocabulario de las lecciones más fundamentales de dicho nivel. Se elaboró un total de 12 lecciones las cuales son:

- **Los colores:** contiene la escritura y pronunciación de los colores principales en inglés.
- **Los números:** contiene la escritura y pronunciación de los números desde el 0 hasta el 9999 en inglés.
- **Las vocales:** contiene la escritura y pronunciación de las vocales en inglés.
- **El abecedario:** contiene la escritura y pronunciación de las letras del abecedario en inglés.
- **El calendario:** contiene la escritura y pronunciación de los días, meses y las estaciones del año en inglés.
- **Las horas:** contiene la escritura y pronunciación de las horas y los tiempos del día (mañana, tarde y noche) en inglés.

- **Las partes del cuerpo:** contiene la escritura y pronunciación de las partes del cuerpo en inglés.
- **Las comidas:** contiene la escritura y pronunciación de diferentes comidas en inglés tales como: las frutas, verduras, bebidas y comidas rápidas.
- **Las prendas de vestir:** contiene la escritura y pronunciación de las principales prendas de vestir tanto masculinas como femeninas.
- **Los animales:** contiene la escritura y pronunciación de diferentes animales aéreos, acuáticos y terrestres.
- **Los medios de transporte:** contiene la escritura y pronunciación de diferentes medios de transporte acuáticos, terrestres y aéreos.
- **Las partes de la casa:** contiene la escritura y pronunciación de las partes de la casa, partes tanto exteriores como interiores de la casa, así como también, todos los muebles y diferentes objetos que se pueden encontrar dentro de ella.

También se realizó la oferta en el mercado de aplicaciones de Google Play para la accesibilidad de cualquier usuario que posea un dispositivo móvil que cumpla con las características mencionadas anteriormente.

### **3.2 Limitaciones**

#### **3.2.1 Técnicas**

- Que el usuario no tenga acceso a internet para descargar los marcadores QR.
- Que el usuario no disponga de los marcadores impresos o estén defectuosos.

#### **3.2.2 Tecnológicas**

- Que los usuarios no cuenten con un dispositivo móvil Android.
- Que la versión del sistema operativo Android no sea la mínima requerida o que el propio dispositivo no sea compatible con la tecnología de realidad aumentada.

#### 4. JUSTIFICACION

El dominio de un segundo idioma se ha convertido en una parte fundamental para el desarrollo personal de los estudiantes, puesto que, debido a la globalización del mundo, tener la capacidad de ser bilingüe o multilingüe a día de hoy es cada vez más necesario para el estudio, el trabajo y la convivencia (Mejia, 2005).

A causa de que el inglés es ampliamente hablado, se le ha denominado “el idioma del mundo”, ya que, a pesar de no ser un idioma oficial en la mayoría de los países, es actualmente la lengua que se enseña con más frecuencia como lengua extranjera, entre estos países que lo enseñan se encuentra Colombia, y esto es gracias al Plan Nacional de Bilingüismo (PNB) implementado por el Ministerio de Educación Nacional (MEN) que tiene como objetivo “lograr ciudadanos y ciudadanas capaces de comunicarse en inglés, de tal forma que puedan insertar al país en los procesos de comunicación universal, en la economía global y en la apertura cultural, con estándares internacionalmente comparables” (Ministerio de Educación Nacional, 2006). Es por ello por lo que está integrado en el pensum de los colegios y las universidades la asignatura de Inglés, para que los estudiantes adquieran estas competencias para su vida personal, profesional, laboral y/o académica.

Los métodos convencionales de enseñanzas no son adecuados para todos los alumnos, dado que cada cual tiene formas de aprendizaje distintas. A raíz de esto, diferentes empresas comenzaron a ofrecer servicios tales como Babbel o Rosetta Stone, servicios de aprendizaje innovadores que emplean diversos recursos donde el alumno interactúa de forma amena y entretenida. Estos servicios en su gran mayoría son servicios de pago, ofrecen un contenido básico, que sirve para incentivar a los usuarios a que adquieran el servicio completo mediante el pago de una suscripción. Hay que mencionar que las aplicaciones mencionadas anteriormente se limitan únicamente al uso de elementos multimedia tales como imágenes, audio y video para enseñar el idioma inglés a los usuarios y no están dirigidos a un público específico, sumado a esto, muchos de estos cursos no tienen un manejo por niveles, donde jóvenes de bachillerato puedan empezar una formación metodológicamente cercana a sus necesidades. Dicho esto, se decide desarrollar una aplicación para dispositivos móviles Android totalmente gratuita, libre de publicidad, de fácil acceso para los niños y estudiantes a través de la tienda de aplicaciones de Google Play Store, que utilice los elementos multimedia

mencionados anteriormente como base, pero con el valor agregado que implica la realidad aumentada como mecanismo didáctico de enseñanza-aprendizaje, sirviendo como herramienta de apoyo en la enseñanza del idioma inglés dentro y fuera de las instituciones educativas, aprovechando las bondades que posee esta tecnología, debido a que, la realidad aumentada permite combinar el mundo real con el virtual, enriqueciendo la experiencia visual y mejorando la calidad de comunicación, añadiendo información visual así como imágenes, audios y modelos 3D a la realidad para poder crear nuevas experiencias interactivas en los jóvenes de 10 a 12 años de edad.

Se toma la plataforma Android como punto de inicio para este proyecto, ya que en la actualidad es el sistema operativo más usado en dispositivos móviles a nivel mundial, apropiándose del 82% del mercado en el año 2017 (Goasduff & Forni Ann, 2017) y del 85.9% en el 2018 (Costello & Hippold Cornelia, 2018); lo cual implica que la mayoría de personas cuenta con este sistema operativo en sus dispositivos móviles, esto con el fin de llegar a más estudiantes de bachillerato en el rango de los 10 a 12 años de edad, que deseen reforzar su inglés para cumplir con el objetivo de alcanzar el nivel A1, tanto dentro como fuera de las aulas y que a su vez no se vean limitados por la carencia de horas de enseñanza, la escasez de profesores en diferentes regiones del país y la falta conocimiento del inglés de los mismos.

## **5. LINEA DE INVESTIGACION**

SOFTWARE, SISTEMAS EMERGENTES Y NUEVAS TECNOLOGÍAS, Conjunto de programas, subprogramas subrutinas y menús que se elaboran a manera de aplicaciones y/o paquetes para cumplir con un fin específico. (Acuerdo No 007 de mayo 29 de 2003) (Universidad de Cundinamarca, 2003).

## CAPITULO 2

### 6. MARCO TEORICO

#### 6.1 Marco Referencial

Dado que a la aplicación que se desarrolló se le dio el enfoque del refuerzo del aprendizaje de idioma ingles nivel A1, haciendo uso de la realidad aumentada, hay que resaltar que, en el mercado móvil existe una gran variedad de aplicaciones que hacen uso de esta tecnología, pero si se realiza un análisis en profundidad, teniendo en cuenta únicamente aplicaciones de enseñanza y aprendizaje de idiomas, se puede verificar que la temática está poco explorada, lo que deja un nicho de mercado donde se puede hacer un aporte considerable.

A continuación, se muestran diferentes ejemplos de aplicaciones con características similares:

##### 6.1.1 MOW (Matching Objects and Words)

MOW es un videojuego educativo para niños portugueses de edades entre los 5 y 12 años que estén cursando la primaria estudiantil. Utiliza la realidad aumentada como una herramienta para que los niños puedan aprender una variedad de palabras tanto en idioma inglés como en portugués.

A través de un primer contacto visual y la comprensión oral, seguida de un reconocimiento continuo y el dominio verbal de las palabras escritas, muy apoyado por la memoria como se ejerce a través de tareas específicas dentro del juego, proporcionando a los niños pequeños una formación y exploración completa de sus habilidades internas y sus capacidades de aprendizaje (Barreira, et al., 2012).

MOW fue desarrollado para computadores con sistema operativo Windows, utilizando las herramientas ARToolKit y la librería OpenCV para el seguimiento y reconocimiento de los marcadores de Realidad Aumentada (nivel 1 de la realidad aumentada).

El videojuego cuenta con 2 opciones: la primera opción es “Juego de Memoria” a través del cual los niños pueden aprender los nombres de animales en cualquiera de los 2 idiomas mencionados con anterioridad, y la segunda es “Juego de parejas” a través del cual los niños pueden practicar sus habilidades uniendo las palabras a los respectivos objetos 3D que se proyectan en pantalla.



Figura 1. Captura de pantalla del funcionamiento del videojuego MOW<sup>1</sup>

### 6.1.2 TeachAR

TeachAR es una herramienta de realidad aumentada, la cual tiene como función enseñar palabras básicas en inglés (colores, formas geométricas y preposiciones) a niños menores de 7 años los cuales el inglés no es su lengua nativa. dicha herramienta fue desarrollada para computadores de escritorio y creada con el motor de juegos Unity junto con la librería ARToolkit, esta librería es la que le permite al sistema reconocer los 20 marcadores que contienen cada uno de los elementos de enseñanza de las palabras en inglés. Además de esto, el sistema usa un sensor Kinect de Microsoft para realizar el reconocimiento de voz de los niños.

Utilizamos el reconocimiento de voz en forma de comandos de voz, para que los estudiantes tengan la experiencia de pronunciar palabras y no sólo escucharlas. Esto se debe a que nosotros creemos que la pronunciación se aprende mejor cuando los niños tienen la experiencia de decir la palabra y no sólo de escucharla. (Che Dalim, Dey, Piumsomboon, Billingham, & Sunar, 2016)

<sup>1</sup> Barreira, J., Bessa, M., Pereira, L., Adão, T., Peres, E., & Magalhães, L. (2012). MOW: Augmented Reality Game to Learn Words in. [Figura]. Recuperado de: <https://ieeexplore.ieee.org/abstract/document/6263236>



Figura 2. Capturas de pantalla del funcionamiento de TeachAR<sup>2</sup>

### 6.1.3 Usando la Realidad Aumentada para Enseñar Vocabulario en inglés a Estudiantes de Kindergarten.

Un equipo de 4 estudiantes de la School of Educational Technology (Escuela de Tecnología Educativa) en China desarrollaron una aplicación móvil para enseñar el vocabulario del idioma inglés a niños de 2 a 5 años que cursan el nivel educativo kindergarten, los temas de enseñanza que emplea la aplicación son: el sistema solar, los medios de transporte y los animales.

El modo de funcionamiento de la aplicación es a través de la cámara del dispositivo la cual captura el entorno real en el cual se quiere sobreponer el contenido 2D y 3D, ya que la aplicación no utiliza ningún tipo de marcador o código de barras si no que se basa en el nivel 2 de la realidad aumentada, entonces, por medio del uso del GPS, giroscopio y el acelerómetro la aplicación es capaz de mezclar el entorno real con el contenido virtual para que así pueda ser visualizado a través de la pantalla del dispositivo móvil. Adicional a esto, la aplicación cuenta con un sistema de monitorio en tiempo real, la cual permite que los padres puedan observar el tiempo de uso de la aplicación por parte de sus hijos además de restringir el uso de la aplicación si se ha excedido un periodo de tiempo establecido por ellos mismos.

El prototipo de la aplicación se puede ejecutar en un dispositivo móvil usando Android 4.4 como sistema operativo y equipado con una cámara, un micrófono y al menos 1 GB RAM de memoria. El sistema de monitorización requiere acceso a la red para que funcione. El prototipo se desarrolló utilizando el motor de juego Unity y el plugin Kudan AR, y utiliza markerless AR. Por lo tanto, no se necesitan materiales de aprendizaje físicos. (Lee, Chau, Chau, & Ng, 2017)

<sup>2</sup> Che Dalim, C., Dey, A., Piumsomboon, T., Billinghamurts, M., & Sunar, S. (2016). TeachAR: An Interactive Augmented Reality Tool for Teaching Basic English to Non-Native Children. [Figura]. Recuperado de: <https://ieeexplore.ieee.org/abstract/document/7836467>



Figura 3. Captura de pantalla de la aplicación en funcionamiento.<sup>3</sup>

#### 6.1.4 Phonic Tricksters

Es una aplicación de la empresa Pearson, diseñada para la enseñanza de pronunciación en inglés, reconociendo los sonidos de las letras. La integración de la realidad aumentada permite interactuar con el entorno y que el aprendizaje sea más divertido, pues se puede perseguir a los tricksters alrededor del salón de casa o dentro del aula de la escuela. (Caro, 2014)



Figura 4. Interfaz gráfica de la aplicación Phonic Tricksters<sup>4</sup>

<sup>3</sup> Lee, L.-K., Chau, C.-h., Chau, C.-H., & Ng, C.-T. (2017). Using Augmented Reality to Teach Kindergarten Students English Vocabulary. [Figura]. Recuperado de: <https://ieeexplore.ieee.org/abstract/document/8005387>

<sup>4</sup> (s.a). (2014). Ejemplo de aplicación Phonic Tricksters en funcionamiento. [Figura]. Recuperado de: [https://digitalteachingandlearning.files.wordpress.com/2014/06/img\\_0261.png](https://digitalteachingandlearning.files.wordpress.com/2014/06/img_0261.png)

### 6.1.5 Letter Alive

Es un software de pago, para la enseñanza de inglés utilizando la realidad aumentada, desarrollado por la compañía Alive Studios y diseñado para enseñar a los niños desde Pre Kínder hasta 3er grado de primaria los conceptos fundamentales del idioma inglés, la cual solo funciona en computadores. (Karlin, 2017)



*Figura 5. Logo de la compañía Alive Studios<sup>5</sup>*

### 6.1.6 Software para la enseñanza de fuentes de energía renovables y no renovables a niños de 5to de primaria de la escuela Normal Superior de Pasto.

Un grupo formado por dos estudiantes de la Universidad de Nariño (Colombia), crearon un software diseñado para niños de quinto de primaria de la escuela Normal Superior de Pasto, el cual tiene como objetivo el enseñar la temática de la energía, así como también los tipos de fuentes de energía renovables y no renovables por medio del uso de la realidad aumentada. Dicho programa está diseñado únicamente para computadores, los cuales a través de una cámara web reconoce los marcadores de nivel 1 de realidad aumentada para mostrar el todo el contenido que tiene la aplicación. Estos marcadores se encuentran impresos en una cartilla creada por ellos mismos, la cual contiene diferentes tipos de imágenes, textos y una breve introducción a la temática que se imparte y a su vez toda la serie de marcadores anteriormente nombrados. Además, este mismo software cuenta con un sistema de cuestionarios en los cuales “el estudiante podía responder por medio de marcadores de realidad aumentada que contenían cierta información. El estudiante

<sup>5</sup> Karlin, M. (2017). Alive Studios: Reading and Math proficiency using Augmented Reality. [Figura]. Recuperado de: <http://www.edtechroundup.org/reviews/alive-studios-reading-and-math-proficiency-using-augmented-reality>

lograba organizar estos elementos en orden correcto, o enfocar en cámara aquellos que considere necesarios según la pregunta que se le sugería. En esta misma actividad se indicaba al estudiante cuales fueron sus aciertos en la actividad desarrollada” (Jaguandoy Tobar & Puchana Legarda, 2014). Cabe añadir que este software se desarrolló teniendo como referencia el plan de aula correspondiente a tecnología e informática propio de la escuela mencionada con anterioridad.



Figura 6. software diseñado por los estudiantes de la Universidad de Nariño <sup>6</sup>

### 6.1.7 ReinAR App

ReinAR App es una aplicación para dispositivos móviles desarrollada por estudiantes de la Universidad Popular del Cesar (Colombia), utilizando el plug-in de Vuforia junto con el motor de videojuegos Unity. Dicha aplicación fue diseñada para reforzar el aprendizaje del área de biología en estudiantes de primaria, usando la realidad aumentada como herramienta didáctica de apoyo para enseñar los diferentes reinos de la naturaleza tales como: animal, vegetal, hongo, monera y protista. Esta aplicación utiliza diversos recursos multimedia como lo son: imágenes, audios y modelos en 3D que representan los temas que expone la aplicación. Al igual que el software expuesto con anterioridad, esta aplicación también utiliza una dinámica de pruebas, las cuales sirven para observar los avances del estudiante en la aplicación, además de contar con un sistema de puntos el cual actúa como un incentivo para que el estudiante trate de responder correctamente todas las preguntas y pueda redimir los puntos obtenidos de cada examen en una tienda propia de

<sup>6</sup> Jaguandoy Tobar, H. F., & Puchana Legarda, C. C. (2014). Software de realidad aumentada diseñado por estudiantes de la Universidad de Nariño (Colombia). [Figura]. Recuperado de: <http://sired.udenar.edu.co/333/1/90234.pdf>

la aplicación. El funcionamiento de la aplicación es el siguiente: la cámara del dispositivo captura los marcadores que hay en el entorno, que para este caso son imágenes impresas en una cartilla, y proyecta el contenido en 3D a través de estos. Mediante una interfaz de usuario el alumno puede interactuar con el contenido proyectado rotándolo, moviéndolo o cambiar el elemento en 3D que se esté proyectando en pantalla. (Restrepo Duran, Cuello, & Contreras Chinchilla, 2017)



Figura 7. Aplicación desarrollada por estudiantes de la Universidad Popular del Cesar<sup>7</sup>

### 6.1.8 AR Ciencias Básicas

AR Ciencias básicas es un proyecto realizado en la Universidad de Medellín, creado para enseñar conceptos básicos sobre las ciencias naturales a estudiantes de quinto grado de primaria de la institución educativa Campo Valdés, los contenidos con los que cuenta la aplicación son: la tierra y sus capas, la atmosfera y sus capas. A diferencia de las 2 aplicaciones mencionadas anteriormente, esta no cuenta con ninguna mecánica para exámenes o evaluaciones de progreso de los alumnos. El modo de funcionamiento es a partir del uso de la realidad aumentada en dispositivos móviles Android, los cuales a través de la cámara son capaces de reconocer los marcadores que para este caso son imágenes (nivel 1 de la realidad aumentada) impresas en una

<sup>7</sup> Restrepo Duran, D., Cuello, S. L., & Contreras Chinchilla, L. (2017). Juegos didácticos basados en realidad aumentada como apoyo en la enseñanza de biología. [Figura]. Recuperado de: <http://documentos.redclara.net/bitstream/10786/1263/1/39-4-4Juegos%20did%C3%A1cticos%20basados%20en%20realidad%20aumentada%20como%20apoyo%20en%20la%20ense%C3%B1anza%20de%20biolog%C3%ADa.pdf>

hoja de papel, y mediante la pantalla del dispositivo se proyecta todo el contenido mencionado anteriormente. Este proyecto fue desarrollado en el Motor de videojuegos Unity junto con diversas librerías de realidad aumentada (Baron, 2014).

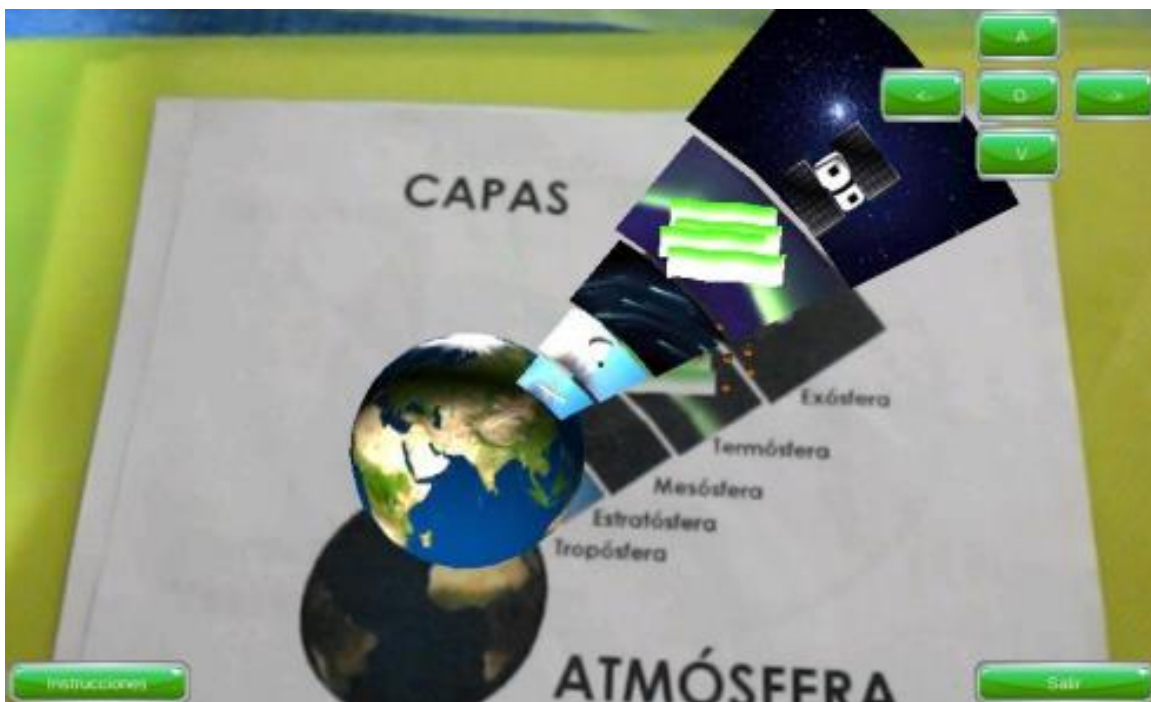


Figura 8. aplicación desarrollada por estudiante de la Universidad de Medellín <sup>8</sup>

## 6.2 Marco conceptual

### 6.2.1 Animación 3D

Es el proceso de crear movimientos o deformaciones de un modelo 3D, en un tiempo determinado, aplicando leyes físicas. Es complejo, debido a que depende de otros procesos tales como, el diseño, modelado y rigging del objeto al cual se desea animar.

### 6.2.2 Rigging

Es el proceso mediante el cual se crea un sistema de huesos (esqueleto virtual) y se asigna a los modelos en 3D, con la finalidad de generar movimientos al objeto 3D o en su defecto deformarlo. “Rigging o Setup es un proceso que lleva mucha lógica, pero también sentido artístico para poder

<sup>8</sup> Baron, O. M. (2014). aplicación desarrollada por estudiante de la Universidad de Medellín. [Figura]. Recuperado de: <https://repository.udem.edu.co/bitstream/handle/11407/1242/Realidad%20aumentada%20como%20estrategia%20did%C3%A1ctica%20en%20curso%20de%20ciencias%20naturales%20de%20estudiantes%20de%20quinto%20grado%20de%20primaria%20de%20la%20Instituci%C3%B3n%20Educativa>

hacer del modelo algo que pueda expresar sentimientos, movimientos y expresiones dependiendo del caso” (Arzuza, 2011). Se anexa la siguiente figura en la cual se observa un esqueleto de un modelo humanoide.

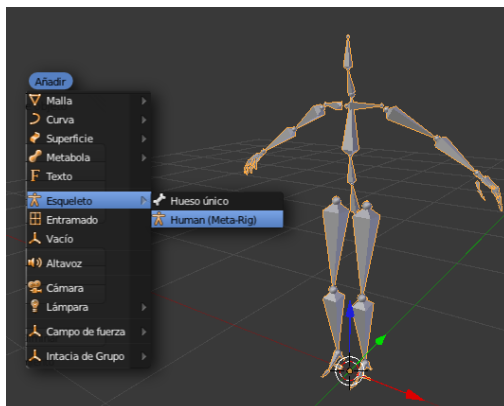


Figura 9. Ejemplo de un sistema de huesos en la herramienta Blender<sup>9</sup>

### 6.2.3 Aplicación móvil

Es un software diseñado específicamente para dispositivos móviles. Una función esencial de este tipo de aplicaciones, es que deba tener una parte ejecutándose en un servidor web o de poder acceder a esta mediante dicho servicio de forma gratuita o de pago.

Las aplicaciones móviles se definen en tres tipos; web, nativas o híbridas. Las de tipo web, son por lo general páginas web, que se enfocan a los dispositivos móviles, las cuales pueden acceder a los recursos del equipo de una manera limitada. Comúnmente, son desarrolladas en HTML, CSS o JavaScript. Este tipo de aplicaciones móviles son ejecutadas en el navegador instalado en el dispositivo.

Por otra parte, las aplicaciones de tipo nativas, son desarrolladas para un dispositivo en concreto, por lo cual pueden aprovechar un gran porcentaje de los recursos de hardware del móvil. Son desarrolladas en Java para el caso de Android, y Objective-C o Swift para IOS.

Por último, las aplicaciones de tipo híbridas, son parte nativas y parte webApp, debido a que son páginas web inmersas en aplicaciones nativas. De esta manera se puede acceder al hardware

<sup>9</sup> Instituto Nacional de Tecnologías Educativas y de Formación del Profesorado (s.f). Blender: 3D en la educación. [Figura]. recuperado de. <http://www.ite.educacion.es/formacion/materiales/181/cd/m10/rigging.html>

del dispositivo, aunque no aprovecha de igual manera los recursos del dispositivo que una aplicación nativa, tienen acceso a gran parte de estos. Además, son aplicaciones que permiten de una manera más común, ser instaladas en diferentes dispositivos.

#### **6.2.4 Aprendizaje**

Existe gran variedad de teorías que hablan del comportamiento del ser humano, estas teorías acerca del aprendizaje tratan de explicar los desarrollos internos cuando se aprende, por ejemplo, cuando se adquiere una habilidad intelectual, la obtención de información o conceptos, e incluso las destrezas motoras o actitudes que se poseen.

Hergenhahn (1976) define el aprendizaje como “un cambio relativamente permanente en la conducta o en su potencialidad que se produce a partir de la experiencia y que no puede ser atribuido a un estado temporal somático inducido por la enfermedad, la fatiga o las drogas.” (Citado en (Federacion de enseñanza CC.OO, 2009))

#### **6.2.5 Banco de imágenes**

Son sitios web, en los cuales los usuarios pueden acceder a cientos de miles de imágenes. Estas pueden ser de tipo gratuito y libres de derechos de autor, o de tipo paga, que, sus imágenes tienen derechos de autor, por lo cual, para acceder y utilizarlas es necesario pagar un costo monetario para esto.

#### **6.2.6 Blender**

Blender es una herramienta diseñada para el modelado en 3D de objetos, además de esta característica también cuenta con otras opciones tales como: texturizado, animación, renderización, pintura digital además de incluir otra opción de gran importancia para dicho software como lo es el contar con un motor de videojuegos interno. “Blender también tiene una memoria relativamente pequeña y requisitos de unidad en comparación con otras suites de creación 3D. Su interfaz utiliza OpenGL para proporcionar una experiencia consistente en todos los equipos y plataformas compatibles.” (Blender, s.f)

### **6.2.7 Android**

Es el sistema operativo en el cual se centró el proyecto, debido a que la aplicación móvil fue desarrollada para funcionar de una manera nativa en dispositivos que utilicen este sistema operativo.

El presente desarrollo está dirigido para sistemas operativos Android de 5.x o más recientes.

### **6.2.8 Desarrollo web**

Es el proceso en el cual se utilizan diferentes lenguajes de programación para la creación de una página en línea que por lo general se almacena en un servidor en internet. Comúnmente, se utilizan lenguajes de programación como pueden ser JavaScript, HTML o CSS, para desarrollar dichos portales para una o muchas funciones que se deseen realizar allí.

### **6.2.9 Educación moderna**

Hace referencia a la educación que ha venido evolucionando, desde el siglo XX hasta la actualidad, debido a la intervención de los avances de la ciencia y la tecnología en la sociedad.

### **6.2.10 Herramientas UML**

Es una herramienta que se utiliza para el análisis de un software en particular. Se usa para el desarrollo de diagramas de lenguaje unificado de modelado, debido a que se puede observar de una manera más sencilla, el funcionamiento de un software. Estas herramientas permiten realizar esquemas o bosquejos XML o también para modelos de bases de datos.

### **6.2.11 Interfaz de usuario**

En el desarrollo de software, una interfaz de usuario es un grupo de formas y métodos que dan la posibilidad al usuario de interactuar con un sistema. cuando se habla del conjunto de formas se hace referencia a botones, ventanas, iconos y fuentes de letra, los cuales representan acciones y funciones dentro del sistema en el que esta implementada la interfaz. Dicho de otra manera, se puede considerar como un apartado grafico en el cual el usuario puede interactuar con el software. En este apartado se le presenta información al usuario, la cual él debe entender e interpretar para decidir qué acciones quiere que el sistema ejecute en base a las entradas dadas por usuario mismo.

“Entonces, el software interpreta esas entradas y genera cambios internos en los modelos que representan la información con la que el usuario está interactuando.” (Oslen s.f.)

### **6.2.12 Lenguaje de programación**

Es todo aquel conjunto de instrucciones que permiten la elaboración de un programa, permitiendo el manejo y control de las funcionalidades y características de este. En palabras más específicas “Un lenguaje de programación consiste en un conjunto de órdenes o comandos que describen el proceso deseado. Cada lenguaje tiene sus instrucciones y enunciados verbales propios, que se combinan para formar los programas de cómputo.” (Sistema Universidad Abierta y Educación a Distancia de la UNAM s.f.)

### **6.2.13 Modelado 3D**

Es un proceso de software, en donde el “modelador 3D da forma a los objetos virtuales mediante diferentes técnicas” (Ruiz, 2011). Las técnicas más utilizadas en el mundo del modelado en 3D son el modelado en caja, el cual consiste en crear un objeto a partir de la modificación de una figura primitiva como puede ser un plano o un cubo. El modelado de tipo escultórico, el cual se basa en la simulación de presión, estiramiento y aplastamiento de una malla 3D haciendo una similitud con el proceso de una escultura en arcilla, y por último, el modelado de superficies y curvas tipo NURB las cuales son “representaciones matemáticas de geometría en 3D capaces de describir cualquier forma con precisión, desde simples líneas, círculos, arcos, o curvas en 2D hasta los más complejos sólidos o superficies orgánicas de formas libres en 3D” (Rhinoceros, s.f.).

### **6.2.14 Motor gráfico**

Es un software especializado para la creación de aplicaciones o videojuegos que necesiten recursos de hardware para la creación de objetos en dos o tres dimensiones. Componentes como la escena, son de los más importantes, debido a que allí es en donde se diseña la maqueta de la aplicación o videojuego. Se podría decir, que, una escena puede representar un nivel distinto del juego o aplicación. Allí, se pueden crear objetos geométricos en dos o en tres dimensiones, además de muchos otros más.

