

UNIVERSIDAD CARLOS III DE MADRID

Escuela Politécnica Superior



Ingeniería en Informática

PROYECTO FIN DE CARRERA

Desarrollo de una plataforma para la grabación y análisis de somniloquias: una aproximación usando tecnologías móviles.

Autor: Jorge Pérez Muñoz

Tutor: Israel González Carrasco

Co-director: Alejandro Rodríguez González

Leganes, Octubre de 2012





Título: Desarrollo de una plataforma para la grabación y análisis de somniloquias: una aproximación usando tecnologías móviles.

Autor: Jorge Pérez Muñoz

Tutor: Israel González Carrasco

Co-director: Alejandro Rodríguez González

TRIBUNAL

Presidente:

Vocal:

Secretario:

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día ___ de octubre de 2012 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE



Agradecimientos

Las personas más importantes para mí, que llevan toda la vida apoyándome y queriéndome, mis padres. No podría agradecerlos nunca todo lo que hacéis por mí.

Gracias a mis amigas y amigos, Charo, Nuria, Lara, Guille, Rubén, Alberto, Sara, Lillo, Virginia. Con vosotros la vida sabe mucho mejor.

A ti Fran, por los momentos vividos y por los que nos quedan por vivir.

Gracias por haber compartido tantos años juntos, siempre unidos, disfrutando y riendo. Virginia, he disfrutado y aprendido mucho en la Universidad contigo.

A todos, en especial a mi familia, os quiero.



Autores

Este proyecto de fin de carrera ha sido realizado por los alumnos Virginia Aparicio Paniagua y Jorge Pérez Muñoz de la Universidad Carlos III de Madrid.

En la página siguiente se muestra una tabla con la repartición del trabajo entre los miembros del equipo, así como la dedicación que se ha tenido para cada tarea. La medición de las horas hace referencia al desarrollo del presente documento.



Tarea		Virginia Aparicio	Jorge Pérez	Total
Introducción y Objetivos	Introducción	1	0	1
	Motivación	1	0	1
	Objetivos	1	0	1
	Posibles aplicaciones	1	0	1
	Medios empleados	1	0	1
	Esquema de la memoria	1	0	1
Estado del arte	Estado del arte en análisis de emociones	7	0	7
	Estado del arte en reconocimiento automático del habla	6	0	6
	Estado del arte en la memoria de Virginia Aparicio	0	13	13
Tecnologías utilizadas	Android	0	5	5
	SQLite	0	2	2
	Java	3	3	6
	J2EE y Servlet	4	0	4
	Apache Tomcat	2	0	2
	Mysql	1	0	1
	SoX	1	0	1
Análisis y Diseño	Introducción	1	0	1
	Metodología de desarrollo	0	1	1
	Casos de uso	5	4	9
	Requisitos	11	11	22
	Arquitectura	0	1	1
	Diagrama de clases	4	5	9
Implementación	Módulo Core Functionality	0	16	16
	Módulo Wav	0	15	15
	Módulo Fourier Transform	0	4	4
	Módulo Management	0	2	2
	Módulo Communication	2	0	2
	Módulo Servlet	2	0	2
	Módulo Analysis	15	0	15
	Módulo Transcription	15	0	15
	Módulo Snore	0	4	4
	Módulo Database Management	1	0	1
Presupuesto	Diagrama de tareas	2	0	2
	Desglose de costes	2	0	2
Conclusiones		0	2	2
Líneas Futuras		0	2	2
Glosario	Definiciones	1	1	2
	Acrónimos	1	1	2



Referencias	3	3	6
Anexos	17	17	34
Horas totales	112	112	224



Resumen

La somniliquia es una parasomnia referida al hablar en voz alta durante el sueño. Esta parasomnia ocurre tanto en la fase de sueño REM como en la NREM. La grabación de estas parasomnias puede ser útil para ayudar en el diagnóstico de ciertas patologías psicológicas, dado que pueden reflejar el estado de ansiedad o algunos comportamientos que podrían ser identificados como criterios de diagnóstico psicológico.

En este proyecto desarrollamos una plataforma móvil que únicamente graba los momentos en los que una persona habla mientras está dormida (excluyendo otros ruidos o sonidos) y los analiza para identificar la emoción principal de la voz, además de realizar una transcripción de la conversación que se presenta.

Palabras clave: psicología, somniliquia, Android, análisis emocional, grabación inteligente, transcripción



Abstract

Somniloquy is a parasomnia that refers to talking aloud while sleep. This parasomnia occurs in REM and NREM sleep phase. The recording of these parasomnias could be useful to help in the diagnosis of certain psychological pathologies given that they can reflect anxiety status or some behaviors which could be identified as psychological diagnosis criterions.

In this project we develop a mobile platform which only records sleep-talking (excluding other noises or sounds) and analyze them to identify the main emotion in the voice, allowing to make a transcription of the conversation is presented.

Keywords: psychology, somniloquy, Android, emotions analysis, smart recording, transcription.

Índice general

1. INTRODUCCIÓN Y OBJETIVOS.....	21
1.1 Introducción.....	22
1.2 Motivación.....	23
1.3 Objetivos.....	24
1.4 Posibles aplicaciones del sistema	24
1.5 Medios empleados	25
1.6 Esquema de la memoria.....	27
2. ESTADO DEL ARTE.....	29
2.1 Estado del arte en tratamiento de señales de audio	30
2.1.1 Introducción	30
2.1.2 El sonido y sus cualidades.....	31
2.1.3 Clasificación de los sonidos	32
2.1.4 El sonido en el audio digital.....	33
2.1.5 Procesamiento digital de audio	34
2.1.6 Representación temporal	35
2.1.7 Representación Frecuencial.....	36
2.1.8 Formato WAV	40
2.1.9 Aplicaciones de análisis de audio en Android	41
2.2 Estado del arte en detección de ronquidos y ruidos.....	48
2.2.1 El Ronquido.....	48
2.2.2 Medidas acústicas del ronquido	48
2.2.3 Parámetros de ruido.....	52
2.2.4 Determinación de sonido Sonoro/Sordo	53
2.2.5 Aplicaciones de análisis de ronquidos	54



2.2.6	Aplicaciones de grabación de somniaquias	56
3.	TECNOLOGÍAS UTILIZADAS.....	58
3.1	Selección de la plataforma.....	59
3.2	Java	60
3.2.1	Historia	60
3.2.2	Características	61
3.2.3	Máquina virtual de Java (JVM).....	61
3.2.4	Kit de desarrollo y entorno de ejecución.....	62
3.2.5	Versiones	62
3.2.6	Distribuciones de Java.....	65
3.3	Android	67
3.3.1	Introducción a Android	67
3.3.2	Arquitectura.....	68
3.3.3	Ciclo de vida de las actividades	72
4.	ANÁLISIS Y DISEÑO.....	74
4.1	Introducción.....	75
4.2	Metodología de desarrollo	75
4.3	Casos de Uso	75
4.3.1	Casos de uso de Gestión de Usuarios.....	76
4.3.2	Casos de uso de Gestión de Configuración de los valores de grabación.....	78
4.3.3	Casos de uso de la funcionalidad principal	80
4.3.4	Casos de uso navegación.....	83
4.4	Requisitos	84
4.4.1	Requisitos de Usuario.....	84
4.4.2	Requisitos de Software.....	98
4.5	Arquitectura	121
4.6	Diagrama de clases	122
4.6.1	Módulo Core Functionality	122
4.6.2	Módulo Wav.....	125
4.6.3	Módulo Fourier Transform.....	126
4.6.4	Módulo Management	126
4.6.5	Módulo Communication	127
4.6.6	Módulo Servlet.....	128
4.6.7	Módulo Analysis	129



4.6.8	Módulo Transcription.....	131
4.6.9	Módulo Snore.....	131
4.6.10	Módulo Database Management.....	132
5.	IMPLEMENTACIÓN.....	134
5.1	Dispositivo móvil	135
5.1.1	Módulo de funcionalidad principal	135
5.1.2	Módulo de Gestión	143
5.1.3	Módulo de comunicación	146
5.2	Servidor	147
5.2.1	Módulo Servlet.....	147
5.2.2	Módulo de gestión de la base de datos	147
5.2.3	Módulo de eliminación de ruidos.....	147
5.2.4	Modulo de Transcripción	157
5.2.5	Módulo de análisis de emociones.....	157
6.	PRESUPUESTO.....	159
6.1	Diagrama de tareas	160
6.2	Desglose de costes	162
6.2.1	Personal	162
6.2.2	Material	162
6.1.1	Transporte.....	164
6.1.2	Resumen de costes	164
6.1.3	Totales	164
7.	CONCLUSIONES	167
8.	LÍNEAS FUTURAS	171
9.	GLOSARIO DE TÉRMINOS Y ACRÓNIMOS	175
9.1	Glosario	176
9.2	Acrónimos	177
10.	BIBLIOGRAFÍA Y REFERENCIAS.....	181
11.	ANEXOS.....	186
11.1	Anexo 1.....	187



Índice de ilustraciones

Ilustración 1. Fórmula del NPS	32
Ilustración 2. Representación temporal de una señal acústica.....	35
Ilustración 3. Representación de la Energía y Cruces por Cero de la palabra 'seis'	38
Ilustración 4. Audalyzer	41
Ilustración 5. Diagrama de flujo de Audalyzer.....	42
Ilustración 6. Sound Meter	43
Ilustración 7. SPL Meter.....	44
Ilustración 8. Diagrama de flujo de SPL Meter.....	45
Ilustración 9. Decibel.....	47
Ilustración 10. 2 señales de ronquido y su correspondiente densidad espectral de potencia.....	51
Ilustración 11. Snore No More	54
Ilustración 12. Antisnore	55
Ilustración 13. Sleep Talk Recorder	56
Ilustración 14. Componentes de Java SE 7.....	65
Ilustración 15. Arquitectura Android.....	68
Ilustración 16. Java vs Dalvik.....	71
Ilustración 17. Ciclo de vida de las actividades Android	73
Ilustración 18. Casos de uso de Gestión de Usuarios	76
Ilustración 19. Casos de uso de configuración de valores de grabación	78
Ilustración 20. Casos de uso de la funcionalidad principal	80
Ilustración 21. Casos de uso de navegación	83
Ilustración 22. Arquitectura cliente-servidor.....	121
Ilustración 23. Diagrama de clases de Core Functionality	122
Ilustración 24. Diagrama de clases de Wav.....	125
Ilustración 25. Diagrama de clases de Fourier Transform.....	126



Ilustración 26. Diagrama de clases de Management	126
Ilustración 27. Diagrama de clases de Communication	127
Ilustración 28. Diagrama de clases de Servlet	128
Ilustración 29. Diagrama de clases de Analysis	129
Ilustración 30. Diagrama de clases de Transcription.....	131
Ilustración 31. Diagrama de clases de Snore	131
Ilustración 32. Diagrama de clases de Database Management.....	132
Ilustración 33. Diagrama de estado del MediaRecorder.....	136
Ilustración 34 Esquema del módulo de detección del SPL.	141
Ilustración 34. Representación en tiempo de la señal de un ronquido, su representación en energía y su representación en ZCR.	149
Ilustración 35. Ejemplo 1, análisis de ronquidos. Gráfica de la Energía.....	151
Ilustración 36. Ejemplo 1, análisis de ronquidos. Gráfica del ZCR	152
Ilustración 37. Ejemplo 2, análisis de ronquidos. Gráfica de la energía	154
Ilustración 38. Ejemplo 2, análisis de ronquidos. Gráfica del ZCR	155
Ilustración 28. Tareas del proyecto.....	160
Ilustración 29. Diagrama de Gantt.....	161
Ilustración 41. Acceder a la aplicación.....	187
Ilustración 42. Login.....	188
Ilustración 43. Nuevo Usuario.....	189
Ilustración 44. Detección del SPL en silencio	190
Ilustración 45. Detección del SPL del habla.....	190
Ilustración 46. Enviando SPL al servidor	191
Ilustración 47. Pestaña de Grabación y Reproducción	192
Ilustración 48. Parar la grabación	193
Ilustración 49. Carpetas de usuario.....	193
Ilustración 50. Seleccionar archivo	194
Ilustración 51. Pausar la reproducción	194
Ilustración 52. Pestaña de Configuración	195
Ilustración 53. Borrar Usuario	196
Ilustración 54. Pestaña de Resultados.....	197
Ilustración 56. Cerrar Sesión	198
Ilustración 55. Menú Opciones.....	198
Ilustración 57. Pantalla de Ayuda.....	199



