

**Universidad Autónoma Metropolitana  
Unidad Azcapotzalco**

División de Ciencias Básicas e Ingeniería  
Licenciatura en Ingeniería en Computación

---

**CLASIFICACIÓN DE  
LLANTOS DE BEBÉ  
CON *WAVELETS***

Proyecto que presenta:  
**Hernández Nieves María Sara**

Para obtener el título de:  
**Ingeniero en Computación**

Asesor:  
Dr. Oscar Herrera Alcántara.

México, D.F.

Agosto de 2012

# Contenido

Agradecimientos .....	4
Resumen.....	5
Capítulo 1. ....	6
Introducción .....	6
Justificación .....	7
1.1. Objetivo General: .....	7
1.2. Objetivos Particulares: .....	7
1.3. Organización del Reporte.....	8
Capítulo 2.....	9
Antecedentes .....	9
2.1. Referencias Internas .....	9
2.2. Referencias Externas .....	10
2.3. Marco Teórico.....	11
2.4. Archivos WAV.....	11
2.5. Codificación Parson .....	13
2.6. Transformada Wavelet Discreta.....	13
2.7. Red Neuronal .....	14
Capítulo 3.....	16
Descripción Técnica.....	16
3.1. Descripción del Sistema .....	17
3.1.1. Módulos Principales.....	17
Capítulo 4.....	19
Diseño y Desarrollo .....	19
4.1. Especificaciones Técnicas.....	20
Capítulo 5.....	26
Recursos Utilizados.....	26
5.1. Hardware .....	27
5.2. Software .....	27
5.2.1. Java.....	27

5.2.2. Java ME.....	27
5.2.3. RapidMiner.....	27
5.3. Otros.....	28
Capítulo 6.....	29
Resultados y Conclusiones.....	29
Resultados .....	29
Conclusiones .....	32
Bibliografía .....	33
Manual de Usuario .....	35

## Agradecimientos

---

*“La imaginación es más importante que el conocimiento”*

Albert Einstein

- A Dios por permitirme culminar una etapa más de mi vida.
- A mi Mami por estar siempre cuando más lo he necesitado a pesar de sí misma y de su comprensión a pesar de mis desplantes, Te amo Mamá.
- A mi Papi que se que está conmigo siempre trascendiendo fronteras de espacio-tiempo y con todo su amor.
- A los tres hermanos que Dios me dio, a uno de ellos hermano del alma, por su amor, cariño, comprensión y paciencia otorgados antes, durante y después de esta etapa, los quiero con todo mi corazón.
- A JAAL por todo el apoyo, cariño y paciencia demostrados.
- A mis amig@s porque en algún momento también me ayudaron de una u otra forma, Analy, David, Ely, Gregorio, Marco, Mary, Mario, Octavio, Sergio.
- A mis profesores que me brindaron sus conocimientos, la oportunidad de aprender y algunas veces algo más que una simple asesoría. Antonin, Claudia, Dolores, Ernesto, Francisco Javier S.R, Franz, Josue, Rafael P. F., Risto Fermín, Silvia G.B.
- A la UAM-A por brindarme la oportunidad de aprender y desarrollarme.
- Al Dr. Carlos A. Reyes García, porque sin conocerme permitió que trabajara con su material.
- Al Dr. Oscar Herrera, por el apoyo, tiempo, asesoría y amistad brindados durante la realización de este proyecto.

## Resumen

---

El presente documento describe la metodología que fue utilizada para la realización del proyecto terminal “*Clasificación de llantos de bebé con wavelets*”, así como de una pequeña aplicación para dispositivos móviles.

El proyecto en conjunto consta de cinco partes o módulos principales, los cuales hacen posible la interpretación de un llanto de bebé almacenado en un archivo .wav de longitud variable y que es procesado dando como resultado una de cinco clases de llanto específicos.

Debido a que en la mayoría de los estudios aplicados a los llantos de bebé no utilizan *Wavelets*, se realizó a aplicación de la Transformada Wavelet Discreta (TWD) con la finalidad de extraer las características distintivas del llanto y así realizar la clasificación de los llantos en base a estas características con la ayuda de una red neuronal.

Se desarrolló una aplicación en Java ME que ayuda a la interacción entre un dispositivo móvil y una computadora para el procesamiento de los datos de entrada dados por el usuario de la aplicación.

A pesar de no poseer archivos de audio suficientes para la realización del proyecto, los resultados obtenidos al final fueron satisfactorios.

# **Capítulo 1.**

## **Introducción**

---

El presente proyecto realiza la clasificación de llantos de bebé con la ayuda de la Transformada Wavelet Discreta y la aplicación de una Red Neuronal Artificial.

---

## Justificación

El ser humano tiene distintas formas de comunicación con su entorno, una de las principales es por medio del habla, pero en el caso de los bebés, el llanto es la forma primordial de comunicación con el medio que le rodea y es en la mayoría de las veces lo que más preocupa a los padres, debido a que no siempre es posible averiguar de manera rápida y sencilla cuál es la causa de este llanto.

Si consideramos que detrás del llanto del bebé, se esconde una necesidad o bien un deseo por satisfacer, lo cual implica que el bebé llora para poder conseguir lo que no puede obtener por sí mismo. Entonces resulta necesario conocer la causa del llanto para poder satisfacer esa necesidad o deseo de la manera más rápida y eficaz posible.

Tanto los padres, como especialistas en el sector salud aprenden a distinguir entre los diferentes tipos de llantos de los bebés, utilizando la percepción individual para la diferenciación auditiva y la interpretación de los diferentes tipos de llanto [8]. Éste tipo de clasificación es completamente subjetiva y el único sustento que tiene radica en el entrenamiento y el grado de especialización del sentido auditivo de cada persona.

Es de gran relevancia destacar que la importancia de este tipo de clasificaciones es el hecho de que ayudan a facilitar la comunicación entre el bebé y las personas que le rodean, siempre y cuando el bebé no haya comenzado a hablar, por qué es justo en ese momento en que el estudio pierde sentido, ya que a partir de ese momento el bebé adquiere la capacidad de comunicarse con su entorno de manera un poco más simple y los llantos pasan a un segundo término, debido a que ya no son su principal forma de comunicación.

### **1.1. Objetivo General:**

Diseñar e implementar un clasificador de llantos de bebé utilizando la Transformada *Wavelet* Discreta (TWD).

### **1.2. Objetivos Particulares:**

- Crear un módulo para la lectura y pre-procesamiento de archivos de audio (.WAV), los cuales almacenan llantos de bebé.

- Crear un módulo para la extracción de las características distintivas de los llantos de bebé mediante la aplicación de la TWD.
- Realizar la clasificación de los llantos de bebé en archivos de audio (.WAV).
- Diseñar una interfaz gráfica de usuario de una aplicación software que facilite el uso de la clasificación de llantos de bebé.

### ***1.3. Organización del Reporte***

El reporte de éste proyecto, se encuentra organizado de la siguiente manera:

- **Capítulo 1. Introducción:** Presenta la motivación, objetivos y un breve resumen de los capítulos que conforma el presente proyecto.
- **Capítulo 2. Antecedentes:** Capítulo donde se plantean los precedentes para el desarrollo de este proyecto terminal.
- **Capítulo 3. Descripción Técnica:** A continuación se explica brevemente la constitución del presente proyecto.
- **Capítulo 4. Diseño y Desarrollo de la Aplicación:** En éste capítulo, se describen todos y cada uno de los módulos que conforman todo el proyecto.
- **Capítulo 5. Tecnologías Utilizadas:** Descripción de las tecnologías utilizadas para el desarrollo de la clasificación de los llantos de bebe, así como de la aplicación.
- **Capítulo 6. Resultados.** Ahora se presentan los resultados obtenidos al término del proyecto.

## Capítulo 2

### Antecedentes

---

A pesar de que se comenzaron a realizar análisis del llanto infantil desde hace 186 años aproximadamente, es precisamente a mediados de los 90 donde aparecen dos hechos significativos. El uso de los modelos conexionistas (Redes Neuronales Artificiales RNA) como un nuevo enfoque en la clasificación del llanto infantil así como el primer trabajo de llanto de bebé (*cry analysis[ref]*).

Dado que en la mayoría de los estudios aplicados a los llantos de bebé no utilizan *Wavelets*, se desea aplicar la TWD, utilizando filtros Daubechies a los llantos de bebé para poder extraer las características y después realizar la clasificación de los mismos con la ayuda de un sistema clasificador con motor de inteligencia artificial.

---

#### **2.1. Referencias Internas**

Dentro del campo de filtrado y aplicación de la TWD, en la Universidad se encontraron los siguientes proyectos:

- “Sistema de identificación de personas a través del iris mediante análisis de texturas por filtrado de wavelets” [1].
- “Sistema de reconocimiento del alfabeto dactilológico utilizando procesamiento de imágenes” [2].

Las diferencias existentes entre el presente proyecto y los proyectos anteriores, son:

- El primer proyecto realiza la identificación de imágenes del iris, aplicando TWD pero de Haar, lo cual lo diferencia de éste proyecto, ya que además de no clasificar imágenes, se aplicará TWD empleando los filtros propuestos por Daubechies y no los de Haar.
- El segundo proyecto mencionado, difiere también en el método de clasificación aplicado, ya que al igual que el primero utiliza TWD de Haar, además de realizar procesamiento de imágenes y en el presente proyecto se realizará la clasificación de archivos de audio.

Dentro del campo de la clasificación, en la Universidad se han realizado los siguientes proyectos:

- “Sistema clasificador de documentos de Proyectos Terminales usando el concepto de memoria asociativa”[3].
- "Sistema clasificador de archivos de música usando el concepto de memoria asociativa"[4].

Respecto a las diferencias entre estos proyectos y el presente proyecto, consisten en que:

- El primero realiza la clasificación de documentos de Proyectos Terminales, en cambio, este proyecto realizará la clasificación de llantos de bebé en formato WAV.
- En el segundo proyecto, a pesar de que pudiera parecerse mucho debido a que también trabaja con archivos de audio, los archivos que se clasifican son de música y en el presente proyecto, se clasificarán llantos de bebé que además de durar menos tiempo, no cumplen con ciertas características musicales.

## **2.2. Referencias Externas**

En cuanto a trabajos externos encontramos que dentro del campo de filtrado y aplicación de la TWD se encuentran los siguientes, por mencionar algunos:

- “Identifying and Retrieving of Audio Sequences by using Wavelet Descriptors and Neural Network with User's Assistance”.[5]

- “Audio Classification and Categorization Based on Wavelets and Support Vector Machine”.[6]
- “Fundamentos Teóricos y Prácticos del Análisis de Llanto Infantil”.[7]

Estos trabajos a pesar de que algunos utilizan *Wavelets*, la forma en la que clasifican es por otros métodos. En el presente trabajo se aplicará la TWD para la extracción de características, y con las cuales serán clasificados los llantos del tipo hambre, dolor, incomodidad, baño y normal.

Cabe mencionar que se pueden realizar otras clasificaciones para detectar tipos de enfermedades o patologías presentes o futuras, lo cual queda fuera del alcance de éste proyecto.

### 2.3. Marco Teórico

El rápido avance de la tecnología hace posible la realización de análisis cada vez más rápidos en la obtención de resultados, tal es el caso de la aplicación de *Wavelets* como herramienta para la realización del filtrado de características del llanto de bebé.

