

**PROTOTIPO DE APLICACIÓN MÓVIL PARA TRATAMIENTO DE SEÑALES DEL  
FONENDOSCOPIO ELECTRÓNICO FONODX**

**DANIEL DAVID JIMENEZ PAZ**



**UNIVERSIDAD MILITAR NUEVA GRANADA**

**FACULTAD DE INGENIERÍA**

**PROGRAMA DE INGENIERÍA MULTIMEDIA**

**BOGOTA D.C.**

**2019**

**PROTOTIPO DE APLICACIÓN MÓVIL PARA TRATAMIENTO DE SEÑALES DEL  
FONENDOSCOPIO ELECTRÓNICO FONODX**



**DANIEL DAVID JIMENEZ PAZ**

**Opción de grado modalidad de desarrollo tecnológico presentado como  
requisito para optar al título de:**

**Ingeniero en Multimedia**

**Director:**

**Eduard Leonardo Sierra Ballén**

**UNIVERSIDAD MILITAR NUEVA GRANADA**

**FACULTAD DE INGENIERIA CAMPUS**

**PROGRAMA INGENIERIA MULTIMEDIA**

**CAJICÁ, 2019**

# Contenido

## Resumen

### 1. Introducción

#### 1.1. Justificación

#### 1.2. Planteamiento del Problema

#### 1.3. Objetivos

##### 1.3.1. Objetivo General

##### 1.3.2. Objetivos Específicos

### 2. Marco de Referencia

#### 2.1. Antecedentes

#### 2.2. Marco Teórico

##### a. Fonocardiografía

##### b. Cardiología y Cardiopatías

#### 2.2.2. Características del sonido

##### a. Coeficientes de Mel

##### b. Transformada rápida de Fourier

##### c. Tono

##### d. Volumen

#### 2.2.3. Métodos para el aprendizaje de maquina

##### a. Discriminante lineal

##### b. Discriminante Cuadrático

##### c. Máquinas de vectores de soporte

#### 2.2.4. El color de la medicina

### 3. Metodología

#### 3.1. Análisis

#### 3.2. Diseño

#### 3.3. Desarrollo

#### 3.4. Diseño Visual de Aplicación

## 4. Desarrollo

### 4.1. Evaluación Heurística

### 4.2. Evaluación con el usuario

### 4.3. Arquitectura de Software

## 5. Métodos de Comparación

### 5.1. Datos para la comparación

### 5.2. Método para Comparación

### 5.3. Experimentos de Comparación

#### 5.3.1. Experimento red neuronal para reconocimiento de patrones

#### 5.3.2. Experimento aprendizaje de clasificación

## 6. Resultados de la Aplicación

### 6.1. Aplicación Móvil

#### 6.1.1. Navegación de aplicación

#### 6.1.2. Código aplicación móvil

### 6.2. Aplicación de Escritorio

#### 6.2.1. Interfaz de la aplicación de escritorio

#### 6.2.2. Código aplicación de escritorio

## 7. Conclusiones

## 8. Anexos

### 8.1. Aplicación Móvil

### 8.2. Aplicación de Escritorio

## Referencias

## TABLA DE FIGURAS

Figura 1: Electrocardiograma de extrasístole ventricular .....	2
Figura 2: Electrocardiograma de extrasístole ventricular .....	2
Figura 3: Modelo de metodología .....	2
Figura 4: Primera pantalla de instrucciones.....	1
Figura 5: Segunda pantalla de instrucciones.....	2
Figura 6: Tercera pantalla de instrucciones.....	2
Figura 7: Pantalla de Preparación .....	1
Figura 8: Pantalla de grabación y comparación.....	2
Figura 9: Pantalla de parar y comparación .....	2
Figura 10: Pantalla de Resultado.....	2
Figura 11: Diagrama Casos de Uso.....	2
Figura 12: Diagrama de objetos.....	2
Figura 13: Modelo para la comparación .....	1
Figura 14: Experimento Red Neuronal 1 .....	2
Figura 15: Experimento Red Neuronal 2 .....	2
Figura 16: Experimento Red Neuronal 3 .....	2
Figura 17: Experimento 1 discriminante lineal 64.1% .....	2
Figura 18: Experimento 1 discriminante Cuadrático: 68.0% .....	2
Figura 19: Experimento 1 SVM Lineal: 68.0% .....	2
Figura 20: Experimento 1 SVM Cuadrático: 68.8% .....	2
Figura 21: Experimento 2 discriminante Lineal: 64.8% .....	1
Figura 22: Experimento 2 discriminante Cuadrático: 67.2% .....	1
Figura 23: Experimento 2 SVM Lineal: 67.2% .....	1
Figura 24: Experimento 2 SVM Cuadrático: 66.4% .....	1
Figura 25: Experimento 3 discriminante Lineal: 64.8% .....	1
Figura 26: Experimento 3 discriminante Cuadrático: 71.9% .....	1
Figura 27: Experimento 3 SVM Lineal: 64.8% .....	1

Figura 28: Experimento 3 SVM Cuadrático: 62.5% .....	1
Figura 29: Primera pantalla de instrucciones.....	2
Figura 30: Segunda pantalla de instrucciones .....	2
Figura 31: Tercera pantalla de instrucciones .....	2
Figura 32: Pantalla de Inicio .....	2
Figura 33: Pantalla de preparación.....	2
Figura 34: Pantalla de graficación .....	2
Figura 35: Pantalla de resultados .....	2
Figura 36: Pantalla de reproducción .....	2
Figura 37: Impresión de Datos.....	2
Figura 38: Graficación .....	2
Figura 39: Interfaz gráfica.....	2

## **Resumen**

Este documento contiene el proceso de diseño y desarrollo de una aplicación móvil para la graficación, grabación y comparación de sonidos fonocardiográficos con el fin de brindar una herramienta para dar una segunda opinión en un diagnóstico.

Por medio de unos requerimientos planteados para el desarrollo de esta aplicación se realizó el desarrollo de un proceso metodológico, guía para el proceso de creación de la aplicación, en donde se encuentran antecedentes de proyectos anteriores donde se han hecho estudios similares, o que contienen parte del desarrollo que se describe alrededor de este documento.

Para los diseños se hace un estudio con respecto a los usuarios, y se plantea una metodología de evaluación con usuarios para realimentación de estos, y cambios posibles en el futuro de la aplicación. También se detallan como se hizo el proceso de desarrollo de la interfaz, y referencias tomadas para este.

## 1. Introducción

Actualmente según la OMS(1), las enfermedades cardiovasculares son el causante de 17.9 millones de personas al año, esto es el equivalente al 31% de las muertes totales anuales en el planeta. Alrededor del 75% de estas muertes suceden en países de mediano y bajo ingreso. Y el 85% de las muertes son causadas por ataques al corazón o paros cardíacos. Aproximando esto a un contexto de país (2) para el 2016, cerca de 60 mil personas murieron a causa de enfermedades cardíacas, lo que convierte a estas enfermedades en las que matan a mayor número de colombianos. Estas enfermedades al ser no transmisibles, no se les suele dar el cuidado y la prevención que se requieren para ser evitadas. Entre las principales causas de estas enfermedades se encuentran la mala alimentación que causa obesidad, o altos niveles de colesterol y triglicéridos, el sedentarismo y la falta de ejercicio, el tabaquismo, el estrés crónico, al igual que algunos factores genéticos.

Otro factor influyente dentro de los fallecimientos por enfermedades cardíacas, es la falta de detección temprana de estas enfermedades. Esto genera que se agraven e incrementen de manera tan desmedida la mortalidad a causa de enfermedades cardiovasculares. Se puede ver reflejado que los países pobres y de medianos recursos tengan el mayor porcentaje de mortalidad por este tipo de enfermedades por la falta de diagnósticos, y formas económicas de detectar y tratar las enfermedades cardiovasculares.

Viendo este problema, se ve como las señales fonocardiográficas, son una forma de hacer un estudio de las enfermedades cardiovasculares, y se vuelven un punto a estudiar, que ha sido olvidado y no muy estudiado por parte de los profesionales en medicina en los últimos años. A raíz de esto, se plantea hacer un estudio de los sonidos del corazón desde su parte matemática, comenzando por el análisis de la representación gráfica, que es el fonocardiograma. Seguido al análisis



gráfico de la señal de audio, se procede a hacer un estudio por medio de coeficientes de Mel, para encontrar diferencias significativas dentro del sistema, también se hace un análisis de Fourier, para eliminar los problemas del inicio de la señal.

Con los datos obtenidos se procede a diseñar una red neuronal que logre diferenciar entre un corazón sano y un corazón potencialmente enfermo, basado en un conjunto de audios que se usan como parte del entrenamiento de la red neuronal.

Al tener estos datos, se crea una aplicación móvil que permite grabar y graficar la señal de audio, al igual que realizar la comparación del audio grabado en el momento y el envío de una sugerencia de diagnóstico.

Para este desarrollo se plantea lo siguiente

### **1.1. Justificación**

Este proyecto busca la realización de un prototipo de aplicación que integre la fonocardiografía al diagnóstico de cardiopatías por medio del dispositivo realizado por la clínica Shaio, Fono DX. Se integrará herramientas multimediales al proceso de desarrollo de este prototipo para lograr una aplicación que logre dar una sugerencia al diagnóstico que esté dando el médico, como una segunda opinión a lo que escucha en el fonendoscopio.

Esto, debido a que las enfermedades cardiovasculares fueron la razón de la muerte de cerca de 60.000 colombianos durante el 2011 que son los datos más recientes de los que se tienen en conocimiento. La razón es que las enfermedades cardiovasculares no son fáciles de detectar por médicos no especializados en el tema de la cardiología. Para dar una ayuda a los profesionales de la medicina y una segunda opinión. Apoyados sobre el diseño del fonendoscopio electrónico FonoDX, que se encuentra en desarrollo en el departamento de investigaciones de la clínica Shaio, sumado al prototipo de aplicación que se quiere desarrollar, una mejora en la pronta diagnosis, y tratamiento de las enfermedades cardiovasculares.

## 1.2. Planteamiento del Problema

Según el boletín No.1 de diciembre 9 del 2013 del Observatorio Nacional de Salud (3), las enfermedades cardiovasculares son unas de las enfermedades no transmisibles responsables de la principal causa de muerte en Colombia y el mundo. Para 2008, 17 millones de personas murieron por esta causa el 30% de todas las muertes registradas en el mundo. Siendo más específicos en el entorno Colombiano, según el boletín No.077 del 21 de marzo del 2014 (2) realizado por el Ministerio de la Salud, "las enfermedades cardiovasculares fueron la causa de muerte de 60.000 colombianos en 2011.

La clínica Shaio está desarrollando un fonendoscopio electrónico para el diagnóstico de cardiopatías, por medio de la comparación de señales, con una biblioteca existente. Para esto se requiere la implementación de una aplicación que permita la transmisión y almacenamiento de la señal de audio que obtiene el fonendoscopio electrónico por medio de un procesamiento digital de las señales que ingresan al Fonendoscopio. Este software debe ser liviano, ya que es necesario que corra en la mayor cantidad de Smartphone posibles, ya que si los requerimientos de sistemas son muy altos, no será muy viable para el público objetivo de este proyecto, el cual es zonas rurales.

Teniendo en cuenta lo anteriormente planteado surge la pregunta de ¿Cómo se puede implementar e integrar un prototipo de aplicación que permita la recepción, graficación, comparación y almacenamiento de una señal recibida desde el fonendoscopio electrónico, que sea para uso de médicos no expertos en el proceso de detección de enfermedades cardíacas por el proceso de auscultación?

### **1.3. Objetivos**

#### **1.3.1. Objetivo General**

- Diseñar y construir un prototipo de aplicación para dispositivo móvil que incluya recepción, verificación y comparación de señales de sonidos cardiacos captadas de un fonendoscopio electrónico, para apoyar el proceso de diagnóstico de cardiopatías por auscultación.

#### **1.3.2. Objetivos Específicos**

- Realizar una revisión bibliográfica que permita definir el estado del arte del proyecto en cuestión.
- Diseñar los elementos lógicos (clases, métodos, objetos) que deben componer la aplicación móvil.
- Establecer un estilo gráfico para el desarrollo de la interfaz de la aplicación.
- Realizar la caracterización y comparación de la señal de un sonido cardiaco.
- Implementar métodos que permitan cargar, reproducir, graficar y comparar una señal de fonocardiografía.

