



Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación  
Centro de Computación Gráfica

**Construcción y visualización de  
libros infantiles empleando  
realidad aumentada**

Trabajo Especial de Grado  
presentado ante la Ilustre  
Universidad Central de Venezuela  
Por el Bachiller

**Alejandro Antonio Acosta Tartaret C.I. 23.691.732**

para optar por el título de  
Licenciado en Computación

Tutor:  
Prof. Esmitt Ramirez



**Universidad Central de Venezuela**  
Facultad de Ciencias  
Escuela de Computación  
Centro de Computación Gráfica



**ACTA DE VEREDICTO**

Quienes suscriben, miembros del jurado designado por el Consejo de Escuela de Computación, para examinar el Trabajo Especial de Grado presentado por el Bachiller Alejandro Antonio Acosta Tartaret C.I. V-23.691.732, con el título "**Construcción y visualización de libros infantiles empleando realidad aumentada**", a los fines de cumplir con el requisito legal para optar al título de Licenciado en Computación, dejan constancia de lo siguiente:

Leído el nombrado trabajo por cada uno de los miembros del jurado, éste fijó el día lunes 22 de Mayo de 2017, para que su autor lo defendiera en forma pública, en el Centro de Computación Gráfica a las 11 am. de la Escuela de Computación, mediante una exposición oral de su contenido, luego de la cual respondió satisfactoriamente a las preguntas que le fueron formuladas por el jurado, todo ello conforme a lo dispuesto en la Ley de Universidades y demás normativas vigentes de la Universidad Central de Venezuela.

Finalizada la defensa pública del Trabajo Especial de Grado, el jurado decidió aprobarlo.

Prof. Esmitt Ramírez  
Tutor

Prof. Francisco Sans  
Jurado Principal

Profa. Mercy Ospina  
Jurado Principal



Caracas, Mayo de 2017

Universidad Central de Venezuela  
Facultad de Ciencias  
Escuela de Computación  
Centro de Computación Gráfica - CCG

## **Construcción y visualización de libros infantiles empleando realidad aumentada**

**Autor:** Alejandro Antonio Acosta Tartaret  
C.I. 23.691.732  
alejandro.aacostat@gmail.com

**Tutor:** Prof. Esmitt Ramírez  
esmitt.ramirez@ciens.ucv.ve

**Fecha:** 22 de mayo del 2017

### **Resumen**

Este Trabajo Especial de Grado se hizo con la finalidad de proveer, a escritores y editoriales, una herramienta que permita la creación de libros con realidad aumentada y la visualización de su contenido virtual, de manera sencilla. Este Trabajo Especial de Grado consiste en el desarrollo de un sistema que permitirá la creación de libros infantiles con realidad aumentada, basada en una aplicación web desarrollada con Angularjs, una aplicación móvil desarrollada con Android Studio y un *backend* desarrollado con Node.js y express. La aplicación web permitirá la creación, búsqueda y descarga de los libros, mientras que la aplicación móvil permitirá la visualización del contenido virtual en realidad aumentada utilizando la cámara del dispositivo.

**Palabras Claves:** Libros infantiles, realidad aumentada, aplicación web, aplicación móvil, *backend*, AngularJS, nodeJS, express, Android, Android Studio, Kudan.

# Tabla de Contenidos

<b>Introducción</b>	<b>IX</b>
<b>1. Planteamiento del problema</b>	<b>1</b>
1.1. Justificación . . . . .	1
1.2. Objetivos . . . . .	2
1.2.1. Objetivo general . . . . .	2
1.2.2. Objetivos específicos . . . . .	2
<b>2. Marco teórico</b>	<b>4</b>
2.1. Realidad Virtual . . . . .	4
2.2. Clasificación de la realidad virtual . . . . .	4
2.2.1. Sistemas inmersivos . . . . .	4
2.2.2. Sistemas no inmersivos . . . . .	4
2.3. Realidad Aumentada . . . . .	5
2.4. Historia y antecedentes . . . . .	5
2.5. Funcionamiento . . . . .	6
2.5.1. Reconocimiento por marcadores . . . . .	6
2.5.2. Reconocimiento basado en objetos . . . . .	6
2.5.3. Reconocimiento facial . . . . .	6
2.5.4. Reconocimiento de movimiento . . . . .	8
2.5.5. Reconocimiento basado en geolocalización . . . . .	8
2.6. Usos de la realidad aumentada . . . . .	9
2.6.1. Publicidad . . . . .	10
2.6.2. Turismo . . . . .	11
2.6.3. Educación . . . . .	12
2.6.4. Entretenimiento . . . . .	13
2.7. Entornos de Desarrollo . . . . .	13
2.7.1. SDK's para desarrollar con realidad aumentada . . . . .	13
2.7.2. Aplicaciones basadas en Android . . . . .	14
2.7.3. Android Studio . . . . .	15
2.7.4. Modelado de las aplicaciones . . . . .	17
2.7.5. Metodología utilizada . . . . .	18

<b>3. Solución planteada</b>	<b>19</b>
3.1. Diseño de la solución . . . . .	19
3.1.1. <i>Backend</i> . . . . .	19
3.1.2. Aplicación web . . . . .	20
3.1.3. Aplicación móvil . . . . .	22
3.2. Desarrollo de la solución . . . . .	23
3.2.1. Desarrollo del <i>backend</i> . . . . .	25
3.2.2. Desarrollo de la aplicación web . . . . .	28
3.2.3. Desarrollo de la aplicación móvil . . . . .	36
<b>4. Pruebas y Resultados</b>	<b>40</b>
4.1. Pruebas cuantitativas (Aplicación móvil) . . . . .	40
4.1.1. Ambiente de pruebas . . . . .	40
4.1.2. Casos de estudio . . . . .	41
4.1.3. Resultados . . . . .	42
4.2. Prueba cualitativa (Aplicación web) . . . . .	43
4.2.1. Resultados . . . . .	46
<b>5. Conclusiones y trabajos futuros</b>	<b>48</b>
5.1. Conclusiones . . . . .	48
5.2. Limitaciones . . . . .	49
5.3. Trabajos a futuro . . . . .	49

# Lista de Figuras

1.	Reconocimiento por Marcadores . . . . .	7
2.	Reconocimiento basado en objetos . . . . .	7
3.	Reconocimiento Facial . . . . .	8
4.	Reconocimiento de Movimiento . . . . .	9
5.	Reconocimiento Basado en Geolocalización . . . . .	9
6.	Realidad aumentada utilizada para publicidad . . . . .	10
7.	Realidad aumentada utilizada en turismo, aplicación Skin & Bones . . . . .	11
8.	Realidad aumentada utilizada en turismo, aplicación Word Lens . . . . .	12
9.	Arquitectura de Android . . . . .	16
10.	Android Studio . . . . .	17
11.	Diseño del <i>backend</i> . . . . .	20
12.	Diseño de la aplicación web . . . . .	21
13.	Diseño de la aplicación móvil . . . . .	22
14.	Diagrama de clases del <i>backend</i> . . . . .	25
15.	Interfaz <i>home</i> de la aplicación web. . . . .	30
16.	Interfaz <i>My Desktop</i> de la aplicación web. . . . .	30
17.	Interfaz <i>My Desktop</i> con la tabla de Libros. . . . .	31
18.	Interfaz <i>Library</i> con lista de Libros. . . . .	31
19.	Interfaz con detalle de un Libro. . . . .	32
20.	Barra de navegación de la interfaz de la herramienta. . . . .	32
21.	Menú de selección de la interfaz de la herramienta. . . . .	33
22.	Interfaz de la herramienta con la selección de texto. . . . .	33
23.	Interfaz de la herramienta con la selección de título. . . . .	34
24.	Interfaz de la herramienta con la selección de imagen. . . . .	35
25.	Interfaz de la herramienta con la selección de título y texto. . . . .	35
26.	Interfaz de la herramienta con la selección de título e imagen. . . . .	36
27.	Actividad de Inicio de sesión. . . . .	37
28.	Actividad de Creación de usuario. . . . .	37
29.	Actividad de Perfil de usuario. . . . .	38
30.	Actividad de Actulización de usuario. . . . .	38
31.	Actividad de Librería. . . . .	38
32.	Actividad de Detalle del Libro. . . . .	38

33.	Actividad de Biblioteca. . . . .	39
34.	Actividad de Realidad Aumentada. . . . .	39
35.	Marcador usado en las pruebas de control . . . . .	41
36.	Resultado de la encuesta sobre las funcionalidades básicas del sistema . . . .	46
37.	Resultado de la encuesta sobre las funcionalidades de creación del libro . . .	47
38.	Resultado de la encuesta sobre la consideración sobre la existencia de este tipo de aplicaciones . . . . .	47

# Lista de Tablas

1.	Plan de trabajo para el desarrollo de la solución . . . . .	24
2.	Métodos del modelo Usuario. . . . .	26
3.	Métodos del modelo Sesión. . . . .	27
4.	Métodos del modelo Libro. . . . .	27
5.	Métodos del contenido y publicación del Libro. . . . .	28
6.	Casos de estudio. . . . .	41
7.	Resultados del tiempo medido de los casos base. . . . .	42
8.	Resultados del tiempo medido de los casos de estudio. . . . .	42
9.	Resultados de la memoria medido de los casos base. . . . .	42
10.	Resultados de la memoria medido de los casos de estudio. . . . .	43
11.	Resultados del CPU medido de los casos base. . . . .	43
12.	Resultados del CPU medido de los casos de estudio. . . . .	43
13.	Ejemplo de encuesta . . . . .	45



# Introducción

La lectura en los niños es un hábito que, en esta época de videojuegos, *tablets* y teléfonos inteligentes se ve muy amenazada. Los libros en físico no se hacen atractivos para sus pequeños lectores cuando compiten con lo que, a la opinión de muchos de ellos, se hace divertido poder jugar mientras se lee en algún dispositivo electrónico.

Un libro físico que involucre a la tecnología, de manera que pueda traer elementos digitales al mundo real utilizando dispositivos inteligentes, como un libro con realidad aumentada, ha de permitir que se reactive el interés sincero de los niños a la lectura, pues además de contener escritos e imágenes impresas para llamar su atención, poseen contenido virtual que pueden ver a través de esos dispositivos que son sus acompañantes a la hora de jugar. Imágenes o animaciones 3D, videos y audios que pueden reproducir mientras leen, a través de la cámara de un tablet o de un teléfono inteligente.

Ahora bien, esto es un reto importante para los escritores, pues no es sencillo incluir contenido virtual a un libro. Para ello se requiere conocimientos de programación (como para añadir contenido virtual a un objeto físico utilizando algún conjunto de herramientas de desarrollo de software) y de diseño o animación 3D, además de tener que desarrollar la aplicación que permita interactuar con ese contenido. En otras palabras, se añaden costos adicionales y mayor tiempo para la publicación de un libro con realidad aumentada.

Este Trabajo Especial de Grado consiste en el desarrollo de un sistema que permitirá la creación de libros infantiles con realidad aumentada, basada en una aplicación web y una aplicación móvil, para crear e interactuar respectivamente, con el contenido virtual que estará incluido en las páginas de los libros creados por un escritor que utilice esta herramienta de trabajo.

Su contenido está dividido en cinco capítulos, donde en el primero se plantea el problema a ser desarrollado, en el segundo se enuncia el Marco Teórico en el que fue basada la solución que se desarrolló, en el tercero se explica la solución planteada, incluyendo el diseño de las aplicaciones y el desarrollo de las mismas, el cuarto describe el ambiente de pruebas y se presentan las pruebas, tanto las cuantitativas como cualitativas, que fueron realizadas en este trabajo y en el quinto y último capítulo, se presentan las conclusiones y los trabajos a futuro.

# Capítulo 1

## Planteamiento del problema

En la revisión de antecedentes realizada, no se encontró un software disponible en el mercado, que permita directamente a escritores, crear libros donde puedan incluir realidad aumentada y a sus lectores interactuar con el contenido digital incluidos en ellos, de manera sencilla.

Realizar un libro de este estilo no es una tarea fácil. Añadir realidad aumentada a un proyecto requiere de conocimientos en programación para añadir contenido virtual a un objeto físico, utilizando algún SDK<sup>1</sup>, y para crear una aplicación que sea capaz de interactuar con dicho contenido. Además son necesarios conocimientos en diseño o animación 3D, para crear contenido digital original, lo que conlleva costos y tiempo adicional.

### 1.1. Justificación

Debido al rápido progreso de la tecnología en los últimos años, las generaciones más jóvenes son atraídas a dispositivos electrónicos como el medio de entretenimiento y aprendizaje de su preferencia, dejando a los libros a un lado y utilizándolos sólo en sus salones de clase o en sus casas, mientras hacen sus labores escolares. Por este motivo algunas editoriales han utilizado la realidad aumentada en sus libros, creando una nueva manera de experimentar y fomentar la lectura en niños y jóvenes, mezclando los libros con la tecnología. Por ejemplo las compañías de Books & Magic[1] y PopAR[2] que combinan libros físicos con aplicaciones digitales para ofrecer un nuevo tipo de libro que inspire a niños a leer utilizando la realidad aumentada.

En la actualidad, las compañías de tecnología, en especial aquellas dedicadas al medio del entretenimiento, están invirtiendo en la realidad virtual y la realidad aumentada para que sea accesible a la mayor cantidad de usuarios posibles, creando desde *headsets* como cascos o lentes, hasta *software* (aplicaciones) para teléfonos o juegos para las consolas de videojuegos. Como ejemplo se encuentra la exitosa aplicación móvil de Pokémon Go [3] que utiliza la

---

<sup>1</sup>SDK: Kit de desarrollo de software por sus siglas en inglés

cámara de dispositivos móviles y la realidad aumentada para mostrar animaciones de objetos 3D en el mundo real.

En este trabajo se quiere aprovechar el interés y popularidad de esta tecnología, para ofrecer a editoriales y escritores una manera sencilla de crear libros con realidad aumentada, reduciendo el tiempo y costo de creación a todos los que deseen escribir sus libros, incluyéndole a éstos contenidos audiovisuales (imágenes 2D, objetos 3D, animaciones, videos y audio).

La elaboración del sistema permitirá a autores y escritores escribir sus libros añadiéndoles contenido digital de manera sencilla, brindando nuevas experiencias a sus lectores y atrayendo a nuevos. Según el punto de vista de los lectores, los niños y jóvenes se verán atraídos a este tipo de libros que combinan la lectura y el aprendizaje con la tecnología a la que están familiarizados, como son los *smartphones* y las *tablets*, haciendo de la lectura una manera divertida de aprender y distraerse.

## 1.2. Objetivos

### 1.2.1. Objetivo general

Desarrollar un sistema para la creación de libros infantiles empleando realidad aumentada, que incluye una aplicación para crear, visualizar y editar los libros y una aplicación móvil para poder visualizar e interactuar con el contenido virtual.

### 1.2.2. Objetivos específicos

- Crear un *backend* para el API de las aplicaciones web y móvil del sistema.
- Implementar en el *frontend* de la aplicación web un módulo de creación y autenticación de usuarios.
- Crear en el *frontend* de la aplicación web un módulo de creación y edición de libros con realidad aumentada.
- Implementar en el *frontend* de la aplicación web un módulo de visualización y descarga de libros creados por los usuarios.
- Crear en el *frontend* de la aplicación móvil un módulo para la creación y autenticación de usuarios.
- Desarrollar en la aplicación móvil un módulo de librería, donde el usuario pueda ver los libros disponibles, sus detalles y pueda descargar el contenido digital asociado a su teléfono.

- Implementar en la aplicación móvil el módulo de biblioteca, donde el usuario pueda seleccionar uno de los libros cuyo contenido ya ha sido descargado y visualizarlo mediante la cámara del dispositivo.
- Desarrollar en la aplicación móvil el módulo de perfil de usuario, donde el usuario pueda ver y actualizar sus datos personales.

# Capítulo 2

## Marco teórico

### 2.1. Realidad Virtual

Según Levis [4], se puede definir a la realidad virtual como una base de datos interactiva capaz de crear una simulación que implique a todos los sentidos, generada por un computador, explorable, visualizable y manipulable en tiempo real bajo la forma de imágenes y sonidos digitales, dando la sensación de presencia en el entorno informático.

### 2.2. Clasificación de la realidad virtual

Se entiende por tipos de realidad virtual a las técnicas utilizadas para la implementación que poseen elementos y rasgos en común, y pueden ser categorizados en dos tipos que se describen a continuación.

#### 2.2.1. Sistemas inmersivos

Estos sistemas se caracterizan por el uso de accesorios como guantes, visores o cascos que dan al usuario la sensación de inmersión completa al mundo virtual. La mayoría de los sistemas actuales se centran en los sentidos de la vista y el oído, debido a las dificultades y costo de simular los otros, utilizando hardware especializado para detectar el movimiento del usuario y proyectar la información del mundo virtual de manera tridimensional.

Estos sistemas son los más costosos a nivel tecnológico y de rendimiento gráfico además del monetario para lograr un nivel satisfactorio de realismo.

#### 2.2.2. Sistemas no inmersivos

Como el nombre sugiere, esta técnica es la implementación menos inmersiva de las técnicas de realidad virtual. El ambiente virtual es visto a través de un monitor y la interacción con este ambiente ocurre a través de periféricos como por ejemplo el mouse y teclado de una computadora o algún hardware empleado en los juegos de consola (e.g. wii mote, play station

move, y otros).

Estos sistemas no requieren de un alto rendimiento gráfico, o ningún hardware especial y pueden ser utilizados en muchos tipos de computadoras. Sin embargo el bajo costo significa que estos sistemas serán superados por otros con implementaciones más sofisticadas debido al bajo nivel de inmersión que provee.

### **2.3. Realidad Aumentada**

La realidad aumentada forma parte de la realidad virtual y se obtiene al tomar información digital o generada por computadora, pudiendo ser imágenes, audio, video y tacto, y superponerla sobre el entorno físico en tiempo real. Esta permite al usuario ver el mundo real, con objetos virtuales superpuestos o en composición con el mundo real.

De acuerdo con Kipper y Rampolla [5] las siguientes tres características definen a la realidad aumentada.

1. La realidad aumentada combina información real y virtual.
2. La realidad aumentada es interactiva en tiempo real.
3. La realidad aumentada opera y es utilizada en un ambiente tridimensional.

### **2.4. Historia y antecedentes**

A lo largo de la historia han existido muchas personas talentosas que se dedicaron a hacer grandes cosas con realidad aumentada, sin contar con el avance tecnológico actual. Uno de ellos fue Ivan Sutherland quien crea el primer sistema de realidad aumentada y realidad virtual llamado The Sword of Damocles [6], el primer sistema con un monitor que iba colocado en la cabeza del usuario. El sistema utilizaba gráficos generados por computadora para mostrar las líneas y vértices de modelos tridimensionales simples.