Dado que en la mayoría de los estudios aplicados a los llantos de bebé no utilizan *Wavelets*, aplicó la TWD, utilizando filtros Daubechies a los llantos de bebé para poder extraer las características y después realizar la clasificación de los mismos con la ayuda de un sistema clasificador con motor de inteligencia artificial mediante una Red Neuronal Artificial RNA.

### 2.4. Archivos WAV

El formato canónico de archivo WAV [9], es un subconjunto de las especificaciones para el almacenamiento de archivos multimedia. Un archivo RIFF comienza con un encabezado de archivo seguido por una sección por una secuencia de fragmentos de datos mencionados en la tabla 1.

File Offset (byte inicial)	Identificador	Field Size (Longitud Bytes)
0	ChunkID	4
4	ChunkSize	4

8	Format	4
12	Subchunk 1 ID	4
16	AudioFormat	2
20	NumChsnnels	2
22	SampleRate	4
24	ByteRate	4
28	BlockAlign	2
32	BitsPerSample	2
36	Subchunk2ID	4
40	Subchunk2Size	4
44	Data	4

Tabla 1. Formato canónico de archivo WAV

A continuación se describen los segmentos de la Tabla 1

<b>ChunkID:</b>	Contiene los caracteres “RIFF” en ASCII.
<b>ChunkSize:</b>	Almacena el valor del tamaño del archivo WAV en menos de 8 bytes.
<b>Format:</b>	Contiene los caracteres “WAVE”.
<b>SubChunk1ID:</b>	Contiene los caracteres “fmt”.
<b>SubChunk1Size:</b>	Almacena el valor del bloque de datos que comienza aquí y termina antes del bloque de datos
<b>AudioFormat:</b>	Un valor de 1 que indica que no hay compresión de datos
<b>NumChannels:</b>	Mono = 1, Stereo = 2.
<b>SampleRate:</b>	Almacena el valor de frecuencia del muestreo
<b>ByteRate:</b>	Almacena el valor de $SampleRate * NumChannels * BitsPerSample / 8$

<b>BlockAlign:</b>	Almacena el valor de $NumChannels * BitsPerSample / 8$
<b>BitsPerSample:</b>	Indica el número de bits por muestra
<b>Subchunk2ID:</b>	Contiene los caracteres “data”.
<b>Subchunk2Size:</b>	Almacena el valor de $NumSamples * NumChannels * BitsPerSample / 8$
<b>Data:</b>	Almacena las muestras de sonido

## 2.5. Codificación Parson

Dado que los datos PCM contenidos en el archivo *.WAV* dan como resultado una señal con muchos picos y variaciones muy grandes entre dato y dato, se utilizó la codificación Parson con el fin de obtener una señal muy parecida a la original pero más suave, ya que es más sencillo trabajar con señales de este tipo.

## 2.6. Transformada Wavelet Discreta

Un *Wavelet* es una función que oscila con soporte compacto además de ser una señal no estacionaria que ofrece la localización temporal de las características distintivas de otra señal parecida a este. Lo que permite realizar un análisis multiresolución.[10]

Un importante grupo de los *Wavelets* son la Transformada Wavelet Discreta con filtros Daubechies que pueden tener un orden  $n$  dependiendo del número de desvanecimientos que se deseen, tal que  $n$  es un entero positivo que denota el número de coeficientes de filtro que tiene esa *wavelet* [11]

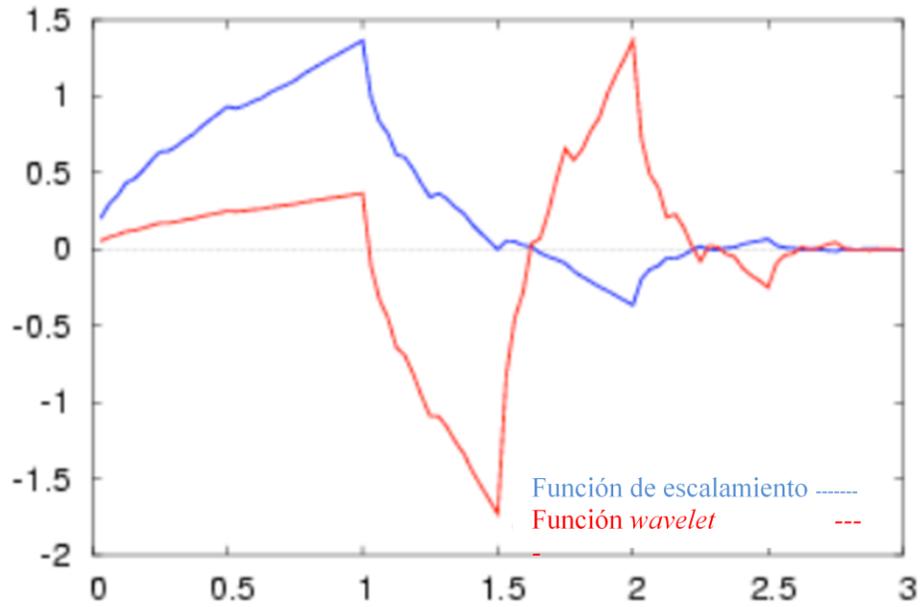


Figura 1. Wavelet de Daubechies 4

## 2.7. Red Neuronal

Una red neuronal artificial (RNA) o ANN por sus siglas en inglés es la analogía de una neurona biológica y se compone de unidades llamadas neuronas las cuales se encargan de procesar la información, es decir, reciben información a través de conexiones con las neuronas de la capa anterior o bien de la primera entrada.

El modelo de una red neuronal más conocido es el modelo de Rosenblat, que se muestra a continuación.

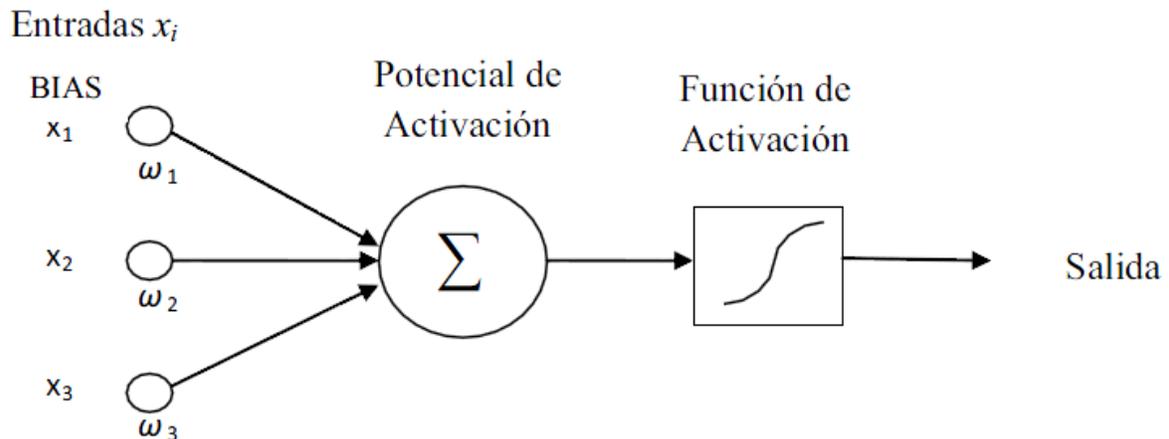


Figura 2. Modelo de Perceptrón propuesto por Rosenblat.

A la conexión de varias de estas unidades elementales (Perceptrones) debidamente organizados en niveles llamados capas se le llama Red Neuronal Simple. Compuesta de una capa de entrada, donde se presentan los datos a la red y una capa de salida que manda la respuesta de la red a una entrada. El resto de las capas reciben el nombre de capas ocultas como se muestra en la Figura 3.

Según el Teorema de Aproximación Universal (TAU), un conjunto de perceptrones es capaz de aproximar funciones matemáticas complicadas sin necesidad de expresarlas explícitamente, es decir, la salida se encuentra dada por la evaluación en cascada de la salida de varios perceptrones que entran a otros y sus salidas son las entradas de otra capa.

Si tomamos en cuenta el TAU junto con la premisa de tener una función matemática que permita clasificar llantos de bebé, entonces, una RNA es capaz de aproximar esta función y otorgar un resultado satisfactorio.

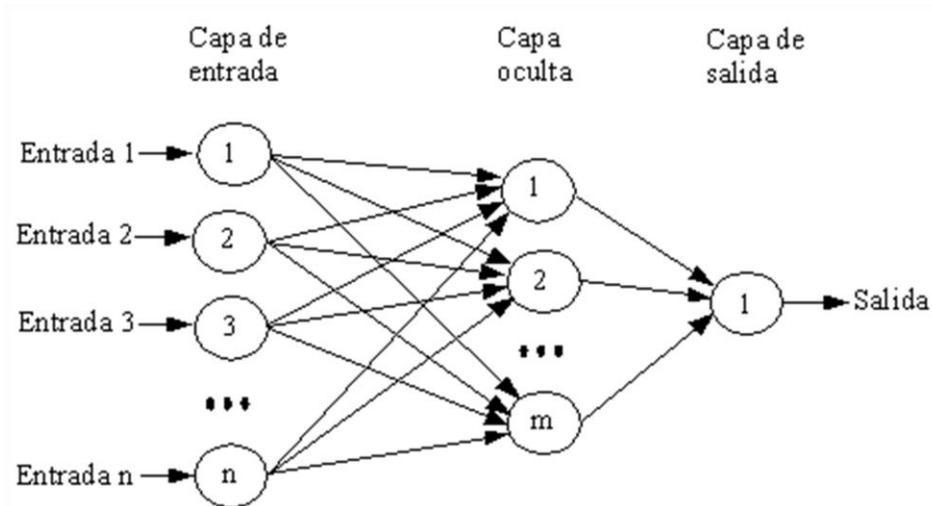


Figura 3. Red Neuronal Simple con 1 capa oculta.

## **Capítulo 3**

### **Descripción Técnica**

---

Este proyecto consta principalmente de cinco módulos (Diagrama 1), los cuales se explican a continuación:

---

### 3.1. Descripción del Sistema

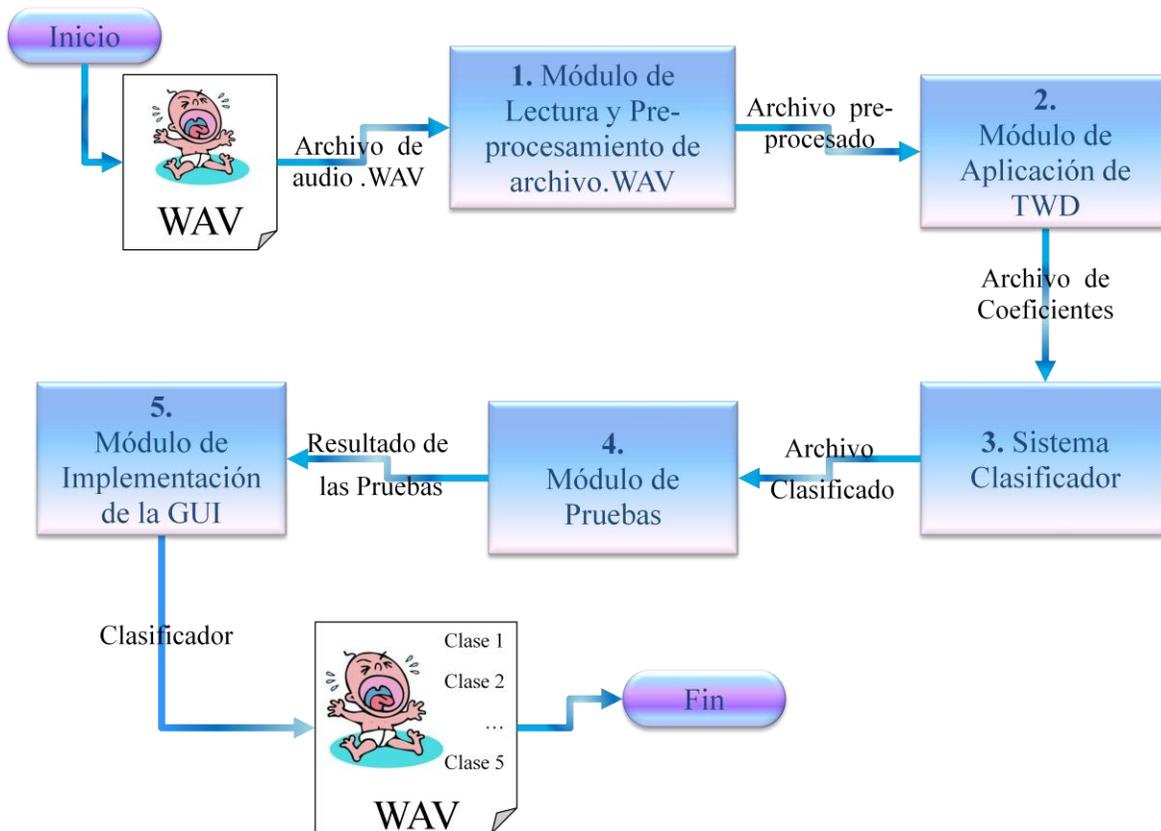


Diagrama 1: Módulos principales para la clasificación de llantos de bebé.

