

Pontificia Universidad Javeriana Cali  
Facultad de Ingeniería y Ciencias.  
Ingeniería de Sistemas y Computación.  
Proyecto de Grado.

Aplicativo móvil para apoyar la inmersión en ambientes de realidad  
aumentada por medio de sonido 3D

Iván Darío Bedoya Suárez  
Diego Andrés Vinasco Trujillo

Director: Dr. Gerardo Mauricio Sarria

Enero 6 de 2025



Santiago de Cali, Enero 6 de 2025.

Señores

**Pontificia Universidad Javeriana Cali.**

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

Cali.

Cordial Saludo.

Por medio de la presente me permito informarle que los estudiantes de Ingeniería de Sistemas y Computación Iván Darío Bedoya Suárez (cod: 8936181) y Diego Andrés Vinasco Trujillo(cod: 8933924) trabajan bajo mi dirección en el proyecto de grado titulado “Aplicativo móvil para apoyar la inmersión en ambientes de realidad aumentada por medio de sonido 3D”.

Atentamente,

A handwritten signature in black ink, consisting of stylized cursive letters that appear to read 'GMSM'. The signature is written above a horizontal line.

Dr. Gerardo Mauricio Sarria

Santiago de Cali, Enero 6 de 2025.

Señores

**Pontificia Universidad Javeriana Cali.**

Dr. Gerardo Mauricio Sarria

Director Carrera de Ingeniería de Sistemas y Computación.

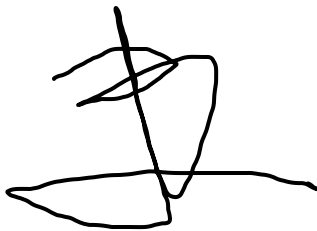
Cali.

Cordial Saludo.

Nos permitimos presentar a su consideración el proyecto de grado titulado “Aplicativo móvil para apoyar la inmersión en ambientes de realidad aumentada por medio de sonido 3D” con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

Al firmar aquí, damos fe que entendemos y conocemos las directrices para la presentación de trabajos de grado de la Facultad de Ingeniería aprobadas el 26 de Noviembre de 2009, donde se establecen los plazos y normas para el desarrollo del proyecto y del trabajo de grado.

Atentamente,



---

Iván Darío Bedoya Suárez  
Código: 8936181



---

Diego Andrés Vinasco Trujillo  
Código: 8933924

# Resumen

- **English:**

The hearing is one of the fundamental human senses that, next with sight, gives us a sense of location and to perceive all our surroundings, so using these senses in a system or software, greater user immersion can be achieved, improving or assisting in different situations.

This document explores the integration of three-dimensional (3D) sound in augmented reality (AR) applications and proposes the development of a mobile application that uses these technologies to enhance user experience. Augmented reality has advanced significantly, but to maximize its immersive potential, it's essential to incorporate auditive elements that complement the visual ones. 3D sound, when simulating the origin and distance of sounds in a three-dimensional environment, allows greater immersion and more natural interaction in AR applications [1].

The proposal focus on developing a mobile application that use 3D sound to improve the user experience in different situations, from entertainment to assist people with visual disabilities. This approach not only promotes technological development, but also has the potential to improve users quality of life by offering richer and more immersive experiences. The presented research examines the benefits of 3D sound in AR, its practical applications, and how this technology can influence the future of human-machine interfaces.

- **Español:**

La audición es uno de los sentidos fundamentales del ser humano que, junto con la vista, nos logran dar el sentido de ubicación y percibir todo nuestro alrededor, por lo que al hacer uso de estos sentidos en un sistema o software se puede lograr una mayor inmersión por parte del usuario permitiendo mejorar o ayudar en ciertas ocasiones.

Este documento explora la integración del sonido tridimensional (3D) en aplicaciones de realidad aumentada (AR) y propone el desarrollo de un aplicativo móvil que utilice estas tecnologías para mejorar la experiencia del usuario. La realidad aumentada ha avanzado significativamente, pero para maximizar su potencial inmersivo, es fundamental incorporar elementos auditivos que complementen las visuales. El sonido 3D, al simular la procedencia y distancia de los sonidos en un entorno tridimensional, permite una mayor inmersión y una interacción más natural en aplicaciones AR [1].

La propuesta se centra en la creación de un aplicativo móvil que utilice sonido 3D para mejorar la experiencia de usuario en diversas situaciones, desde entretenimiento hasta asistencia a personas con discapacidades visuales. Este enfoque no solo promueve el desarrollo tecnológico, sino que también tiene el potencial de mejorar la calidad de vida de los usuarios, ofreciendo experiencias más ricas y envolventes. La investigación presentada examina los beneficios del sonido 3D en AR, sus aplicaciones prácticas, y cómo esta tecnología puede influir en el futuro de las interfaces hombre-máquina.

# Introducción

La convergencia de tecnologías móviles, realidad aumentada (AR) y sonido tridimensional (3D) ha abierto nuevas posibilidades para crear experiencias interactivas más inmersivas y útiles. En este contexto, el desarrollo de aplicaciones móviles que aprovechan estas tecnologías se está convirtiendo en un área de interés clave para la innovación y la mejora de la usabilidad en diversas situaciones cotidianas[2].

Este documento propone el diseño y desarrollo de una aplicación móvil que utiliza la cámara del dispositivo y un objeto que emite sonido desde la aplicación para que el usuario pueda ubicar su posición utilizando sonido 3D. Esta tecnología permite a los usuarios percibir la dirección y la distancia del objeto en un espacio tridimensional, recreando una experiencia auditiva realista que se basa en la percepción humana natural del sonido.

El uso de sonido 3D en esta aplicación no solo aumenta la precisión con la que los usuarios pueden localizar objetos en su entorno, sino que también tiene potenciales aplicaciones en campos como la asistencia a personas con discapacidades visuales, la seguridad, y la navegación en entornos complejos.[3] Al integrar estas tecnologías, se espera mejorar la accesibilidad y proporcionar nuevas herramientas para la interacción con el entorno físico y digital. Este documento explora los principios técnicos detrás del sonido 3D, las capacidades de la realidad aumentada en dispositivos móviles, y el proceso de desarrollo de la aplicación propuesta.

# Descripción del Problema

---

## 1.1. Planteamiento del Problema

La realidad aumentada (AR) ha demostrado ser una tecnología innovadora con un potencial significativo para transformar la manera en que interactuamos con el mundo. Un ejemplo notable es Pokémon Go, un juego que revolucionó la industria de los videojuegos móviles al combinar AR con geolocalización, permitiendo a los usuarios capturar criaturas virtuales en el mundo real. Sin embargo, la interacción en Pokémon Go y aplicaciones similares se ha centrado principalmente en la visualización y el tacto, dejando de lado el sentido auditivo como una dimensión clave para la inmersión y la experiencia del usuario [4].

A pesar de los avances en AR, existe una brecha significativa en la integración del sonido tridimensional (3D) en aplicaciones móviles. El sonido 3D ofrece la capacidad de simular la procedencia del sonido desde diferentes direcciones y distancias en un entorno tridimensional, replicando cómo los humanos perciben el sonido en la vida real [5]. Esta tecnología podría mejorar considerablemente la experiencia de usuario en aplicaciones de AR al proporcionar una dimensión adicional de inmersión.

El problema central que se aborda es la falta de aplicaciones móviles que utilicen de manera efectiva el sonido 3D para mejorar la localización y la interacción con objetos virtuales en entornos de AR. Mientras que aplicaciones como Pokémon Go han demostrado el potencial de la AR, la incorporación del sonido 3D podría llevar estas experiencias a un nuevo nivel, permitiendo a los usuarios no solo ver, sino también escuchar y localizar objetos con mayor precisión. Esto es especialmente relevante en contextos donde la visibilidad es limitada o cuando se desea mejorar la accesibilidad para personas con discapacidades visuales.

Este planteamiento subraya la necesidad de investigar y desarrollar aplicaciones móviles que integren de manera efectiva el sonido 3D con AR, superando las limitaciones actuales y explorando nuevas formas de interacción y usabilidad. La propuesta es precisamente abordar esta necesidad mediante el diseño de un aplicativo móvil que permita a los usuarios localizar objetos a través del sonido 3D, mejorando así la experiencia inmersiva y ampliando las posibilidades de aplicación de la realidad aumentada en diferentes contextos.

### 1.1.1. Formulación

¿Cómo desarrollar una aplicación móvil con sonido 3D para apoyar la inmersión de los usuarios en realidad aumentada?.

### 1.1.2. Sistematización

- ¿Cuáles son las principales características de la aplicación móvil a desarrollar?
- ¿Que algoritmos y librerías de realidad aumentada para Android , pueden contribuir a un avance o mejora para la inmersión de los usuarios?
- ¿Cómo desarrollar una aplicación móvil que ponga un objeto virtual en un espacio físico y se pueda localizar por medio del sonido 3D?
- ¿Cómo evaluar que la espacialización de sonido en la aplicación apoya la inmersión de los usuarios?

### 1.1.3. Aplicaciones relacionadas

En el área de la realidad aumentada, se han desarrollado distintas aplicaciones con enfoques variados, como la accesibilidad o apoyo a las personas con discapacidad visual, ofreciendo una asistencia que suple las limitaciones de estos usuarios.

Aplicaciones como e-Glance ofrecen soluciones innovadoras que implementan la realidad aumentada, pero no una integración con el sonido 3D, lo que genera oportunidades de mejora a las funcionalidades. Dicha aplicación fue diseñada para facilitar la navegación a personas invidentes por medio de vibraciones y sonidos particulares; sin embargo, esta especialización generada podría optimizarse haciendo uso del sonido 3D, apoyando la orientación del usuario y complementando la información que brinda la aplicación para finalmente guiar al usuario de manera más ágil e intuitiva.

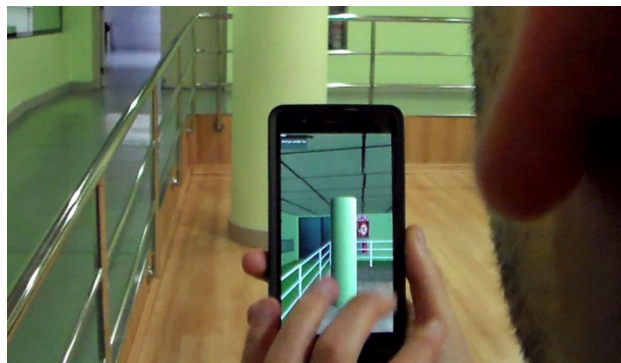


Figura 1.1: Logo e-Glance

## 1.2. Objetivos

### 1.2.1. Objetivo General

Desarrollar una aplicación móvil que permita a los usuarios identificar la posición de un objeto virtual situado en el mundo real por medio del sonido 3D .

### 1.2.2. Objetivos Específicos

- Definir las características del aplicativo móvil a desarrollar.
- Diseñar el sistema (algoritmos y librerías) de realidad aumentada para Android que serán usados en la aplicación.
- Implementar un aplicativo móvil que permita identificar la posición de un objeto virtual en un espacio físico a través del sonido 3D.
- Evaluar qué tan acertada es la espacialización del sonido en la inmersión de los usuarios.

## 1.3. Justificación

La integración de la tecnología de sonido tridimensional (3D) en aplicaciones de realidad aumentada (AR) móviles presenta una oportunidad única para enriquecer la interacción del usuario con el entorno virtual, ofreciendo una experiencia más completa y envolvente. Esta propuesta se justifica por varias razones clave que subrayan la importancia y la necesidad de esta innovación tecnológica.

- **Mejora de la Experiencia del Usuario:** El sonido 3D puede transformar significativamente la percepción del espacio virtual, ofreciendo una experiencia más realista y envolvente. Esta mejora es crucial para aplicaciones como juegos, entrenamiento virtual, educación, y asistencia a personas con discapacidades visuales, donde la percepción espacial juega un papel fundamental.
- **Innovación Tecnológica:** A pesar del éxito de juegos como Pokémon Go, la AR ha sido predominantemente visual. Integrar sonido 3D es un paso hacia la innovación que puede establecer nuevos estándares para el desarrollo de aplicaciones móviles y abrir nuevas vías para la investigación y el desarrollo en la tecnología de AR.
- **Accesibilidad y Usabilidad:** Implementar sonido 3D en aplicaciones de AR puede hacer estas tecnologías más accesibles e intuitivas para un rango más amplio de usuarios, incluidos aquellos con discapacidades visuales. El sonido espacial permite a los usuarios localizar objetos y navegar en entornos virtuales con más precisión, basándose en señales auditivas.