Desde entonces han habido avances en la realidad aumentada, como un sistema móvil y multiusuario, que permitía combinar la realidad aumentada móvil y las capacidades colaborativas entre usuarios en un espacio compartido [7]. También se creó un sistema de realidad aumentada para turismo y educación llamado Archeoguide [8], construido en el sitio histórico de Olimpia, Grecia. Este sistema contenía una interfaz de navegación, modelos 3D de templos y estatuas antiguas y avatares compitiendo en una carrera.

En los años recientes grandes empresas como Google, Facebook y Microsoft han mostrado interés en la realidad aumentada buscando y creando proyectos como los HoloLens de Microsoft[9], lentes que permiten a los usuarios que se los pongan visualizar objetos virtuales por encima de los objetos reales.

## 2.5. Funcionamiento

En un sistema de realidad aumentada la tarea principal es identificar el escenario para que pueda ser procesado y luego añadir la información del mundo virtual. Para lograr esto se necesitan dispositivos capaces de detectar el escenario del mundo real, estos dispositivos pueden ser agrupados en dos conjuntos.

1. Dispositivos *video-through*, dentro de este grupo se encuentran los dispositivos de captura de video que están separados de los dispositivos de visualización. En este conjunto se encuentran las cámaras de video y cámaras web.
2. Dispositivos *see-through*, estos dispositivos se encargan de capturar la escena real y mostrar la información aumentada. Una característica importante de estos dispositivos es que trabajan en tiempo real. Dentro de este conjunto se encuentran dispositivos como *smartphones*, tablets y algunos cascos como los utilizados en aviones de combate para mostrar información al piloto.

En el proceso de identificar la escena es necesario encontrar el escenario físico que se quiere aumentar con la información virtual, esto puede lograrse utilizando alguno de los métodos que se presentan a continuación.

### 2.5.1. Reconocimiento por marcadores

En un sistema de realidad aumentada, un marcador es un objeto cuya imagen ya conocida por el sistema. El marcador es recibido y reconocido a través de la cámara, gracias a esto se consigue la posición en el mundo real donde se podrá desplegar el contenido virtual, como podemos ver en la Fig. 1.

Para el reconocimiento de marcadores se utiliza un primer escaneo sobre la imagen, que presenta un alto costo de cómputo, para localizar el marcador que se busca. Una vez localizado el marcador, se calcula la posición y la dirección donde desplegará el contenido según la distancia del marcador y el ángulo que se encuentra respecto a la cámara.

### 2.5.2. Reconocimiento basado en objetos

Este método es más complejo que el anterior a nivel de implementación y conlleva a un mayor costo de cómputo. En lugar de reconocer una imagen, el sistema debe reconocer uno o varios objetos, haciendo una comparación con una base de datos que los clasifica según su forma. Un ejemplo de este método puede observarse en la Fig. 2.

### 2.5.3. Reconocimiento facial

Se utilizan algoritmos capaces de reconocer el rostro de una persona y se utiliza este como el marcador para sobreponer el objeto virtual al momento en que ocurre el reconocimiento.



Figura 1: Reconocimiento por marcadores.  
Imagen extraída de [10].



Figura 2: Reconocimiento basado en objetos.  
Imagen extraída de [11].

Un ejemplo de esto es la conocida aplicación Snapchat [12], una red social que permite al usuario publicar fotos y videos cortos, y compartirlos con sus seguidores, la cual provee distintas opciones llamadas *Snapchat Lenses* que cambian el aspecto del usuario como se aprecia en la Fig. 3.



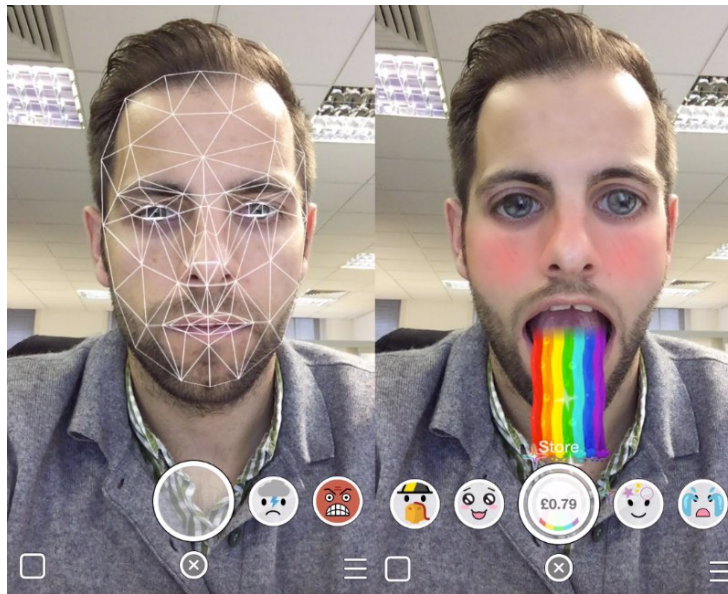


Figura 3: Reconocimiento facial.  
Imagen extraída de [13].

#### 2.5.4. Reconocimiento de movimiento

Este tipo de reconocimiento utiliza algoritmos para detectar ciertos puntos del cuerpo como las manos, pies, el tronco y la cabeza, y detectar el movimiento de la persona. Un sistema conocido por utilizar este tipo de reconocimiento fue el dispositivo Kinect de Microsoft [14] que utiliza hardware especializado para detectar el movimiento del usuario como puede verse en la Fig. 4.

#### 2.5.5. Reconocimiento basado en geolocalización

Este reconocimiento requiere de un sistema GPS con sensores capaces de reconocer la orientación del dispositivo, como brújulas digitales, acelerómetros, entre otros. Usando puntos de referencia como coordenadas, el dispositivo puede aproximar el objeto virtual con el real tomando en cuenta el ángulo de visión y su distancia.

Gracias a los *smartphones* que cuentan con el hardware antes mencionado, cada día existen nuevas aplicaciones desarrolladas basados en este tipo de reconocimiento como Waze [16], una aplicación que trabaja en tiempo real y utiliza el sistema GPS del smartphone y a los usuarios para detectar lo que ocurre en las carreteras y calles. Otro ejemplo puede ser la aplicación de Pokémon Go [3] de la compañía Niantic, que utiliza la posición del usuario y al mundo real para crear un mundo virtual, como puede verse en la Fig. 5.



Figura 4: Reconocimiento de movimiento.  
Imagen extraída de [15].



Figura 5: Reconocimiento basado en geolocalización.  
Imagen extraída de [17].

## 2.6. Usos de la realidad aumentada

La realidad aumentada posee un amplio rango de aplicaciones en diversos campos y gracias al creciente uso de dispositivos inteligentes y el avance de la tecnología ahora está al alcance de la mayoría de las personas. A continuación se describen algunos campos donde la

realidad aumentada puede ser y está siendo utilizada en la actualidad.

### 2.6.1. Publicidad

Cada día nuevas marcas utilizan la disponibilidad de los teléfonos celulares para integrar la realidad aumentada con sus campañas publicitarias. Compañías de automóviles como Nissan, Toyota, BMW, entre otras, utilizan la realidad aumentada como medio publicitario en revistas, ofreciéndole al usuario el modelo 3D de sus vehículos. La empresa de construcción de bloques de juguetes LEGO ofrece en alguna de sus tiendas un sistema de realidad aumentada, el cual permite ver la versión completa del *set* de Lego que está dentro de la caja que sostienen sus visitantes. Incluso la industria del cine ha tomado ventaja de esta tecnología para promocionar películas como Transformers, Iron Man y Star Trek.

La realidad aumentada puede o no representar mucho al consumidor común, pues esta tecnología por sí misma no crea una conexión con éste. Por ello la meta principal de una campaña publicitaria que utilice la realidad aumentada es crear una experiencia interactiva que atraiga a los clientes. Se puede ver un ejemplo de esto en la Fig. 6, una campaña publicitaria de *National Geographic* donde utilizan la realidad aumentada para que las personas interactúen con diversas animaciones.



Figura 6: Realidad aumentada utilizada para publicidad.  
Imagen extraída de [18].

### 2.6.2. Turismo

Con el lanzamiento de dispositivos *smartphone*, la realidad aumentada se ha convertido en el nuevo auge tecnológico, sin embargo, aún no se ha explotado su potencial en el turismo. A continuación se presentaran algunos ejemplos donde se muestra el potencial que esta tecnología posee para revolucionar la manera como se experimentan nuevos destinos y servicios dentro de la industria.

Las visitas educacionales son unos de los factores más importantes de la industria turística, y la realidad aumentada tiene un gran potencial para la participación activa de los turistas por conocer y experimentar a los museos y sus artefactos. Recientemente se han aplicado técnicas de narración digital interactiva a museos para incrementar su potencial educativo. Instalaciones con realidad aumentada en los museos pueden simular experiencias que de otra manera serían imposibles, ya que con estas instalaciones podría ser posible revivir especies animales ya extintas, o artefactos históricos desgastados. Como ejemplo en la Fig. 7 se observa la aplicación *Skin & Bones* [19] del Museo Nacional Smithsonian de Historia Natural [20], que ofrece una manera nueva de observar su exhibición de esqueletos.

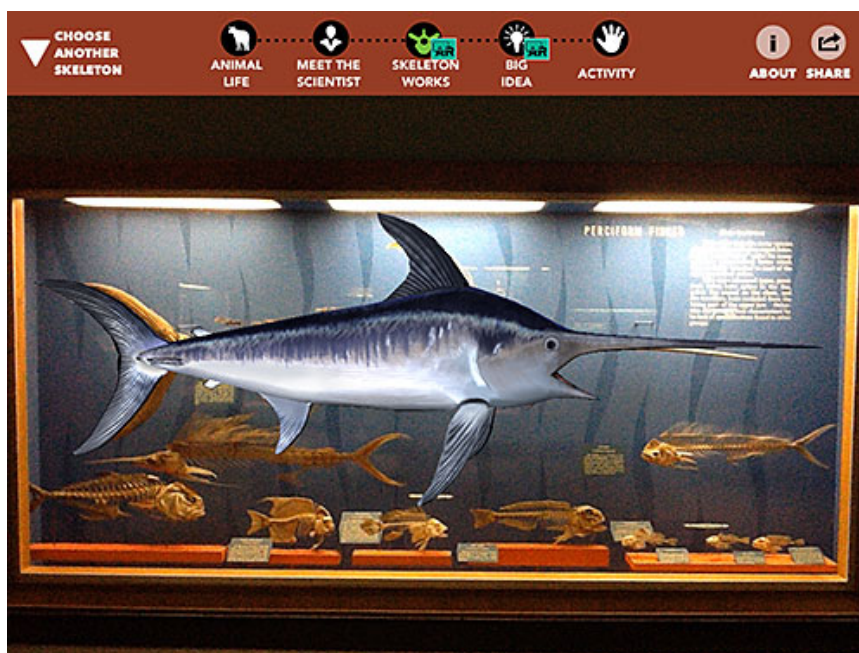


Figura 7: Realidad aumentada utilizada en turismo, aplicación Skin & Bones. Imagen extraída de [19].

La exploración de nuevos lugares puede ser influenciada por la falta o mala traducción de carteles, señales o instrucciones. Las aplicaciones de realidad aumentada pueden proveer de traducción inmediata en tiempo real de textos escritos en menús, horarios de trenes e incluso titulares de periódicos de una lengua extranjera a la lengua nativa del usuario. Un ejemplo de

esto es la aplicación *Word Lens* como puede verse en la Fig. 8.



Figura 8: Realidad aumentada utilizada en turismo, aplicación Word Lens. Imagen extraída de [21].

### 2.6.3. Educación

La tecnología está cambiando la educación de muchas maneras, empezando por la cantidad de información disponible y que puede ser accedida a través del Internet, hasta métodos de aprendizaje en línea e interfaces interactivas como las pizarras inteligentes.

La realidad aumentada se apega a la creencia que los estudiantes aprenden más cuando ellos toman el control de su propio aprendizaje e interactúan con los ambientes reales y virtuales. Con la realidad aumentada los estudiantes pueden manipular objetos virtuales o representaciones de objetos reales que serían imposibles de sostener. El beneficio de utilizar la realidad aumentada en el aprendizaje es que no hay errores reales, sólo nuevas experiencias de las cuales aprender. La realidad aumentada también tiene un gran potencial para la educación avanzada y el mantener un conjunto de habilidades. Por ejemplo, mecánicos podrían aprender a reparar nuevas piezas de un equipo, un bombero pudiese aprender como combatir un incendio en un escenario específico sin que haya ninguna consecuencia real si ocurren errores.

## 2.6.4. Entretenimiento

El entretenimiento fue uno de los primeros usos de la realidad aumentada, y en los últimos años ha sido utilizada principalmente en el mercado de los videojuegos. La compañía Sony integró la realidad aumentada a su consola de videojuegos *PlayStation 3* en el año 2010 con el *PlayStation Move*[22] que incluía una cámara y un control que el usuario podía utilizar para interactuar con objetos virtuales.

Por otra parte la compañía Microsoft lanzó el *Kinect* unos meses después del mismo año, este dispositivo reconocía los movimientos y ciertos gestos del usuario para interactuar y controlar la consola de videojuegos *Xbox 360*, sin necesidad de un control. Desde entonces, la realidad aumentada ha sido utilizada en videojuegos tanto en consolas, portátiles y de escritorio, como en teléfonos celulares.

Además de videojuegos, la realidad aumentada está empezando a tener un impacto en la manera como el arte es creado y mostrado. Una aplicación móvil llamada *Konstruct* permite al usuario generar obras de arte en un ambiente de realidad aumentada. En el 2010, el Museo de Arte Moderno de Nueva York [23] fue sede de una exhibición que permitía a visitantes con teléfonos iPhone o Android ver obras de arte escondidas en una exhibición de realidad aumentada.

La realidad aumentada es una tecnología emergente que se ha popularizado en los últimos años, y está siendo utilizada en diferentes ámbitos, a través de los avances tecnológicos actuales para brindar una manera de vivir la realidad con nuevas experiencias. En el siguiente capítulo se describirán algunos entornos de desarrollo que se utilizan en la creación de estas experiencias.

## 2.7. Entornos de Desarrollo

### 2.7.1. SDK's para desarrollar con realidad aumentada

El desarrollo de aplicaciones de realidad aumentada se puede realizar a través de varios kit de desarrollo de software o SDK. Un SDK es un conjunto de herramientas de desarrollo que le permite al programador crear aplicaciones.

Puede ser algo tan sencillo como una interfaz de programación de aplicaciones o API, creada para permitir el uso de cierto lenguaje de programación, o puede también, incluir hardware sofisticado para comunicarse con un determinado sistema.

Las herramientas más comunes incluyen soporte para la detección de errores de programación como un entorno de desarrollo integrado o IDE y otras utilidades. Los SDK frecuentemente incluyen códigos de ejemplo y notas técnicas de soporte u otra documentación para

ayudar a clarificar ciertos puntos del material de referencia primario.

La realidad Aumentada no puede ser programada en los entornos conocidos de aplicaciones móviles (XCode para iOS y Eclipse para Android, por ejemplo) sin un SDK o un conjunto de bibliotecas propias, desarrolladas específicamente para tal fin.

En la actualidad existen diversos SDK's con los que se pueden desarrollar aplicaciones con realidad aumentada. Entre ellas tenemos a Vuforia[24], ARToolKit[25], Wikitude[26], AR-media[27] y Kudan[28], cinco de las herramientas más utilizadas y exitosas para realidad aumentada que existen en este momento en el mercado mundial y que son utilizadas para desarrollar aplicaciones para *smartphones*, tablets o incluso *smart-glasses*.

En este documento se detallará exclusivamente el SDK de Kudan, el cual fue utilizado para la realización de este proyecto y contaba con las siguientes funcionalidades.

- Reconocimiento de imágenes para ser utilizadas como marcadores.
- Rastreo sin marcadores, en vez de los marcadores habituales el rastreo se basa en el uso de características naturales como esquinas, bordes o texturas de objetos físicos.
- Mapeo de figuras tridimensionales mediante el uso de la biblioteca OpenGL.

Con respecto a otros SDK, Kudan es más rápido y robusto, probando que su detección y rastreo es más estable [29]. Sus bibliotecas permiten a aplicaciones móviles con realidad aumentada el mapeo de objetos físicos y los importa como modelos multipoligonales 3D. Además el número de imágenes que puede reconocer como marcadores no es limitada y requiere menos memoria para guardar archivos en los dispositivos.

### 2.7.2. Aplicaciones basadas en Android

Android es un paquete que engloba un sistema operativo basado en el núcleo Linux, un *runtime* de ejecución basado en Java, un conjunto de bibliotecas de bajo nivel y un conjunto inicial de aplicaciones destinadas al usuario final. Está diseñado principalmente para dispositivos móviles con pantalla táctil como teléfonos inteligentes o tablets y además para relojes inteligentes, televisores y automóviles.

La arquitectura de Android está basada en cuatro niveles como se muestra en la Fig. 9 detallados a continuación de manera ascendente.

1. Kernel Linux versión 2.6 que sirve como base de la pila de software y se encarga de las funciones más básicas del sistema como la gestión de controladores, gestión de batería, seguridad, etc.

2. Una capa de bibliotecas de bajo nivel en los lenguajes C y C++, como la biblioteca SQLite[30] (sistema de gestión de bases de datos relacional) para la persistencia de datos, soporte de OpenGL ES[31] (variante simplificada de la API gráfica OpenGL para dispositivos integrados tales como teléfonos móviles, tablets y consolas de videojuegos.) para la gestión de gráficos 3D y el motor Webkit[32] (plataforma para aplicaciones que funciona como base para navegadores web) como navegador web embebido en el motor de renderizado HTML<sup>1</sup>.
3. Este nivel posee el *framework* para el desarrollo de aplicaciones, se encuentra dividido en subsistemas para el manejo de los diferentes administradores del sistema, como el administrador de paquetes, el de telefonía, entre otros. La arquitectura de estos subsistemas está diseñada para simplificarle a los desarrolladores la reutilización de funciones de los distintos APIs que ofrece cada administrador. También incluye el subsistema de vistas para manejar la interfaz de usuario de las aplicaciones, incluyendo la posibilidad de visualización de mapas o vistas web directamente en la ventana de la aplicación.
4. La capa de aplicaciones incluyen todas las aplicaciones del dispositivo ya sea que vengan con el dispositivo o aquellas que instale el usuario. También se encuentra la aplicación principal del sistema llamada Inicio que permite ejecutar otras aplicaciones proporcionando la lista de aplicaciones instaladas y mostrando diferentes escritorios donde se pueden colocar accesos directos a aplicaciones o incluso pequeñas aplicaciones incrustadas o *widgets*, que son también aplicaciones de esta capa.