#### 3.1.1. Módulos Principales

1. **Módulo de Lectura de Archivo de Audio .WAV:** En este primer módulo, se tomó el archivo .WAV y se preparó para la extracción de sus características eliminando el ruido y recortándolo a una longitud total de 2.7 segundos aproximadamente. La salida de éste módulo, es otro archivo .WAV preparado para la extracción de sus características distintivas, con la ayuda de la aplicación de la codificación Parson.
2. **Módulo de Aplicación de la TWD:** Se obtuvieron las características distintivas de la función resultante de la codificación Parson en un archivo de texto (.txt), lo que permitió la aplicación de la TWD empleando los filtros Daubechies D-4 para la extracción de las características distintivas de los llantos, las cuales permitieron la clasificación de los archivos de audio.

Los coeficientes obtenidos, se procesaron a través de la distribución de la energía en las distintas bandas de frecuencia proporcionadas por la TWD, las cuales se almacenaron en vectores unidimensionales dentro de un archivo de texto .txt.

3. **Sistema Clasificador:** Los resultados obtenidos en el módulo anterior, se procesaron por un sistema clasificador con enfoque supervisado, llamado RapidMiner, que fue capaz de aprender y distinguir los diferentes llantos, este aprendizaje se realizó por medio de una Red Neuronal Artificial y la entrada de 512 características distintivas de los 72 tipos de llanto.
4. **Módulo de Pruebas:** Para la realización de este módulo, se utilizó la metodología de validación cruzada; que consiste en utilizar el 80% de los datos para entrenamiento y el otro 20% para pruebas y de esta manera, asegurarse de obtener un resultado óptimo y que el sistema pueda ser utilizado con otras muestras de llanto de bebé. Esta validación se realizó de manera automática con la ayuda de Rapid Miner
5. **Módulo de implementación de la GUI:** Por último se extrajo el modelo de la RNA para su implementación en la plataforma JAVA en combinación con JAVA ME para su implementación en dispositivos móviles. El modelo de la RNA, son los archivos de texto que describen a la RNA mediante los pesos sinápticos para cada neurona y de los 72 tipos distintos de llanto, los parámetros de entrenamiento del sistema clasificador implementados en un pequeño sistema de interacción entre la computadora y un dispositivo móvil.

La realización de éste modulo está en dos partes, debido a que JAVA ME no soporta la realización de operaciones con archivos.

## **Capítulo 4**

### **Diseño y Desarrollo**

---

A continuación se explica cual fue el diseño y desarrollo del proyecto en sus 5 módulos principales, desglosándolos en pequeños sub.módulos de procesamiento..

## 4.1. Especificaciones Técnicas

A continuación, se explicará con mayor detalle las entradas y salidas a los módulos que componen el presente proyecto (Diagrama 2):

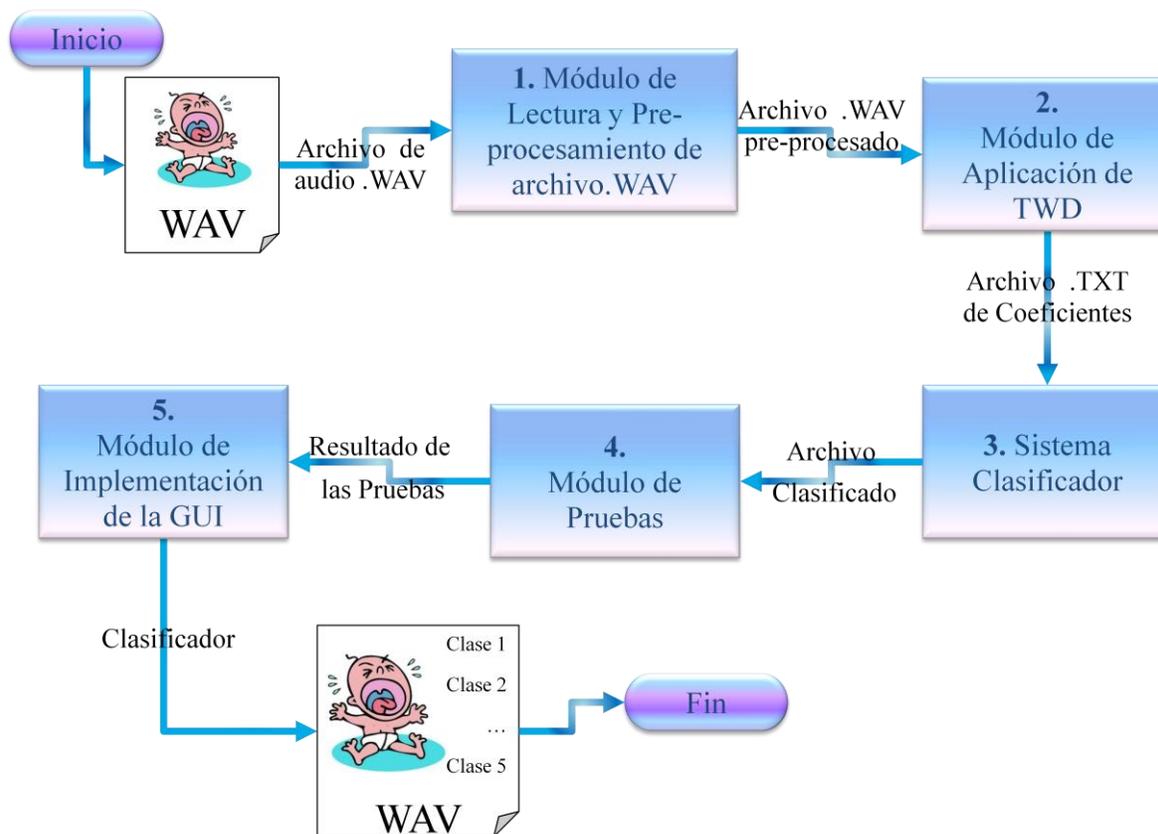


Diagrama 2: Sub-módulos de Análisis y Clasificación

## 1. Módulo de Lectura de Archivo de Audio .WAV:

1.1. Aquí tomamos el archivo .wav original y hacemos otro a partir de este.

1.2. En este sub-módulo, se toma el archivo .WAV y se prepara para la extracción de sus características eliminando el ruido y recortándolo a una longitud total de 2.7 seg, mono y a 6KHz. La salida de éste módulo, será otro archivo .WAV preparado para la extracción de sus características distintivas. Diagrama 3.

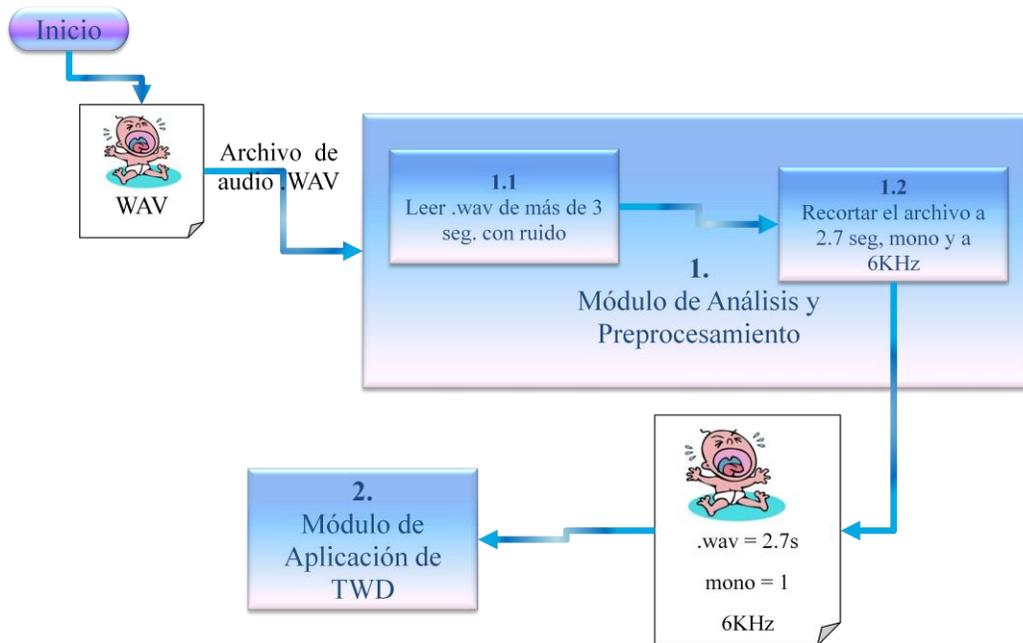


Diagrama 3. Submódulos del Módulo de Análisis.

## 2. Módulo de Aplicación de la TWD:

2.1. Extracción de los datos PCM del archivo .wav procedente del módulo anterior Diagrama 4

2.2. y aplicación de la clasificación Parson con la finalidad extraer las características distintivas, se utilizó 1 cuando los datos son iguales, 0 cuando el siguiente dato es menor y 2 cuando el siguiente dato es mayor. Diagrama 4

2.3. La salida de el submódulo anterior, permite la aplicación de la TWD empleando los filtros Daubechies D-4, las cuales permitirán la clasificación de los archivos de audio. Diagrama 4

Los coeficientes obtenidos, se procesan a través de la distribución de la energía en las distintas bandas de frecuencia proporcionadas por la TWD, las cuales

se almacenan en vectores unidimensionales de tamaño 512 dentro de un archivo de texto .txt. con los 72 llantos. Diagrama 4.