Índice de tablas

Tabla 1. Hardware utilizado	25
Tabla 2. Software utilizado.....	27
Tabla 3. Formato WAV	40
Tabla 4. Características Android	67
Tabla 5. Librerías Android	70
Tabla 6. CU-01	77
Tabla 7. CU-02	77
Tabla 8. CU-03	78
Tabla 9. CU-04	78
Tabla 10. CU-05	79
Tabla 11. CU-06	79
Tabla 12. CU-07	80
Tabla 13. CU-08	81
Tabla 14. CU-09	81
Tabla 15. CU-10	82
Tabla 16. CU-11	82
Tabla 17. CU-12	82
Tabla 18. CU-13	83
Tabla 19. CU-14	83
Tabla 20. CU-15	83
Tabla 21. CU-16	84
Tabla 22. CU-17	84
Tabla 23. RUC-A01	85
Tabla 24. RUC-A02	85
Tabla 25. RUC-A03	85
Tabla 26. RUC-A04	85
Tabla 27. RUC-A05	85
Tabla 28. RUC-A06	86



Tabla 29. RUC-A07.....	86
Tabla 30. RUC-A08.....	86
Tabla 31. RUC-A09.....	86
Tabla 32. RUC-A10.....	86
Tabla 33. RUC-A11.....	86
Tabla 34. RUC-A12.....	87
Tabla 35. RUC-A13.....	87
Tabla 36. RUC-A14.....	87
Tabla 37. RUC-A15.....	87
Tabla 38. RUC-A16.....	87
Tabla 39. RUC-A17.....	87
Tabla 40. RUC-A18.....	88
Tabla 41. RUC-A19.....	88
Tabla 42. RUC-A20.....	88
Tabla 43. RUC-A21.....	88
Tabla 44. RUC-A22.....	88
Tabla 45. RUC-A24.....	88
Tabla 46. RUC-W01.....	89
Tabla 47. RUC-W02.....	89
Tabla 48. RUC-W03.....	89
Tabla 49. RUC-W04.....	89
Tabla 50. RUC-W05.....	89
Tabla 51. RUC-W06.....	90
Tabla 52. RUC-W07.....	90
Tabla 53. RUC-W08.....	90
Tabla 54. RUC-W09.....	90
Tabla 55. RUC-W10.....	90
Tabla 56. RUC-W11.....	90
Tabla 57. RUC-W12.....	91
Tabla 58. RUC-W13.....	91
Tabla 59. RUC-W14.....	91
Tabla 60. RUC-W15.....	91
Tabla 61. RUC-W16.....	91
Tabla 62. RUR-A01.....	92
Tabla 63. RUR-A02.....	92
Tabla 64. RUR-A03.....	92
Tabla 65. RUR-A04.....	92
Tabla 66. RUR-A05.....	92
Tabla 67. RUR-A06.....	93
Tabla 68. RUR-A07.....	93
Tabla 69. RUR-A08.....	93
Tabla 70. RUR-A09.....	93



Tabla 71. RUR-A10.....	93
Tabla 72. RUR-A11.....	94
Tabla 73. RUR-A12.....	94
Tabla 74. RUR-A13.....	94
Tabla 75. RUR-A14.....	94
Tabla 76. RUR-A15.....	94
Tabla 77. RUR-A16.....	95
Tabla 78. RUR-A17.....	95
Tabla 79. RUR-A18.....	95
Tabla 80. RUR-A19.....	95
Tabla 81. RUR-A20.....	95
Tabla 82. RUR-W01.....	96
Tabla 83. RUR-W02.....	96
Tabla 84. RUR-W03.....	96
Tabla 85. RUR-W04.....	96
Tabla 86. RUR-W05.....	96
Tabla 87. RUR-W06.....	97
Tabla 88. RUR-W07.....	97
Tabla 89. RUR-W08.....	97
Tabla 90. RUR-W09.....	97
Tabla 91. RUR-W10.....	97
Tabla 92. RUR-W11.....	98
Tabla 93. RUR-W12.....	98
Tabla 94. RUR-W13.....	98
Tabla 95. RUR-W14.....	98
Tabla 96 RSF-A01.....	99
Tabla 97 RSF-A02.....	99
Tabla 98 RSF-A03.....	99
Tabla 99 RSF-A04.....	100
Tabla 100 RSF-A05.....	100
Tabla 101 RSF- A06.....	100
Tabla 102 RSF- A07.....	100
Tabla 103 RSF- A08.....	100
Tabla 104 RSF- A09.....	101
Tabla 105 RSF- A10.....	101
Tabla 106 RSF- A11.....	101
Tabla 107 RSF- A12.....	101
Tabla 108 RSF-A13.....	101
Tabla 109 RSF-A14.....	102
Tabla 110 RSF-A15.....	102
Tabla 111 RSF-A16.....	102
Tabla 112 RSF-A17.....	102



Tabla 113 RSF-A18.....	102
Tabla 114 RSF-A19.....	103
Tabla 115 RSF-A20.....	103
Tabla 116 RSF-A21.....	103
Tabla 117 RSF-A22.....	103
Tabla 118. RSF-A23.....	103
Tabla 119. RSF-A24.....	104
Tabla 120. RSF-A25.....	104
Tabla 121. RSF-A26.....	104
Tabla 122 RSF-W01.....	104
Tabla 123 RSF-W02.....	105
Tabla 124 RSF-W02.....	105
Tabla 125 RSF-W02.....	105
Tabla 126 RSF-W05.....	105
Tabla 127 RSF-W06.....	105
Tabla 128 RSF-W07.....	106
Tabla 129. RSF-W08.....	106
Tabla 130. RSF-W09.....	106
Tabla 131. RSF-W10.....	106
Tabla 132. RSF-W11.....	106
Tabla 133. RSF-W12.....	107
Tabla 134. RSF-W13.....	107
Tabla 135. RSF-W14.....	107
Tabla 136. RSF-W15.....	107
Tabla 137. RSF-W16.....	107
Tabla 138. RSF-W17.....	108
Tabla 139. RSF-W18.....	108
Tabla 140. RSF-W19.....	108
Tabla 141. RSNF-A01.....	108
Tabla 142 RSNF-A02.....	109
Tabla 143 RSNF-A03.....	109
Tabla 144. RSNF-A04.....	109
Tabla 145. RSNF-A05.....	109
Tabla 146. RSNF-A06.....	109
Tabla 147 RSNF-A07.....	110
Tabla 148 RSNF-A08.....	110
Tabla 149 RSNF-A09.....	110
Tabla 150. RSNF-A10.....	110
Tabla 151. RSNF-A11.....	111
Tabla 152 RSNF-A12.....	111
Tabla 153 RSNF-A13.....	111
Tabla 154 RSNF-A14.....	111



Tabla 155 RSNF-A15	111
Tabla 156 RSNF-A16.....	112
Tabla 157 RSNF-A17	112
Tabla 158 RSNF-A18.....	112
Tabla 159 RSNF-A19.....	112
Tabla 160 RSNF-A20.....	112
Tabla 161. RSNF-A20.....	113
Tabla 162 RSNF-A21.....	113
Tabla 163. RSNF-A22.....	113
Tabla 164. RSNF-A23.....	113
Tabla 165. RSNF-A23.....	114
Tabla 166. RSNF-A24.....	114
Tabla 167. RSNF-A25.....	114
Tabla 168. RSNF-A26.....	114
Tabla 169. RSNF-A27.....	114
Tabla 170. RSNF-A28.....	115
Tabla 171. RSNF-A29.....	115
Tabla 172. RSNF-A30.....	115
Tabla 173 RSNF- A31.....	115
Tabla 174 RSNF- A32.....	116
Tabla 175. RSNF-A33.....	116
Tabla 176. RSNF-A34.....	116
Tabla 177 RSNF-W01.....	116
Tabla 178 RSNF-W02.....	117
Tabla 179 RSNF-W03.....	117
Tabla 180 RSNF-W04.....	117
Tabla 181 RSNF-W05.....	117
Tabla 182 RSNF-W06.....	117
Tabla 183 RSNF-W07.....	118
Tabla 184 RSNF-W08.....	118
Tabla 185 RSNF-W09.....	118
Tabla 186 RSNF-W10.....	119
Tabla 187 RSNF-W11.....	119
Tabla 188. RSNF-W12.....	120
Tabla 189. RSNF-W13.....	120
Tabla 190. RSNF-W14.....	121
Tabla 191. RSNF-W15.....	121
Tabla 192. Tabla de Usuario en SQLite.....	143
Tabla 193. Jerarquía de carpetas.....	145
Tabla 194. Ejemplo 1, análisis de ronquidos.....	150
Tabla 195. Resultados ejemplo 1, análisis de ronquidos.....	151
Tabla 196. Ejemplo 2, análisis de ronquidos.....	153



Tabla 197. Resultados Ejemplo 2, análisis de ronquidos	154
Tabla 216. Coste de los equipos	162
Tabla 217. Coste de las licencias	163
Tabla 218. Resumen de los costes	164
Tabla 219. Coste total.....	164
Tabla 218. Glosario	177
Tabla 219. Acrónimos	179



Capítulo 1

Introducción y objetivos

“La esperanza es el sueño del hombre despierto.”

Aristóteles (384 AC-322 AC)

1.1 Introducción

En la actualidad, la sociedad lleva un ritmo de vida frenético, existiendo una necesidad de realizar un gran número de actividades en el menor tiempo posible, llevando a las personas a situaciones estresantes y pudiendo causar problemas de salud y psicológicos. Además, en los tiempos de crisis que corren, con las grandes dificultades financieras y laborales que sufre la población, el número de individuos con ansiedad o problemas emocionales se está viendo elevado.

Existen diversos estudios, de los que hablaremos a continuación, en los que se muestra la relación que hay entre problemas psicológicos con trastornos del sueño, como la somniloquia.

Por definición, la somniloquia [1] consiste en la emisión de palabras durante el sueño, sin que exista noción subjetiva y simultánea por parte del sujeto. Suele ser un proceso poco frecuente y de corta duración, que puede aparecer en cualquier persona sin importar la edad y el sexo aunque predomina en la población infantil y la adolescente y afecta mayoritariamente a hombres.

La somniloquia se ha estudiado poco en relación con otros trastornos del sueño. Sin embargo, se plantea que factores de maduración y desarrollo son de gran importancia dentro de las causas. Se han observado antecedentes hereditarios en algunas personas y puede verse asociado a la enuresis (orinarse dormido) y el sonambulismo (caminar y, también, realizar diferentes actos mientras se duerme). Puede estar condicionado al consumo de sustancias psicoactivas, la fiebre, la sobreexcitación, el estrés emocional y trastornos afectivos.

El hablar en sueños puede aparecer tanto en la fase REM como en la NREM, y hay varios estudios, como los que realizaron Arkin [2] y Rechtschaffen [3], en los que se explica que existe relación entre la somniloquia producida en la fase REM con momentos de carácter emocional. Según estos autores, los sueños que ocurren en la fase NREM tienden a relacionarse con acontecimientos recientes, sin contenido afectivo, siendo habitualmente situaciones reales de la vida diaria.

Por otra parte, Andriani [4] en 1892, también declaró que el contenido del habla durante un sueño es normalmente emocional, y revela un deseo insatisfecho, un placer inesperado, un lamento, o más frecuentemente un estado de miedo, ansiedad, angustia o terror.

El sistema propuesto en este proyecto pretende principalmente ayudar a profesionales del área de la psicología como herramienta de ayuda para diagnosticar ciertas patologías, como la ansiedad o el estrés.

1.2 Motivación

Como se ha descrito en el punto [anterior](#), la somniloquia se relaciona de una manera muy estrecha con problemas emocionales de la persona en la que se presenta. Además, el habla en sí misma también contiene información extra-lingüística sobre características físicas, estados fisiológicos y emocionales. Darwing señaló [5] que las emociones y su expresión son innatas e instintivas y que la parte no verbal de la voz puede ser un medio de expresión de emociones. Esto se debe a que los diferentes estados emocionales de un hablante producen cambios fisiológicos en el aparato fonador y se reflejan en características de la señal de la voz.

Debido a la gran demanda de personas que buscan ayuda en manos de un psicólogo para intentar mitigar y solucionar sus problemas psicológicos y emocionales, es razonable tener a disposición herramientas o medios que faciliten la labor de estos profesionales. Sin embargo, actualmente este mercado no posee un gran desarrollo por lo que es necesaria la creación de recursos.

Este proyecto se centra en la creación de una plataforma de grabación de personas somníflocuas y el posterior análisis de los audios obtenidos para intentar clasificarlos por emociones primarias.

Se propone un modelo de grabación inteligente de las somniloquias, obteniendo únicamente los fragmentos de audio en los que se detecten palabras o frases de una determinada persona durante el proceso del sueño. Esto aporta mejoras frente a la grabación tradicional, ya que permite ahorrar en tamaño de almacenamiento, debido a que se descartan los periodos de silencio así como los ruidos que no corresponden a palabras. Esto hace que se reduzca el tiempo de duración por lo que el usuario que desee oír sus grabaciones perderá menos tiempo en este proceso.