Desde su creación en 2007, Android como sistema operativo ha tenido diversas versiones. Cada versión con una o varias mejoras a nivel de hardware y software con respecto a la anterior, así como avances en el ámbito tecnológico. A continuación se presentan las 13 versiones de Android hasta llegar a su versión actual. Estas son Apple Pie, Banana Bread, Cupcake, Donut, Eclair, Froyo, Gingerbread, Honeycomb, Ice Cream Sandwich, Jelly Bean, KitKat, Lollipop y Marshmallow.

### 2.7.3. Android Studio

Android Studio es un entorno de desarrollo integrado o IDE (*Integrated Development Environment*), que reemplaza a Eclipse como el IDE oficial para el desarrollo de aplicaciones para Android. La interfaz de usuario de Android Studio se muestra en la Fig. 10.

Está basado en el software IntelliJ IDEA, que es un IDE para programas escritos en Java creado por una compañía dedicada a crear herramientas para desarrolladores llamada JetBrains. Android Studio es publicado de forma gratuita a través de la Licencia Apache 2.0 y está disponible para las plataformas de Microsoft Windows, Mac OS X y GNU/Linux.

Además del editor de código y las herramientas de desarrollo, Android Studio ofrece un emulador de Android que permite probar la aplicación de manera más sencilla antes de

---

<sup>1</sup>Lenguaje de marcado para la elaboración de páginas web



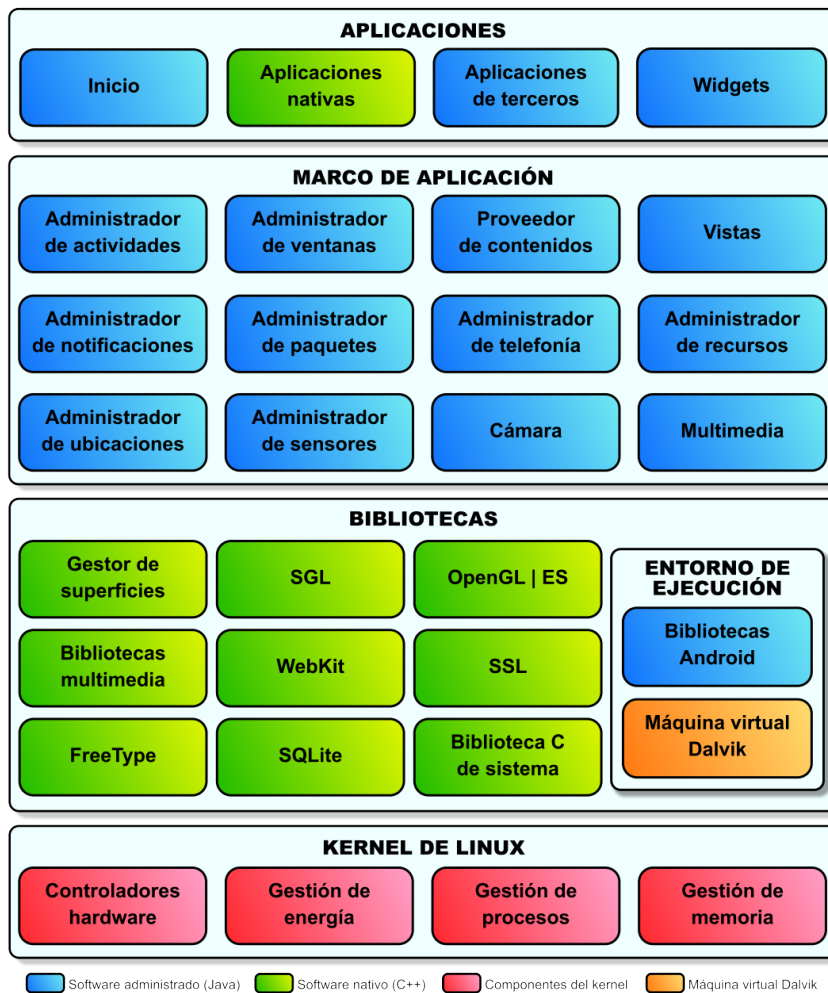


Figura 9: Arquitectura de Android.  
Imagen extraída de [33].

probarlo en un dispositivo. El emulador de Android Studio posee una interfaz de usuario y control de sensores que permiten manipular el dispositivo y observar como se comporta la aplicación. También provee de un entorno unificado donde se puede desarrollar para todos los dispositivos Android incluyendo teléfonos, tablets, y dispositivos *wearables* como el Android Wear.

Desde su segunda versión Android Studio presenta el *Instant Run*, que permite al usuario desarrollar y probar la aplicación de manera más rápida. Al ejecutar la aplicación Instant Run analiza los cambios hechos y determina la manera más rápida de desplegar la aplicación, si es posible sin la necesidad de reconstruir y reinstalar el archivo APK. Este tipo de archivo es una variante del formato JAR de Java y se utiliza para distribuir e instalar componentes Android, como las aplicaciones.

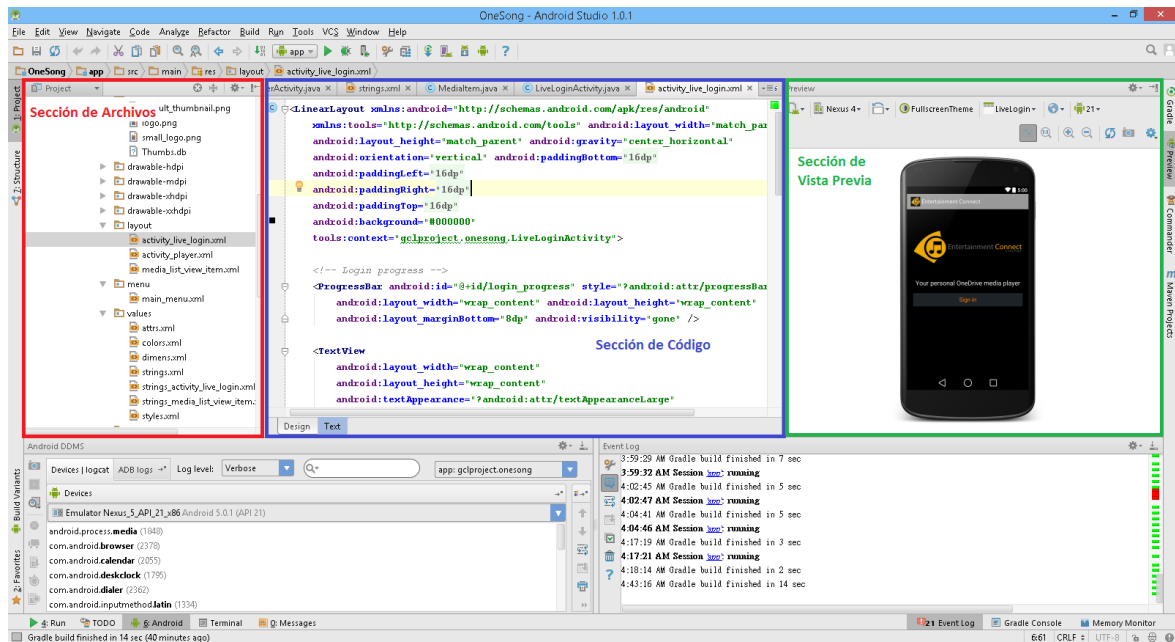


Figura 10: Android Studio, visualización de la sección de archivos, la sección para edición de código y la sección de vista previa, en rojo, azul y verde respectivamente.

Imagen extraída de [34].

Finalmente Android Studio permite el desarrollo de aplicaciones utilizando el lenguaje de programación C++ a través de *Android Native Development Kit* o NDK. Android NDK es utilizado mayormente por motores de juego, permitiendo a los desarrolladores escribir el código de la aplicación en el lenguaje de programación C/C++, que es compilado a código de máquina para la CPU obteniendo un mejor rendimiento del dispositivo.

### 2.7.4. Modelado de las aplicaciones

Cada aplicación de Android ejecuta su propio proceso de Linux y a su vez, cada uno de estos procesos ejecutan su propia máquina virtual Java, de esta manera se aísla la ejecución entre aplicaciones. Para facilitar la reutilización de código y agilizar el proceso de desarrollo, las aplicaciones Android se basan en componentes. Los componentes pueden ser de cuatro tipos y se describen a continuación.

1. **Actividades:** Son interfaces visuales esperando alguna acción del usuario, por ejemplo, una aplicación para el envío de mensajes podría tener una actividad que fuera la lista de contactos de la que el usuario elige a uno, otra que fuera el editor de textos para que el usuario escriba el mensaje y otra con parámetros de configuración de la aplicación. Una aplicación puede tener una actividad o más, y desde una actividad se pueden invocar a otras y volver nuevamente a la original. Todas las actividades extienden la clase *Activity*. El contenido visual de cada actividad lo proporcionan una serie de objetos

derivados de la clase *View*. Android proporciona muchos de estos objetos prediseñados como botones, selectores, menús, etc.

2. Servicios: Los servicios no tienen interfaz gráfica. Un ejemplo sería la reproducción de una canción. Para una aplicación de reproducción de música se podrían tener varias actividades para mostrar listas de canciones o un reproductor con botones, pero el usuario esperará que la canción siga sonando aún al salir de la aplicación (terminar la actividad), por lo que esta aplicación deberá controlar un servicio para que se reproduzca la música. Cualquier servicio extiende de la clase *Service*.
3. Receptores de Eventos: Estos componentes simplemente están escuchando a que se produzcan determinados eventos (e.g. batería baja, cambio del idioma del dispositivo, la descarga de una imagen nueva, etc.) Cualquier aplicación puede tener tantos receptores para tantos eventos como quiera, cada uno debe extender la clase *BroadCastReceiver*. Un ejemplo sería una galería de imágenes que agrega cualquier imagen que el usuario se descargue desde el navegador.
4. Proveedores de contenidos: Permite que una aplicación ponga ciertos datos a disposición de otras aplicaciones. Por ejemplo, una grabadora de sonidos puede compartir esos datos con un reproductor de música. Estos datos pueden almacenarse en el sistema de archivos o en base de datos. Para proveer contenidos, se debe extender la clase *ContentProvider*.

### 2.7.5. Metodología utilizada

Para el desarrollo de este proyecto se decidió utilizar una combinación de las metodologías Scrum y Programación Extrema. De la metodología Scrum se utilizó el concepto de *Sprint*, o iteraciones los cuales se realizan en bloques de tiempo cortos y fijos (entre dos a tres semanas). Cada *sprint* tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite. Cada *sprint* parte de una lista de objetivos o requisitos priorizada del producto, que actúa como plan del proyecto. Al final del *sprint* se presentan los requisitos completados y los cambios que hayan habido en el contexto del proyecto.

Por parte de la metodología de la Programación Extrema se tomó el énfasis que hace en la adaptabilidad antes que la previsibilidad. Es decir, que se considera que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos. También se utilizan las características de *refactoring* del código, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento.

Ambas metodologías se basan en el desarrollo iterativo e incremental creando pequeñas mejoras, unas tras otras lo que hace viable la combinación de sus métodos.

# Capítulo 3

## Solución planteada

En el presente capítulo se describirá el diseño y desarrollo de la solución al problema planteado en el capítulo 1 de este documento. El diseño fue dividido en en *backend* y las dos aplicaciones a desarrollar, la aplicación web y la aplicación móvil, donde se definen las características para cada una de ellas. De igual manera, el desarrollo de la solución explica la elaboración de ambas aplicaciones y el *backend* con el cual éstas se comunican.

### 3.1. Diseño de la solución

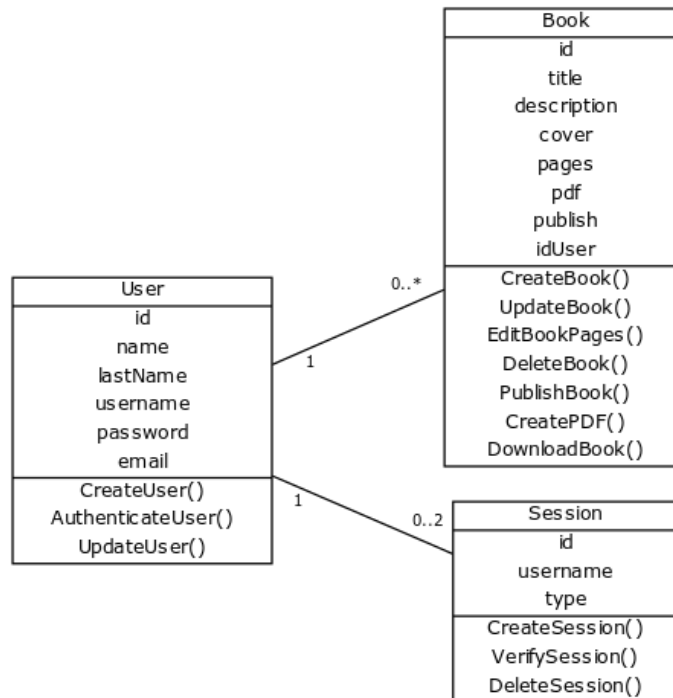
#### 3.1.1. *Backend*

En el diseño del *backend* se plantearon tres entidades, el usuario, el libro y la sesión. En la Fig. 11 se observa el modelo de datos que incluye estas entidades.

La entidad del usuario posee los datos personales del mismo, incluyendo su nombre, apellido, el nombre de usuario con el que se identifica en el sistema, su correo electrónico y su contraseña. Sus funciones incluyen crear una nueva instancia de usuario, autenticar los datos del usuario para el inicio de sesión y actualizar sus datos personales.

El libro posee el título, la descripción o sinopsis del mismo, una portada, el contenido de sus páginas, el archivo PDF asociado, si se encuentra publicado o no y el usuario que lo creó. Sus funciones incluyen la creación de una nueva entidad de libro, actualizar sus datos, editar el contenido del libro, eliminar el libro del sistema, cambiar su estado de publicado, crear el documento PDF del libro y descargar el documento del libro.

Por último la sesión posee el nombre de usuario que inició sesión y si fue iniciada desde la aplicación móvil o la aplicación web. Sus funciones incluyen crear una nueva instancia de sesión, verificar que una sesión existente sea valida y eliminar del sistema una instancia de sesión.

Figura 11: Diseño del *backend*.

### 3.1.2. Aplicación web

En el diseño de esta aplicación se encuentran el *frontend* con el cual el usuario interactúa directamente; el *backend* que recibe peticiones del *frontend*, genera una búsqueda en la base de datos y envía respuestas de vuelta al *frontend*; la base de datos encargada de guardar todas las instancias necesarias del sistema. Este diseño puede verse más claramente en la Fig. 12.

En esta aplicación el usuario puede ser clasificado en dos tipos, el usuario escritor y el usuario lector, de acuerdo al uso que hará de la aplicación. Las acciones del usuario escritor se concentran en la creación de libros, utilizando la herramienta encargada de esta tarea. Por otra parte, el usuario lector, utiliza esta aplicación para la búsqueda y descarga de libros.

De igual manera la aplicación puede dividirse en dos secciones, una sección donde el usuario podrá buscar libros, ver su descripción o sinopsis, y descargarse el documento asociado a éste. Por otra parte la aplicación cuenta con una sección enfocada en la creación de libros con realidad aumentada, ofreciendo una herramienta que permita crear el contenido de éstos. Finalmente la aplicación web ofrecerá los siguientes procesos.

- Creación y autenticación de usuarios, donde el usuario podrá registrarse en el sistema suministrando los datos requeridos. Y posteriormente ingresar al sistema utilizando su nombre de usuario y contraseña.

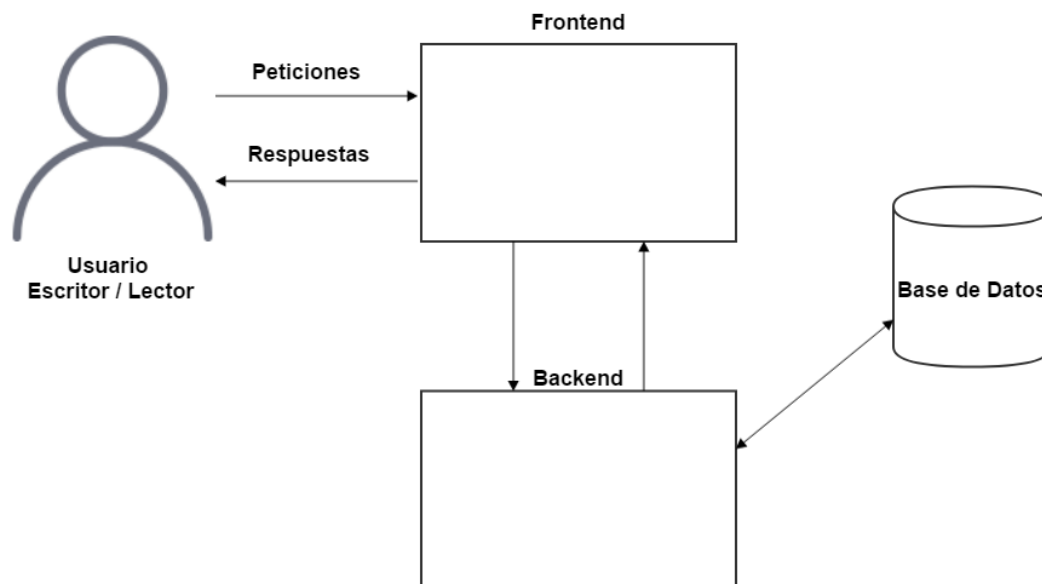


Figura 12: Diseño de la aplicación web.

- Actualización de datos del usuario. El usuario, ya sea lector o escritor, contará con la posibilidad de actualizar los datos suministrados cuando se registró.
- Creación y actualización de libros. El usuario podrá crear la instancia del libro en el sistema ingresando el título del libro, su descripción o sinopsis y la portada. También podrá cambiar algunos de estos datos luego de su creación si así lo quisiese.
- Creación del contenido del libro, para esto el usuario utilizaría la herramienta desarrollada para crear el contenido del libro. Con esta herramienta el usuario podrá añadir y eliminar páginas a su gusto, suministrar contenido ya sea en formato de texto o imágenes. Además esta herramienta ofrece la habilidad de añadirle a una imagen contenido virtual para realidad aumentada, que podrá ser visualizada con la aplicación móvil. Este contenido virtual podrá ser imágenes, videos, audios y objetos 3D. Una vez escrito el libro, el usuario podrá generar el documento asociado para publicarlo, de esta manera podrá ser visto y descargado por los demás usuarios.
- Eliminación de libros y anulación de publicación, estas son dos opciones que se le presentará al usuario escritor para eliminar su libro del sistema ó simplemente anular la publicación del mismo, lo que evitará que otros usuarios puedan ver y descargar el libro.
- Visualización, búsqueda y descarga de libros. Un usuario, ya sea escritor o lector, podrá visualizar y buscar libros publicados por otros usuarios, obteniendo así su título

y sinopsis; además de la posibilidad de descargar el libro, esta descarga sólo será del documento asociado al libro y no del contenido de realidad aumentada.