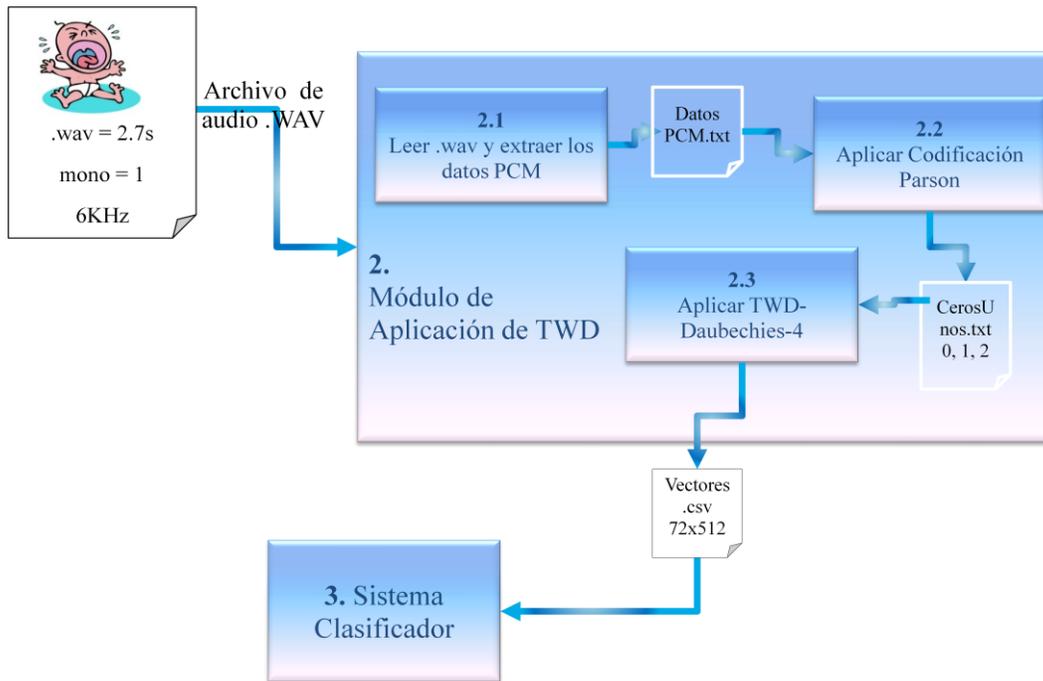


Diagrama 4. Submódulos del Módulo de Aplicación de TWD.

3. **Sistema Clasificador:** Los resultados obtenidos en el módulo anterior, serán procesados por un sistema clasificador con enfoque supervisado, llamado Rapid Miner, que sea capaz de aprender y distinguir los diferentes llantos.

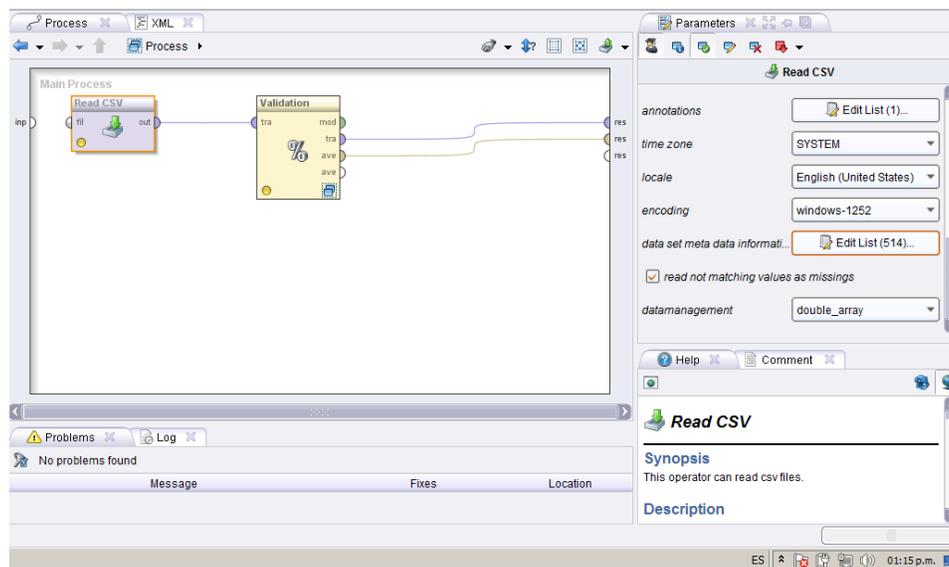


Figura 4. Procesamiento de los datos en RapidMiner

Para la I

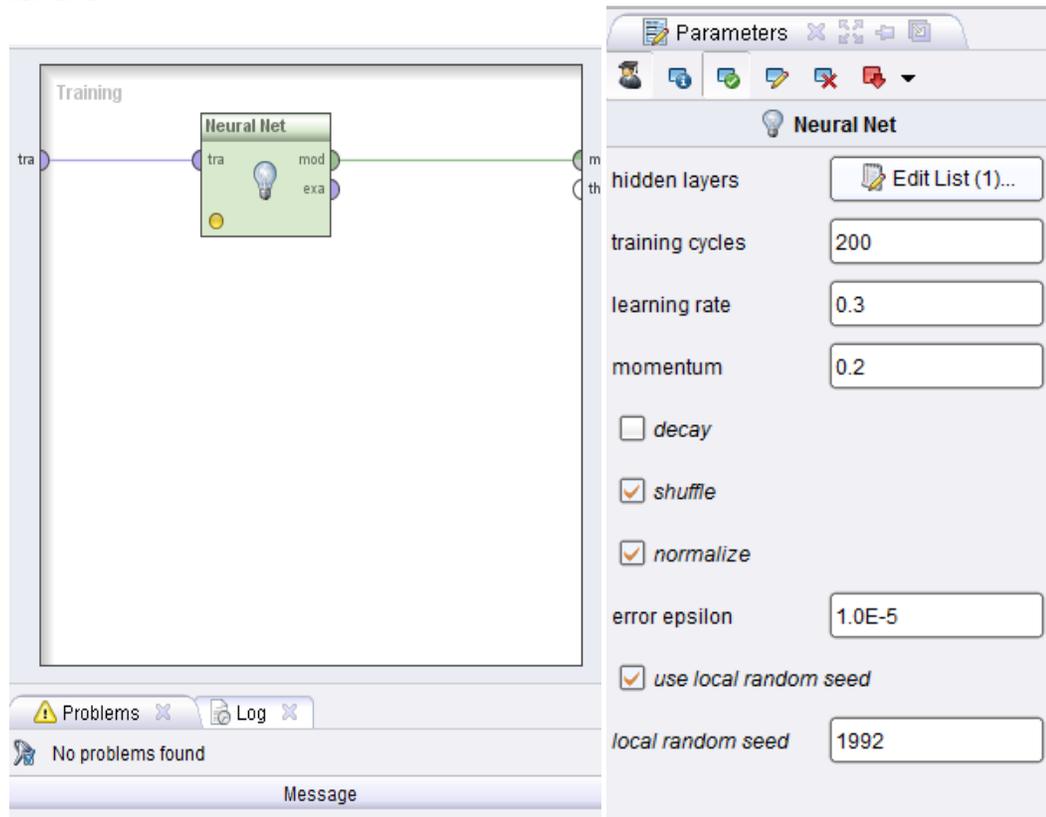


Figura 5. Implementación de la Red Neuronal

- Módulo de Pruebas:** Para la realización de este módulo, se utilizará la metodología de validación cruzada; que consiste en utilizar el 80% de los datos para entrenamiento y el otro 20% para pruebas y de esta manera, asegurarse de obtener un resultado óptimo y que el sistema pueda ser utilizado con otras muestras de llanto de bebé.

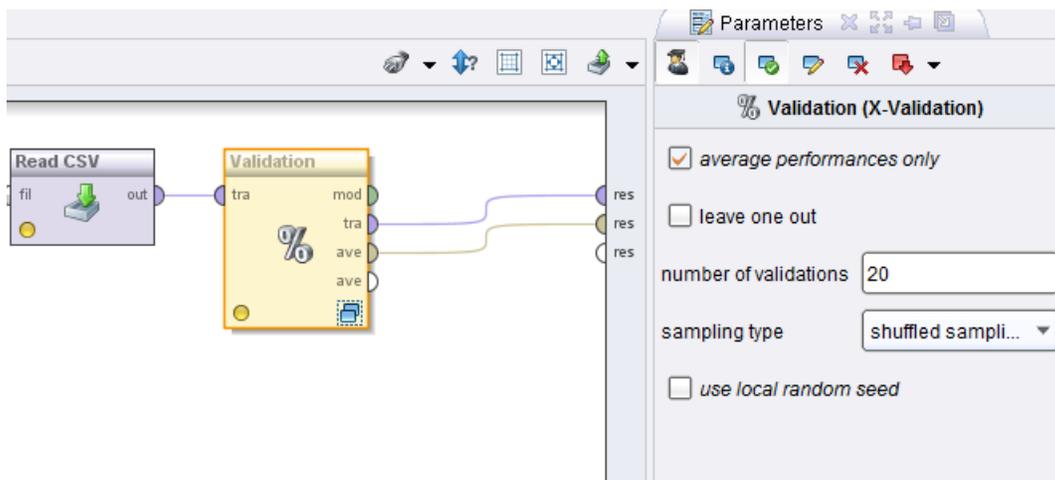


Figura 6. Validación cruzada con 20 validaciones de 72

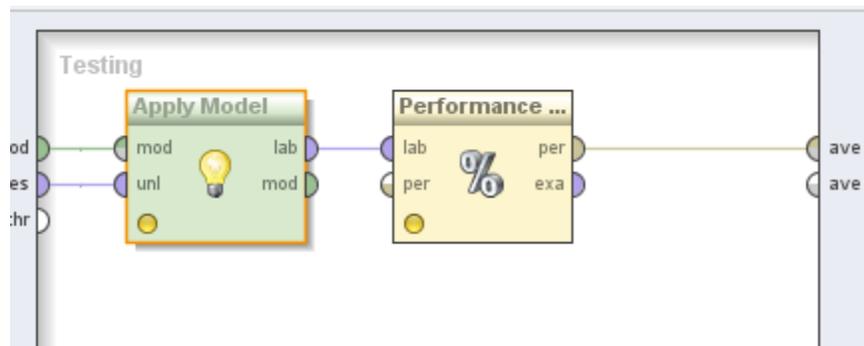


Figura 7. Aplicación de prueba con el mismo modelo y los números de casos seleccionados(20)

5. **Módulo de implementación de la GUI:** Por último se obtendrá el modelo para su implementación en la plataforma JAVA ME para su futura implementación en dispositivos móviles. El modelo, serán los archivos .txt con los parámetros de entrenamiento del sistema clasificador, los cuales son llevados a una representación visual.
  - 5.1. **La Red Neuronal Artificial:** Se compone de 12 neuronas en la capa oculta y 5 en la capa de salida, lo que significa que son 5 clases de llanto como ya lo sabíamos (Dolor, Hambre, Sueño, Incomodidad y Normal)