- **Potencial Comercial y de Mercado:** La creciente popularidad de la AR en varios sectores ofrece un mercado expansivo para aplicaciones innovadoras. Incorporar sonido 3D podría diferenciar un producto en el mercado competitivo, atrayendo a usuarios y potenciales inversores interesados en nuevas tecnologías.
- **Contribución al Campo Académico y Científico:** El desarrollo y estudio del sonido 3D en AR proporciona material valioso para la investigación académica, contribuyendo al conocimiento y la comprensión de las interacciones humanas con entornos digitales aumentados. Esto es especialmente relevante en campos como la psicoacústica, la informática y la ingeniería de software.

#### 1.4. Delimitaciones y Alcances

- El tiempo estimado para diseñar, codificar y documentar el aplicativo es de 6 meses.
- La aplicación será desarrollada solo para sistema operativo Android de la versión 8.0.0 hacia adelante.
- El aplicativo se limita a una sola fuente de sonido (la emitida por la aplicación), es decir un solo objeto.
- La fuente que emite el sonido permanece estática.
- Los usuarios únicamente controlan la posición 3D por medio de la cámara del dispositivo.
- La aplicación será desarrollada para que el sonido sea percibido por medio de auriculares .

##### 1.4.1. Entregables

- Una aplicación funcional para dispositivos Android que permite que un usuario identifique la posición de un objeto virtual en el mundo real por medio del sonido.
- Un manual de usuario de la aplicación.
- Código fuente Documentado.
- Documentación de librerías y algoritmos usados para el desarrollo de la aplicación , enfocado principalmente en la parte auditiva.
- Un repositorio con el código fuente y toda la documentación mencionada anteriormente.

# Desarrollo del Proyecto

---

## 2.1. Marco de Referencia

### 2.1.1. Áreas Temáticas

En esta sección se enuncian áreas afines al problema tratado que se encuentran en la ACM Computing Classification System (ACM CCS)[Referencia], enfocándose principalmente en sonido, realidad aumentada y creación de software.

- **Software and its engineering** - Software creation and management
- **Human-centered computing** - Human computer interaction (HCI) - Interaction devices - Sound-based input / output
- **Human-centered computing** - Human computer interaction (HCI) - Interaction paradigms - Mixed / augmented reality
- **Human-centered computing** - Ubiquitous and mobile computing - Ubiquitous and mobile devices - Smartphones

### 2.1.2. Marco Teórico

El desarrollo de una investigación que explora la integración de sonido 3D en aplicaciones de realidad aumentada (AR) móviles abarca varias áreas clave para establecer una base sólida de conocimientos. A continuación, se describen los principales componentes teóricos que fundamentan este enfoque:

- **Realidad Aumentada (AR):** Es fundamental comprender los fundamentos y la evolución de la AR, que superpone información digital al mundo real y se ha utilizado en diversas aplicaciones, desde juegos hasta aplicaciones industriales. La AR se diferencia de la realidad virtual porque complementa, en lugar de reemplazar, el entorno real con elementos virtuales[5].
- **Interacción Humano-Computadora (HCI):** Analizar cómo los usuarios interactúan con tecnologías AR y sonido 3D es esencial para entender la usabilidad y la experiencia del usuario, abarcando métodos de entrada y salida en AR, como gestos y seguimiento ocular[6].
- **Psicoacústica:** Este campo estudia la percepción humana del sonido y es vital para entender cómo las personas localizan fuentes de sonido en un ambiente y cómo estas percepciones pueden ser replicadas o alteradas mediante tecnología para crear experiencias más inmersivas[6].

- **Tecnologías de Apoyo y Accesibilidad:** Es importante investigar cómo el sonido 3D en AR puede mejorar la accesibilidad para personas con discapacidades, mejorando su movilidad y independencia en la vida diaria[6].
- **Desarrollo de Aplicaciones Móviles:** Revisar las tecnologías actuales de desarrollo de aplicaciones móviles, así como los marcos y herramientas que soportan AR y sonido 3D, proporciona un contexto práctico para la implementación de los conceptos teóricos[6].

### 2.1.3. Trabajos Relacionados

En el ámbito de las aplicaciones móviles que utilizan sonido 3D junto con realidad aumentada, hay menos ejemplos que en el sector del juego o el entretenimiento audiovisual puro, como las películas o la música. Sin embargo, algunas aplicaciones han empezado a explorar esta integración para mejorar la experiencia de usuario:

- **RjDj:** Aunque ya no está activa, esta aplicación fue pionera en el uso de sonido 3D para crear experiencias auditivas inmersivas que respondían al entorno del usuario en tiempo real. Utilizaba el sonido del entorno capturado por los micrófonos del dispositivo para modificar la música y los efectos de sonido, creando una experiencia de audio altamente personalizada y contextual[7].
- **Minecraft Earth:** En este juego de AR, los sonidos del juego se reproducen desde ubicaciones virtuales en el mundo real, creando una experiencia más inmersiva. Por ejemplo, el sonido de un zombie en Minecraft Earth parece provenir de su ubicación en el mundo virtual superpuesto al real[8].



Figura 2.1: Logo Minecraft

- **Apps de navegación para invidentes:** Aplicaciones como "Soundscape" de Microsoft utilizan sonido 3D para ayudar a las personas con visibilidad reducida a navegar por el espacio

urbano. La aplicación ofrece información auditiva sobre el entorno, usando señales de audio espaciales para indicar la dirección de los puntos de interés[9].

# Sonido 3D

---

## 3.1. Qué es el sonido espacial

El sonido espacial es una tecnología y concepto fundamental en la creación de experiencias auditivas inmersivas, ya que busca replicar cómo los seres humanos perciben el sonido en un entorno tridimensional. En la naturaleza, el sonido no solo llega desde un plano horizontal, sino también desde diferentes alturas y profundidades. La percepción del sonido en estas dimensiones es clave para interpretar nuestro entorno, localizar objetos y generar una experiencia sensorial completa. Esta sección analiza los principios, tecnologías y aplicaciones del sonido espacial, destacando su relevancia en contextos como la realidad virtual, aumentada y otros sistemas tecnológicos avanzados.

### 3.1.1. Definición de Sonido Espacial

El sonido espacial se refiere a la reproducción del audio con características tridimensionales, donde los sonidos parecen emanar de ubicaciones específicas en el espacio. A diferencia del sonido estéreo, que se limita a un plano bidimensional, o el sonido envolvente tradicional (surround), que abarca direcciones horizontales, el sonido espacial introduce una dimensión vertical que amplía las posibilidades perceptivas del usuario. Esto incluye la percepción de sonidos provenientes de arriba, abajo, adelante, atrás y a los lados, creando un entorno sonoro más realista y envolvente.

### 3.1.2. Fundamentos Técnicos del Sonido Espacial

El desarrollo del sonido espacial se basa en principios acústicos y tecnologías avanzadas que buscan imitar la manera en que los humanos escuchamos en un entorno real:

- **HRTF (Head-Related Transfer Function)** Un modelo matemático que describe cómo las características físicas de la cabeza, los oídos y el torso afectan la percepción del sonido. Esto permite simular con precisión cómo el sonido es modificado antes de llegar al tímpano.
- **Audio binaural:** - Técnica que utiliza micrófonos especializados o procesamiento digital para recrear cómo el sonido llega a cada oído de forma natural, logrando una experiencia tridimensional con auriculares.
- **Procesamiento espacial** En sistemas digitales, se emplean algoritmos para manipular las propiedades del sonido (frecuencia, intensidad y tiempo) de manera que el oyente perciba ubicaciones específicas de las fuentes sonoras.

### 3.1.3. Aplicaciones del Sonido Espacial

El sonido espacial ha transformado múltiples campos tecnológicos y creativos al ofrecer experiencias auditivas más inmersivas y realistas. Algunas de sus aplicaciones más relevantes incluyen:

- **Realidad Virtual y Aumentada (VR/AR):** En estos entornos, el sonido espacial complementa la experiencia visual al proporcionar un contexto auditivo que refleja la posición y el movimiento de objetos virtuales en tiempo real.
- **Videojuegos:** Mejora la jugabilidad al permitir que los jugadores identifiquen con precisión la ubicación de enemigos, objetos o eventos en un espacio tridimensional.
- **Cine y Entretenimiento:** Tecnologías como Dolby Atmos utilizan canales de altura para generar una experiencia cinematográfica inmersiva, donde el espectador puede percibir sonidos como si estuvieran dentro de la escena.
- **Música:** La música inmersiva aprovecha el sonido espacial para distribuir los instrumentos y efectos en un espacio tridimensional, ofreciendo una nueva forma de experimentar las composiciones.
- **Entrenamiento y Simulación:** En simuladores militares, médicos o de aviación, el sonido espacial ayuda a recrear entornos realistas para entrenamientos más efectivos.

### 3.1.4. Ventajas y Desafíos del Sonido Espacial

El sonido espacial presenta ventajas significativas, como una mayor inmersión sensorial, mejor precisión en la localización de fuentes sonoras y la posibilidad de crear experiencias únicas y memorables. Sin embargo, también enfrenta desafíos técnicos, como la necesidad de dispositivos específicos (auriculares compatibles, sistemas de altavoces avanzados) y la complejidad en el diseño y procesamiento del audio.

## 3.2. HRTF

El HRTF (Head-Related Transfer Function, o Función de Transferencia Relacionada con la Cabeza) es un modelo matemático y acústico que describe cómo las características físicas de la anatomía humana afectan las propiedades del sonido antes de llegar a los tímpanos. Este modelo encapsula las transformaciones que experimenta una onda sonora al interactuar con la cabeza, las orejas, el torso y otras estructuras del cuerpo humano, permitiendo que el cerebro interprete la posición, distancia y dirección de las fuentes sonoras en un espacio tridimensional (HRTF, 2019)[1].

Desde un punto de vista técnico, el HRTF se define como una función que relaciona una fuente sonora específica en el espacio con el sonido percibido en los tímpanos, considerando los efectos de:

- **La forma y tamaño de la cabeza:** Actúa como una barrera que bloquea o atenúa frecuencias específicas, especialmente las altas, dependiendo de la dirección de la fuente sonora.

Este fenómeno, conocido como efecto de sombra acústica, es clave para identificar la dirección horizontal de los sonidos.

- **El torso y los hombros:** Modifican las ondas sonoras al causar reflexiones y difracciones adicionales, influyendo en la percepción de la profundidad y la altura.
- **El pabellón auricular (oído externo)** Su estructura única introduce resonancias y atenuaciones que afectan la respuesta en frecuencia del sonido. Esto es fundamental para identificar la ubicación vertical de una fuente sonora.
- **La distancia entre los oídos (Interaural Time Difference, ITD):** Representa la diferencia de tiempo con la que el sonido llega a cada oído.
- **La diferencia de nivel entre los oídos (Interaural Level Difference, ILD):** Corresponde a la disparidad en la intensidad del sonido percibida por cada oído debido a la dirección de la fuente.

En términos funcionales, el HRTF se representa matemáticamente como un conjunto de filtros de respuesta en frecuencia que varían en función de la ubicación de la fuente sonora en un sistema de coordenadas tridimensional. Estos filtros, aplicados al sonido original, generan una “firma acústica” específica que el cerebro interpreta para localizar la fuente.

### 3.2.1. Importancia

El HRTF es esencial para la percepción del espacio sonoro en humanos. Sin estas modificaciones naturales del sonido, no seríamos capaces de localizar con precisión las fuentes sonoras en nuestro entorno. El cerebro utiliza las señales acústicas procesadas por el HRTF para determinar la dirección (azimut y elevación), la distancia y el contexto espacial de los sonidos. Este procesamiento es crucial para tareas cotidianas como identificar el origen de una voz en una multitud o percibir el peligro en situaciones de alerta.

### 3.2.2. Modelo Matematico

El Head-Related Transfer Function (HRTF) se modela como una función de transferencia dependiente de la posición de una fuente sonora en el espacio tridimensional. Matemáticamente, la HRTF puede representarse como un sistema de filtros que relacionan la señal de entrada (el sonido original) con la señal que llega a cada tímpano, incorporando los efectos de reflexión, difracción y atenuación introducidos por las características anatómicas del individuo.