## **2. Marco de Referencia**

Dentro de esta sección del documento, se encuentran antecedentes de desarrollos anteriores de estetoscopios electrónicos, y sistemas de aprendizaje de máquina para la clasificación de señales cardíacas, y su aplicación en diferentes dispositivos. También se encuentra la teoría que se utilizó a la hora de implementar las etapas de diseño, las herramientas que se usaron, la matemática detrás del procesamiento de señales, y la red neuronal que se usó para la evaluación de las señales, y el sistema de comparación.

### **2.1. Antecedentes**

Los antecedentes de este proyecto indican que ya existen prototipos de estetoscopios electrónicos [4] que facilitan el diagnóstico de patologías a partir del análisis de señales de ruido cardíaco, como lo indica el estudio de “Prototipo de fonendoscopio digital” desarrollado en la Fundación Universitaria de san Gil, donde desarrollaron un prototipo capaz de convertir los ruidos cardíacos en señales para ser procesados posteriormente en un software desarrollado en Lab View. Sin embargo, dicha aplicación es netamente de análisis pre-diagnóstico, no compara las señales obtenidas con señales en una base de datos de patologías.

Por otro lado, el proyecto “An electronic stethoscope with diagnosis capability” [5] dice haber logrado aislar características claves en una señal de ruido cardíaco para así poder analizarla en tiempo y frecuencia después de su extracción y segmentación para posteriormente aplicar algoritmos de diagnosis basados en una base de datos con diferentes patologías.

Otra publicación como [6] “The electronic stethoscope”, que es una revisión muy general del estado del arte del estetoscopio electrónico en el cual se describen componentes (sensores, filtros, conductores, etc.) comúnmente utilizados en los diferentes desarrollos de este. También se analizan los algoritmos apropiados para las fases de pre procesamiento y procesamiento de las señales de ruido cardíaco

en una lista. Además de referenciar otros artículos que pueden resultar útiles para el desarrollo del proyecto y los problemas que pueden surgir en este mismo, un ejemplo de esto son las referencias utilizadas para referirse a los algoritmos más comunes de reducción de ruido en señales de sonido cardiaco. Respecto a los avances móviles en Smartphone, esta revisión cita tres aplicaciones con similitudes al proyecto que se quiere desarrollar, la más notable entre estas es SensiCardiac capaz de identificar el sonido cardiaco, localización, segmentación y hacer auto diagnóstico a partir de una red neuronal artificial entrenada con 2000 ruidos cardiacos.

“An electronic stethoscope for heart diseases based on micro-electro-mechanical-system microphone” [7], generalmente, consta del desarrollo y diseño tanto de software como de hardware de un estetoscopio electrónico. En el artículo se describe el hardware utilizado, que consta de un micrófono electrónico INMP411 análogo conectado a un circuito responsable de amplificar y filtrar las señales para enviarlas al Smartphone a través de un adaptador de audio. La aplicación recibe la señal en tiempo real y es capaz de graficar su fonocardiograma en tiempo real además de grabar el sonido y almacenarlo en la nube para posteriormente enviarlo a un servidor perteneciente a un hospital. El documento incluye la descripción de la respectiva experimentación y pruebas del prototipo, donde se especifica que se captaron señales de sonido cardiaco en cuatro áreas diferentes del corazón y fueron enviadas a diferentes dispositivos Smartphone, que generaron con éxito los fonocardiogramas. En términos generales el dispositivo parece estar desarrollado de manera correcta, y en la conclusión del documento se especifica que fue posible desarrollar el estetoscopio digital, sin embargo, no hay una discusión sobre los resultados del experimento, no se habla de las implicaciones de que los fonocardiogramas varíen según el Smartphone. Hace falta concluir si la aplicación puede enviar las señales de manera exitosa a centros de salud, y no hay mención de los formatos que deberían ser manejados para el procesamiento y el envío de las señales en cuestión. En conclusión, esta publicación se vale de recursos (fotos

y gráficos) para probar que el desarrollo se llevó a cabo y funciona, pero se queda corto al explicar sus resultados y las implicaciones de estos.

Mientras que “A low-cost 3-D printed stethoscope connected to a Smartphone” [8] se construyó alrededor de la idea de facilitar el proceso de diagnóstico en pacientes en zonas alejadas de centros médicos. Principalmente se desarrolló hardware y software de un estetoscopio digital con componentes de bajo costo, a excepción del Smartphone la producción del estetoscopio es realmente barata. El desarrollo se encarga de captar, reproducir y graficar señales de audio Raw a partir de APIS para java aplicadas a Smartphones. Además de estas funciones la aplicación permite enviar estas señales a centros médicos; Aun así, hay problemas respecto a la reducción de ruido debido a que el filtro de banda que fue aplicado durante la experimentación no resultó ser tan eficaz. El proyecto posee una discusión de resultados convincente y una conclusión relativamente satisfactoria, básicamente el proyecto fue realizado según lo que los investigadores esperaban, pero se da a entender que hace falta demasiada experimentación para que el prototipo sea un producto completo; la ausencia de personal médico (cardiólogos, anesthesiólogos, doctores, etc.) durante el testeo hacen que los resultados sean fiables desde un punto de vista médico. En conclusión, este desarrollo resulta útil al analizar los componentes de bajo costo que fueron utilizados para lograr un producto funcional pero sus resultados no son confiables debido a que aún es un trabajo en desarrollo.

Dentro del desarrollo del desarrollo de la comparación se realizó la revisión de diferentes artículos alrededor de estudios previos que se relacionarán con el tema de estudios de señales fonocardiográficas y el uso de redes neuronales para la comparación de estas.

El primer artículo al respecto a mencionar es “Characteristics of phonocardiography waveforms that influence automatic feature recognition” [9]

donde el autor se basa en una base de datos brindada por PhysioNet, en esta se realiza un filtrado para reducir la cantidad de ruido en la señal, con el fin de que la comparación se pueda hacer de una mejor manera. Dentro del análisis realizado encontraron que en una de las 5 bases de datos se presentaba un aumento del nivel de señal en los casos no normales. Los pasos que se llevaron a cabo dentro de esta investigación fueron los siguientes: Reducción de ruido, Auto correlación, detección de picos de sonido 1 y 2, medición de la amplitud de los picos y el tiempo de intervalo entre picos.

En otro estudio se realiza el procesamiento de señales para el diagnóstico automático de defectos en la ventrícula septal en recién nacidos y niños en el artículo "Phonocardiography signal processing for automatic diagnosis of ventricular septal defect in newborns and children" [10]. Dentro de este artículo se resume el proceso desde la adquisición de las señales hasta la simulación de las mismas para verificar el funcionamiento del algoritmo y la obtención de datos.

Para el procesamiento de datos inician con el filtrado de los datos con un filtro Butterworth de orden 57 pasa bajo con un corte a 900HZ. Una vez los datos pre-procesados se normalizan para un mejor manejo dentro del sistema, hicieron selección de un espacio de tiempo como referencia para la extracción de las características a evaluar. Para el análisis de la señal se usa una transformada de Fourier de tiempo corto y para la clasificación se usa el algoritmo KNN, que significa K vecinos más cercanos (K Nearest neighbors), el cual es un algoritmo usado dentro del minado de datos, y es muy eficiente a la hora de aplicar aprendizaje de máquina.

## **2.2. Marco Teórico**

### **2.2.1. Fonocardiografía y Sonidos Cardiacos**

#### **a. Fonocardiografía**

La fonocardiografía es la graficación de los sonidos cardiacos y los murmullos. Con esta se busca dar una interpretación más óptima a la auscultación y extender su exactitud. A pesar de su salida temprana, su desarrollo ha sido lento hasta los últimos años donde la calidad de los micrófonos y otros dispositivos necesarios para la toma del audio se incrementó.

Se espera que la gráfica pueda imitar el sonido captado por el humano a través de un estetoscopio. Las señales fonocardiográficas se evalúan con respecto a la ocurrencia de un fenómeno patológico, entre las características de este se encuentran la intensidades relativas, variaciones de la intensidad, y duración de eventos.

La importancia tanto de la fonocardiografía como la auscultación reside en la simplicidad con la que se ejecuta el proceso y las fuertes habilidades que se han demostrado por parte del oído humano para el reconocimiento de patrones en los fenómenos acústicos.

#### **b. Cardiología y Cardiopatías**

La cardiología (11) es una rama de la medicina la cual se encarga de estudiar y tratar el corazón y las enfermedades relacionadas con este, las cuales son conocidas como cardiopatías. Los profesionales que se desempeñan en esta área son conocidos como cardiólogos, y también son un derivado de la medicina interna.

Los procedimientos que se pueden llevar a cabo dentro de la cardiología son electrocardiograma (ECG) el cual captura la actividad eléctrica del corazón, un

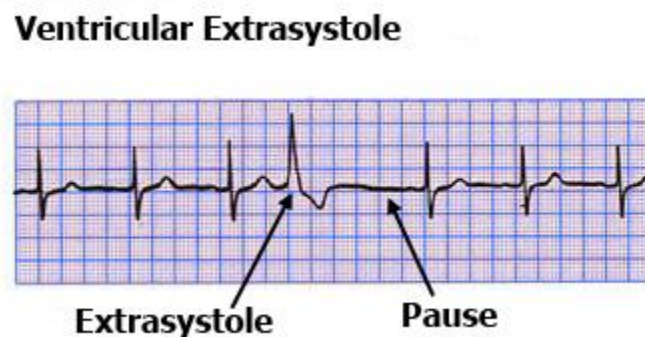


ECG ambulatorio, donde se conectan pequeños electrodos al paciente que carga consigo un monitor de Holter y se le permite realizar actividades diarias regulares para conocer el comportamiento del corazón. Por último se puede ver el ecocardiograma, el cual arroja una imagen de ultrasonido mostrando la estructura, cámaras, y estructuras circundantes del corazón.

Por otro lado las cardiopatías son aquellos males que aquejan a las personas que sufren del corazón, a pesar de que existen muchas de ellas en este trabajo nos concentramos en dos.

La primera de ella se conoce como cardiopatía extrasístole ventricular (12). Esta cardiopatía consiste en un latido adicional, producido por una activación eléctrica anómala la cual se produce en los ventrículos antes de que se produzca un latido normal. Las personas más propensas a sufrir este tipo de cardiopatía son las personas de mayor edad, y esta se puede presentar malestar psíquico o estrés físico. Ciertos alimentos y bebidas como lo son la cafeína y el alcohol pueden generar los síntomas anteriormente mencionados.

**Figura 1: Electrocardiograma de extrasístole ventricular**

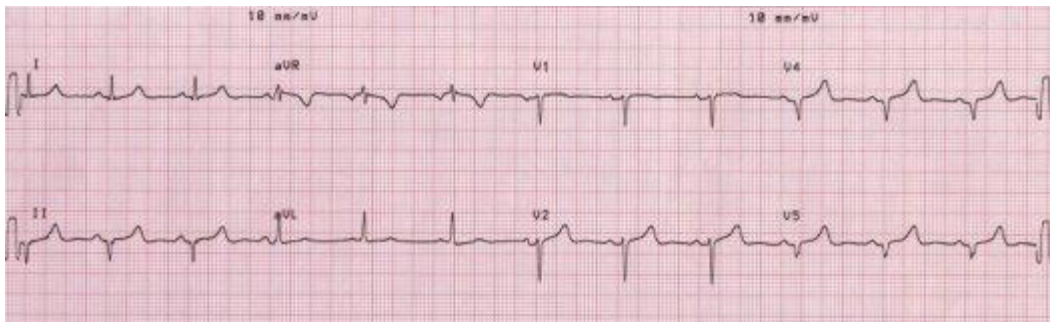


En la imagen anterior se puede observar un ejemplo de como se ve la cardiopatía de extrasístole ventricular en un electrocardiograma.

La siguiente cardiopatía se conoce como soplo cardiaco (13) o murmullo. Estos se pueden identificar a la hora de realizar la auscultación por medio de silbidos o susurros normalmente capturados por el estetoscopio. Esta enfermedad puede surgir desde el nacimiento de forma congénita, o desarrollarse a lo largo de la vida. Como tal no se considera una enfermedad pero si como un indicador de que puede haber un problema cardiaco que no ha sido encontrado.

La imagen presenta el electrocardiograma de una mujer de 67 años que presenta un murmullo.