Tal y como se comentó anteriormente, la voz contiene información emocional que se puede extraer mediante un análisis de la señal que la representa. En este proyecto se realizará un módulo encargado de la predicción de la emoción predominante en el audio.

Por último, con el objetivo de aportar una mayor cantidad de información a los profesionales de la psicología, la aplicación proporcionará un método de transcripción a

texto del archivo de audio, de forma que éstos puedan realizar una interpretación leída de las somniaquias.

1.3 Objetivos

El objetivo de este proyecto es crear un sistema que permita grabar el habla de las personas mientras duermen de un modo eficiente, para posteriormente procesar acústicamente los archivos de audios y predecir, en función de unas reglas previamente definidas, la emoción predominante del archivo y generar una transcripción del mismo.

En base a este objetivo principal, se proponen los siguientes objetivos parciales:

- Crear una aplicación en una plataforma móvil encargada de la grabación de somniaquias.
- Grabar audios de una manera optimizada evitando la generación de archivos que contengan grandes silencios y que por lo tanto sean muy pesados.
- Eliminación de ruidos en los archivos grabados.
- Manejar un sistema de comunicación entre la aplicación de la plataforma móvil y un servidor de aplicaciones.
- Detección de la emoción predominante entre las cuatro emociones primarias de enfado, alegría, tristeza y miedo.
- Transcripción a texto del audio interpretado.
- Presentar un sistema fácil e intuitivo para el usuario final.

Para lograr estos objetivos es necesario estudiar los siguientes aspectos:

- Adquirir conocimientos sobre análisis acústico de una señal.
- Adquirir conocimientos sobre la programación en Android.
- Adquirir conocimientos en programación J2EE.

1.4 Posibles aplicaciones del sistema

Al tratarse de un sistema con tanta funcionalidad, las aplicaciones que puede tener son muy variadas, entre las que podemos destacar:

- El sistema podría utilizarse para grabar a personas mientras duermen para saber si roncan o si padecen apneas, o ayudar a prevenir cualquier otro problema que pueda detectarse con audios.

- El sistema podría aplicarse para realizar grabaciones en exteriores, ya que al tratarse de ambientes con ruidos, se les facilitarían la eliminación de sonidos innecesarios.
- La aplicación podría interesar a entidades con call centers automáticos, para detectar la frustración del locutor por medio de la emoción del audio y por tanto la necesidad de remitirlo a un operador humano, o simplemente para detectar la satisfacción que puede estar produciendo la calidad del servicio.
- Atención de llamadas en servicios de emergencia con el fin de cuantificar la gravedad del asunto de la llamada.
- El sistema puede ayudar en investigaciones de psicología, psiquiatría, o neurología donde el reconocimiento de emociones puede mejorar la calidad de los estudios obteniendo mayor fiabilidad en las medidas y mayor velocidad en tareas manuales de procesamiento de datos sobre el comportamiento emocional.

1.5 Medios empleados

A lo largo del proyecto ha sido necesario emplear una serie de dispositivos físicos y aplicaciones informáticas con el fin de recopilar información, realizar tareas de análisis, diseño e implementación de código, llevar a cabo simulaciones y pruebas, documentar los distintos procesos, etc. A continuación se muestran los distintos medios utilizados de manera detallada y clasificados en función de su tipo.

Hardware:

Hardware	Características	
	Ordenador portátil Toshiba Satellite L655	Intel Core i3-370M, 2.40 GHz, 4GB DDR3, 500GB
	Ordenador portátil Toshiba Satellite A300	Intel Core 2 Duo, 2.00GHz, 4GB DDR2, 320GB
	Teléfono móvil Samsung Galaxy i9000	Pantalla WVGA Super AMOLED de 4 pulgadas, procesador de 1GHz y el sistema operativo Android 2.1.
	Micrófono ordenador	Grabación

Tabla 1. Hardware utilizado

Software:

Software		Características
	Windows 7 Professional	Sistema Operativo
	Windows 7 Home Premium	Sistema Operativo
	Microsoft Office Word 2007	Editor de textos
	Microsoft Excel 2007	Hoja de cálculo
	Microsoft Project Professional 2007	Planificación
	Microsoft Office Powerpoint 2007	Presentación y diagramas
	SoX	Procesamientos de audio
	Eclipse Helios	Programación
	Eclipse Índigo IDE	Programación
	Android SDK	Programación
	MySQL 5.5	Base de datos

	SQLite	Base de datos
	Apache Tomcat	Servidor web
	Java Platform (JDK) 7	Plataforma de programación

Tabla 2. Software utilizado

1.6 Esquema de la memoria

Esta sección está destinada a ofrecer una breve explicación de cada uno de los capítulos de los que consta el presente documento.

- **Estado del arte.** En este apartado hablaremos de las distintas tecnologías y estudios relacionados con el sistema propuesto.
- **Tecnologías utilizadas.** Este capítulo está dedicado a explicar las distintas herramientas y tecnologías empleadas en la realización del sistema.
- **Análisis y Diseño.** El objetivo de esta sección es la especificación de la funcionalidad a través de los casos de uso y los requisitos. Además, se explicará el diseño del sistema mediante su arquitectura y su diagrama de clases.
- **Implementación.** Este apartado está dedicado a explicar cómo se ha realizado el sistema de una manera detallada. Se especifican las decisiones que se han tomado y las dificultades encontradas en el proceso.
- **Presupuesto.** En este apartado se hace un análisis de los costes relacionados con las distintas fases de desarrollo en función del tiempo y recursos invertidos en cada una de ellas.
- **Conclusiones.** En este capítulo se comentan las propiedades del sistema y se realiza una valoración final del sistema.
- **Líneas futuras.** Para finalizar se tratarán aquellas funcionalidades que pueden ser incluidas en el futuro para convertir nuestro sistema en una herramienta más completa.





Capítulo 2

Estado del arte

*“Antes de iniciar la labor de cambiar el mundo,
da tres vueltas por tu propia casa.”*

Proverbio chino

2.1 Estado del arte en tratamiento de señales de audio

2.1.1 Introducción

El procesado de señales, también llamado tratamiento o procesamiento de señales, es la disciplina que desarrolla y estudia las técnicas de tratamiento, el análisis y la clasificación de las señales. Se basa en los resultados de la teoría de la información, de la estadística y la matemática aplicada.

Una señal es un flujo de información proveniente de una fuente, la cual puede tener una naturaleza diversa: mecánica, óptica, magnética, eléctrica, acústica, etc. Por lo general, para poder ser procesadas, las señales se transforman en señales eléctricas mediante transductores.

Podemos definir una señal como la variación en el tiempo o el espacio de una magnitud física o de otra naturaleza. Para su análisis, las señales habitualmente son modeladas como funciones matemáticas.

La clasificación más básica de las señales se produce según su representación respecto a las variables de las que dependen, como el tiempo o el espacio:

- Señal analógica es aquella que representa una magnitud de manera continua. Puede provenir de captadores, como por ejemplo, los micrófonos. Un micrófono es un transductor electro-acústico que capta sonidos y los traslada a señales eléctricas. Otros ejemplos pueden ser un termómetro, para captar temperaturas, una sonda barométrica, que capta presiones, un velocímetro, etc.
- Señal digital es aquella que toma valores sólo para una cantidad discreta de puntos, y además sus valores son únicamente discretos. Se pueden considerar como ejemplos de señales digitales: un programa de ordenador, el contenido de un CD. Aunque también podría ser la información recibida de un semáforo, el código Morse, etc.

De manera parecida a la señal digital, una señal discreta sólo tiene valores en una cantidad discreta de puntos. La diferencia está en que estos valores pueden tomar cualquier valor, es decir, no están cuantificados. Estas señales provienen normalmente de conversores analógico-digitales, o lo que es lo mismo, de la discretización de señales

continuas. Cuando una señal discreta es cuantificada mediante un cuantificador se transforma en una señal digital.

2.1.2 El sonido y sus cualidades

Desde un punto de vista físico el sonido es una vibración que se propaga en un medio elástico. Para que se produzca sonido se requiere la existencia de un cuerpo vibrante, denominado foco (cuerda tensa, varilla, una lengüeta) y de un medio elástico que transmita esas vibraciones, que se propagan por él constituyendo lo que se denomina onda sonora.

El sonido es un fenómeno vibratorio transmitido en forma de ondas. Como todo movimiento ondulatorio, el sonido puede representarse como una suma de curvas sinusoides, que se pueden caracterizar por las mismas magnitudes que a cualquier onda de frecuencia bien definida:

- Longitud de onda (λ)
- Frecuencia (f) o inversa del período (T)
- Amplitud (A)

Si nos fijamos en la propia percepción auditiva humana, podemos distinguir fácilmente los sonidos y los ruidos. Si se obtienen gráficas de registro de las vibraciones de las ondas del sonido se observa que, en general, los sonidos musicales poseen ondas casi sinusoidales, aunque alteradas a veces apreciablemente por la presencia de sus armónicos. Los restantes sonidos armónicos conservan todavía una total periodicidad aunque su gráfica se aleje notablemente de una senoide, por estar compuestos de varios grupos de ondas de frecuencias fundamentales distintas, acompañadas de algunos de sus armónicos. Por último los ruidos presentan, de ordinario, gráficas carentes de periodicidad y es precisamente esta peculiaridad lo que produce que la sensación cerebral resulte desagradable o molesta.

A continuación se describen las características más importantes que caracterizan al sonido:

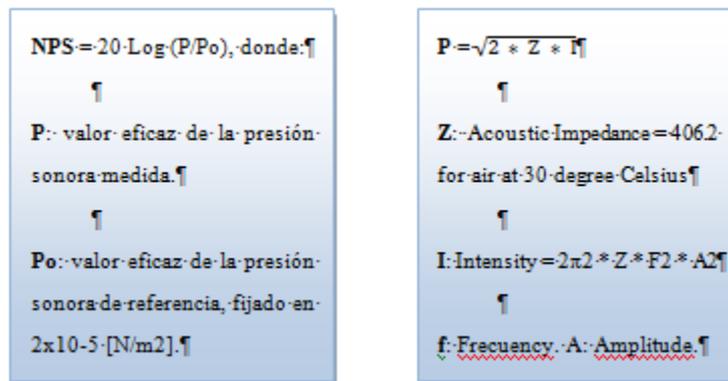
La amplitud define la intensidad de la onda. No hay que confundir amplitud con volumen o potencia acústica, que es la cantidad de energía radiada en forma de ondas por unidad de tiempo por una fuente determinada.

La frecuencia nos da una medida del número de ondas que podemos encontrar en un determinado tiempo. Si se producen más ondas en una misma unidad de tiempo, tendremos un sonido de mayor frecuencia, y por lo tanto un tono más agudo; los tonos más graves corresponden a frecuencias menores.

La intensidad es la cantidad de energía acústica que contiene un sonido. Para el desarrollo de nuestro proyecto conviene centrarse en la intensidad auditiva, la cual se fundamenta en la ley psicofísica de Weber-Fechner [6]. Esta ley establece una relación logarítmica entre la intensidad física del sonido que es captado y la intensidad física mínima audible por el oído humano.

La presión sonora o acústica es producto de la propia propagación del sonido.

El Nivel de Presión Sonora (NPS) [7] se mide en decibelios SPL, y se abrevia "dB SPL", por las siglas en inglés de Sound Pressure Level. En términos matemáticos, el Nivel de Presión Sonora (NPS) es la medición logarítmica del valor promedio de la presión sonora, respecto a un nivel de referencia.



$NPS = 20 \cdot \text{Log}(P/P_0), \text{ donde:}$	$P = \sqrt{2 \cdot Z \cdot I}$
P : valor eficaz de la presión sonora medida.	Z : Acoustic Impedance = 406.2 for air at 30 degree Celsius
P_0 : valor eficaz de la presión sonora de referencia, fijado en 2×10^{-5} [N/m ²].	I : Intensity = $2\pi^2 \cdot Z \cdot f^2 \cdot A^2$
	f : Frecuency. A : Amplitude.