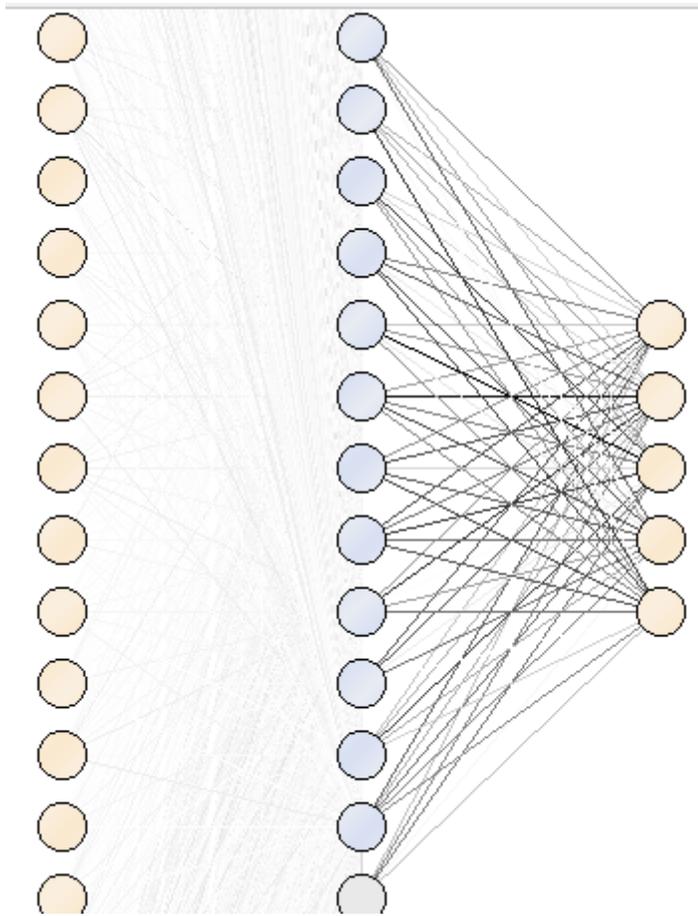


Figura 7. Modelo de la Red Neuronal

## **Capítulo 5**

### **Recursos Utilizados**

---

En este capítulo se presentan los recursos de hardware y software que fueron utilizados para la realización del presente proyecto

---

## **5.1. Hardware**

Para la realización de este proyecto se cuenta con una computadora portátil Dell XPS M1530 con las siguientes características:

- ✓ Intel(R) Core(TM)2 Duo @ 2.20 GHz
- ✓ 3 GB RAM
- ✓ 300 GB HDD
- ✓ Windows Vista Home Premium

Teléfono Celular Sony Ericsson K510a

## **5.2. Software**

Este proyecto fue realizado en su mayoría con distintas tecnologías de java, debido a su gran adaptabilidad, también se utilizaron otras tecnologías de software libre que ayudaron a la obtención de mejores resultados; a continuación se realiza una breve descripción de estos elementos de software.

### **5.2.1. Java**

Java SE está diseñada para permitir el desarrollo de seguros y portátiles, aplicaciones de alto rendimiento para la más amplia gama de plataformas informáticas posibles. Al hacer que las aplicaciones disponibles a través de entornos heterogéneos, las empresas pueden aumentar la productividad del usuario, la comunicación y la colaboración, y reducir drásticamente el coste de propiedad de aplicaciones tanto empresariales como de consumo.[12]

### **5.2.2. Java ME**

Java Platform, Micro Edition (Java ME) proporciona un entorno robusto y flexible para aplicaciones que se ejecutan en los dispositivos móviles y embebidos otros: teléfonos móviles, asistentes personales digitales (PDAs), TV set-top boxes, e impresoras.

Java ME incluye interfaces de usuario flexibles, seguridad robusta, una función de protocolos de red, y soporte para las aplicaciones en red y fuera de línea que se pueden descargar de forma dinámica. Las aplicaciones basadas en Java ME son portables a través de muchos dispositivos, sin embargo, aprovechar las capacidades nativas de cada dispositivo. [13]

### **5.2.3. RapidMiner**

RapidMiner es sin duda el líder mundial de código abierto del sistema para la minería de datos. Está disponible como una aplicación independiente para el análisis de datos y como un motor de minería de datos para la integración en los productos propios. Miles de

aplicaciones de RapidMiner en más de 40 países dan a sus usuarios una ventaja competitiva. [14]

### **5.3. Otros**

Base de datos de llantos de bebé<sup>©</sup>, facilitados por el Dr. Carlos A. Reyes García.

<sup>©</sup>Derechos Reservados <http://ccc.inaoep.mx/directorio/investigadores/kargaxxi.php>

## Capítulo 6

### Resultados y Conclusiones

---

A continuación se presentaran los resultados obtenidos así como las conclusiones a las que se llego con la realización de este proyecto.

---

## Resultados

accuracy: 35.83% +/- 23.88% (mikro: 36.11%)						
	true BANO	true DOLOR	true HAMBRE	true INCOMODO	true NORMAL	class precision
pred. BANO	0	0	0	0	1	0.00%
pred. DOLOR	1	1	1	2	0	20.00%
pred. HAMBRE	2	19	25	4	7	43.86%
pred. INCOMODO	0	2	0	0	2	0.00%
pred. NORMAL	1	0	1	3	0	0.00%
class recall	0.00%	4.55%	92.59%	0.00%	0.00%	

Tabla 1 Tabla de concurrencia

```

PerformanceVector:
accuracy: 35.83% +/- 23.88% (mikro: 36.11%)
ConfusionMatrix:
True:  BANO  DOLOR  HAMBRE  INCOMODO  NORMAL
BANO:  0     0     0     0     1
DOLOR:  1     1     1     2     0
HAMBRE: 2    19    25    4     7
INCOMODO: 0     2     0     0     2
NORMAL: 1     0     1     3     0
classification_error: 64.17% +/- 23.88% (mikro: 63.89%)
ConfusionMatrix:
True:  BANO  DOLOR  HAMBRE  INCOMODO  NORMAL
BANO:  0     0     0     0     1
DOLOR:  1     1     1     2     0
HAMBRE: 2    19    25    4     7
INCOMODO: 0     2     0     0     2
NORMAL: 1     0     1     3     0

```

Tabla 2. Matriz de Confusión

Table / Plot View
  Text View
  Annotations

DANO:	0	0	0	0	1	
DOLOR:	1	1	1	2	0	
HAMBRE:	2	19	25	4	7	
INCOMODO:	0	0	2	0	0	2
NORMAL:	1	0	1	3	0	

absolute\_error: 0.671 +/- 0.194 (mikro: 0.669 +/- 0.414)  
 relative\_error: 67.13% +/- 19.42% (mikro: 66.92% +/- 41.38%)  
 relative\_error\_lenient: 67.13% +/- 19.42% (mikro: 66.92% +/- 41.38%)  
 relative\_error\_strict: 13,196.42% +/- 14,117.22% (mikro: 12,915.67% +/- 23,887.51%)  
 normalized\_absolute\_error: ∞ (mikro: 0.626)  
 root\_mean\_squared\_error: 0.776 +/- 0.139 (mikro: 0.787 +/- 0.000)  
 root\_relative\_squared\_error: ∞ (mikro: 0.670)  
 squared\_error: 0.621 +/- 0.213 (mikro: 0.619 +/- 0.452)  
 correlation: 0.232 +/- 0.318 (mikro: 0.021)  
 squared\_correlation: 0.155 +/- 0.264 (mikro: 0.000)  
 cross-entropy: 3.398 +/- 1.306 (mikro: 3.402)  
 margin: 0.006 +/- 0.005 (mikro: 0.006)

Tabla 3. Errores

## Conclusiones

---

- 1) Se pudo cumplir con el objetivo principal que fue diseñar e implementar un clasificador de llantos de bebé utilizando la TWD a pesar de que no fue como se tenía contemplado en un principio, ya que hubo que hacer uso de un dispositivo móvil y una computadora.
- 2) Dado que los resultados obtenidos no fueron del todo satisfactorios se llegó a la conclusión de que si se hubiesen tenido mayor cantidad de llantos para analizar los resultados hubieran sido mucho mejores.
- 3) Debido a que la tecnología de JAVA ME no soporta el manejo de operaciones con archivos se recomienda que para trabajos futuros se piense en utilizar otra tecnología menos limitada. A pesar de contar con esta tecnología, se implementó una solución que realiza la transferencia de archivo y su procesamiento con la computadora y el dispositivo móvil.

## Bibliografía

---

- [1] Hernández, T. A. y Santiago, S. T., “*Sistema clasificador de documentos de Proyectos Terminales usando el concepto de memoria asociativa*” Proyecto terminal, Universidad Autónoma Metropolitana Unidad Azcapotzalco, Trimestre 09-O.
- [2] Marín D. L., “*Sistema de reconocimiento del alfabeto dactilológico utilizando procesamiento de imágenes*” Proyecto Terminal, Universidad Autónoma Metropolitana Unidad Azcapotzalco, Trimestre 10-P
- [3] Ugalde A. L., “*Sistema de identificación de personas a través del iris mediante análisis de texturas por filtrado de wavelets*” Proyecto Terminal, Universidad Autónoma Metropolitana Unidad Azcapotzalco, Trimestre 09-P.
- [4] Rivera B. S., “*Sistema clasificador de archivos de música usando el concepto de memoria asociativa*” Proyecto Terminal, Universidad Autónoma Metropolitana Unidad Azcapotzalco, Trimestre 09-O.
- [5] Đorđević V., Reljin N, and Reljin I., “*Identifying and Retrieving of Audio Sequences by using Wavelet Descriptors and Neural Network with User's Assistance*”, EUROCON 2005, Serbia and Montenegro, Belgrade, Nov 2005.
- [6] Lin C., Chen S., Member IEEE, Truong T., Fellow IEEE, and Chang Y., Member, IEEE, “*Audio Classification and Categorization Based on Wavelets and Support Vector Machine*”, IEEE TRANSACTIONS ON SPEECH AND AUDIO PROCESSING, VOL. 13, NO. 5, SEP. 2005.
- [7] Reyes G. C., Cano O. S., “*Fundamentos Teóricos y Prácticos del Análisis de Llanto Infantil*”, INAOE, CONACYT, SMIA, México, D.F, 2009, 141 págs.
- [8] Orozco G. J., Reyes G. C., “*Clasificación de Llanto del Bebé Utilizando una Red Neural de Gradiente Conjugado Escalado*”, Instituto Nacional de Astrofísica Óptica y Electrónica, Tonantzintla, Puebla, México.
- [9] Formato de un archivo WAV canónico,  
<https://csrc.stanford.edu/courses/422/projects/WaveFormat/> Revisado por última vez Agosto 2012
-

---

[10] Transformada Wavlet <http://ia.azc.uam.mx/FeriaDelLibro28Feb2011.pdf>, Revisado por última vez Agosto 2012.

[11] Teoría de Wavelets, [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lem/hernandez\\_d\\_m/capitulo2.pdf](http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/hernandez_d_m/capitulo2.pdf) Revisado por última vez Agosto 2012.

[12] Java Platform, Standard Edition (Java SE). <http://www.oracle.com/us/technologies/java/overview/index.html?origref=http://www.oracle.com/us/technologies/java/standard-edition/support/overview/index.html>, Revisado por última vez Agosto 2012.

[13] Java ME and Java Card Technology <http://www.oracle.com/technetwork/java/javame/index.html>, Revisado por última vez Agosto 2012.

[13] RapidMinner <http://rapid-i.com/content/view/181/190/>, Revisado por última vez Agosto 2012.

# Manual de Usuario

---



**Por: Hernández Nieves María Sara**

**México, D.F.**

**Agosto de 2012**

## *Contenido*

Capítulo 1 .....	37
Instalación en Dispositivos Móviles .....	37
1.1. Requerimientos Mínimos para la Instalación .....	38
1.2. Instalación en Dispositivo Móvil .....	38
1.2.1 Opción 1 Vía USB.....	38
1.2.2 Opción 2 Vía Bluetooth.....	38
Capítulo 2.....	41
Características Generales .....	41
2.1. Espacio de Almacenamiento .....	42
2.2. Resolución de la Pantalla .....	42
Capítulo 3.....	43
Comience Aquí .....	43
3.1. Navegando por la aplicación .....	44
Glosario de Términos.....	55

## Capítulo 1

### Instalación en Dispositivos Móviles

---

En el proyecto terminal “Clasificación de llantos de bebé con wavelets”, se realizó la aplicación **KONENE CHOKA** que se encarga de establecer la comunicación entre cualquier dispositivo móvil compatible con tecnologías bluetooth y java.