### 3.2.3. Definición Formal

El HRTF se define como la relación entre la presión acústica en la fuente sonora  $P_{source}(f)$  y la presión acústica en el tímpano  $P_{ear}(f)$  y depende de:

1. La frecuencia del sonido  $f$ .

2. La posición espacial de la fuente sonora, descrita por el vector  $(\theta, \phi, r)$  donde:

- a)  $\theta$  Ángulo azimutal (posición horizontal).
- b)  $\phi$  Ángulo de elevación (posición vertical).
- c)  $r$  Distancia entre la fuente y el oyente.

Se expresa como:

$$HRTF(\theta, \phi, r, f) = \frac{P_{ear}(f)}{P_{source}(f)}$$

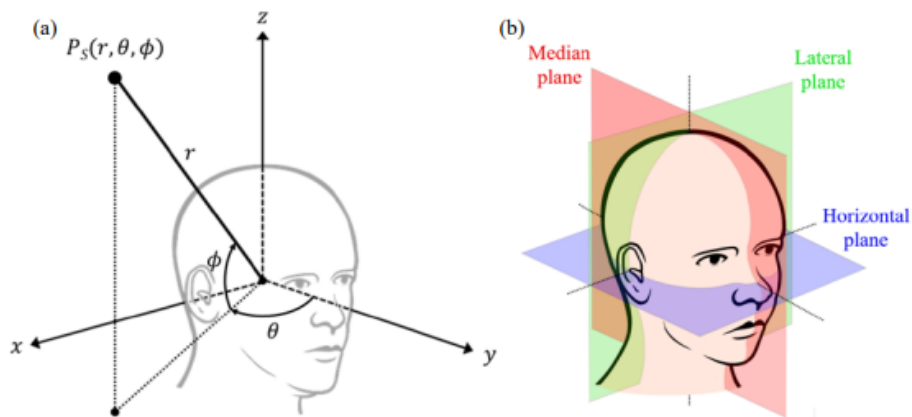


Figura 3.1: Modelo transformación del sonido[27]

Donde:

- a)  $P_{source}(f)$ : Espectro de la señal emitida por la fuente.
- b)  $P_{ear}(f)$ : Espectro de la señal percibida en el oído tras ser modificada por la interacción con la anatomía del oyente.

### 3.2.4. Modelo como Filtro Lineal

El HRTF puede interpretarse como un sistema lineal y estacionario en el dominio de la frecuencia, y por tanto, su operación sobre un sonido de entrada  $x(t)$  puede modelarse mediante convolución:

$$y(t) = h(t) * x(t)$$

En el dominio de la frecuencia, la convolución se traduce en un producto directo:

$$Y(f) = H(f) \cdot X(f)$$

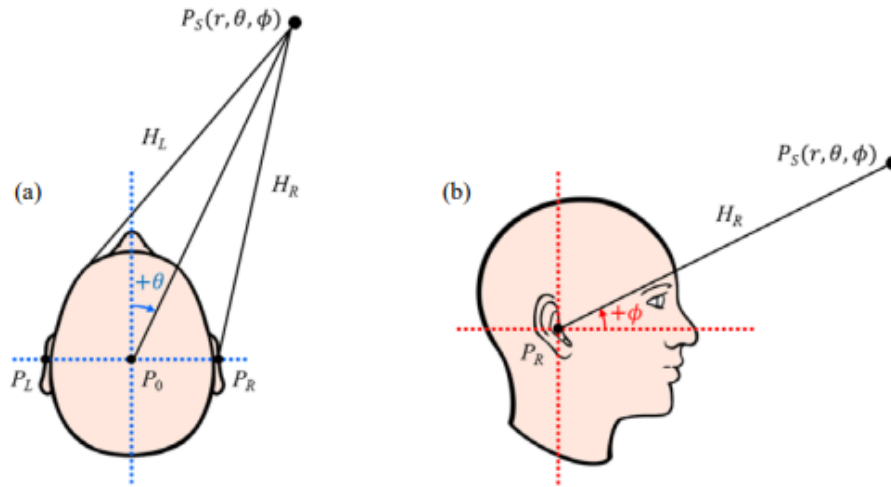


Figura 3.2: Modelo filtro lineal[27]

Donde:

1.  $X(f)$ : Transformada de Fourier del sonido de entrada.
2.  $H(f)$ : HRTF en función de la frecuencia.
3.  $Y(f)$ : Transformada de Fourier del sonido procesado.

Para cada oído, el sistema utiliza un conjunto de HRTFs específico:

1.  $H_L(f)$ : Función de transferencia para el oído izquierdo.
2.  $H_R(f)$ : Función de transferencia para el oído derecho.

### 3.2.5. Dependencia de la Posición

La HRTF no es constante; cambia con la posición espacial de la fuente sonora. En coordenadas esféricas, la HRTF es una función multidimensional:

$$HRTF(\theta, \phi, f) = H(f, \theta, \phi)$$

Donde:

1.  $\theta$ : Ángulo de azimut, mide la dirección horizontal ( $0^\circ$  para frente,  $\pm 180^\circ$  para detrás).
2.  $\phi$ : Ángulo de elevación, mide la posición vertical ( $0^\circ$  para el nivel del oído,  $+90^\circ$  para arriba,  $-90^\circ$  para abajo).

### 3.2.6. Datos Empíricos y Medición

El HRTF se mide generalmente en una cámara anecoica colocando micrófonos en los oídos de una persona o un maniquí antropométrico (dummy head). Se registra la respuesta a señales emitidas desde diferentes posiciones en el espacio. Los resultados producen un conjunto de datos que puede representarse como un mapa HRTF, que contiene respuestas de frecuencia específicas para cada ángulo  $(\theta, \phi)$  y oído.

### 3.2.7. Análisis Espectral

El HRTF incluye componentes clave que afectan cómo percibimos el sonido:

1. Interaural Time Difference (ITD): Representa la diferencia en tiempo de llegada del sonido a cada oído debido a la distancia relativa entre la fuente y los oídos. Matemáticamente, se calcula como:

$$ITD = \frac{d}{c} * \sin(\theta)$$

Donde:

- a)  $d$ : Distancia entre los oídos (aproximadamente 18 cm en humanos).
  - b)  $c$ : Velocidad del sonido en aire (343 m/s).
  - c)  $\theta$ : Ángulo de incidencia horizontal.
2. Interaural Level Difference (ILD): Es la diferencia en intensidad del sonido percibido entre los oídos debido a la atenuación causada por la cabeza. Esto depende de la frecuencia, ya que las frecuencias altas son más susceptibles al efecto de sombra acústica.

### 3.2.8. HRTF en Sistemas Digitales

En aplicaciones prácticas, los datos de HRTF se almacenan en tablas o bases de datos tridimensionales que mapean cada posición  $(\theta, \phi)$  a su correspondiente respuesta en frecuencia. Los sistemas de procesamiento digital de señales (DSP) utilizan estas bases para aplicar filtros y simular la percepción del sonido en un entorno tridimensional.

Por ejemplo, en un sistema de realidad virtual, para simular un sonido que proviene de un ángulo dado, se selecciona el filtro HRTF correspondiente y se aplica al audio en tiempo real.

# Análisis y Diseño

---

En este capítulo se detalla el proceso de análisis y escogencia de las herramientas a usar para el desarrollo del aplicativo, en conjunto con los requerimientos tanto funcionales como no funcionales y las interfaces implementadas.

## 4.1. Requisitos Funcionales y No Funcionales

### 4.1.1. Requisitos Funcionales

1. **Inicio de la Aplicación:** La aplicación debe mostrar un menú principal con tres botones: **Iniciar**, **Información** y **Salir**.
2. **Botón de Información:** Al presionar el botón de "Información", se debe abrir una ventana emergente que muestre un texto explicativo sobre la funcionalidad de la aplicación.
3. **Botón de Salir:** Al presionar el botón de "Salir", la aplicación debe cerrarse completamente.
4. **Botón de Iniciar:** Al presionar el botón de "Iniciar", la aplicación debe habilitar la cámara del dispositivo y mostrar un entorno de realidad aumentada. Debe generar un objeto 3D en forma de parlante en una posición aleatoria.
5. **Emisión de Sonido:** El objeto 3D debe emitir un sonido tridimensional que permita al usuario identificar su dirección (izquierda, derecha, arriba, abajo, etc.).
6. **Interacción con el Usuario:** El usuario debe poder apuntar la cámara hacia el objeto 3D, seleccionarlo haciendo clic en la pantalla, y el objeto debe cambiar de posición inmediatamente.
7. **Actualización de Coordenadas:** Al encontrar el objeto, las coordenadas del objeto seleccionado deben mostrarse en la parte inferior de la pantalla.
8. **Ciclo Continuo:** El proceso de generación del objeto 3D, emisión de sonido, interacción del usuario y cambio de posición debe repetirse indefinidamente hasta que se cierre la aplicación.

### 4.1.2. Requisitos No Funcionales

1. **Compatibilidad:** La aplicación debe ser compatible con dispositivos Android, con versiones mínimas de sistema operativo específicas (por ejemplo, Android 8.0 o superior).

2. **Eficiencia y Rendimiento:** La aplicación debe operar de manera fluida, asegurando una buena tasa de fotogramas por segundo (FPS) y sin retrasos perceptibles, incluso en dispositivos de gama media.
3. **Interfaz Intuitiva:** El menú principal y las ventanas de interacción deben ser intuitivos y fáciles de usar para usuarios sin conocimientos técnicos previos.
4. **Consumo de Recursos:** La aplicación debe optimizar el uso de la batería y el procesamiento del dispositivo para evitar un consumo excesivo.
5. **Velocidad de Generación del Objeto:** El objeto 3D debe generarse en menos de 0.5 segundos tras ser seleccionado, asegurando una experiencia fluida para el usuario.

#### 4.1.3. Flujo del Aplicativo

La aplicación sigue un flujo estructurado que asegura la correcta interacción entre el usuario y el entorno virtual. A continuación, se describe el flujo principal:

##### 1. Inicialización de la Aplicación:

- La aplicación solicita permisos necesarios, como el acceso a la cámara y los sensores del dispositivo.
- Unity carga la escena inicial, en donde muestra el logo de Unity, configurando el entorno y los parámetros base de la aplicación.
- Finalmente Unity completa la carga y muestra el menú principal.

##### 2. Generación del Objeto Virtual:

- Un objeto 3D (Parlante) se genera en una posición predefinida frente al usuario dentro del espacio virtual.
- El objeto empieza a emitir una melodía constantemente la cual es usada para guiar al usuario.
- Se establece un patrón de movimiento para el objeto, que define las ubicaciones a las que se moverá durante la interacción.

##### 3. Simulación de Sonido Espacial:

- Resonance Audio utiliza HRTF para procesar el sonido asociado al cubo, simulando su origen en la posición actual del objeto dentro del espacio tridimensional.
- El sonido cambia en tiempo real a medida que el objeto se mueve a nuevas ubicaciones, ajustando parámetros como el tiempo de llegada (ITD) y la diferencia de nivel entre los oídos (ILD).

##### 4. Interacción del Usuario:

- El usuario busca el objeto utilizando tanto la cámara del dispositivo como las señales auditivas generadas por Resonance Audio.
- Una vez que encuentra el objeto, el usuario interactúa con él tocándolo en la pantalla, cada interacción registrada genera un cambio en la ubicación del objeto a un punto alrededor del usuario.

#### 5. Registro de Interacciones:

- El tiempo transcurrido entre la generación de una nueva posición del objeto y la interacción del usuario se registra.
- Se calcula una nueva posición generando un número entre 1 y 360, el cual representa el nuevo ángulo al que será trasladado el objeto, nuevamente se genera un número aleatorio entre -0.5 y 0.5 definiendo la altura a la que se encontrará debido a que su posición varía en un cilindro alrededor del usuario teniendo en cuenta que este se encuentra en las coordenadas (0, 0, 0).