Ilustración 1. Fórmula del NPS

2.1.3 Clasificación de los sonidos

Los sonidos se pueden clasificar según los siguientes criterios [8]:

1. La acción de las cuerdas vocales:
 - a. Sonoros, si las cuerdas vocales se aproximan y comienzan a vibrar.
 - b. Sordos, si las cuerdas vocales se acercan entre sí pero sin llegar a vibrar.
2. La acción del velo del paladar:
 - a. Orales o Bucales, si se encuentra adherido a la pared de la laringe. El aire sale solo entonces por la cavidad bucal. (/s/, /b/, /p/).
 - b. Nasales, si parte del aire sale a través de la cavidad nasal. (/n/, /m/, /g/).
3. El modo de articulación, el cual puede definirse como la posición de los órganos articulatorios en referencia a su apertura o cierre, corresponde a la siguiente clasificación:

- a. Vocales, si está en posición abierta. En general los sonidos vocálicos se caracterizan por su considerable amplitud.
 - b. Consonantes
 - c. Oclusivas, cuando hay un cierre completo de los órganos articulatorios (/p/, /t/, /k/, /b/, /d/, /g/, /m/, /n/)
 - d. Fricativas, cuando el sonido se forma a partir del estrechamiento de dos órganos articulatorios, en diferentes posiciones del tracto vocal sin que éstos nunca lleguen a juntarse. (/f /, /s/, /x/)
 - e. Africadas, cuando tras un cierre completo de los órganos articulatorios sigue una pequeña apertura por donde se deslizará el aire contenido en un primer momento. (/c/, /j/).
 - f. Nasales, cuando está abierta la cavidad nasal. (/n/, /m/, /g/).
 - g. Líquidas, que pueden ser a su vez, Laterales (/l/), si el aire sale por un lado o ambos de la cavidad vocal, y Vibrantes (/y/, /w/, /r/) si se produce una vibración del ápice de la lengua.
4. El lugar de la articulación, corresponde a la siguiente división:
- a. Bilabiales, si se juntan los labios (/p/, /b/, /m/).
 - b. Labiodentales, si se sitúan los dientes superiores en el labio inferior (/f /, /v/).
 - c. Interdentales, si se coloca la lengua entre los dientes (/t/, /d/).
 - d. Alveolares, si se pone la lengua adosada en las encías (/s/, /z/).
 - e. Palatales, si se coloca la lengua adherida a la parte media y anterior del paladar duro (/s/, /z/).
 - f. Velares, si se ubica la lengua contra el velo del paladar (/k/).
 - g. Glotales, si se mantienen fijas y próximas las cuerdas vocales.

2.1.4 El sonido en el audio digital

El audio digital es la codificación digital de una señal eléctrica que representa una onda sonora. Consiste en una secuencia de valores enteros y se obtienen de dos procesos: el muestreo y la cuantificación digital de la señal eléctrica.

El muestreo consiste en fijar la amplitud de la señal eléctrica a intervalos regulares de tiempo (tasa de muestreo).

La cuantificación consiste en convertir el nivel de las muestra fijadas en el proceso de muestreo, normalmente, un nivel de tensión, en un valor entero de rango finito y predeterminado.

En el audio digital, las variaciones de amplitud y de frecuencia se recogen en forma de números, y se toman valores para registrar en qué altura está la onda en cada instante.

Para ello se divide la escala en una serie de tramos que corresponderán a unos valores numéricos entre 0 y el número máximo de tramos que estemos considerando.

Ese número máximo de tramos está definido por los “bits” de sonido con los que estemos trabajando. Así, con 8 bits podemos tener hasta 256 valores posibles, mientras para un sonido de 16 bits tenemos 65536. Esto significa que al trabajar con 16 bits la precisión que obtenemos al reproducir el sonido original es mucho mayor.

Pero además de tener distintas escalas de altura, al digitalizar un sonido también podemos tomar muestras de esa altura con mayor o menor frecuencia. Esto es lo que se denomina la “frecuencia de muestreo”, que indica cuántas medidas del sonido que está siendo digitalizado llevamos a cabo en un segundo. Así, si tenemos una frecuencia de 44.1 KHz, estamos tomando 44.100 mediciones en cada segundo.

2.1.5 Procesamiento digital de audio

Una señal digital no es audible, ya que requiere ser decodificada antes de su reproducción. Sin embargo, la codificación y posterior decodificación de una señal digital para su audición tiene varias ventajas, aunque también algunos inconvenientes.

Ventajas:

- La señal discreta digital es más fácil de transmitir, almacenar o manipular.
- La señal digital es inmune al ruido y es menos sensible a las interferencias que la señal analógica.
- Se puede tomar una muestra de sonido y cambiar cualquiera de sus parámetros para generar un sonido diferente sin tener que recrearlo en la realidad.
- La señal digital permite la multi-generación infinita sin pérdidas de calidad.
- Ante la pérdida de cierta cantidad de información, la señal digital puede ser reconstruida gracias a los sistemas de regeneración de señales, usados también para amplificar la señal sin introducir distorsión. Existen sistemas de detección y corrección de errores que permiten introducir el valor de una muestra dañada, obteniendo el valor medio de las muestras adyacentes. Este fenómeno se conoce como interpolación.
- El envío de la señal digital supone un costo muy bajo, sin que la señal sufra variaciones o alteraciones de calidad severas.
- Con el tiempo no se degrada.

Inconvenientes:

- Se necesita una conversión analógica-digital previa y una decodificación posterior, en el momento de la recepción.
- Hay una pérdida inherente de información al convertir la información continua en discreta. Existe siempre un mínimo error de cuantificación que impide que la señal digital sea exactamente equivalente a la señal analógica de origen.
- La señal digital requiere mayor ancho de banda para ser transmitida que la analógica. Además, requiere una sincronización precisa entre los tiempos del reloj del transmisor, con respecto a los del receptor. Un desfase, por insignificante que sea, cambia la señal por completo.

Si se utiliza compresión con pérdida, será imposible reconstruir la señal

2.1.6 Representación temporal

Un método simple de representación de una señal sonora es dibujarla en una gráfica dependiente del tiempo. Esta representación se denomina representación en el dominio temporal (o *time domain representation*). En este caso, representamos la evolución de la amplitud (de la magnitud que medimos: presión, voltaje, etc.) respecto al tiempo. En el caso del sonido, la amplitud representa la variación de la presión atmosférica respecto al tiempo. En general, la amplitud se representa a partir del valor 0 (posición de equilibrio o valor medio de la presión) hasta el punto de máxima amplitud de la forma de onda.

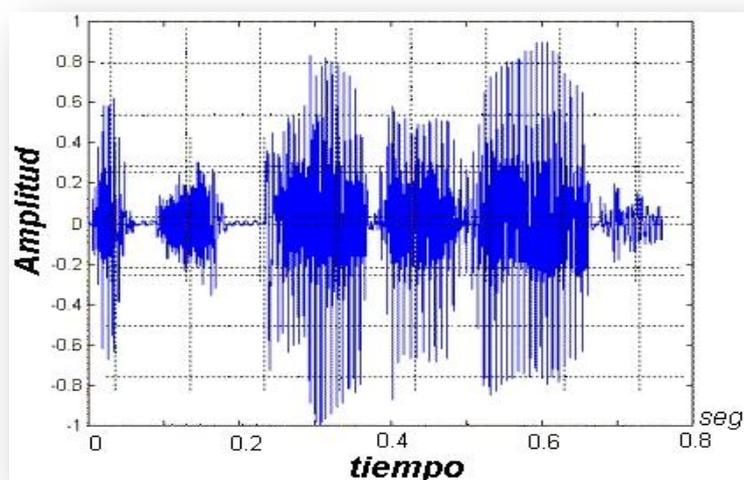


Ilustración 2. Representación temporal de una señal acústica

Cruces por Cero

Los cruces por cero indican el número de veces que una señal continua toma el valor de cero. Para las señales discretas, un cruce por cero ocurre cuando dos muestras consecutivas difieren de signo, o bien una muestra toma el valor de cero. Consecuentemente, las señales con mayor frecuencia presentan un mayor valor de esta característica, el ruido también genera un gran número de cruces por cero.

2.1.7 Representación Frecuencial

La representación frecuencial captura las características espectrales de una señal de audio. Además de la frecuencia fundamental, existen muchas frecuencias presentes en una forma de onda. Una representación en el dominio frecuencial (o *frequency domain representation*) o representación espectral muestra el contenido frecuencial de un sonido. Las componentes de frecuencias individuales del espectro pueden denominarse armónicos o parciales. Las frecuencias armónicas son enteros simples de la frecuencia fundamental. Cualquier frecuencia puede denominarse parcial, sea o no múltiplo de la frecuencia fundamental. De hecho, muchos sonidos no tienen una fundamental clara.

El contenido frecuencial de un sonido puede mostrarse de diversas maneras. Una forma estándar es la de dibujar cada parcial como una línea en el eje x. La altura de cada línea correspondería a la fuerza o amplitud de cada componente frecuencial. Una señal sinusoidal pura viene representada por una sola componente frecuencial.

El objetivo del análisis espectral es describir la distribución, en dominio de la frecuencia, de la potencia contenida en una señal, mediante el análisis de un conjunto finito de datos.

El espectro de potencia de un proceso estacionario aleatorio se relaciona matemáticamente con la secuencia de auto-correlación descrita por la *transformada discreta de Fourier*.

Frecuencia Fundamental

El *pitch* o *frecuencia fundamental* F_0 se determina por la velocidad de apertura o cierre de las cuerdas vocales en la laringe durante la fonación de sonidos sonoros, cuyo inverso corresponde a su vez al periodo fundamental (T_0). De forma general, la definición de periodicidad se determina para intervalos infinitos de análisis. Sin embargo, a efectos prácticos, por ejemplo en el cálculo computacional, su estimación se realiza sobre intervalos finitos que permitan cubrir varios periodos del pitch, o de

manera instantánea a partir de la diferencia entre dos momentos consecutivos del cierre glótico [9].

En la estimación del pitch, se consideran las siguientes restricciones [9]:

- Los segmentos de voz son altamente no estacionarios, o corresponden al producto de vibraciones irregulares de las cuerdas vocales.
- La excitación glótica no es rigurosamente periódica.
- Existe una importante interacción entre la excitación y el tracto vocal, en donde la periodicidad que se observa en la señal resultante, es debida a la acción conjunta de la excitación casi-periódica y los primeros formantes de banda estrecha.
- La segmentación del inicio y del final de las zonas sonoras debe realizarse con alta precisión, tarea que no siempre es fácil de implementar.
- Alto margen dinámico de variación del parámetro F_0 . Por tanto, dicha estimación debe restringirse a un intervalo de valores permitidos, en donde las ventanas de tiempo se ajusten dependiendo del hablante considerado, pudiendo abarcar entre 2 y más de 20 ms (50 - 500Hz), para cubrir voces desde niños o sopranos, hasta barítonos, o entre 6 y 12 ms (80-170Hz), para el caso de locutores adultos promedio.

La estimación del pitch es importante en aplicaciones como la codificación de voz o el desarrollo de sistemas de ayuda a discapacitados, como el entrenamiento para sordos. El pitch se emplea en la determinación de la entonación y las características emocionales de la voz. Así mismo, sus desviaciones pueden indicar la presencia de desórdenes funcionales y patologías [10].

Energía

La función de energía de una señal representa la energía disipada por una resistencia de un ohmio (Ω) cuando se le aplica un voltaje $x(t)$. En una señal continua, la Energía total (E) en el intervalo de tiempo - t_1 a t_2 - está definida como:

$$E = \int_{t_1}^{t_2} |x(t)|^2 dt$$

Para el caso de las señales discretas donde N es el número de muestras de la señal, la energía se define por:

$$E = \sum_{m=0}^{N-1} x(m)^2$$

La variación de energía en la señal de voz se debe a la variación de la presión subglotal y de la forma del tracto vocal. La energía es útil para distinguir segmentos sordos y sonoros en la señal de voz, debido a que los valores de esta característica aumentan en los sonidos sonoros respecto a los sordos.

Se muestra las gráficas de energía y cruces por cero de la palabra 'seis'. Como se puede observar, el valor de la energía varía en relación directa con la amplitud de la señal. La función de Densidad de Cruces por Cero alcanza sus valores más altos cuando se trata de sonidos tales como la 's', que son conocidos como sonidos fricativos.

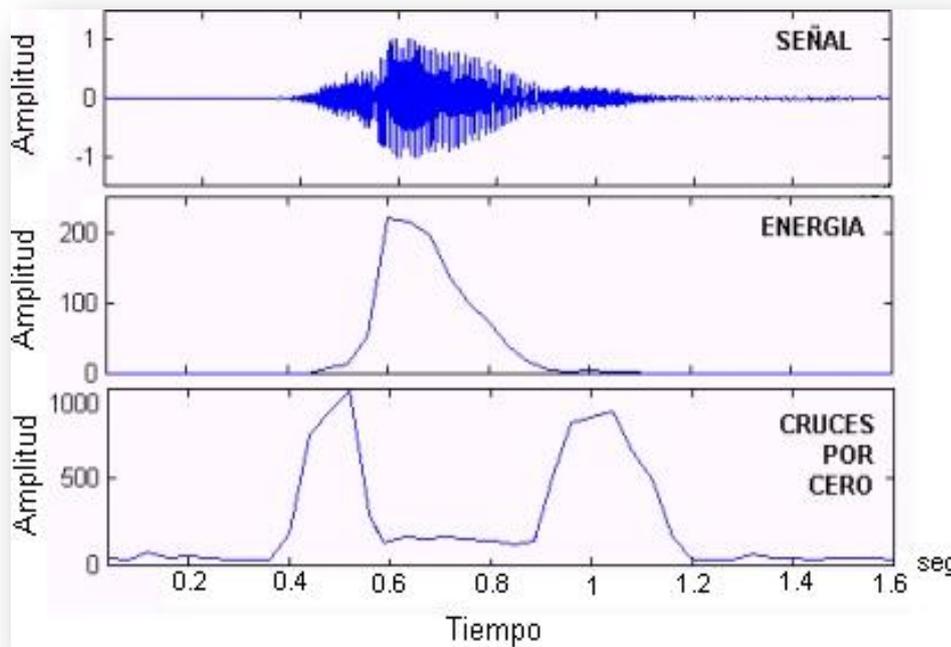


Ilustración 3. Representación de la Energía y Cruces por Cero de la palabra 'seis'

Transformada de Fourier

La transformada de Fourier es una operación matemática que transforma una señal de dominio de tiempo a dominio de frecuencia y viceversa. Estamos acostumbrados a señales con dominio de tiempo en la vida cotidiana. En el dominio de tiempo, la señal se expresa con respecto al tiempo. En el dominio de frecuencia, una señal es expresada con respecto a la frecuencia.