---

## **1.4. Requerimientos Mínimos para la Instalación**

- ✓ Para el Dispositivo Móvil:
  - Conexión Bluetooth en el dispositivo móvil.
  - Compatibilidad con la máquina virtual de Java en el dispositivo móvil.
  
- ✓ Para la Computadora:
  - Adaptador Bluetooth

## **1.5. Instalación en Dispositivo Móvil**

Existen distintas formas de realizar la instalación de la aplicación en el dispositivo móvil, a continuación se van a explicar 2 de las formas más sencillas y generales.

### **1.2.1 Opción 1 Vía USB**

1. Conectar el dispositivo móvil directamente a la computadora por medio del cable USB, siempre y cuando el dispositivo cuente con dicha opción, de lo contrario, véase opción 2 Vía Bluetooth.
2. Revisar las unidades de almacenamiento y buscar el archivo **ComentarioV1.jar** el cual se encuentra en la carpeta de disco con el nombre de EJECUTABLES. *Nota: El destino puede variar de dispositivo en dispositivo.*
3. Una vez localizado el archivo ejecutable, pasarla al dispositivo móvil.
4. El dispositivo preguntará si se desea realizar la instalación del archivo, seleccionar “Si” o “Aceptar” según sea el caso.
5. Se ejecutará la instalación del archivo de manera automática y estará lista para usarse en la carpeta de **Aplicaciones** con el nombre de **Llantos**.

### **1.2.2 Opción 2 Vía Bluetooth**

1. Emparejar el dispositivo móvil con la computadora con que se va a trabajar.
2. Revisar las unidades de almacenamiento y buscar el archivo **ComentarioV1.jar** el cual se encuentra en la carpeta de disco con el nombre de EJECUTABLES. *Nota: El destino puede variar de dispositivo en dispositivo.*
3. Seleccionar el archivo y seleccionar enviar a Dispositivo Bluetooth (Véase Figura 1, Figura 2, Figura 3)

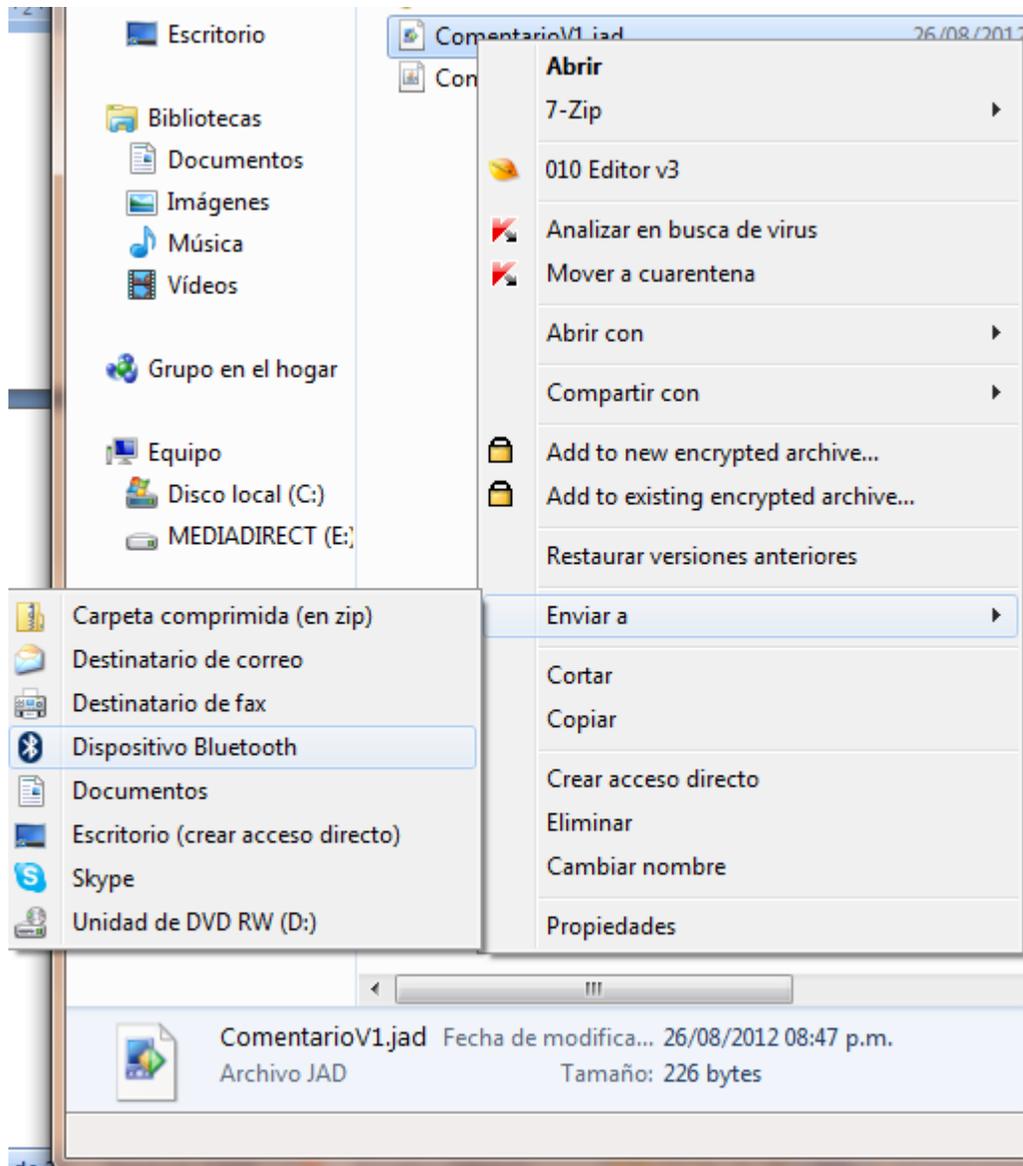


Figura 1. Seleccionar destino de archivo

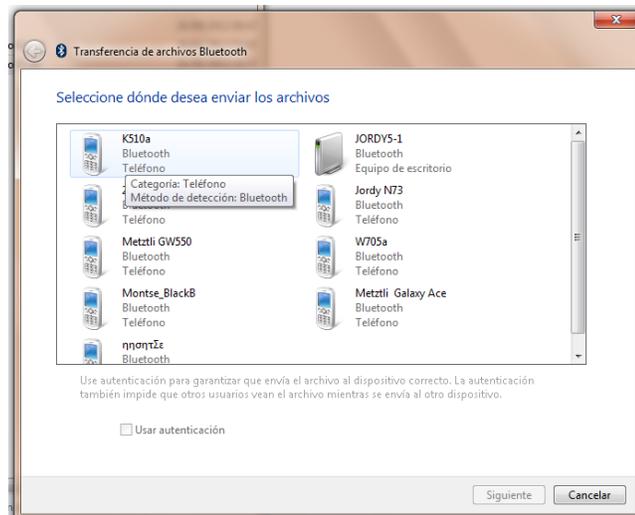


Figura 2. Selección del dispositivo

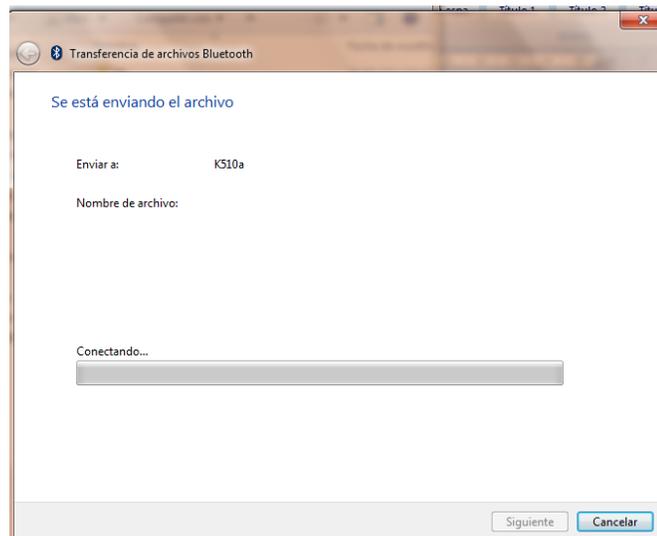


Figura 3. Envío del archivo al dispositivo seleccionado

4. El dispositivo preguntará si se desea realizar la instalación del archivo, seleccionar “Si” o “Aceptar” según sea el caso.
5. Se ejecutará la instalación del archivo de manera automática y estará lista para usarse en la carpeta de **Aplicaciones** con el nombre de **Llantos**.

## **Capítulo 2**

### **Características Generales**

---

A continuación se describirán las características generales de la aplicación en los dispositivos pertinentes.

---

#### **4.1. *Espacio de Almacenamiento***

Al término de la instalación, la aplicación ocupará un espacio total en el dispositivo móvil de 12Kb, debido a que las respuestas o los datos que se envían no son almacenados en el dispositivo, no se requiere de contar con mayor espacio disponible.

#### **4.2. *Resolución de la Pantalla***

En cuanto a la resolución de la pantalla, la aplicación se encuentra diseñada con las configuraciones más básicas para dispositivos móviles con la finalidad de que pueda adecuarse a cualquier dispositivo sin necesidad de realizar modificaciones en la gran mayoría de los dispositivos móviles (CLDC-1.1 y MIDP-1.0).

La resolución y distribución de los objetos en la pantalla, se encuentran diseñados para trabajar a partir de una pantalla de 128 x 160 pixeles en adelante.

La paleta de colores se encuentra en 65 Kb, es decir, manejando 65,000 colores en pantalla o más.

## **Capítulo 3**

### **Comience Aquí**

---

A continuación, aprenderemos el funcionamiento de la aplicación, así como la interacción entre el dispositivo móvil y la computadora para la obtención de los resultados.