También sigue un flujo específico respecto a las interacciones con el usuario:

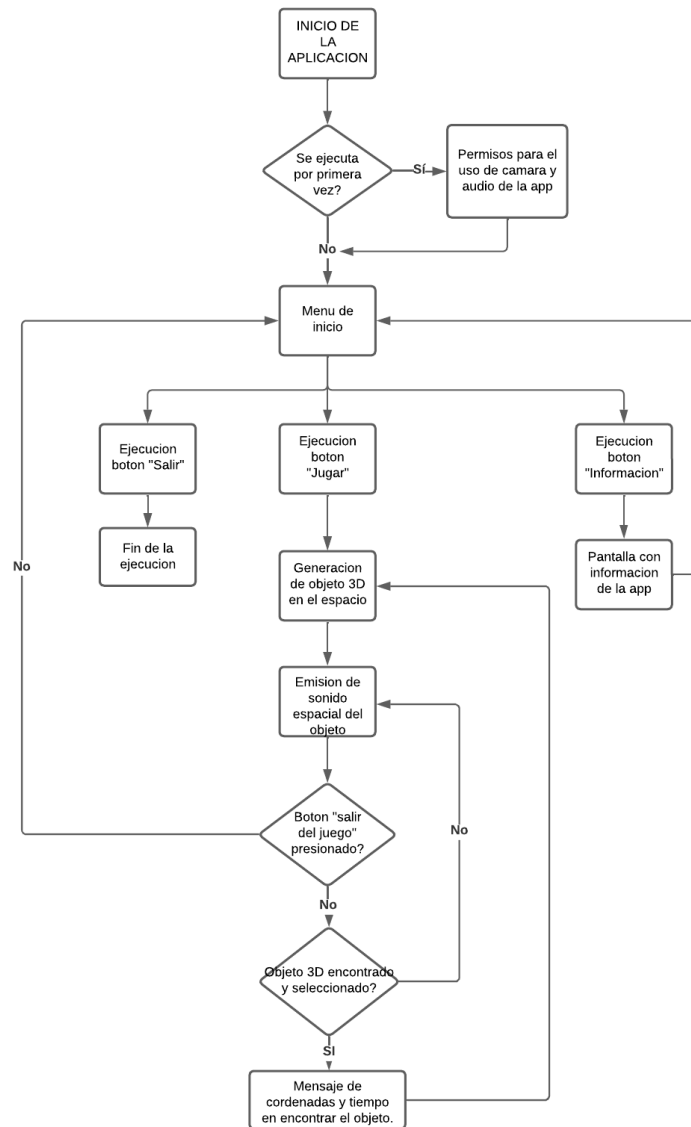


Figura 4.1: Flujograma App

## 4.2. Herramientas para el desarrollo de aplicaciones de realidad aumentada

El desarrollo de aplicaciones de realidad aumentada (RA) requiere la elección de herramientas que se adapten a las necesidades específicas del proyecto. A continuación, se presenta una comparativa entre las principales plataformas disponibles para la creación de aplicaciones de RA.

### 4.2.1. Unity

Unity es uno de los motores de desarrollo más populares y versátiles, utilizado ampliamente para la creación de aplicaciones interactivas en 2D y 3D, incluyendo realidad aumentada (RA) y realidad virtual (RV). En este proyecto, se seleccionó Unity como la herramienta principal debido a su capacidad multiplataforma, facilidad de uso y soporte para audio espacial mediante librerías como Resonance Audio.



Figura 4.2: Logo Unity

A continuación, se detallan las ventajas de Unity y se justifica su elección para este proyecto.

#### 4.2.1.1. Ventajas

- **Multiplataforma:** Unity permite desarrollar aplicaciones para una amplia variedad de plataformas, incluyendo Android, iOS, Windows, macOS, y dispositivos de RV/RA. Esta flexibilidad asegura que los desarrollos puedan extenderse a otras plataformas con mínimos ajustes.
- **Facilidad de Uso:** Unity cuenta con una interfaz visual intuitiva y herramientas de diseño avanzadas, lo que facilita el desarrollo de aplicaciones incluso para desarrolladores con experiencia limitada.
- **Compatibilidad con Librerías de Audio Espacial:** Unity admite la integración de librerías como Resonance Audio, Steam Audio y Oculus Spatializer, lo que permite implementar sonido 3D inmersivo de manera eficiente.
- **Comunidad Activa y Documentación:** Unity tiene una comunidad global activa y una amplia documentación, lo que facilita la resolución de problemas y el acceso a recursos educativos.
- **Bajo Costo:** Unity ofrece una versión gratuita con funcionalidades completas, adecuada para proyectos pequeños y medianos, lo que elimina barreras económicas para su adopción.
- **Motor Gráfico Potente:** Proporciona herramientas avanzadas para renderizado en tiempo real, sombreado e iluminación, asegurando un diseño visual atractivo.

## 4.2.1.2. Comparación

Aspecto	Unity	Alternativas (Unreal Engine, Vuforia)
Multiplataforma	Amplia compatibilidad: Android, iOS, PC, y más.	Varía; Unreal tiene mejor soporte gráfico, Vuforia requiere Unity.
Facilidad de Uso	Intuitivo para principiantes y proyectos medianos.	Unreal más complejo; Vuforia limitado a RA específica.
Compatibilidad con Audio Espacial	Compatible con Resonance Audio, Steam Audio, etc.	Similar, pero depende del motor y las librerías usadas.
Costo	Gratuito hasta cierto nivel de ingresos.	Unreal aplica regalías; Vuforia tiene costos de licencia.
Optimización para Móviles	Requiere ajustes manuales para dispositivos de gama baja.	Unreal más demandante; Vuforia más ligero pero limitado.

Cuadro 4.1: Comparación entre Unity y otras herramientas de desarrollo.

## 4.2.2. Unreal Engine

Unreal Engine es una plataforma avanzada de desarrollo de videojuegos y aplicaciones inmersivas, reconocida por su capacidad de renderizado de gráficos de alta calidad y sus herramientas de simulación en tiempo real. Aunque Unity fue la elección principal para este proyecto, Unreal Engine podría considerarse una alternativa válida gracias a sus potentes funcionalidades y su enfoque en experiencias interactivas inmersivas. A continuación, se describen en detalle sus ventajas y desventajas en el contexto del desarrollo de una aplicación de realidad aumentada (RA) con sonido espacial.



Figura 4.3: Logo Unreal Engine

#### 4.2.2.1. Ventajas

- **Calidad Gráfica Superior:** Unreal Engine es ampliamente reconocido por su motor de renderizado Lumen y su capacidad de proporcionar gráficos hiperrealistas. Esto podría mejorar significativamente la experiencia visual del usuario, especialmente para aplicaciones de RA que buscan un alto nivel de realismo.
- **Herramientas Integradas para RA y RV:**
  - Soporte nativo para ARCore (Android) y ARKit (iOS), lo que lo convierte en una opción flexible para aplicaciones multiplataforma.
  - Funciones avanzadas para gestionar iluminación, sombreado y efectos visuales en entornos de RA.
- **Audio Avanzado:** Unreal Engine incluye un motor de audio robusto que permite simular efectos sonoros avanzados, incluyendo oclusión, reverberación y propagación del sonido en tiempo real. También soporta la integración con librerías externas como Steam Audio o Resonance Audio.
- **Open Source (Parcial):** Aunque no completamente de código abierto, Unreal Engine permite acceder y modificar su código fuente, lo que brinda flexibilidad para personalizaciones avanzadas.
- **Documentación y Comunidad:** La documentación de Unreal Engine es extensa y está respaldada por una comunidad activa de desarrolladores, lo que facilita resolver problemas y aprender nuevas técnicas.
- **Desempeño en Hardware Potente:** Está optimizado para aprovechar al máximo las capacidades de dispositivos de gama alta, lo que podría mejorar la calidad de la experiencia para usuarios con hardware avanzado.

#### 4.2.2.2. Desventajas

- **Curva de Aprendizaje Pronunciada:**
  - La complejidad de Unreal Engine puede ser un desafío para desarrolladores principiantes o aquellos sin experiencia previa en su uso.
  - Su sistema de scripting basado en Blueprints es poderoso, pero puede ser más difícil de manejar en proyectos complejos comparado con el scripting en C# de Unity.
- **Requisitos de Hardware:** Unreal Engine demanda un hardware más potente tanto para el desarrollo como para la ejecución de las aplicaciones, lo que podría limitar su adopción en dispositivos móviles de gama baja.

- **Tiempo de Desarrollo:** Las aplicaciones en Unreal Engine tienden a requerir más tiempo de desarrollo debido a su enfoque en la calidad visual y la complejidad de sus herramientas.
- **Peso del Proyecto:** Los proyectos desarrollados en Unreal Engine suelen tener tamaños de archivo significativamente mayores, lo que podría ser un inconveniente en aplicaciones móviles donde el almacenamiento es limitado.

#### 4.2.2.3. Comparativa

Aspecto	Unity	Unreal Engine
Facilidad de Uso	Más fácil para principiantes y proyectos medianos.	Más complejo, ideal para proyectos avanzados.
Gráficos	Buen nivel, personalizable según la necesidad.	Gráficos hiperrealistas por defecto.
Audio Espacial	Compatible con librerías externas como Resonance Audio.	Audio integrado con simulaciones avanzadas.
Rendimiento en Gama Baja	Mejor optimizado para dispositivos móviles.	Demanda hardware más potente.
Licencia	Gratuita hasta cierto umbral de ingresos.	Gratuita con regalías sobre ingresos altos.

Cuadro 4.2: Comparación entre Unity y Unreal Engine para el desarrollo de aplicaciones de RA.

#### 4.2.3. Vuforia

Vuforia es una plataforma ampliamente utilizada para el desarrollo de aplicaciones de realidad aumentada (RA), conocida por su especialización en el reconocimiento de imágenes y objetos. Fue una de las primeras herramientas comerciales en este campo, lo que le ha permitido consolidarse como una opción popular entre desarrolladores que buscan implementar RA en dispositivos móviles y otros sistemas. A continuación, se describen sus ventajas y desventajas en el contexto del desarrollo de una aplicación de RA que incorpora sonido espacial.



Figura 4.4: Logo Vuforia

#### 4.2.3.1. Ventajas

- **Reconocimiento Avanzado de Imágenes y Objetos:**
  - Vuforia es especialmente eficaz en el reconocimiento de marcadores visuales, imágenes planas, objetos 3D y superficies. Esta funcionalidad podría ser útil si el proyecto requiere detectar y asociar objetos del mundo real con elementos virtuales.
  - Su tecnología Model Targets permite identificar objetos tridimensionales con gran precisión, lo que es ideal para entornos industriales o educativos.
- **Compatibilidad Multiplataforma:**
  - Compatible con Unity, Android, iOS, y Microsoft HoloLens, lo que permite una amplia gama de dispositivos y casos de uso.
  - Integración sencilla con Unity mediante un SDK dedicado, lo que facilita el flujo de trabajo para desarrolladores.
- **Seguimiento Robusto:** Proporciona un seguimiento confiable incluso en condiciones de iluminación desafiantes o cuando el marcador está parcialmente visible.
- **Fácil Implementación:** Ofrece herramientas intuitivas para desarrolladores, como el Vuforia Engine, que incluye plantillas preconfiguradas para proyectos básicos de RA
- **Soporte Activo y Documentación:** Cuenta con una comunidad activa de desarrolladores y documentación detallada que facilita el aprendizaje y la solución de problemas.

#### 4.2.3.2. Desventajas

- **Licencia Comercial Costosa:** Aunque Vuforia tiene una versión gratuita con funcionalidades limitadas, el acceso completo a sus capacidades avanzadas requiere una licencia comercial, lo que podría aumentar significativamente los costos del proyecto.
- **Dependencia de Marcadores:** Aunque admite RA sin marcadores, su rendimiento óptimo se logra con marcadores visuales, lo que podría no ser ideal en aplicaciones que buscan interacción sin depender de objetos físicos específicos.
- **Rendimiento en Dispositivos de Gama Baja:** Aunque es robusto, su procesamiento intensivo de imágenes puede afectar el rendimiento en dispositivos móviles de gama baja, especialmente si se combina con otras tecnologías demandantes como sonido espacial.
- **Soporte Activo y Documentación:** A diferencia de herramientas como Unity o Unreal Engine, Vuforia no incluye soporte nativo para audio espacial, lo que implica depender completamente de librerías externas como Resonance Audio.

## 4.2.3.3. Comparativa

Aspecto	Vuforia	Unity
Facilidad de Uso	Mayor complejidad, requiere Unity al ser integrado como plugin.	Intuitivo y listo para usar con ARCore y ARKit.
Gráficos	El renderizado depende de Unity, tiene control grafico limitado.	Potente motor grafico con alta personalización en el ambito visual.
Audio Espacial	Soporte basico del cual se encarga Unity.	Integración avanzada con distintas librerias tales como Resonance Audio.
Rendimiento en Gama Baja	Eficiente con tareas de Realidad Aumentada basica, limitado por la dependencia a Unity	Gran capacidad de optimización, puede resultar exigente a mayor calidad grafica
Licencia	Versión gratuita limitada; funcionalidades avanzadas requieren licencia paga.	Gratuito hasta cierto umbral de ingresos; costos adicionales solo para proyectos grandes.