**Figura 2: Electrocardiograma de extrasístole ventricular**



## 2.2.2. Características del sonido

### a. Coeficientes de Mel

Para la extracción de características se usará el método de Mel Frequency Cepstral Coefficients (MFCC) [14], este modelo consiste en la separación de la señal en pequeños tramos, a cada uno de estos tramos se le aplica una transformada de Fourier discreta para obtener la potencia de la señal, se aplican los filtros correspondientes a la escala de Mel y se suman las energías de cada tramo, tomar el logaritmo de cada frecuencia y aplicar transformada de coseno discreta a estos. Con estos datos se obtienen los coeficientes que se necesitan para el estudio de las señales.

Para la realización de la extracción de las características de los audios a procesar, se debe revisar el código con el que se cuenta para realizar el método de MFCC[14], debido a que este está creado con el fin de hacer procesamiento de voz, lo que conlleva a que las frecuencias tomadas en este son de 4 kHz de ancho de banda, lo que lleva a que se haga un análisis de frecuencia de los audios que se tienen, para con esto, modificar los filtros que se tienen, ya que la tasa de muestreo de los audios se encuentra en 4 kHz, por lo que se perdería mucha información dentro de los sonidos cardiacos si se aplica el sistema que ya se ha desarrollado.

### **b. Transformada rápida de Fourier**

Se hace el uso de la transformada rápida de Fourier (FFT) (15) como parte de las características a evaluar dentro del proceso de la comparación, con esta se busca no incurrir en el problema de que los audios se encuentran desfasados, ya que al igual que MFCC, no tiene en cuenta la posición en el tiempo del sonido, si no, su frecuencia.

Los datos obtenidos al usar la FFT, se les debe realizar una normalización debido a que los resultados arrojados por este método dan una parte real y una imaginaria, con lo que se logra la reinterpretación de estos dos datos como un solo dato real, que ya puede ser utilizado como característica a la hora de realizar el estudio y comparación.

### **c. Tono**

El tono o pitch en inglés es una característica del audio la cual está relacionada con la frecuencia fundamental del audio en un espacio de tiempo. Los datos arrojados al aplicar el pitch al sonido son un total de 295, sacados de un audio de 3 segundos, con un intervalo de 10 milisegundos. A estos datos se les realiza un promedio, con el fin de tener un valor característico por cada audio, que es la media de las frecuencias fundamentales en el intervalo de los 3 segundos.

Una vez se ha realizado todo los procesos sobre el audio para hallar un valor final para el pitch, ya se puede incluir dentro de los experimentos.

#### **d. Volumen**

La característica volumen o Loudness (16) es un valor numérico el cual se halla con un algoritmo el cual divide el audio fragmentos de 0.4 segundos, con una superposición del 75%, y procede a realizar un promedio de los valores hallados para tener un valor escalar final. Esto se suma como una característica extra dentro de los experimentos a desarrollar entorno a la comparación.

#### **2.2.3. Métodos para el aprendizaje de maquina**

Con el fin de encontrar un método (17) que brinde la mejor efectividad a la hora de evaluar las cuatro características planteadas en el subcapítulo anterior, se deciden hacer pruebas con diferentes clases, para poder determinar cuál es la mejor.

Para esto se usan cuatro métodos de aprendizaje supervisado y categórico debido a que todos los datos con los que se está haciendo el estudio se encuentran etiquetados, con los que se espera que el sistema halle un patrón para poder etiquetar aquellos elementos que no se conocen. De estos cuatro métodos dos se clasifican dentro de Naive-Bayes, los cuales son discriminante lineal y cuadrático, y los otros dos se clasifican como máquinas de vectores de soporte una lineal y la otra cuadrática.

##### **a. Discriminante lineal**

El discriminante lineal (18) es una técnica de clasificación lineal supervisada usada en el aprendizaje de máquina, con el fin de reducir la dimensionalidad del espacio el cual se está estudiando, esto se logra al remover elementos redundantes y dependientes al transformar las características de un espacio de mayor dimensión a uno de menor.

El discriminante lineal pretende la maximización de la separación entre las clases maximizando la distancia entre los centroides pertenecientes a cada clase. Con esto se espera tener 2 clases separadas lo suficiente como para que ser separadas por una línea recta. El objetivo del discriminante lineal es proyectar las características en una mayor dimensión, hacia una menor dimensión.

### **b. Discriminante Cuadrático**

El discriminante cuadrático (19), basa su modelo en la probabilidad de cada clase como una distribución Gaussiana, que usa la distribución posterior para estimar la clase dada por el punto de prueba. Los parámetros Gaussianos para cada caso se puede estimar desde los datos de entrenamiento con la estimación de máxima probabilidad. Este modelo es el mejor situado cuando no se tiene mucha información por su facilidad para caracterizar sin importar si el set de datos es pequeño.

Gracias a las características del modelo de discriminante cuadrático, se puede ver que se sitúa como una buena opción, ya que no se dispone con un gran set de datos, y las características que se están extrayendo son limitadas.

### **c. Máquinas de vectores de soporte**

Este método está basado en discriminantes, acompañado por el principio de Vapnik, el cual menciona que no se debe resolver un problema más grande que el problema anterior. Alpaydin en su libro "Introduction to the machine learning" (20) menciona como ejemplo el hecho de que si la tarea es determinar o aprender el determinante, no es necesario estimar la densidad de las clases o la probabilidad posterior, solo se necesitan estimar los límites donde reside la clase.

Los parámetros del modelo lineal se escriben de tal manera que se convierten en un subconjunto de datos llamados vectores de soporte. Esto son los que definen los límites que se plantean dentro del sistema para la separación.

Estos límites están definidos por medio de líneas o hiperplanos que separan las clases. En el caso de aplicación dentro de este proyecto se manejan dos tipos de líneas, dependiendo del tipo de máquina de vectores de soporte (SVM por sus siglas en inglés) que se esté usando. La primer SVM que se plantea es lineal, lo que nos indica que los límites están definidos por ecuaciones de rectas. La segunda SVM es cuadrática, por lo que los límites están dados por curvas parabólicas para realizar la separación entre las clases que se planteen en el experimento.

#### **2.2.4. Método para la Evaluación de Heurística**

La evaluación heurística (21) permite revisar la usabilidad de lo que se está desarrollando sin la necesidad de usuarios que hagan uso del producto. Este se realiza por medio de la inspección por parte de expertos de la calidad de la interfaz a la hora de ser usada, esto basado en el cumplimiento de unos principios definidos dependientes del método elegido para la evaluación.

Como principal objetivo la evaluación heurística mide la calidad de la interfaz y su facilidad para ser aprendido y usado por primera vez. A su vez este evalúa el flujo de tareas del aplicativo e inspecciona problemas potenciales, por medio del experto, tratando de predecir errores que se pudiese encontrar.

Como método para la evaluación heurística dentro de este proyecto se escogen los 10 principios de Nielsen (26), los cuales como primera versión fueron presentados en 1990, y revisados más adelante en 1994, siendo resumidos para el surgimiento de lo que se conoce hoy como los 10 principios heurísticos de Nielsen.

Los 10 principios heurísticos de Nielsen son los siguientes:

- Visibilidad del estado del sistema
- Relación entre el sistema y el mundo real
- Control y libertad del usuario
- Consistencia y estándares

- Prevención de errores
- Reconocimiento antes que recuerdo
- Flexibilidad y eficiencia de uso
- Estética y diseño minimalista
- Ayuda y documentación

#### **2.2.4. El color de la medicina**

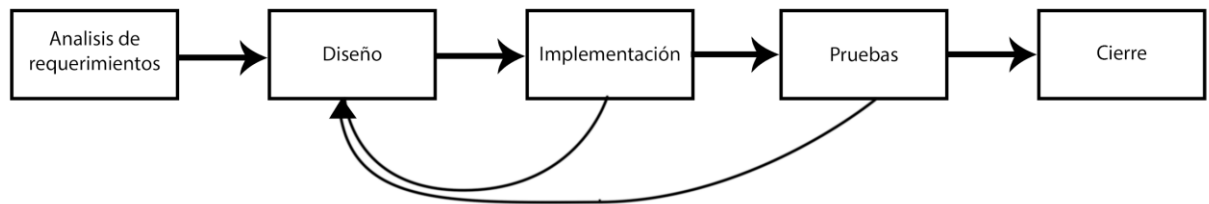
David Pantalony (22), nombra su investigación alrededor de los colores usados dentro de la medicina. Este menciona que desde 1914 se hizo el uso del color verde. En esta época se usó ese color por parte del doctor Harry Sherman, este lo hizo usando la teoría de colores complementarios, basado en el color de la hemoglobina, al implementar todo esto dentro de su sala de operaciones, logró percibir un gran cambio dentro de su percepción de detalles. Alrededor del mismo tiempo, el doctor William Ludlow se dedicaba también al estudio del color y sus efectos terapéuticos. Dentro del estudio realizado, encontró que el blanco se considera como un color negativo, y los convalecientes necesitan colores positivos como lo son la gama de verdes suaves los que pueden contribuir a una mejor recuperación de los pacientes.

### 3. Metodología

La metodología que se lleva a cabo para este proyecto se fundamenta en investigación y desarrollo de aplicaciones móviles realizados previamente, se planea utilizar una metodología ágil de desarrollo, para crear un entorno de retroalimentación junto con el cliente, y evitar el gasto de recursos y tiempo dentro del desarrollo del proyecto.

Para esto se tiene un plan de desarrollo por fases planteado en el siguiente esquema:

**Figura 3: Modelo de metodología**



Las etapas de desarrollo se desarrollaran de la siguiente manera.

#### 3.1. Análisis

En la etapa de análisis se obtuvieron los requerimientos primarios del cliente para la aplicación que se está desarrollando, teniendo en cuenta, las necesidades básicas para el funcionamiento correcto de la misma.

Tras este paso, se clasificaron los requerimientos planteados, entre los funcionales y los no funcionales, con el fin de establecer cuáles serán parte del desarrollo principal, en el caso de los funcionales, todo lo que se tendrá dentro del sistema, y los no funcionales como parte de lo que se deberá implementar posteriormente como requisitos de usabilidad, rendimiento



Por otra parte el análisis conlleva un estudio del usuario final, para poder adaptar la aplicación a las necesidades, y con esto generar aceptación por parte de quien se beneficiará de la aplicación.

Esto conlleva al análisis de los requerimientos de entorno, en busca de la plataforma sobre la que se desarrollara, y el entorno gráfico que para esta aplicación

Los requerimientos funcionales que se han planteado son, el diseño e implementación de 4 etapas, estas se desarrollaran en el orden que se mencionan a continuación, etapa de recepción y grabación de datos, graficación de la señal obtenida, procesamiento de la señal que ingresa a la aplicación, y por último se realizará la comparación de la señal entrante, con una señal de referencia.

Los requerimientos no funcionales, se definen como aquellos que no son parte esencial del desarrollo, pero sí lo son para la mejor aceptación del producto. Entre los requerimientos planteados para este proyecto se tienen diseño visual del producto, con estándares institucionales, la optimización de la aplicación.

La fase de desarrollo e implementación pueden generar que se regrese a la fase de análisis y diseño para generar un cambio que sea completamente necesario en pro del mejoramiento sustancial de la aplicación.

### **3.2. Diseño**

La etapa de diseño se compone del desarrollo de esquemas, bocetos, diagramas que guiaron la construcción de la aplicación.

Lo primero que se debe diseñar es la definición del escenario en el que se va a implementar la aplicación, en este caso tiene que dar características que la hagan liviana, permitan el correcto uso en dispositivos de gama media, y el uso sin internet.

Después de esto se deben diseñar los modelos de casos de uso, modelo estructural, modelo de interacción, cada uno de estos modelos define una etapa importante en el desarrollo.

- El primer modelo mencionado, el de casos de uso define la ruta de cómo se comportará la aplicación, cuáles serán los usos, y la secuencia que se tendrá dentro de la aplicación.
- Para el segundo modelo, se define la estructura del software, lo que se puede considerar como objetos y clases dentro del programa.
- El modelo de interacción define como el usuario hará el uso de la aplicación, cuál será la secuencia de las acciones que este lleve dentro de la misma.

Se realizará un diagrama de los requerimientos con el fin de trazar una meta por medio de objetivos que se tendrán con el fin de llenar los requisitos.

Para el diseño visual y la documentación del proyecto se usarán los estándares institucionales, tanto de la clínica Shaio, como de la universidad Militar Nueva Granada.