También existe como en todo software especializado, el explorador, en donde se muestra la lista en donde se encuentran las carpetas que componen al proyecto, y por lo cual permite al

desarrollador mantener ordenados los diferentes archivos que se utilizarán en el proyecto, como imágenes, scripts, modelos tridimensionales, texturas, audios entre otros.

Por otra parte, en este tipo de herramientas, existe un elemento que se conoce generalmente como el inspector, el cual permite al desarrollador ver las propiedades y definir variables, archivos o atributos de un objeto en escena.

Además, en este tipo de software, existe comúnmente una función que permite visualizar el proyecto en ejecución en tiempo real que se puede generar en distintas resoluciones.

### **6.2.15 Open Source**

Hace referencia a código abierto. “Los programas informáticos de código abierto son la parte del ecosistema de los programas informáticos que ofrece a los programadores y usuarios de programas informáticos una forma alternativa de desarrollar y distribuir programas informáticos” (Davidson J, 2004).

En el mercado existen desarrollos informáticos de código abierto que son utilizados para el desarrollo de software, como pueden ser sistemas operativos, aplicativos o programas con funciones específicas.

### **6.2.16 Software educativo**

Se considera software educativo a una herramienta desarrollada principalmente para el ámbito educativo, esta herramienta tiene como objetivo principal la enseñanza o el auto-aprendizaje a niños o adultos de algún tema educativo específico, además de permitir realizar pruebas y seguimientos del avance de la persona que está aprendiendo.

En la actualidad este tipo de software está diseñado tanto para computadores de escritorio como para dispositivos móviles. Así mismo está desarrollado para uso doméstico o también para uso educativo institucional.

La obtención de un software educativo puede ser a través de tiendas virtuales de software para computadores o a mediante tiendas de aplicaciones para dispositivos móviles.

### **6.2.17 Texturizado**

Es el proceso de dar color a un objeto en dos o tres dimensiones, además, “permite simular diferentes materiales (metal, madera, etc.) y dar mayor detalle a determinadas formas. Las texturas pueden pintarse en un software de creación de imágenes digitales o puede extraerse de fotografías de texturas reales.” (Ruiz, 2011)

### **6.2.18 Tecnologías de la información y comunicación (TIC)**

Hacen referencia a la nueva manera de utilizar las diferentes herramientas tecnologías digitales. Además, las TIC “se desarrollan a partir de los avances científicos producidos en los ámbitos de la informática y las telecomunicaciones. Las TIC son el conjunto de tecnologías que permiten el acceso, producción, tratamiento y comunicación de información presentada en diferentes códigos”. (Ortí s.f.)

## **6.3 Marco ingenieril**

### **6.3.1 Síntesis de habla**

La síntesis de habla conocida en inglés también como text-to-speech (TTS) es un proceso en el cual se busca que un dispositivo (hardware) o sistema (software) genere un audio simulando un habla humana artificial en base a un texto. “es una tecnología lingüística que hace posible que un ordenador transforme automáticamente un texto escrito en su correspondiente forma sonora, lo más cercana posible a la lectura que realizaría una persona” (Llisterri, 2001).

Un sistema o software que emplea la síntesis de habla se compone en de dos partes las cuales son:

- Front-end
- Back-end

“A grandes rasgos, el front-end toma como entrada texto y produce una representación lingüística fonética. El back-end toma como entrada la representación lingüística simbólica y produce una forma de onda sintetizada” (Academic, s.f).

La calidad de un sistema text-to-speech, se puede clasificar de 2 maneras: por su inteligibilidad, lo que hace referencia a con qué facilidad o dificultad se puede reconocer y entender el habla artificial generada. Y la segunda es por su naturalidad, con esto se refiere a la medida en la que el habla se asemeja a la voz real de una persona.

### **6.3.2 Amazon Web Services (AWS) Polly**

Para el desarrollo de voces en la aplicación móvil fue necesario de este servicio. Gracias a que este mismo, es capaz de convertir el texto en habla realista, “lo que permite crear aplicaciones y categorías totalmente nuevas de productos con esta capacidad. Amazon Polly es un servicio de texto a voz que utiliza tecnologías de aprendizaje profundo avanzadas para sintetizar habla que se asemeja a una voz humana.” (Amazon, 2019)

### **6.3.3 Android**

Android es un sistema operativo creado por Google, y diseñado para ser implementado en dispositivos con pantallas táctiles, es decir dispositivos móviles tales como Tablet, teléfonos, relojes, aunque en la actualidad ya se encuentra este sistema operativo incluido en televisores e incluso en automóviles. Este sistema está basado en el kernel de Linux y otros softwares de código abierto, lo que lo hace un Open Source y gratuito. Android posee una serie de librerías en C y C++ las cuales permiten el desarrollo de aplicaciones mediante la utilización de Dalvik, una adaptación de la máquina virtual de Java, pero para dispositivos con una capacidad limitada de memoria. Además, cuenta con una serie de API's para el uso de diferentes elementos del dispositivo como, por ejemplo: GPS, giroscopio, acelerómetro, sensores de proximidad, etc.

### **6.3.4 Blender**

Blender es una suite de creación de objetos en 3D gratuita y de código abierto. Cuenta con una gran diversidad de opciones y herramientas tales como el modelado, animación, renderizado, rigging o sistema de huesos, sistema de partículas, edición de video, sistema de nodos para la creación de texturas y materiales que tienen una mayor complejidad y profesionalismo, además de contar con un motor de juegos interno, Además de las funciones mencionadas anteriormente, los usuarios más avanzados en el tema usan la API de Blender para programar scripts en Python que sirven para añadir nuevas herramientas especializadas al propio software como tal. “A menudo estos se incluyen en futuras versiones de Blender. Blender está bien adaptado a individuos y

pequeños estudios que se benefician de su proceso de desarrollo unificado de tuberías y respuesta.” (Blender, s.f). Cabe añadir que Blender es un software multiplataforma con soporte para sistemas operativos Linux, Windows y Mac. Su interfaz utiliza OpenGL para proporcionar una experiencia consistente. Algunas procesos y actividades que se pueden realizar en Blender son:

- **Transform (Modelado)**

Es un apartado de Blender, en donde se modela el objeto en escena, utilizando las tres principales funciones (Mover, Rotar y Escalar) para transformar dicho objeto, además existen otras funciones importantes para lograr el modelado que el usuario desea. Por otra parte, se puede transformar cada arista y vértice del objeto o en conjunto mediante la herramienta selección de Blender. A continuación, se observa una captura de pantalla donde se observa el panel de las funciones de transformación a la izquierda y a la derecha la escena en donde se pueden modificar las aristas y vértices.

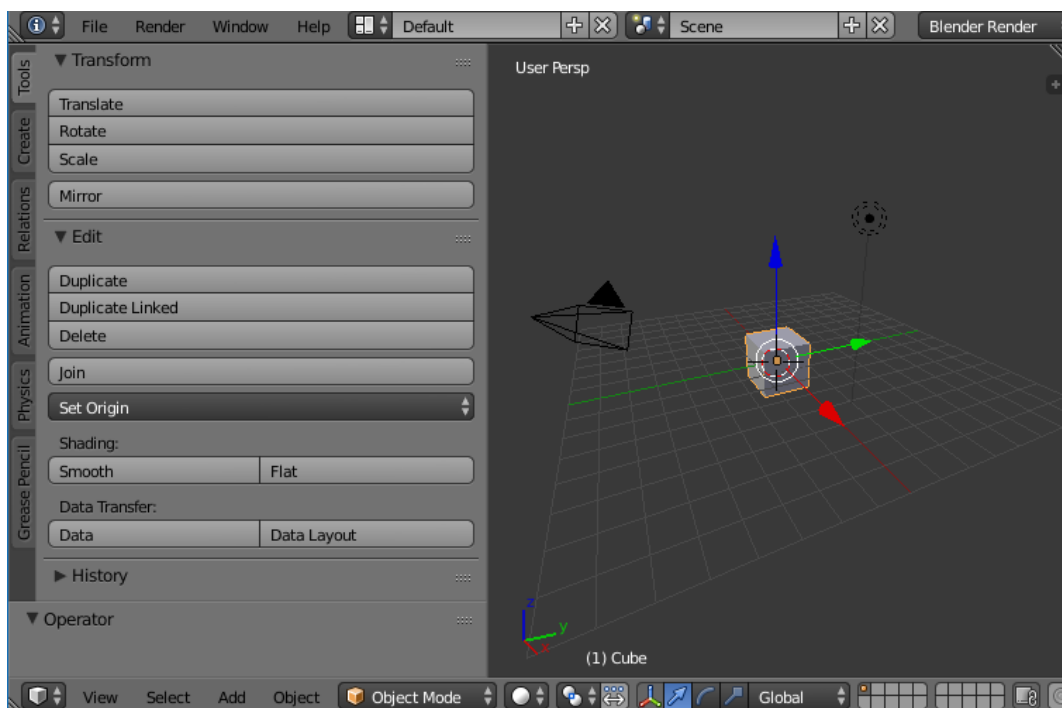


Figura 10. Captura de pantalla del panel transform y edit en Blender. Elaboración propia.

- **Material & Texture (Texturizado)**

La función de texturizado de Blender se encuentra en un panel, que, por lo general se encuentra en la parte derecha del programa. A continuación, se muestra una imagen de este panel.

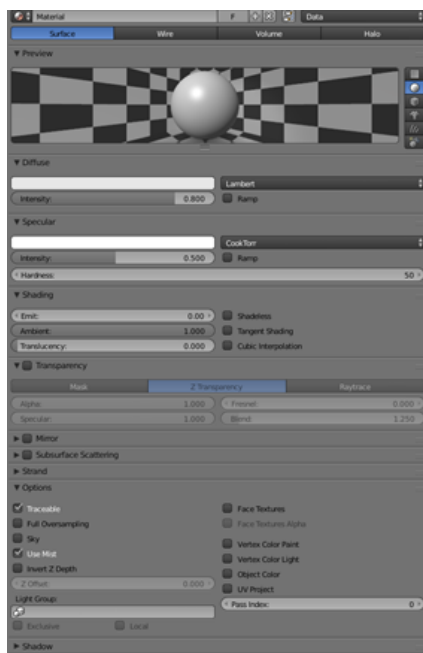


Figura 11. Panel Material de Blender. Elaboración propia

Allí se encuentra una región en donde se puede apreciar el material aplicado en una esfera, esto para tener una prevista de este. En la parte de abajo existen las propiedades Difuse, Specular, Shading, Transparency, entre otras. Estas propiedades permiten darle realismo al objeto mediante el material editado.

Para la parte, el panel de texturizado que se muestra en la figura 12, tiene como función principal, la inserción de imágenes de texturas a un material existente. Además, se pueden modificar propiedades para darle el realismo necesario a cada imagen de textura.

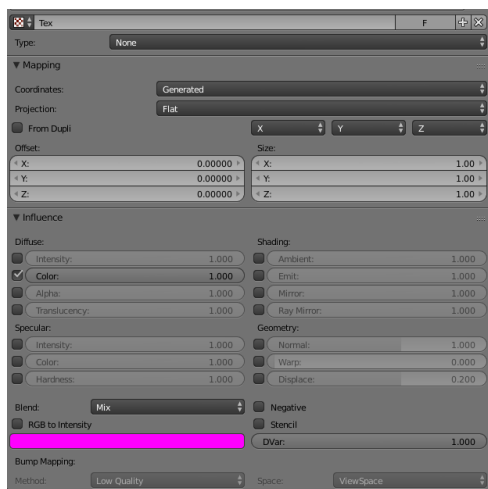


Figura 12. Panel Texture de Blender. Elaboración propia

### 6.3.5 C#

Pronunciado como C-Sharp, es un lenguaje estándar de desarrollo de software similar a Java o C++, orientado a objetos y fue creado por la empresa Microsoft. “La sintaxis y estructuración de C# es muy similar a la C++, ya que la intención de Microsoft con C# es facilitar la migración de códigos escritos en estos lenguajes a C# y facilitar su aprendizaje a los desarrolladores habituados a ellos.” (González s.f.)

### 6.3.6 GIMP

GIMP es un software de creación, edición, composición y retoque de imágenes, diseñado para ser multiplataforma, este mismo puede ser instalado en sistemas operativos Windows, Linux y Mac OS. El acrónimo GIMP viene de GNU Image Manipulation Program, es gratis y de código abierto el cual brinda la posibilidad de modificar su código fuente para crear nuevas versiones y así poder distribuirlo.

“Mediante GIMP podemos crear y editar imágenes de mapas de bits principalmente. También podemos trabajar con imágenes vectoriales con el plug-in correspondiente, pero no es el programa más adecuado para tratar imágenes vectoriales complejas” (Sistemas Multimedia e Interacción Gráfica, 2006).

Al igual que otros programas de código abierto, GIMP permite crear scripts tanto en Perl Python que con la finalidad de agregar nuevas funcionalidades al software e incluso permitir la automatización de funciones ya existentes por defecto.

El formato de archivos que GIMP usa por defecto es el formato XCF el cual “está concebido para conservar toda la información que GIMP puede contener con respecto a una imagen” (GIMP, 2018). Así mismo, se pueden importar y exportar imágenes en otros formatos más distinguidos como por ejemplo JPG el formato más conocido para compresión de imágenes en color y blanco y, PNG para imagen que posean la propiedad de transparencia y TIFF para imágenes con que cuenten con alta resolución de pixeles, entre otros formatos.

### **6.3.7 HTML**

Es un lenguaje de programación. Sus siglas significan HyperText Markup, lo que describe que es un lenguaje de marcas de hipertexto. Es un lenguaje que se usa para la creación de páginas web, por lo que se ha convertido en un estándar para agregar contenido multimedia a este tipo de desarrollos como imágenes, sonidos videos, entre otros.

### **6.3.8 CSS (Cascading Style Sheets)**

En español se traduce como hojas de estilo en cascada. Es un lenguaje de diseño de gráfico para detallar y desarrollar el diseño visual de interfaz de usuarios, páginas web, entre otros. Por lo general este lenguaje se trabaja con el lenguaje de marcado HTML para la creación de portales web.

### **6.3.9 Unity3D**

Unity es un motor de videojuegos creado por la compañía Unity Technologies, diseñado para desarrollar todo tipo de contenido interactivo tanto en 2D como en 3D compuesto por imágenes, audio, video y objetos en 3D. (Ouazzani, 2012) afirma que:

Este motor no permite la modelización, pero permite crear escenas que soportan iluminación, terrenos, cámaras, texturas. Fue creado en un principio para la plataforma Mac y ha sido exportado a Windows, permite obtener aplicaciones compatibles con Windows, Mac OS X, iOS, Android, Wii, PlayStation 3, Xbox 360, Nintendo, iPad, iPhone con Web gracias a un plugin y recientemente desde la versión 3.5 con el formato Flash de Adobe.

Unity3D cuenta con 3 tipos de licencia: la licencia de uso personal diseñada para principiantes, la licencia plus ideal para personas experimentadas o aficionados y, por último, la versión pro para grandes equipos de desarrollo o FreeLancer (Unity Technologies, 2019).

Cabe destacar que Unity tiene una comunidad de desarrollo muy grande, la cual produce una gran variedad de controladores para poder integrar otras herramientas a este o también puede leer una gran variedad de formatos de archivos tridimensionales como: .fbx, .dae, .3ds, .obj, .blend entre otros.

Dentro de Unity, hay elementos y propiedades que son los más básicos, pero a su vez más utilizados para el desarrollo de cualquier juego y aplicación los cuales son:

- **GameObject**

Es el tipo de objeto que más se utiliza en Unity, porque tiene un alcance y utilidad bastante importante, debido a que, en sí, son objetos vacíos y se le pueden agregar innumerables propiedades, como audios, colliders, prefabs, controlador de animaciones, modelos tridimensionales, materiales, objetos especiales como Canvas, entre otras.

- **Collider**

Es una propiedad que se le asigna a un GameObject, la cual es un área específica, que por lo general se utiliza para establecer un límite físico con otro GameObject que también contenga un Collider. Pero también tiene la función de Collider as Trigger, la cual es detectar cuando un objeto entra en el espacio de dicho Collider. Cabe resaltar que el otro objeto que entra dentro de este Collider también debe tener esta propiedad, pero no como un Trigger. Entonces, la función de estos Collider as Trigger, es de iniciar un evento cuando se detecta otro objeto en su espacio.

- **Canvas**

El Canvas es un área en donde se agregan todos los elementos que hacen parte de la interfaz gráfica de usuario. “El Canvas es un GameObject con un componente Canvas en él, y todos los elementos UI deben ser hijos de dicho Canvas” (Unity Technologies, 2018). Dicho de otra manera, es el elemento padre que contendrá todos los objetos que conformen la interfaz para poder ser mostrados en pantalla, estos objetos pueden ser: textos, botones, sliders, check box, campos de texto y demás objetos que Unity tiene a disposición.

- **Script**

La programación de un GameObject depende de los componentes que este contenga, por lo general esto se detalla en un Script, que, por lo general, es una clase en donde se le asignan métodos, condicionales y bucles a la programación del objeto que contenga dicho Script.

- **Vector 3**

Más que una propiedad, es una variable, que almacena una coordenada en el espacio de la escena de un GameObject, y también se utiliza para dar una dirección de traslación de una animación de un objeto.

- **Lenguaje de programación de Unity**

Unity tiene soporte nativo para un lenguaje de programación el cual es C# (pronunciado C-Sharp), un lenguaje estándar de la industria similar a Java o C++; Los scripts desarrollados en el actual proyecto fueron realizados utilizando dicho lenguaje.

Adicional a este, “otros lenguajes .NET pueden ser utilizados con Unity si estos compilan un DLL compatible.” (Unity Technologies, 2018)

### **6.3.10 Visual Studio**

Visual Studio es un entorno de desarrollo IDE el cual cuenta con un grupo de herramientas y tecnologías de desarrollo de software que se basa en componentes para crear aplicaciones con alto grado de eficacia y rendimiento, la cual permite a desarrolladores crear todo tipo de aplicaciones, desde aplicaciones, sitios y servicios web hasta aplicaciones móviles con el plug-in de Xamarin. Es un IDE multiplataforma, ya que se puede ejecutar en Windows, Linux y Mac OS (a partir de la versión 2017), además, cuenta con soporte para múltiples lenguajes de programación tales como C#, Visual Basic, C++, .NET, F#, Java, Python, Ruby y PHP (MSN, s.f.).

### **6.3.11 Realidad Aumentada**

La realidad aumentada es una tecnología que permite combinar el mundo real con elementos virtuales, mediante diferentes procesos informáticos y dispositivos que hacen esto posible. (Zenith, 2014) define a la realidad aumentada como:

El concepto que se utiliza para definir una visión a través de un dispositivo tecnológico, directa o indirecta, de un entorno físico del mundo real, cuyos elementos se combinan con elementos virtuales creando así una realidad mixta en tiempo real. Esta es la principal diferencia con la realidad virtual, puesto que no sustituye la realidad física, sino que sobreimprime datos virtuales al mundo real.

Esta tecnología se compone de 4 niveles. “Se pueden entender los niveles como una forma de medir la complejidad de las tecnologías involucradas en el desarrollo de sistemas de RA. En principio, a más nivel, mayores son las posibilidades de las aplicaciones” (Espinosa, 2015). Dichos niveles están enumerados de 0 a 3, los cuales son:

- **Nivel 0:** Los sistemas de realidad aumentada hiperenlazan el mundo físico mediante códigos de barras o códigos QR, estos códigos se encargan de crear enlaces a otros contenidos como por ejemplo una página web, por lo cual, no se genera ningún tipo de contenido multimedia a través de este, haciendo la función de una URL .html, pero sin la necesidad de escribirla en ningún navegador web.



Figura 13. Código de barras y Código QR.<sup>10</sup>

- **Nivel 1:** Este nivel es el reconocimiento de patrones 2D, está basado en la utilización de marcadores, los activadores o triggers son marcas impresas en blanco y negro, con formas sencillas o cuadrangulares, como también pueden ser imágenes a color impresas en papel. Hay que añadir que, en este nivel la forma más avanzada de reconocimiento es el de objetos físicos reales.

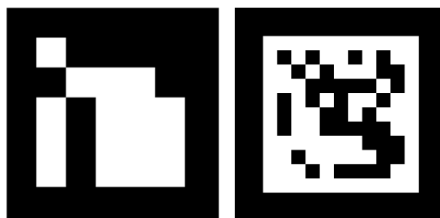


Figura 14. Ejemplo de marcadores nivel 1 de Realidad aumentada.<sup>11</sup>

<sup>10</sup> Caro. (2013). Códigos de barras vs Códigos QR. [Figura]. Recuperado de: <http://newlandsnewlove.blogspot.com/2013/10/codigo-de-barra-vs-codigo-qr.html>

<sup>11</sup> Zhats. (2016). AR Tracking with different makers. [Figura]. Recuperado de: <https://it-jim.com/2016/09/19/augmented-reality-tracking-with-different-markers/>

- **Nivel 2:** realidad aumentada sin marcadores conocido también como “Markeless AR”. El sistema hace uso del GPS, la brújula y el acelerómetro del dispositivo para localizar la posición y orientación del usuario para así poder superponer puntos de interés en las imágenes del mundo real.



Figura 15. Ejemplo de nivel 2 de Realidad aumentada.<sup>12</sup>

- **Nivel 3:** Visión Aumentada, este nivel se ve representado mediante el uso de dispositivos especiales como las gafas de Google “Google Glass”, las Hololens de Microsoft, “lentes de contacto de alta tecnología u otros que, en el futuro, serán capaces de ofrecer una experiencia completamente contextualizada, inversiva y personal” (López Maradiaga, Medina Yuen, & Octavo De La Cruz, 2018).

Son 4 elementos los que son parte fundamental del funcionamiento y que a su vez hacen posible la realidad aumentada los cuales son:

- **Cámara:** es el dispositivo que será el encargado de capturar la información del mundo real y la envía al sistema de realidad aumentada.
- **Sistema:** o software el cual es el encargado de procesar toda la información captada y recibida por la cámara.
- **Marcador:** conocido también como “trigger” que puede ser una imagen, un código QR, la silueta de un objeto o elementos del entorno físico como por ejemplo una escultura, o en general cualquier elemento que se considere como marcador según los niveles de la realidad aumentada el cual la cámara pueda reconocer y habilitar al sistema para que este lo interprete y en base a esa información muestra algún tipo de contenido de imagen, video, audio, un modelo en 3D o lo direcciona a un sitio web específico.

<sup>12</sup> (s.a). (2018). Everything you need to know build Location-Based AR App. [Figura]. Recuperado de: <https://blog.vakoms.com/everything-you-need-to-knowto-build-location-based-ar-app/>

- **Pantalla:** la cual permite proyectar todo el contenido anteriormente dicho para así poder generar nuevas experiencias interactivas. (Betancourth, 2009)

A continuación, en la figura 16 se muestra el esquema del funcionamiento de la realidad aumentada.

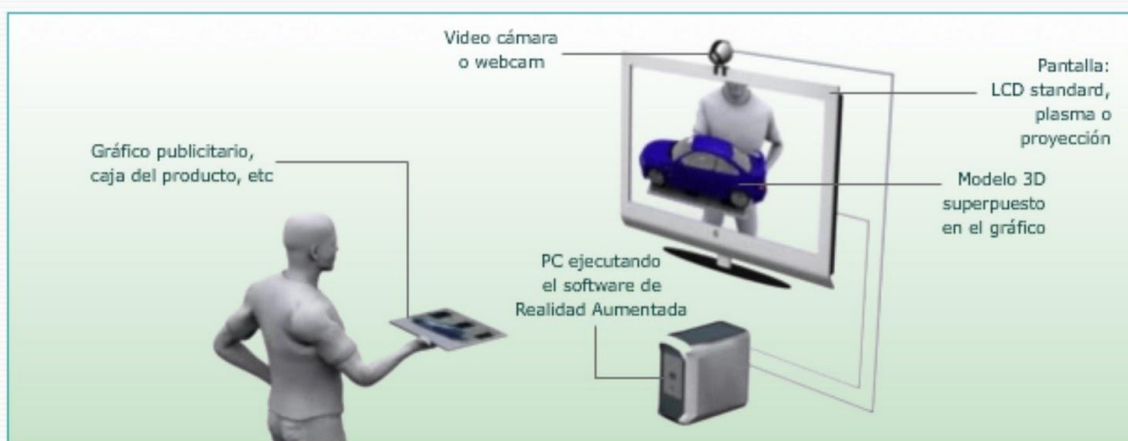


Figura 16. Esquema de funcionamiento de la realidad aumentada.<sup>13</sup>

### 6.3.12 Códigos QR

Las siglas QR hacen referencia a Quick Response (Respuesta Rápida), un código QR básicamente es un código de barras el cual posee dos dimensiones, “utilizado para contener información en una matriz de puntos bidimensional tanto horizontal como vertical que contiene información codificada en ella, a diferencia de los tradicionales códigos de barras, los cuales solo pueden almacenar información en una sola dimensión horizontal.” (Servicio de Sociedad de la Información. s.f.)

Estos códigos QR, cuentan con características que permiten almacenar mayor información que los convencionales códigos de barras. A continuación, se muestra una figura en donde se puede entender mejor dicha característica, mediante una comparación de los dos tipos de códigos.

<sup>13</sup> Yordan. (2011). Realidad Aumentada. [Figura]. Recuperado de: <http://realidad-aumentada-3d.blogspot.com/2011/02/la-realidad-aumentada-es-el-termino-que.html>



Figura 17. Comparación de un código QR y un código de barras.<sup>14</sup>

El funcionamiento de los códigos QR, radica en los tres cuadros grandes en las esquinas de éste. Además, también se observan otras partes del código, como se explica en la siguiente figura.

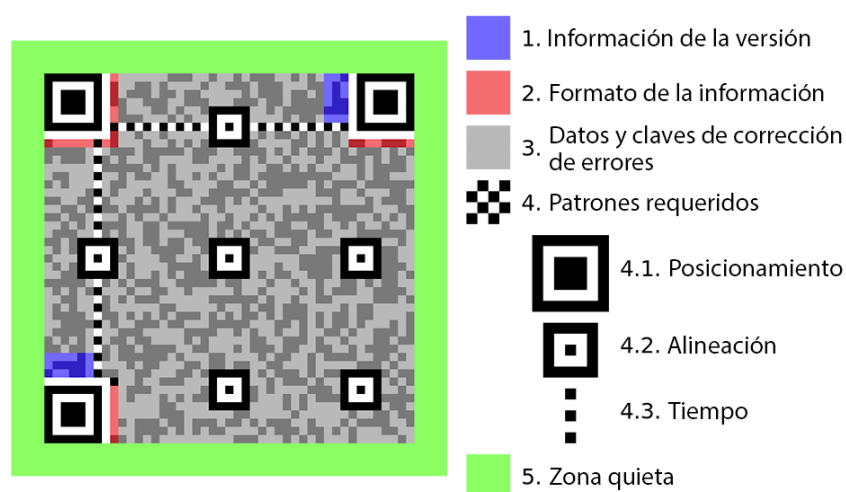


Figura 18. Partes de un código QR.<sup>15</sup>

Para poder decodificar este tipo de códigos “es necesario el uso de un teléfono móvil con cámara y de un programa que sea capaz de traducir este tipo de lenguaje. Estos códigos pueden ser visualizados desde cualquier ángulo sin perder la información original contenida en ellos.” (Servicio de Sociedad de la Información. s.f.). Dichos programas o aplicaciones se pueden obtener en cualquier tienda del dispositivo.

<sup>14</sup> (Servicio de Sociedad de la Información. s.f.). Comparación de un código QR y un código de barras. [Figura]. Recuperado de: [https://www.dipucadiz.es/export/sites/default/galeria\\_de\\_ficheros/sociedad\\_de\\_la\\_informacion/destacados/Manual.CodigosQR.pdf](https://www.dipucadiz.es/export/sites/default/galeria_de_ficheros/sociedad_de_la_informacion/destacados/Manual.CodigosQR.pdf)

<sup>15</sup> Lucas. (2016). 30 cosas que debes saber sobre los códigos QR. [Figura]. Recuperado de: <https://uqr.me/es/blog/cosas-que-debes-saber-sobre-codigos-qr/>

### 6.3.13 Vuforia

Vuforia es un kit de desarrollo de software (SDK) que permite la construcción de aplicaciones de realidad aumentada (AR) y Realidad Mixta (MR) para múltiples plataformas tales como: Sistemas operativos móviles como Android o iOS, PC's con Windows, y lentes AR como las HoloLens de Microsoft. Los desarrolladores pueden agregar la funcionalidad de realidad aumentada o Realidad Mixta a cualquier aplicación, permitiendo el reconocimiento de imágenes y objetos además de interactuar con espacios del mundo real.

- **Arquitectura de Vuforia**

La arquitectura de Vuforia se puede observar en la siguiente figura:

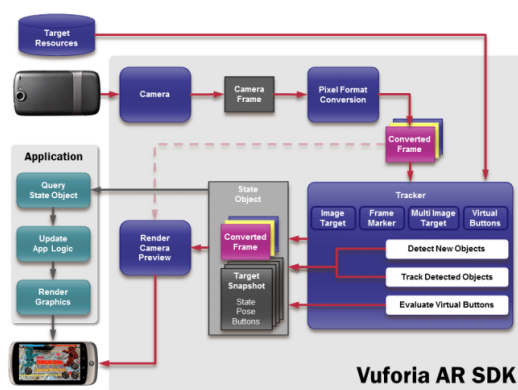


Figura 19. Arquitectura de Vuforia.<sup>16</sup>

Esta arquitectura se compone de una serie de elementos esenciales los cuales son:

- **Cámara:** a través de la cámara, el dispositivo capta un fotograma del escenario del mundo real.
- **Convertidor de imágenes:** Vuforia internamente realiza la conversión de densidad de píxeles del fotograma captado por la cámara, para posteriormente ser tratada mediante algoritmos de visión computacional, luego de este proceso, Vuforia se encarga de analizar la imagen en busca de coincidencias en las bases de datos (online y offline) donde se almacenan los marcadores.
- **Tracker:** este elemento se encarga de realizar el proceso de detección y seguimiento de los marcadores que están siendo capturados por la cámara.