En matemática, la transformada de Fourier, denominada así por el matemático francés *Jean Baptiste Joseph Fourier* (1768-1830) [31], es una aplicación que hace corresponder a una función f , con valores complejos y definida en la recta, con otra función g definida de la manera siguiente:

$$g(\xi) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} f(x) e^{-i\xi x} dx$$

La transformada de Fourier así definida goza de una serie de propiedades de continuidad que garantizan que puede extenderse a espacios de funciones mayores e incluso a espacios de funciones generalizadas.

La transformada de Fourier es básicamente el espectro de frecuencias de una función. Un buen ejemplo de eso es lo que hace el oído humano, ya que recibe una onda auditiva y la transforma en una descomposición en distintas frecuencias (que es lo que finalmente se escucha). El oído humano va percibiendo distintas frecuencias a medida que pasa el tiempo, sin embargo, la transformada de Fourier contiene todas las frecuencias contenidas en todos los tiempos en que existió la señal; es decir, en la transformada de Fourier se obtiene un sólo espectro de frecuencias para toda la función.

Transformada discreta de Fourier (DFT)

La DFT es una función basada en la transformada continua de Fourier.

$$X(j\omega) = \int_{-\infty}^{\infty} x(t) \exp(-j\omega t) dt$$

Para aplicar la transformada de Fourier a una señal muestreada a una frecuencia de muestreo $F_s = 1/T_s$, es necesario discretizarla mediante la transformación en suma de N rectángulos de altura $x(nT_s)$. Siendo T_s el tiempo de muestreo correspondiente. El resultado es la DFT como una aproximación a la función continua.

$$X_d(\omega_k) = \sum_{n=0}^{N-1} x(nT_s) \exp(-j\omega_k nT_s)$$

El proceso de cálculo optimizado de la DFT se denomina Fast Fourier Transformation (*FFT*). La *FFT* es un algoritmo de computación rápido que permite realizar la estimación espectral de series temporales siempre y cuando la longitud de éstas sea potencia de dos. En caso contrario se puede extender la serie a la potencia superior de 2 mediante la introducción de ceros ("*zero padding*"). El principal problema asociado con el análisis frecuencial mediante *FFT* surge de la propia naturaleza de la señal a analizar que tiene un alto grado de aleatoriedad. Esto provoca la aparición en el espectro de picos espureos asociados a componentes no periódicas de la señal. Parte de estos problemas quedan solventados con el empleo de la estimación espectral a partir de métodos auto-regresivos.

2.1.8 Formato WAV

WAV, o WAVE es un formato de audio digital de datos desarrollado por *Microsoft* e *IBM* que se utiliza para almacenar sonidos en el computador. Admite archivos “mono” y “estéreo” a diversas resoluciones y velocidades de muestreo.

WAVE, apócope de “*WAVEform audio file format*” se puede traducir al castellano como el *formato de fichero de audio con forma de onda*, y su extensión es “.wav”.

Este formato es una variante del formato “*Resource Interchange File Format*” (*RIFF*), ó *formato de fichero para intercambio de recursos*, y es el formato principal usado por Windows.

A pesar de que el formato WAV es compatible con casi cualquier códec de audio, se utiliza principalmente con el formato *PCM* (no comprimido) y, al no tener pérdida de calidad, es adecuado para uso profesional. El formato de los ficheros *wav* es el siguiente:

Bytes	Contenido Usual	Propósito/Descripción
00 – 03	“RIFF”	Bloque de identificación (sin comillas)
04 – 07	¿??	Entero largo. Tamaño del fichero en bytes, incluyendo cabecera.
08 – 11	“WAVE”	Otro identificador
12 – 15	“fmt”	Otro identificador
16 – 19	16, 0, 0, 0	Tamaño de cabecera hasta ese punto
20 – 21	1, 0	Etiqueta de formato (versión del tipo de formato utilizado)
22 – 23	1, 0	Número de canales (2 si es estéreo, 1 si es mono)
24 – 27	¿??	Frecuencia de muestreo (muestras/segundo)
28 – 31	¿??	Número medio de bytes/segundo
32 – 33	1, 0	Alineamiento de bloque
34 – 35	8, 0	Número de Bits por muestra (normalmente 8,16 o 32)
36 – 39	“data”	Marcador que indica el comienzo de los datos de las muestras
40 – 43	¿??	Número de bytes muestreados
resto	¿??	Muestras (cuantificación uniforme)

Tabla 3. Formato WAV

2.1.9 Aplicaciones de análisis de audio en Android

Para la realización de este proyecto se han estudiado diferentes aplicaciones que han servido como base para conocer distintas formas de manipular los sonidos obtenidos mediante el micrófono del terminal.

Audalyzer.

Audalyzer es un analizador simple de audio para Android. Su funcionamiento consiste en examinar el sonido del micrófono y mostrar su análisis en forma de ondas, como un espectro de frecuencia, y como un medidor de decibelios (*dB*). Los niveles de *dB* son relativos al nivel de entrada máximo del dispositivo en cuestión.

Dado que la sensibilidad del micrófono varía entre los distintos dispositivos, es imposible generar lecturas precisas del nivel de sonido con esta aplicación, de modo que los resultados se deben estudiar con cierta precaución. Sin embargo, *Audalyzer* puede ser muy interesante cuando se utiliza para comparar las lecturas de sonido en diferentes espacios.

A continuación se observa el aspecto que tiene la interfaz gráfica de la aplicación.

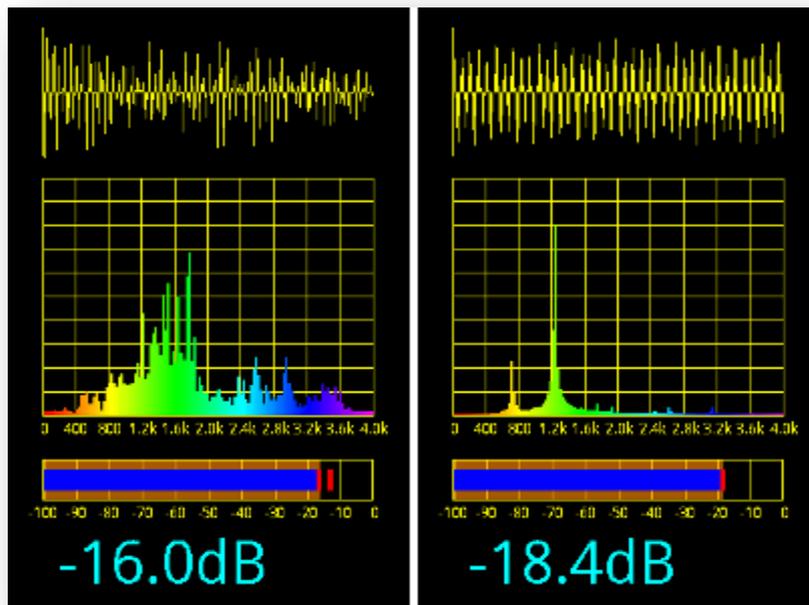


Ilustración 4. Audalyzer

El funcionamiento de esta aplicación está marcado por el uso de diversas bibliotecas comunes a distintas aplicaciones desarrolladas por el mismo equipo que ha desarrollado Audalyzer. Estas bibliotecas son:

HermitLibrary: Destinada a contener utilidades relacionadas con geolocalización, cálculos astronómicos, transformadas de Fourier y herramientas generales.

HermitAndroid: Contiene diversas clases para el manejo de datos de entrada-salida, redes, widgets, etc.

El código es muy modular y está documentado con mucho detalle. En líneas generales, el diagrama simplificado de funcionamiento de la aplicación es el mostrado en la Ilustración 5. Si bien esta organización es fácil e intuitiva tiene un inconveniente a tener en cuenta: depende de la potencia de cálculo del dispositivo para un funcionamiento fluido.

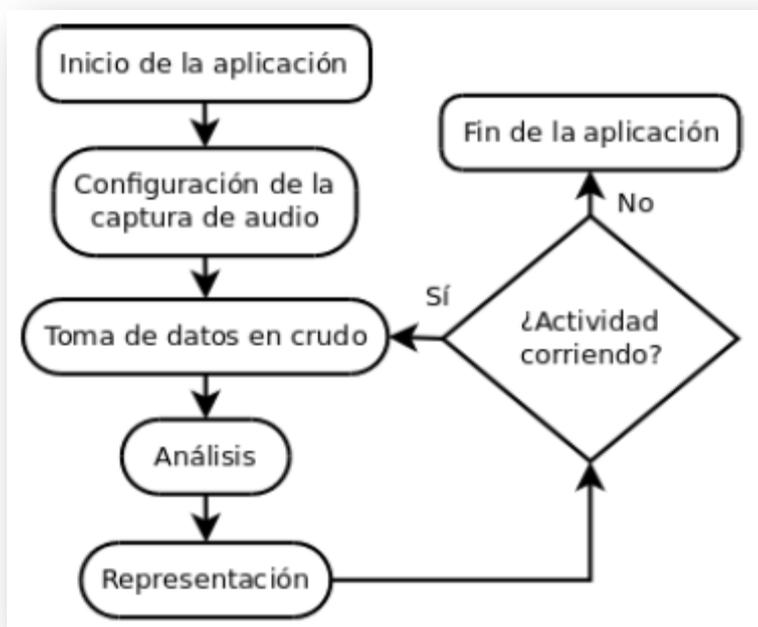


Ilustración 5. Diagrama de flujo de Audalyzer

Sound Meter

Sound Meter es un medidor del *sound pressure level (SPL)* ó nivel de presión sonora (*NPS*), que utiliza el micrófono del teléfono para medir el volumen de ruido en decibelios, y muestra una estimación del *SPL* calculado en pantalla.

En la siguiente ilustración se muestra el aspecto de la interfaz de la aplicación:

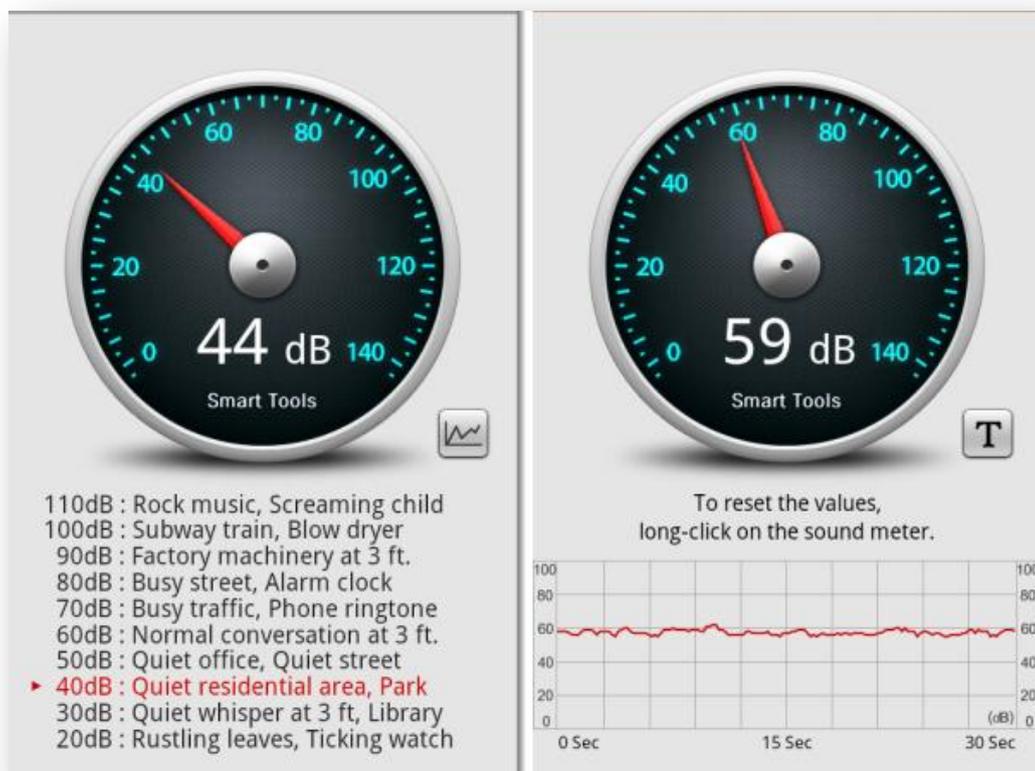


Ilustración 6. Sound Meter

SPL Meter

SPL Meter [11] es una aplicación diseñada para medir la presión acústica y almacenar el máximo de una sesión. Está disponible, al igual que *Audalyzer*, bajo licencia GPL v2. La interfaz de la aplicación se puede observar en la siguiente figura:



Ilustración 7. SPL Meter

Al tener una funcionalidad tan específica, la aplicación es realmente simple, contando con tan solo dos clases, una encargada de la interfaz, que podemos ver en la Ilustración 7, y la otra encargada de leer y analizar los datos del micrófono mediante un hilo.