---

### 3.1. Navegando por la aplicación

1. Seleccione la aplicación **KoneneChoka** en el menú de las aplicaciones (Véase Figura 4).

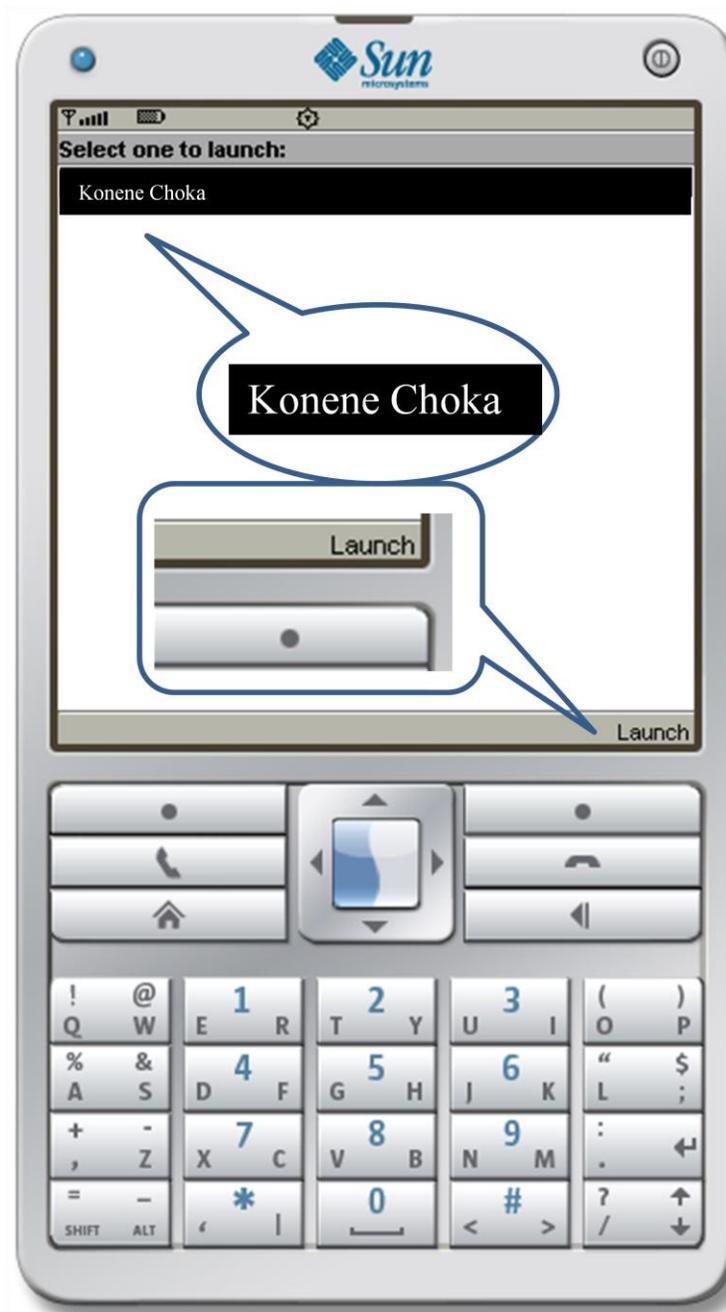


Figura 4. Seleccionando la aplicación

2. Para comenzar con la aplicación, pulsar el botón **Comenzar** (Véase Figura 5).



Figura 5. Comenzar

3. Escribir el llanto que se cree que podría ser y elegir el llanto que se desea analizar. Si se desea calcular, Seleccionar **Calcular** (Véase Figura 6).

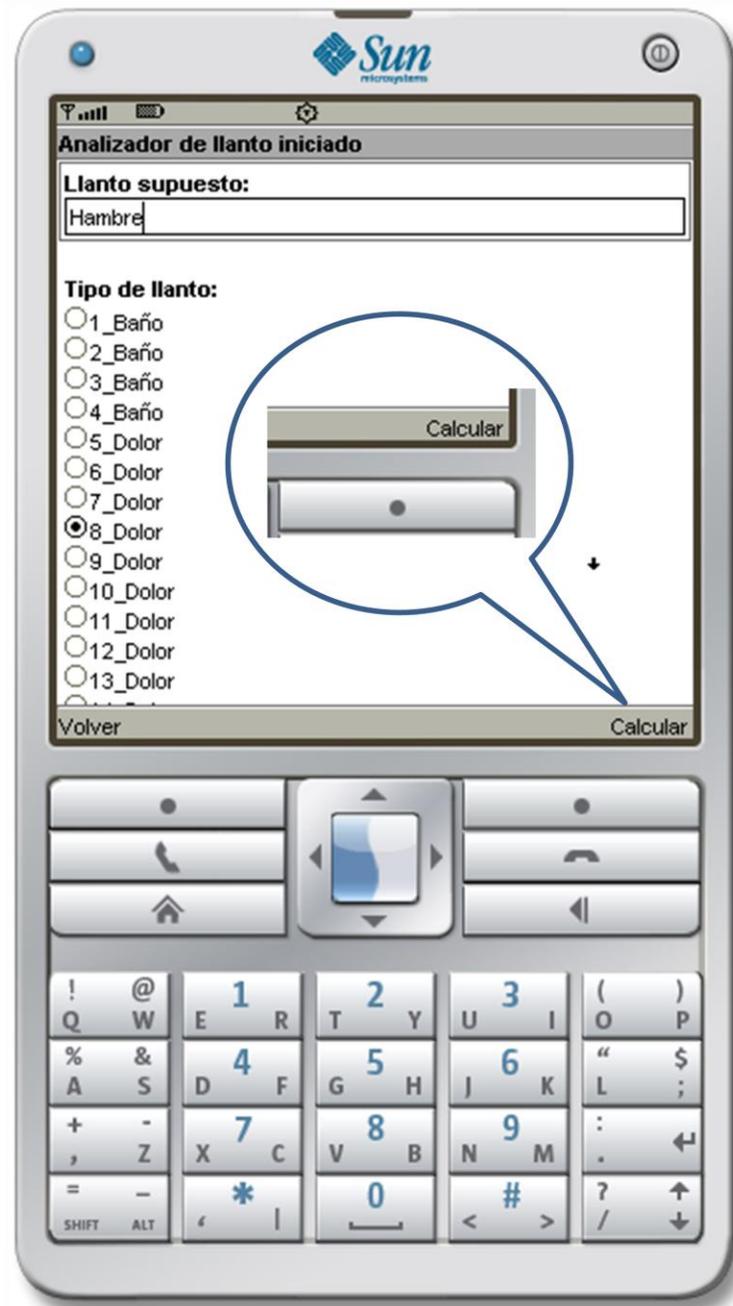


Figura 6. Calcular

4. Ejecutar el sistema que nos ayudará a obtener el resultado del tipo de llanto (Véase Figura 7).

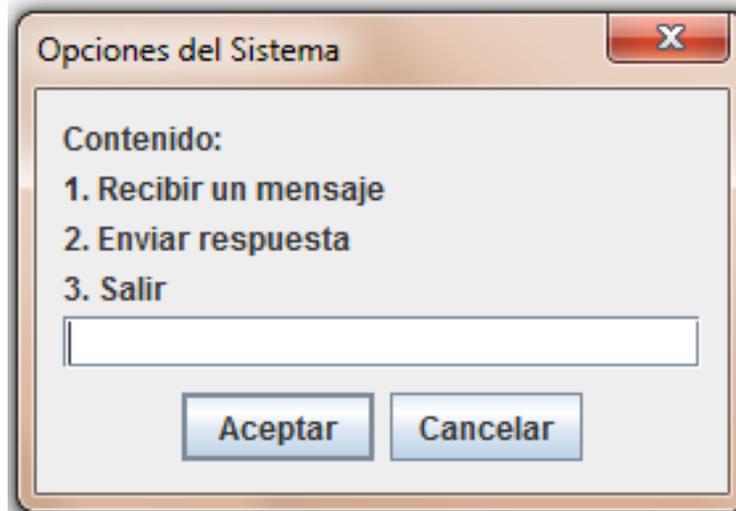


Figura 7. Sistema en ejecución.

5. Como seleccionamos la opción Calcular en el dispositivo móvil, se debe elegir la opción 1 en el sistema para que este reciba el mensaje del llanto a calcular y una supuesta respuesta (Véase Figura 8)..