### 3.1.3. Aplicación móvil

En esta aplicación el usuario enviará peticiones al *backend*, el cual se comunicará con la base de datos para realizar las búsquedas asociadas a las solicitudes hechas por el usuario. Además el usuario podrá descargar el contenido de realidad aumentada a su dispositivo, guardándose en su sistema de archivos. Este diseño puede verse más claramente en la Fig. 13.

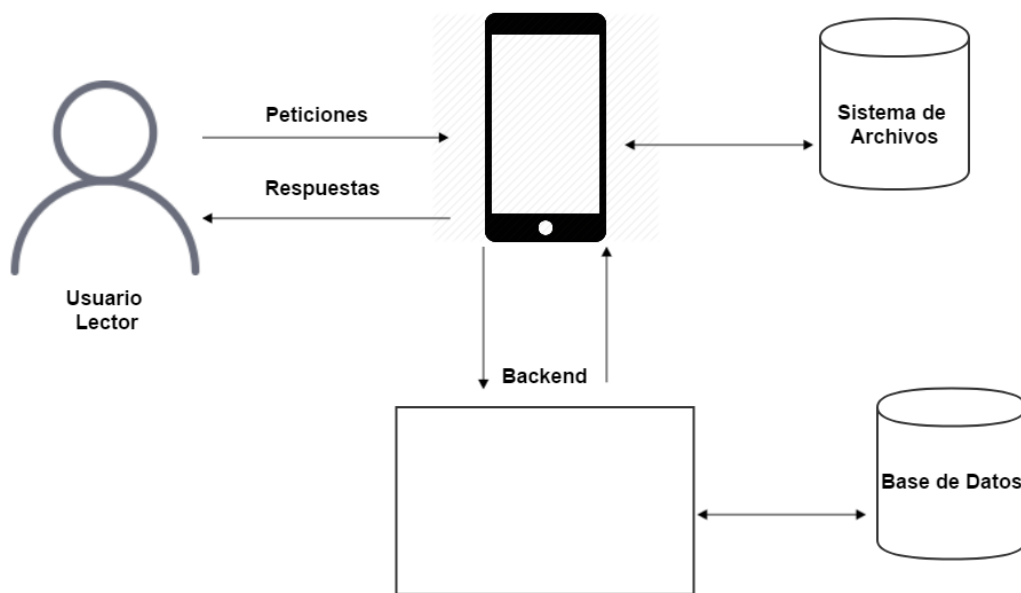


Figura 13: Diseño de la aplicación móvil.

En esta aplicación el usuario es clasificado sólo como usuario lector, pues las acciones que éste realiza se concentran en la búsqueda, descarga y visualización del contenido virtual del libro. Para esto la aplicación se divide en cuatro secciones, una sección de creación y autenticación de usuarios. Una sección de librería donde el usuario pueda buscar libros ya publicados, ver su sinopsis y descargar su contenido. Otra sección será de visualización de contenidos de realidad aumentada de los libros descargados en el dispositivo. Por último habrá una sección donde el usuario podrá ver y actualizar sus datos. Esta aplicación móvil ofrecerá los siguientes procesos.

- Creación y autenticación de usuarios, donde el usuario podrá registrarse en el sistema suministrando los datos requeridos. Y posteriormente Ingresar al sistema utilizando su nombre de usuario y contraseña.

- Actualización de datos del usuario. El usuario contará con la posibilidad de actualizar los datos suministrados cuando se registró.
- Búsqueda, selección y descargas de libros. Al usuario se le ofrecerá una lista de libros en la cual pueda seleccionar uno de ellos, viendo así la descripción o sinopsis del libro. Si así lo desea el usuario puede descargar lo contenidos virtuales del libro, que podrán ser visualizados una vez se descarguen y el usuario seleccione dicho libro desde su biblioteca.
- Selección y visualización del contenido de libros en realidad aumentada. El usuario podrá seleccionar uno de los libros que tenga en su biblioteca, una vez hecho esto el dispositivo entrará en modo cámara. Cuando uno de los maradores del libro se encuentre visible se desplegará el contenido virtual asociado a dicho marcador.

### 3.2. Desarrollo de la solución

Para hacer seguimiento de los avances del desarrollo se estableció el siguiente plan de trabajo (ver Tabla 1) de acuerdo a la metodología usada, descrita en el capítulo 2 de este documento. Este plan de trabajo se divide en *Sprints*, cada *Sprint* abarca un área de trabajo que puede ser el *backend*, la aplicación web o la aplicación móvil. Además cada *Sprint* se divide en tareas que deben cumplirse en un tiempo estimado. Estos *Sprints* serán detallados más adelante.

Sprint	Área de desarrollo	Tareas	Tiempo Estimado
1	<i>Backend</i>	- Estructura. - Base de datos. - Modelo del Usuario. - Controlador del Usuario. - Rutas del Usuario.	2 semanas
2	<i>Backend</i>	- Modelo de la Sesión. - Controlador de la Sesión. - Rutas de la Sesión.	1 semana
3	Aplicación web	- Estructura. - Interfaz de creación y autenticación del Usuario. - Interfaz de escritorio del Usuario. - Controlador del Usuario. - Servicios del Usuario.	2 semanas
4	<i>Backend</i>	- Modelo del Libro. - Controlador del Libro. - Rutas del Libro. - Función para la carga de archivos de tipo imagen, para la portada del libro.	1 semana



5	Aplicación web	<ul style="list-style-type: none"> <li>- Interfaz de creación del Libro en el escritorio del usuario.</li> <li>- Interfaz de búsqueda y muestra del Libro.</li> <li>- Interfaz de detalle del Libro.</li> <li>- Controlador del Libro.</li> <li>- Servicios del Libro.</li> </ul>	3 semanas
6	<i>Backend</i>	<ul style="list-style-type: none"> <li>- Adición de funciones en el Controlador del Libro para el manejo del contenido del Libro.</li> <li>- Adición de rutas del Libro para el manejo del contenido del Libro.</li> <li>- Adición de funciones para publicar, crear y descargar documento del Libro.</li> </ul>	1 semana
7	Aplicación web	<ul style="list-style-type: none"> <li>- Interfaz de la herramienta para la creación del Libro, haciendo uso de directivas.</li> <li>- Controlador de la herramienta para la creación del Libro.</li> <li>- Servicios de la herramienta para la creación del Libro.</li> </ul>	4 semanas
8	Aplicación móvil	<ul style="list-style-type: none"> <li>- Estructura.</li> <li>- Actividad de creación del Usuario. -</li> <li>- Actividad de autenticación del Usuario.</li> <li>- Modelo del Usuario.</li> <li>- Actividad de Perfil del Usuario.</li> </ul>	1 semana
9	Aplicación móvil	<ul style="list-style-type: none"> <li>- Actividad de la Librería.</li> <li>- Modelo del Libro.</li> <li>- Actividad del detalle del Libro.</li> <li>- Actividad de la Biblioteca.</li> <li>- Actividad de la cámara con Realidad Aumentada.</li> </ul>	1 semana
10	Aplicación móvil Aplicación web <i>Backend</i>	<ul style="list-style-type: none"> <li>- Búsqueda y corrección de errores.</li> </ul>	1 semana

Tabla 1: Plan de trabajo para el desarrollo de la solución

### 3.2.1. Desarrollo del *backend*

Para la elaboración del *backend* se desarrolló un API REST con el cual las aplicaciones web y móvil pudiesen comunicarse y así obtener los datos solicitados por el usuario. Un API REST es cualquier interfaz entre sistemas que usen el protocolo HTTP para obtener datos o generar operaciones sobre dichos datos, en todos los formatos posibles. Para este proyecto se decidió usar el formato JSON [35] como el formato de las peticiones y respuestas del API.

El desarrollo del *backend* se realizó en una máquina virtual con el sistema operativo Ubuntu en su versión 16.04 con 2GB de memoria RAM y 30GB de memoria SATA. Se decidió utilizar Node.js v6.9 [36] y express.js [37] como infraestructura para el desarrollo del API REST en conjunto con paquetes como bcrypt [38], para la encriptación del password y multer [39] como *middleware* para el manejo de archivos. Para la base de datos se decidió utilizar MySQL [40] y como ORM se decidió utilizar bookshelf.js [41].

#### 3.2.1.1. Implementación

En el desarrollo del *backend* se siguió el diagrama de clases visto en la Fig. 14 para crear las tablas de la base de datos y los modelos de Usuario, Sesión y Libro en el API. Para estos modelos se elaboraron funciones CRUD, es decir, de creación, lectura, actualización y borrado utilizando los métodos POST, GET, PUT y DELETE respectivamente, en las peticiones HTTP.

El trabajo de desarrollo de cada modelo fue dividido en distintos *sprints*. A continuación se presentan los *sprints* dedicados al *backend*, haciendo énfasis en su desarrollo y resultados.

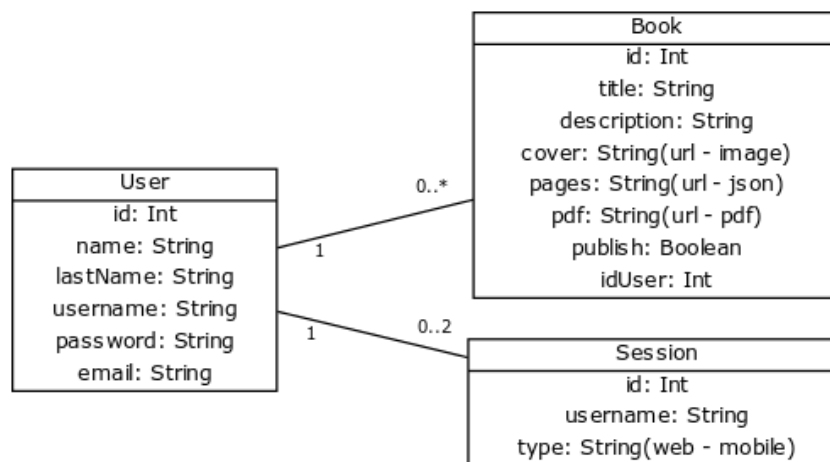


Figura 14: Diagrama de clases del *backend*.

### 3.2.1.2. Sprint 1

En el primer *sprint* se creó una estructura de archivos que pudiese ser navegable y escalable, se descargaron los paquetes de terceros necesarios para la aplicación, se instaló el manejador de base de datos MySQL y se creó la base de datos a utilizar. También se creó un archivo de rutas en donde se guardarán las rutas de los modelos. Una vez hecho esto, se creó el modelo del Usuario, su controlador y se escribieron las rutas en el archivo de rutas.

En la Tabla 2 se describen los métodos del modelo del usuario que fueron desarrollados y las rutas asociadas, así como los parámetros (*params*, *headers* y *body*), que deben ser enviados en la petición HTTP para recibir una respuesta del *backend*.

Nombre	Método	Ruta	Parámetros
createUser	POST	/api/users	<ul style="list-style-type: none"> <li>- name : <i>string (body)</i></li> <li>- lastName: <i>string (body)</i></li> <li>- username: <i>string / 8 - 16 char (body)</i></li> <li>- password: <i>string / 8 - 16 char (body)</i></li> <li>- email: <i>string (body)</i></li> <li>- device: <i>string (body)</i></li> </ul>
getUsers	GET	/api/users	- token: <i>json-web-token (header)</i>
getUserById	GET	/api/users/:id	<ul style="list-style-type: none"> <li>- token: <i>json-web-token (header)</i></li> <li>- id: <i>integer (params)</i></li> </ul>
updateUser	PUT	/api/users/:id	<ul style="list-style-type: none"> <li>- token: <i>json-web-token (header)</i></li> <li>- id: <i>integer (params)</i></li> <li>- name : <i>string (body)</i></li> <li>- lastName: <i>string (body)</i></li> <li>- password: <i>string / 8 - 16 char (body)</i></li> <li>- email: <i>string (body)</i></li> </ul>
deleteUser	DELETE	/api/users/:id	<ul style="list-style-type: none"> <li>- token: <i>json-web-token (header)</i></li> <li>- id: <i>integer (params)</i></li> </ul>

Tabla 2: Métodos del modelo Usuario.

### 3.2.1.3. Sprint 2

En el segundo *sprint* se desarrolló el modelo de la sesión, donde se estableció la creación del token que identifica al usuario dentro del sistema y su controlador. Además se añadieron las rutas asociadas a este modelo al archivo de rutas creado en el primer *sprint*. En la Tabla 3 se describen las funciones desarrolladas para este modelo.

Nombre	Método	Ruta	Parámetros
createSession	POST	/api/sessions	- username: <i>string</i> / 8 - 16 <i>char</i> ( <i>body</i> ) - password: <i>string</i> / 8 - 16 <i>char</i> ( <i>body</i> ) - device: <i>string</i> / <i>web</i> o <i>mobile</i> ( <i>body</i> )
deleteUser	DELETE	/api/sessions	- token: <i>json-web-token</i> ( <i>header</i> )
verifySession	MIDDLEWARE	Sin ruta.	Ninguno. Verifica si la sesión (token) es valido.
deleteExistingSession	MIDDLEWARE	Sin ruta.	Ninguno. Si el usuario ya tiene una sesión activa, ésta es borrada.

Tabla 3: Métodos del modelo Sesión.

#### 3.2.1.4. Sprint 4

En este *sprint* se desarrolló el modelo del libro, y su controlados, sin embargo sólo se desarrollaron sus funciones CRUD. También se creó una función para la carga de imágenes para la portada del libro. Posteriormente se añadieron las rutas asociadas a este modelo al archivo de rutas creado en el primer *sprint*. Las funciones de este modelo se presentan en la tabla 4.

Nombre	Método	Ruta	Parámetros
createBook	POST	/api/books	- token: <i>json-web-token</i> ( <i>header</i> ) - title: <i>string</i> ( <i>body</i> ) - description: <i>string</i> ( <i>body</i> ) - cover: <i>file</i> ( <i>body</i> ) - idUser: <i>integer</i> ( <i>body</i> )
getBooks	GET	/api/books	- token: <i>json-web-token</i> ( <i>header</i> )
getBookById	GET	/api/books/:id	- token: <i>json-web-token</i> ( <i>header</i> ) - id: <i>integer</i> ( <i>params</i> )
updateBook	PUT	/api/books/:id	- token: <i>json-web-token</i> ( <i>header</i> ) - id: <i>integer</i> ( <i>params</i> ) - description: <i>string</i> ( <i>body</i> ) - cover: <i>file</i> ( <i>body</i> )
deleteUser	DELETE	/api/books/:id	- token: <i>json-web-token</i> ( <i>header</i> ) - id: <i>integer</i> ( <i>params</i> )

Tabla 4: Métodos del modelo Libro.

### 3.2.1.5. Sprint 6

En este último *sprint* se añadieron funciones para el manejo de carga de archivos de audio (.mp3 y .wav), video (.mp4 y .3gp), imágenes (.jpg y .png) y *octet-stream* (.jet). Estos archivos serán del contenido y los marcadores del libro. También se desarrollaron las funciones para la carga y descarga del documento del libro en formato PDF y la publicación del libro. En la Tabla 5 se describen las funciones que se añadieron al controlador del Libro y posteriormente a sus rutas.

Nombre	Método	Ruta	Parámetros
publishBook	PUT	/api/books/:id/publish	- token: <i>json-web-token</i> (header) - id: <i>integer</i> (params) - publish: <i>boolean</i> (body)
updateBookPages	GET	/api/books/:id/save	- token: <i>json-web-token</i> (header) - id: <i>integer</i> (params) - pages : <i>json object</i> (body)
getPDF	GET	/api/books/:id/pdf	- token: <i>json-web-token</i> (header) - id: <i>integer</i> (params)
savePDF	PUT	/api/books/:id/pdf	- token: <i>json-web-token</i> (header) - id: <i>integer</i> (params) - pdf: <i>file</i> (body)
addContent	POST	/api/books/:id/content	- token: <i>json-web-token</i> (header) - id: <i>integer</i> (params) - marker: <i>file</i> (body) - content: <i>file</i> (body) - markerPath: <i>string</i> (body) - contentPath: <i>string</i> (body)

Tabla 5: Métodos del contenido y publicación del Libro.

### 3.2.2. Desarrollo de la aplicación web

La aplicación web se desarrolló en una computadora con el sistema operativo Windows 10 64-bit, con 4GB de memoria RAM y procesador Intel(R) Core(TM) i5-6200. El desarrollo se hizo utilizando el *framework* AngularJS [42] en su versión 1.6.1, Bootstrap [43] en su versión 3.3.7 y ui-Bootstrap [44] en su versión 2.5 como frameworks de estilo (CSS). Además se utilizó la biblioteca JSPDF [45] para la generación de documentos PDF y la biblioteca ngFileUpload [46] en su versión 12.2.13 para la carga de archivos.

### 3.2.2.1. Implementación

AngularJS utiliza una arquitectura Modelo Vista Controlador (MVC), en esta aplicación nuestro modelo viene dado por el *backend* donde se estructura y mantienen los datos de los diferentes objetos descritos en el diagrama de clase de la Fig. 14. Las vistas son las interfaces con las que el usuario interactúa presentadas más adelante. Y los controladores son las funciones que reallizan operaciones sobre los datos ingresados ó solicitados por el usuario. Por último se definieron servicios, los que realizan las peticiones HTTP al *backend*; y directivas las cuales definen objetos HTML para su utilización en las distintas interfaces, sin necesidad de repetir código.

Para el desarrollo de la aplicación, la cual se decidió nombrar *Typewriter AR*, se dividió por modelos y según las funcionalidades definidas anteriormente en el diseño de la aplicación. A continuación se presentan los *sprints* dedicados a la aplicación web haciendo énfasis en su desarrollo y mostrando los resultados.