Cuadro 4.3: Comparación entre Vuforia y Unity para el desarrollo de aplicaciones de RA.

## 4.2.3.4. Justificación de la Elección de Unity

Los principales factores que influyeron en la decisión de usar Unity como motor principal para el desarrollo del aplicativo fueron inicialmente la optimización que brinda en los dispositivos móviles, teniendo en cuenta el rendimiento de la aplicación como una herramienta para aumentar la inmersión, puesto que influye directamente en una experiencia de usuario fluida.

De igual manera, se evaluó el costo de la licencia puesto que se busca hacer uso de herramientas fácilmente accesibles para todo público, lo que implica realizar la menor cantidad de gastos posible y finalmente la rápida curva de aprendizaje que ofrece Unity, la cual puede ser aprovechada para desarrollar aplicativos de calidad en un menor tiempo haciendo uso de todas sus capacidades.

## 4.3. Librerías para el desarrollo de aplicaciones con sonido 3D

## 4.3.1. Resonance Audio

Resonance Audio, desarrollado por Google, es una librería de audio espacial diseñada para proporcionar experiencias inmersivas en entornos tridimensionales. Es especialmente adecuada para dispositivos móviles y aplicaciones ligeras gracias a su eficiencia en el uso de recursos. En este proyecto, se eligió Resonance Audio como la librería principal para implementar sonido espacial debido a su flexibilidad, simplicidad y compatibilidad con Unity.



Figura 4.5: Logo Resonance Audio

A continuación, se detallan las ventajas de Resonance Audio y se justifica su elección para este proyecto.

#### 4.3.1.1. Ventajas de Resonance Audio

- **Optimización para Dispositivos Móviles:** Resonance Audio está específicamente diseñada para garantizar un rendimiento eficiente en dispositivos móviles, lo cual es fundamental para aplicaciones de realidad aumentada (RA) en Android.
- **Simplicidad de Uso e Integración:** Su integración con Unity es directa y permite a los desarrolladores implementar sonido espacial sin necesidad de configuraciones complejas. Las funciones básicas de sonido 3D, como la ubicación direccional y la percepción de distancia, se pueden lograr con pocos pasos.
- **Multiplataforma:** Además de Android, Resonance Audio es compatible con iOS, Windows, macOS y sistemas de realidad virtual como Oculus y HTC Vive, lo que garantiza flexibilidad para futuros desarrollos.
- **Código Abierto:** Al ser de código abierto, Resonance Audio permite personalización y adaptaciones específicas a las necesidades del proyecto sin restricciones comerciales.
- **Ligereza:** La librería es menos demandante en términos de recursos computacionales en comparación con alternativas como Steam Audio o Oculus Spatializer, lo que la hace ideal para dispositivos de gama media y baja.

- **Documentación y Comunidad Activa:** Google ofrece documentación detallada, y la comunidad de desarrolladores proporciona soporte adicional, facilitando la resolución de problemas.

#### 4.3.1.2. Justificación de la Elección de Resonance Audio

Resonance Audio fue seleccionada como la librería de sonido espacial para este proyecto debido principalmente a su simplicidad de uso e integración la cual conlleva a tener una gran compatibilidad con Unity, lo que permite enfocar los esfuerzos en la creación de contenido y diseño, en lugar de en configuraciones técnicas complejas.

Adicionalmente al escoger Android como el sistema operativo para el óptimo funcionamiento de la aplicación se requirió una librería con gran compatibilidad Multiplataforma, que destaque en el área de dispositivos móviles y finalmente resaltando la flexibilidad y costos ya que al Resonance Audio ser una librería de código abierto ofrece todas sus funcionalidades sin limitaciones.

#### 4.3.1.3. Comparación en el Contexto del Proyecto

Aspecto	Resonance Audio	Alternativas (Steam Audio, Oculus Spatializer)
Facilidad de Uso	Alta, integración rápida en Unity.	Varía; Steam Audio y Oculus requieren más configuración.
Optimización para Móviles	Excelente; diseñada para dispositivos Android e iOS.	Steam Audio demanda más recursos ideal para PC; Oculus optimizado para VR.
Funcionalidades Avanzadas	Limitada; se enfoca en sonido espacial básico.	Alternativas ofrecen oclusión, reverberación y propagación detallada.
Costo	Gratuito y de código abierto.	Gratuito, pero con restricciones en algunos casos (Oculus).
Compatibilidad	Multiplataforma: Android, iOS, PC, y sistemas VR.	Multiplataforma, pero a menudo más específica, como Steam para PC u Oculus para su ecosistema.

Cuadro 4.4: Comparación entre Resonance Audio y alternativas.

## 4.4. Interfaz de Usuario (UI)

### 4.4.1. Descripción

Referente al diseño del aplicativo, la interfaz del aplicativo móvil consta de tres pantallas principales: la pantalla inicial, la pantalla del entorno de realidad aumentada y la pantalla de información.

Se optó por un diseño simple con el objetivo de facilitar su uso por parte de los usuarios, haciendo que la aplicación sea intuitiva y fácil de manejar. Adicionalmente, en la sección del entorno de realidad aumentada se incorporaron diversos sonidos e imágenes para fomentar el reconocimiento auditivo y visual, mejorando así la experiencia interactiva.

#### 4.4.2. Aplicativo

- **Pantalla Principal** La pantalla principal consta de tres botones con diferentes funcionalidades. El primer botón, “INICIAR” da acceso a la pantalla del juego, donde se activa el uso de la cámara del dispositivo móvil y del audio interno. El segundo botón, “INFORMACIÓN”, abre una ventana que proporciona detalles sobre la aplicación y explica el propósito de su creación. Finalmente, el botón “SALIR” cierra la aplicación de forma inmediata.



Figura 4.6: Pantalla Principal

- **Pantalla de Información** Esta pantalla contiene información básica del aplicativo.

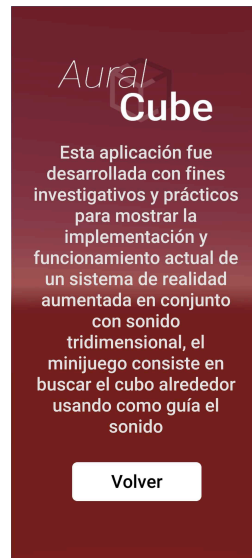


Figura 4.7: Pantalla de Información

- **Pantalla de Juego** En la pantalla de juego, el fondo visual que percibe el usuario depende exclusivamente de la cámara del dispositivo y de la dirección en la que esta sea apuntada. Además, se incorporaron dos imágenes de altavoces diferentes como objetos tridimensionales dentro de la aplicación.



Figura 4.8: Parlantes



Figura 4.9: Pantalla de Juego

#### 4.4.3. Funcionamiento

Al ejecutarse, la aplicación muestra inicialmente el logo de Unity, plataforma en la que fue desarrollada, seguido de la pantalla inicial. En esta pantalla, el usuario puede elegir interactuar con alguno de los tres botones disponibles. Si se seleccionan los botones de “SALIR” o “INFORMACIÓN”, la aplicación continuará el flujo previamente explicado. Por otro lado, al seleccionar la opción de la pantalla del entorno de realidad aumentada, la aplicación comenzará a utilizar la cámara del dispositivo para localizar un objeto 3D que se posiciona de manera aleatoria alrededor del usuario. Cuando el usuario encuentre dicho objeto y lo seleccione, este desaparecerá y se generará automáticamente otro objeto en una nueva ubicación aleatoria, permitiendo que la búsqueda continúe.

# IMPLEMENTACIÓN

---

En este capítulo se detalla el proceso de implementación de la aplicación de realidad aumentada desarrollada para dispositivos Android. Se describen los flujos del aplicativo, el diseño arquitectónico del sistema, los fragmentos de código clave y las herramientas utilizadas, con un énfasis especial en la integración de sonido espacial mediante la librería Resonance Audio de Google.

## 5.1. Desarrollo

En esta sección se describe el proceso de implementación de la aplicación, detallando las funciones principales que integran los componentes tanto visuales como auditivos. El proyecto se llevó a cabo empleando conceptos avanzados como la **Función de Transferencia Relacionada con la Cabeza (HRTF)** para enriquecer la experiencia auditiva y lograr una mayor percepción del entorno virtual por parte del usuario.

### 5.1.1. Funciones Principales

La implementación del sistema se organiza en funciones clave que garantizan la correcta operación de la aplicación. Estas funciones incluyen:

- **Carga de Escenas:** Una función simple se encarga de gestionar la debida carga de las escenas correspondientes y el flujo de la aplicación, asimismo como terminarla.

```
public class SceneLoader : MonoBehaviour
{
    public void Sceneloader(int SceneIndex)
    {
        SceneManager.LoadScene(SceneIndex);
    }

    public void QuitApp()
    {
        Application.Quit();
    }
}
```

Figura 5.1: Función de carga de escenas

- **Control del Movimiento del Objeto para pruebas:** Para mantener un ambiente de pruebas controlado y justo entre los usuarios a los cuales se les realizaría las mediciones correspondientes para la sección posterior de pruebas, se pre-definieron 5 posiciones iniciales a las cuales se movería el objeto 3D (Parlante) y de las cuales se tomaron los datos de prueba. Las posiciones en cuestión son:

- El objeto se posiciona en el lado izquierdo del usuario.
  
- El objeto se posiciona detrás del usuario.
  
- El objeto se posiciona en el lado derecho del usuario.
  
- El objeto se posiciona en la parte izquierda superior del usuario.
  
- El objeto se posiciona en la parte derecha inferior del usuario.

Durante cada interacción la aplicación registra el lapso que tardó el usuario entre un click y otro al objeto, por lo cual se puede conocer el tiempo en milisegundos y usarlo como métrica de evaluación.

```
void OnMouseDown()
{
    float radius = 1.5f;
    float angle = 0.0f;
    float y = 0.0f;
    float x = 0.0f;
    float z = 0.0f;
    if(times == 1)
    {
        x = -1.5f;
        y = 0.0f;
        z = 0.0f;
    }
    else if(times == 2)
    {
        x = 1.5f;
        y = 0.0f;
        z = 0.0f;
    }
    else if(times == 3)
    {
        x = 0.0f;
        y = 0.0f;
        z = -1.5f;
    }
    else if(times == 4)
    {
        x = 1.5f;
        y = 0.5f;
        z = 0.0f;
    }
    else if(times == 5)
    {
        x = 0.0f;
        y = -0.5f;
        z = -1.5f;
    }
    else {
        angle = Mathf.Deg2Rad * UnityEngine.Random.Range(0,365);
        y = UnityEngine.Random.Range(-0.5f, 0.5f);
        x = radius * Mathf.Cos(angle);
        z = radius * Mathf.Sin(angle);
    }
    Cube.transform.position = new Vector3(x, y, z);
    testText.text = $"Time took: {(DateTime.UtcNow - interval).TotalMilliseconds.ToString()}";
    interval = DateTime.UtcNow;
    times++;
}
```

Figura 5.2: Función de movimientos predefinidos

- **Aleatorización del Movimiento del Objeto:** Para los usuarios comunes del aplicativo se tiene definido que cada interacción con el objeto lo transporta a una posición aleatoria alrededor del usuario, tal que primero se ubica en un punto manteniendo la altura del objeto constante y luego se calcula esta aparte usando la aleatorización nativa de Unity, de forma que la posición varía en un cilindro alrededor del usuario en cuestión, este comportamiento también es tomado en el ambiente de pruebas una vez se realizan los 5 movimientos predefinidos en aquel caso.

```
public class MoveObject : MonoBehaviour
{
    // Se obtiene el objeto de texto en el cual se muestra cierta
    // Informacion de los calculos realizados
    public Text testText;
    // Se obtiene el cubo al cual se le realizara el movimiento
    public GameObject Cube;

    // El evento OnMouseDown es usado para detectar cuando se
    // realice interaccion con el objeto
    void OnMouseDown()
    {
        // Se define el radio en el que estará el objeto
        float radius = 1.5f;
        // Se calcula el angulo al que se desea mover el objeto
        float angle = Mathf.Deg2Rad * Random.Range(0,365);
        // Se calcula la posicion en el eje X
        float x = radius * Mathf.Cos(angle);
        // Se toma un valor aleatorio en Y para variar su altura
        float y = Random.Range(-0.5f, 0.5f);
        // Se calcula la posicion en el eje Z
        float z = radius * Mathf.Sin(angle);
        // Se asigna la nueva posicion al objeto
        Cube.transform.position = new Vector3(x, y, z);
        // Actualiza el texto con los valores calculador
        testText.text = $"angle: {angle} cos: {x} sin: {z} y: {y}";
    }
}
```

Figura 5.3: Función de movimientos aleatorios

- **Cálculo de posición del Objeto:** Al momento de aleatorizar la posición del objeto se requiere obtener un punto alrededor del usuario para mover el objeto una vez se realice la interacción, para esto se hace uso de coordenadas polares, por medio de la aleatorización nativa de Unity se obtiene un número entre 1 y 360 el cual representa un ángulo alrededor del usuario, debido a que Unity se maneja como un plano tridimensional en el cual hay coordenadas X, Y y Z, se hace uso de la ecuación para conversión de coordenadas polares a rectangulares para calcular las coordenadas tanto en X como en Z puesto que la altura del objeto varía 0.5 metros verticalmente partiendo desde la altura inicial de la cabeza del usuario, adicionalmente se debe tener en cuenta que el radio en el que aparece del objeto tomando como origen el punto en el que se inició el aplicativo, es de 1.5 metros.