<sup>16</sup> Che Mat R. (2015). Data flow diagram of the SDK. [Figura]. Recuperado de: [https://www.researchgate.net/figure/Data-flow-diagram-of-the-SDK-source-24\\_fig1\\_287797584](https://www.researchgate.net/figure/Data-flow-diagram-of-the-SDK-source-24_fig1_287797584)

- **Aplicación:** la cual contiene tiene integrada a Vuforia. Si en el proceso de conversión y tratamiento del fotograma se encuentra una coincidencia del marcador capturado y uno que este almacenado en las bases de datos mencionadas con anterioridad, entonces la aplicación realizara el proceso de renderizado del contenido que le fue asignado a dicho marcador (Imágenes, videos, modelos en 3D, reproductores de audio, etc.).

### 6.3.14 SDK para Unity

“La integración de Unity en Vuforia le permite crear aplicaciones y juegos de visión para Android e iOS utilizando un flujo de trabajo de creación de arrastrar y soltar.” (Unity Technologies, 2018).

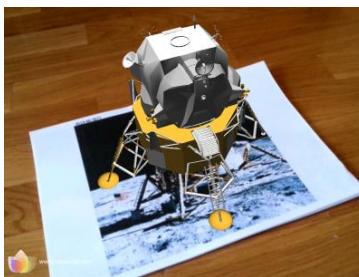


Figura 20. Uso de Vuforia junto a Unity en un marcador 2D.<sup>17</sup>

Para desarrollar una aplicación en Unity usando la SDK de Vuforia se debe contar con los marcadores digitales que van a ser usados y reconocidos por el dispositivo, con esto se hace referencia a imágenes, marcadores en blanco y negro o códigos QR. Estos marcadores deben ser subidos a la plataforma de desarrolladores de Vuforia más específicamente al apartado llamado “Target Manager” el cual permite 2 opciones: la primera es almacenar los marcadores mencionados anteriormente en una nube los cuales usara la aplicación de manera online, o bien descargar un fichero que es reconocido por Unity, el cual contiene la base de datos de los marcadores que la aplicación podrá reconocer de manera offline.

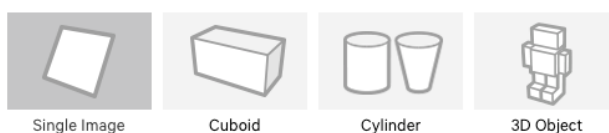


Figura 21. Tipos de marcadores (Triggers) reconocidos por Vuforia.<sup>18</sup>

<sup>17</sup> (Interactionstation, 2017). Marcador 2D siendo reconocido por una app que utiliza Vuforia. [Figura]. Recuperado de: <http://interactionstation.wdka.hro.nl/mediawiki/images/thumb/b/b8/AR.png/400px-AR.png>

<sup>18</sup> Unity Technologies. (2018). Tipos de marcadores reconocidos por Vuforia. [Figura]. Recuperado de: <https://docs.unity3d.com/2018.1/Documentation/Manual/vuforia-sdk-overview.html>

## CAPITULO 3

### 7. METODOLOGIA

Elegir una metodología para el desarrollo de un software es vital para el éxito del mismo, siendo la metodología el marco que permite la planificación, la estructuración y el control de un proyecto de desarrollo (Medina Velandia & López López, 2015). Se deben seguir actividades que forman parte de cada etapa, de tal manera que, si se realizan de manera ordenada, el producto final será entendible, sencillo y de fácil manejo. Así mismo, hay que tener en cuenta otros dos factores que son claves para el desarrollo de un proyecto de software, que son: el equipo de trabajo y el tiempo que este tiene para la elaboración del mismo; es en este punto en donde se optó por elegir la metodología ágil Mobile-D para el desarrollo de la aplicación móvil, esta se caracteriza por tener ciclos de desarrollo muy rápidos en un corto periodo de tiempo además de estar diseñada para equipos de trabajo pequeños. a esta metodología se le realiza una adaptación en cuanto al entregable de modelado de datos ya que no es requerida por la metodología.

En la metodología Mobile-D, el ciclo del proyecto se compone de distintas fases que son: exploración, inicialización, fase de producto, fase de estabilización y fase de pruebas (Blanco, Camarero, Fumero, Werterski, & Rodriguez, 2009).

#### **Fase de exploración**

En la fase de exploración se realizó una definición del alcance del proyecto, se delimito lo que se quería hacer y hasta a donde se quiso llegar, así como también se definieron los conceptos básicos. El entregable de esta fase fue el pliego de requisitos y los casos de uso.

#### **Fase de inicialización**

En la segunda fase que es la fase de inicialización se realizó el modelado del proyecto con el fin de identificar y preparar todos los elementos y recursos que fueron necesarios para la elaboración. Se realizó una búsqueda y análisis de los conceptos básicos y el uso de las herramientas que se iban a usar. Se hizo entrega del cronograma de actividades, las etapas del desarrollo, y el Wireframe de la aplicación.

### Fase de producción

En la fase de producción básicamente lo que se hizo fue repetir la programación del desarrollo, se planifico, se elaboró y se finalizó cada funcionalidad del desarrollo iterativamente con la finalidad de verificar que todo el desarrollo hasta el momento funciono de una manera correcta. El entregable en esta fase fue el ejecutable del prototipo base de la aplicación.

### Fase de estabilización

En la fase de estabilización se realizaron las tareas de integración del proyecto para convertirlo en uno solo. Se ensablo el código fuente, modelos en 3D, imágenes, y demás elementos que conforman la aplicación. Adicionalmente a partir de esta fase se elaboró toda la documentación del proyecto. El Entregable de esta fase fue el instalador de la aplicación móvil y los manuales técnico y de usuario.

### Fase de pruebas

Y la última fase es la fase de pruebas en la cual se realizó un test a toda la aplicación móvil. Una vez terminada la aplicación se realizaron todas las pruebas que fueron necesarias para tener una aplicación totalmente estable y funcional. La entrega de esta fase fue la aplicación terminada.



Figura 22. Etapas de desarrollo Mobile-D.<sup>19</sup>

<sup>19</sup> Marulanda I, Arias F. (2014). Etapas de desarrollo de la metodología Mobile-D. [Figura]. Recuperado de: <https://docplayer.es/619523-Ismael-calle-marulanda.html>

## 8. DESARROLLO DEL PROYECTO

### 8.1 Exploración

El entregable en esta fase es el pliego de requisitos y los casos de uso.

De acuerdo a las funcionalidades que se planean implementar en la aplicación, se determinan cuáles serán los requisitos funcionales, requisitos no funcionales, las restricciones del aplicativo y las reglas del negocio.

#### 8.1.1 Requisitos funcionales

Se describen las actividades que la aplicación realiza, además de definir cómo se comporta frente a las acciones solicitadas por el usuario, en la figura 23, se observa un esquema de 4 funcionalidades esenciales, las cuales integran el software.

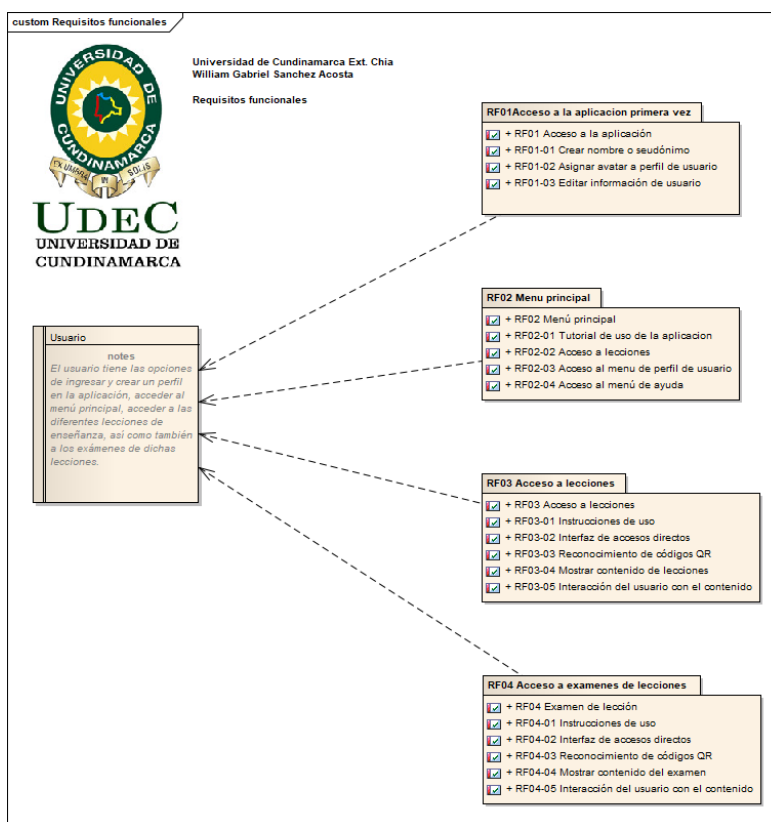


Figura 23. Requisitos funcionales. Elaboración propia

Dichas funcionalidades se sub-dividen para especificar más a detalle cómo debe responder la aplicación en diferentes escenarios (pantallas) de uso, a continuación, se da una vista detallada de los requisitos funcionales de cada escenario:

## Requisitos funcionales: Acceso a la aplicación por primera vez

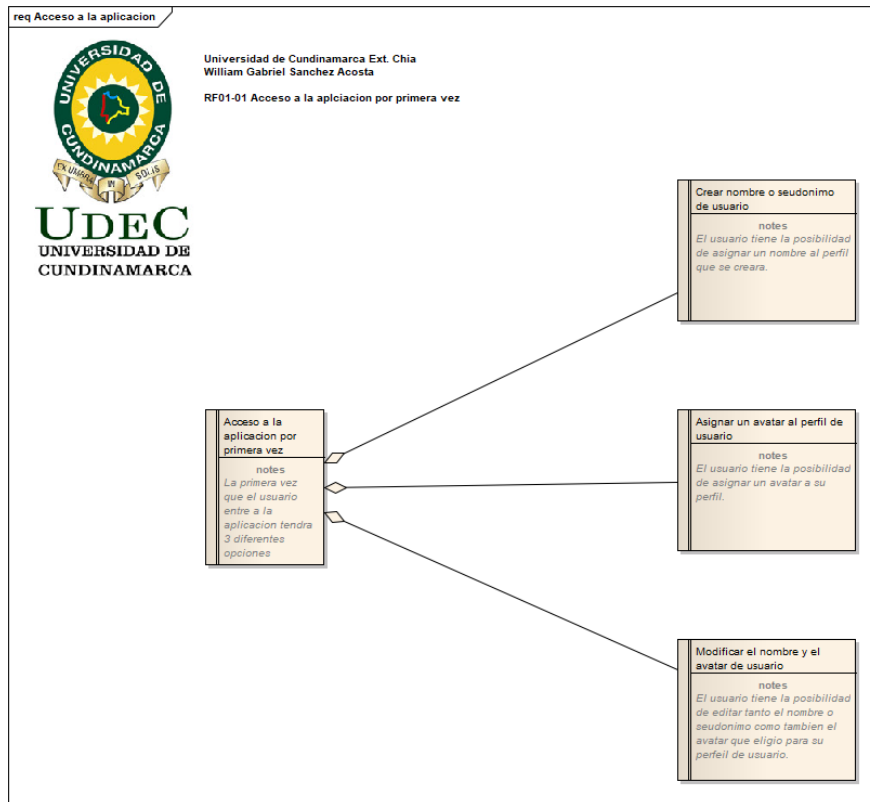


Figura 24. Requisitos funcionales: acceso a la aplicación por primera vez. Elaboración propia

En la figura anterior, se muestran las funcionalidades que incluye la aplicación al momento de que el usuario haga uso de esta por primera vez. En donde, se observan 3 opciones que el sistema proporciona al usuario, las cuales son: crear nombre o seudónimo de usuario, asignar un avatar al perfil de usuario y la opción de modificar la información anteriormente dada por el usuario.

- **Requisitos funcionales: Menú principal**

En la siguiente figura, se observa el diagrama que contiene las funcionalidades que debe tener el aplicativo cuando el usuario se encuentre en el menú principal.

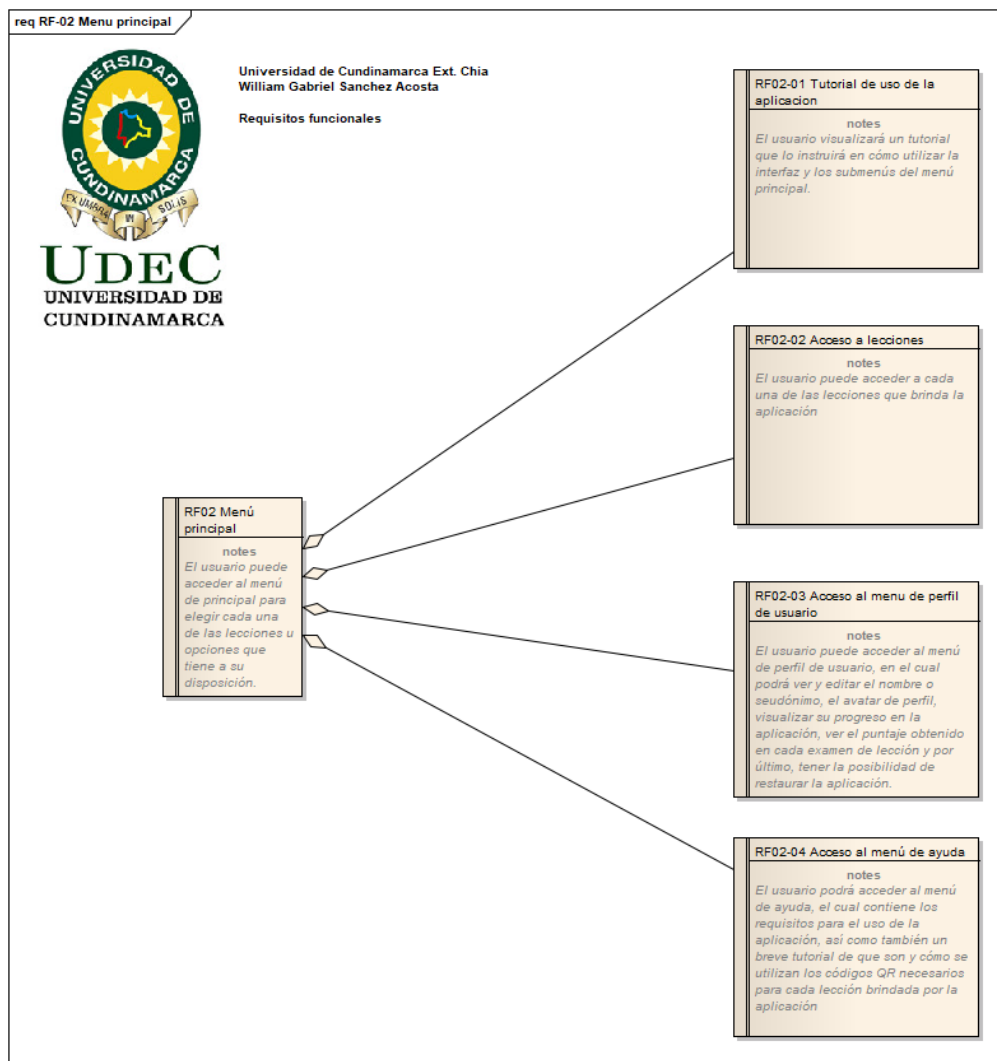


Figura 25. Requisitos Funcionales: Menú principal. Elaboración propia

- **Requisitos funcionales: Acceso a lecciones**

En la figura 26, se muestran los requisitos funcionales que la aplicación debe suplir en el escenario de lecciones de enseñanza, los cuales son: mostrar instrucciones de uso, contar con una interfaz de accesos directos a diferentes funciones, ser capaz de reconocer códigos QR y a su vez proyectar el contenido necesario de cada lección, además de permitir que el usuario pueda interactuar con este contenido.

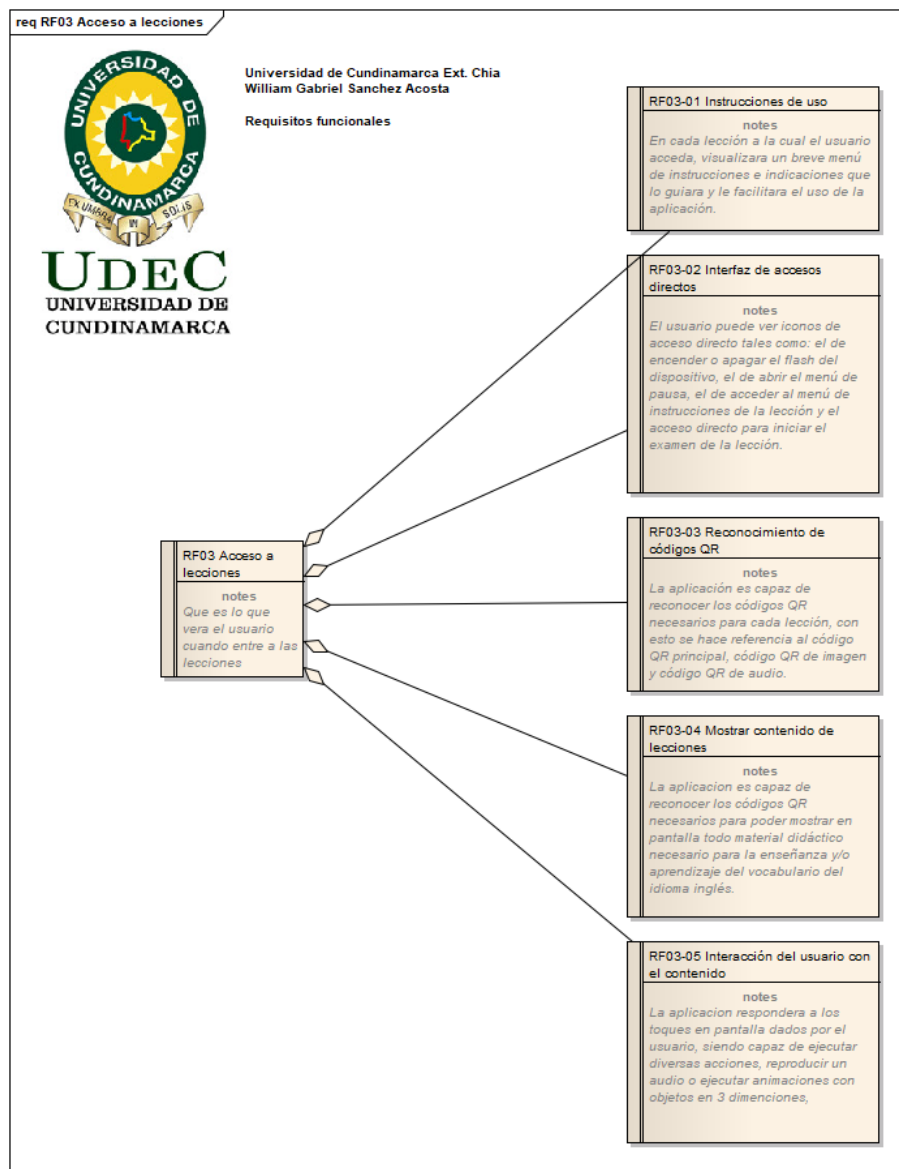


Figura 26. Requisitos Funcionales: Acceso a lecciones. Elaboración propia.

- **Requisitos funcionales: acceso a exámenes de lecciones.**

Las funcionalidades que debe incluir la aplicación en este escenario son muy similares a las funcionalidades cuando se accede a una lección ya que la dinámica es la misma, tal como se puede apreciar en la figura 27.

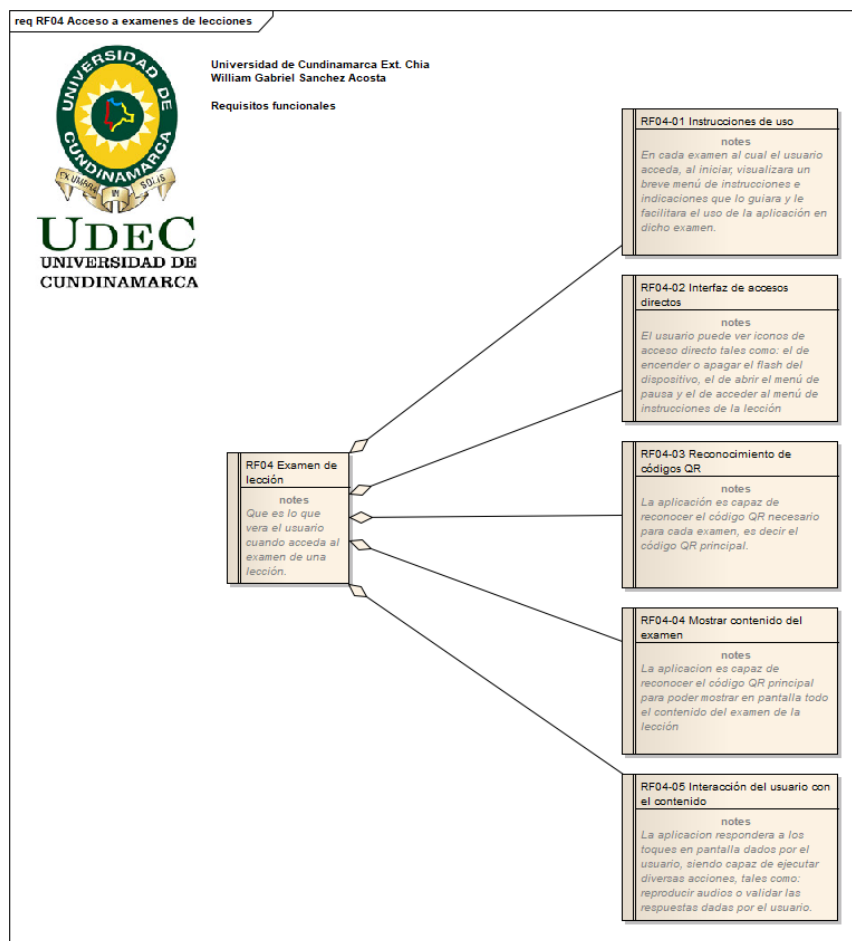


Figura 27. Requisitos Funcionales: Acceso a exámenes de lección. Elaboración propia.

### 8.1.2 Requisitos no funcionales

Los requisitos no funcionales representan características generales y restricciones de la aplicación, estos requisitos se basan en las funcionalidades que no están directamente relacionadas con el desarrollo. Dichos requisitos se resaltan en la figura 28.

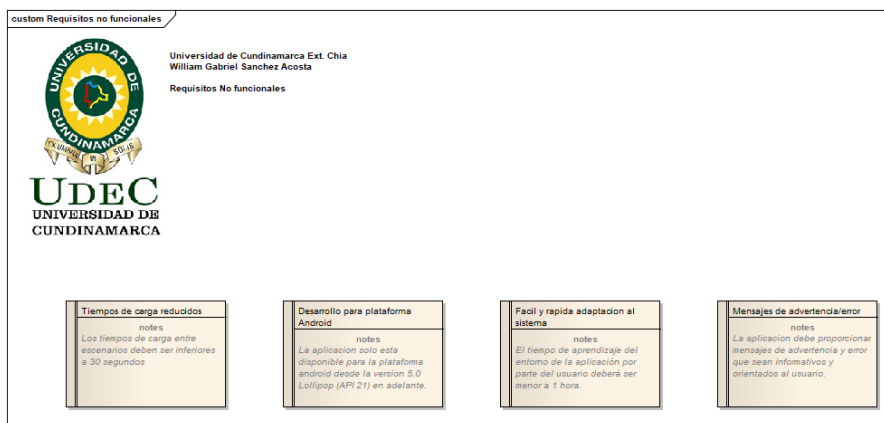


Figura 28. Requisitos no funcionales. Elaboración propia.

### 8.1.3 Reglas de negocio

Las reglas del negocio definen o delimitan un aspecto del negocio con la finalidad de establecer una estructura que condiciona el comportamiento de los actores del negocio, dicho de otra manera, las reglas del negocio tienen como finalidad establecer las reglas que se pactaran entre el usuario y el desarrollador, dichas reglas pueden ser: El uso de archivos multimedia en los escenarios de lección, implementación de exámenes de conocimiento y disponibilidad de la aplicación, tal como se aprecia en la figura 29.

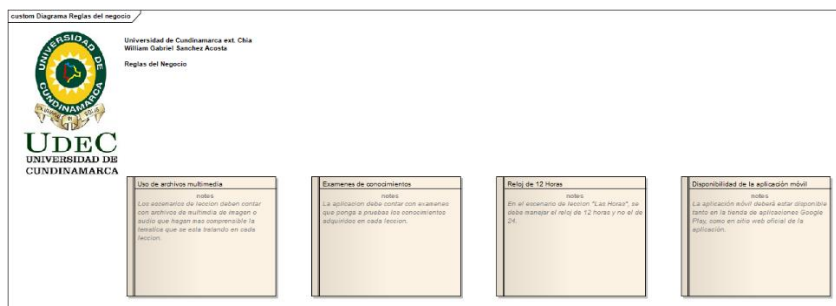


Figura 29. Reglas del negocio. Elaboración propia

### 8.1.4 Restricciones

Se determinan las diferentes restricciones las cuales limitan al aplicativo, estas restricciones pueden ser: el uso exclusivo de la aplicación en el sistema operativo Android o que la aplicación ocupe un tamaño determinado en la memoria del dispositivo en el cual se desea instalar, tal como se muestra en la figura 30.

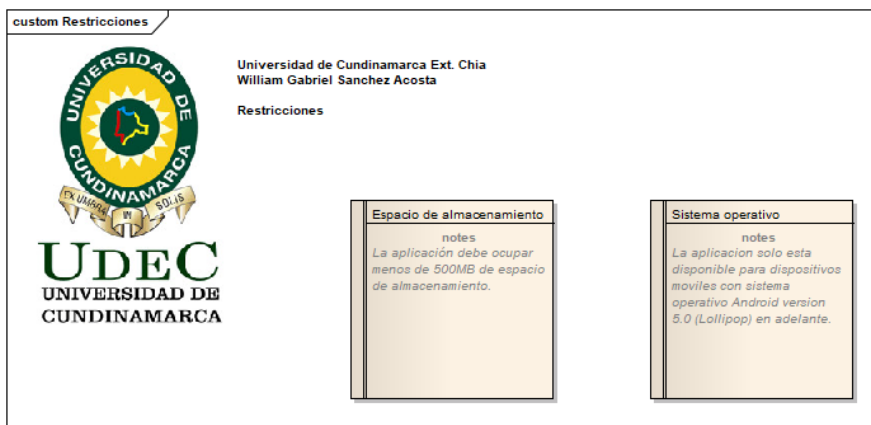


Figura 30. Restricciones. Elaboración propia

### 8.1.5 Casos de uso

Los casos de uso describen las acciones de la aplicación desde la perspectiva del usuario, proporcionan información de cómo deberá interactuar el sistema con el usuario para ejecutar una función específica.

En la siguiente figura se muestra el diagrama de casos de uso en el cual se exponen las principales acciones brindadas por el sistema las cuales el usuario tendrá disponibles para su usabilidad.

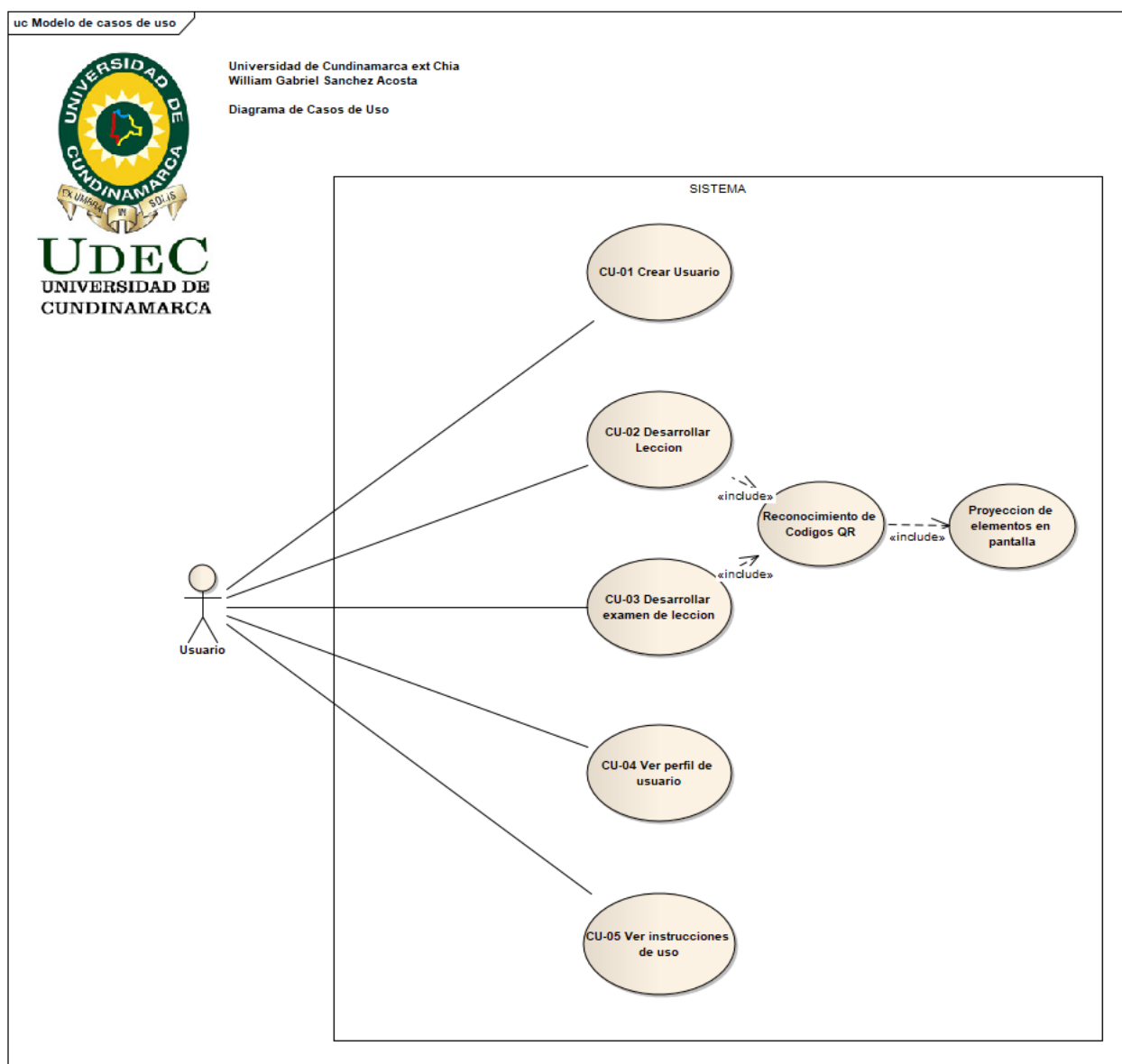


Figura 31. Diagrama de casos de uso. Elaboración propia

- **Crear usuario:** Este caso de uso contiene la creación y modificación de un perfil de usuario en este caso, el usuario será el encargado de la apertura de la aplicación, el sistema deberá mostrar una serie de interfaces las cuales servirán como formulario para solicitar información básica al usuario para la posterior creación de un perfil de usuario. De igual manera, esta información podrá ser modificada nuevamente por el usuario en cualquier otro momento.