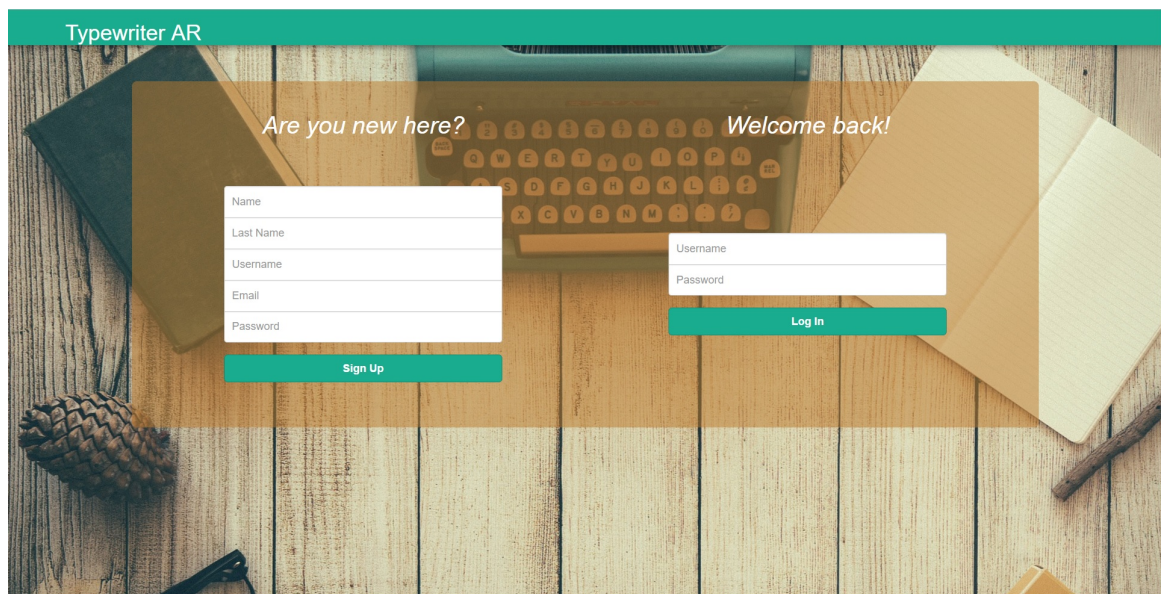
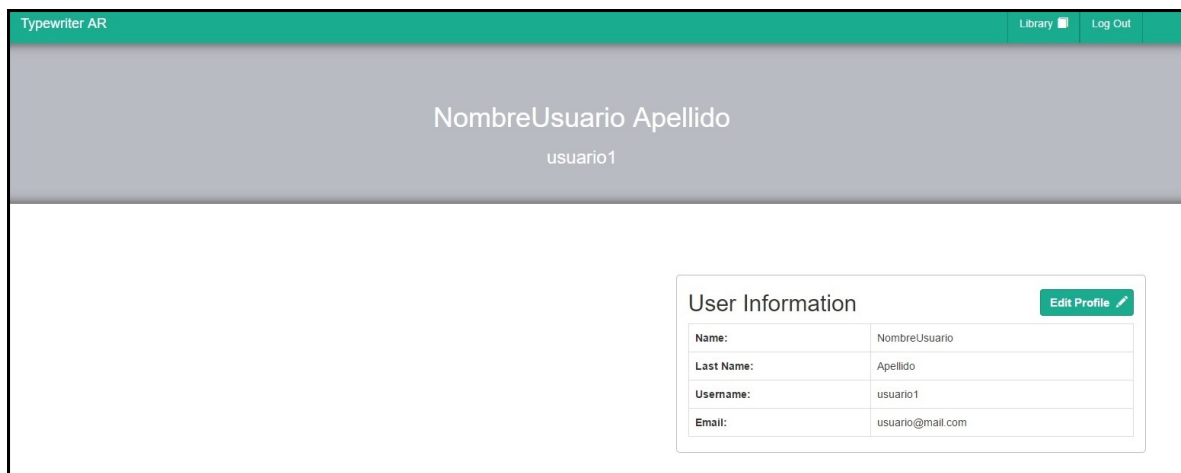
### 3.2.2.2. Sprint 3

En este *sprint* se creó la estructura del proyecto que pudiese ser navegable y escalable, se añadieron las bibliotecas de terceros mencionadas previamente y se creó el archivo que contiene las rutas de navegación. Posteriormente se creó la vista principal (*home*), la cual muestra los formularios de creación y autenticación de usuarios. Luego se creó una segunda vista la cual lleva el nombre de *My Desktop* donde se muestra al usuario sus datos y se le da la posibilidad de cambiar alguno de ellos.

Una vez terminadas las vistas se desarrollaron los controladores y los servicios asociados al usuario, los cuales verifican que los datos introducidos por el usuario sean correctos y envían las peticiones HTTP al *backend* respectivamente. Los controladores también se encargan de obtener la respuesta del *backend* y mostrar al usuario su *My Desktop*, en caso de que la respuesta haya sido buena, ó un mensaje de error, en caso de que haya habido un error en la petición. Las Interfaces de *home* y de *My Desktop* pueden verse en la Fig. 15 y la Fig. 16 respectivamente.

### 3.2.2.3. Sprint 5

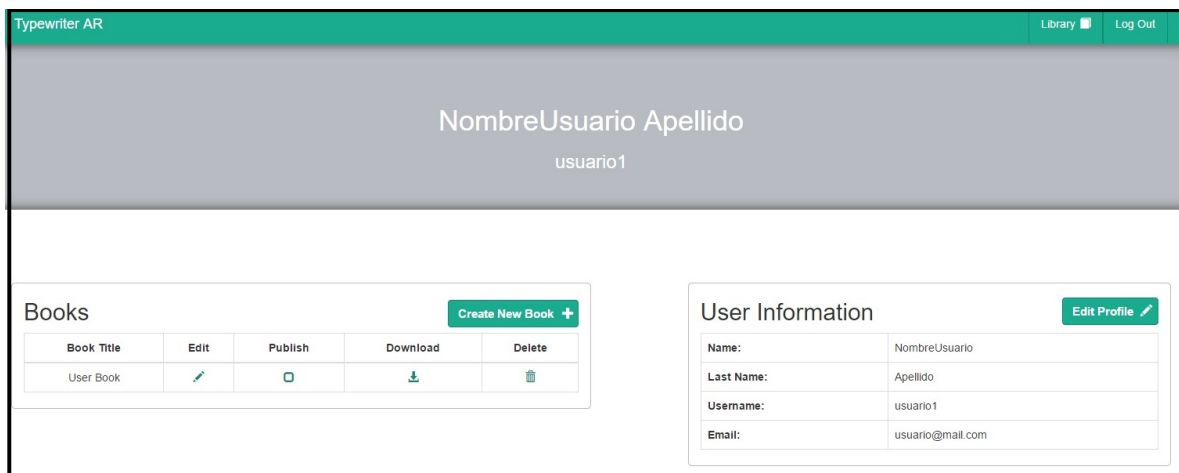
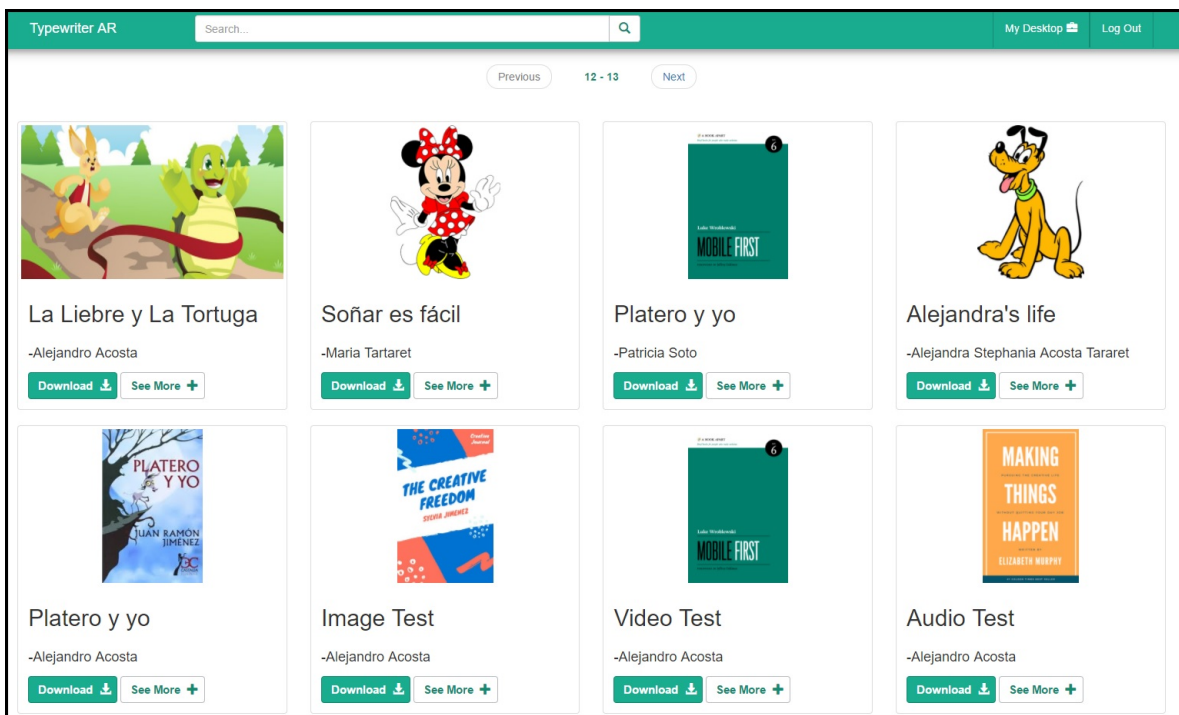
En este *sprint* se empezó añadiendo a la interfaz de *My Desktop* una tabla de los libros del usuario y un formulario para la creación de libros, donde el usuario ingresa el título y la descripción (sinopsis) del libro y escoge una portada. Además se da al usuario las opciones de editar la portada y la descripción del libro, al seleccionar la celda del título en la tabla de libros. También se le da al usuario las opciones de editar, publicar, descargar y eliminar el libro. En el controlador asociado a la interfaz de *My Desktop* se añadieron las validaciones para el formulario y las funciones para la carga del archivo. Seguido de esto se creó el servicio asociado al libro, con todas las funciones para enviar peticiones al *backend*. Estos cambios se

Figura 15: Interfaz *home* de la aplicación web.Figura 16: Interfaz *My Desktop* de la aplicación web.

pueden apreciar en la Fig. 17

Seguido a esto se crea una interfaz llamada *Library* donde se listan los libros que han sido publicados, y se le da al usuario la opción de buscar un libro por su título, por el nombre del autor o por el apellido del autor. Esta interfaz se puede ver en la Fig. 18. Por otra parte se crea una interfaz para mostrar un libro en detalle, mostrando al autor y la descripción o sinopsis del libro, como puede verse en la Fig. 19.

Por último se creó el controlador del libro, el cual se encarga de listar los libros obtenidos

Figura 17: Interfaz *My Desktop* con la tabla de Libros.Figura 18: Interfaz *Library* con lista de Libros.

del *backend*, obtener el detalle de un libro cuando el usuario lo selecciona de la lista y generar una petición al backend para descargar el documento del libro. Esta descarga puede hacerse desde la interfaz de la lista o desde la interfaz con el detalle del libro.



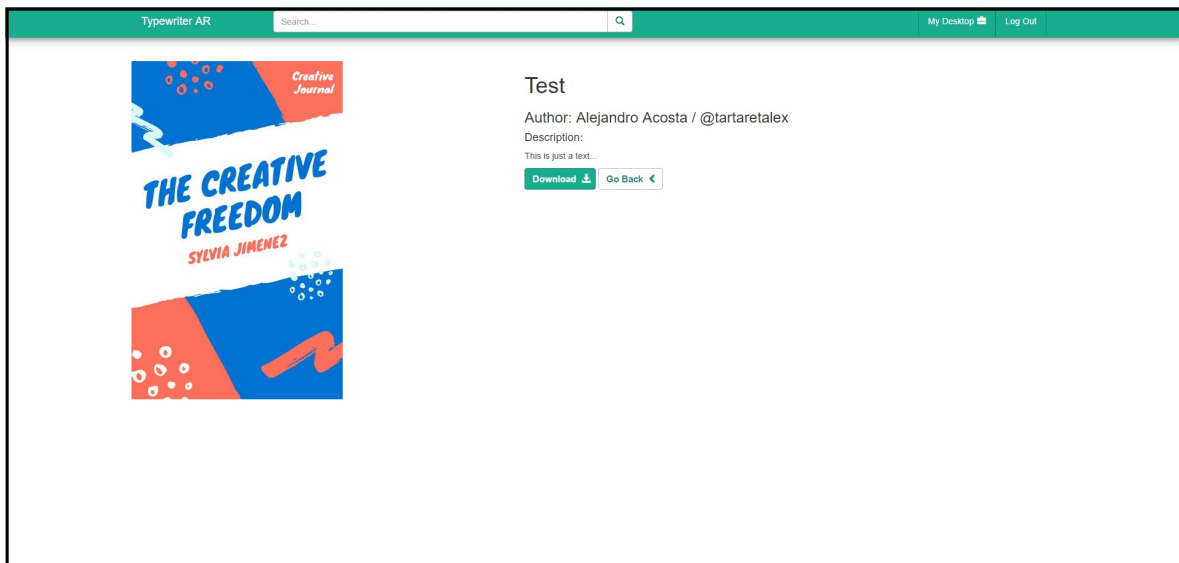


Figura 19: Interfaz del detalle de un Libro.

#### 3.2.2.4. Sprint 7

Este *sprint* se dedicó al desarrollo de la herramienta para la creación del contenido del libro con realidad aumentada. Para esta interfaz se hizo uso de las directivas que proporciona AngularJS, las cuales permiten crear elementos HTML que pueden ser reutilizados en distintas partes de la interfaz. Esta interfaz puede dividirse en cuatro secciones. La primera siendo la barra de navegación o *navbar* que se encuentra en la parte superior de la pantalla, ésta posee tres botones el primero que permite al usuario guardar los cambios hechos al contenido del libro en el *backend*; el segundo permite al usuario crear el documento en PDF del libro, lo que genera la descarga del documento en el navegador y luego lo guarda en el *backend*; por último el tercer botón permite al usuario egresar a la interfaz de *My Desktop*. El *navbar* puede apreciarse en la Fig. 20.



Figura 20: Barra de navegación de la interfaz de la herramienta.

La siguiente sección es el menú de selección, en este menú el usuario puede escoger uno de los cinco tipos de página que desee crear. Estos tipos pueden ser de texto, título e imagen, y también la combinación de título con texto y título con imagen. Este menú puede verse en la Fig. 21.

La tercera sección se encuentra debajo del menú de selección, a la derecha de la pantalla. En esta sección se aprecian los botones para añadir o eliminar páginas y la vista previa de la página actual que se está editando. En esta vista previa se mostrará todo el contenido que

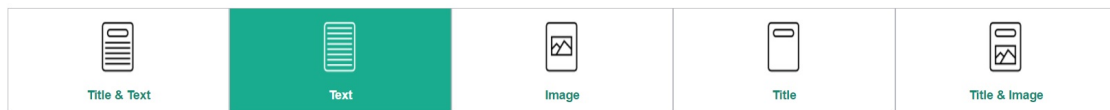


Figura 21: Menú de selección de la interfaz de la herramienta.

añada el usuario, excepto por el contenido virtual para realidad aumentada.

Finalmente, la cuarta sección se encuentra debajo del menú de selección, a la izquierda de la pantalla. En esta sección se tiene un botón de selección con el cual el usuario puede escoger la página que desea editar. Por debajo de este botón está un formulario que cambia dependiendo del tipo de página seleccionado en el menú de selección. Si se selecciona el tipo de página con texto, se le presenta al usuario un *textbox* en donde el usuario puede introducir el texto a mostrar en la página, una vez añadido el texto el usuario debe presionar el botón de *Submit Text* para que el texto se añada a la vista previa de la página. Por último se le presentan al usuario dos botones de selección, el primero para escoger el tamaño del texto y el segundo para escoger el tipo de fuente. Esta interfaz puede apreciarse en la Fig. 22.

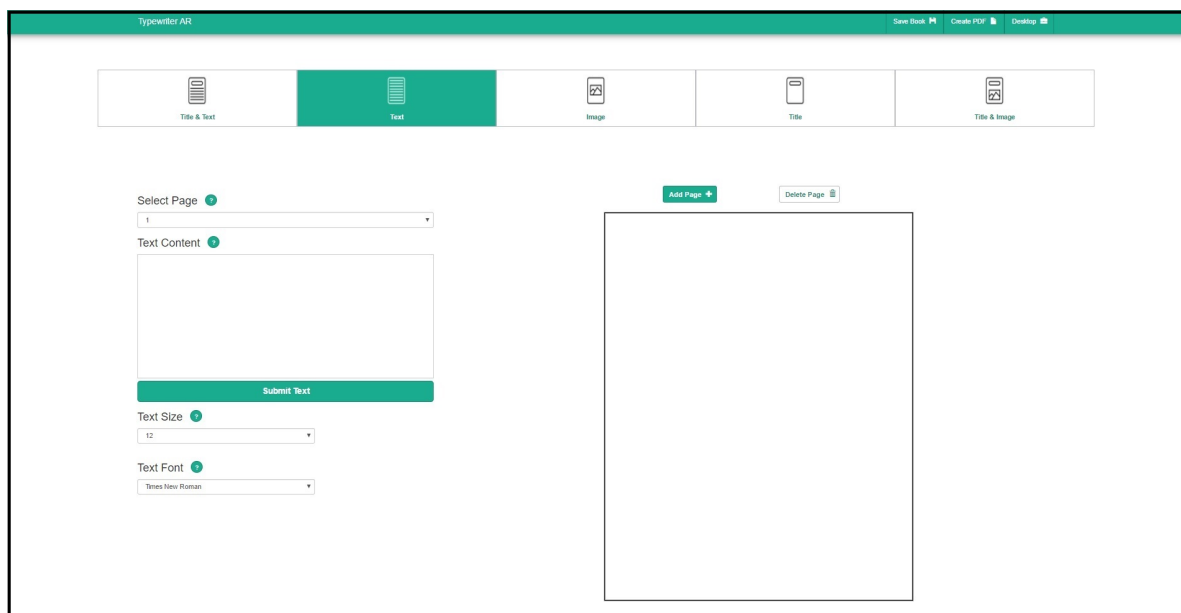


Figura 22: Interfaz de la herramienta con la selección de texto.

Si el tipo de página seleccionado es de título, al usuario se le presenta un *textbox* donde puede escribir, a diferencia del *textbox* descrito anteriormente, no se necesita presionar un botón para que el texto se aprecie en la vista previa de la página, esto ocurre automáticamente. Por debajo de esto se encuentran los botones de selección con los cuales el usuario puede cambiar el tamaño del texto y su tipo de fuente. Esta vista puede apreciarse en la Fig. 23.