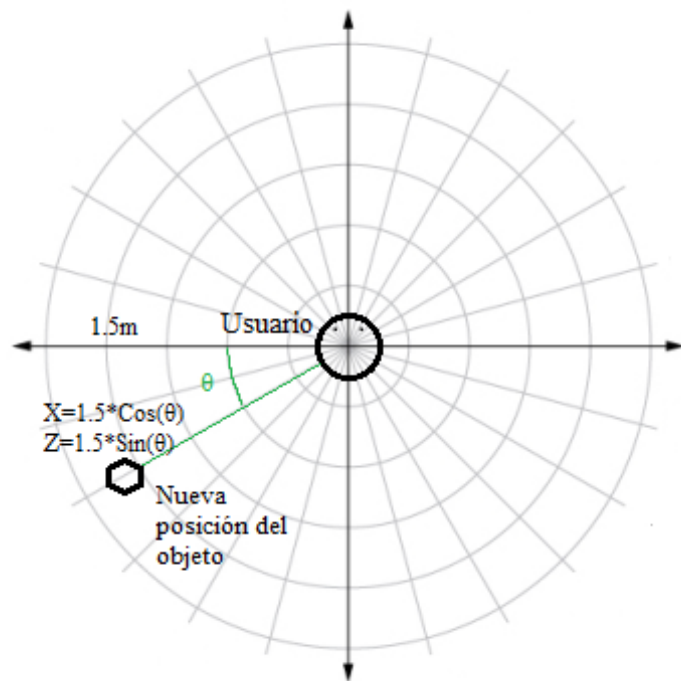


Figura 5.4: Calculo de nueva posición del objeto

- **Generación de Sonido Espacial:** Resonance Audio incorpora HRTF para emular cómo los sonidos son modificados por la interacción con la cabeza, las orejas y el torso humano. Esto permite simular de manera realista la ubicación y dirección del sonido, proporcionando una experiencia auditiva tridimensional precisa. En Unity se le asigna el uso de la librería de Resonance Audio, de esta forma se aplican los filtros derivados del HRTF tales como el Interaural Time Difference (ITD) para ajustar el tiempo de llegada y el Interaural Level Difference (ILD) para la diferencia niveles en ambos oídos, ajustando algunos parámetros manualmente tales como:
  - **Spatial Blend:** define la espacialización del audio, el cual se asigna el valor de 1 puesto que se requiere completamente.
  - **Min Distance:** hace referencia a la distancia mínima para el efecto de espacialización.
  - **Max Distance:** hace referencia a la distancia máxima para el efecto de espacialización.
  - **Room Effects Gain DB:** define la ganancia en decibeles de los efectos del ambiente aplicados al audio, debido a que se maneja un entorno virtual donde únicamente se encuentran el usuario y el objeto (Parlante) no se requiere un incremento.
  - **Nivel de Doppler:** relacionado con el efecto Doppler, cambia la frecuencia del sonido dependiendo del movimiento del objeto, debido a que en el aplicativo constantemente se

está cambiando la posición del objeto 3D, se debe tener en cuenta.

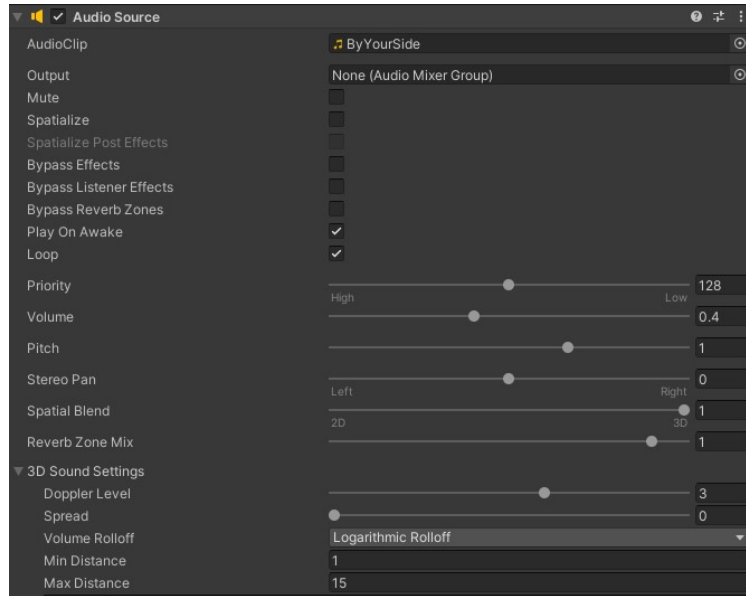


Figura 5.5: Parámetros de la fuente de audio de Unity

Una vez se es vinculado el Resonance Audio al objeto, Unity le provee algunos parámetros al HRTF para su correcto funcionamiento, tales como la ubicación del objeto, de la cual se puede extraer el ángulo azimutal (posición horizontal), el ángulo de elevación (posición vertical) y la distancia entre la fuente y el oyente, todos estos esenciales para el cálculo del efecto a aplicar.

Finalmente cuando se define la fuente del sonido usando Resonance Audio, se muestran parámetros adicionales tales como:

- **Directividad del Oyente (Listener Directivity):** controla cómo el oyente recibe audio del objeto dependiendo de su orientación, este se compone de 2 parámetros adicionales que son Alpha y Sharpness, Alpha define el cómo se escucha el sonido dependiendo de la posición de donde provenga, su valor por defecto es 0, el cual representa la omnidireccionalidad escuchándose de igual forma independiente de donde provenga la onda, mientras que en el valor 1 toma una forma cardioide, en la que si el origen del sonido se encuentra detrás o a un costado del usuario, este puede incluso no percibirlo. En cuanto al parámetro Sharpness define la transición del audio entre zonas donde se escucha más fuerte y más débil, en su valor por defecto, dicha transición es gradual, mientras que si toma valores altos, va a tener una caída más notoria entre los niveles de volumen.
- **Directividad de la fuente (Source Directivity):** tiene una funcionalidad muy similar a la directividad del oyente, ambos poseen los mismos parámetros Alpha y Sharpness,

su variación está en que este controla como la fuente emite el audio, independientemente la orientación del usuario.

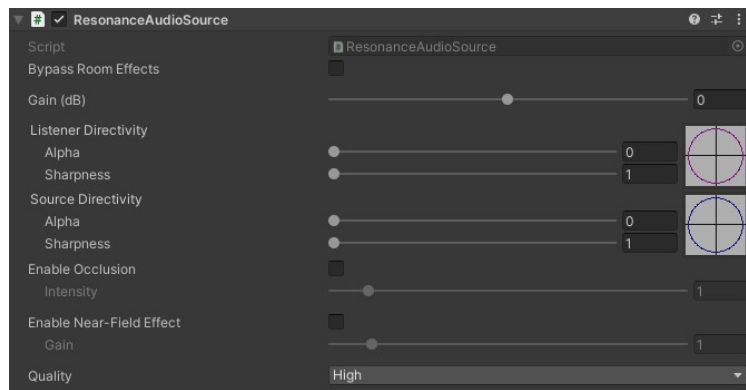


Figura 5.6: Parámetros de la fuente de audio de Resonance Audio

- **Registro del Tiempo de Interacción:** Adicional a la medida implementada en la sección previa en la cual el aplicativo toma registro del tiempo entre interacciones para el ambiente de pruebas, se contó con un sujeto adicional que realizaba monitoreo a los tiempos que se mostraban en el aplicativo para evitar distracciones en el usuario. Los datos recolectados se utilizan para evaluar la eficiencia del sistema y la capacidad del usuario para localizar el objeto.

# RESULTADOS

---

## 6.1. Pruebas

En este capítulo se describen las pruebas realizadas para evaluar la funcionalidad y efectividad de la aplicación de realidad aumentada, enfocada en la localización espacial de objetos 3D mediante sonido espacial. Se detallan los métodos utilizados, el diseño experimental, los participantes, y los resultados obtenidos, junto con su análisis.

### 6.1.1. Objetivo de las Pruebas

El objetivo principal de las pruebas fue determinar la capacidad de los usuarios para localizar un objeto virtual en el espacio tridimensional utilizando únicamente señales auditivas generadas a través de la tecnología de sonido espacial. Se buscó medir la efectividad del sistema mediante el tiempo que los participantes demoraron en identificar la dirección del objeto 3D.

### 6.1.2. Diseño Experimental

Las pruebas se llevaron a cabo utilizando dispositivos con sistema operativo Android, equipados con auriculares de uso cotidiano. Con el fin de garantizar la consistencia en la recopilación de datos, todas las pruebas fueron realizadas en un único dispositivo móvil, asegurando que las variaciones observadas dependieran únicamente de las capacidades de los usuarios y no de las diferencias entre dispositivos.

Para estandarizar las condiciones experimentales y eliminar la variable de aleatoriedad en la ubicación del objeto 3D, la aplicación fue modificada específicamente para el desarrollo de estas pruebas. Se implementó una función que posicionó el objeto virtual en cuatro ubicaciones predefinidas en el espacio tridimensional: izquierda, derecha, adelante y atrás. Esta modificación permitió evaluar la capacidad de los usuarios para localizar el sonido espacial bajo condiciones controladas y replicables.

Las pruebas consistieron en la colocación del objeto virtual en cuatro posiciones distintas en relación con la posición inicial del participante:

- **Izquierda**
- **Derecha**
- **Adelante**

- **Atrás**

Los participantes utilizaron audífonos compatibles con sonido espacial y un dispositivo móvil para interactuar con la aplicación. El objeto virtual solo era visible a través de la cámara del dispositivo, y su posición cambiaba de manera no aleatoria sin que el participante tuviera conocimiento previo de su ubicación.

- **Variables evaluadas:**

- **Tiempo de localización:** Tiempo transcurrido desde el inicio de la prueba hasta que el participante identificó correctamente la ubicación del objeto.
- **Precisión:** Si el participante señaló correctamente la dirección del objeto en su primer intento.

- **Entorno y Equipos:**

- Espacio controlado sin ruido externo.
- Auriculares estandarizados con soporte para sonido espacial.
- Dispositivo móvil Android.

### 6.1.3. Participantes

Se seleccionó un grupo de 10 participantes (9 hombres y 1 mujer) del público general con edades comprendidas entre 20 y 50 años, bajo un criterio en el cual se tuvieran conocimientos previos sobre los conceptos de realidad aumentada, pero no sobre sonido 3D, ni haber tenido contacto con aplicaciones móviles que lo implementaran. Ninguno de los participantes presentaba problemas auditivos declarados. Antes de iniciar las pruebas, se les proporcionó una breve introducción al uso de la aplicación y una demostración de prueba para familiarizarse con el sistema.

### 6.1.4. Procedimiento

- **Configuración inicial:** Para garantizar la uniformidad en las pruebas y reducir la influencia de variables aleatorias, se ajustó la función responsable de calcular la nueva posición del objeto virtual. Durante los primeros cinco movimientos del objeto, se estableció un patrón fijo que fue idéntico para todos los usuarios participantes. Este patrón de movimiento siguió la siguiente secuencia predefinida:
  - El usuario realiza un clic sobre el objeto para iniciar la interacción.
  - El objeto se posiciona en el lado izquierdo del usuario.
  - El objeto se posiciona en la parte trasera del usuario.
  - El objeto se posiciona en el lado derecho del usuario.
  - El objeto se posiciona en la parte superior izquierda del usuario.