### **3.3. Desarrollo**

El desarrollo se basa en implementar las fases del diseño del producto en cuestión.

- Codificación: esta primera etapa se escribe el código en el lenguaje escogido, y se implementan cada uno los módulos planteados dentro de los modelos planteados dentro del diseño.
  - Implementación de sistema de recepción, grabación y reproducción: Para este primer módulo se realizará la implementación de un código el cual recibirá la señal de audio desde el micrófono, usará un sistema de registro para guardar la información, y por último un modo de reproducción de los audios guardados.
  - Implementación sistema de graficación: Para este caso se usará una librería que permite transformar la señal de audio en una señal gráfica

que representa el comportamiento de la información que está ingresando.

- Sistema de procesamiento de señales: En este módulo, se realizará se le aplicará filtros para mejorar la forma de la señal, y obtener parámetros claves para el posterior uso en otro módulo.
- Sistema de comparación: esta etapa realizará la comparación de parámetros de una señal base, con parámetros de una señal de entrada, con esto se pretende caracterizar la señal de un corazón.
- Pruebas unitarias: esta etapa se prueban cada uno de los módulos separados con el fin de detectar fallas o errores para realizar la depuración de una manera más sencilla evitando el gasto de tiempo.
- Iteración con cliente: en este punto se muestran los módulos de la aplicación al cliente para que éste los apruebe, o haga correcciones, en caso de ser necesarias correcciones, se volverá a la etapa de diseño para replantear el módulo, o módulos que sean necesarios corregir.
  - Documentación del código: Se realiza en medio de la codificación, cada módulo se documenta para el desarrollo, para realizar cambios dentro de la aplicación dentro de ciertos módulos y que estos sean fáciles de encontrar.

### **3.4. Diseño Visual de Aplicación**

Durante el proceso de diseño se tuvo en cuenta un artículo llamado “La auscultación del corazón, un arte en vías de extinción” en este se habla de la influencia de la tecnología sobre la técnica de la auscultación, y como de la mano con la tecnología se puede dar un gran paso dentro de esta.

A la hora de pensar en el diseño de la aplicación se pensó en que los usuarios son médicos, personas con mucha información en la cabeza y que si desea lograr generar impacto con una aplicación como la que se propone, se debe dar una confiabilidad y una facilidad a la hora del uso, porque de complicarse, generaría repudio por parte de los posibles usuarios.

Así se llevó a cabo un diseño de la aplicación teniendo en cuenta la teoría del color planteada por Sherman y Ludlow (22), usando el verde como color de referencia, al igual que mantener la simplicidad, y la usabilidad dentro de la aplicación. El reto se tornó alrededor de lo que debe hacer la aplicación que son los siguientes puntos:

- Graficar el sonido que está ingresando
- Grabar el sonido.
- Procesar la señal.
- Comparar con la base de datos.

Con estos requerimientos a implementar se plantea en primera instancia un tutorial donde se indica el proceso de preparación del paciente para la toma del examen como se observa a continuación.

**Figura 4: Primera pantalla de instrucciones.**

## Tome los datos del paciente



**Fuente:** Modificado por el autor. Basado en diseño de Freepik, tomado de [https://www.freepik.es/vector-gratis/doctora-portapapeles\\_3132792.htm](https://www.freepik.es/vector-gratis/doctora-portapapeles_3132792.htm)

**Figura 5: Segunda pantalla de instrucciones.**

**Siente al paciente y pida  
que se retire la camisa**



**Fuente:** Modificado por el autor. Basado en diseño de gstudioimagen, tomado de [https://www.freepik.es/vector-premium/doctor-masculino-paciente-femenino-sentado\\_2122998.htm](https://www.freepik.es/vector-premium/doctor-masculino-paciente-femenino-sentado_2122998.htm).

**Figura 6: Tercera pantalla de instrucciones.**

**Ten listo tu telefono para hacer  
la graficación y comparación**

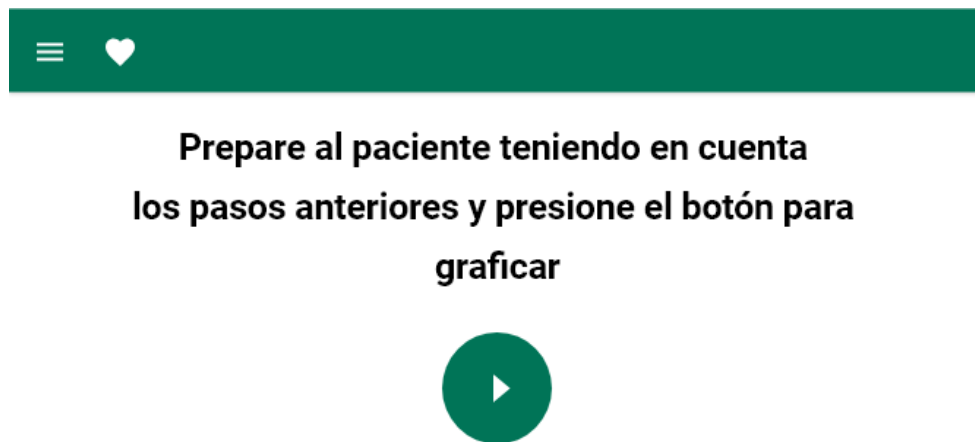


**Fuente:** Modificado por el autor. Basado en diseño de Biro Eموke, tomado de <https://www.shutterstock.com/es/image-vector/mobile-phone-hand-holding->

[smartphone-flat-413552269?src=LCU07\\_1rBBzgacv\\_XyqwpA-1-87](https://smartphone-flat-413552269?src=LCU07_1rBBzgacv_XyqwpA-1-87).

Una vez completado el tutorial, el usuario entrará a una pantalla que le indicará que debe realizar la preparación indicada para comenzar con el proceso de auscultación. En este se le indicará que presione un botón para que continúe a la siguiente sección de la aplicación.

**Figura 7: Pantalla de Preparación**



El botón observado en la parte superior en forma de corazón se da para volver al inicio. Una vez el usuario presione el botón con la flecha en su interior la página esta enviará al usuario a una ventana donde puede observar cómo se grafica la señal que está ingresando, y posee dos opciones, grabar y comparar, para poder realizar la comparación se requiere que el usuario haya hecho una grabación de audio previamente. En las siguientes 2 imágenes se verá cómo cambia de estado el botón de grabación al ser presionado, al igual que el resultado al realizar la comparación.

Figura 8: Pantalla de grabación y comparación

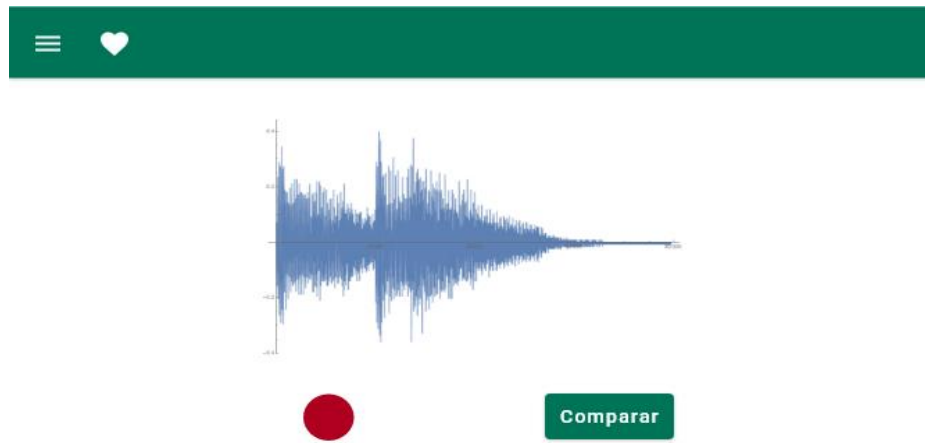


Figura 9: Pantalla de parar y comparación



### Figura 10: Pantalla de Resultado



Resultado: La señal del corazón se asemeja a un Corazón sano

Este sistema de 6 pantallas trata de simplificar al máximo la ruta que debe recorrer el usuario con el fin de no generar una sobrecarga mental, y el esfuerzo sea mínimo para obtener el resultado más óptimo.



## 4. Desarrollo

En esta sección se encuentra todo lo concerniente al desarrollo de la aplicación, las herramientas y materiales usados para el diseño y puesta en marcha de la aplicación y del sistema de comparación.

Se escogió el conjunto de datos brindado por Peter Bentley [9] en su página web “Classifying Heart Sounds Challenge”. De esta página web fueron extraídos los audios etiquetados para su posterior estudio. El conjunto de audios que se escogió para realizar la extracción de las características y realizar la evaluación está conformado por un total de 656 audios, los cuales se dividen en 195 sin etiquetar, 46 de la cardiopatía extrasístole ventricular, 66 de la cardiopatía murmullo, 29 de murmullo con ruido, 200 de un corazón normal, y 120 de un corazón normal con ruido. Se descartan para la extracción de características los audios sin etiquetar, los audios que contienen alto nivel de ruido. En segunda instancia se descartaron todos los audios con duración inferior a 3 segundos. Al realizar el proceso de selección de información se obtiene un total de 207 audios, los cuales están compuestos por 115 audios de corazón normal y 92 de cardiopatías mencionadas anteriormente. La clasificación de los datos permite llevar a cabo la siguiente fase del desarrollo, en esta se hará uso de la herramienta MatLab con el fin de realizar la extracción de datos y simular una red neuronal para ver como las características extraídas responden a este sistema.

### 4.1. Evaluación Heurística

Principio	Revisión
Visibilidad del estado del sistema	Cumple
Relación entre el sistema y el mundo real	Cumple
Control y libertad del usuario	No Cumple
Consistencia y estándares	Cumple

Prevención de errores	No Cumple
Reconocimiento antes que recuerdo	Cumple
Flexibilidad y eficiencia de uso	No Cumple
Estética y diseño minimalista	Cumple
Ayudar a los usuarios a reconocer	Cumple
Ayuda y documentación	No Cumple

## 4.2. Evaluación con el usuario

El paso a seguir es realizar la evaluación con el usuario, para esto se realizará una prueba dada por los siguientes pasos:

1. Exponer la idea que se quiere desarrollar.
2. Realizar una entrevista al usuario.
3. Se le entrega el prototipo de la aplicación para hacer la evaluación de usabilidad.
4. Lo siguiente es realizar preguntas al respecto de la experiencia.
5. Realizar un análisis de la información obtenida, realizar los cambios pertinentes, y realizar otra iteración.

Para la realización de la evaluación de usabilidad se generaron unos objetivos que se le entregarán a cada usuario para que realice, teniendo en cuenta el éxito o fracaso de la prueba, al igual que el tiempo y la facilidad para el usuario de navegar a través de la aplicación. Los objetivos que se diseñaron para esto fueron los siguientes:

**Usuario 1:**

- Navegar a través del tutorial
- Graficar el audio
- Grabar el Audio
- Repetir la grabación

**Usuario 2:**

- Saltar el tutorial.
- Graficar el audio.
- Grabar Audio.
- Comparar Audio.

**Usuario 3:**

- Saltar el tutorial.
- Volver al tutorial.
- Graficar audio de la aplicación en distintos casos.

Para las primeras pruebas se puede realizar dentro de un entorno controlado como el de un emulador, que nos permite correr la aplicación en una máquina más potente como lo es el computador.

En segunda instancia se prueba la aplicación en dispositivos móviles, con usuarios reales, para poder ver cómo funciona en un entorno real, como interactúan los usuarios con ella, medir tiempos de respuesta en diferentes dispositivos, y ver si se están cumpliendo los requisitos planteados desde un principio. En caso que se encuentren fallos, o el cliente pida ajustes dentro de la aplicación se debe volver a la etapa de diseño para corregir los errores encontrados.