<b>Caso de uso: CU-01 CREAR DE USUARIO</b>	
<b>Actor: Usuario</b>	
<b>Descripción de los procesos:</b>	<b>Alternativas:</b>
El usuario al acceder al sistema por primera vez, este último deberá mostrar una secuencia de interfaces, las cuales solicitarán información básica del usuario como el nombre y un avatar para el perfil de usuario.	El sistema deberá validar el nombre de usuario, debe verificar que cumpla con las condiciones de mínimo y máximo de caracteres, así como también los caracteres que están permitidos.
El sistema deberá presentar la primera interfaz que le solicitará el nombre de perfil al usuario.	
El usuario deberá designar un nombre de perfil, digitándolo en el cuadro de texto que aparecerá en pantalla.	
El sistema deberá validar la información de texto introducida por el usuario.	Si el nombre de perfil, tiene menos de 3 caracteres de texto el sistema debe mostrar un mensaje de advertencia al usuario. Así mismo el sistema validará si el nombre tiene menos de 40 caracteres, de no cumplirse dicha condición, deberá mostrar otro mensaje de advertencia al usuario.
El sistema deberá mostrar la siguiente interfaz que mostrara una lista de avatares disponibles para la selección del perfil de usuario.	
El usuario deberá elegir un avatar de la lista de avatares brindada por el sistema.	
El sistema deberá redirigir al usuario al menú principal.	
<b>Comentarios</b>	
Si el usuario ya accedió por primera vez al sistema, este último debe omitir la muestra de interfaces de solicitud de información, y deberá ir directamente al menú principal.	

*Cuadro 1. Caso de uso: Crear usuario. Elaboración propia.*

- Desarrollar lección:** En este caso de uso, el sistema deberá permitir la visualización del contenido de la lección elegida por el usuario, a través del reconocimiento de los códigos QR captados por la cámara del dispositivo, así mismo, el sistema deberá mostrar al usuario una interfaz que le ofrecerá al usuario una serie de opciones adicionales. Cabe añadir que este caso de uso aplica también al caso de uso “Desarrollar examen de lección” ya que estos dos poseen las mismas dinámicas de funcionamiento y tanto el usuario como el sistema se comportaran de la misma manera en ambos casos.

<b>Caso de uso: CU-02 Desarrollar Lección</b>	
<b>Actor: Usuario</b>	
<b>Descripción de los procesos</b>	<b>Alternativas</b>
El usuario tendrá a disposición la lista de lecciones disponibles dadas por la aplicación.	
El sistema deberá ser capaz de dirigir al usuario a la lección seleccionada por el mismo.	
Mediante el uso de los códigos QR, el usuario indicara al sistema el contenido que desea ver, ubicando el código QR frente a la cámara del dispositivo.	
<p>El sistema deberá ser capaz de reconocer los códigos QR captados por la cámara del dispositivo, y a su vez proyectar el contenido que le será asignado para mostrar. Es decir, que según sea el código QR, el sistema deberá presentar:</p> <ul style="list-style-type: none"> <li>- Modelos en 3D.</li> <li>- Imágenes.</li> <li>- Controles para la reproducción de audios.</li> </ul>	
<p>El sistema aparte del reconocimiento de los marcadores, deberá proveer al usuario de una interfaz que contenga diferentes opciones que sirvan como apoyo a la lección.</p> <p>Estas opciones son:</p> <ul style="list-style-type: none"> <li>- Cambiar el estado de la luz flash del dispositivo (encenderla o apagarla).</li> <li>- Mostrar un menú de pausa, que contendrá ajustes de audio, opción para mostrar un tutorial de uso y la opción de salir al menú principal.</li> <li>- Mostrar las instrucciones de la lección.</li> <li>- Opción para presentar el examen de la lección.</li> </ul>	
El usuario al interactuar con la opción de cambiar estado de luz flash del dispositivo: el sistema deberá acceder al hardware del dispositivo para encender o apagar el flash del mismo.	

El usuario al interactuar con la opción de Mostrar menú de pausa: el sistema deberá mostrar al usuario un menú el cual contiene opciones para ajustes de audio de la aplicación, mostrar un tutorial de uso de la aplicación, y la opción de regresar al menú principal.	
El usuario al interactuar con la opción de mostrar instrucciones de uso (en lección): el sistema deberá mostrar un menú el cual le muestre al usuario las instrucciones de uso de los elementos proyectados por el código QR mostrados en la pantalla del dispositivo.	
El usuario al interactuar con la opción de acceder al examen de lección: el sistema deberá redirigir al usuario al Escenario de Examen de Lección.	
<b>Comentarios</b>	
La opción “Elegir lección” hace referencia a los 12 botones disponibles para cada una de las lecciones	

*Cuadro 2. Caso de uso: Desarrollar Lección. Elaboración propia.*

- **Menú de perfil de usuario:** El caso de uso muestra las opciones que el usuario tendrá disponibles las cuales serán mostradas por el sistema, al momento de que el usuario acceda al menú del perfil de usuario a través de la opción del menú principal.

<b>Caso de uso: CU-04 VER PERFIL DE USUARIO</b>	
<b>Actor: Usuario</b>	
<b>Descripción de los procesos</b>	<b>Alternativas</b>
El sistema deberá mostrarle al usuario 3 opciones: <ul style="list-style-type: none"> <li>- Modificar nombre de perfil usuario.</li> <li>- Modificar avatar de perfil usuario.</li> <li>- Restaurar aplicación.</li> </ul>	
El usuario al interactuar con la opción de modificar nombre de usuario: el sistema mostrara el teclado en pantalla para que el usuario pueda realizar la modificación del nombre del perfil de usuario.	
El usuario al interactuar con la opción de modificar avatar de perfil de usuario: el sistema deberá mostrar el panel que contiene todos los avatares disponibles para la selección del usuario.	
El usuario al interactuar con la opción de restaurar aplicación: el sistema deberá eliminar todos los datos del usuario, incluyendo nombre, avatar, configuraciones de la aplicación.	
<b>Comentarios</b>	
Ninguno	

*Cuadro 3. Caso de uso: Ver perfil de usuario. Elaboración propia.*

- **Ver Instrucciones de uso:** El siguiente caso de uso muestra las opciones que debe mostrar el sistema al usuario al momento de que este último desee ver las instrucciones y/o el tutorial de uso del sistema.

<b>Caso de uso: CU-03 VER INSTRUCCIONES DE USO</b>	
<b>Actor: Usuario</b>	
<b>Descripción de los procesos</b>	<b>Alternativas</b>
El sistema deberá mostrarle al usuario 2 opciones: <ul style="list-style-type: none"> <li>- Mostrar Requisitos.</li> <li>- Mostrar tutorial de uso.</li> </ul>	
El usuario al interactuar con la opción de Requisitos: el sistema deberá mostrar una interfaz que muestre los requisitos para el uso de la aplicación	
El usuario al interactuar con la opción de mostrar tutorial de uso el sistema deberá mostrar una interfaz la cual muestre un tutorial para un uso correcto de la aplicación y de los diferentes códigos QR	
<b>Comentarios</b>	
Ninguno	

*Cuadro 4. Caso de uso: Ver instrucciones de uso. Elaboración propia*

## 8.2 Inicialización

### 8.2.1 Cronograma de actividades

En la figura 32 se observa el cronograma en el cual se estipulan las actividades a desarrollar teniendo en cuenta las fases de la metodología “Mobile-D”, esta metodología se divide en 5 fases las cuales cuentan con actividades y entregables específicos que guiaran el desarrollo a la meta de entregar un software funcional y óptimo.

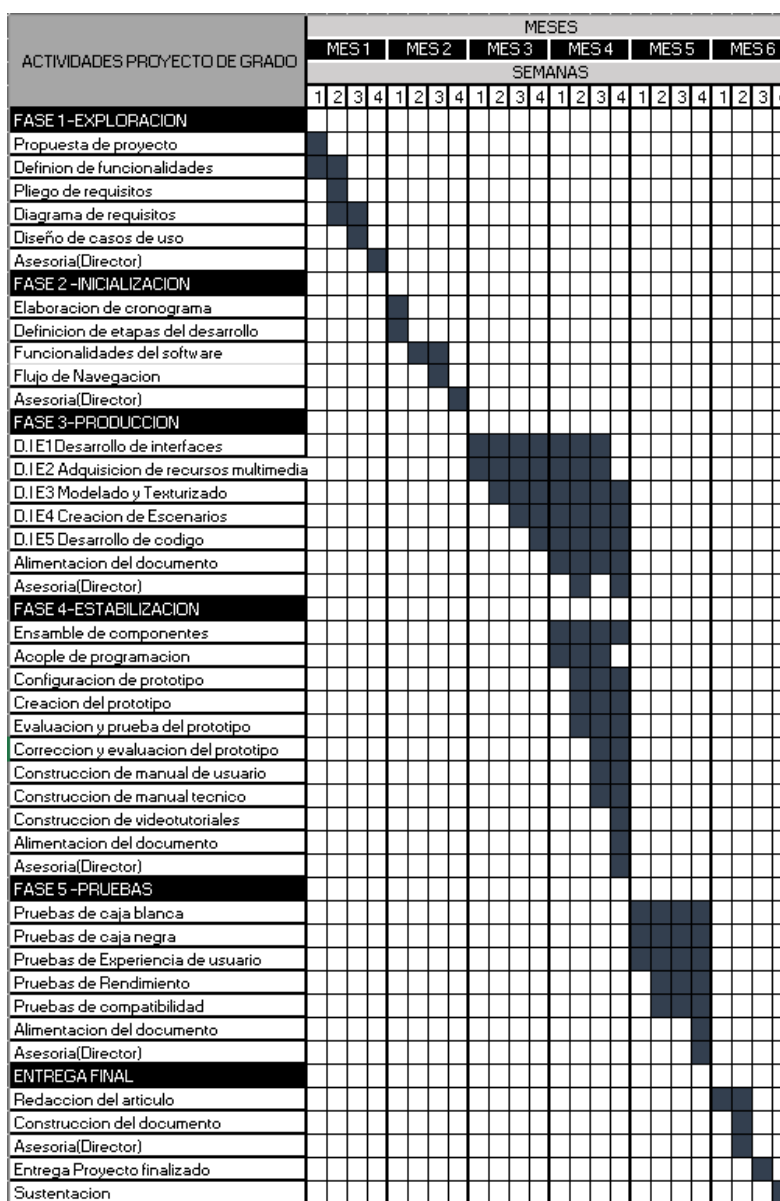


Figura 32 Cronograma de actividades. Elaboración propia.

El aplicativo que se desarrollara contendrá un total de 12 lecciones las cuales tendrán temáticas diferentes cada una de ellas, los temas que se trataran son: Los colores, los números, las vocales, el abecedario, el calendario, las horas, las partes del cuerpo, las comidas, las prendas de vestir, los medios de transporte, los animales y por ultimo las partes de la casa.

Así mismo se desarrollará un escenario de exámenes de lección, el cual servirá para evaluar los conocimientos adquirido por el usuario en cada una de las lecciones, la dinámica y mecánicas de este escenario serán las mismas para los 12 escenarios de exámenes, lo único que cambiara en cada uno de estos serán las preguntas, ya que estas mismas deben ir acorde con la temática de la lección.

De acuerdo con la metodología “Mobile-D”, en esta fase de inicialización se estipulan las etapas del desarrollo las cuales son:

- Etapa 1: Desarrollo de interfaces de usuario.
- Etapa 2: Adquisición de recursos multimedia.
- Etapa 3: Modelado, texturizado y animación de objetos en 3D.
- Etapa 4: Creación de escenarios.
- Etapa 5: Desarrollo de código.

### **8.2.2 Desarrollo de interfaces**

En esta fase de desarrollo, se realizan todas las interfaces de usuario que tendrá la aplicación, las cuales son:

- Interfaz de pantalla inicial
- Interfaz de creación de usuario
- Interfaz del menú principal
- Interfaz del menú de ayuda
- Interfaz del menú perfil de usuario
- Interfaz de escenarios de lecciones y exámenes
- Interfaz de tutoriales de uso
- Interfaz de mensajes informativos y de advertencia.

A continuación, en la figura 33 se observa el Wireframe, el cual representa visualmente, de una forma muy sencilla y esquemática la estructura de la aplicación.

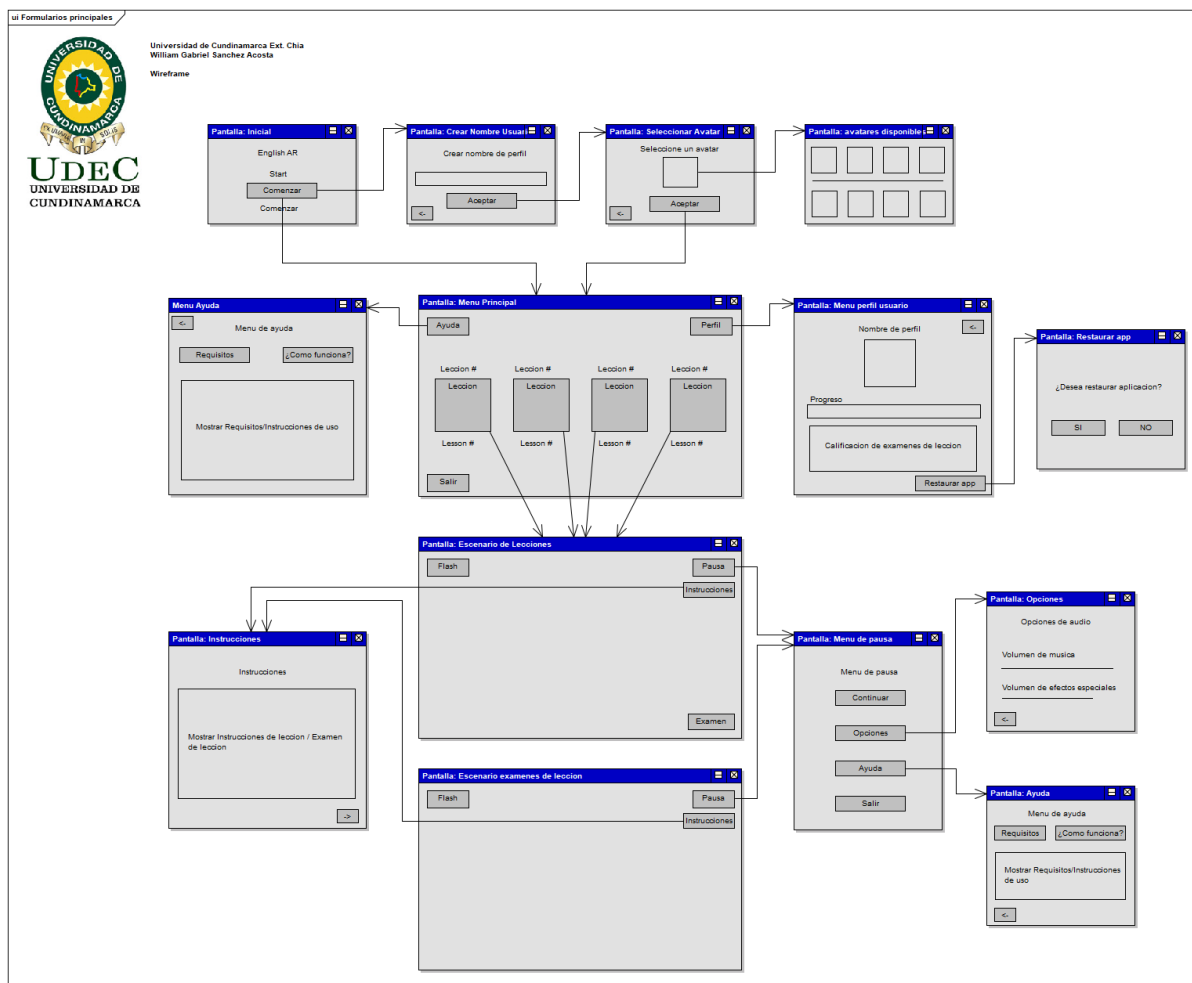


Figura 33. Wireframe de la aplicación. Elaboración propia.

### 8.2.3 Adquisición de recursos multimedia

En esta etapa de desarrollo, se realiza la búsqueda y adquisición de todos los recursos multimedia que son necesarios para la elaboración de cada uno de los escenarios de la aplicación, dichos recursos multimedia son archivos de imágenes y audio. Para los archivos de imágenes e iconos, se utilizaron los bancos de imagen Pixabay.com y Flaticon.com, y para los archivos de audio se utilizó el banco de audios sin copyright llamado FreePD.com. Hay que resaltar que todos los archivos multimedia son de uso libre y sin copyright.

### 8.2.4 Modelado y texturizado de los objetos en 3D

En esta etapa de desarrollo, se elaboran todos los modelos 3D necesarios para cada uno de los escenarios que componen a la aplicación. Se realiza el modelado de los objetos 3D y luego pasan por el proceso de texturizado para finalmente ser exportados al software Unity.

## 8.2.5 Creación de escenarios

En esta etapa de desarrollo, una vez teniendo la interfaz de usuario, los modelos en 3D y los archivos multimedia de imagen y audio, se procede a seleccionar los componentes anteriormente nombrados necesarios para cada uno de los escenarios que deben ir en la aplicación.

## 8.2.6 Desarrollo de código

En esta etapa de desarrollo, se elabora toda la programación necesaria para cada uno de los elementos de la aplicación. Se debe desarrollar la programación de:

- Componentes de interfaces de usuario (Botones, Sliders, Campos de texto)
- Componentes de los escenarios de lecciones (modelos en 3D, elementos de audio, elementos de imagen, animaciones).
- Escenarios de exámenes de las lecciones (modelos en 3D, elementos de audio, animaciones, elementos de imagen).
- Otros elementos que sean vitales para el correcto funcionamiento del aplicativo.

## 8.3 Producción

### 8.3.1 Desarrollo iterativo: Etapa 1. Desarrollo de interfaces

Se empezó el desarrollo de la primera etapa, elaborando el prototipo de cada una de las interfaces que irían en la aplicación, comenzando con la creación de la interfaz de la pantalla inicial, puesto que, es la primera interfaz que se verá al abrir el aplicativo. En la figura 34 se observa resultado del primer prototipo.

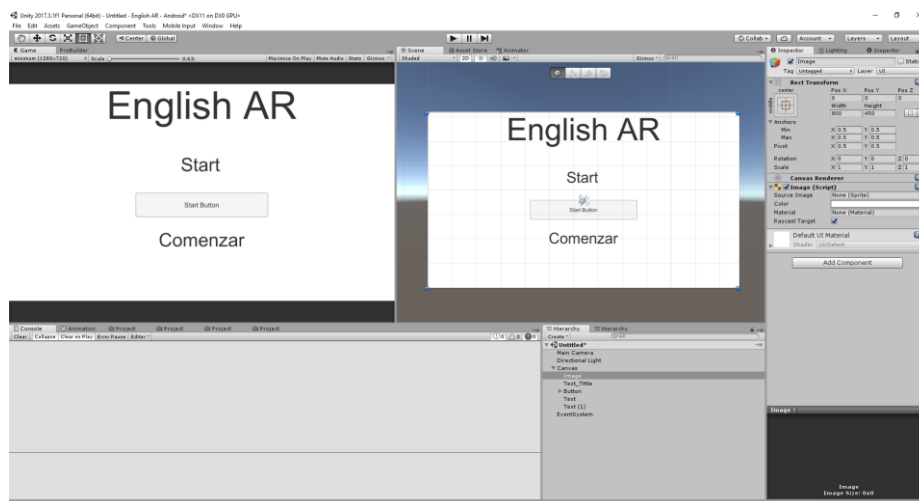


Figura 34. Captura de pantalla de prototipo de interfaz de pantalla inicial. Elaboración propia.

Posteriormente, y teniendo en cuenta que el usuario al acceder por primera vez a la aplicación debe crear un perfil de usuario, se tuvo que desarrollar una interfaz secuencial, en la cual, el usuario primeramente debe crear un nombre de perfil y luego, debe asignar un avatar a dicho perfil, esto con la finalidad de no saturar la pantalla con demasiada información, ya que, la aplicación al ser desarrollada para dispositivos móviles, estos en su gran mayoría no cuentan con una pantalla lo suficientemente grande para mostrar una gran cantidad de información al usuario. Para entenderlo mejor, basta con observar la figura 35, la cual muestra el desarrollo de la interfaz secuencial que es la encargada de permitir la creación de un perfil al usuario.



Figura 35. Proceso secuencial de la creación de un perfil de usuario. Elaboración propia.

Una vez desarrollada la interfaz de la pantalla inicial, se empezó a elaborar la siguiente interfaz, que es la del menú principal. Esta interfaz cuenta con una serie de botones que son encargados de abrir cada uno de los escenarios de las lecciones de enseñanza. Así mismo, como son 12 escenarios que se crearan, también son 12 botones los que se deben colocarse en pantalla, al ser una cantidad considerablemente alta de botones para una sola pantalla, se optó por dividir esta cantidad en tres

interfaces diferentes e implementar dos botones de navegación, tal como se aprecia en la siguiente figura.

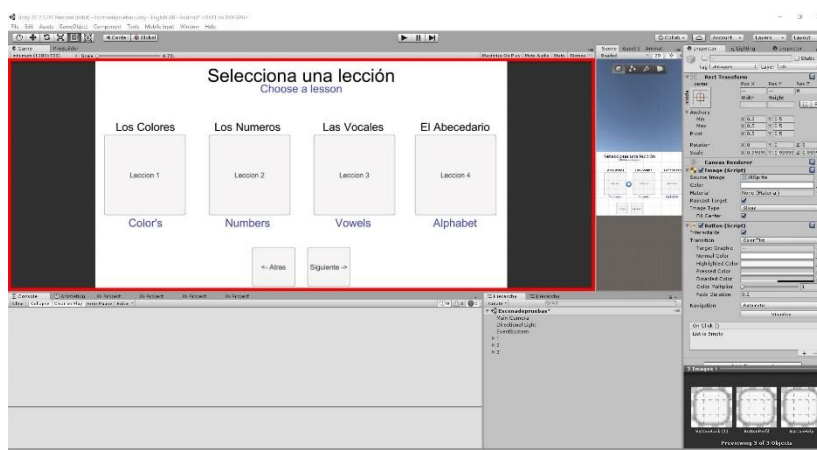


Figura 36. Desarrollo de interfaz del menú principal. Elaboración propia.

Siendo los botones “atrás” y “Siguiente” los encargados de permitir al usuario una navegación entre las interfaces que contenían las 8 lecciones restantes.

Luego de ello, se añadieron los botones que se consideraron importantes incluir en el menú principal, como lo son el botón de ayuda el cual proporcionara información relacionada a los requisitos de la aplicación y a las instrucciones de uso, el botón de mostrar el perfil del usuario cuyo propósito es el de mostrar información del perfil y el progreso del usuario en la aplicación, y, por último, el botón de salir de la aplicación el cual se encarga del cierre de la misma. Tal como se aprecia en la figura 37.

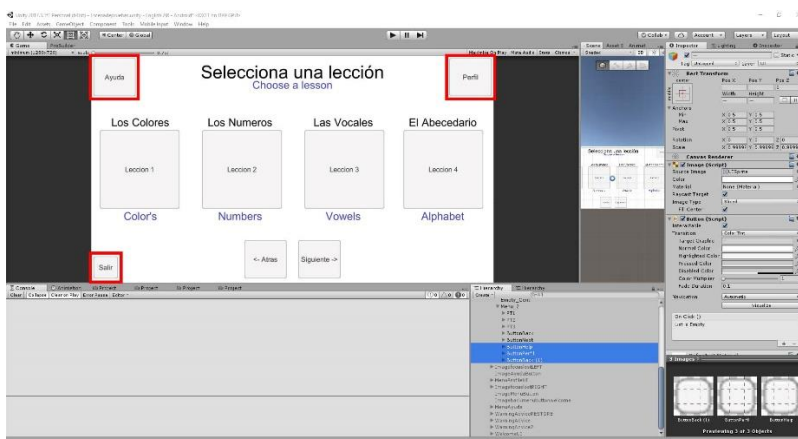


Figura 37 adición de los botones ayuda, perfil y salir en el menú principal. Elaboración propia.

Después de dar fin a la elaboración de la interfaz del menú principal, se empezó con el desarrollo de la interfaz del menú de perfil del usuario, en este apartado se consideró pertinente mostrar el nombre y el avatar del perfil elegido por el usuario e incluir una barra de progreso que ilustra los avances del usuario en la aplicación y a su vez, incluir un apartado en el cual se muestran los puntajes obtenidos en cada examen de lección. En la figura 38, se puede observar lo anteriormente descrito.

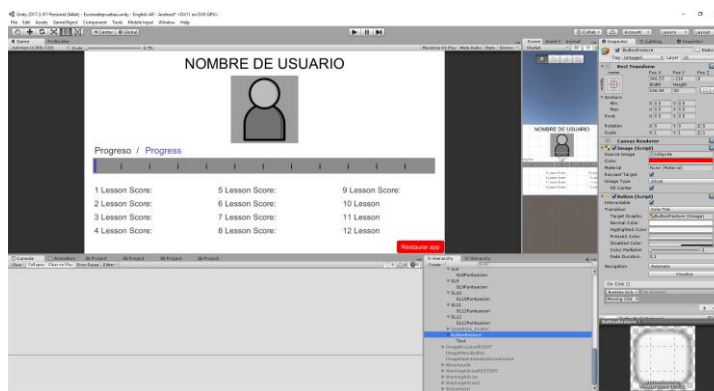


Figura 38 Captura de pantalla de interfaz de menú perfil de usuario. Elaboración propia.

Antes de finalizar con la elaboración de la interfaz del menú de perfil, se agregó un botón llamado “restaurar app” cuya función es restaurar la aplicación a su estado por defecto, dicha acción se encarga de borrar todas las configuraciones de usuario y progresos en la aplicación.

Luego de haber finalizado completamente con la interfaz del menú de perfil, se empezó con el desarrollo de la interfaz del menú ayuda, la cual debía ser capaz de mostrar toda la información que fuese instructiva para el usuario. Dentro de ese menú se agregaron dos botones más: “Requisitos” y “Como funciona”, como se observa en la figura 39.

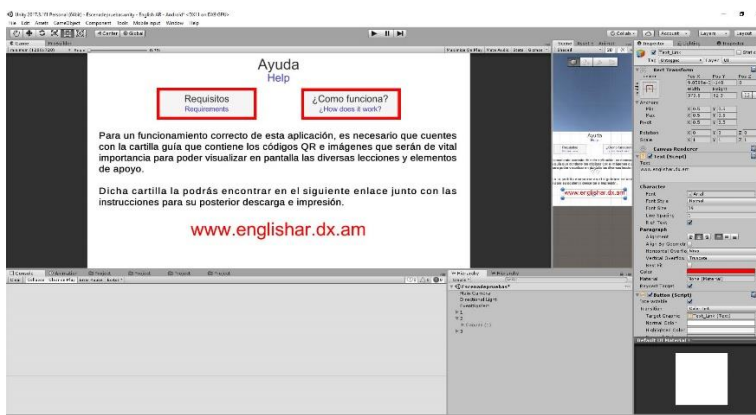
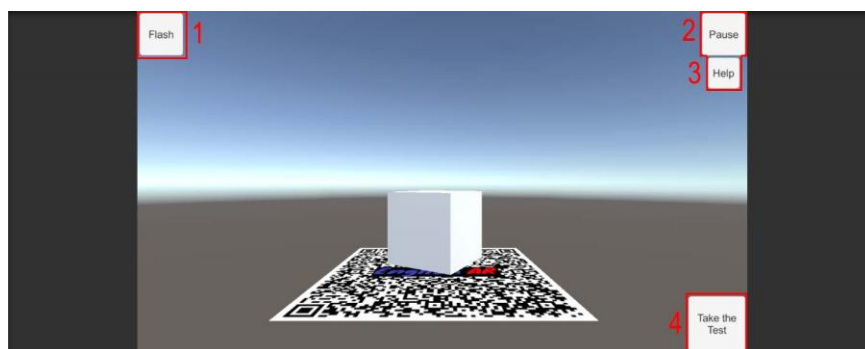


Figura 39 Desarrollo de interfaz de menú ayuda. Elaboración propia.

Estos botones deben proporcionar toda la información con respecto a los requisitos necesarios para el correcto funcionamiento de la aplicación, así como también un manual con las instrucciones de uso.

Una vez completado el desarrollo de todas las interfaces que deben ir tanto en la pantalla inicial como en el menú principal, se dio inicio a la elaboración de una de las interfaces más esenciales de la aplicación la cual es: la interfaz de los escenarios de lecciones y los escenarios de exámenes. Al ser 12 escenarios de lecciones y 12 escenarios más de exámenes, se tuvo que desarrollar una interfaz que fuera genérica y que se pudiera incluir en todos los 24 escenarios mencionados anteriormente.



*Figura 40 Interfaz de escenario de lecciones y exámenes. Elaboración propia.*

Para el desarrollo de esta interfaz, se añadieron los siguientes botones:

1. **Botón de estado de flash:** este botón debe cambiar el estado del flash del dispositivo, este debe ser capaz de encenderlo o pagarlo. Se implementó este botón con la finalidad de facilitarle a la cámara del dispositivo, el reconocimiento de los códigos QR en condiciones de poca luz.
2. **Botón de pausa:** este botón es el encargado de detener todos los procesos de ejecución que hay en el escenario, las operaciones que se suspenden momentáneamente son: la cámara deja de reconocer los códigos QR, se silencian los audios que hay en el escenario y todas las animaciones que estén ejecutándose pasaran a un estado de pausa. Posterior a eso, se debe mostrar en pantalla la interfaz del menú de pausa, la cual se puede observar en la figura 40.