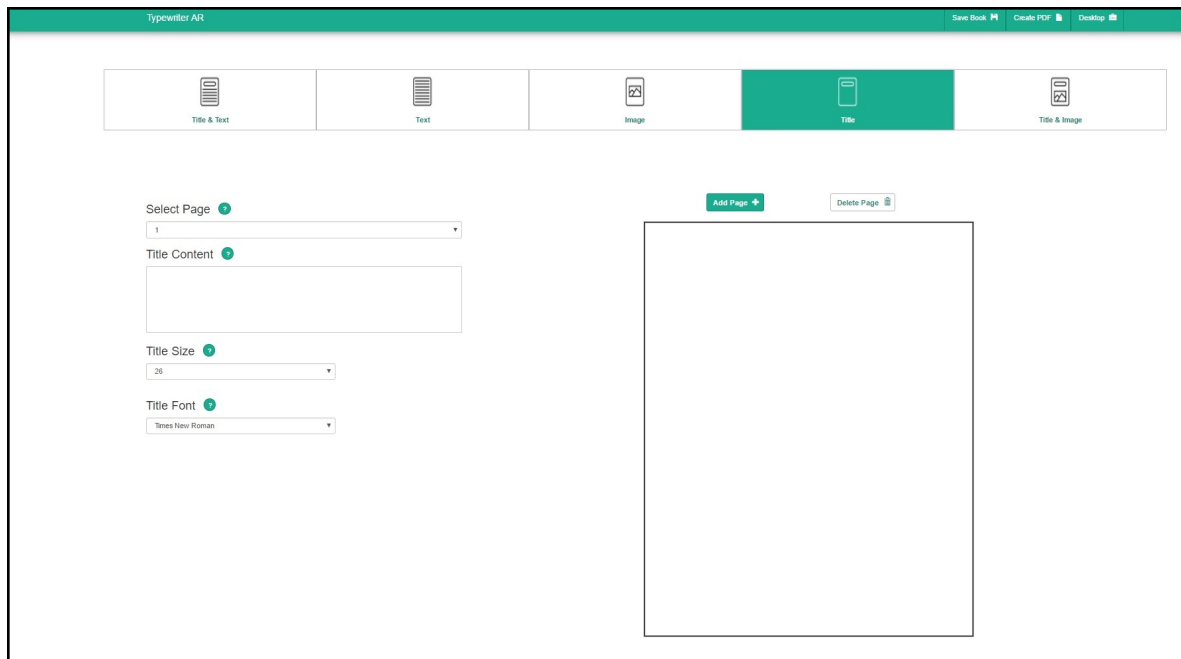


Figura 23: Interfaz de la herramienta con la selección de título.

Si el usuario ha seleccionado imagen como el tipo de página se presenta con una interfaz más compleja que las anteriores. Primeramente esta interfaz tiene un botón con el cual el usuario selecciona una imagen de tipo *.jpeg* que será usada como el marcador del contenido de realidad aumentada a elegir próximamente. Antes de cargar el contenido de realidad aumentada el usuario debe indicar que tipo de archivo va a usar, estos tipos pueden ser imagen (*.jpg*, *.png*), audio (*.mp3*, *.wav*), video (*.mp4*, *.3gp*) y objetos 3D (*.obj*, *.fbx*). Los objetos 3D deben ser transformados al formato *.jet* utilizando el *Toolkit* provisto por Kudan en su página [47], este tipo de formato está diseñado para resultar en un archivo de pequeño tamaño y que sea rápido de cargar [48]. Además los objetos 3D deben ser cargados con un archivo de textura (*.jpg*, *.png*), lo que añadirá color al objeto cuando sea renderizado con la aplicación.

Una vez cargado el marcador, seleccionado el tipo de contenido y cargados los archivos, el usuario debe presionar el botón *Upload Files* y esperar a que la imagen seleccionada como marcador aparezca en la vista previa. Esta imagen puede ser editada usando los controles que están más abajo. Los primeros dos controles permiten cambiar las dimensiones de la imagen, ancho y alto de la imagen respectivamente. Se recomienda mantener la resolución de la imagen original para evitar conflictos a la hora de usar la aplicación. El segundo par permite cambiar la posición de la imagen horizontal y verticalmente respectivamente. Esto puede verse en la Fig. 24.

Los últimos dos tipos de imagen son una combinación de los previamente mencionados. El tipo de título y texto permite crear una página que posea ambos, y en la sección izquierda

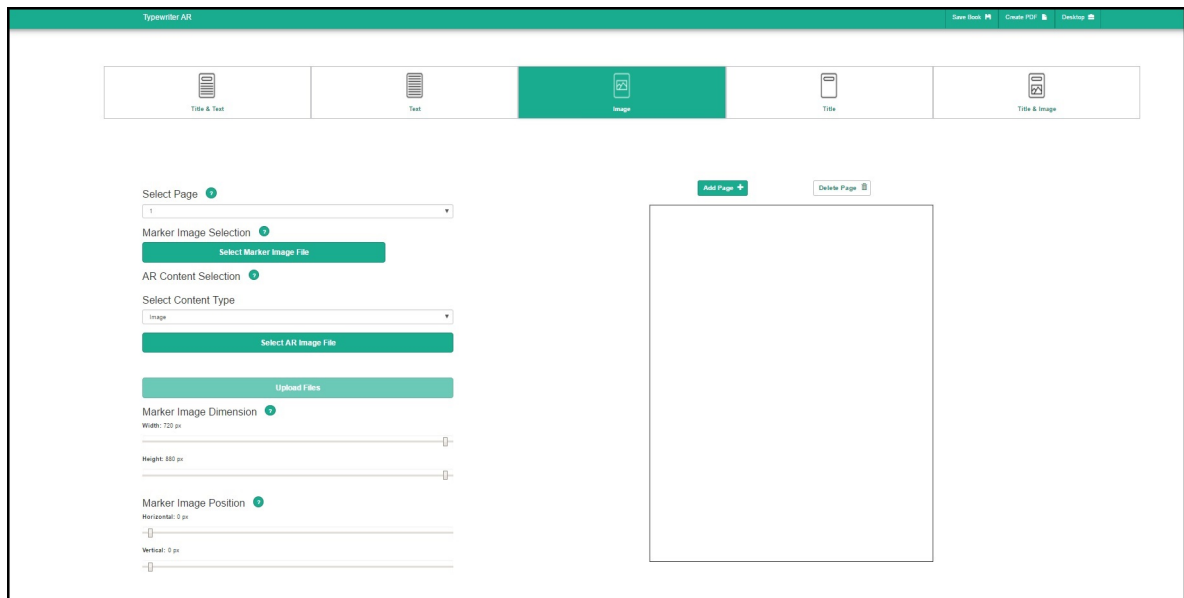


Figura 24: Interfaz de la herramienta con la selección de imagen.

se añaden ambas directivas. Esto ocurre igualmente si el tipo de página elegido es título e imagen, las directivas de título e imagen se combinan en el lado izquierdo de la pantalla. Esto puede verse en la Fig. 25 y la Fig. 26, respectivamente.

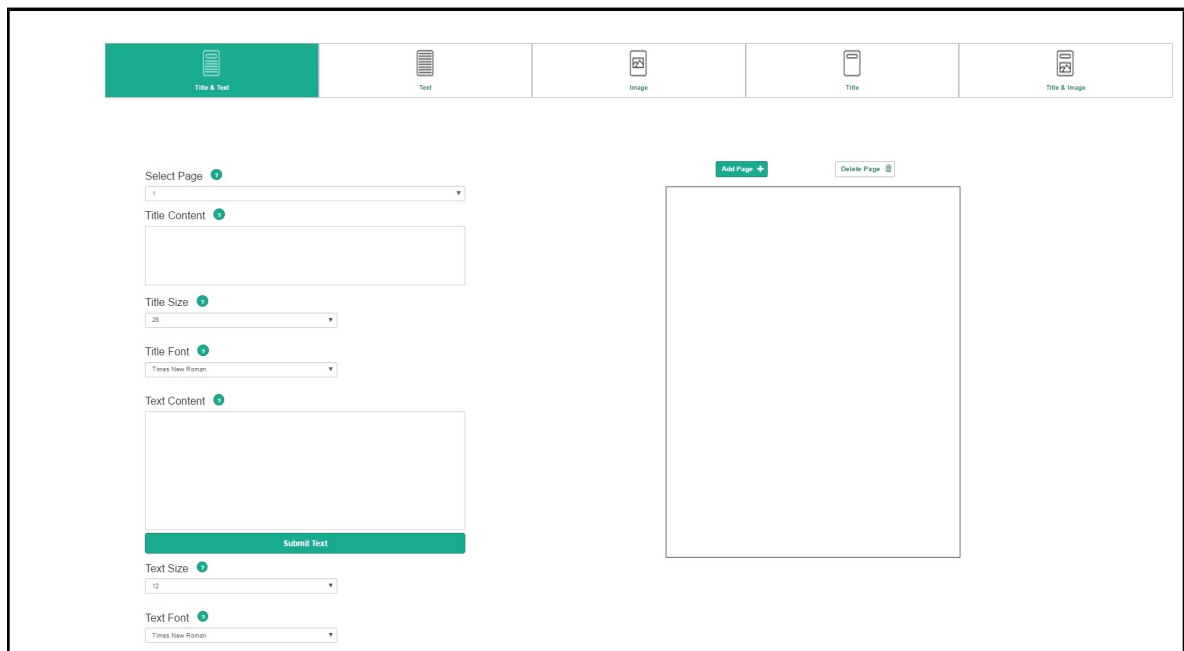


Figura 25: Interfaz de la herramienta con la selección de título y texto.

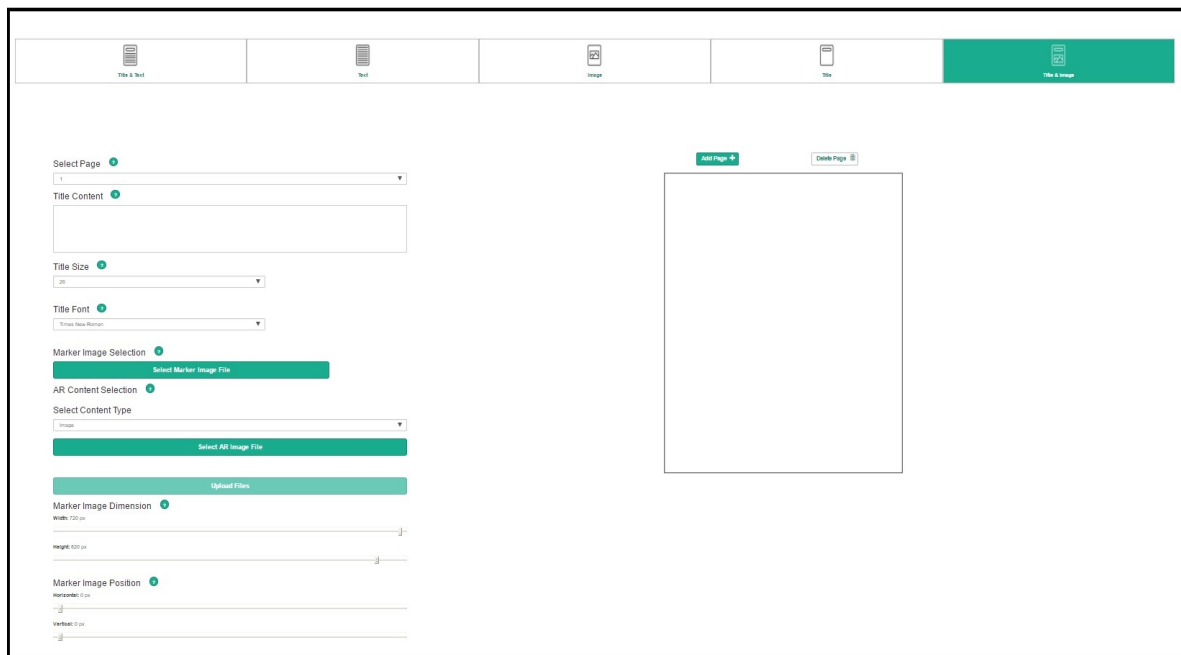


Figura 26: Interfaz de la herramienta con la selección de título e imagen.

### 3.2.3. Desarrollo de la aplicación móvil

La aplicación móvil se desarrolló en una computadora con el sistema operativo Windows 10 64-bit, con 4GB de memoria RAM y procesador Intel(R) Core(TM) i5-6200. Para el desarrollo de la aplicación se utilizó Android Studio en su versión 2.3. Para el rastreo de marcadores y renderizado del contenido virtual se utilizó el SDK para android Kudan [49]. Como dispositivo de prueba se utilizó un ZTE Axon 7 con sistema operativo Android en su versión 6.0.1 *Marshmallow*.

#### 3.2.3.1. Implementación

En Android Studio cada interfaz, llamada actividad, está compuesta por un archivo de *layout* y una clase Java. También se crearon clases Java de los modelos que fueron descritos en el diagrama de clases visto en la Fig. 14. Para el manejo de peticiones HTML se utilizó el paquete Volley de Google [50], para la lectura de JSON se utilizó el paquete GSON de Google [51] y para la descarga de archivos al dispositivo móvil se utilizó el Download Manager provisto por Android, que permite múltiples descargas de archivos al mismo tiempo.

#### 3.2.3.2. Sprint 8

En el primer *sprint* se creó el proyecto de Android Studio, en el archivo Manifest se añadieron los permisos correspondientes y se siguió la guía de Kudan [52] para crear un nuevo proyecto. Luego se creó la actividad principal, que viene dada por la actividad de *Inicio de*

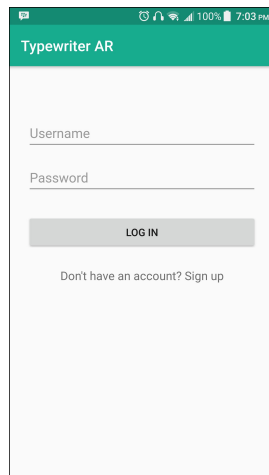


Figura 27: Actividad de Inicio de sesión.

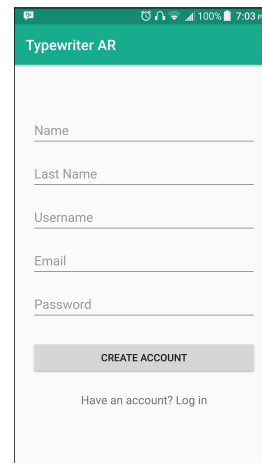


Figura 28: Actividad de Creación de usuario.

*sesión*, posteriormente se creó la actividad de *Creación de usuario*, cada una con las validaciones pertinentes de los datos del usuario. Dentro de cada una se hicieron, además, las peticiones HTML al *backend*. Para guardar la sesión y el usuario se crearon dos archivos guardados en *Shared Preferences* para ser vistos, solamente, por las otras actividades de la aplicación. También se creó una clase para el modelo del Usuario, de esta manera el archivo puede convertirse en un objeto Java para interactuar con sus valores y luego volver a salvarlo en el archivo. Las actividades de *Inicio de sesión* y de *Creación de usuario* pueden verse en la Fig. 27 y la Fig. 28 respectivamente.

Posteriormente se creó la actividad de *Perfil de usuario* donde se muestran su nombre, apellido y nombre de usuario. Además, se le da la opción de actualizar sus datos básicos, así como cerrar la sesión y volver a la pantalla de inicio de sesión. Esta actualización de datos se hace en una nueva actividad. En esta actividad, si el usuario decide actualizar sus datos, se envía la petición al *backend*, se reescriben los datos del usuario y se actualizan en la aplicación. Una vez que ocurra esto se regresa a la actividad del *Perfil de usuario*. Las actividades de *Perfil de usuario* y de *Actualización de usuario* pueden verse en la Fig. 29 y la Fig. 30 respectivamente.

### 3.2.3.3. Sprint 9

En este *sprint* se desarrollaron las actividades de *Librería*, donde se muestra al usuario la lista de libros que han sido publicados. Para obtener la información de los libros desde el *backend* se crea una clase Libro la cual contiene los mismos atributos que el modelo del libro descrito anteriormente. Al seleccionar uno de estos libros se crea una actividad donde se muestra el detalle del libro, incluyendo su título, su descripción o sinopsis y el nombre y apellido del autor. en esta actividad además se muestra un botón con el cuál el usuario puede descargar los marcadores y el contenido virtual del libro. Para realizar esta descarga primeiramente se descarga el archivo *pages.json* el cual contiene la información del contenido del



Figura 29: Actividad de Perfil de usuario.

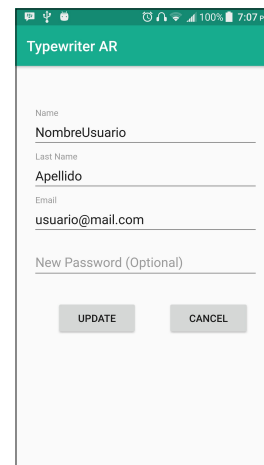


Figura 30: Actividad de Actualización de usuario.

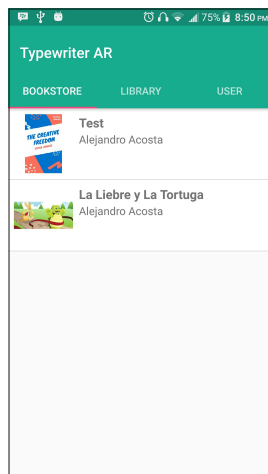


Figura 31: Actividad de Librería.

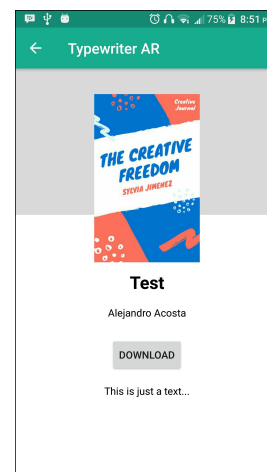


Figura 32: Actividad de Detalle del Libro.

libro. Una vez descargado *pages.json* se generan las descargas del resto de los archivos. La actividad de *Librería* y del *Detalle del Libro* se aprecian en la Fig. 31 y la Fig. 32 respectivamente.

Por último se desarrolló la actividad de *Biblioteca* la cual muestra los libros descargados al dispositivo por el usuario. Para esto la actividad ingresa a la carpeta donde se descargan los libros y obtiene la portada asociada a cada uno de ellos. Cuando el usuario selecciona alguno de los libros, se carga la actividad de *Realidad Aumentada* que inicia la cámara del dispositivo para la visualización en realidad aumentada del contenido del libro. En esta actividad se crean los *trackers* o rastreadores de cada uno de los marcadores del libro. Estos rastreadores, dependiendo del tipo de contenido que haya seleccionado el usuario, generan a su vez nodos, los cuales a su vez, contienen la dirección del contenido asociado al marcador. Para los

objetos 3D se implementó una escala, la cual permite la proporción de 1 unidad a 1 cm. Es decir, que por cada unidad de vértice, el objeto ocupará un espacio de 1cm. La actividad de *Biblioteca* y de *Realidad Aumentada* se aprecian en la Fig. 33 y la Fig. 34 respectivamente.