### 4.3. Arquitectura de Software

Figura 11: Diagrama Casos de Uso

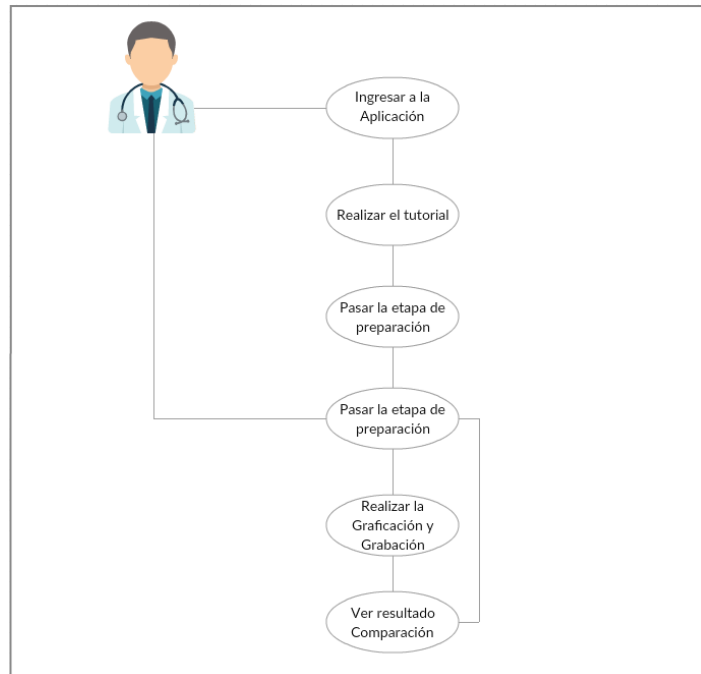
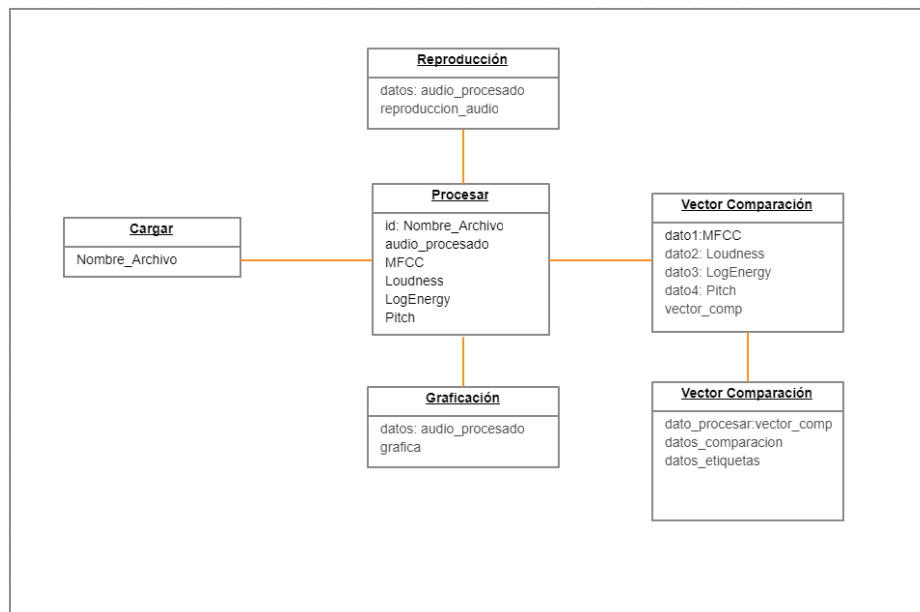


Figura 12: Diagrama de objetos



## 5. Métodos de Comparación

En esta sección se dan a conocer los datos que se usaron, la manera en que se organizaron, y los experimentos que se realizaron con los diferentes métodos de comparación con los cuales se escogió el más óptimo para la aplicación final.

### 5.1. Datos para la comparación

Se escogió el conjunto de datos brindado por Peter Bentley [9] en su página web "Classifying Heart Sounds Challenge". De esta página web fueron extraídos los audios etiquetados para su posterior estudio. El conjunto de audios que se escogió para realizar la extracción de las características y realizar la evaluación está conformado por un total de 656 audios, los cuales se dividen en 195 sin etiquetar, 46 de la cardiopatía extrasístole ventricular, 66 de la cardiopatía murmullo, 29 de murmullo con ruido, 200 de un corazón normal, y 120 de un corazón normal con ruido. Se descartan para la extracción de características los audios sin etiquetar, los audios que contienen alto nivel de ruido. En segunda instancia se descartaron todos los audios con duración inferior a 3 segundos. Al realizar el proceso de selección de información se obtiene un total de 207 audios, los cuales están compuestos por 115 audios de corazón normal y 92 de cardiopatías mencionadas anteriormente. La clasificación de los datos permite llevar a cabo la siguiente fase del desarrollo, en esta se hará uso de la herramienta MatLab con el fin de realizar la extracción de datos y simular una red neuronal para ver como las características extraídas responden a este sistema.

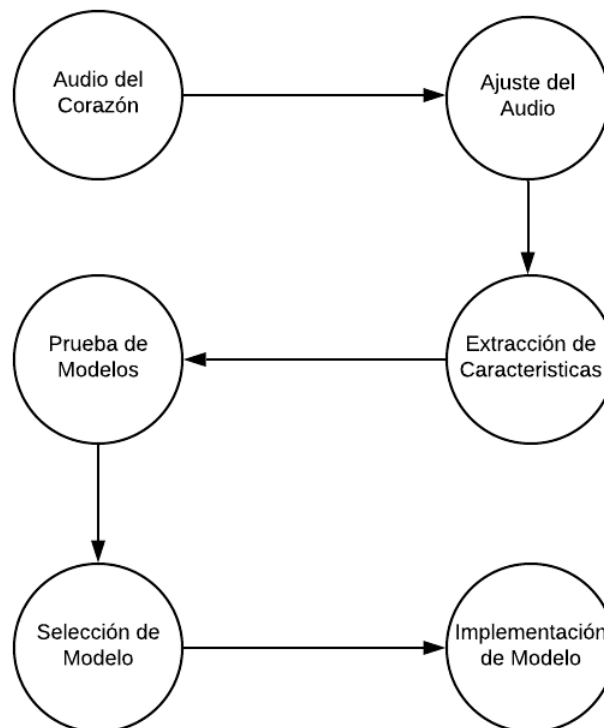
Para realizar las pruebas de los audios, se decidió cortar todos a los primeros 3 segundos, considerando que esto es un tiempo prudente con el que se pueden evaluar al menos 3 ciclos de latido del corazón.

Después de realizar la evaluación y extracción de los coeficientes de Mel, esto arroja una matriz de 14x298, el primero de los 14 datos corresponde a la energía de la señal en la ventana de espacio que se está evaluando, y los otros 13

datos corresponden a los coeficientes de Mel. Se hace un promedio de los registros de energía y los coeficientes de Mel de la señal completa con el fin de obtener un vector de 14x1. Estos se usan en uno de los 2 tipos de experimento.

## 5.2. Método para Comparación

Figura 13: Modelo para la comparación



Partiendo desde los audios que se tienen de la base de datos del corazón se realiza la selección de aquellos que cumplen con una duración mayor a 3 segundos. Ya con estos datos listos para procesar, se realiza el ajuste para que estos 3 posean las mismas características de tiempo. Se prosigue realizando la extracción de características, se escogen 5 tipos de características a evaluar con combinaciones entre ellas, estas fueron escogidas debido a estudios previos realizados alrededor del análisis de señales, la comparación de audios, y sus grandes usos dentro del aprendizaje de máquina.

Se procede al uso de las herramientas de MatLab de reconocimiento de patrones de una red neuronal, y la de aprendizaje de clasificación, con el fin de evaluar diferentes métodos. Una vez dentro de estos se realiza la selección por medio de una comparación de resultados, en los casos de exactitud del método y la herramienta, y la estabilidad que se genera al realizar múltiples estudios de la misma.

### **5.3. Experimentos de Comparación**

En esta sección se describen y muestran los experimentos de comparación realizados con MatLab para la generación de la base de datos.

#### **5.3.1. Experimento red neuronal para reconocimiento de patrones**

El primer experimento que se muestra a continuación, se usa la herramienta “Neural Network Pattern Recognition” de MatLab (24). Con esta se esperaba encontrar resultados que se pudieran dar veracidad al experimento.

Se realiza la revisión de estos datos dentro de la herramienta de reconocimiento de patrones, lo que arroja un resultado como el que se puede ver a continuación.

Figura 14: Experimento Red Neuronal 1

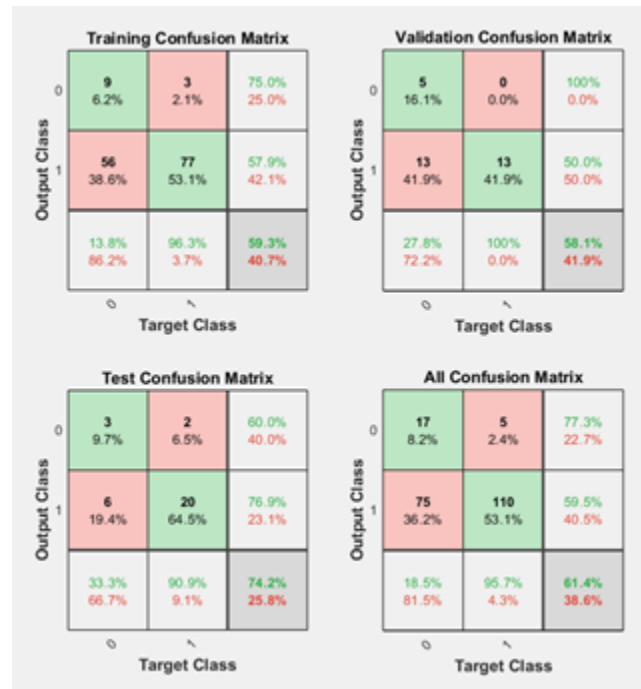


Los resultados de la anterior comparación arrojan un error del 28%, por lo que se considera evaluar otra característica a evaluar dentro de esto.

Se realiza una transformada de Fourier a la señal, y teniendo en cuenta la parte real de esta se procede a hallar la magnitud del vector de valores correspondiente a 12.000 con el fin de reducir a la magnitud total de la transformada a un valor de vector, esto para poder probar si las magnitudes de la transformada de Fourier varían entre un corazón sano y uno enfermo. Con esto se obtiene la siguiente matriz de confusión:



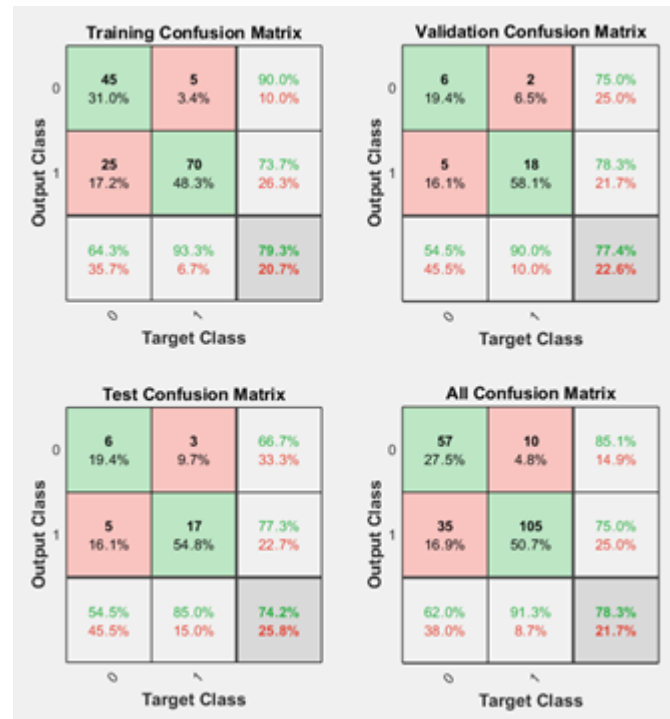
Figura 15: Experimento Red Neuronal 2



Al analizar los resultados de únicamente la magnitud de la transformada de Fourier, se puede notar que existe un error del 38,6% lo cual no mejora respecto a la prueba anterior, por lo que se decide añadir la característica de la frecuencia del audio.

A los 12.000 datos que componen estas se le realiza una media para obtener un solo dato y que pueda aplicarse junto con los otros 2 valores probados anteriormente.

**Figura 16: Experimento Red Neuronal 3**



Al mezclar estas 3 características, se obtiene un promedio de error del 21.7%, de este se puede apreciar que el error es mucho más bajo que los anteriores, debido a que la unión de las 3 características permite diferenciar de mejor manera.

Estos resultados se obtuvieron después de realizar múltiples intentos, sobre los mismos datos, esto arrojaba valores muy diferentes que dejaban dudas al respecto de si era el método más apropiado para dar un resultado final.