La estructura de esta aplicación es muy fácil de seguir y, aunque el código no está tan bien documentado, es bastante intuitivo.

Al separar la parte más compleja del código en un hilo, la interfaz se comporta de manera más fluida ya que el sistema se encarga de organizar los recursos, logrando que la adquisición de datos no se bloquee porque se esté dibujando la interfaz en pantalla.

En la siguiente ilustración se puede comprobar cómo se comunica la parte principal de la aplicación con el hilo que realiza el trabajo de la escucha. Cada hilo se muestra con un gradado de fondo mientras que el paso de mensajes se representa por bloques de fondo gris.

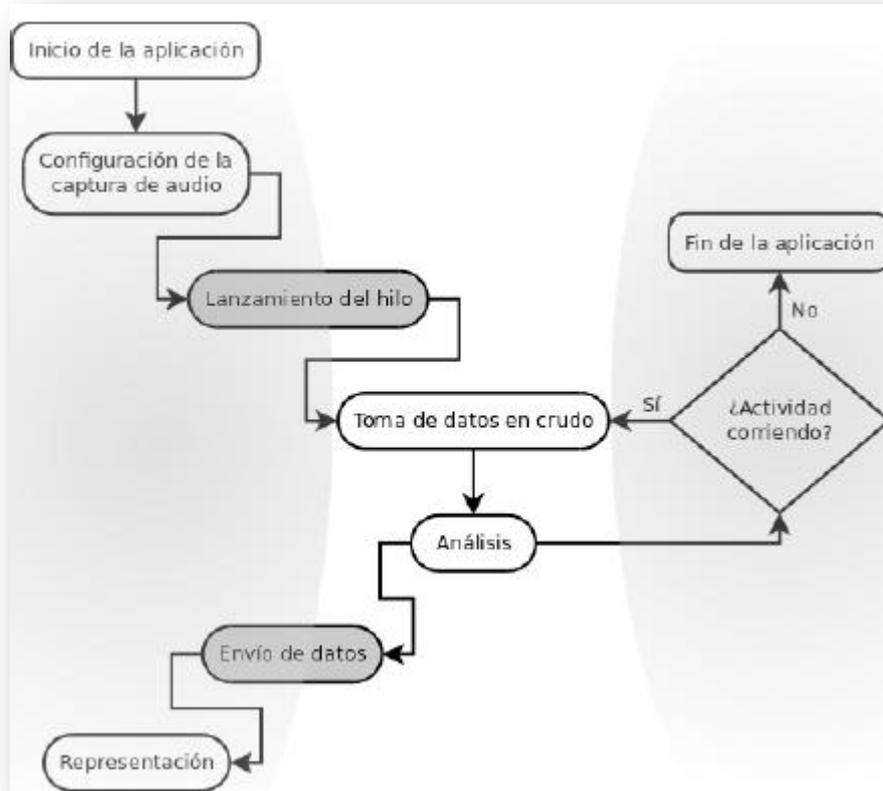


Ilustración 8. Diagrama de flujo de SPL Meter

El análisis de los datos que lleva a cabo la aplicación para obtener el nivel del ruido ambiente se basa en aplicar las siguientes fórmulas:

$$I = Z \times (2\pi \times f)^2 \times A^2 \quad (a)$$

$$\rho = \sqrt[2]{(I \times Z)} \quad (b)$$

$$SPL = 20 \times \log_{10}(\rho/\rho_0) \quad (c)$$

Siendo:

Z Impedancia acústica, su valor es 406.2 W para aire a 30°C. (a)

f Frecuencia en Hz. (a)

A Área de medida en m². (a)



ρ Presión sonora en Pa \times s/m. (b)(c)

ρ_0 Presión sonora de referencia, 0.00002 Pa \times s/m. (c)

SPL Nivel de sonido en dB. (c)

Los pasos vistos en el análisis de la Ilustración 8 son los siguientes:

- Leer las muestras del micrófono configuradas para 8000Hz, 16bit, MONO en una cola de de 4096 bytes.
- Realizar las 4096 transformadas de Fourier para obtener frecuencias y amplitudes.
- Calcular el nivel de sonido con las fórmulas a, b y c.
- Repetir los pasos 1, 2 y 3 hasta que se finalice la aplicación.

Decibel

Decibel es el resultado de un esfuerzo serio para hacer un medidor de SPL Android basado en lo más cercano a un dispositivo de calidad profesional. Opera en función de las capacidades que el hardware del teléfono posee.

La aplicación utiliza el algoritmo de la “Transformada Rápida de Fourier” para obtener una alta precisión. Un filtro de ponderación y sofisticadas técnicas de procesamiento digital de señal se utilizan para alcanzar lecturas del SPL con un alto grado de exactitud.

Entre otras cosas, muestra los valores actuales, máximos y mínimos en decibelios, y genera 3 tipos de nivel de presión sonora mediante gráficos de tiempo. La pantalla del dispositivo puede girarse para permitir apuntar el micrófono hacia la fuente sonora.



Ilustración 9. Decibel

2.2 Estado del arte en detección de ronquidos y ruidos

2.2.1 El Ronquido

El ronquido es un sonido respiratorio que se produce durante el sueño, ya sea nocturno o diurno. El ronquido puede ser inspiratorio, espiratorio o puede ocupar todo el ciclo respiratorio. Tiene su origen en la vibración de los diferentes tejidos de la vía aérea superior. Se han descrito numerosos métodos para analizarlo, desde el simple interrogatorio, pasando por cuestionarios estándares, hasta llegar a los métodos acústicos más sofisticados, que se han desarrollado gracias al gran avance de las técnicas biomédicas en los últimos años.

Los factores necesarios para que se produzca el ronquido son, [12]:

- Sueño.
- Presión negativa inspiratoria torácica (succión).
- Reducción del área de sección.
- Limitación del flujo.
- Estructura vibrante (paladar blando y tejidos blandos de la orofaringe).

2.2.2 Medidas acústicas del ronquido

Análisis en intensidad

La intensidad es la principal característica temporal del ronquido. Cuando se utilizan analizadores de ruido ambientales, la intensidad se expresa en energía media o nivel de intensidad excedido durante un determinado porcentaje de tiempo. El ronquido generado en el paladar es el más intenso y los pacientes afectados del síndrome de apnea-hipopnea durante el sueño (SAHS) roncan con mayor intensidad que los roncadores simples [13]. La intensidad del ronquido es un parámetro que presenta poca variación entre las noches y es, por tanto, bastante reproducible. Se correlaciona con la presión esofágica: cuanto más intenso es el ronquido, mayor es la presión esofágica generada. Esta propiedad se puede ver en los pacientes afectados de síndrome de resistencia de la vía aérea superior, en los que se observa que al ir aumentando la resistencia en la vía aérea superior se produce una mayor intensidad sonora, la que se incrementa de forma gradual hasta finalizar abruptamente. La intensidad del ronquido también se ha aplicado en la detección de apneas e hipoapneas.

Análisis en frecuencia

El análisis en frecuencia estima la densidad espectral de potencia (*PSD*) del ronquido (Ilustración 10). La *PSD* se suele calcular al promediar el periodograma de segmentos consecutivos de la señal. Para calcular el periodograma se aplica la transformada rápida de Fourier. Las frecuencias media y pico son los parámetros más reproducibles, con un coeficiente de variación del 14–16% y están correlacionadas con el índice de apneas y hipoapneas por hora (*IAH*) [14]. En cuanto al rango de frecuencia del ronquido, la mayor parte del espectro está por debajo de los 2.000Hz, aunque el ronquido no rítmico tiene también energía entre los 2.000–3.000Hz. Los modelos matemáticos más recientes indican que el ronquido se puede originar a lo largo de toda la vía aérea superior y cada zona vibra a una frecuencia fundamental. Así, el paladar blando vibra entre 20–80 Hz. Diferentes vibraciones naturales corresponden a diferentes áreas generadoras, que van desde la nariz a la base de la lengua, y presentan un carácter secuencial, y completan la activación de las diferentes estructuras a una frecuencia cercana a los 60 Hz. Este comportamiento de carácter secuencial en orígenes diferentes explicaría el hecho de que en un mismo ronquido los componentes frecuenciales puedan cambiar durante su duración y en toda la noche.

Miyazaki et al [15] estudiaron la frecuencia fundamental del ronquido en función de la localización del colapso medida a través de la presión en la vía aérea. El paladar blando y la úvula generan vibraciones regulares de menos de 150 Hz. El ronquido estenótico, que se genera en la base de la lengua, es de morfología irregular y su frecuencia fundamental está entre los 300–400Hz. El ronquido generado en la laringe tiene carácter musical y su frecuencia fundamental oscila alrededor de los 300 Hz.

Beck et al describieron 2 tipos diferentes de ronquido [16]: ronquido simple y complejo. El ronquido simple tiene un patrón cuasi sinusoidal sin oscilaciones secundarias dentro de la onda. La frecuencia varía entre 170–300Hz y en su espectro se pueden observar de 1–3 picos, el primero de los picos es el más intenso. Es infrecuente encontrar ronquidos con un solo pico y además en el ronquido los picos suelen estar equiespaciados, al contrario de lo que sucede en las sibilancias. En el ronquido simple, el cierre de la vía aérea es parcial.

El ronquido complejo está formado por estructuras temporales repetitivas compuestas de unas pocas oscilaciones. El máximo de la onda coincide con el flujo máximo. Mediante ecuaciones seno exponenciales se puede modelizar este tipo de ronquido. Durante el ronquido complejo se produce el cierre total de la vía aérea. Al ronquido simple se le denomina «flutter» y las paredes oscilan alrededor de la posición media, mientras que en el ronquido complejo tiene lugar el cierre y la apertura de la vía aérea.

Para Meslier et al [17], Dalmasso y Pérez-Padilla et al [18] la morfología más frecuente presenta una frecuencia fundamental de entre 40–80Hz acompañada de armónicos. Pérez-Padilla et al identificaron 2 tipos de espectros: uno similar a la voz generada por las oscilaciones de la glotis, compuesto por una frecuencia fundamental y armónico, y otro patrón caracterizado por un sonido fricativo (h raspada), producido por la fricción del flujo turbulento sobre una vía aérea reducida. Su espectro es amplio, sin un claro componente fundamental, y los componentes armónicos son más difíciles de definir. El ronquido post-apneico pertenece a este último grupo y su energía máxima se sitúa alrededor de los 450 Hz, pero con componentes de más alta frecuencia sobre los 1.000Hz.

El pitch es el periodo fundamental de una señal oscilatoria. Su inversa es la frecuencia fundamental. Los seres humanos perciben el ronquido como si fuera un conjunto de paquetes de energía que se repiten. La distancia entre estos paquetes es el pitch. Mediante el análisis del pitch es posible diferenciar a los pacientes roncadores simples y los afectados de SAHS. Se demostró que el pitch medio de los pacientes con SAHS es mayor que en los roncadores sin SAHS. La frecuencia fundamental y el pitch o periodo fundamental varían también dentro del mismo.

Las frecuencias de los formantes del espectro se pueden calcular mediante la envolvente espectral, que contiene información de los pulsos de presión y del efecto producido por el filtro anatómico. La envolvente se obtiene al aplicar modelos auto-regresivos de bajo orden. La mayor parte de los ronquidos presentan entre 3–5 formantes, que indican las frecuencias de resonancia de la vía aérea superior, ubicados en unas bandas frecuenciales bien definidas. Según un estudio de Solà -Soler et al [19], existe una mayor variabilidad en la distribución de los formantes en los roncadores con SAHS en relación con los roncadores simples. Estos mismos autores han estudiado recientemente los formantes de la respiración a volumen corriente en los sujetos roncadores con y sin SAHS. Los formantes de la respiración aparecen en las mismas bandas frecuenciales que en el caso del ronquido y algunos de ellos también muestran una variabilidad significativamente superior en los sujetos con SAHS. La mayor variabilidad de la frecuencia de los formantes, observada tanto en el sonido de la respiración como en el ronquido de los sujetos patológicos, podría explicarse por la mayor inestabilidad de la vía aérea superior de los pacientes con SAHS, que hace que el origen de las vibraciones sea más variable que en los roncadores simples.