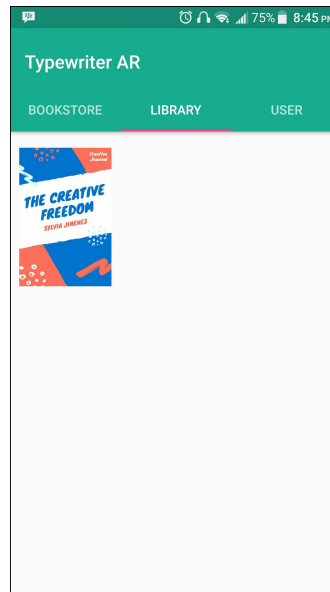


Figura 33: Actividad de Biblioteca.

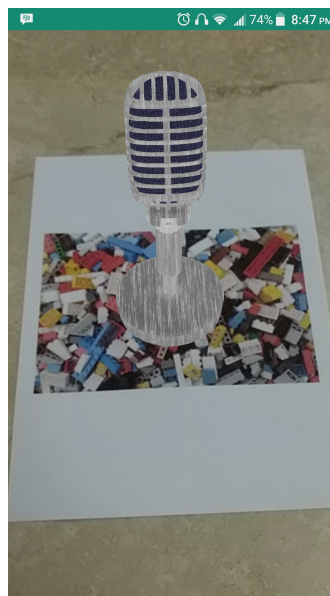


Figura 34: Actividad de Realidad Aumentada.



# Capítulo 4

## Pruebas y Resultados

En el presente capítulo se describen las pruebas realizadas a cada una de las aplicaciones, web y móvil. Para la aplicación móvil, se realizaron pruebas cuantitativas, midiendo tiempo, memoria y rendimiento al momento de visualizar el contenido de realidad aumentada. Por otra parte, para la aplicación web, la prueba fue cualitativa a través de encuestas. Estas pruebas y sus resultados se presentan a continuación.

### 4.1. Pruebas cuantitativas (Aplicación móvil)

#### 4.1.1. Ambiente de pruebas

Estas pruebas se realizaron en una computadora con el sistema operativo Windows 10 64-bit, con 4GB de memoria RAM y procesador Intel(R) Core(TM) i5-6200. Como dispositivos de prueba se utilizaron un ZTE Axon 7 con sistema operativo Android *Marshmallow* en su versión 6.0.1, CPU Qualcomm Snapdragon 820 2.15G 2x2.15 GHz y 2x1.6 GHz, capacidad RAM de 4GB, al que se denominó Dispositivo 1; y un Nexus 6 con sistema operativo Android *Nougat* en su versión 7.0, CPU Qualcomm Snapdragon 805 2.7GHz, capacidad RAM de 3GB, al que se denominó Dispositivo 2.

Para monitorear el comportamiento de los dispositivos durante el uso de la aplicación, se utilizó Android Monitor de Android Studio, el cual permite visualizar, de manera gráfica, el comportamiento de la memoria, CPU, GPU y peticiones de red.

Las pruebas consistieron en la medición del tiempo de respuesta del dispositivo desde el momento en que el usuario selecciona un libro, hasta que la cámara del dispositivo esté disponible para la visualización de los contenidos virtuales del mismo y además, la medición del uso de la memoria y el uso del CPU del dispositivo.

Con la finalidad de establecer una condición base para la medición, se crearon cuatro libros de prueba uno por cada tipo de contenido (imagen, video, audio y objeto 3D), los cuales se denominaron Base 1, Base 2, Base 3 y Base 4 respectivamente. Para evitar que el

marcador influyera en los resultados de estas pruebas, se decidió utilizar el mismo marcador para cada una de éstas. Este marcador se presenta en la Fig. 35. Para Base 1 se utilizó una imagen de 2590x2391 píxeles y 1.5 MB, Base 2 contiene un video de 14.6MB, en Base 3 se utilizó un audio de 2.57MB y para Base 4 se utilizó un objeto de 20.4KB con una textura de 48.9KB.



Figura 35: Marcador usado en las pruebas de control. Figura extraída de [53]

#### 4.1.2. Casos de estudio

Como casos de estudio se crearon 4 libros cuyos contenidos virtuales difieren en cuanto al tipo y la cantidad de los mismos, como se presenta en la Tabla 6.

	Libro 1	Libro 2	Libro 3	Libro 4
Imagen	1 (2,47MB)	2 (1,82MB y 2,47MB)	0	1 (305KB)
Video	1 (14,6MB)	0	2 (14,6MB y 5,58MB)	2 (14,6MB y 5,58MB)
Audio	1 (9,75MB)	2 (2,57MB y 457KB)	0	2 (2,57MB y 457KB)
Objeto 3D	1 (4,62MB)	0	2 (4KB y 20,4KB)	3 (4KB, 4,62MB y 20,4KB)

Tabla 6: Casos de estudio.

### 4.1.3. Resultados

#### 4.1.3.1. Tiempo

	Base 1	Base 2	Base 3	Base 4
Dispositivo 1	1,5 s	1,1 s	2 s	0,9 s
Dispositivo 2	2 s	1,3 s	3 s	1.1 s

Tabla 7: Resultados del tiempo medido de los casos base.

	Libro 1	Libro 2	Libro 3	Libro 4
Dispositivo 1	3 s	4 s	2 s	4 s
Dispositivo 2	3,4 s	5 s	2.5 s	5 s

Tabla 8: Resultados del tiempo medido de los casos de estudio.

En vista de estos resultados, se puede concluir que el tiempo de respuesta del dispositivo depende de la cantidad de contenido virtual del libro seleccionado por el usuario. Además el tipo de contenido que tarda más en cargar es el audio. Esto se debe a que no sólo se debe cargar el archivo de audio, sino además el objeto 3D que se muestra mientras éste se reproduce.

#### 4.1.3.2. Uso de la memoria

Para obtener una base del uso de la memoria, se midió la cantidad de MB de memoria usada por la aplicación antes de entrar a la actividad que permite visualizar los contenidos de realidad aumentada de un libro, ésta estuvo en un rango de 88,56 a 105 MB para el Dispositivo 1 y de 62MB a 88MB para el Dispositivo 2. Cabe destacar que el Dispositivo 2 sólo tiene 3GB de RAM, mientras que el dispositivo 1 tiene 4GB.

	Base 1	Base 2	Base 3	Base 4
Dispositivo 1	83-97MB	83-97MB	83-97MB	83-97MB
Dispositivo 2	62-88MB	62-88MB	62-88MB	62-88MB

Tabla 9: Resultados de la memoria medido de los casos base.

	Libro 1	Libro 2	Libro 3	Libro 4
Dispositivo 1	87-102MB	87-102MB	87-102MB	87-102MB

Dispositivo 2	65-92MB	65-92MB	65-92MB	65-92MB
---------------	---------	---------	---------	---------

Tabla 10: Resultados de la memoria medido de los casos de estudio.

Estos resultados muestran que el sistema operativo asigna cierta cantidad de memoria al dispositivo y sólo lo aumenta cuando es necesario. En las distintas pruebas que se hicieron, una vez iniciada la actividad de visualización de realidad aumentada, la cantidad de memoria asignada a la aplicación aumentó debido a que se utilizan los recursos de la cámara.

#### 4.1.3.3. Uso de la CPU

Para obtener una base del uso de la CPU, se midió el porcentaje usado por la aplicación antes de entrar a la actividad que permite visualizar los contenidos de realidad aumentada de un libro, ésta estuvo en un rango de 4 % y 35 % para el Dispositivo 1 y en un rango de 4 % y 25 % para el Dispositivo 2.

	Base 1	Base 2	Base 3	Base 4
Dispositivo 1	25-38 %	26-40 %	23-40 %	29-41 %
Dispositivo 2	22-32 %	20-28 %	15-25 %	25-33 %

Tabla 11: Resultados del CPU medido de los casos base.

	Libro 1	Libro 2	Libro 3	Libro 4
Dispositivo 1	36-44 %	36-47 %	35-49 %	37-50 %
Dispositivo 2	25-36 %	26-38 %	25-37 %	26-39 %

Tabla 12: Resultados del CPU medido de los casos de estudio.

Durante las pruebas, el uso de la CPU se mantenía siempre en un rango bien definido. La única excepción ocurría cuando se estaba iniciando la actividad de visualización de realidad aumentada, donde el uso de la CPU aumentaba drásticamente debido a la búsqueda de los archivos del contenido virtual en el dispositivo.

## 4.2. Prueba cualitativa (Aplicación web)

Esta prueba se hizo por medio de encuestas, en donde a cada encuestado se le dio un conjunto de tareas que debía completar antes de realizar dicha encuesta. En ésta, se presentan

afirmaciones las cuales fueron evaluadas usando la técnica de Likert [54], la cual permite medir actitudes y conocer el grado de conformidad del encuestado con cualquier afirmación que se le proponga.

Las tareas dadas a cada encuestado fueron las siguientes.

- Registrarse en la aplicación como un nuevo usuario.
- Cerrar sesión e iniciarla de nuevo como un usuario ya existente.
- Seleccionar algún libro de la biblioteca, ver su detalle y descargarlo.
- Entrar al escritorio y actualizar algún dato de su perfil.
- Crear un libro.
- Actualizar la descripción del libro, seleccionando el título en la tabla de libros.
- Dentro de la herramienta de edición del libro, añadir las siguientes páginas.
  - Con contenido de texto.
  - Con un título.
  - Con un título y contenido de texto.
  - Con una imagen cuyo contenido virtual sea una imagen.
  - Con una imagen cuyo contenido virtual sea un video.
  - Con una imagen cuyo contenido virtual sea un audio.
  - Con una imagen cuyo contenido virtual sea un objeto 3D con su textura.
  - Cambiar una de las imágenes al tipo título imagen y añadir el título.
- Borrar la página de su elección.
- Salvar el libro y generar el archivo PDF.
- Publicar el libro.
- Buscar el libro publicado en la biblioteca.

Para cada una de las afirmaciones a evaluar por los encuestados, la escala de evaluación fue la siguiente.

1. Totalmente en desacuerdo.
2. En desacuerdo.
3. Ni de acuerdo ni en desacuerdo.
4. De acuerdo.
5. Totalmente de acuerdo.

Un ejemplo de la encuesta puede verse en la Tabla 13.

	1	2	3	4	5
<b>Funcionalidades básicas</b>					
Fue sencillo registrarme como usuario de la aplicación.					
Fue sencillo iniciar sesión como usuario ya existente.					
Me resultó fácil seleccionar un libro de la biblioteca, ver su detalle y descargarlo.					
No tuve ningún problema actualizando mis datos de usuario.					
<b>Creación del libro</b>					
Fue muy sencillo crear un libro.					
Me resultó fácil actualizar la descripción del libro.					
Fue fácil añadir una página con sólo texto					
Fue sencillo añadir una página con título					
Fue sencillo añadir una página con título y texto					
Me resultó fácil añadir una página con una imagen cuyo contenido virtual sea una imagen.					
Me resultó fácil añadir una página con una imagen cuyo contenido virtual sea un video.					
Me resultó sencillo añadir una página con una imagen cuyo contenido virtual sea un audio.					
Fue fácil añadir una página con una imagen cuyo contenido virtual sea un objeto 3D con su textura.					
Me fue sencillo cambiar una de las imágenes al tipo título imagen y añadir el título.					
Fue sencillo borrar una página del libro.					
Me fue fácil guardar y crear el documento del libro.					
No tuve problemas para publicar el libro.					
Me resultó sencillo buscar mi libro en la biblioteca.					
<b>Consideración sobre este tipo de aplicaciones</b>					
Creo que este tipo de aplicaciones deberían estar al alcance de escritores que quieran crear libros con realidad aumentada.					

Tabla 13: Ejemplo de encuesta

### 4.2.1. Resultados

Se encuestaron a seis (6) personas de distintas edades y profesiones. Como se mencionó anteriormente, cada uno de los encuestados completó la lista de tareas enumeradas anteriormente, antes de responder la encuesta. Ésta fue dividida en tres secciones que se presentan a continuación con los resultados de cada una de ellas.

La primera sección evalúa las funcionalidades básicas de la aplicación web, que incluyen el registro y autenticación de usuario, búsqueda y selección de un libro ya publicado por otro usuario y su descarga. Como puede observarse en el gráfico de la Fig. 36, el cien por ciento (100 %) de los encuestados respondió que estaban completamente de acuerdo les fue fácil utilizar estas funcionalidades.

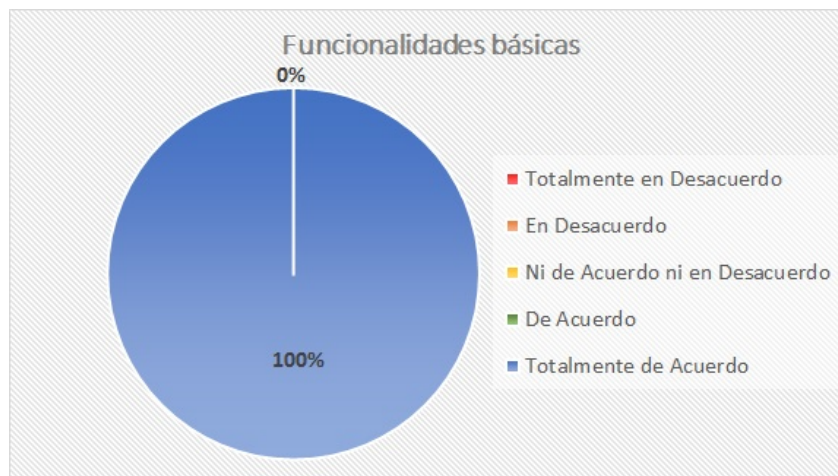


Figura 36: Resultado de la encuesta sobre las funcionalidades básicas del sistema.

La segunda sección evalúa las funcionalidades de la herramienta web para la creación de libros, que incluyen generar una nueva página, añadirle texto, un título o una imagen con contenido virtual. Así como borrar una página, guardar los cambios hechos al libro, crear el documento PDF asociado a éste y posteriormente publicarlo. Como puede observarse en el gráfico de la Fig. 37, el noventa y dos por ciento (92 %) de los encuestados respondió que estuvo totalmente de acuerdo en que les fue sencillo utilizar estas funcionalidades. Y el ocho por ciento (8 %) restante respondió que estaban de acuerdo con las afirmaciones.

La tercera sección evalúa si el encuestado considera que este tipo de aplicaciones deberían estar al alcance de escritores que quieran crear libros con realidad aumentada. El cien por ciento (100 %) de los encuestados respondieron que estaban totalmente de acuerdo con esta afirmación, como puede observarse en la Fig. 38.

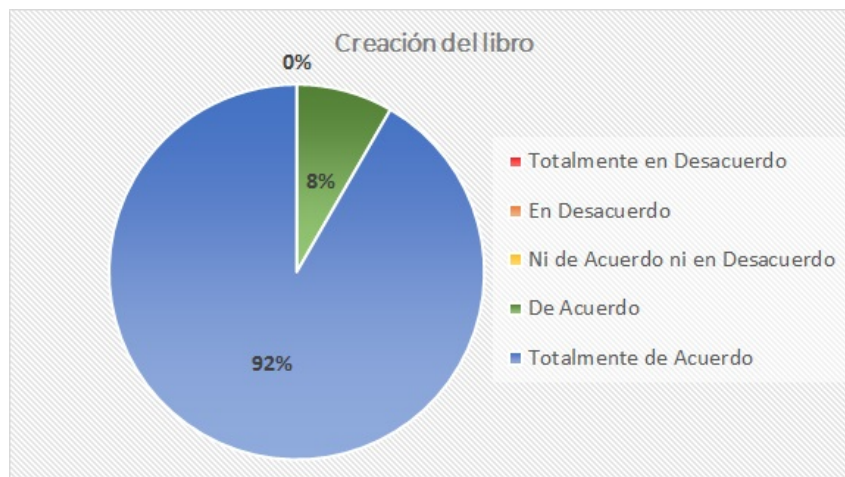


Figura 37: Resultado de la encuesta sobre las funcionalidades de creación del libro.

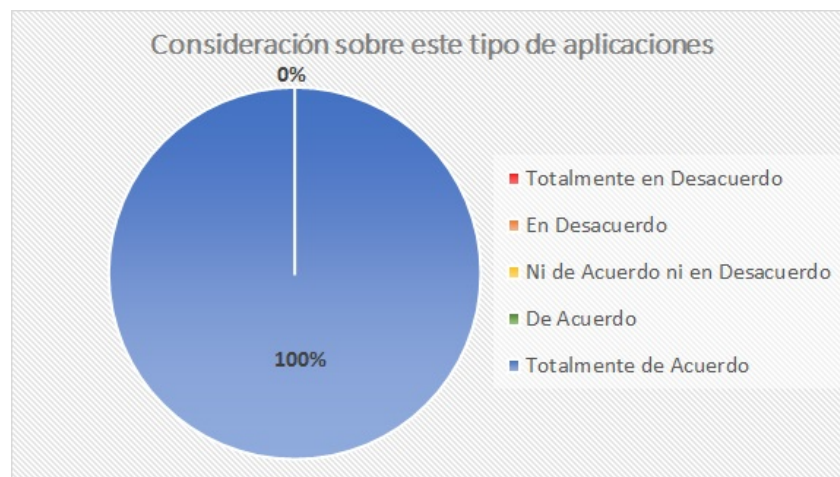


Figura 38: Resultado de la encuesta sobre la consideración sobre la existencia de este tipo de aplicaciones.



# Capítulo 5

## Conclusiones y trabajos futuros

En este capítulo se presentan las conclusiones y los trabajos que pueden desarrollarse a futuro tomando como base este Trabajo Especial de Grado (T.E.G).

### 5.1. Conclusiones

Las aplicaciones desarrolladas, aún con limitaciones, logran cumplir el objetivo general de este T.E.G., desarrollar un sistema para la creación de libros infantiles empleando realidad aumentada. Con la aplicación web, se logró crear, visualizar y editar libros; y con la aplicación móvil se logró visualizar e interactuar con su contenido virtual.

En cuanto al *backend*, se creó un API REST que se comunica con ambas aplicaciones, proveyendo los datos necesarios a las peticiones que éstas le hacen.

En la aplicación web se desarrolló el módulo donde el usuario puede registrarse o autenticarse, el módulo donde puede buscar, visualizar y descargar libros; y se desarrolló la herramienta para crear y editar libros con realidad aumentada.

En la aplicación móvil se desarrolló el modulo de creación y autenticación de usuarios los cuales llevan a la actividad principal que se divide en tres secciones. La primera sección incluye el módulo de librería, en el cual el usuario puede seleccionar los libros que desee descargar. La segunda sección que incluye el módulo de biblioteca, donde el usuario puede seleccionar los libros que ya hayan sido descargados al dispositivo y visualizar el contenido virtual por medio de la cámara de éste. La tercera y última sección que incluye el módulo de perfil de usuario donde éste puede visualizar y actualizar sus datos.