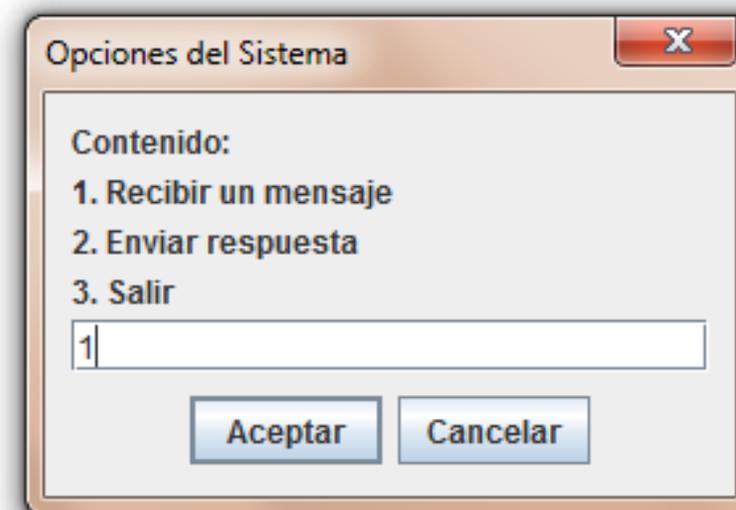


Figura 7. Recibir un mensaje

6. Permitirle al dispositivo móvil que realice el uso del Bluetooth y elegir **Sí**. (Véase Figura 8).

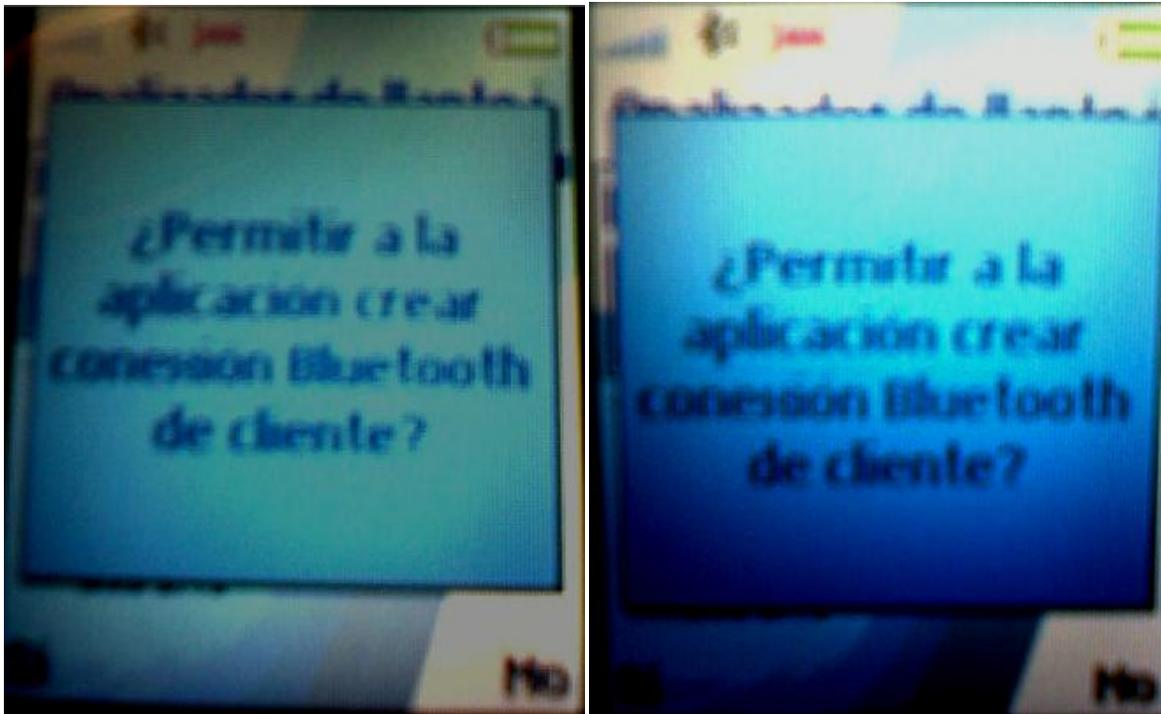


Figura 8. Permitir a la aplicación el uso de bluetooth

7. El sistema avisará si se recibió el mensaje de manera correcta (Véase Figura 9).

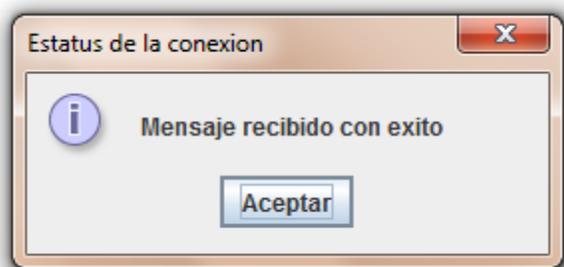


Figura 9 Mensaje del estatus de la conexión

8. Una vez enviados los datos, seleccionar **Volver** en la barra inferior de la aplicación (Véase figura 10).

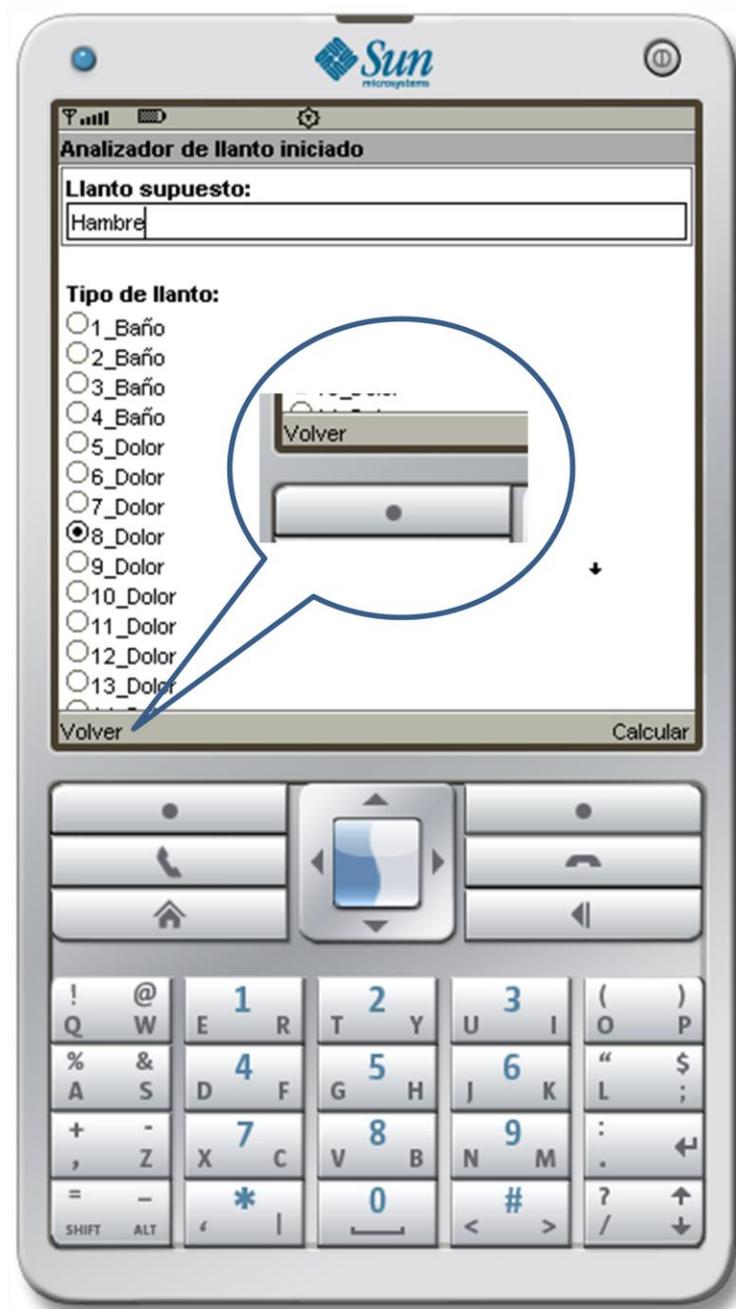


Figura 10. Volver

9. Elegir la opción **Respuesta** en el **menú** (Véase Figura 11) para que el sistema envíe el resultado de a información enviada.



Figura 11. Menu Respuesta

10. Ahora en las opciones del sistema elegir la opción **2** (Vease Figura 12).

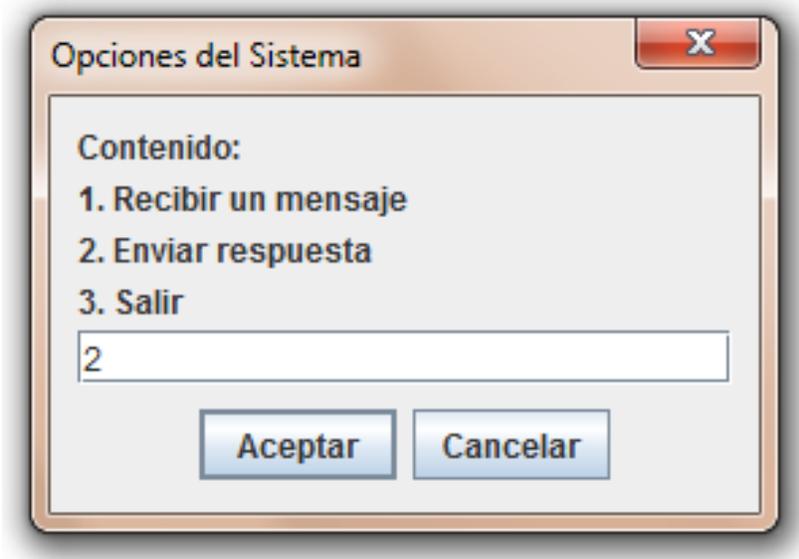


Figura 12. Envío de respuesta

11. El sistema enviará el resultado según los parámetros introducidos en la pantalla del dispositivo móvil (Véase Figura 13).

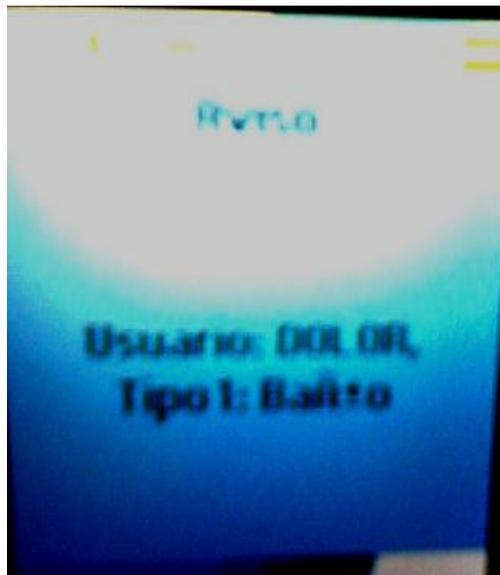


Figura 13 Resultado en la pantalla de Dispositivo Móvil

12. Ahora el sistema informará que ha terminado de ejecutarse (Véase Figura 14).



Figura 14. Mensaje de sistema terminado

13. Si se desea salir del sistema elegir la opción **Salir** en el **Menu** (Véase Figura 15).



Figura15.Menú Salir

14. Al elegir la opción 3 en las opciones del sistema (Véase Figura 16) aparecerá un mensaje de finalización (Véase Figura 17)

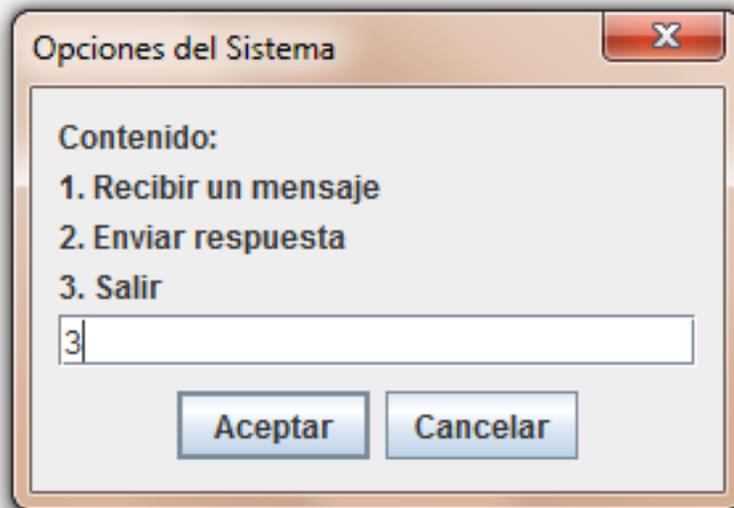


Figura 16. Salir

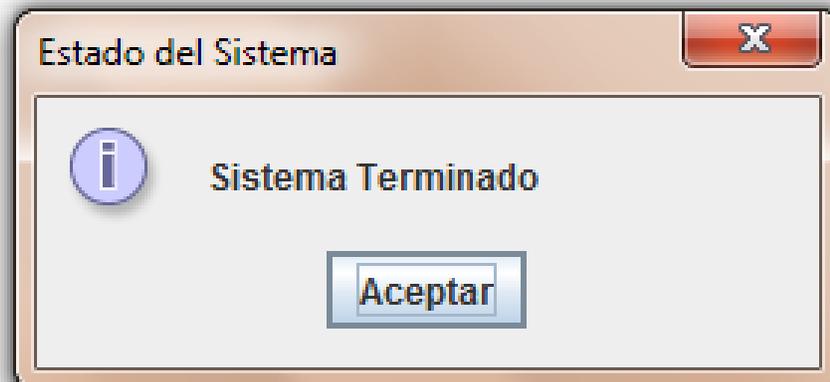


Figura 17. Sistema Terminado

*Nota: En caso de haber en algún momento un error o problema de conexión, se mostrará el siguiente mensaje.(Véase Figura 18)*

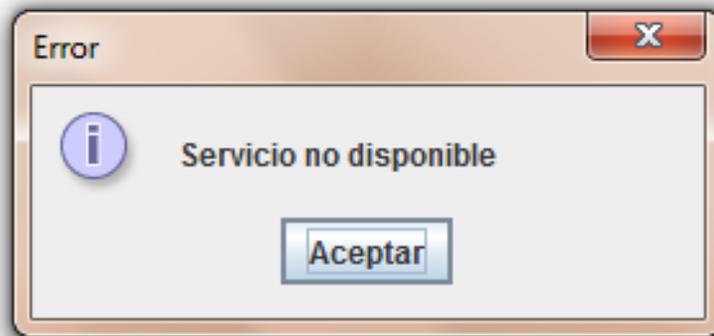


Figura 18. Error

## Glosario de Términos

---

- .jar:** Archivo comprimido que incluye una estructura de directorios con clases java.
- Bluetooth:** Especificación industrial para Redes Inalámbricas de Área Personal (*WPANs*) que posibilita la transmisión de voz y datos entre diferentes dispositivos mediante un enlace por radiofrecuencia.
- CLDC:** *Connected Limited Device Configuration* es una especificación de una framework para aplicaciones Java ME dirigido a dispositivos con recursos limitados
- Emparejar** Proceso de establecimiento de un nuevo vínculo entre dos dispositivos equipados con tecnología Bluetooth durante el que se intercambia una clave.
- Gb:** Unidad de almacenamiento de información digital equivalente a 1024 megabytes (Mb)
- Kb:** Unidad de almacenamiento de información digital equivalente a 1024 bytes.
- Mb:** Unidad de almacenamiento de información digital equivalente a 1024 Kilobytes (Kb)
- MIDP** Versión de J2ME (*Java 2 Micro Edition*) integrada en el hardware de celulares que permite el uso de programas java denominados MIDlets, que pueden ser juegos, aplicaciones u otro tipo de software.
- Píxel** Acrónimo de “*picture element*”, es decir “elemento de imagen”, es la mayor unidad homogénea en color que forma parte de una imagen digital.
- USB:** Acrónimo de *Universal Serial Bus* (bus universal en serie) que describe un puerto que sirve para conectar periféricos a una computadora.