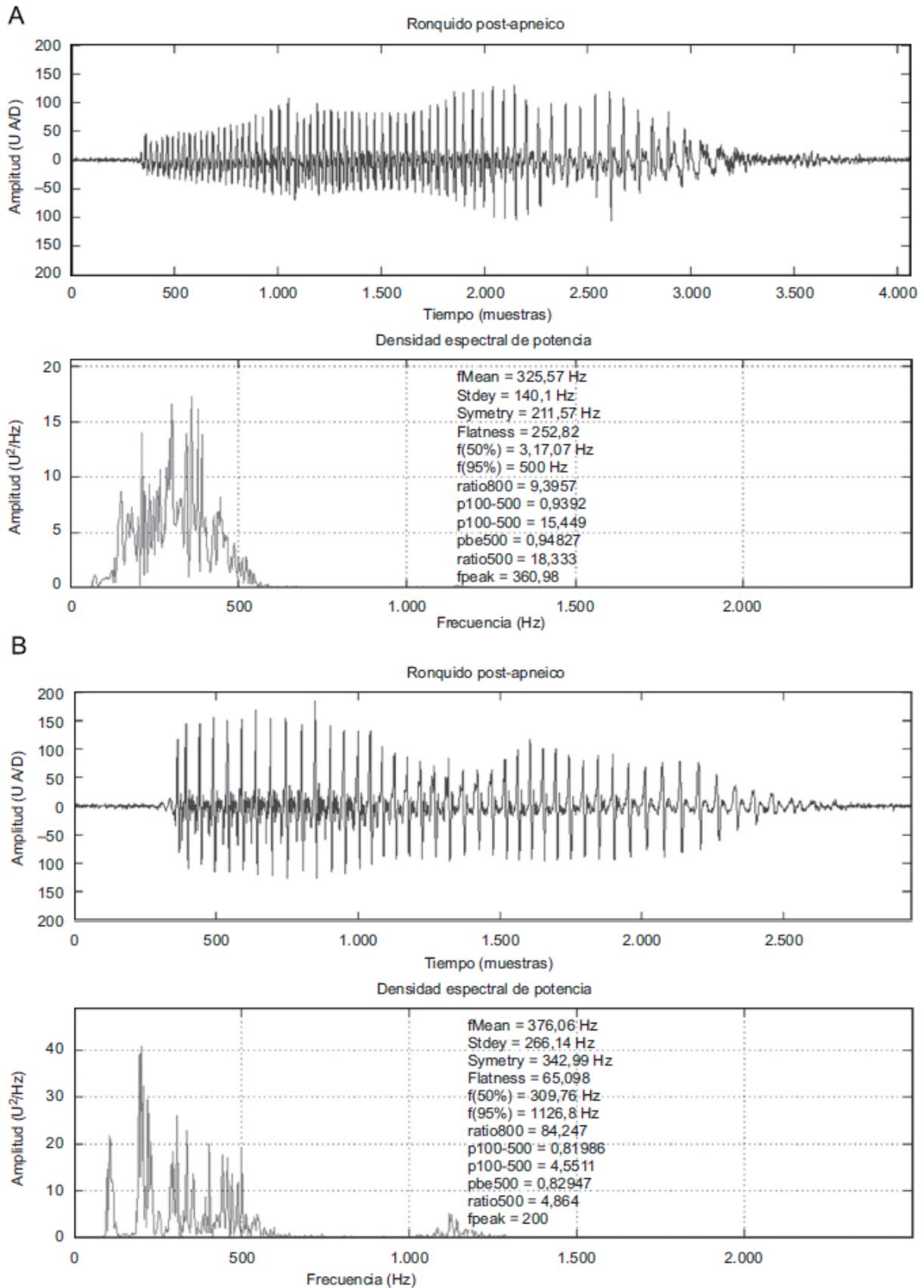


Ilustración 10. 2 señales de ronquido y su correspondiente densidad espectral de potencia

El sistema MESAM IV1 [20] fabricado en Alemania (Madaus Medizin Elektronik, Alemania. Nivel 4) recoge la SaO₂, la frecuencia cardíaca, el ronquido y la posición. Calcula la potencia del espectro entre 100–800 Hz. Si excede en este rango por encima del 50% del total, se identifica el acontecimiento como ronquido, y es intenso cuando el volumen excede un umbral determinado. Una versión posterior al MESAM1 es el POLIMESAM1 (nivel 3). El sistema incorpora la medida de flujo nasal además de los canales del MESAM IV1.

De lo arriba expuesto, podemos concluir que el ronquido puede analizarse a través de métodos cualitativos, analizadores de sonido que fundamentalmente recogen la intensidad, la densidad, la variabilidad y las frecuencias predominantes. Los métodos acústicos son los que más se acercan a la realidad física del ronquido. Pero, desafortunadamente, existe una gran escasez de monitores-analizadores del ronquido, ya que los monitores existentes en su mayoría se han diseñado para detectar acontecimientos nocturnos tipo apnea-hipoapnea y realizar un primer cribado de identificación de los pacientes afectados de SAHS.

2.2.3 Parámetros de ruido

Los parámetros basados en la relación entre la energía armónica y la energía de ruido tienen amplia aplicación en la clínica de la voz, por su estrecha relación con muchas disfonías. Por otra parte son muy sensibles a los sistemas de registro empleados ya que éstos pueden introducir niveles de ruido ajenos a los de la propia voz alterando los resultados. La formulación concreta de éstos resultados varía de unos algoritmos a otros.

Por ejemplo, para calcular el *HNR* (razón armónico ruido, harmonics-to-noise ratio) o su recíproco *NHR* (razón ruido armónico, noise-to-harmonics ratio) en [29] se considera que la energía de ruido se encuentra en la región de los 2800 a los 5800 Hz, dado que la mayor parte del contenido de ruido de la voz pertenece a altas frecuencias.

Varios métodos se han propuesto para separar la señal de voz en sus componentes periódico y aperiódico: ellos están basados en modelado senoidal, en modelado armónico más ruido, entre otros. Toda la energía presente en las frecuencias armónicas está asociada a la componente determinística, pero en las señales de voz la energía del ruido está esparcida por todo el ancho de banda.

No en todos los problemas relacionados con extracción de ruido se tiene una definición exacta de lo que es ruido, en lugar de ello, Coifman y Wickerhauser [30] proponen un método de reducción de ruido el cual tiene la habilidad de reducir el ruido incoherente en una señal sin afectar los componentes coherentes. La parte incoherente

de la señal (ruido) es aquella que no está lo suficientemente correlacionada con las formas de onda de las bases.

2.2.4 Determinación de sonido Sonoro/Sordo

La *ZCR* (Zero Crossing Rate) se basa en contar el número de veces que la señal cruza por cero en la longitud de una muestra de audio.

Con este ratio, se trata de reforzar el efecto de dos medidas independientes relacionadas con el carácter sordo o sonoro de la señal de voz. En efecto, la tasa de cruces por cero es en general mayor para los sonidos sordos, que suelen tener componentes a más altas frecuencias. La energía o la potencia, por el contrario tiende a ser menor para los sonidos sordos. Por ello, calcular el ratio energía entre *ZCR* hace que ambos efectos se refuercen: para tramas sordas el valor será pequeño, y será mucho mayor (4 órdenes de magnitud aprox.) cuando la trama sea sonora.

El ronquido debería poder diferenciarse de otros sonidos que se pueden registrar durante el sueño. Pérez-Padilla [21] lo define como un sonido inspiratorio cuya amplitud es cinco veces superior al sonido cardíaco. Posteriormente el mismo autor lo define como cualquier ruido repetitivo de baja frecuencia y/o con componentes vibratorios. Hoffstein [22] como cualquier pico de sonido cuya intensidad excede los 55 dB. Wilson [23] define y cuantifica el ronquido en función de la media de la amplitud de la presión sonora y el porcentaje de tiempo que excede al nivel de ruido ambiental. Sin embargo, probablemente el principal problema de todas las mediciones es la falta de validación biológica. Antes de aceptar que una señal acústica de determinada amplitud, duración o frecuencia es un ronquido, debe verificarse que esa señal corresponde al sonido que un oyente (o preferiblemente más de uno) percibe como un ronquido.

R.Beck et al [16], determinan dos tipos fundamentales de ondas en los ronquidos, a las que denominan “simples” y “complejas”; las primeras de características cuasi-senoidales, corresponderían a ronquidos sin obstrucción respiratoria, y los segundos que comienzan con una alta defeción seguida de una amortiguación de la onda, estarían asociadas a breves periodos de pérdida de respiración.

2.2.5 Aplicaciones de análisis de ronquidos

Para la realización de este proyecto se han utilizado dos aplicaciones, que han servido como base para conocer distintas formas de detectar ronquidos desde el dispositivo móvil.

Snore No more

“Snore No more” analiza el sonido ambiente y es capaz de reconocer ronquidos.

Esta aplicación puede configurarse para despertar al usuario cuando ronque por la vibración o el zumbido. También registra la actividad que se produce durante el sueño, para saber cuándo y contabilizar el número de veces que el usuario ha roncado.



Ilustración 11. Snore No More

Antisnore

AntiSnore registra y monitoriza el comportamiento del usuario durante el sueño y detecta los sonidos del ronquido a través del micrófono. El móvil se coloca en posición vertical en el bolsillo del pijama con los auriculares conectados. La aplicación detecta la posición del durmiente por sensores de aceleración integrados en el Smartphone. Si se detecta el ronquido, se comienza a oír el molesto sonido de un mosquito que se va incrementando poco a poco, pero sólo en un auricular. Esto causa que el durmiente se dé la vuelta alejándose del origen del sonido. El cambio de posición hace que cese el ronquido.

Para no despertar al durmiente, el sonido se suaviza inmediatamente tan pronto como se detecta el movimiento. Si la posición no se ha corregido suficientemente y el durmiente sigue roncando, el sonido comienza de nuevo hasta que el cambio de posición se haya finalizado y el ronquido se pare.

Todos los parámetros de la aplicación, como el nivel para empezar con el sonido del mosquito, volumen máximo, *fade-out* de volumen, *fade-in* duración, *fade-out* de duración, son ajustables individualmente.

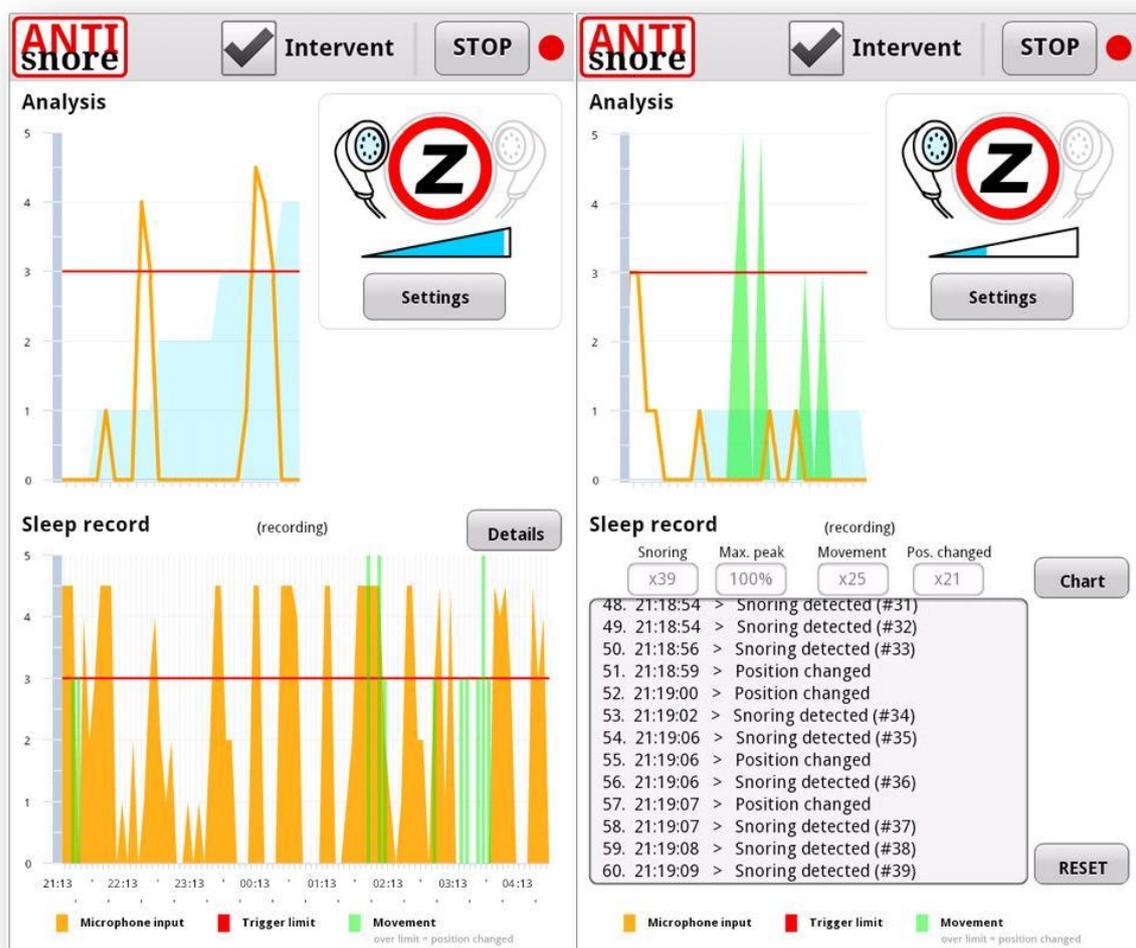


Ilustración 12. Antisnore

2.2.6 Aplicaciones de grabación de somniaquias

Para la realización de este proyecto se ha utilizado una aplicación que ha servido como base para el desarrollo del módulo encargado de grabar somniaquias.