En las pruebas cuantitativas realizadas a la aplicación móvil, ambos dispositivos lograron ejecutar la aplicación sin ningún problema, en ambos se pudieron observar los contenidos virtuales de los libros de prueba y se comportaron de manera similar en cuanto al tiempo, la memoria utilizada y el porcentaje de CPU utilizado.

En cuanto a la prueba cualitativa realizada por medio de encuestas, se obtuvo un alto grado de aceptación por parte de los encuestados en la utilización de la herramienta web para crear libros con realidad aumentada. Además todos concluyeron que deberían existir este tipo de herramientas y estar al alcance de escritores que deseen desarrollar este tipo de libros.

## 5.2. Limitaciones

En el desarrollo de este T.E.G. se encontraron tres limitaciones, la primera fue la falta de una herramienta que permita, a partir de una página web, generar archivos PDF que conserven su estilo. Estos estilos incluyen el tipo de color del texto, su alineación en la página, el estilo del texto (negritas, itálica, etc.) y permita tipos de fuente adicionales a las tres implementadas.

La segunda limitación está relacionada al SDK Kudan, utilizado en el desarrollo de la aplicación móvil. Esta limitación consiste en que este SDK sólo renderiza archivos de objetos 3D de extensión .jet (Android) y .armodel (iOS), las cuales son extensiones de archivos creadas por Kudan. Para hacer uso de un archivo de objetos 3D se requiere entonces obligatoriamente, la utilización de una herramienta de conversión propia de Kudan, la cual, hasta la fecha, sólo funciona en el sistema operativo Windows 7 para la conversión a archivos .jet. Además, sólo se pueden convertir archivos .obj, .fbx y .dae. Éstos dos últimos no pueden tener animaciones asociadas.

Por último, el SDK Kudan sólo renderiza objetos que se encuentren entre -120 y 120 en cada eje de sus vértices. Es decir, que cualquier objeto que exceda estos límites, a medida que la cámara se mueva de posición, el objeto se verá cortado.

## 5.3. Trabajos a futuro

Tomando como base el desarrollo de este T.E.G., a continuación se presentan algunas de las mejoras que se pueden implementar al sistema Typewriter AR.

- Generar una biblioteca o aplicación web, que sea escalable y que permita redactar documentos, con las mismas facilidades de un editor de texto (como la aplicación Word de Microsoft Office) con la cual se puedan generar archivos PDF. Con ello se puede resolver la primera limitación descrita en la sección anterior.
- Mejorar la funcionalidad de escala de los marcadores para sobrescribir ó mantener la proporción del ancho y alto de la imagen.
- Añadir la opción de que no todas las imágenes del libro posean contenido virtual.
- Una vez creado el libro añadir a la primera hoja de éste la imagen de la portada.

- Escalar la aplicación web para diversificar los tipos de producto que se pueden hacer con ella. Por ejemplo carteleras, catálogos, afiches, trípticos, etc.
- Escalar la aplicación móvil para que tenga funcionalidades de una red social, tales como, un servicio que permita a usuarios seguir a los escritores de su elección, un servicio de valoración de libros y que permita a usuarios dejar comentarios, etc.
- Añadir a la aplicación móvil una barra de búsqueda en el módulo de librería que permita buscar libros o autores.
- Desarrollar la aplicación móvil en otros sistemas operativos móviles como iOS.
- Desarrollar una funcionalidad que le permita al usuario interactuar con el contenido virtual. Como por ejemplo, pausar y detener audios y videos; escalar y rotar objetos 3D de manera táctil.
- Implementar en la aplicación web una sección que permita visualizar y editar cómo se verá el contenido virtual sobre el marcador.

# Bibliografía

- [1] Books y Magic. (2016, Oct) Books and magic web page. [En línea]. Disponible en: <http://booksandmagic.com/>
- [2] PopAR. (2016, Oct) Popar web page. [En línea]. Disponible en: <https://popartoy.com/>
- [3] The Pokemon Company. Pokemongo. [En línea]. Disponible en: <http://www.pokemongo.com/>
- [4] D. Levis. (1997/2006) Que es la realidad virtual. [En línea]. Disponible en: [http://www.diegolevis.com.ar/secciones/Articulos/Que\\_es\\_RV.pdf](http://www.diegolevis.com.ar/secciones/Articulos/Que_es_RV.pdf)
- [5] G. Kipper y J. Rampolla, *Augmented Reality: An Emerging Technologies Guide to AR*. Elsevier, 2012.
- [6] O. Baus y S. Bouchard. (2014, Mar) Moving from virtual reality exposure-based therapy to augmented reality exposure-based therapy: A review. [En línea]. Disponible en: [https://www.researchgate.net/publication/260761455\\_Moving\\_from\\_Virtual\\_Reality\\_Exposure-Based\\_Therapy\\_to\\_Augmented\\_Reality\\_Exposure-Based\\_Therapy\\_A\\_Review](https://www.researchgate.net/publication/260761455_Moving_from_Virtual_Reality_Exposure-Based_Therapy_to_Augmented_Reality_Exposure-Based_Therapy_A_Review)
- [7] G. Reitmayr y D. Schmalstieg, “Mobile collaborative augmented reality,” 2001. [En línea]. Disponible en: <http://graphics.cs.columbia.edu/courses/mobwear/resources/reitmayr-isar01.pdf>
- [8] Vassilos Vlahakis et al., “Archeoguide: First results of an augmented reality, mobile computingsystem in cultural heritage sites,” 2001. [En línea]. Disponible en: [https://www.researchgate.net/publication/220955275\\_ARCHEOGUIDE\\_first\\_results\\_of\\_an\\_augmented\\_reality\\_mobile\\_computing\\_system\\_in\\_cultural\\_heritage\\_sites](https://www.researchgate.net/publication/220955275_ARCHEOGUIDE_first_results_of_an_augmented_reality_mobile_computing_system_in_cultural_heritage_sites)
- [9] Microsoft HoloLens. (2017) Sensorama. [En línea]. Disponible en: <https://www.microsoft.com/en-us/hololens>
- [10] Webnode. (2015) Realidad aumentada. [En línea]. Disponible en: <http://realidad-aumentada54.webnode.es/>
- [11] AR-media. I-mechanic, the ar app that turns yourself into a mechanic. [En línea]. Disponible en: <http://www.armedia.it/i-mechanic>

- [12] Snap Inc. Snapchat. [En línea]. Disponible en: <https://www.snapchat.com/>
- [13] C. Graphics. How enlarge parts of a video like snapchat lens filter. [En línea]. Disponible en: <http://computergraphics.stackexchange.com/questions/1939/how-enlarge-parts-of-a-video-like-snapchat-lens-filter>
- [14] Microsoft, “Kinect para xbox one,” 2017. [En línea]. Disponible en: <http://www.xbox.com/es-MX/xbox-one/accessories/kinect>
- [15] E. Makuch, “New xbox one update removes kinect gesture support for dashboard navigation,” 2015. [En línea]. Disponible en: <http://www.gamespot.com/articles/new-xbox-one-update-removes-kinect-gesture-support/1100-6432160/>
- [16] Waze Mobile. Waze. [En línea]. Disponible en: <https://www.waze.com>
- [17] I. Flores, “Que es pokemon go,” 2016. [En línea]. Disponible en: <http://www.aldiadallas.com/2016/07/11/que-es-pokemon-go/>
- [18] A. Ghazarian, “Challenges and opportunities of ux in augmented reality,” 2014. [En línea]. Disponible en: <http://designmodo.com/augmented-reality-ux/>
- [19] Smithsonian National Museum of Natural History. (2015) A hall through new eyes. [En línea]. Disponible en: <http://naturalhistory.si.edu/exhibits/bone-hall/>
- [20] ———. (2017) Smithsonian National Museum of Natural History. [En línea]. Disponible en: <https://naturalhistory.si.edu/>
- [21] NBC NEWS, “Google translate app improves conversation, ‘word lens’ features,” 2015. [En línea]. Disponible en: <http://www.nbcnews.com/tech/mobile/google-translate-app-improves-conversation-word-lens-features-n285656>
- [22] Sony. (2017) Mando de movimiento playstation move. [En línea]. Disponible en: <https://www.playstation.com/es-es/explore/accessories/playstation-move-motion-controller/>
- [23] The Museum of Modern Art of New York, “The museum of modern art of new york,” 2017. [En línea]. Disponible en: [https://www.moma.org/visit/infoplans/spanish\\_plan](https://www.moma.org/visit/infoplans/spanish_plan)
- [24] Vuforia. (2016) Pagina web de vuforia. [En línea]. Disponible en: <https://www.vuforia.com/>
- [25] ARToolKit. (2016) Pagina web de artoolkit. [En línea]. Disponible en: <https://artoolkit.org/>
- [26] Wikitude. (2016) Pagina web de wikitude. [En línea]. Disponible en: <http://www.wikitude.com/>
- [27] AR-media. (2016) Pagina web de ar-media. [En línea]. Disponible en: <http://www.armedia.it/>

- [28] Kudan. (2016) Pagina web de kudan. [En línea]. Disponible en: <https://www.kudan.eu/>
- [29] Kudan. (2016) Look before you leap part ii: Why kudan? [En línea]. Disponible en: <https://www.kudan.eu/kudan-university/look-leap-part-ii-kudan/>
- [30] [En línea]. Disponible en: <https://www.sqlite.org/>
- [31] [En línea]. Disponible en: <https://www.opengl.org/>
- [32] [En línea]. Disponible en: <https://webkit.org/>
- [33] A. J. Vico. (2011, Feb) Arquitectura de android. [En línea]. Disponible en: <https://columna80.wordpress.com/2011/02/17/arquitectura-de-android>
- [34] C. Lin. (2015, Feb) Fixed issues in entertainment connect (android). [En línea]. Disponible en: <https://cuteprogramming.files.wordpress.com/2015/02/android-studio.png>
- [35] JSON.org, “Introducción a json,” May 2017. [En línea]. Disponible en: <http://www.json.org/json-es.html>
- [36] nodejs.org, “node.js,” May 2017. [En línea]. Disponible en: <https://nodejs.org/es/>
- [37] expressjs.com, “express.js,” May 2017. [En línea]. Disponible en: <http://expressjs.com/es/>
- [38] kelektiv, “bcrypt.js,” May 2017. [En línea]. Disponible en: <https://www.npmjs.com/package/bcrypt>
- [39] expressjs, “multer,” May 2017. [En línea]. Disponible en: <https://github.com/expressjs/multer>
- [40] Oracle Corporation, “Mysql,” May 2017. [En línea]. Disponible en: <https://www.mysql.com/>
- [41] bookshelfjs.org, “bookshelf.js,” May 2017. [En línea]. Disponible en: <http://bookshelfjs.org/>
- [42] Google, “AngularJS,” May 2017. [En línea]. Disponible en: <https://angularjs.org/>
- [43] M. Otto, “Bootstrap,” May 2017. [En línea]. Disponible en: <http://getbootstrap.com/>
- [44] Angular UI Team, “Ui bootstrap,” May 2017. [En línea]. Disponible en: <https://angular-ui.github.io/bootstrap/>
- [45] J. Hall, “Mysql,” May 2017. [En línea]. Disponible en: <https://github.com/MrRio/jsPDF>
- [46] D. Farid, “Mysql,” May 2017. [En línea]. Disponible en: <https://github.com/danialfarid/ng-file-upload>

- [47] Kudan, “Kudan toolkit,” May 2017. [En línea]. Disponible en: <https://www.kudan.eu/download-kudan-ar-sdk/>
- [48] Kudan wiki, “3d models,” May 2017. [En línea]. Disponible en: [https://wiki.kudan.eu/3D\\_Models#Supported\\_3D\\_formats](https://wiki.kudan.eu/3D_Models#Supported_3D_formats)
- [49] Kudan, “Android AR SDK kudan,” May 2017. [En línea]. Disponible en: [https://wiki.kudan.eu/3D\\_Models#Supported\\_3D\\_formats](https://wiki.kudan.eu/3D_Models#Supported_3D_formats)
- [50] Google, Android Developers, “Transmitting network data using volley,” May 2017. [En línea]. Disponible en: <https://developer.android.com/training/volley/index.html>
- [51] Google, “A java serialization/deserialization library to convert java objects into json and back,” May 2017. [En línea]. Disponible en: <https://github.com/google/gson>
- [52] Kudan Wiki, “Getting started android,” May 2017. [En línea]. Disponible en: [https://wiki.kudan.eu/Getting\\_Started\\_Android](https://wiki.kudan.eu/Getting_Started_Android)
- [53] L. Lopez, “Las mejores apps de star wars para android e iphone,” Dec 2015. [En línea]. Disponible en: <http://www.hobbyconsolas.com/reportajes/las-mejores-apps-star-wars-para-android-e-iphone-133508>
- [54] O. Llauro, “La escala de likert: que es y como utilizarla,” Dec 2014. [En línea]. Disponible en: <https://www.netquest.com/blog/es/blog/es/la-escala-de-likert-que-es-y-como-utilizarla>
- [55] P. Milgram, h. Takemura, A. Utsumi, y F. Kishino, “Augmented reality: A class of displays on the reality-virtuality continuum,” 1994. [En línea]. Disponible en: [http://etclab.mie.utoronto.ca/publication/1994/Milgram\\_Takeura\\_SPIE1994.pdf](http://etclab.mie.utoronto.ca/publication/1994/Milgram_Takeura_SPIE1994.pdf)
- [56] Lindsey. (2015, Oct) Virtual reality vs. augmented reality. [En línea]. Disponible en: <http://www.augment.com/blog/virtual-reality-vs-augmented-reality/>
- [57] Augment. (2016, May) Infographic: The history of augmented reality. [En línea]. Disponible en: <http://www.augment.com/blog/infographic-lengthy-history-augmented-reality/>
- [58] C. Trilnick. Sensorama. [En línea]. Disponible en: <http://proyectoidis.org/sensorama/>
- [59] David. (2010, Mar) Myron krueger. [En línea]. Disponible en: <http://www.inventinginteractive.com/2010/03/22/myron-krueger/>
- [60] T. Caudell, “Ar at boeing,” 1990. [En línea]. Disponible en: <http://www.idemployee.id.tue.nl/g.w.m.rauterberg/presentations/hci-history/tsld096.htm>
- [61] D. Davalos, “Desarrollo de publicidad utilizando realidad aumentada para estrenos cinematograficos en la ciudad de riobamba,” 2013. [En línea]. Disponible en: <http://dspace.epoch.edu.ec/bitstream/123456789/3151/1/78T00126.pdf>

- [62] Digital Tourism Think Tank, “Augmented reality in tourism 10 unique applications explained,” 2013. [En línea]. Disponible en: <https://thinkdigital.travel/wp-content/uploads/2013/04/10-AR-Best-Practices-in-Tourism.pdf>
- [63] D. Kazovskaja, “Top 5 tools for creating augmented reality apps,” Mar 2016. [En línea]. Disponible en: <http://cases.azoft.com/top-5-tools-creating-augmented-reality-apps/>
- [64] P. e. a. Blanco. (2009) Metodología de desarrollo agil para sistemas moviles introduccion al desarrollo con android y el iphone. [En línea]. Disponible en: [http://www.adamwesterski.com/wp-content/files/docsCursos/Agile\\_doc\\_TemasAnv.pdf](http://www.adamwesterski.com/wp-content/files/docsCursos/Agile_doc_TemasAnv.pdf)
- [65] A. Vico. (2011, Feb) Arquitectura de android. [En línea]. Disponible en: <https://columna80.wordpress.com/2011/02/17/arquitectura-de-android/>
- [66] Wikipedia. (2016, Ago) Android version history. [En línea]. Disponible en: [https://en.wikipedia.org/wiki/Android\\_version\\_history](https://en.wikipedia.org/wiki/Android_version_history)
- [67] Android. (2016, Ago) Meet android studio. [En línea]. Disponible en: <https://developer.android.com/studio/intro/index.html>
- [68] J. Eason. (2016, Abr) Android studio 2.0. [En línea]. Disponible en: <http://android-developers.blogspot.com/2016/04/android-studio-2-0.html>
- [69] Android. (2016, Ago) Everything you need to build on android. [En línea]. Disponible en: <https://developer.android.com/studio/features.html>
- [70] A. Mullis. (2016, Jun) Android ndk, everything you need to know. [En línea]. Disponible en: <http://www.androidauthority.com/android-ndk-everything-need-know-677642/>
- [71] JetBrains. (2016, Ago) JetBrains web page. [En línea]. Disponible en: <https://www.jetbrains.com/>
- [72] SpellBound. (2016, Oct) Spellbound web page. [En línea]. Disponible en: <http://getspellboundbooks.com/>
- [73] Sony. (2016, Oct) Wonderbook web page. [En línea]. Disponible en: <https://www.playstation.com/es-es/games/wonderbook-book-of-spells-ps3>
- [74] Dealspwn. (2016, Oct) Dealspwn web page. [En línea]. Disponible en: <http://www.dealspwn.com/wonderbook-book-of-spells-bundle-diggs-nightcrawler-zavvi-ps3-139186>
- [75] Carlton Publishing Group. (2016, Oct) Carlton publishing group web page. [En línea]. Disponible en: <http://www.carltonbooks.co.uk/books/products?keywords=augmented+reality>
- [76] Carlton Publishing Group. (2016, Oct) Monsters inc. realidad aumentada. [En línea]. Disponible en: <https://www.youtube.com/watch?v=PRaEDM1qE0k>



- [77] ZapWorks. (2016, Oct) Zapworks web page. [En línea]. Disponible en: <https://zap.works/>
- [78] Zappar. (2016, Oct) Zappar web page. [En línea]. Disponible en: <https://www.zappar.com>
- [79] H. Aurasma. (2016, Oct) Aurasma web page. [En línea]. Disponible en: <https://www.aurasma.com/>
- [80] H. Altinpulluk y M. Kesim, “The classification of augmented reality books: A literature review,” Mar 2016. [En línea]. Disponible en: [https://www.academia.edu/23286142/THE\\_CLASSIFICATION\\_OF\\_AUGMENTED\\_REALITY\\_BOOKS\\_A\\_LITERATURE\\_REVIEW](https://www.academia.edu/23286142/THE_CLASSIFICATION_OF_AUGMENTED_REALITY_BOOKS_A_LITERATURE_REVIEW)
- [81] Universidad Simón Bolívar, Venezuela., “Desarrollo agil de software, caso programacion extrema xp,” Jul 2016. [En línea]. Disponible en: <http://ldc.usb.ve/~abianc/materias/ci4713/metodologiasagiles.pdf>