### 5.3.2. Experimento aprendiz de clasificación

Para este experimento se utilizó la aplicación “Classification Learner” (25) de la herramienta MatLab. Para este experimento se restringieron los datos a un total de 128. 32 de cada cardiopatía, y 64 datos de corazones sanos en las condiciones expuestos al principio del capítulo. Esta aplicación permite el uso de diferentes

algoritmos, de los cuales para este experimento usaron discriminantes lineales y cuadráticos, SVM linear, SVM cuadrático.

Para el primer experimento se usaron los datos después de ser procesados con el algoritmo de mfcc, el cual arrojó, una matriz de 128x14. Se añadió una columna extra con las etiquetas, para que el sistema pueda realizar la clasificación. Los resultados de los experimentos:

**Figura 17: Experimento 1 discriminante lineal 64.1%**

	Enfermo	Sano
Enfermo	36	28
Sano	18	46

**Figura 18: Experimento 1 discriminante Cuadrático: 68.0%**

	Enfermo	Sano
Enfermo	42	22
Sano	19	45

**Figura 19: Experimento 1 SVM Lineal: 68.0%**

	Enfermo	Sano
Enfermo	35	29
Sano	12	52

**Figura 20: Experimento 1 SVM Cuadrático: 68.8%**

	Enfermo	Sano
Enfermo	42	22
Sano	18	46

El SVM cuadrático es el que mejor resultado ofrece para el primer experimento, este da un porcentaje del 68.8%. Al analizar los datos de la matriz de confusión se puede decir que de cada 64 pacientes enfermos, 22 no serían tratados, lo que se traduce en un 34.37% de las cardiopatías pasarían por alto. Por el otro lado, 18 personas de las 64 saludables, serían marcadas como enfermas, por lo cual un 28.12% de los pacientes, serán diagnosticados de manera errónea.

El segundo experimento que se llevó a cabo, se le añadió una columna extra a la matriz, con la cual se añade la normal obtenida de procesar la transformada rápida de Fourier, con esto se evalúa una matriz de 16x128. De estos se obtienen los siguientes resultados.

**Figura 21: Experimento 2 discriminante Lineal: 64.8%**

	Enfermo	Sano
Enfermo	39	25
Sano	20	44

**Figura 22: Experimento 2 discriminante Cuadrático: 67.2%**

	Enfermo	Sano
Enfermo	46	18
Sano	24	40

**Figura 23: Experimento 2 SVM Lineal: 67.2%**

	Enfermo	Sano
Enfermo	37	27
Sano	15	49

**Figura 24: Experimento 2 SVM Cuadrático: 66.4%**

	Enfermo	Sano
Enfermo	42	22
Sano	21	43

Este experimento, arroja como mejor resultado 67.2% en 2 casos, el primero es usando el algoritmo de discriminante cuadrático, cuando se hace el análisis de los datos por parte de la matriz de confusión. En ese caso el 28.12% de los pacientes con cardiopatías son diagnosticados sanos, frente a un 37.5% de pacientes los cuales su resultado es negativo. El segundo caso es el de SVM lineal, en el cual, un 23.43% de las personas evaluadas se da un resultado positivo para enfermedad, y por su contraparte un 42.18% de los pacientes arrojaron un falso positivo, lo que sugeriría no seguir tratamiento. De escogerse este experimento para proceder, lo ideal sería que se decantara por el algoritmo de SVM lineal, debido a su bajo porcentaje de falsos positivo, lo que deja a menos personas por fuera de tratamiento, las repercusiones de este sería una implicación alta de pacientes sanos que se diagnostiquen enfermos.

Para el tercer experimento se retira la columna de la transformada de Fourier, y se agregan 2 nuevas, la primera es el pitch, y la segunda es loudness, con lo cual la matriz a evaluar es de 16x128. Con estos datos los resultados obtenidos son:

**Figura 25: Experimento 3 discriminante Lineal: 64.8%**

	Enfermo	Sano
Enfermo	41	23
Sano	22	42

**Figura 26: Experimento 3 discriminante Cuadrático: 71.9%**

	Enfermo	Sano
Enfermo	45	19
Sano	17	47

**Figura 27: Experimento 3 SVM Lineal: 64.8%**

	Enfermo	Sano
Enfermo	38	26
Sano	19	45

**Figura 28: Experimento 3 SVM Cuadrático: 62.5%**

	Enfermo	Sano
Enfermo	41	23
Sano	25	39

Al realizarse el análisis de los resultados de los diferentes algoritmos se ve que el mejor resultado se da con el discriminante cuadrático. Este da un porcentaje de clasificación igual al 71.9%, este se descompone en 26.56% de falsos negativos, y 29.68% de falsos positivos. Los resultados encontrados implican que los diagnósticos arrojados por este grupo de datos se puede usar a modo de sugerencia para un médico, más no como un diagnóstico definitivo. A pesar de no poder brindar un diagnóstico definitivo, este algoritmo reduce de gran manera los falsos negativos y positivos al tiempo, a diferencia de otras cuando uno de los 2 era menor al 30% el otro lo supera este porcentaje, en este caso los 2 resultados se mantienen bajo el 30%.

## **6. Resultados de la Aplicación**

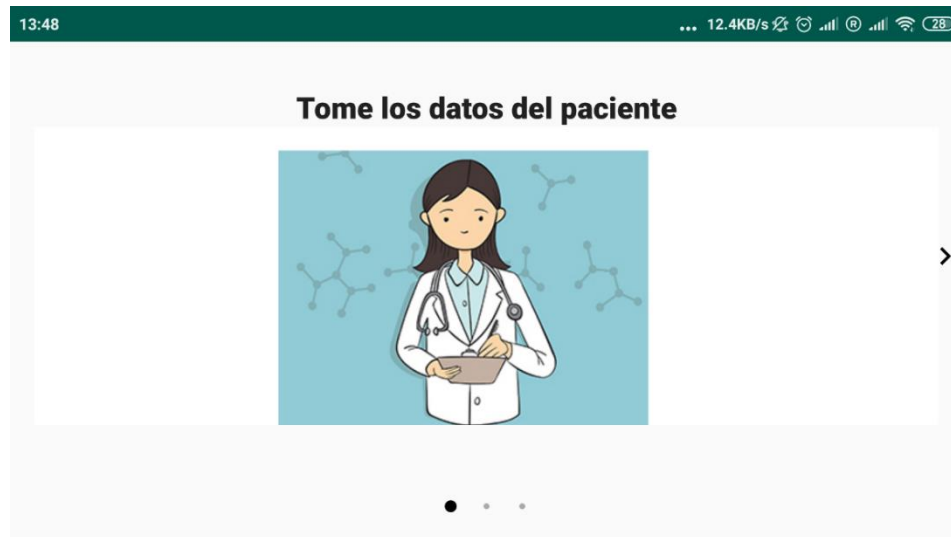
### **6.1. Aplicación Móvil**

Se desarrolló una aplicación móvil en lenguaje Java, en el entorno de desarrollo Android Studio. Esta aplicación está basada en los diseños mostrados en el capítulo 4.5. de este trabajo. Esta aplicación contiene las instrucciones de uso, una pantalla de preparación, la graficación en tiempo real, la grabación del audio, un bosquejo de lo que sería la aplicación conectada al sistema de comparación, y un reproductor donde se puede reproducir un sonido grabado anteriormente. Para esta se cumplen algunos de los principios heurísticos planteados por Nielsen, para hacer esta aplicación lo más usable posible.

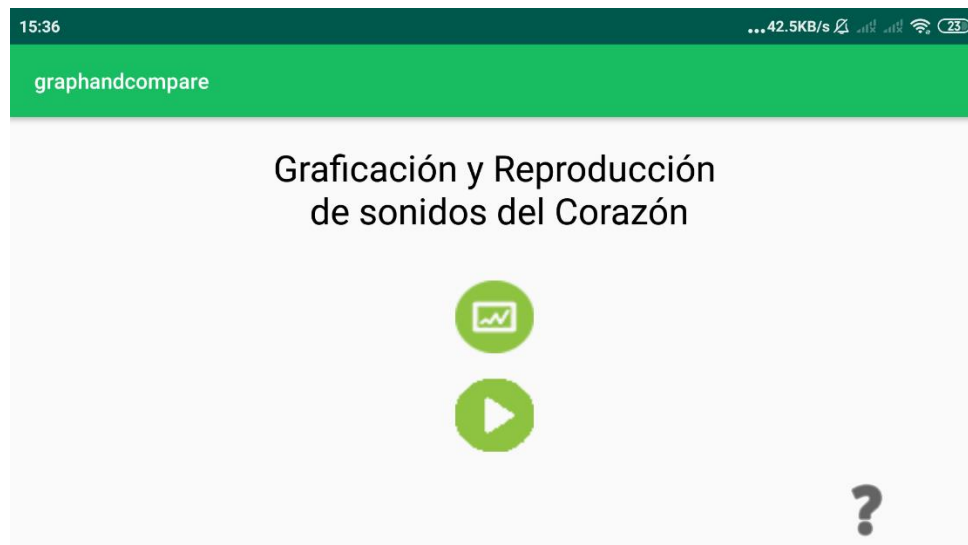
Por cuestiones técnicas, no se pudo incluir el modelo de comparación y procesamiento del audio, debido a que la capacidad de hacer esta clase de análisis matemáticos requiere una gran potencia por parte del celular, y ciertas librerías que no se poseen dentro del entorno de desarrollo.

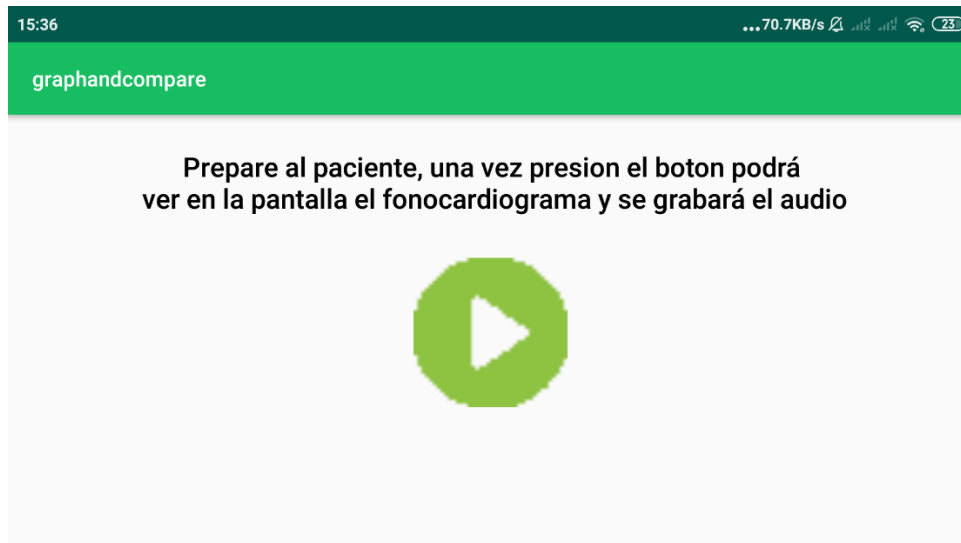
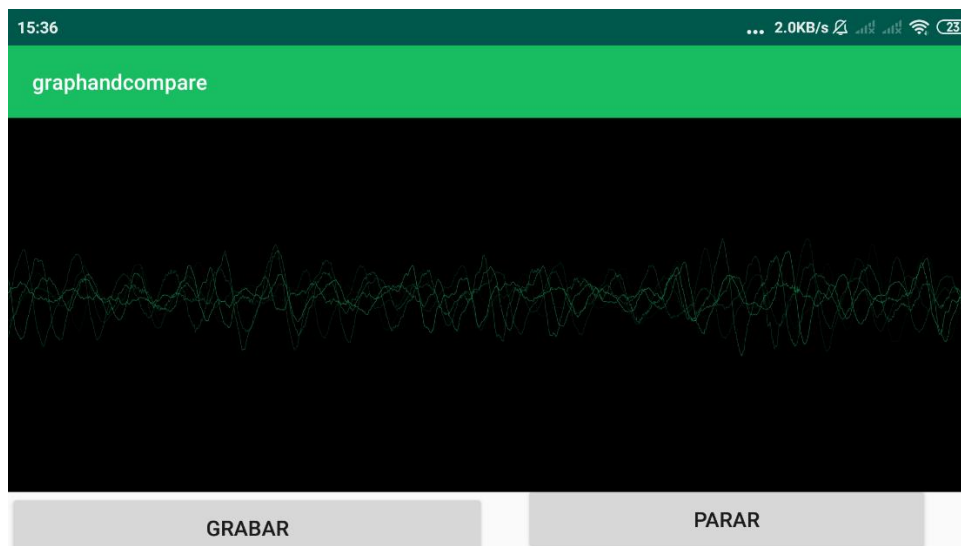
#### **6.1.1. Navegación de aplicación**