3. **Botón de ayuda de lección y examen:** se agregó este botón para proveer una interfaz que muestre información de ayuda e instrucciones de la lección o el examen según sea el caso. Dicha interfaz se puede ver en la figura 42 de la página 76.
4. **Botón para presentar examen:** se añadió este botón el cual permite dirigir al usuario al escenario de examen de lección. El haber puesto dicho botón en la interfaz de lecciones y no en el menú principal, fue con la finalidad de que el usuario primero debe ver el tema que se está tratando en la lección antes de presentar el examen. Cabe resaltar que este botón estará presente únicamente en la interfaz de los escenarios de lección.

Luego de dar por terminado el desarrollo de la interfaz de los escenarios de lecciones y exámenes, se elaboró la interfaz del menú de pausa, la cual debe contener una serie de botones que realizan diferentes acciones cada uno. Estos botones son los encargados de reanudar la lección, mostrar los ajustes de audio de la lección, mostrar nuevamente las instrucciones de uso y requisitos y, por último, regresar al usuario al menú principal. En la siguiente figura se muestra lo anteriormente dicho.



Figura 41 Menú de pausa con las interfaces de submenús de opciones y ayuda. Elaboración propia.

Para finalizar el desarrollo de los prototipos de las interfaces, se realizaron las últimas interfaces que eran las que contenían los mensajes instructivos de las lecciones y exámenes, así como también, los mensajes de advertencia necesarios para cada acción no permitida que cometiese el usuario.

- **Mensajes informativos e instructivos de lecciones y exámenes de lecciones:** al inicio de cada lección y examen debe aparecer una breve interfaz que le indique al usuario lo que

aprenderá en la lección, las instrucciones de uso y los códigos QR necesarios para cada lección o examen, y todo esto de manera secuencial. El resultado de la elaboración se puede observar en la siguiente figura.

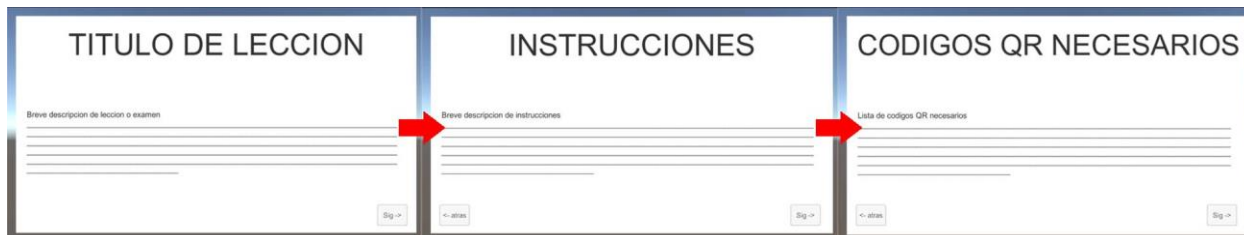


Figura 42 Secuencia de menú informativo e instructivo de lecciones y exámenes. Elaboración propia.

- **Mensajes de advertencia:** estos mensajes deben aparecer cuando el usuario comete una acción no permitida o que pueda causar un cambio significativo en el uso del aplicativo como puede ser: no escribir ningún nombre de usuario o escribir un nombre que tenga menos de 3 caracteres, un mensaje advirtiendo al usuario que se restaurara toda la información o un mensaje de alerta de cierre de la aplicación. Tal como se muestra en la figura 43.



Figura 43 Diversos mensajes de advertencia en la aplicación. Elaboración propia.

Una vez teniendo todos los prototipos hechos, se dio inicio a la fase final de la etapa 1, la cual consiste en añadir todos los detalles visuales a cada una de las interfaces. Se debe agregar los respectivos iconos a los botones, cambiar la fuente de letra y ajustar los tamaños de la misma y, por último, agregar una paleta de colores a todos los elementos de interfaz.

Para adquirir los iconos se tuvo que recurrir una página llamada Flaticon, la cual aloja una gran cantidad de iconos creados para ser usados en interfaces de aplicaciones, se adquirió una licencia



Cabe destacar que todos los recursos adquiridos para el diseño y desarrollo de las interfaces son libres de derechos de autor.

Se finalizó el desarrollo de interfaces, luego de reunir todos los recursos visuales y asignarlos a todos los elementos de los prototipos previamente creados. A continuación, en la figura 47 se puede observar una recopilación de las principales interfaces terminadas.



Figura 46 Recopilación del resultado final de las principales interfaces de usuario. Elaboración propia.

### 8.3.2 Desarrollo iterativo: Etapa 2. Adquisición de recursos multimedia

En la segunda etapa se realizó la búsqueda y adquisición de todos los recursos multimedia que se requerían para el desarrollo de los escenarios, tales recursos son: archivos de imagen y archivos de audio.

Para los escenarios de las lecciones se requerían imágenes alusivas al tema que se pretendía enseñar, se debían buscar imágenes de las diferentes temáticas cuales iban a componer una de las

partes más importantes de la aplicación. Debido a que era una cantidad considerable de imágenes las que se iban a utilizar, estas debían ser extraídas de bancos de imágenes online.

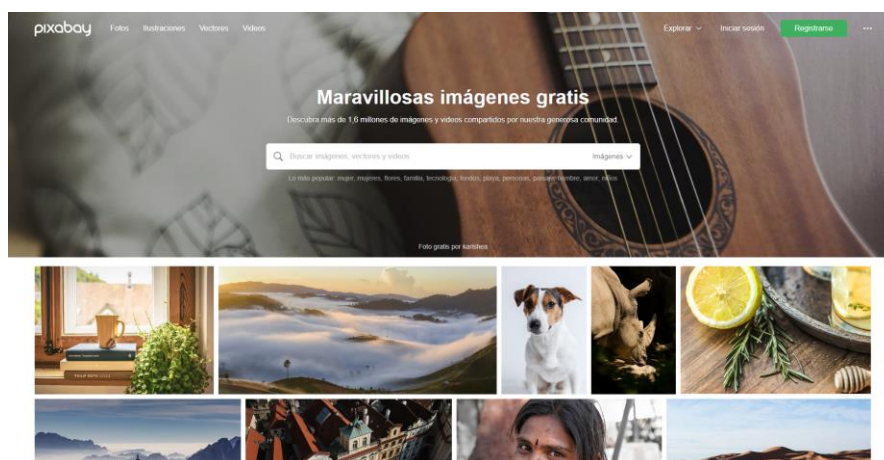


Figura 47 Banco de imágenes Pixabay.com.<sup>21</sup>

Dado que la aplicación debía subirse a la tienda de aplicaciones Google Play, esta debía contar con recursos que fueran 100% gratuitos y sin copyright, se tuvo que realizar una breve consulta acerca de los bancos de imágenes que ofrecen ilustraciones gratuitas y libres de derechos para así evitar infringir las normas de derechos de autor, puesto que, en internet se encuentran muchos bancos de imágenes que aparentemente son gratuitas cuando verdaderamente no lo son. Se eligió el sitio web llamado Pixabay.com como banco de imágenes principal para descargar todas las imágenes necesarias para la aplicación.

Para los archivos de audio, especialmente la música de fondo que debían llevar los diferentes escenarios, se tuvo que realizar la misma dinámica de las imágenes, es decir, que se tuvo que realizar una búsqueda de sitios web que contaran con música libre de copyright y que se pudiera usar tanto para fines comerciales como no comerciales. En dicha búsqueda se encontró el sitio web llamado FreePD.com, el cual aloja una biblioteca de audio de diferentes géneros y proyectos realizados por diferentes artistas que permiten el uso de su música sin ningún tipo de beneficio a cambio.

Adicional a la música que debía agregarse a los escenarios, se tuvo que realizar una búsqueda más para los audios de pronunciación en inglés, estos elementos de audio son de vital importancia

<sup>21</sup> Pixabay.com. (2019). Banco de imágenes libres de derechos de autor Pixabay.com. [Figura]. Recuperado de: <https://pixabay.com/>

para las lecciones de enseñanza ya que, como se mencionó anteriormente, su función principal es la de enseñar a oír y pronunciar las diversas palabras del vocabulario en inglés que se pretenden mostrar en la aplicación.

En el mercado, existen diversos programas que hacen síntesis de habla, algunos de ellos tienen un costo muy elevado, otros son más económicos pero la calidad de audio es muy baja, entonces, se tuvo que buscar un software que tuviera un balance entre calidad/costo. Uno de los más asequibles que contara con dicho balance además de contar con licencia de uso comercial fue el servicio web de Amazon llamado Polly. A continuación, en la siguiente figura se muestra un cuadro el cual contiene la comparación entre los diferentes software y servicios web de síntesis de habla disponibles en el mercado, para dar como evidencia las bondades que ofrece este servicio.

Software/Servicio de síntesis de habla	Características	Tipo	Licencia	Costo	Página oficial de precios
Voice Reader 15	Aplicación para la conversión de texto a voz: ofrece la posibilidad de convertir textos de documentos Word, correos electrónicos y pdf en transmisiones de audio, además de contar con 67 tipos de voces y 45 idiomas diferentes.	Software para PC con Windows vista/7/8/10	Comercial	499€ al software base + 49€ por cada voz/idioma	<a href="https://www.linguatic.de/en/shop-tts/">https://www.linguatic.de/en/shop-tts/</a>
Capti Voice	Aplicación de síntesis de habla ideal para el sector educativo, muy usada en diversas escuelas y universidades de todo el mundo, cuenta con soporte para más de 20 idiomas distintos, que cumplen con diversas funciones como por ejemplo: mejorar el vocabulario, narrar gran variedad de contenido como libros electrónicos, artículos y páginas web.	Software para PC	Personal/ Para uso educativo	\$19,99 * usuario o \$3,999 la licencia para instituciones educativas	<a href="https://www.captivoice.com/capti-site/public/entry-get_capti_subscription#starter">https://www.captivoice.com/capti-site/public/entry-get_capti_subscription#starter</a>
Natural Reader	Servicio web online de síntesis de habla, este servicio permite convertir textos escritos como documentos de Word y PDF, libros electrónicos y páginas web de una manera similar al habla de un humano. Cuenta con un total de 61 voces y 18 idiomas diferentes.	Servicio web Online	Personal/Comercial	\$49 Mensual por usuario o \$79 Mensual, para un equipo de 4 usuarios	<a href="https://www.naturalreaders.com/commercial.html">https://www.naturalreaders.com/commercial.html</a>
iSpeech	Servicio web online de síntesis de habla, el cual cuenta con 27 lenguajes y diferentes voces tanto masculinas como femeninas disponibles, capaces de transformar diferentes textos, libros electrónicos o páginas web en audios con un habla poco similar a la de un humano	Servicio Web Online	Comercial	\$100 dolares por cada 900 palabras, \$500 dolares por cada 10,000 palabras o \$1500 dolares por cada 50,000 palabras	<a href="https://www.ispeech.org/create_text_to_speech_audio">https://www.ispeech.org/create_text_to_speech_audio</a>
Amazon Web Service Polly	Amazon Polly es un servicio de síntesis de habla web online el cual permite la conversión de texto a voz en una diversidad de idiomas, siendo un total de 24 idiomas y de 1 a 5 voces disponibles por cada idioma, lo que caracteriza a este servicio es que utiliza diferentes tecnologías avanzadas de aprendizaje profundo, lo cual deja como resultado una calidad de habla muy natural simulando la voz, pronunciación y velocidad de habla de un humano.	Servicio Web Online	Comercial/Personal	el primer año gratis + 5 millones de caracteres mensuales y \$4 dolares mensuales (1 millón de caracteres al mes) después del año gratuito.	<a href="https://aws.amazon.com/es/polly/pricing/">https://aws.amazon.com/es/polly/pricing/</a>

Figura 48 Cuadro comparativo de los diferentes software y servicios web de síntesis de habla disponibles en el mercado. Elaboración propia.

Para la elaboración de los audios de pronunciación en inglés, se utilizó dicho servicio para convertir las palabras de texto en audios de voz, este servicio tiene varias opciones para modificar el audio tales como: la pronunciación, el volumen, la velocidad de habla, entre otras. Para este caso, se requería que la pronunciación fuera fácil de entender, por ello se utilizaron las etiquetas SSML del servicio web de Amazon Polly para hacer los debidos ajustes al audio de pronunciación. La sigla SSML es la abreviación de Speech Synthesis Markup Language, que en español se traduce como Lenguaje de Marcado de Síntesis de Voz, El Consorcio World Wide Web (W3C, 2010) afirma que:

Está diseñado para proporcionar un rico lenguaje de marcado basado en XML para ayudar a la generación de voz sintética en la Web y otras aplicaciones. La función esencial del lenguaje de marcado es proporcionar a los autores de contenido sintetizable una forma estándar de controlar aspectos de la salida del habla, como pronunciación, volumen, tono, velocidad, etc.

Para entender mejor este proceso se anexa una captura de pantalla del proceso realizado:

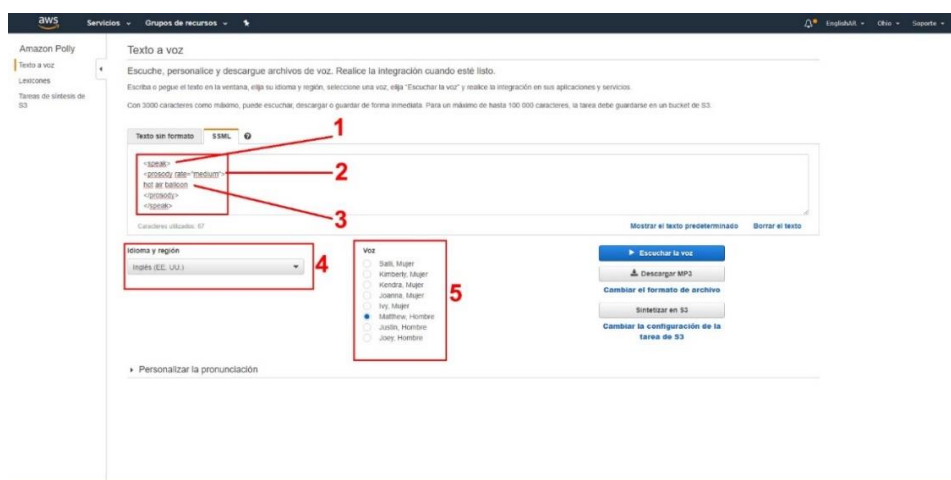


Figura 49 Captura de pantalla de herramienta de síntesis de habla Polly.<sup>22</sup>

En la figura anterior se observa lo siguiente:

- 1. Etiqueta < speak >:** esta es la etiqueta base de todo el texto que se quiere modificar a través de las etiquetas SSML de Amazon Polly. Todo el texto mejorado con SSML debe ir entre las etiquetas < speak > y < /speak >.

<sup>22</sup> Amazon Web Services (AWS) Polly. (2019). Herramienta de síntesis de habla llamado Polly. [Figura]. Recuperado de: <https://aws.amazon.com/es/polly/>

2. **Etiqueta <prosody>**: esta etiqueta en general tiene como función controlar el volumen, la velocidad y el tono de voz. En este caso se utilizó el atributo “rate” para ajustar la velocidad de la voz y dejarla en medio.
3. **Texto a convertir**: este es el texto que se quiere convertir a voz.
4. **Idioma**: se seleccionó el idioma de voz inglés de estados unidos (EE.UU).
5. **Voz**: Amazon brinda una lista de las voces disponibles de acuerdo al idioma seleccionado, en este caso se utilizaron voces tanto masculinas como femeninas para la pronunciación de todas las palabras que se incluirán en cada una de las lecciones.

### 8.3.3 Desarrollo iterativo: Etapa 3. Modelado y texturizado de modelos 3D

Una vez finalizada la etapa de adquisición de recursos multimedia, se dio inicio a la etapa del modelado y texturizado de los objetos en 3D necesarios para cada escenario. Para la realización de los modelos en 3D se utilizó la herramienta Blender gracias a que esta posee las dos características mencionadas anteriormente y adicionalmente permite exportar los modelos 3D en formatos de archivo que son reconocidos y fáciles de adaptar en Unity, para así, evitar el uso de software adicional.

Se elaboraron modelos en 3D que fueran una representación de cada tema de enseñanza, cada objeto en 3D que se creara debía ser significativo de la materia que se quería impartir.

La elaboración de cada uno de los modelos 3D se realizó a partir de la modificación de una primitiva geométrica como por ejemplo un cubo, utilizando las herramientas básicas de Blender, las cuales son: rotar, escalar, mover y extruir para dar la forma correspondiente a cada uno de los objetos que se van a crear. Así mismo, también se realizaron las texturas correspondientes para cada modelo usando el software GIMP.

- **Modelado y Texturizado: Lección Los Colores**

Para la lección de los colores, se diseñó y elaboró en Blender un globo de aire caliente el cual, al ser programado debía cambiar de su color, esto con la finalidad de representar los principales colores y como se pronuncian en inglés. A continuación, en la figura 53 una muestra del modelado y texturizado en Blender.

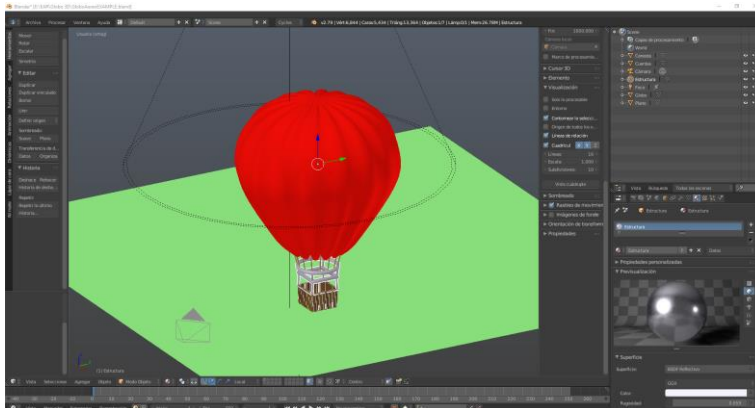


Figura 50 Muestra del proceso de modelado en Blender

Adicionalmente, en Blender se asignaron los materiales al globo, estos materiales debían ser los colores que iban a representarse en la lección, es por esto que se asignaron al objeto 3D para así poder exportar los materiales junto con el modelo a Unity.

- **Modelado y Texturizado: Lección Las vocales, El Alfabeto y Los Números**

Para las 3 lecciones se diseñaron todas las letras del abecedario, así como también los números en 3 dimensiones, gracias a una herramienta de Blender que permite crear texto en 3D. en la siguiente figura se da una muestra de lo anteriormente dicho.

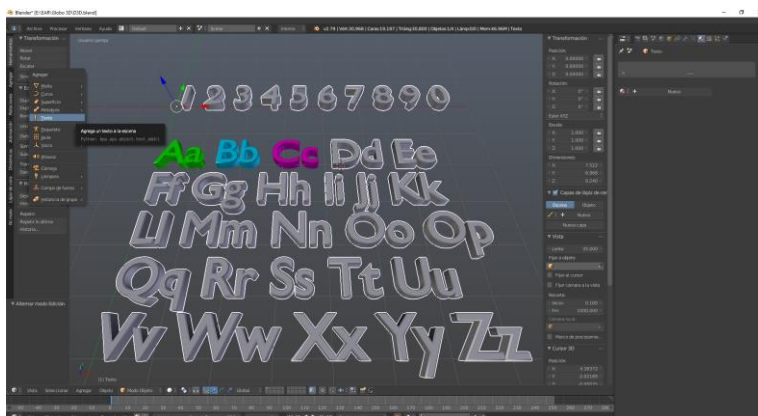


Figura 51 Elaboración de las letras y los números en 3D en Blender. Elaboración propia.

Se realizó el proceso de crear las letras y números en 3D, luego texturizar cada elemento y finalmente exportar los modelos a Unity como archivos independientes.

- **Modelado y Texturizado: Lección El Calendario**

Para elaborar el calendario en 3D primero se creó la textura base teniendo como referencia la foto de un calendario físico. A partir de esa foto se realizó una imagen similar que tuviese los días

de la semana y a su vez los días del mes, todo esto se hizo en el software GIMP tal como se aprecia en la siguiente figura.

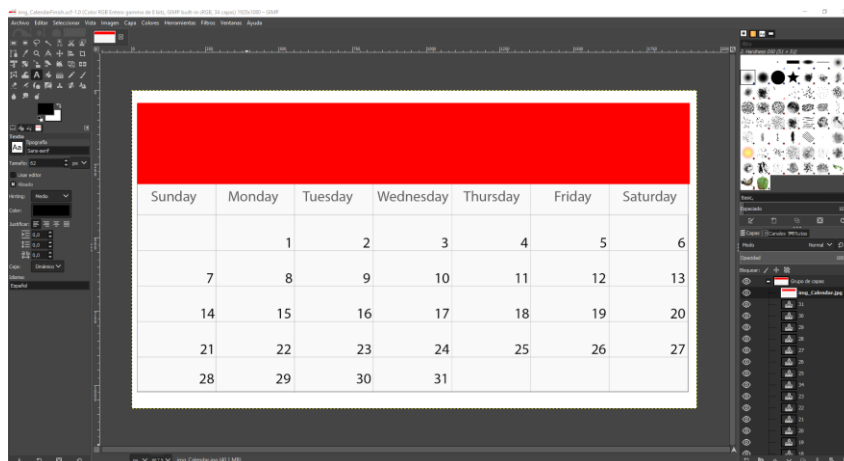


Figura 52 Textura base del calendario 3D. Elaboración propia.

Como se puede observar en la anterior figura, se elaboró el diseño base del calendario, pero no se agregaron los meses puesto que, estos se van a añadir en la etapa de creación de escenarios, ya que forman parte de una de las mecánicas de la lección.

Una vez teniendo la textura del calendario, en Blender se creó un cubo en 3D al cual se le iba a aplicar dicha textura, mediante la técnica de despliegue UV de Blender. Este proceso de despliegue UV “es una técnica usada en cualquier programa de diseño 3D para texturizar un modelo. Consiste en desplegar toda la superficie del modelo en un plano bidimensional donde se pueda añadir una textura plana” (D’Elia, 2018). Este proceso se muestra en la figura 56.

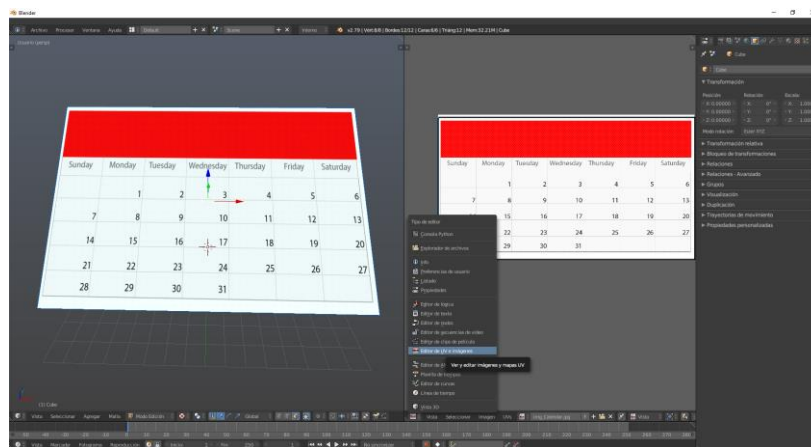


Figura 53 Proceso de texturizado usando la herramienta de mapeo y despliegue UV en Blender. Elaboración propia.

- **Modelado y Texturizado: Lección Las Horas**

Para la lección de las horas, se requería modelar un reloj de estilo digital con el cual se pueda interactuar simulando un reloj físico real. La elaboración de este modelo se hizo por partes, ya que primero se creó la estructura del reloj y luego los botones por separado, ya que estos deben cumplir una función específica dentro de la lección de las horas. Posteriormente, se agregaron los materiales correspondientes a cada malla y, para los números que iban a ilustrar las horas se utilizaron los números en 3D que previamente se habían hecho. Se anexa una captura de pantalla que muestra el resultado de la elaboración del reloj digital.



Figura 54 Reloj digital hecho en Blender. Elaboración propia.

- **Modelado y Texturizado: Lección Partes del Cuerpo**

Para la creación del modelo en 3D del cuerpo humano, se utilizó un software llamado Make Human, el cual permite crear una anatomía del cuerpo humano en base a diferentes parámetros que se pueden ajustar dentro del propio software. Tal como se observa en la figura 58.

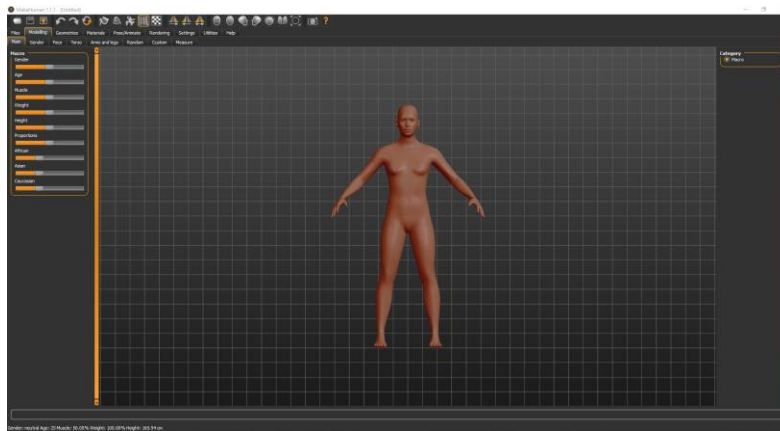


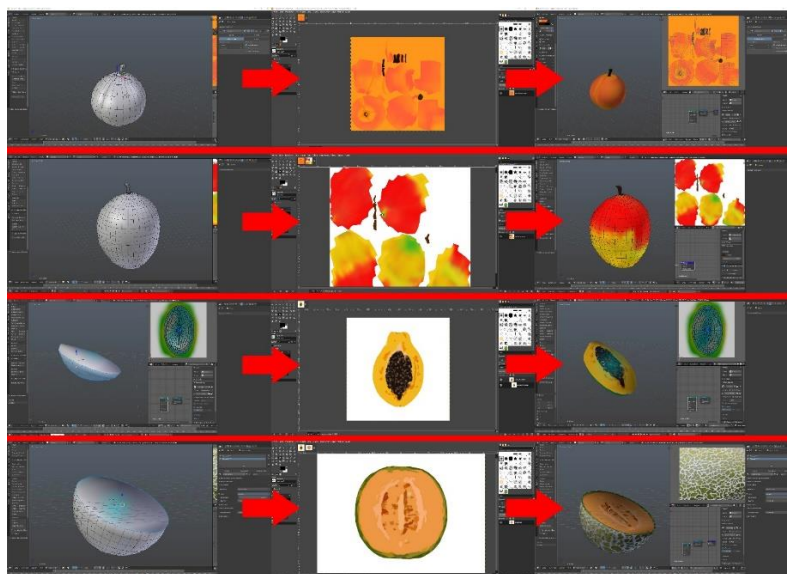
Figura 55. Captura de pantalla del software Make Human. Elaboración propia.

Se creó un cuerpo masculino que no tuviese las partes íntimas y que no diera indicios sexuales por dos razones: la primera y más importante es que la aplicación está dirigida a niños menores de 13 años, y la segunda es que al momento de subir la aplicación a la tienda Google Play, van presentarse problemas por tener contenido inapropiado si la aplicación está clasificada para uso de menores de edad.

Luego de ajustar diferentes parámetros dentro del software tales como: la edad, el género, la estatura y demás, se exporto el modelo del cuerpo humano a Unity para agregar los detalles finales de la lección.

- **Modelado y Texturizado: Lección Las Comidas**

Se realizó el modelado de diferentes tipos de comidas, tales como: frutas, verduras comidas rápidas y bebidas. Nuevamente se utilizó el software GIMP para realizar las respectivas texturas para luego ser asignadas a cada modelo usando las herramientas de Blender. En la figura 60 se muestra un resumen de la elaboración de algunos de estos modelos desarrollados.



*Figura 56 Proceso de modelado, creación de texturas y texturizado de comidas en 3D. Elaboración propia.*

En la figura anterior se detalla la elaboración de los modelos 3D en Blender, luego la elaboración de las texturas en GIMP y, por último, el proceso de texturizado hecho nuevamente en Blender.

- **Modelado y Texturizado: Lección Prendas de Vestir**

Para el desarrollo de los modelos en 3D de la ropa se utilizó nuevamente el software Make Human, el cual además de brindar la posibilidad de crear cuerpos humanos, tiene una herramienta adicional la cual permite crear ropa para los mismos.

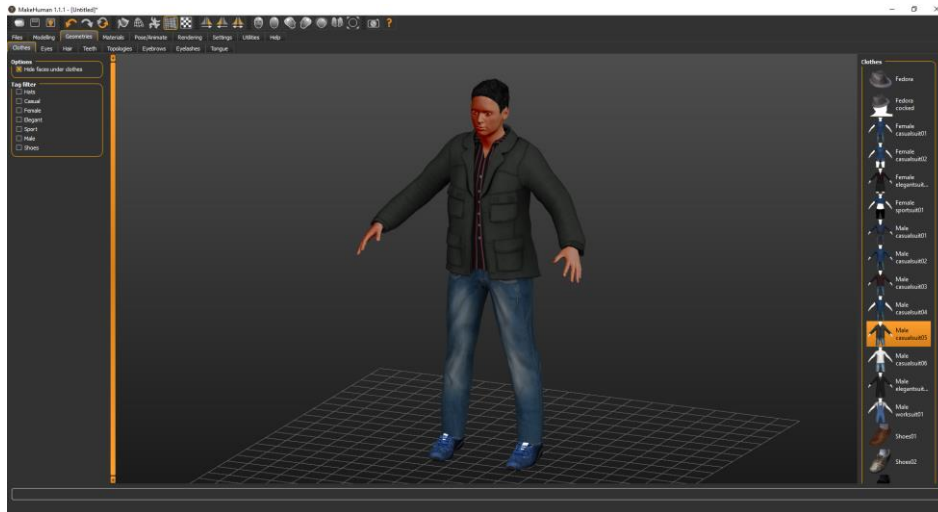


Figura 57 Prendas de vestir disponibles en el software Make Human. Elaboración propia.