- El objeto se posiciona en la parte inferior derecha del usuario.

Durante estas interacciones, la aplicación registró en tiempo real el intervalo transcurrido entre el momento en que el usuario hacía clic en el objeto y encontraba su siguiente posición. Este registro, mostrado en la parte inferior de la pantalla, permitió recopilar datos precisos sobre el tiempo de interacción de los usuarios en milisegundos, proporcionando una métrica objetiva para analizar su desempeño en la tarea de localizar y seleccionar el objeto virtual.

- **Ambiente:** Habitación sin ruido externo.
- **Inicio de la prueba:** El objeto virtual se coloca en la primera posición sin que los usuarios sean conscientes del patrón.
- **Localización:** Los participantes usan las señales auditivas para identificar la dirección del objeto y posicionan la cámara del dispositivo móvil hasta que el sea visible en pantalla.
- **Registro de datos:** Se registró el tiempo que el participante demoraba en encontrar el objeto.

Cada participante completó un total de 15 pruebas (tres para cada posición predefinida del objeto). Los resultados se promediaron para obtener un análisis más robusto.

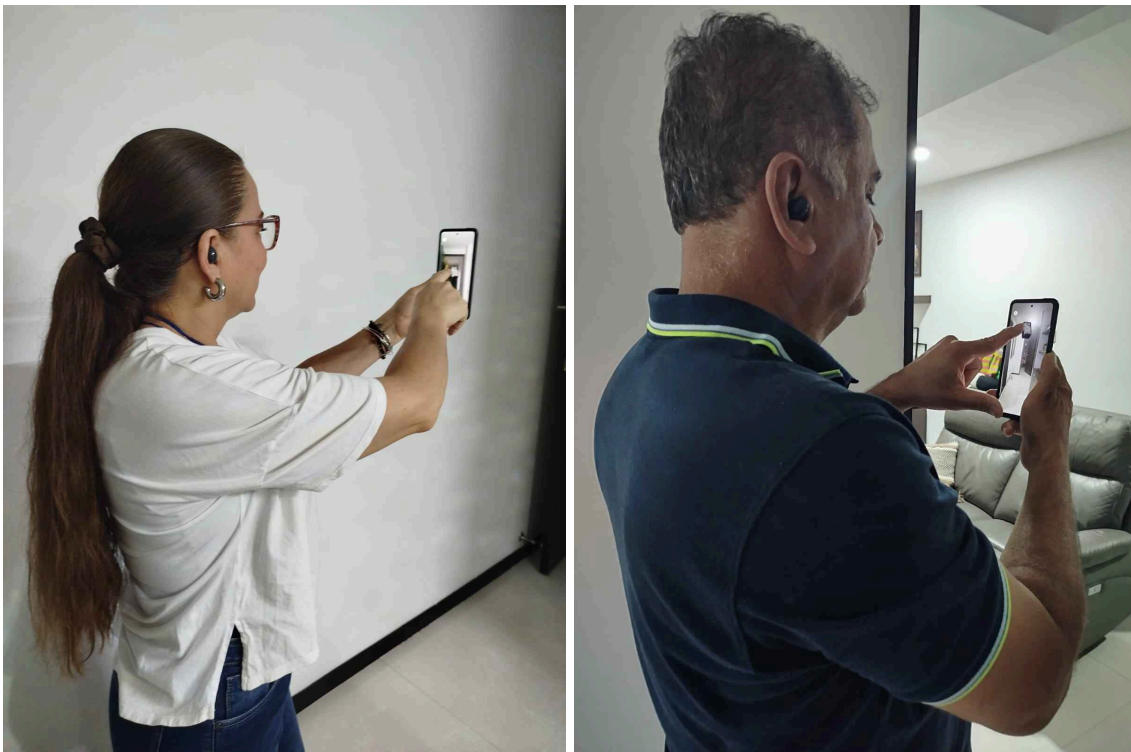


Figura 6.1: Pruebas de usuario

### 6.1.5. Resultados Pruebas de Aceptación

A continuación, se presenta una tabla de resultados promedios de los tiempos de localización (en segundos) por persona y posición:

Usuario	Edad	Tiempo en alcanzar los objetivos					Tiempo promedio entre objetivos	Menor tiempo	Mayor tiempo
		Objetivo 1 (Izquierda)	Objetivo 2 (Atras)	Objetivo 3 (Derecha)	Objetivo 4 (Izquierda.Ariba)	Objetivo 5 (Derecha Abajo)			
Usuario de Prueba 1	24 Años	3,6 Segundos	5,3 Segundos	3,5 Segundos	3,7 Segundos	4,1 Segundos	4,0 Segundos	3,5 Segundos	5,3 Segundos
Usuario de Prueba 2	22 Años	2,7 Segundos	4,2 Segundos	3,9 Segundos	4,2 Segundos	3,2 Segundos	3,6 Segundos	2,7 Segundos	4,2 Segundos
Usuario de Prueba 3	56 Años	4,1 Segundos	6,5 Segundos	3,8 Segundos	5,5 Segundos	3,5 Segundos	4,7 Segundos	3,5 Segundos	6,5 Segundos
Usuario de Prueba 4	24 Años	2,6 Segundos	4,6 Segundos	3,2 Segundos	6,1 Segundos	2,7 Segundos	3,8 Segundos	2,6 Segundos	6,1 Segundos
Usuario de Prueba 5	23 Años	2,9 Segundos	4,0 Segundos	3,1 Segundos	3,1 Segundos	2,8 Segundos	3,2 Segundos	2,8 Segundos	4,0 Segundos
Usuario de Prueba 6	24 Años	3,2 Segundos	6,3 Segundos	2,5 Segundos	3,4 Segundos	1,9 Segundos	3,5 Segundos	1,9 Segundos	6,3 Segundos
Usuario de Prueba 7	25 Años	2,8 Segundos	7,2 Segundos	2,3 Segundos	8,2 Segundos	3,1 Segundos	4,7 Segundos	2,3 Segundos	8,2 Segundos
Usuario de Prueba 8	25 Años	3,2 Segundos	4,6 Segundos	3,5 Segundos	4,4 Segundos	3,7 Segundos	3,9 Segundos	3,2 Segundos	4,6 Segundos
Usuario de Prueba 9	25 Años	2,6 Segundos	3,9 Segundos	2,5 Segundos	6,1 Segundos	3,4 Segundos	3,7 Segundos	2,5 Segundos	6,1 Segundos
Usuario de Prueba 10	52 Años	3,9 Segundos	5,8 Segundos	3,6 Segundos	5,7 Segundos	4,3 Segundos	4,7 Segundos	3,6 Segundos	5,8 Segundos
Promedio general:							4,0 Segundos		

Figura 6.2: Tabla de Tiempos

### 6.1.6. Análisis de pruebas de aceptación

- Los participantes localizaron el objeto más rápido cuando estaba ubicado por los lados izquierdo y derecho.
- La dirección más desafiante fue la posición trasera, lo que podría deberse a las limitaciones en la percepción del sonido espacial en el plano posterior.
- La precisión general fue alta en todas las posiciones, lo que sugiere que el sistema de sonido espacial implementado es efectivo para transmitir la dirección del objeto virtual.

Además, se observó que la familiarización con el sistema mejora el desempeño de los usuarios, lo que sugiere que el tiempo de localización podría reducirse aún más con un entrenamiento más prolongado.

### 6.1.7. Resultados Pruebas de Precisión

Para evaluar la precisión del sistema de audio tridimensional por medio del aplicativo, se tomó en cuenta la capacidad de los usuarios para identificar correctamente la dirección de la fuente del sonido en un primer intento, por lo cual se registró si el primer movimiento de los usuarios coincide con la dirección en la que se encuentra el objeto, adicionalmente cuando se ubica en la parte trasera del usuario, se evaluó si se detuvo la rotación buscando el objeto o si lo ubicó en primera instancia, de esta forma se obtuvieron los siguientes resultados.

Usuario	El giro del usuario coincide con la dirección del objeto?				
	Objetivo 1 (Izquierda)	Objetivo 2 (Atras)	Objetivo 3 (Derecha)	Objetivo 4 (Izquierda-Arriba)	Objetivo 5 (Derecha Abajo)
Usuario de Prueba 1	Correcto	Correcto	Correcto	Correcto	Incorrecto
Usuario de Prueba 2	Correcto	Correcto	Correcto	Correcto	Correcto
Usuario de Prueba 3	Correcto	Incorrecto	Correcto	Correcto	Incorrecto
Usuario de Prueba 4	Correcto	Incorrecto	Correcto	Correcto	Correcto
Usuario de Prueba 5	Correcto	Incorrecto	Correcto	Correcto	Correcto
Usuario de Prueba 6	Correcto	Correcto	Correcto	Correcto	Correcto
Usuario de Prueba 7	Correcto	Incorrecto	Correcto	Correcto	Correcto
Usuario de Prueba 8	Correcto	Correcto	Correcto	Correcto	Correcto
Usuario de Prueba 9	Correcto	Correcto	Correcto	Correcto	Correcto
Usuario de Prueba 10	Correcto	Incorrecto	Correcto	Incorrecto	Correcto

Figura 6.3: Tabla de pruebas de precisión

Con los resultados individuales obtenidos de los usuarios de prueba, se calculó el porcentaje de precisión que tuvieron ubicando al objeto en una primera instancia para cada objetivo, asimismo se realizó un promedio de los tiempos individualmente, de esta forma se pueden reflejar dichos resultados en la siguiente tabla.

Objetivos	% de precisión	Tiempo promedio
Objetivo 1 (Izquierda)	100%	3,2 Segundos
Objetivo 2 (Atras)	50%	5,2 Segundos
Objetivo 3 (Derecha)	100%	3,2 Segundos
Objetivo 4 (Izquierda-Arriba)	90%	5,0 Segundos
Objetivo 5 (Derecha Abajo)	80%	3,3 Segundos

Figura 6.4: Tabla de precisión

### 6.1.8. Análisis de pruebas de precisión

Por medio de los resultados obtenidos se puede intuir que las posiciones laterales tienden a ser más fáciles de localizar en primera instancia, al contrario que la posición trasera que resultó más desafiante tanto para identificar como para llegar a tal punto, dando a entender que la precisión de Resonance Audio tiene su fuerte en puntos laterales e incluso variando en altura debido a que aunque el porcentaje disminuyó respecto a cuando se mantiene una altura neutra, se sostuvo con la gran mayoría de usuarios acertando.

### 6.1.9. Pruebas de Rendimiento

El lapso en el cual se ejecutaron las pruebas con los usuarios fue de 28 minutos, a lo largo de esta actividad se usó constantemente el dispositivo móvil de pruebas (Samsung Galaxy S20 FE) el cual al momento de inicio contaba con un porcentaje de batería del 73%, a lo largo de todo el proceso únicamente se usó el aplicativo y audífonos conectados por bluetooth lo cual influye en el consumo energético, adicionalmente la cámara estuvo en uso continuo debido a que es necesario para el funcionamiento. Al finalizar el proceso de pruebas el dispositivo no tuvo un incremento notorio al tacto en la temperatura, el porcentaje final de batería resultó ser de 62% y por medio de la monitorización de recursos nativa de Samsung se obtuvieron los siguientes resultados



Figura 6.5: Consumo de recursos del dispositivo

Se puede detallar que el consumo del aplicativo respecto al 11% de batería usada fue del 6.6%, adicionalmente durante la ejecución no se presentaron caídas en la tasa de fotogramas del dispositivo ni ralentización en la aplicación, el sonido fue constante no se registraron interrupciones ni problemas notorios en la reproducción lo cual es crítico para el funcionamiento correcto del aplicativo.

### 6.1.10. Análisis de pruebas de rendimiento

- El consumo energético fue del 6.6% sobre el 11% de batería usada en el lapso de 28 minutos, lo que quiere decir que el aplicativo finalmente consumió el 0.7% de todo el dispositivo, por lo cual es moderadamente eficiente en términos de consumo energético.

- El uso de la CPU fue de 40 minutos lo cual está por encima de los 28 minutos de su ejecución, lo que indica que el aplicativo como tal requiere recursos adicionales debido principalmente al uso constante de la cámara y al procesamiento dinámico del audio dependiendo de la ubicación del objeto en el entorno virtual.
- El aplicativo tuvo un rendimiento general estable, por lo cual la experiencia de usuario no se vio afectada, otorgando así fluidez constante reflejando optimización.