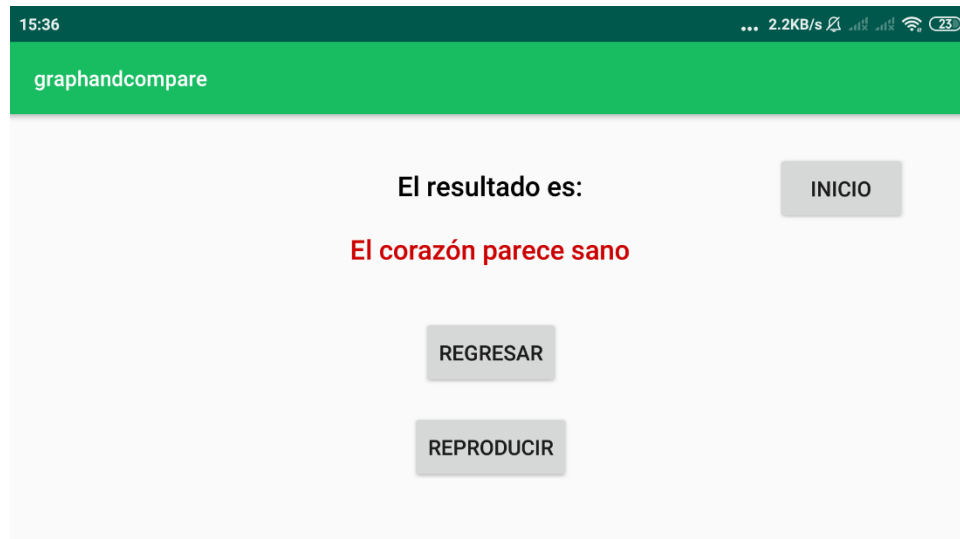
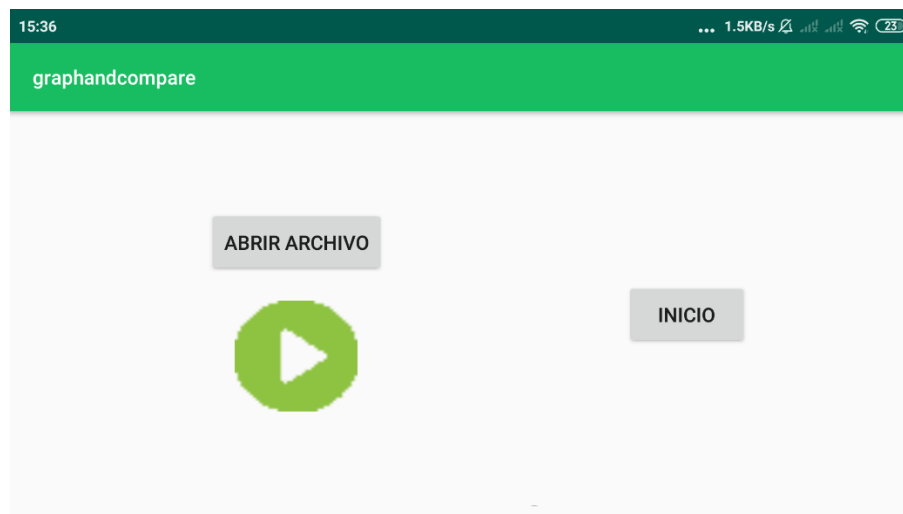
A continuación se muestran las impresiones de pantalla de la aplicación en su estado actual de funcionamiento:

**Figura 29: Primera pantalla de instrucciones****Figura 30: Segunda pantalla de instrucciones**



**Figura 31: Tercera pantalla de instrucciones****Figura 32: Pantalla de Inicio**

**Figura 33: Pantalla de preparación****Figura 34: Pantalla de graficación**

**Figura 35: Pantalla de resultados****Figura 36: Pantalla de reproducción**

La pantalla de graficación no permite fijar la grabación en el momento, por lo que no se puede hacer un análisis de la gráfica final, este problema se resuelve en la segunda aplicación. La última pantalla es un bosquejo, debido a que no realiza ningún proceso ni comparación, esta se muestra para dar pie a lo que se lograría con esta.

### 6.1.2. Código aplicación móvil

La aplicación se compone de 7 actividades propias y 4 obtenidas de un módulo llamado waveform.

6 actividades están referidas a las pantallas que posee la aplicación final y una a un módulo ViewPager, estas son:

- MainActivity
- PrepareActivity
- GraphActivity
- ResultActivity
- LoadActivity
- TutorialActiviy
- ImageAdapter

MainActivity es la actividad que se muestra al inicio de la aplicación, esta posee métodos que permiten navegar dentro de la aplicación por medio de los botones, y está encargada de realizar la solicitud del permiso de escritura de la aplicación.

PrepareActivity solo está compuesta de un por el método que permite abrir la actividad de Graficación.

GraphActivity permite realizar la grabación y graficación del sonido de audio. Para esto, está actividad realiza la solicitud de permisos de grabación y uso de audio dentro del dispositivo. Hace uso de la librería MediaRecorder de Android, para realizar la grabación y posterior creación del archivo de sonido dentro del sistema. Esta clase realiza el llamado del módulo waveform, este actúa como librería dentro de la aplicación, y permite la graficación del sonido en tiempo real en frecuencia.

ResultActivity muestra un resultado genérico que no cambia, ya que está establecido como un archivo de texto dentro del código, además permite volver al inicio, a la pantalla de preparación, o ir a la pantalla de reproducción.

LoadActivity hace uso de las librerías MediaPlayer, y AudioManager de Android para realizar la reproducción de archivos que se encuentran dentro del teléfono móvil. Este audio es obtenido por la aplicación por medio de la función MaterialFilePicker, que permite escoger un archivo, y al escogerlo, la función PlayFile, reproduce el sonido automáticamente.

TutorialActivity trabaja junto ImageAdapter para generar un carrusel de imágenes, este está dedicado para que el usuario pueda entrar, y ver el tutorial. El TutorialActivity se usa como medio para hacer el llamado del ImageAdapter, en el cual se configura el orden de las imágenes y se carga el contenido que va a ser mostrado.

## **6.2. Aplicación de Escritorio**

La aplicación de escritorio se realiza como complemento de la aplicación móvil, para lograr los requerimientos faltantes que no se pudieron implementar dentro de la aplicación móvil. Se desarrolló en el lenguaje de programación Python, debido a su gran versatilidad para la creación de modelos de aprendizaje de máquina, al igual que su implementación y la amplia librería matemática que posee, lo que permite la extracción de características satisfactoria. Esta permite la carga por medio de la url con la ubicación del audio, el procesamiento de la señal de audio, su graficación para análisis después de ser grabada, su reproducción y la comparación.

Se deja planteado la unión de estos dos sistemas por medio de un servidor web, para la recepción y envío de la información que esta aplicación tiene la capacidad de procesar. Al estar pensada como back end, no posee una interfaz gráfica por fuera de la graficación, debido a que se quiere mostrar la capacidad de esta para realizar esta tarea.

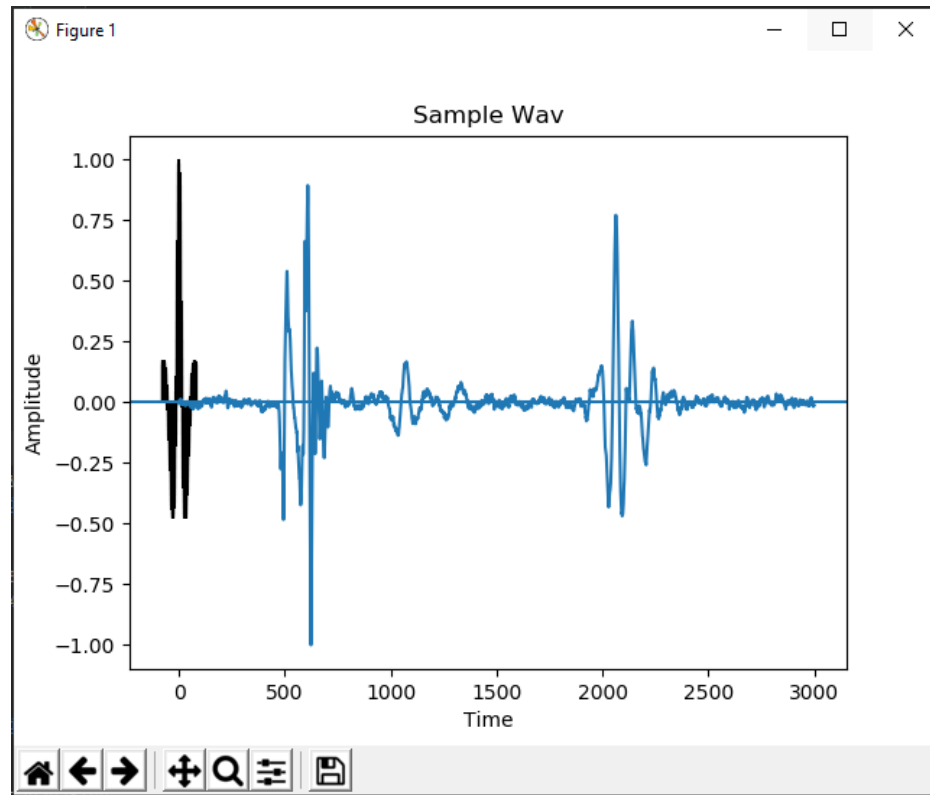
### 6.2.1. Interfaz de la aplicación de escritorio

A continuación se muestran los resultados que arroja dentro de la consola en este caso de PyCharm, que fue el entorno de desarrollo escogido para esta aplicación:

**Figura 37: Impresión de Datos**

```
C:\Users\Daniel\AppData\Local\Programs\Python\Python36-32\python.exe "C:/Users/Daniel/Desktop/2018-2/Tesis/speechpy-master/example/featureExtraction .py"
C:\Users\Daniel\AppData\Local\Programs\Python\Python36-32\lib\site-packages\scipy\io\wavfile.py:273: WavFileWarning: Chunk (non-data) not understood, skipping it.
WavFileWarning)
298 12000 80 40.0
mfcc = [-3.4381534  4.3593957  2.33730383  0.91152719  0.89897824  0.31025324
 -0.11881644 -0.06554652 -0.32734069 -0.25323873  0.18391474 -0.22676289
 0.23463432]
tamano= 13
logenergy -8.428591966184447
loudness= -20.571972316872685
C:\Users\Daniel\AppData\Local\Programs\Python\Python36-32\lib\site-packages\scipy\io\wavfile.py:273: WavFileWarning: Chunk (non-data) not understood, skipping it.
WavFileWarning)
pitch = 400.0
dato [-8.42859197e+00 -3.43815340e+00  4.35939570e+00  2.33730383e+00
 9.11527190e-01  8.98978237e-01  3.10253243e-01 -1.18816438e-01
 -6.55465233e-02 -3.27340686e-01 -2.53238726e-01  1.83914741e-01
 -2.26762886e-01  2.34634319e-01 -2.05719723e+01  4.00000000e+02]
['Enfermo']
C:\Users\Daniel\AppData\Local\Programs\Python\Python36-32\lib\site-packages\scipy\io\wavfile.py:273: WavFileWarning: Chunk (non-data) not understood, skipping it.
WavFileWarning)
```

En esta imagen se puede apreciar la extracción de características, y que se van a evaluar con el modelo de aprendizaje de máquina. También se puede observar el vector de características nombrado dato, el cual será el que se comparará. Por último se observa la respuesta del método frente a las características del audio, en este caso, un corazón detectado como enfermo.