Había prendas de vestir que no estaban en el software como, por ejemplo: guantes, medias, vestidos y ropa íntima para mujer entre otros, y es por esto que se tuvo que modelar estas prendas de vestir adicionales usando como base los modelos de las partes del cuerpo humano. Para entenderlo mejor se anexa una captura de pantalla de este proceso:

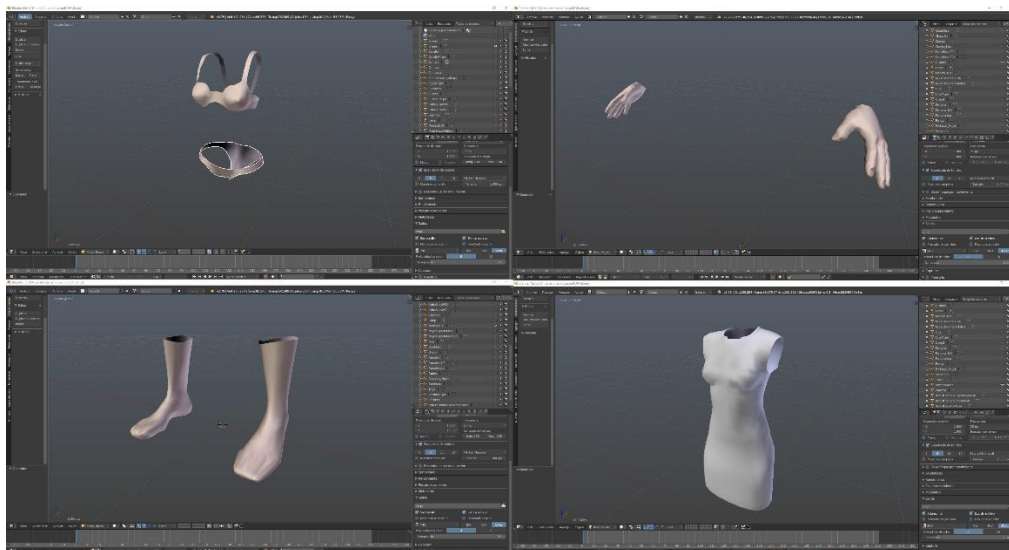


Figura 58 Modelado de prendas de vestir adicionales hechas en Blender. Elaboración propia.

- **Modelado y Texturizado: Lección Medios de Transporte**

Se realizó el modelado y texturizado de los principales medios de transporte aéreos, terrestres y acuáticos tales como: helicóptero, bote, bicicleta, tren, globo dirigible entre otros.

- **Modelado y Texturizado: Lección Los Animales**

Para el desarrollo de los modelos que irían en la lección de animales, se modelaron los principales animales y los más conocidos tales como: perro, gato, león, caballo, vaca entre otros. El proceso fue el mismo que para todos los demás modelos realizados hasta el momento, se realizó el modelado del animal, luego se elaboró la textura en el software GIMP y, por último, usando la herramienta de mapeo y despliegue UV de Blender se terminó el proceso de texturizado.

- **Modelado y Texturizado: Palabras del vocabulario en 3D**

En Blender, con la herramienta ya antes mencionada de creación de texto en 3D, se elaboraron todas las palabras del vocabulario que pertenecen cada uno de los modelos en 3D hechos anteriormente, esto se realizó con la finalidad mostrar la escritura de las palabras en inglés en cada una de las lecciones. A continuación, se muestra el resultado del proceso anteriormente descrito.

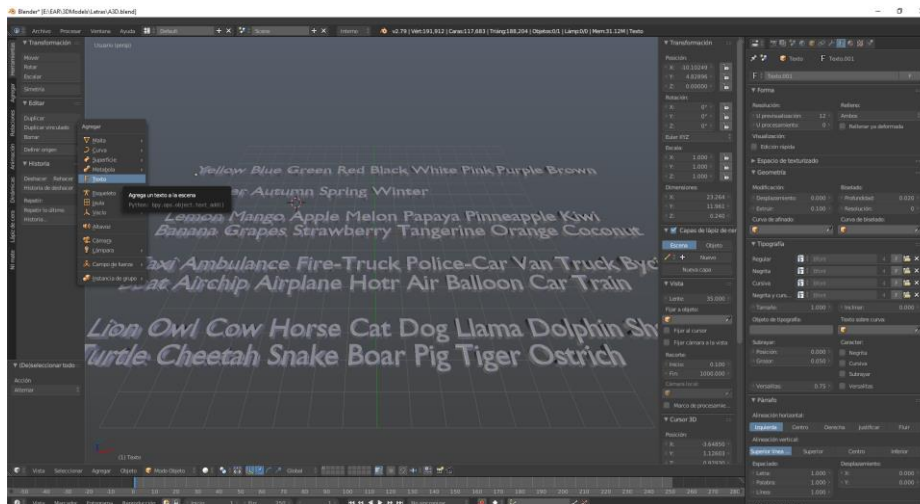


Figura 59 Creación de palabras del vocabulario inglés en 3D hechas en Blender. Elaboración propia.

Luego de esto, se exportó cada palabra como archivo independiente a Unity para ser asignada a cada uno de las lecciones y modelos correspondientes.

- **Modelado y Texturizado: La Casa**

Para finalizar la etapa del modelado y texturizado de los objetos 3D necesarios para la construcción de los escenarios, se elaboró el modelo en Blender de la casa, realizando el modelado tanto del interior como del exterior. En la figura 66 se muestra el proceso anteriormente dicho.

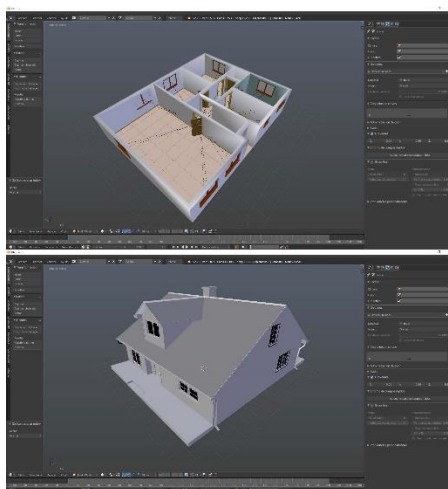


Figura 60 Modelado de interior y exterior de una casa en Blender. Elaboración propia.

### 8.3.4 Desarrollo Iterativo: Etapa 4. Creación de escenarios

En esta etapa de desarrollo, se elaboran todos los escenarios de enseñanza haciendo uso de los recursos previamente hechos y adquiridos.

Lo primero que se hizo en esta etapa fue la creación de los códigos QR los cuales van a ser los encargados de proyectar el contenido en pantalla, para esto, se utilizó una página llamada “QR CODEMONKEY” la cual permite genera códigos QR aleatorios.

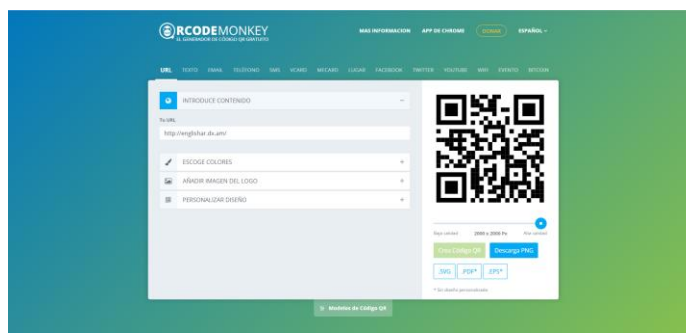


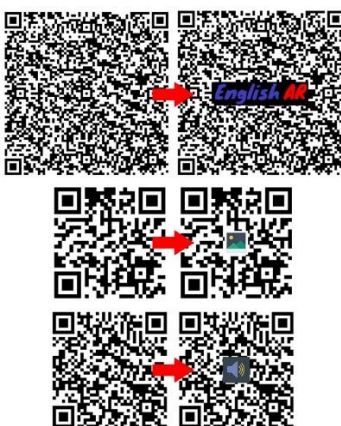
Figura 61 Pagina web QR CODEMONKEY.<sup>23</sup>

<sup>23</sup> QR CODEMONKEY. (2019). Captura de pantalla de página web QR CODEMONKEY. [Figura]. Recuperado de: <https://www.qrcode-monkey.com/es>

En esta página web se generaron 3 Códigos QR:

- **Código QR principal:** el cual se encarga de proyectar el contenido principal de la lección.
- **Código QR de imagen:** el cual tiene como función, mostrar imágenes alusivas al tema que se está tratando en cada lección.
- **Código QR de audio:** el cual sirve para reproducir los audios de la pronunciación de las palabras en inglés.

Luego de esto, se modificaron en el software GIMP para añadirles una imagen que identificara la función de cada uno, en la figura 68 se muestra lo anteriormente dicho



*Figura 62 Códigos QR: Principal, Imagen y Audio. Elaboración propia.*

Una vez finalizada la edición de los códigos QR, estos, se agregaron a la base de datos de Vuforia desde la propia página web, con la finalidad de que, desde el portal de desarrolladores, se descargara un archivo que contiene una configuración necesaria en Unity de la base de datos de los marcadores QR, la cual permite el uso de dichos marcadores de manera local sin la necesidad de contar con acceso a internet.

Luego de realizar dicho proceso, se empezó con el desarrollo de los escenarios, al ser 12 escenarios los que se debían crear, estos mismos debían contar con mecánicas y/o dinámicas similares para hacer más fácil el proceso de desarrollo y a su vez para facilitar el uso de la aplicación sin confundir al usuario final.

Para las lecciones de: los colores, las vocales, el abecedario, las comidas, los medios de transporte y los animales, se crearon las mismas mecánicas, las cuales son:

- Deben existir 2 botones los cuales permitan mostrar los diferentes objetos 3D de la lección.
- Deben estar presentes los nombres en 3D de cada objeto tanto en idioma inglés como en español.
- Al pulsar los nombres en inglés de los objetos, debe reproducirse el audio de pronunciación.
- Según sea la lección, mediante el uso del código QR de imagen, deben aparecer imágenes alusivas al tema principal de la lección.
- Según sea la lección, mediante el uso del código QR de audio, debe aparecer un botón el cual, al momento de ser pulsado, éste, debe mostrar y reproducir la pronunciación en inglés del objeto que este en pantalla.

Se elaboró el prototipo del escenario que debe tener las mecánicas mencionadas anteriormente, a continuación, se muestra el prototipo realizado en Unity:

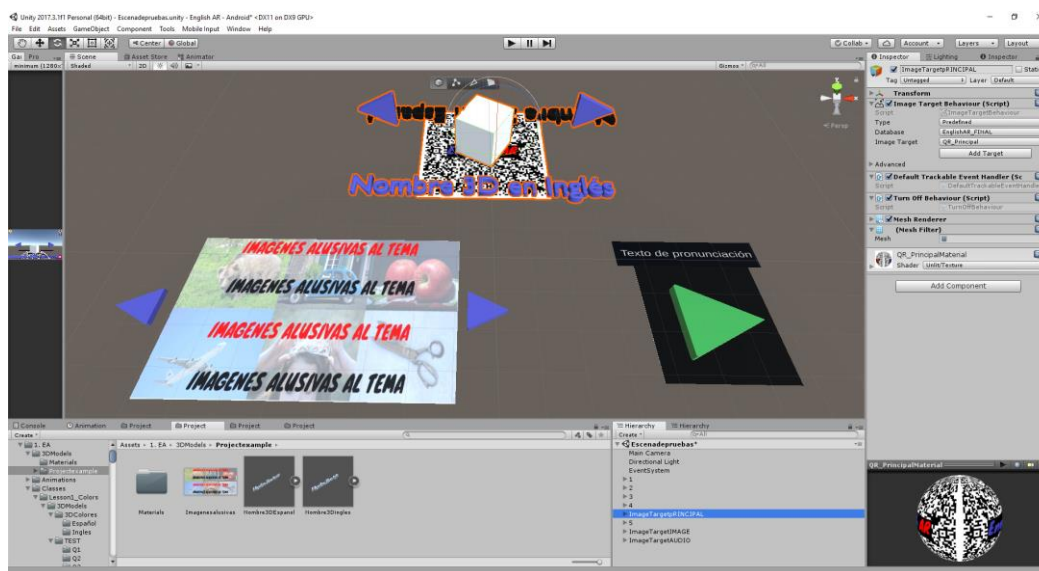


Figura 63 Prototipo de escenario de lecciones. Elaboración propia.

Para el escenario del calendario, se implementaron unas mecánicas similares, la diferencia radica en el uso del código QR de imagen, ya que este no proyecta imágenes sino modelos en 3D ilustrando las 4 estaciones del año. En la figura 70 se aprecia lo anteriormente mencionado.

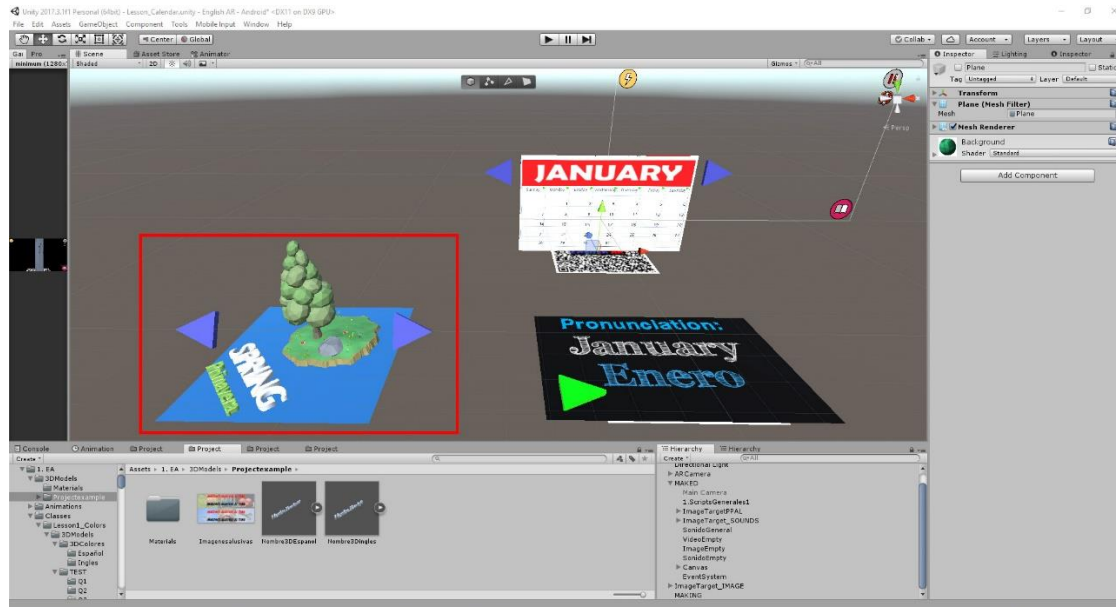


Figura 64 Escenario de lección: El Calendario. Elaboración propia.

Para la creación del escenario de los números, se creó una mecánica la cual consiste en modificar un numero de entre 1 dígito hasta 4 mediante el uso de dos botones que permitan añadir decenas, centenas o milésimas a dicho número. Además de hacer uso del código QR de audio el cual muestra un botón que al ser pulsado deberá mostrar la pronunciación del número que este en pantalla. Para entenderlo mejor se anexa una captura de pantalla del proceso de elaboración de este escenario:

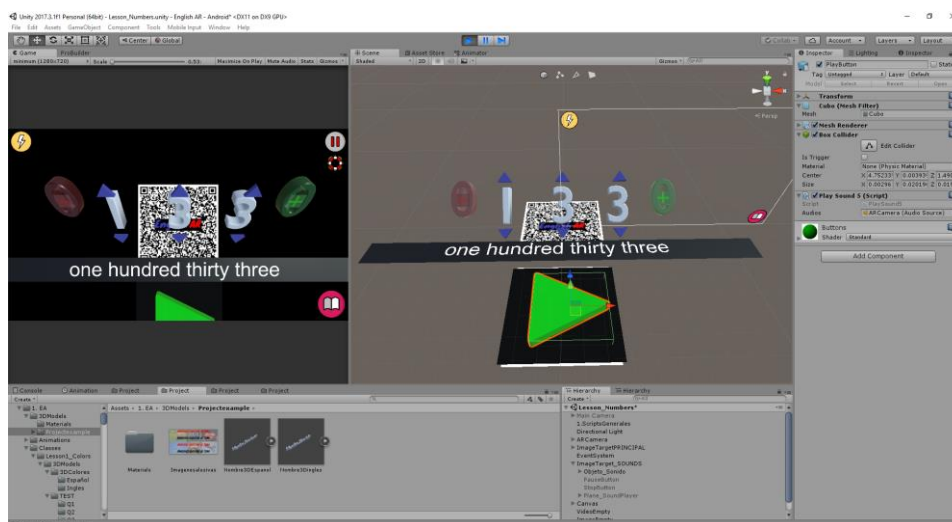


Figura 65 Proceso de elaboración del escenario de Los Números. Elaboración propia.

Luego de realizar el escenario de los números se continuo proceso de elaboración del escenario del reloj.

Las mecánicas de este escenario eran simples, solo se debe simular la interacción con el reloj como si fuera un reloj físico real. Para lograr esto, lo que se hizo fue colocar botones (azules) que permitieran ajustar la hora y luego agregar un botón (verde) el cual, al momento de ser pulsado, reproduzca el audio de pronunciación de la hora que se había ajustado.

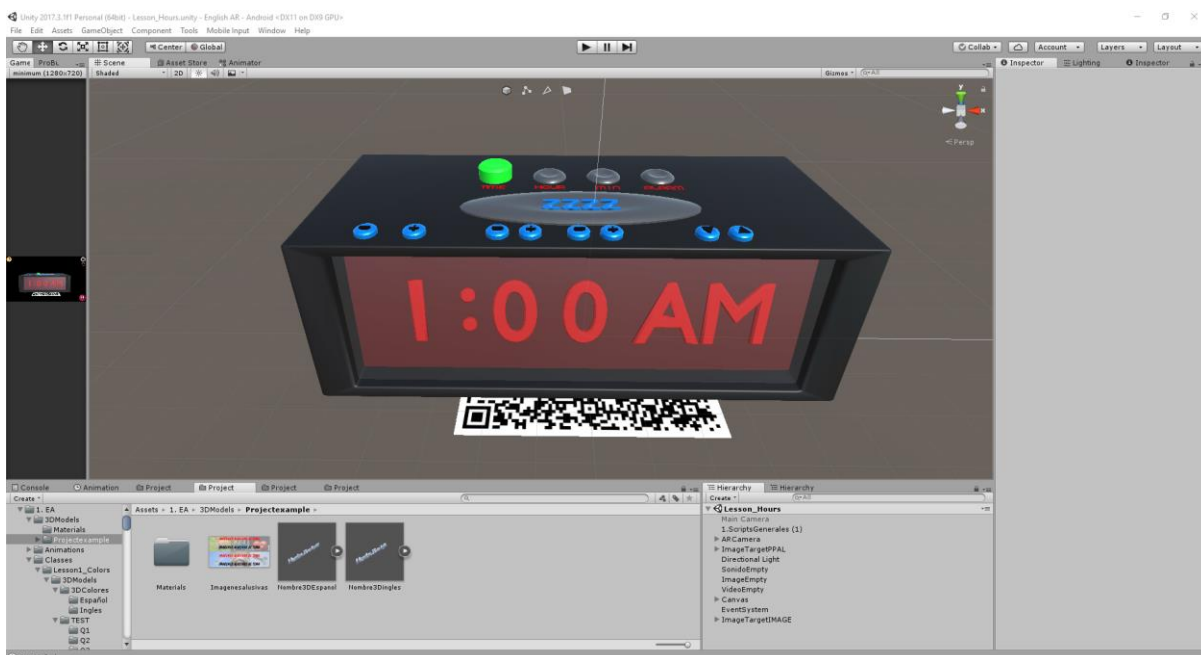


Figura 66 Elaboración del escenario de Las Horas. Elaboración propia.

Para la elaboración de los escenarios de: las partes del cuerpo y las prendas de vestir, se crearon las mismas mecánicas las cuales consisten en:

- Según sea la lección, las partes del cuerpo o las prendas de vestir deben aparecer señaladas por el nombre en inglés y en español
- Al lado del nombre en inglés, debe estar el botón de reproducir el audio de pronunciación de la parte del cuerpo o la prenda de vestir según sea el caso.

En la figura 73 se muestra el resultado de la construcción de ambos escenarios.

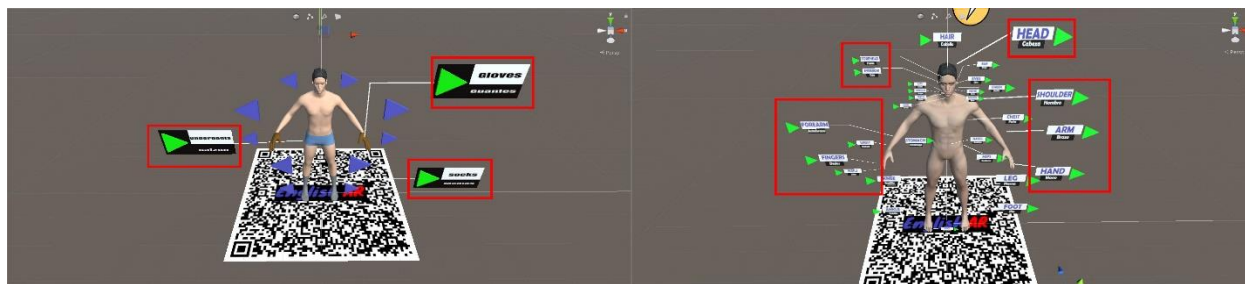


Figura 67 Construcción de escenario y mecánicas de las prendas de vestir y partes del cuerpo. Elaboración propia.

Para la elaboración del escenario de las partes de la casa, se importó a Unity el modelo del interior y exterior de la casa previamente hecho en Blender, luego, se agregaron al escenario, modelos 3D de muebles gratuitos descargados de la propia tienda de Unity con el fin de simular el entorno de una casa. Además, se colocaron los nombres de las partes de la casa tanto en inglés como en español en la parte de superior de la misma y, por último, se añadió un botón el cual permitía mostrar el interior o el exterior de la casa según fuera el caso. Se anexa una captura de pantalla para observar el resultado de este proceso.

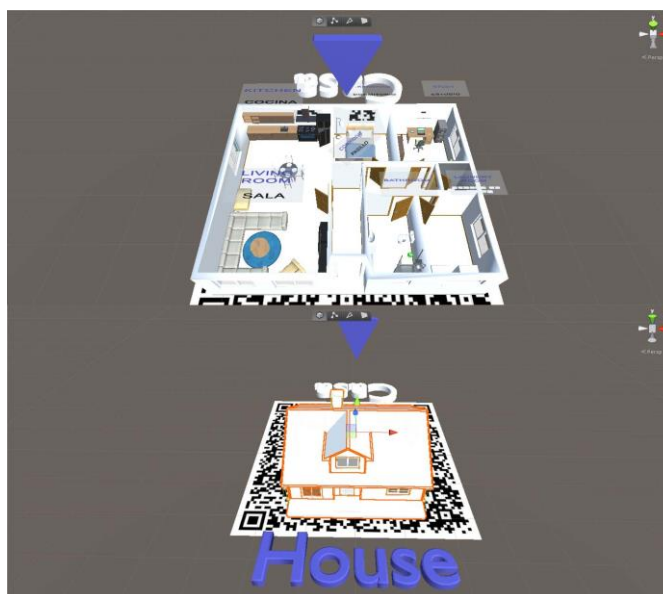


Figura 68 Creación del escenario de las partes de la casa. Elaboración propia.

Para finalizar la etapa de creación de los escenarios se realizó el prototipo del escenario de los exámenes para cada lección.

Se creó un escenario que fuera genérico para 12 exámenes, dichos escenarios de exámenes debían tener las mismas dinámicas de preguntas y respuestas. Los escenarios contienen preguntas de respuesta múltiple, así como también preguntas de falso y verdadero.

Se diseñó el escenario de manera que simulara una hoja de examen, las preguntas y opciones de respuesta se elaboraron en 3D en el software Blender como anteriormente se había hecho con la herramienta de texto en 3D, para algunas de las preguntas y respuestas se reutilizaron los audios que ya se habían creado para los escenarios de lección, así como también algunos de los nombres 3D de los modelos. Tal como se aprecia en la siguiente figura:

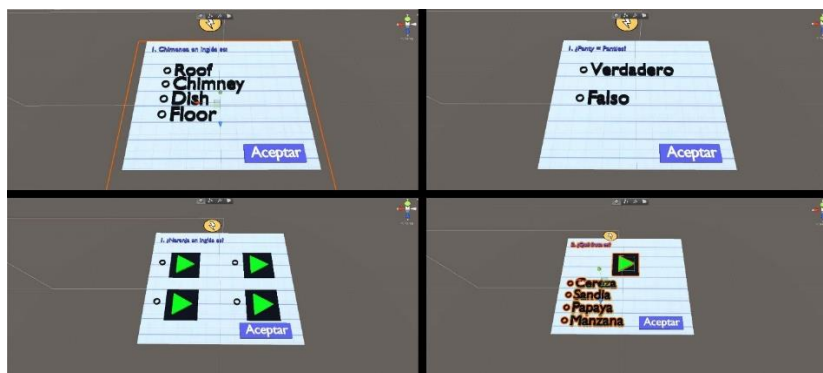


Figura 69 Prototipo de exámenes de lección. Elaboración propia.

### 8.3.5 Desarrollo Iterativo: Etapa 5. Desarrollo de código

En esta etapa se desarrolló todo el código fuente de las mecánicas para los escenarios, se crearon los scripts para reproducir audio, visualizar diferentes objetos, ejecutar animaciones.

Para la elaboración de los scripts se utilizó el software Visual Studio. A continuación, se muestran unas capturas de pantalla de los principales scripts usados en los escenarios.

- **Script para reproducir Audio:**

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class PlayAudio : MonoBehaviour {
6
7      //Estas variables deben ser asignadas desde el inspector de Unity
8      public AudioSource Source;
9      public AudioClip Clip;
10
11     void Start () { ... }
12
13     void Update () { ... }
14
15     //El objeto que tenga este script accederá a esta función al momento de ser pulsado.
16     public void playAudio()
17     {
18         //Se reproducirá el clip de audio que se asigne desde el inspector de Unity.
19         Source.PlayOneShot(Clip);
20     }
21
22
23
24
25

```

Figura 70 captura de pantalla de fragmento de código para reproducir audio. Elaboración propia.

- **Script para visualizar u ocultar elementos del escenario:**

```

public class SeeandHideObjects : MonoBehaviour {

    int pointer; //apuntador de tipo entero.
    public GameObject[] Objects; //Array de objetos que se mostraran u ocultaran
    void Start () {

    }

    void Update () {

    }

    public void Next()
    {
        pointer++;

        if (pointer > 0)
        {
            Objects[pointer].SetActive(true);
            Objects[pointer - 1].SetActive(false);
        }

        if (pointer >= Objects.Length - 1)
            pointer = 0;
    }

    public void Previous()
    {
        pointer--;

        if (pointer < 0)
            pointer = Objects.Length - 1;

        if (pointer < Objects.Length - 1)
        {
            Objects[pointer].SetActive(true);
            Objects[pointer + 1].SetActive(false);
        }
    }
}

```

Figura 71 captura de pantalla de fragmento de código para mostrar u ocultar elementos. Elaboración propia.

- **Script para reproducir animaciones:**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class PlayAnimation : MonoBehaviour {

    public Animator Anim; //esta variable debe ser asignada desde el inspector de Unity
                        // debe ser un objeto que tenga la propiedad animator y que
                        // a su vez tenga un controlador de animaciones el cual
                        // cambie los estados de las animaciones, es decir, las pause
                        // las detenga, o las reproduzca.

    void Start () {

    }
    void Update () {

    }

    public void playAnimation()
    {
        //activa 1 sola vez la animacion que se asigne desde el inspector de Unity
        Anim.SetTrigger("AnimationToPlay");
    }
}

```

Figura 72 Captura de pantalla de fragmento de código para reproducir animaciones. Elaboración propia.