#### 6.1.11. Discusión

Los resultados obtenidos confirman que el sistema es eficaz para ayudar a los usuarios a localizar objetos en un entorno tridimensional mediante señales auditivas. Sin embargo, la mayor dificultad en la posición trasera resalta un posible desafío en la percepción espacial del sonido en esa dirección, lo cual podría abordarse mediante ajustes en el procesamiento del sonido o la implementación de guías visuales complementarias.

#### 6.1.12. Conclusiones de las pruebas

Las pruebas realizadas validan la funcionalidad y precisión del sistema para la localización auditiva de objetos 3D en una aplicación de realidad aumentada. La alta precisión en las direcciones izquierda, derecha y adelante demuestra que el sonido espacial es una herramienta poderosa para aplicaciones de RA inmersivas. Sin embargo, futuras iteraciones del sistema podrían optimizar la percepción en direcciones traseras para mejorar aún más la experiencia del usuario.

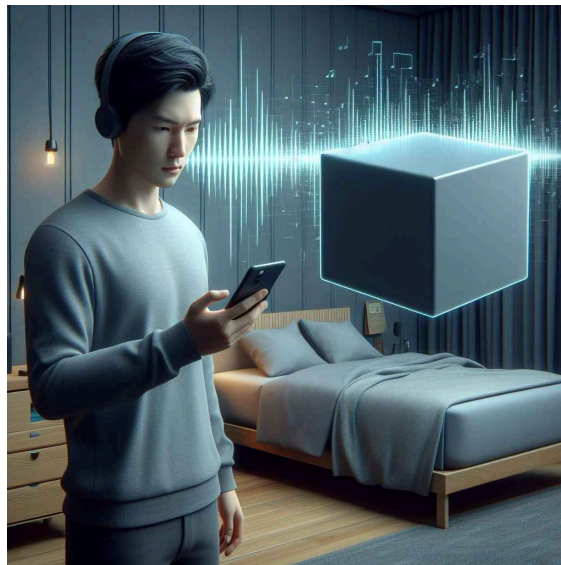


Figura 6.6: Icono Aplicación

# CONCLUSIONES

---

El proyecto consistió en el desarrollo de un aplicativo móvil que integrase los componentes visuales y auditivos por medio de la realidad aumentada con el sonido 3D, haciendo uso de las librerías a disposición de todos, mostrando las capacidades que pueden ofrecer, guiando al usuario a un objeto usando el sentido de la audición en un ambiente virtual, para esto se hizo uso de herramientas como el motor de desarrollo de la aplicación Unity en conjunto con la librería de Google Resonance Audio para finalmente implementar un sistema funcional sin descuidar la eficiencia.

A lo largo de todo el proceso de desarrollo y pruebas, se lograron cumplir los objetivos estipulados, adicionalmente se lograron capacidades del aplicativo tales como:

- Desarrollar un aplicativo eficiente, cuyo funcionamiento es consistente, que a lo largo de distintas pruebas con variedad de usuarios no se generaron errores y en los varios dispositivos en los que se ejecutó la aplicación, se mantuvo su funcionamiento estable.
- Se mantuvo una buena experiencia de usuario, la integración entre la parte auditiva y visual logran complementarse lo suficiente como para ofrecer una experiencia inmersiva, permitiendo a los usuarios tener interacción de manera intuitiva con el objeto virtual.
- Se demostró el desarrollo de un aplicativo eficiente en términos del consumo de recursos del dispositivo con las debidas pruebas, por lo cual genera la posibilidad de poder realizar implementaciones en ambientes donde se tengan recursos más limitados, aumentando la cantidad de usuarios que tengan acceso al aplicativo e incluso a futuros desarrollos que pueda tener.

De igual forma, se dio a conocer las capacidades y el funcionamiento de la librería Resonance Audio por medio del HRTF, mostrando su desempeño y precisión en entornos virtuales de realidad aumentada junto con la integración con motores de desarrollo como Unity, el cual ofrece múltiples características para crear aplicativos en el sistema operativo Android, así demostrando que con un dispositivo móvil es posible generar una inmersión y simular tales entornos con las debidas herramientas.

Asimismo, con los resultados del proyecto se logró aportar al campo de la realidad aumentada y sonido 3D, puesto que se demostró que es posible desarrollar aplicaciones de alta calidad haciendo uso de herramientas fácilmente accesibles para todo tipo de usuarios al ser código abierto, brindando una gran cantidad de funcionalidades e incluso permitiendo personalización con la debida capacitación en el software o recurso, al poder realizar modificaciones en el funcionamiento del mismo por medio del código fuente.

Finalmente el proyecto cumplió con las expectativas planteadas en un comienzo, con una solución tecnológica funcional, intuitiva y estable, con potencial para futuras y diversas aplicaciones en áreas tales como la educación, el entretenimiento o apoyando la accesibilidad por medio de las debidas implementaciones, incluso permitiendo la incorporación de herramientas adicionales o adaptándolo a otros ambientes en conjunto con nuevas tecnologías.

El repositorio con el proyecto de Unity está disponible en GitHub, libre para futuros desarrollos o mejoras en el siguiente enlace:

<https://github.com/diaohh/Tesis-AuralCube>

# TRABAJO FUTURO

---

Aunque se obtuvieron resultados positivos a lo largo del proyecto, el aplicativo puede llegar a tener diversas mejoras variando en aspectos de la funcionalidad para agregar inmersión en ciertos puntos específicos a la vez que se genera optimización de la misma e incluso incorporar distintas extensiones para agregar diversidad tales como:

- **Adicionar Funcionalidades:** Se puede agregar menús adicionales de selección tanto del objeto 3D como del audio a reproducir, para generar más inmersión en el usuario de forma que tenga más facilidad de navegar en el entorno virtual al sentir familiaridad con la personalización.
- **Incrementar Compatibilidad:** Expandir el aplicativo de forma que no esté limitado al uso del sistema operativo Android, exportándolo en iOS o incluso llevándolo a las nuevas tecnologías tales como las Oculus Quest que a pesar de estar enfocadas en Realidad Virtual empiezan a llegar al campo de la Realidad Aumentada y a la vez explorando los límites a los que puede ser llevado en estos dispositivos.
- **Optimización de Recursos:** Aunque el aplicativo generó resultados positivos a lo largo de las pruebas ejecutadas, se puede analizar la posibilidad de incrementar la optimización para llegar a dispositivos con capacidades más limitadas, mejorando a su vez el desempeño en los dispositivos actuales, debido a que el uso constante de la cámara y el procesamiento del audio pueden resultar tareas exigentes.
- **Oportunidades de mejora en la librería:** Al Resonance Audio ser una librería de código abierto, se da la oportunidad de analizar posibles oportunidades de mejora para llegar a abarcar algunas áreas tales como lo reflejado en las pruebas de aceptación, donde se concluyó que cuando el objeto se posiciona detrás del usuario resultaba ser un poco más complejo de identificar a comparación de las otras posiciones, de forma que realizando ajustes directamente a la librería todas las posiciones queden cubiertas con la misma dificultad.

De esta forma con las posibilidades de trabajo futuro no se espera únicamente perfeccionar la aplicación sino expandir límites de manera que pueda llegar a tener otras aplicaciones en distintas áreas hasta el punto de ser una herramienta con distintas utilidades en varios aspectos.

# Bibliografía

## Referencias

- [1] HRTF. (2019). *Head-Related Transfer Function shapes 3D audio hearing*. Recuperado de <https://www.vrtonung.de/en/hrtf-head-related-transfer-functions-shapes-audio-3d-experiences/>
- [2] Audiosorcerer. (2023). *What is binaural audio? Understanding 3D sound technology*. Recuperado de <https://audiosorcerer.com/post/binaural-audio/>
- [3] Audient. (2020). *The essential guide to binaural simulation for Dolby Atmos*. Recuperado de <https://audient.com/tutorial/the-essential-guide-to-binaural-simulation-for-dolby-atmos/>
- [4] Knowledge. (2016). *How Pokemon Go took augmented reality mainstream*. Recuperado de <https://knowledge.wharton.upenn.edu/article/how-pokemon-go-took-augmented-reality-mainstream/>
- [5] Audioxpress. (2022). *Dolby announces Dolby Atmos personalized rendering beta for immersive mixing using headphones*. Recuperado de <https://audioxpress.com/news/dolby-announces-dolby-atmos-personalized-rendering-beta-for-immersive-mixing-using-headphones>
- [6] Scribbr. (2021). *Exploratory research: Definition, guide, & examples*. Recuperado de <https://www.scribbr.com/methodology/exploratory-research/#:~:text=Exploratory%20research%20is%20a%20methodology,can%20be%20quantitative%20as%20well.>
- [7] Blauert, J. (1997). *Spatial hearing: The psychophysics of human sound localization*. MIT Press.
- [8] Begault, D. R. (1994). *3D sound for virtual reality and multimedia*. NASA Ames Research Center.
- [9] Wightman, F. L., & Kistler, D. J. (1989). Headphone simulation of free-field listening. I: Stimulus synthesis. *The Journal of the Acoustical Society of America*, 85(2), 858–867. <https://doi.org/10.1121/1.397557>
- [10] Zotkin, D. N., Duraiswami, R., & Davis, L. S. (2003). Rendering localized spatial audio in a virtual auditory space. *IEEE Transactions on Multimedia*, 6(4), 553–564. <https://doi.org/10.1109/TMM.2003.819569>
- [11] Algazi, V. R., Duda, R. O., Thompson, D. M., & Avendano, C. (2001). The CIPIC HRTF database. *IEEE Workshop on the Applications of Signal Processing to Audio and Acoustics*. <https://doi.org/10.1109/ASPAA.2001.969552>

- [12] Kulkarni, A., Isabelle, S. K., & Colburn, H. S. (1999). Sensitivity of human subjects to head-related transfer-function phase spectra. *The Journal of the Acoustical Society of America*, 105(5), 2821–2830. <https://doi.org/10.1121/1.426899>
- [13] Dolby Laboratories. (n.d.). Introduction to spatial audio and Dolby Atmos. Recuperado de <https://www.dolby.com>
- [14] Unity Technologies. (n.d.). *Unity Manual*. Recuperado de <https://docs.unity3d.com/Manual/>
- [15] Unity Technologies. (n.d.). *Unity Learn*. Recuperado de <https://learn.unity.com/>
- [16] Unreal Engine. (n.d.). *Unreal Engine Documentation*. Recuperado de <https://docs.unrealengine.com/>
- [17] Epic Games. (n.d.). *Unreal Engine vs Unity: A Comparison*. Recuperado de <https://www.unrealengine.com/>
- [18] Vuforia. (n.d.). *Vuforia Developer Library*. Recuperado de <https://library.vuforia.com/>
- [19] Google. (n.d.). *ARCore Developer Documentation*. Recuperado de <https://developers.google.com/ar>
- [20] Apple. (n.d.). *ARKit Documentation*. Recuperado de <https://developer.apple.com/augmented-reality/>
- [21] Cameron, A., & Harris, J. (2020). A comparison of Unity, Unreal Engine, and Vuforia for augmented reality development. *Journal of Interactive Media*, 12(3), 45–60.
- [22] Mekni, M., & Lemieux, A. (2014). Augmented reality: Applications, challenges and future trends. *Procedia Computer Science*, 5, 59–67. <https://doi.org/10.1016/j.procs.2014.07.070>
- [23] Schreiner, J. (2020). Comparative analysis of ARCore, ARKit, and Vuforia for mobile AR applications. *International Journal of Mobile Development*, 8(2), 95–110.
- [24] Google. (n.d.). *Resonance Audio Documentation*. Recuperado de <https://resonance-audio.github.io/resonance-audio/>
- [25] Lentz, T. (2020). Optimizing spatial audio libraries for mobile applications. *International Journal of Mobile Development*, 9(1), 65–80.
- [26] Cameron, A., & Harris, J. (2019). A comparison of Resonance Audio, Steam Audio, and Oculus Spatializer in immersive applications. *Journal of Audio Engineering*, 10(4), 125–140.
- [27] Lee, Gyeong-Tae & Choi, Sang-Min & Ko, Byeong-Yun & Park, Yong-Hwa. (2022). HRTF measurement for accurate identification of binaural sound localization cues. 10.48550/arXiv.2203.03166.