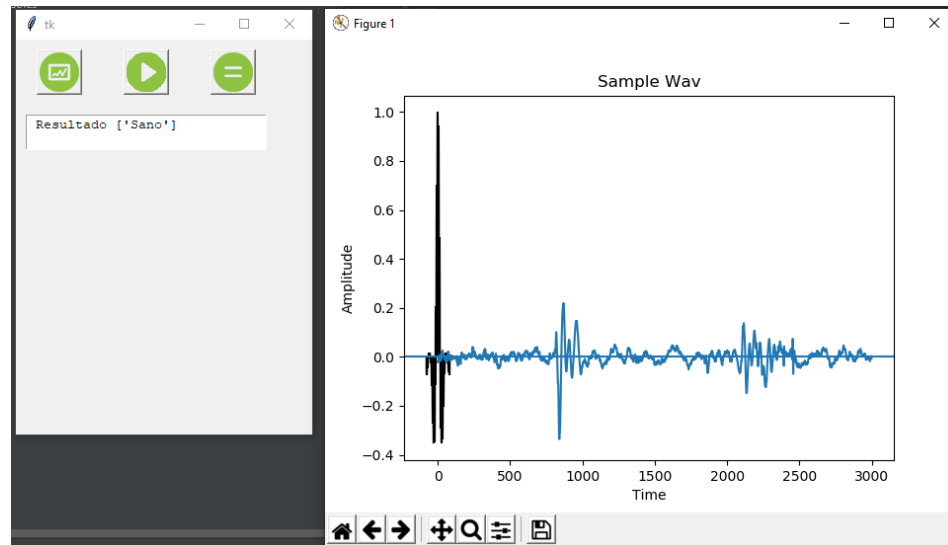
**Figura 38: Graficación**

**Fuente:** Autor

En esta figura se puede observar la gráfica del audio del corazón y los 3 ciclos que esta contempla. Debido a que no se puede tomar una captura de pantalla a la reproducción de audio, esta no se muestra dentro de las figuras.

Como último paso dentro del desarrollo de la aplicación de escritorio para suplir los requerimientos se decidió realizar una interfaz a aplicación donde el usuario de manera intuitiva pueda hacer uso de esta, al mismo tiempo, para la interfaz de usuario se usaron los mismos lineamientos de diseño como en la aplicación móvil, esto se puede observar en la siguiente figura.

**Figura 39: Interfaz gráfica**



**Fuente:** Autor

### 6.2.2. Código aplicación de escritorio

La aplicación de escritorio usa 3 funciones que permiten su funcionamiento, aparte de esto, hace uso de 14 librerías que permiten la estabilidad del código sin riesgo de fallos, las librerías están mencionadas en la sección 9.2 de este documento. La interfaz de la aplicación se hizo con la biblioteca Tkinter, que permite el uso de botones y es sencilla de modificar para lograr un buen aspecto visual con el usuario.

Al cargar el dato, de este se extraen características con el fin de realizar el análisis de esta señal más adelante. La primera característica extraída por la aplicación es la frecuencia, esta se usa en los siguientes procesos como medio de referencia. A continuación se extrae la señal de la onda. Después se realiza un pre-énfasis, y se definen los cuadros de análisis. Una vez con estos datos usando la librería de speechpy, se realiza la extracción del MFCC, un registro de energía, el volumen y el tono. Por último, en una variable tipo matriz llamada dato, se reúnen estos coeficientes, los cuales están listos para ser comparados.



Dentro de esta se realiza el llamado de las 3 funciones las cuales son:

- Comparación
- Graficación
- Reproducción

La primera función permite realizar la comparación, esta realiza la carga de un modelo de clasificación entrenado previamente. Esta carga un modelo entrenado anteriormente y carga el vector con los datos para ser usado. Este modelo entrenado se tomó de datos procesados anteriormente con MatLab, y las etiquetas que se tenían de los archivos conocidos. Esto usando la librería Sklearn de Python, la cual permite implementar el análisis de discriminante cuadrático, y además, permite evaluar el método.

La segunda función, toma la librería MatPlot, y genera una gráfica del audio, tomando intervalos de 1ms por cada punto de la señal, para un total de 3000 puntos graficados.

Por último, la función de reproducción usa la librería winsound, con la que reproduce el audio introducido al principio.

## 7. Conclusiones

El desarrollo de una comparación por medio de aprendizaje de máquina requiere que se estudien diferentes parámetros para el procesamiento del audio con el fin de obtener características, estudiar diferentes tipos de algoritmos de aprendizaje de máquina, y escoger los métodos con mejores porcentajes de exactitud, sensibilidad y susceptibilidad de los diferentes sistemas estudiados. Las características escogidas dentro de la comparación se hacen con el fin de hacer un sistema que tenga la capacidad de procesar, y realizar los cálculos matemáticos de una manera rápida, por lo cual se escogieron 16 características en total para el desarrollo del análisis y posterior implementación dentro de la comparación.

El diseño visual como parte integral dentro del diseño de una aplicación se hizo teniendo en cuenta los colores que influyen dentro de la medicina y cuáles son los mejores a la hora de crear o diseñar algo con enfoque médico.

Se logró el desarrollo de aplicaciones, una móvil y otra para computador, para poder completar los requerimientos establecidos al inicio del proyecto, la aplicación móvil cubre la graficación en tiempo real y grabación del sonido cardiaco que se desea, la segunda aplicación cubre la carga, la reproducción, el procesamiento y la comparación del audio. Para el desarrollo de las aplicaciones se requirió la profundización en el estudio de los lenguajes Java y Python, y el uso de diferentes librerías para lograr la correcta implementación de las funciones que se plantearon para el desarrollo de este proyecto.

La aplicación se entrega cumpliendo todos los requerimientos que se plantearon pero no se hizo de manera que se pudiera ejecutar desde una única interfaz. Para lograr esto se debe atravesar una barrera técnica la cual se basa en la unión de los 2 software por medio de un servidor o framework, el cual permita que se compartan los datos, o usar un solo lenguaje de código que permita el desarrollo de una aplicación móvil, y que tenga la capacidad de realizar los cálculos

matemáticos necesarios para lograr la extracción de características del sonido que se ingrese.

Se puede realizar la mejora del sistema de comparación agregando otras características, teniendo en cuenta que para esto, se debe contar con un buen servidor que pueda realizar estas operaciones en la nube, al igual que se debe obtener una mayor biblioteca con la que se pueda mejorar los porcentajes y reducir el margen de error presente dentro del modelo de clasificación.

Por último, realizar las pruebas de usabilidad planteadas para que con estas se pueda mejorar la aplicación alrededor del estilo visual, y la usabilidad, para lograr una aplicación más agradable en el momento que se realice el despliegue dentro de la plataforma que se escoja.

## 8. Anexos

### 8.1. Aplicación Móvil

Se agrega dentro de la carpeta junto a este archivo, una carpeta con el nombre “aplicación móvil”, la cual contiene el código de desarrollo de esta, y un archivo .apk que se puede instalar en celulares con tecnología Android, la cual contiene lo mostrado en el capítulo 7.1.

### 8.2. Aplicación de Escritorio

Se incluye dentro de los archivos de este proyecto una carpeta con el nombre de “aplicación de escritorio” que contiene el código del desarrollo el cual puede ser ejecutado en cualquier entorno de desarrollo de Python, para correr este código se requieren las siguientes librerías:

- Spicy
- Numpy
- Speechpy
- Soundfile
- Pyloudnorm
- Matplotlib
- Pitch
- Pandas
- Pickle
- Sklearn
- Windsound.

## Referencias

- [1] Cardiovascular diseases (CVDs). World Health Organization [En Línea], 17 de mayo del 2017. Disponible en internet: [https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-\(cvds\)](https://www.who.int/en/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds)).
- [2] Colombia enfrenta epidemia de enfermedades cardiovasculares y diabetes. Ministerio de Salud [En Línea], 21 de marzo del 2014. Disponible en internet: <https://www.minsalud.gov.co/Paginas/Colombia-enfrenta-epidemia-de-enfermedades-cardiovasculares-y-diabetes.aspx>
- [3] Boletín No. 1, Diciembre 9 de 2013. Instituto Nacional de Salud [En Línea], 9 de Diciembre del 2013. Disponible en internet: <https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/IA/INS/Boletin-tecnico-1-ONS.pdf>.
- [4] SANTAFÉ-RAMÓN, Y., et al. Diseño y Construcción de un Estetoscopio Digital Inalámbrico. En: V Latin American Congress on Biomedical Engineering CLAIB 2011 May 16-21.(2011, Habana, Cuba). Springer Berlin Heidelberg, 2013. p. 686-689
- [5] W. W. Myint and B. Dillard. An electronic stethoscope with diagnosis capability. En: Proceedings of the 33rd Southeastern Symposium on System Theory (Cat. No.01EX460).2001. p. 133-137
- [6] LENG,Shuang, et al. The electronic stethoscope. En: BIOMEDICAL ENGINEERING ONLINE. vol. 14, p. 66
- [7] OU,D., et al. An electronic stethoscope for heart diseases based on micro-electro-mechanical-system microphone. 2016. p. 882-885
- [8] AGUILERA-ASTUDILLO,C., et al. A low-cost 3-D printed stethoscope connected to a smartphone. 2016. p. 4365-4368
- [9] STANTON,S.; TSIMENIDIS,C. and MURRAY,A. Characteristics of phonocardiography waveforms that influence automatic feature recognition. 2016. p. 1173-1176

- [10] M. Ghaffari, et al. Phonocardiography signal processing for automatic diagnosis of ventricular septal defect in newborns and children. En: 2017 9th International Conference on Computational Intelligence and Communication Networks (CICN).2017. p. 62-66
- [11] Nordqvist, Christian. What is cardiology. Medical News Today [En Línea], 3 de abril del 2017. Disponible en internet:  
<https://www.medicalnewstoday.com/articles/248935.php>.
- [12] Mitchell, L. Brent. Latidos ventriculares ectópicos; Extrasístoles ventriculares. Manual MSD [En Línea]. Disponible en Internet:  
<https://www.msmanuals.com/es-co/hogar/trastornos-del-coraz%C3%B3n-y-los-vasos-sangu%C3%ADneos/arritmias/extras%C3%ADstoles-ventriculares>.
- [13] Personal Mayo Clinic. Soplos cardíacos. Mayo Clinic [En Línea]. Disponible en internet: <https://www.mayoclinic.org/es-es/diseases-conditions/heart-murmurs/symptoms-causes/syc-20373171>.
- [14] RABINER, Lawrence and SCHAFER, Ronald. Theory and Applications of Digital Speech Processing. 1st ed. Upper Saddle River, NJ, USA: Prentice Hall Press, 2010.
- [15] Seppänen, Jarno. Audio Signal Processing Basics. Audio Research Group [En Línea], 27 de mayo de 1999. Disponible en Internet:  
<https://www.cs.tut.fi/sgn/arg/intro/basics.html>.
- [16] integratedLoudness. MathWorks [En Línea], 2016. Disponible en internet:  
<https://www.mathworks.com/help/audio/ref/integratedloudness.html>.
- [17] KLAPURI, Anssi and DAVY, Manuel. Signal Processing Methods for Music Transcription. Berlin, Heidelberg: Springer-Verlag, 2006.

- [18] SAWLA, Srishti . Linear Discriminant Analysis. Medium [En Línea], 5 de junio del 2018. Disponible en Internet: <https://medium.com/@srishtisawla/linear-discriminant-analysis-d38decf48105>.
- [19] NAGARKAR, Sharvil. Linear Discriminant Analysis. Medium [En Línea], 30 de enero del 2019. Disponible en internet: <https://medium.com/@sharvill/linear-discriminant-analysis-37859b65abd1>.
- [20] ALPAYDIN, Ethem. Introduction to Machine Learning. 2nd ed. The MIT Press, 2010.
- [21] MODROÑO, TOMÁS. Metodologías de UX: Evaluación Heurística (PARTE I). Interactius [En Línea], 1 de septiembre del 2017. Disponible en internet: <https://blog.interactius.com/metodolog%C3%ADas-de-ux-evaluaci%C3%B3n-heur%C3%ADstica-parte-i-b5d02b566987>
- [22] STIPE, D. Color in Medical Products. FORMA Medical Device Design [En Línea]. Disponible en Internet: <https://www.formamedicaldevicedesign.com/white-papers/color-medical-products/>
- [23] BENTLEY, P AND NORDEHN G AND COIMBRA, M AND MANNOR, S. The PASCAL Classifying Heard Sounds Challenge. Peter J Bentley [En Línea], 2011. Disponible en internet: <http://www.peterjbentley.com/heartchallenge>.
- [24] MATLAB 2008a. Neural Network Pattern Recognition. MathWorks, 2008.
- [25] MATLAB 2011b. Classification Learner. MathWorks, 2011.
- [26] MOLICH, Rolf and NIELSEN, Jakob. Improving a Human-computer Dialogue. En: COMMUN.ACM. mar.vol. 33, no. 3, p. 338-348