- **Script para cambio de texturas (Imágenes):**

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class ChangeMaterial : MonoBehaviour {

    int pointer; //apuntador
    public GameObject PlaneImage; //plano al cual se le cambiara el material por las texturas de imagenes
    public Material[] Images; //array de texturas de imagenes

    void Start () {
    }
    void Update () {
    }
    public void Next()
    {
        pointer++;

        if (pointer > 0)
        {
            PlaneImage.GetComponent<MeshRenderer>().material = Images[pointer];
        }

        if (pointer >= Images.Length - 1)
            pointer = 0;
    }

    public void Previous()
    {
        pointer--;

        if (pointer < 0)
            pointer = Images.Length - 1;

        if (pointer < Images.Length - 1)
        {
            PlaneImage.GetComponent<MeshRenderer>().material = Images[pointer];
        }
    }
}

```

Figura 73 Captura de pantalla de fragmento de código para cambio y visualización de imágenes. Elaboración propia.

## 8.4 Estabilización

En esta fase, se realizó el acople de todos los componentes elaborados en la fase de producción, con la finalidad de construir un aplicativo funcional y estructurado.

- **Adición de modelos 3D:** se agregaron todos los modelos 3D y sus respectivos nombres al prototipo de escenario de las lecciones de: los colores, las vocales, el abecedario, las comidas, los medios de transporte y los animales. Y en general todos los modelos faltantes a cada escenario de lección. Tal como se muestra en la siguiente figura:

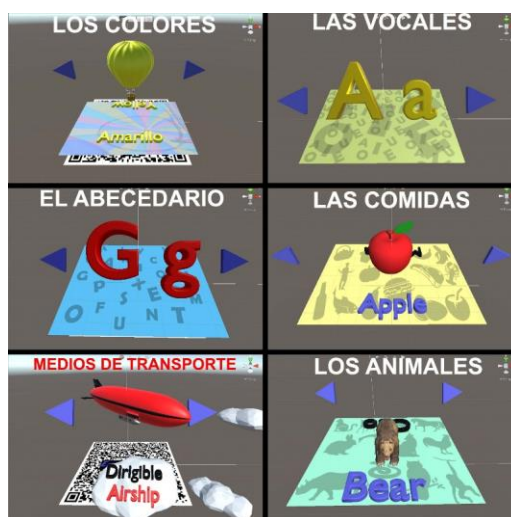


Figura 74 adición de modelos 3D a diversos prototipo de escenarios. Elaboración propia.

- **Adición de Imágenes y Audios:** Se agregaron todas las imágenes que van a ser proyectadas mediante el uso del código QR, para cada uno de los escenarios en donde se requieran dichas imágenes. Adicionalmente se agregaron todos los audios de las pronunciaci3nes. El resultado de este proceso se observa en la figura 81



Figura 75 adici3n de im3genes 3D a diversos prototipo de escenarios. Elaboraci3n propia.

- **Adici3n de c3digo (scripts) a escenarios:** Se agregaron los scripts programados de las mec3nicas a todos los escenarios correspondientes, incluyendo todo el c3digo necesario para los ex3menes. A continuaci3n, se muestra el resultado de este proceso en diferentes escenarios.

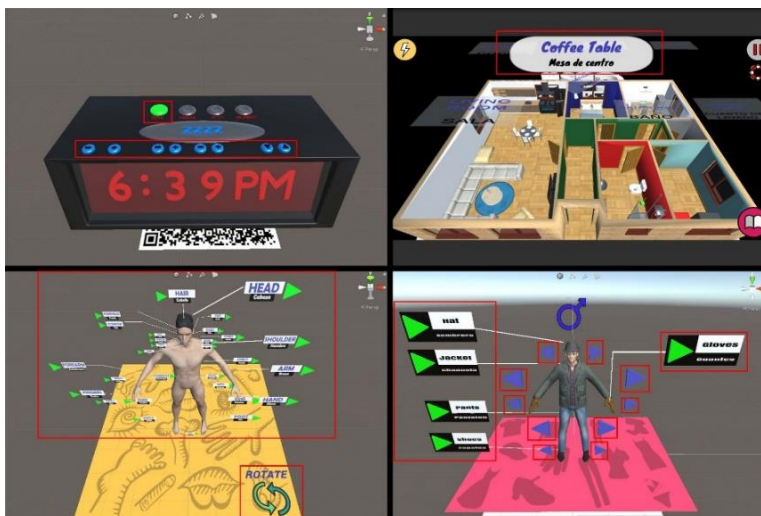


Figura 76 Ejemplos de mec3nicas programadas en funcionamiento en cada escenario. Elaboraci3n propia.

- Adición de código(scripts) a menú principal:** Se agregó un script necesario a cada botón de lecciones en el menú principal, esto con la finalidad de conectar el menú principal con cada escenario de lección, al momento de que sea pulsado cada botón, se debe abrir la lección correspondiente. Así mismo, se agregó un script a los botones que tienen como función abrir el examen de la lección, con el fin de conectar los escenarios de las lecciones con los escenarios de los exámenes de lección correspondiente.

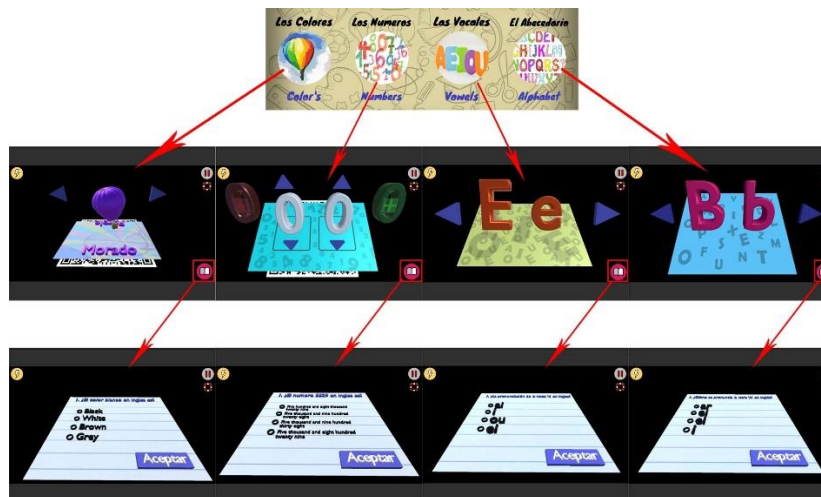


Figura 77 Conexión entre: menú principal, escenarios de lecciones y escenarios de exámenes. Elaboración propia.

Luego de ensamblar todos los elementos que componen a cada uno de los escenarios, y hacer ajustes a la programación de las interfaces y mecánicas, se realizó la compilación de todo el proyecto usando la herramienta de Unity de compilación, para esto, se seleccionó la plataforma objetivo que para este caso fue Android, y adicionalmente se agregaron todos los escenarios en una lista la cual debía ir un orden específico de acuerdo al Wireframe y al nivel de complejidad de cada lección. Para entenderlo mejor, se anexa la siguiente figura en donde se detallan dichos procesos.

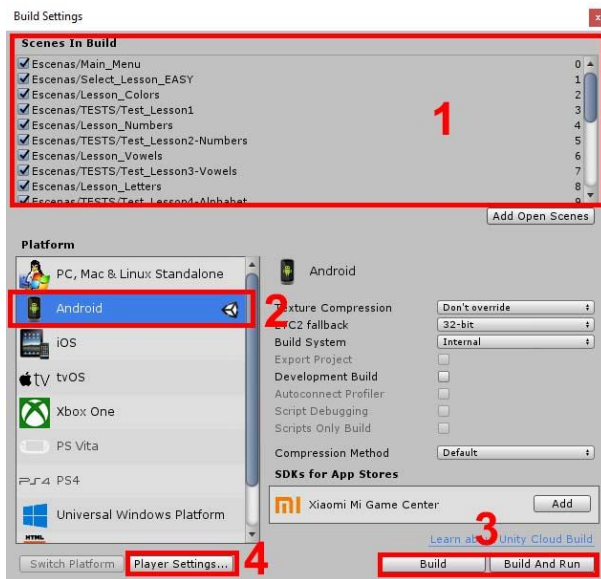


Figura 78 Proceso de compilación en Unity. Elaboración propia.

En la figura anterior se observa lo siguiente:

- **Escenas en construcción (1):** como se mencionó anteriormente, en este apartado se agregan todos los escenarios realizados que van en la aplicación.
- **Plataforma (2):** En este apartado se selecciona la plataforma objetivo del proyecto, para este caso se seleccionó la plataforma Android.
- **Compilar/ Compilar y ejecutar (3):** Estas opciones permiten dar inicio a la compilación de la aplicación, la única diferencia entre estas 2 es que la opción “Build” compila la aplicación y la almacena en el disco duro del computador, y la opción “Build and run” Compila la aplicación y la instala directamente en el dispositivo Android que esté conectado al computador vía USB. Durante la fase de pruebas se utilizó primeramente la opción de “Build and Run” para ahorrar tiempos de espera, y la opción “Build” se utilizó para hacer la compilación final de la aplicación.
- **Ajustes de Unity (4):** Antes de poder compilar cualquier aplicación Unity pide serie de datos de identificación que deben ser diligenciados. En la figura 85 se observan dichos datos.

Identification	
Package Name	com.ToxicStudios.EnglishAR
Version*	1.5
Bundle Version Code	6
Minimum API Level	Android 5.0 'Lollipop' (API level 21)
Target API Level	Automatic (highest installed)

Figura 79 Información de identificación de la app. Elaboración propia.

- **Nombre del paquete:** “Un identificador de paquete es una cadena que identifica una aplicación. Está escrito en lo que se conoce como estilo DNS inverso” (Unity Technologies, 2017).
- **Versión:** La versión de la aplicación que se compilara.
- **Código de versión del paquete:** un número de versión interno. Este número se usa solo para determinar si una versión es más reciente que otra, con números más altos que indican versiones más recientes.
- **Nivel de API Mínimo:** Aquí se selecciona el nivel de API o versión de Android mínima que requieren los dispositivos para poder ejecutar la aplicación, para este caso se seleccionó Android 5.0 Lollipop (API 21).
- **Objetivo API:** A nivel interno, Unity selecciona la API más reciente que está en el equipo para poder realizar todo el proceso de compilación.

Luego de ajustar los parámetros anteriores, se modificó la configuración de resolución y presentación de la aplicación, en donde se seleccionó la orientación de la aplicación la cual debía ser horizontal (Landscape), esto con la finalidad de que la aplicación solo funcionara de dicha manera y no vertical (Portrait). Para entender esto un poco mejor se agrega una figura que muestra este proceso.

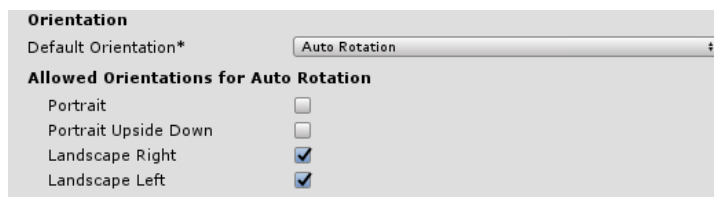


Figura 80 Configuración de orientación de la aplicación. Elaboración propia.

Adicionalmente, se agregó el icono que sería mostrado en todos los dispositivos en los que estuviera instalada la aplicación en el apartado de Unity de presentación de la aplicación. Este icono debía tener una serie de resolución de pixeles específica como por ejemplo 192x192, 144x144 o 96x96. El resultado de este proceso se puede observar en la siguiente figura.

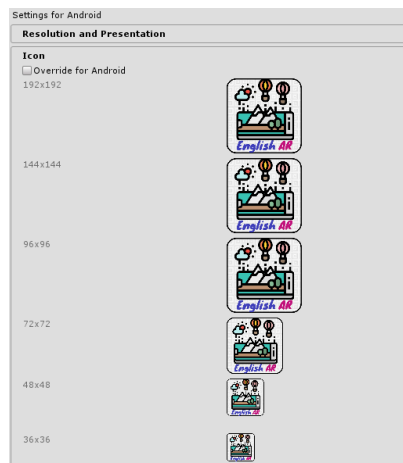


Figura 81 Icono de la aplicación con diferente resolución de píxeles. Elaboración propia.

Finalmente, se realizó la configuración de publicación. Esta configuración es única y exclusivamente para aplicaciones que serán lanzada en alguna tienda de aplicaciones, como en este caso se realizaría la oferta de la aplicación en la Google Play se tuvo que realizar dicha configuración.

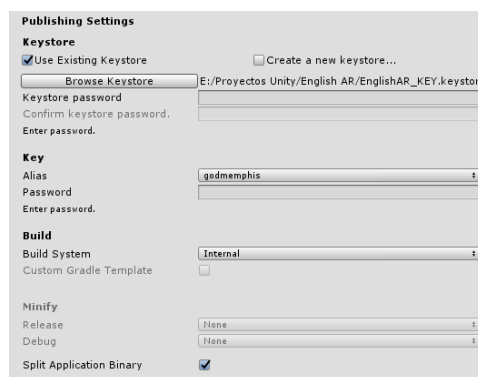


Figura 82 Configuraciones de publicación. Elaboración propia.

En este apartado se crea una firma digital única para la aplicación, la cual indica a la Google Play o cualquier otra tienda de aplicaciones que la persona o empresa que está tratando de subir la aplicación a sus servidores es auténtica y propietaria de la misma. Además, esta firma sirve para el momento en el que se desee realizar una actualización a la aplicación, si no se cuenta con esta firma no se podrá realizar ningún tipo de modificación a la aplicación a futuro.

Luego de haber realizado toda la configuración de compilación del prototipo se dio inicio a la fase de pruebas del mismo.

## 9. TESTER

### 9.1 Pruebas de caja blanca

#### 9.1.1 Prueba de reconocimiento de Códigos QR

Se realizaron pruebas para validar el funcionamiento correcto de la cámara del dispositivo al momento de detectar los marcadores puestos por el usuario en el entorno real, con esto se hace referencia a los códigos QR.



Figura 83. Reconocimiento de Códigos QR. Elaboración propia.

Se observó, que, cuanto mayor tamaño tuviera el código QR, más alcance de visualización de objetos tendría la cámara del dispositivo, y a menor tamaño del código QR los elementos que proyectara dicho código desaparecerían. Cabe resaltar que lo anterior, no sucedía por la pérdida del enfoque de la cámara, este acontecimiento sucedía por la propia configuración de recorte de planos predeterminada de la librería Vuforia, a continuación, en la figura 89 (imagen parte derecha), se muestra lo anteriormente descrito.

Dicho problema se solucionó, ajustando el parámetro de distancia de los recortes de plano propios de Unity, dejando como evidencia que la configuración realizada fue efectiva al observar que, si se aleja la cámara del dispositivo de los códigos QR, los elementos proyectados seguirían viéndose incluso si la distancia es mayor tal como se aprecia en la figura 91.

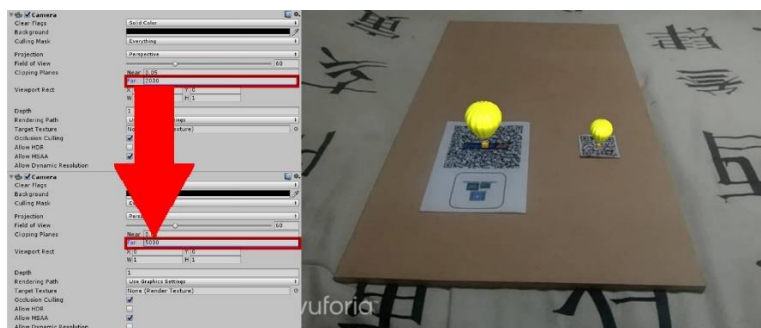


Figura 84. Resultado de ajustes a la distancia de recortes de plano. Elaboración propia.

### 9.1.2 Prueba de interacción del usuario con elementos proyectados

Puesto que el usuario debía interactuar con los elementos proyectados, se tuvo que verificar que el sistema respondiera a las peticiones del usuario, es decir, el usuario tenía que realizar lecturas de toques en pantalla para saber qué acción es la que debía ejecutar. Para este caso, se programó un script, el cual tenía como función, detectar dichos toques en pantalla para simularlos en forma de clics, como si fuese el mouse de un computador. A continuación, se muestra un fragmento del script encargado de dicha función.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class DetectInput : MonoBehaviour {

    int Counter;
    // Use this for initialization
    void Start() {

    }

    // Update is called once per frame
    void Update() {

    }

    /* La función OnMouseDown() detecta cuando el usuario hizo clic
    sobre el elemento que contiene este script, en este caso
    no será un clic de un mouse sino que será un toque en pantalla
    dado por el usuario, el script al detectar dicha acción ejecutará
    las líneas de código que se encuentren en dicha función.

    Para este ejemplo, al hacer clic o tocar la pantalla sobre este
    elemento este deberá aparecer un mensaje en la consola indicando
    si se pulso o no sobre el objeto y cuantas veces lo hizo
    */

    private void OnMouseDown()
    {
        /* Cada vez que el usuario pulse el objeto la variable "Counter"
        * sumará +1 a su valor, luego de esto se concatenará con una cadena
        * string para visualizar el mensaje completo de si se pulso o no y
        * cuantas veces se hizo la misma acción*/
        Counter++;

        Debug.Log("Usted pulso este objeto " + Counter + " Veces");
    }
}
```

Figura 85. Fragmento de código de detección de toques en pantalla. Elaboración propia.

Dicho script debía estar asociado a un elemento en 3D, con el cual el usuario pudiera interactuar. Además, de que éste debía tener la propiedad llamada “Box Collider” que así, este elemento detectara si el usuario estaba tratando de interactuar con él y este hiciera su respectiva función. En la siguiente figura se observa lo anteriormente dicho.

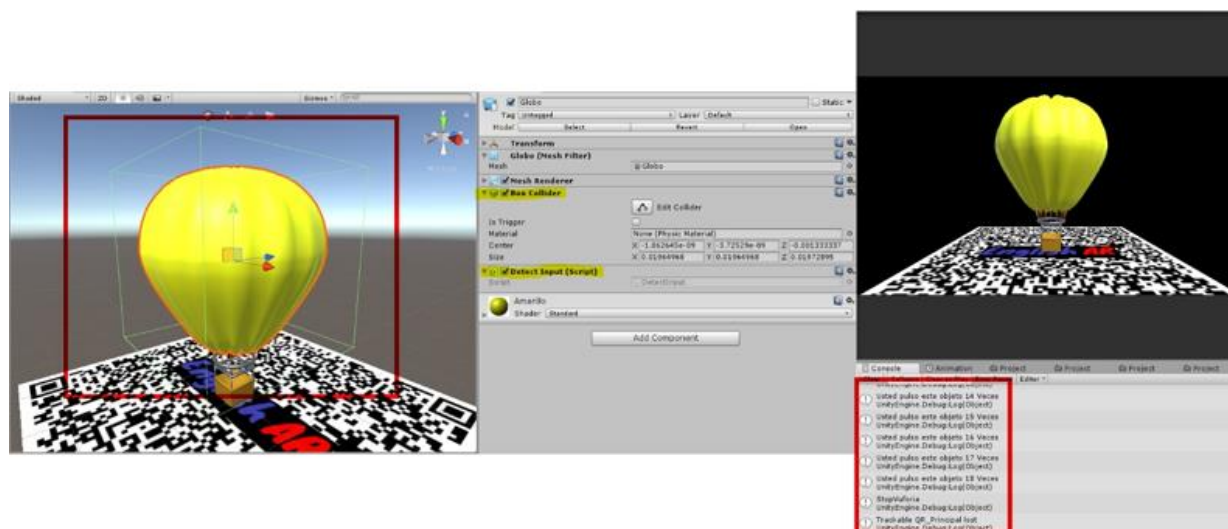


Figura 86. Prueba de funcionamiento de script que detecta toques en pantalla. Elaboración propia.

Esta función podía realizar algo tan sencillo como mostrar un mensaje en consola, hasta algo más complejo como poder ejecutar partes de código de otro script; si dicho elemento no tenía esta propiedad, no se ejecutaría ningún tipo de acción, por eso fue de vital importancia realizar las debidas pruebas al código y verificar que cada elemento interactivo tuviese dicha propiedad.

## 9.2 Pruebas de caja negra

### 9.2.1 Pruebas de funcionamiento

- **Pruebas de interfaces de usuario**

Un aspecto importante que se debe tener en cuenta a la hora de realizar un desarrollo de software es el de contar con una interfaz de usuario que haga más fácil el manejo de la aplicación. Es por ello que se realizaron diversas pruebas para verificar el correcto funcionamiento de diferentes aspectos, tales como; la navegación entre las diferentes interfaces de usuario, y la validación de cada botón de dicha interfaz que cumpliera con su respectiva función.

- **Prueba a interfaz de menú principal:** Se verificó que cada botón de la interfaz del menú principal ejecutara la acción que le fue programada, la acción más importante que debía ponerse a prueba era la redirección del usuario desde el menú principal hacia cada una de las lecciones de aprendizaje.



Figura 87. Secuencia de acciones al seleccionar una lección. Elaboración propia.

En la figura anterior, se observa que al seleccionar la lección “Los colores”, el sistema debía mostrar una escena de carga y posteriormente redirigir al usuario hacia dicha lección. Esta secuencia debía suceder con todos los botones de las lecciones restantes. Por eso, fue importante verificar y corregir todos los errores de mal direccionamiento, para evitar que el usuario entrara a una lección equivocada.

También se realizaron pruebas a los botones “perfil de usuario” y “ayuda” para verificar si estaban mostrando los datos correctos y no un intercambio de información. El botón de “ayuda” debía mostrar una pantalla de instrucciones y el botón de “perfil de usuario” debía mostrar una pantalla con la información del usuario, un ejemplo de esto se observa en la figura 95.



Figura 88. apertura de pantalla de ayuda al pulsar el botón de ayuda. Elaboración propia.

Por último, se realizaron pruebas a los botones de navegación del menú principal, en donde se verificó que los botones al ser pulsados mostraran las demás lecciones que había en el menú principal, tal como se muestra en la figura 97.



Figura 89. Flujo de navegación entre lecciones de menú principal. Elaboración propia.

## 9.2.2 Pruebas de experiencia de usuario

Se puede definir a la experiencia de usuario como el sentimiento de una persona al utilizar un producto, servicio o sistema. Esta misma, está ligada a la usabilidad, pero no se pueden considerar como lo mismo, ya que la usabilidad se refiere a la facilidad con la que cuenta un usuario para cumplir un objetivo en concreto haciendo uso de una herramienta. Mientras que el termino de experiencia de usuario se define más concretamente como “las percepciones y respuestas de las personas, resultantes del uso o anticipación de uso de un producto, sistema o servicio” (ISO, 2010).

Dicho lo anterior, se tuvo en cuenta el realizar pruebas a las diferentes interfaces de usuario de la aplicación, ya que estas mismas, son con las que el usuario va a interactuar en gran parte dentro de la aplicación, por eso fue de vital importancia realizar las debidas pruebas para garantizar su usabilidad y generar una buena experiencia de usuario.

Se empezaron a realizar pruebas a la interfaz de creación de nombre de perfil y selección de avatar, en la cual se observó que tanto el nombre como el avatar seleccionado no se estaba almacenando en la memoria del dispositivo, lo cual dejaba como resultado que el usuario cada vez que entrara a la aplicación debía crear un perfil nuevo todas las veces. Se realizaron las correcciones pertinentes a este problema para dar la solución al mismo.

Luego de esto, se realizaron pruebas a las interfaces que contienen las guías de uso por primera vez, se verifico que estas interfaces solo se mostraran la primera vez que se accedía al menú principal.



Figura 90. Interfaz de bienvenida al acceder por primera vez a la aplicación. Elaboración propia.

Como ya se habían realizado pruebas a la interfaz del menú principal previamente, se omitió el proceso de examinar el funcionamiento y experiencia de las mismas, para así, proceder a la verificación del correcto funcionamiento de la interfaz que había en el escenario de lecciones y exámenes de lección, ya que, al ser la misma para ambos escenarios, se debía verificar que dicha interfaz funcionara adecuadamente.

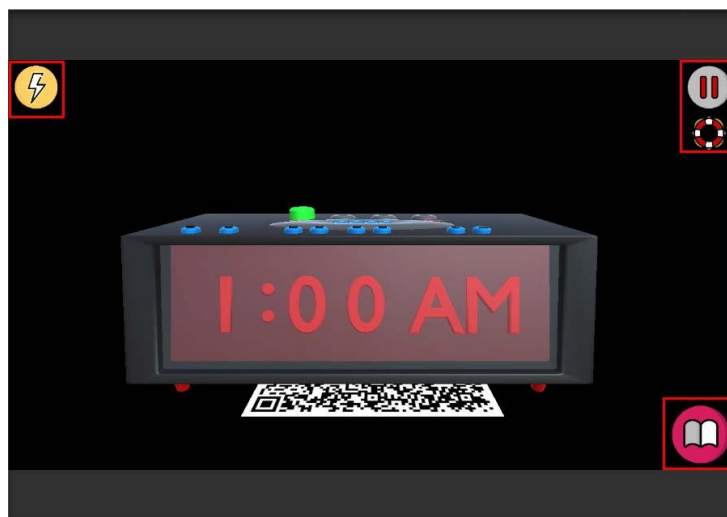


Figura 91. Interfaz de usuario de escenario de lecciones y exámenes de lección. Elaboración propia.

Se detectó que el botón de activación/desactivación del flash del dispositivo no cumplía con su función, se realizaron las debidas correcciones para validar su funcionalidad.

También se detectó que el botón encargado de enviar al usuario al examen de lección, presentaba anomalías en el funcionamiento ya que, al momento de ser pulsado, direccionaba al usuario directamente al examen de lección, en vez de mostrar un mensaje de advertencia previo al usuario.

Se realizaron las modificaciones pertinentes para corroborar el correcto funcionamiento de este botón, dejando como resultado lo que se observa en la figura 100.

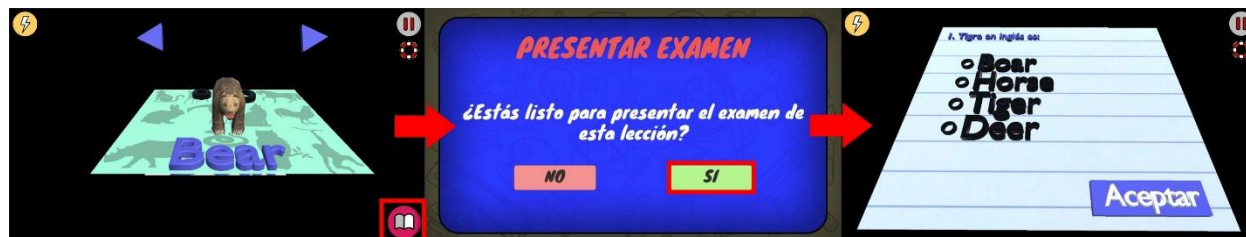


Figura 92. Secuencia de ejecución de examen de lección. Elaboración propia.

### 9.2.3 Pruebas de rendimiento

Se realizaron pruebas de rendimiento para observar cómo era el funcionamiento en general de la aplicación en diferentes dispositivos. Esto se realizó con la finalidad de poder ver más a detalle cómo se comportaba la aplicación con diferente hardware y software. En dichas pruebas se tuvieron en cuenta 4 aspectos los cuales son: tiempos de carga, consumo de memoria RAM, consumo y temperatura de CPU y, por último, el consumo de batería.

Las pruebas se realizaron en 3 dispositivos diferentes los cuales fueron:

- 1 teléfono móvil LEECO modelo LePro3 x727 cuyas características principales son: versión de Android 6.0.1 Marshmallow, 4GB de memoria RAM, 64GB de memoria de almacenamiento y procesador Snapdragon x821 de cuatro núcleos.
- 1 teléfono móvil Huawei modelo Honor 4x cuyas características principales son: Android 5.1.1 Lollipop, 2GB de memoria RAM, 8GB de memoria de almacenamiento y un procesador Kirin 620 de 8 núcleos.
- 1 teléfono móvil LEECO modelo LeMax2 cuyas características principales son: versión de Android 6.0 Marshmallow, 4GB de memoria RAM, 32GB de memoria de almacenamiento y procesador Snapdragon x820 de cuatro núcleos.

- **Prueba: tiempos de instalación**

Como se puede apreciar en la tabla 1 y en la tabla 2 los tiempos de instalación y desinstalación más largos fue el del teléfono móvil Huawei y el más corto fue el del móvil LEECO LePro3.

DISPOSITIVO	TIEMPO DE INSTALACION
Huawei Honor 4X	6,05 Minutos
LEECO LePro3 x727	27,11 Segundos
LEECO LeMax 2	30,05 Segundos

*Tabla 1. Tiempos de instalación de la aplicación. Elaboración propia.*

#### **Prueba: tiempos de desinstalación**

DISPOSITIVO	TIEMPO DE DESINSTALACION
Huawei Honor 4X	10,34 Segundos
LEECO LePro3 x727	1,41 Segundos
LEECO LeMax 2	2,15 Segundos

*Tabla 2. Tiempos de desinstalación de la aplicación. Elaboración propia.*

- **Prueba: tiempos de carga entre escenas**

Se realizaron las pruebas de tiempo de carga de escenas, en la tabla 3 se observa el tiempo en segundos y milésimas de segundo que le tomo a cada dispositivo cargar cada una de las 12 lecciones.

DISPOSITIVO	T.E1	T.E2	T.E3	T.E4	T.E5	T.E6	T.E7	T.E8	T.E9	T.E10	T.E11	T.E12	PROMEDIO
Huawei Honor 4X	5,53	4,32	3,17	5,44	8,33	4,25	5,01	5,36	3,42	4,26	5,09	7,14	5,11
LEECO LePro3 x727	2,88	2,57	3,08	2,49	4,26	2,55	2,39	3,02	2,22	2,87	3,14	5,32	3,07
LEECO LeMax 2	3,04	3,15	3,66	4,03	4,77	2,63	2,54	3,48	2,52	2,33	3,89	6,18	3,52

Tabla 3. Tiempos de carga de escenas en los 3 dispositivos de prueba. Elaboración propia.

- **Consumo de memoria RAM**

Para realizar esta prueba se instaló una aplicación en los 3 dispositivos de prueba llamada “CPU/GPU Meter” la cual era capaz de mostrar la memoria RAM disponible del dispositivo, para calcular cual era el consumo promedio, se tuvo que realizar una sencilla operación matemática la cual consistía en tomar el valor de memoria RAM disponible antes de abrir la aplicación y después de abrirla para así tener dos valores y poder restarlos entre si y obtener un resultado el cual sería el total consumido en ese lapso de tiempo. A continuación, en las tablas 4 y 5, se observa el consumo de memoria RAM en el dispositivo Huawei Honor 4X, luego, en las tablas 6 y 7 se aprecia el consumo en el dispositivo LEECO LePro3 y, por último, en las tablas 8 y 9 se muestra el consumo en el dispositivo LEECO LeMax2 antes y durante la ejecución de la aplicación.

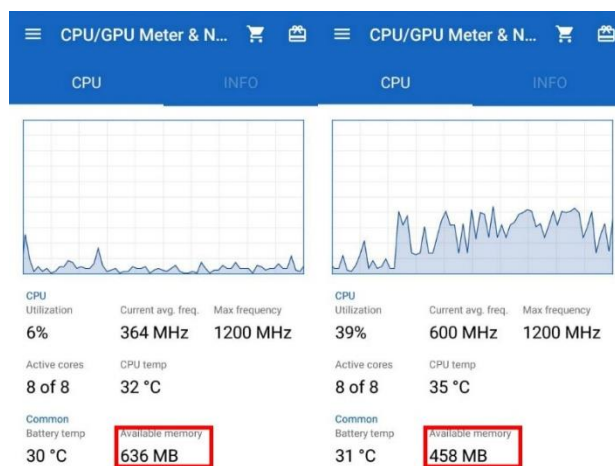


Figura 93. Consumo de memoria RAM telefono móvil Huawei. Elaboración propia.

DISPOSITIVO	MEMORIA DISPONIBLE ANTES	MEMORIA DISPONIBLE DURANTE	PROMEDIO
Huawei Honor 4X	636MB	458MB	178MB

Tabla 4. Promedio de consumo de memoria RAM antes y después de ejecución Huawei Honor 4X. Elaboración propia.

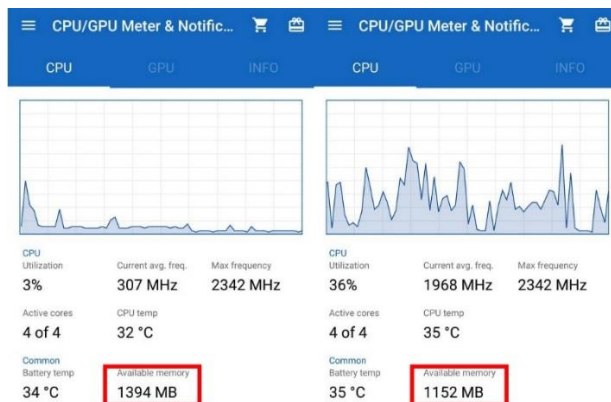


Figura 94. Consumo de memoria RAM teléfono móvil LEECO LePro3. Elaboración propia.

DISPOSITIVO	MEMORIA DISPONIBLE ANTES	MEMORIA DISPONIBLE DURANTE	PROMEDIO
LEECO LePro3 x727	1394MB	1152MB	269MB

Tabla 5. Promedio de consumo de memoria RAM antes y después de ejecución LEECO LePro3. Elaboración propia.

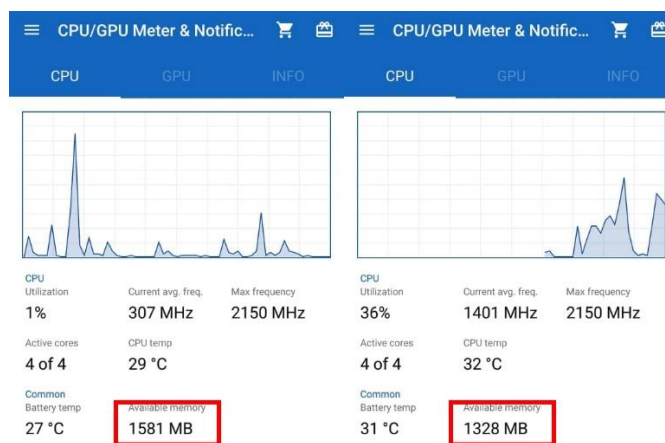


Figura 95. Consumo de memoria RAM teléfono móvil LEECO LeMax2. Elaboración propia.

DISPOSITIVO	MEMORIA DISPONIBLE ANTES	MEMORIA DISPONIBLE DURANTE	PROMEDIO
LEECO LeMax 2	1581MB	1328MB	253MB

Tabla 6. Promedio de consumo de memoria RAM antes y después de ejecución LEECO LeMax2. Elaboración propia.

- **Prueba: Consumo y temperatura de CPU**

La misma aplicación mencionada anteriormente durante la prueba permitió observar el consumo de CPU, así como también ver la temperatura del mismo, mientras se ejecutaba la aplicación, a continuación, se muestran los resultados obtenidos.

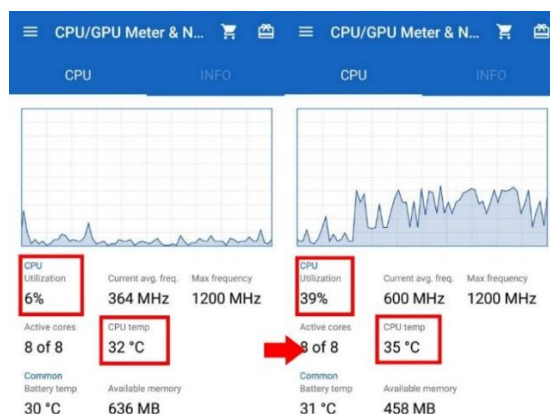


Figura 96. Consumo y temperatura de CPU en teléfono móvil Huawei Honor 4X. Elaboración propia

En la figura anterior se observa el consumo y la temperatura de CPU mientras la aplicación se ejecutaba en el teléfono móvil Huawei Honor 4X, se puede observar que el pico más alto fue de 39% de uso de CPU y la temperatura solo aumento 3°C.

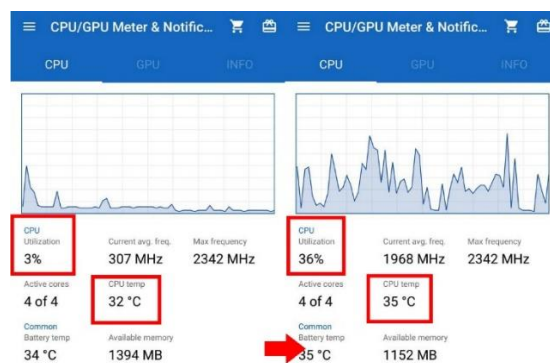


Figura 97. Consumo y temperatura de CPU en teléfono móvil LEECO LePro3. Elaboración propia.

Como se pudo observar en la figura 105, el pico más alto en cuanto al consumo de CPU en el teléfono móvil LEECO LePro3 fue de 36% de uso, y al igual que el teléfono Huawei la temperatura subió 3°C únicamente.

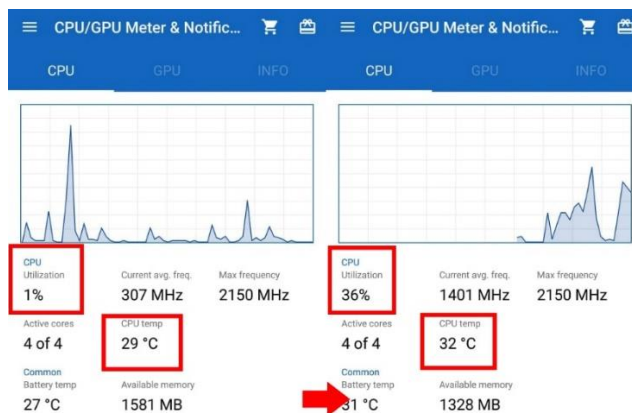


Figura 98. Consumo y temperatura de CPU en teléfono móvil LEECO LeMax2. Elaboración Propia.

La figura anterior muestra los resultados del consumo y temperatura de CPU en el teléfono móvil LEECO LeMax2, dando como resultado un 36% el pico más alto de uso, y un aumento en la temperatura de 3°C.

- **Prueba: Consumo de batería**

Por último, se realizaron pruebas de consumo de batería en un lapso de 20 minutos de uso de la aplicación en cada dispositivo de pruebas. Los resultados se observan en la tabla 7.

DISPOSITIVO	% INICIAL	ACTUAL	% DESPUES DE 20 MIN USO	CONSUMO TOTAL
Huawei Honor 4X		57%	45%	12%
LEECO LePro3 x727		58%	47%	11%
LEECO LeMax 2		72%	63%	9%

Tabla 7. Promedio de consumo de batería. Elaboración Propia.

La tabla anterior contiene los porcentajes del consumo de batería mientras se ejecutaba la aplicación durante 20 minutos, siendo 11% el promedio de consumo. Cabe resaltar que dicho porcentaje de consumo de batería varía según el dispositivo, ya que algunos dispositivos cuentan con baterías de más capacidad que otros.

### 9.2.3 Pruebas de compatibilidad

Se realizaron las pruebas de compatibilidad de la aplicación con diferentes sistemas operativos, con la finalidad de comprobar si la aplicación era capaz de ejecutarse en sistemas operativos Android inferiores a Android 5.0 Lollipop (API 21), o si sólo funcionaba en versiones superiores a Android 5.0.

Se realizó la prueba en 2 dispositivos los cuales fueron:

- 1 Teléfono móvil de la marca Huawei modelo Honor 4x con versión de Android Lollipop 5.1.1
- 1 Teléfono móvil de la marca Samsung modelo S5 sm-g900m con versión de Android Kit-Kat 4.4.2

En la prueba con el teléfono móvil Huawei, se observó que la aplicación se instaló sin ningún problema o error, ya que, como se mencionó anteriormente, este dispositivo si cuenta con uno de los requisitos más importantes que es la versión de Android mínima requerida. Tal como se aprecia en la siguiente figura.

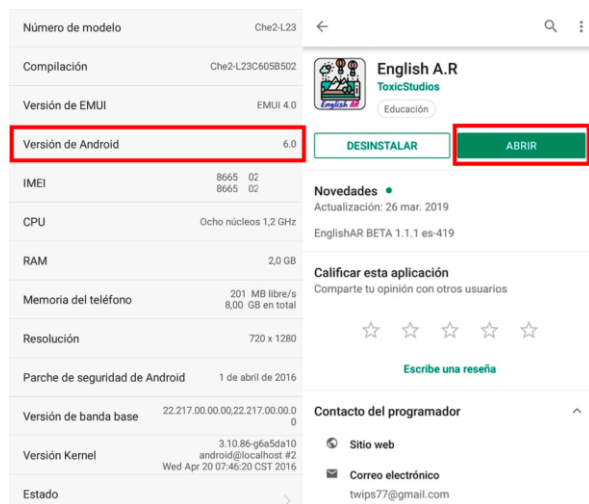


Figura 99. Prueba de compatibilidad App en dispositivo Huawei. Elaboración propia

En la parte izquierda de la figura 107, se puede observar el modelo del dispositivo y la versión de Android con la que cuenta el dispositivo, y en la parte derecha se observa que el proceso de instalación fue completado satisfactoriamente.

A diferencia del teléfono móvil Huawei, el dispositivo móvil Samsung no cuenta con el requisito de la versión mínima de Android ya que en su versión 4.4.2 Kit-Kat, al realizar la prueba de instalación de la aplicación desde la tienda de aplicaciones Google Play, apareció un error indicando que el dispositivo no era compatible con la aplicación. En la figura 108 se puede observar lo descrito anteriormente.

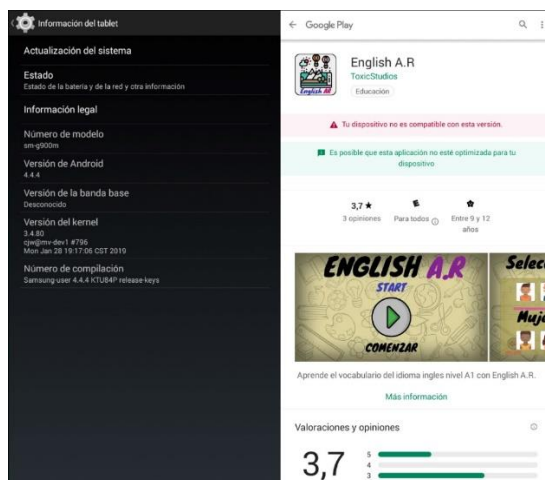


Figura 100. Prueba de incompatibilidad App en dispositivo Samsung S5. Elaboración propia.

En la parte izquierda de la figura anterior se observan las características del dispositivo Samsung S5, de las cuales se pueden observar el modelo y la versión de Android con la que cuenta, y en la parte derecha de la figura se observa el mensaje de incompatibilidad del dispositivo con la aplicación.

## CAPITULO 4

### 10. CONCLUSIONES

Con la elaboración de este proyecto se pudo evidenciar y a su vez poner en práctica los conocimientos y experiencias adquiridas durante el desarrollo de la carrera, lo cual deja como evidencia la entrega del producto final de este proyecto que es una aplicación móvil totalmente funcional y de ayuda para la población de menores que quieran enriquecer sus conocimientos en el dominio de una segunda lengua.

El software Unity permite crear aplicaciones móviles de una manera ágil y sencilla y si a esto se le agrega la fácil adaptación del SDK de Vuforia, se puede concluir que el desarrollo de un aplicativo con realidad aumentada y para dispositivos móviles no es algo complicado de realizar si se compara frente al uso de otras herramientas similares, además de que abre un mar de posibilidades para seguir realizando aplicaciones de este tipo usando estas tecnologías.

Contar con una metodología puede aportar muchas ventajas al desarrollo de un software, ya que esta, es la que se encarga de ordenar y definir los límites del desarrollo, además de planificar las tareas y procesos que se elaboraran a lo largo de la construcción de un proyecto. Dicho esto, se concluyó que el uso de la metodología Mobile-D en este proyecto fue de gran importancia ya que permitió establecer los requisitos del sistema, las restricciones, las reglas del negocio y definir las funcionalidades que debía poseer la aplicación, además de dividir la construcción del aplicativo en diferentes etapas las cuales dejaron como resultado, una elaboración organizada de cada uno de los componentes que integran a la aplicación.

Durante el desarrollo de este proyecto se obtuvieron nuevos conocimientos acerca de los tipos de tecnologías que tenían relación con este proyecto. Se adquirieron conceptos fundamentales para el desarrollo de software dentro de las limitaciones propias de un dispositivo móvil. También se obtuvieron conocimientos avanzados del lenguaje de programación C#, así como también conocimientos en el uso de las herramientas Unity, Vuforia, Blender y GIMP las cuales hicieron posibles el desarrollo de la aplicación para el sistema operativo Android.

## **11. RECOMENDACIONES**

Se recomienda desarrollar la aplicación de realidad aumentada para diferentes sistemas operativos móviles (iOS, Windows Phone), así como también realizar una adaptación de la aplicación para sistemas operativos de computadores (Windows, MacOS, Linux), con la finalidad de aumentar el público objetivo, además de poder brindarle a las personas más opciones de uso de la aplicación.

Realizar todas las mejoras que se vean o se crean pertinentes a la aplicación, ya sean mejoras de rendimiento, mejoras visuales, o adición de elementos que se consideren importantes y que hagan un aporte al objetivo principal de la aplicación.

## 12. PROYECCIONES

- Se desea agregar módulos de lecciones para distintas edades, así como también agregar más idiomas a la interfaz de usuario para que la aplicación sea asequible a un público internacional y que a su vez no se vea limitada a usuarios de habla hispana, lo cual deje como resultado un aplicativo más completo y competitivo en el mercado de aplicaciones.
- Por otro lado, que la aplicación sea implementada en colegios a nivel nacional y sea utilizada en diferentes grados de primaria y bachillerato, además de ser usada por la población de niños colombianos que cumplan o no con la edad objetivo planteada en este proyecto.

## REFERENCIAS

- Academic. (s.f). *Sintesis de habla*. Recuperado el 08 de Marzo de 2019, de <http://www.esacademic.com/dic.nsf/eswiki/1093223>
- Amazon. (2019). *Amazon Polly*. Recuperado el 12 de Enero de 2019, de <https://aws.amazon.com/es/polly/>
- Arzuza, J. (13 de Abril de 2011). *¿Que es Rigging?* Recuperado el 18 de Febrero de 2019, de <http://www.artzuza.com/2011/04/character-animation-technical-director.html>
- Baron, O. M. (2014). *REALIDAD AUMENTADA COMO ESTRATEGIA DIDÁCTICA EN CURSO DE CIENCIAS NATURALES DE ESTUDIANTES DE QUINTO GRADO DE PRIMARIA DE LA INSTITUCIÓN EDUCATIVA CAMPO VALDÉS*. Recuperado el 20 de Noviembre de 2018, de <https://repository.udem.edu.co/bitstream/handle/11407/1242/Realidad%20aumentada%20como%20estrategia%20did%C3%A1ctica%20en%20curso%20de%20ciencias%20naturales%20de%20estudiantes%20de%20quinto%20grado%20de%20primaria%20de%20la%20Instituci%C3%B3n%20Educativa>
- Barreira, J., Bessa, M., Pereira, L., Adão, T., Peres, E., & Magalhães, L. (31 de 08 de 2012). *MOW: Augmented Reality Game to Learn Words in*. Recuperado el 01 de Diciembre de 2018, de <https://ieeexplore.ieee.org/abstract/document/6263236>
- Betancourth, S. B. (29 de 10 de 2009). *Maestros del Web*. Recuperado el 25 de 02 de 2019, de *¿Qué es la realidad aumentada?:* <http://www.maestrosdelweb.com/que-es-realidad-aumentada/>
- Blanco, P., Camarero, J., Fumero, A., Werterski, A., & Rodriguez, P. (2009). *Metodología de desarrollo ágil para sistemas móviles Introducción al desarrollo con Android y el iPhone*. Recuperado el 20 de Agosto de 2018, de ResearchGate: [https://www.researchgate.net/publication/267795011\\_Metodologia\\_de\\_desarrollo\\_agil\\_para\\_sistemas\\_moviles\\_Introduccion\\_al\\_desarrollo\\_con\\_Android\\_y\\_el\\_iPhone](https://www.researchgate.net/publication/267795011_Metodologia_de_desarrollo_agil_para_sistemas_moviles_Introduccion_al_desarrollo_con_Android_y_el_iPhone)
- Blender. (s.f). *Manual de Blender*. Recuperado el 20 de Febrero de 2019, de [https://docs.blender.org/manual/es/dev/getting\\_started/about/introduction.html](https://docs.blender.org/manual/es/dev/getting_started/about/introduction.html)
- Caro, M. (12 de Febrero de 2014). *Phonic Tricksters, el juego de realidad aumentada de Pearson para aprender inglés, ya en Android*. Recuperado el 01 de Septiembre de 2018, de <http://applicantes.com/phonic-tricksters-juego-realidad-aumentada-de-pearson-para-aprender-ingles-android-2014/>
- Che Dalim, C., Dey, A., Piumsomboon, T., Billinghamurts, M., & Sunar, S. (2016). *TeachAR: An Interactive Augmented Reality Tool for Teaching Basic English to Non-Native Children*. Recuperado el 02 de Diciembre de 2018, de <https://ieeexplore.ieee.org/abstract/document/7836467>

- Costello, K., & Hippold Cornelia, S. (29 de Mayo de 2018). *Gartner Says Worldwide Sales of Smartphones Returned to Growth in First Quarter of 2018*. Recuperado el 31 de Octubre de 2018, de Gartner: <https://www.gartner.com/newsroom/id/3876865>
- Davidson J, S. (2004). *WIPO*. Recuperado el 24 de Febrero de 2019, de [https://www.wipo.int/export/sites/www/sme/es/documents/pdf/opensource\\_software\\_primer.pdf](https://www.wipo.int/export/sites/www/sme/es/documents/pdf/opensource_software_primer.pdf)
- D'Elia, M. (21 de 01 de 2018). *Dinno3D*. Recuperado el 01 de Noviembre de 2018, de Mapeado U.V en Blender: <https://dinno3d.wixsite.com/dinno3d/dinno3d/mapeado-uv-en-blender>
- Equipo Colombia Bilingüe. (2016). *Colombia Aprende*. Obtenido de Orientaciones y Principios Pedagógicos: Currículo Sugerido de Inglés: <http://aprende.colombiaaprende.edu.co/sites/default/files/naspublic/Anexo%2014%20Orientaciones%20y%20principios%20Pedagogicos.pdf>
- Espinosa, C. P. (01 de 2015). *REALIDAD AUMENTADA Y EDUCACIÓN: ANÁLISIS DE EXPERIENCIAS PRÁCTICAS*. Recuperado el 14 de Marzo de 2019, de <https://recyt.fecyt.es/index.php/pixel/article/viewFile/61619/37631>
- Federacion de enseñanza CC.OO. (02 de 05 de 2009). *Aprendizaje definicion, factores y clases. Temas para la educacion*. Recuperado el 19 de Febrero de 2019, de <https://www.feandalucia.ccoo.es/docu/p5sd4922.pdf>
- GIMP. (2018). *GIMP, UN POTENTE EDITOR DE IMÁGENES GRATUITO*. Recuperado el 06 de Marzo de 2019, de <https://gimp.es/>
- Goasduff, L., & Forni Ann, A. (15 de febrero de 2017). *gartner*. Recuperado el 01 de 05 de 2018, de <https://www.gartner.com/newsroom/id/3609817>
- González s.f., J. A. (s.f.). *Universidad de Murcia*. Recuperado el 06 de Mazo de 2019, de <http://dis.um.es/~bmoros/privado/bibliografia/LibroCsharp.pdf>
- ICFES Interactivo. (s.f.). *Reportes de porcentajes*. Recuperado el 31 de Octubre de 2018, de [http://www2.icfesinteractivo.gov.co/resultados/informes/snee\\_rep\\_por\\_ctr.jsp](http://www2.icfesinteractivo.gov.co/resultados/informes/snee_rep_por_ctr.jsp)
- Interactionstation. (12 de Enero de 2017). *Using Vuforia and Unity to make an AR app*. Obtenido de <http://interactionstation.wdka.hro.nl/mediawiki/images/thumb/b/b8/AR.png/400px-AR.png>
- ISO, O. I. (2010). *Online Browsing Platform*. Recuperado el 20 de Marzo de 2019, de <https://www.iso.org/obp/ui/#iso:std:iso:9241:-210:ed-1:v1:en>
- Jaguandoy Tobar, H. F., & Puchana Legarda, C. C. (2014). *Universidad de Nariño*. Recuperado el 25 de Noviembre de 2018, de <http://sired.udenar.edu.co/333/1/90234.pdf>
- Karlin, M. (07 de Noviembre de 2017). *Alive Studios: Reading and Math proficiency using Augmented Reality*. Recuperado el 01 de Septiembre de 2018, de

- <http://www.edtechroundup.org/reviews/alive-studios-reading-and-math-proficiency-using-augmented-reality>
- Lee, L.-K., Chau, C.-h., Chau, C.-H., & Ng, C.-T. (2017). *Using Augmented Reality to Teach Kindergarten Students English Vocabulary*. Recuperado el 03 de Diciembre de 2018, de <https://ieeexplore.ieee.org/abstract/document/8005387>
- Lemoine, R., Alfonso, M. C., Bohorquez, J. C., Maldonado, D., Sánchez, F., Cortés, D., . . . Velandia, J. (26 de Agosto de 2015). *EVALUACIÓN DE IMPACTO Y DE LA SOSTENIBILIDAD DE COMPUTADORES PARA EDUCAR EN LA CALIDAD DE LA EDUCACION EN LAS SEDES EDUCATIVAS BENEFICIADAS*. Recuperado el 28 de Octubre de 2018, de [http://www.computadoresparaeducar.gov.co/sites/default/files/inline-files/Articulo\\_impacto\\_Computadores\\_para\\_Educar\\_Colombia.pdf](http://www.computadoresparaeducar.gov.co/sites/default/files/inline-files/Articulo_impacto_Computadores_para_Educar_Colombia.pdf)
- Llisterri, J. (2001). *La conversion de texto en habla*. Recuperado el 12 de Marzo de 2019, de [http://liceu.uab.cat/~joaquim/publicacions/CTH\\_Quark\\_01.pdf](http://liceu.uab.cat/~joaquim/publicacions/CTH_Quark_01.pdf)
- López Maradiaga, J. J., Medina Yuen, B., & Octavo De La Cruz, A. A. (2018). *Ingles para niños con realidad aumentada "E.K.A.R"*. Recuperado el 09 de Marzo de 2019, de [https://www.academia.edu/36809015/Ingles\\_para\\_ni%C3%B1os\\_con\\_realidad\\_aumentada](https://www.academia.edu/36809015/Ingles_para_ni%C3%B1os_con_realidad_aumentada)
- Medina Velandia, L. N., & López López, W. M. (Diciembre de 2015). ESCOGER UNA METODOLOGÍA PARA DESARROLLAR SOFTWARE, DIFICIL DESICION. *Revista Educación en Ingeniería, 10(20)*, 11. Recuperado el 01 de Octubre de 2018, de <https://www.educacioneningenieria.org/index.php/edi/article/viewFile/579/275>
- Mejia, A.-M. T. (2005). Vivir en un mundo global. (Altablero, Entrevistador) Recuperado el 30 de Octubre de 2018, de <https://www.mineducacion.gov.co/1621/article-97500.html>
- Ministerio de Educación Nacional. (2006). *El reto!: Lo que necesitamos saber y saber hacer*. Recuperado el 31 de Octubre de 2018, de [https://www.mineducacion.gov.co/1621/articles-115174\\_archivo\\_pdf.pdf](https://www.mineducacion.gov.co/1621/articles-115174_archivo_pdf.pdf)
- Ministerio de Educacion Nacional. (2014). *Sistema nacional de indicadores educativos para los niveles de preescolar, basica y media en Colombia*. Recuperado el 30 de Octubre de 2018, de [https://www.mineducacion.gov.co/1621/articles-329021\\_archivo\\_pdf\\_indicadores\\_educativos\\_enero\\_2014.pdf](https://www.mineducacion.gov.co/1621/articles-329021_archivo_pdf_indicadores_educativos_enero_2014.pdf)
- Moncada, A. G. (2015). *¿nos han desplazado? ¿o hemos claudicado? el debilitado papel crítico de universidades públicas y los formadores de docentes en la implementación de la Política educativa lingüística del inglés en Colombia*. Recuperado el 30 de Octubre de 2018, de [https://www.researchgate.net/profile/Adriana\\_Gonzalez\\_Moncada2/publication/280836533\\_Nos\\_han\\_desplazado\\_O\\_hemos\\_claudicado\\_El\\_debilitado\\_papel\\_critico\\_de\\_universidades\\_publicas\\_y\\_los\\_formadores\\_de\\_docentes\\_en\\_la\\_implementacion\\_de\\_la\\_politica\\_educativa\\_lin](https://www.researchgate.net/profile/Adriana_Gonzalez_Moncada2/publication/280836533_Nos_han_desplazado_O_hemos_claudicado_El_debilitado_papel_critico_de_universidades_publicas_y_los_formadores_de_docentes_en_la_implementacion_de_la_politica_educativa_lin)

- MSN. (s.f.). *¿Qué es y para qué sirve Visual Studio 2017?* Obtenido de <https://www.msn.com/es-es/noticias/microsoftstore/%C2%BFqu%C3%A9-es-y-para-qu%C3%A9-sirve-visual-studio-2017/ar-AAAnLUC9>
- Olarte, A. C. (Junio de 2016). *La enseñanza del inglés en la escuela primaria: algunos problemas críticos*. Recuperado el 30 de Octubre de 2018, de <http://www.redalyc.org/pdf/3057/305745627001.pdf>
- Ortí s.f., C. B. (s.f.). *Unidad de Tecnología Educativa - Universidad de Valencia*. Recuperado el 05 de Marzo de 2019, de <https://www.uv.es/~bellohc/pdf/pwtic1.pdf>
- Oslen s.f., D. (s.f.). *Repositorio digital de la Facultad de Ingeniería - UNAM*. Recuperado el 04 de Marzo de 2019, de en <http://www.ptolomeo.unam.mx:8080/xmlui/bitstream/handle/132.248.52.100/210/A6%20Interfaces%20de%20Usuario.pdf?sequence=6>
- Ouazzani, I. (29 de 08 de 2012). *universidad Carlos III*. Recuperado el 06 de Marzo de 2019, de [http://netsites.mx/Unity3dTutorials/Unity\\_univ\\_carlos3.pdf](http://netsites.mx/Unity3dTutorials/Unity_univ_carlos3.pdf)
- Restrepo Duran, D., Cuello, S. L., & Contreras Chinchilla, L. (2017). *Juegos didácticos basados en realidad aumentada como apoyo en la enseñanza de biología*. Recuperado el 18 de Noviembre de 2018, de <http://documentos.redclara.net/bitstream/10786/1263/1/39-4-4Juegos%20did%C3%A1cticos%20basados%20en%20realidad%20aumentada%20como%20apoyo%20en%20la%20ense%C3%B1anza%20de%20biolog%C3%ADa.pdf>
- Rhinoceros. (s.f.). *¿Que son las NURBS?* Recuperado el 24 de Febrero de 2019, de <https://www.rhino3d.com/la/nurbs>
- Ruiz, M. F. (2011). *Universidad Carlos III de Madrid*. Recuperado el 04 de Marzo de 2019, de <https://core.ac.uk/download/pdf/30045016.pdf>
- Servicio de Sociedad de la Información. s.f. (s.f.). *Diputación de Cadiz*. Recuperado el 08 de Marzo de 2019, de en [https://www.dipucadiz.es/export/sites/default/galeria\\_de\\_ficheros/sociedad\\_de\\_la\\_informacion/destacados/Manual.CodigosQR.pdf](https://www.dipucadiz.es/export/sites/default/galeria_de_ficheros/sociedad_de_la_informacion/destacados/Manual.CodigosQR.pdf) s.f.
- Sistema Universidad Abierta y Educación a Distancia de la UNAM s.f. (s.f.). *SUAYEDFCA*. Recuperado el 22 de 02 de 2019, de [http://fcasua.contad.unam.mx/apuntes/interiores/docs/98/4/informatica\\_4.pdf](http://fcasua.contad.unam.mx/apuntes/interiores/docs/98/4/informatica_4.pdf)
- Sistemas Multimedia e Interacción Gráfica. (2006). *Universidad de Murcia*. Obtenido de <http://dis.um.es/~jfernand/0506/smig/gimp.pdf>
- Unity Technologies. (2017). *Unity Tutorials*. Recuperado el 20 de Marzo de 2019, de <https://unity3d.com/es/learn/tutorials/topics/mobile-touch/building-your-unity-game-android-device-testing>
- Unity Technologies. (2018). *Canvas*. Recuperado el 07 de Marzo de 2019, de <https://docs.unity3d.com/es/current/Manual/UICanvas.html>

- Unity Technologies. (2018). *Unity Documentation*. Recuperado el 07 de Marzo de 2019, de <https://docs.unity3d.com/es/current/Manual/CreatingAndUsingScripts.html>
- Unity Technologies. (2018). *Unity Documentation*. Recuperado el 12 de Marzo de 2019, de <https://docs.unity3d.com/es/2018.1/Manual/vuforia-sdk-overview.html>
- Unity Technologies. (2019). *Unity* . Recuperado el 10 de 9 de 2018, de <https://store.unity.com/es>
- Universidad de Cundinamarca. (29 de Mayo de 2003). *LINEAS DE INVESTIGACION*. Recuperado el 01 de Septiembre de 2018, de <http://ingenieria.ucundinamarca.edu.co/index.php/investigacion/lineas-de-investigacion>
- W3C. (07 de Septiembre de 2010). *W3C Recomendaton*. Recuperado el 29 de Noviembre de 2018, de <https://www.w3.org/TR/2010/REC-speech-synthesis11-20100907/>
- Zenith. (01 de 08 de 2014). *Bloggin Zenith*. Recuperado el 12 de Marzo de 2019, de <https://blogginzenith.zenithmedia.es/que-es-y-como-funciona-la-realidad-aumentada-diccionario/>