Sleep Talk Recorder

Sleep Talk Recorder graba sonidos y soliloquios nocturnos, pero no se trata sólo de una grabación en curso. Una de las funciones avanzadas de filtrado califica sonidos y desencadena la tecnología Sleep Talk Recorder en una milésima de segundo.

La aplicación tiene una interfaz elegante con ajustes individuales para la sensibilidad y el filtrado. Mantiene un historial que guarda las grabaciones y los ordena en una línea de tiempo para una fácil navegación.

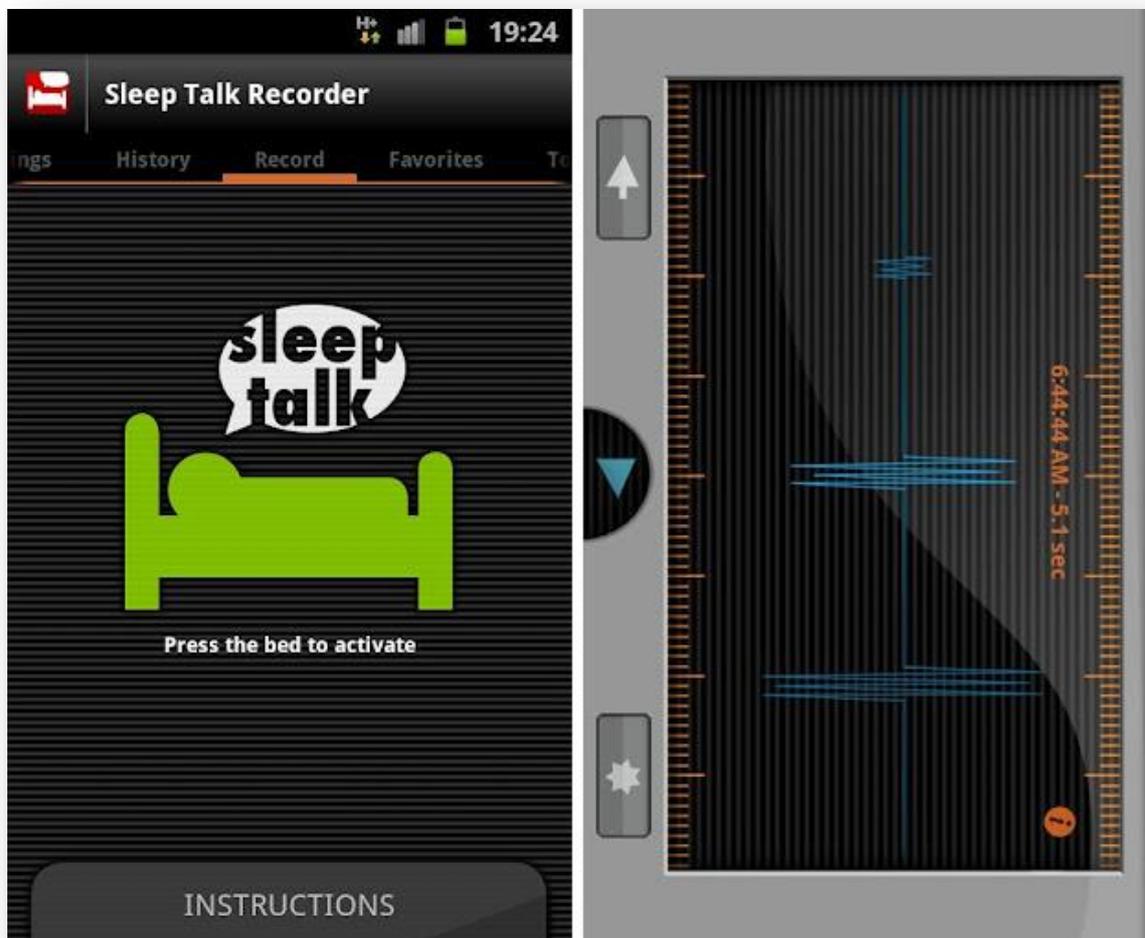


Ilustración 13. Sleep Talk Recorder





Capítulo 3

Tecnologías utilizadas

*"Nunca ande por el camino trazado, porque él
sólo conduce a donde ya fueron otros."*

Alexander Graham Bell (1847-1922)

3.1 Selección de la plataforma

Una de las primeras decisiones que tuvimos que tomar fue elegir la plataforma a utilizar en el desarrollo del proyecto. Para ello debíamos seleccionar aquella de las disponibles en el mercado de los teléfonos móviles que se ajustase a las necesidades que teníamos tanto a nivel hardware como software.

Desde un primer momento pensamos en Android o IOS como plataformas para desarrollar la aplicación móvil, por su madurez y popularidad. Ambas ofrecen una serie de herramientas bastante avanzadas para el desarrollo de las aplicaciones y, además, llevan suficiente tiempo en el mercado como para obtener *feedback* en la red en el caso de que surjan problemas. A continuación analizamos algunos aspectos relacionados con las dos plataformas:

Por un lado, la accesibilidad que ofrece Android es mayor que IOS. El entorno de desarrollo “Eclipse” está disponible desde cualquier sistema operativo, mientras que XCode sólo está disponible para MAC. Además, IOS está ligado a los dispositivos de Apple, mientras que Android es adaptable por los fabricantes, y por lo tanto accesible a través de una amplia gama de dispositivos. Eso sí, el desarrollo en IOS es más homogéneo ya que no es preciso tener en cuenta diferentes tamaños de pantalla, resoluciones, disposición de los botones, etc.

En cuanto al lenguaje de programación, analizamos las ventajas y desventajas de ambos lenguajes: Java, utilizado en Android, y Objective-C, utilizado en IOS. Java es un lenguaje de programación muy conocido, por lo tanto en Android se puede tener una base ya adquirida. Las ventajas culturales y prácticas que ofrece una comunidad tan grande como Java son mayores que las ofrecidas por Objective-C.

3.2 Java

3.2.1 Historia

Surgió en 1991, cuando un grupo de ingenieros de Sun Microsystems trataron de diseñar un nuevo lenguaje de programación destinado a electrodomésticos. La reducida potencia de cálculo y memoria de los electrodomésticos llevó a desarrollar un lenguaje sencillo capaz de generar código de tamaño reducido.

Java [19] fue diseñado para soportar múltiples arquitecturas, debido a que desarrollaron un código independiente del tipo de electrodoméstico, el cual se ejecutaba sobre una máquina virtual denominada Java Virtual Machine (JVM). Era la JVM quien interpretaba el código convirtiéndolo a código particular dependiendo de la CPU utilizada. Esto permitía lo que luego se ha convertido en el principal lema de lenguaje: "Write Once, Run Everywhere". A pesar de los esfuerzos realizados por sus creadores, ninguna empresa de electrodomésticos se interesó por el nuevo lenguaje.

Con la llegada del navegador web Mosaic el equipo reorientó el proyecto hacia internet. Para ello, el código compilado tenía que poder transportarse a través de redes, operar en cualquier cliente, y asegurar al cliente que el código era seguro para ejecutarse.

La popularización de la World Wide Web hizo que estos atributos fueran mucho más interesantes. Los navegadores web permitían a millones de personas navegar por la red y acceder a contenido multimedia de una manera sencilla. Sin embargo los usuarios de internet pronto descubrieron que el formato HTML era demasiado limitado. Las extensiones HTML, tales como formularios, sólo pusieron de relieve las limitaciones, dejando claro que ningún navegador podía incluir todas las características que los usuarios querían.

El navegador de Sun, HotJava™, fue el primero en mostrar las interesantes propiedades que poseía la plataforma y el lenguaje de programación Java para integrar programas dentro de páginas HTML. Los programas eran descargados de forma transparente en el navegador junto con las páginas HTML en las que aparecían. Al igual que las páginas HTML, los programas compilados eran independientes de la red y del host.

En 1994, se hizo una demostración de HotJava y la plataforma Java a los ejecutivos de Sun. Java 1.0 pudo descargarse por primera vez en 1994, pero hubo que esperar al 23 de mayo de 1995, durante las conferencias de SunWorld, a que vieran la luz pública Java y HotJava, el navegador Web. El acontecimiento fue anunciado por John Gage, el

Director Científico de Sun Microsystems. El acto estuvo acompañado por una pequeña sorpresa adicional, el anuncio por parte de Marc Andreessen, Vicepresidente Ejecutivo de Netscape, de que Java sería soportado en sus navegadores. El 9 de enero de 1996, Sun fundó el grupo empresarial JavaSoft para que se encargase del desarrollo tecnológico. Dos semanas más tarde la primera versión de Java fue publicada.

3.2.2 Características

Las principales características pueden resumirse en:

- Java es un lenguaje de programación de propósito general, concurrente y orientado a objetos.
- Su sintaxis es similar a C y C ++, pero omite muchas de las características que hacen que C y C ++ sea complejo, confuso e inseguro.
- Independiente de la plataforma de ejecución.
- Lenguaje compilado e interpretado.
- Lenguaje orientado a internet.
- JVM se implementa en código dependiente de la plataforma.
- Fuertemente tipado.
- Gestión de memoria por la JVM: para ello utiliza el operador new para la creación de objetos y la liberación de recursos es gestionada por el recolector de basura.

3.2.3 Máquina virtual de Java (JVM)

La máquina virtual de Java es la piedra angular de la plataforma Java. Es el componente de la tecnología responsable del hardware y de que sea independiente del sistema operativo, así como del pequeño tamaño de su código compilado, y de proteger a los usuarios de programas maliciosos.

La JVM es una máquina de computación abstracta. Al igual que una máquina de computación real, tiene un conjunto de instrucciones y manipula varias áreas de memoria diferentes en tiempo de ejecución.

La implementación del primer prototipo de la máquina virtual de Java, hecho en Sun Microsystems, Inc., emuló el conjunto de instrucciones de la JVM en un software alojado en un dispositivo de mano. Las implementaciones actuales de Oracle emulan la máquina virtual de Java en dispositivos móviles, en ordenadores personales y en servidores, pero la máquina virtual de Java no asume ninguna tecnología de implementación particular, hardware o sistema operativo del host donde se aloje.

La máquina virtual de Java no entiende el lenguaje de programación Java, sólo de un formato binario especial, el formato de los ficheros “.class”. Un fichero de clase contiene instrucciones de la máquina virtual de Java y una tabla de símbolos, así como otra información complementaria. Por razones de seguridad, la máquina virtual Java impone fuertes restricciones sintácticas y estructurales en el código en un archivo “.class”.

3.2.4 Kit de desarrollo y entorno de ejecución

El Java Development Kit (JDK) es un entorno de desarrollo para aplicaciones Java. JDK no es un entorno integrado de desarrollo en el sentido de aglutinar todas las herramientas bajo la misma interfaz gráfica si no que se ejecutan en modo de comando. Los comandos principales son:

- javac: compila el código y genera los archivos class.
- java: intérprete de la JVM. Ejecuta los bytecodes de los ficheros class.
- javadoc: genera documentación
- AppletViewer: visualiza páginas HTML.
- jdb: depura aplicaciones Java.

El Java Runtime Environment (JRE) es el software necesario para ejecutar cualquier aplicación desarrollada para la plataforma Java, y está incluido dentro del JDK. Puede también obtenerse como un paquete independiente, y puede considerarse como el entorno necesario para ejecutar una aplicación Java, mientras que un desarrollador debe además contar con otras facilidades que ofrece el JDK.

3.2.5 Versiones

A continuación especificaremos el historial de versiones de Java.

JDK 1.0	Nombre clave: OAK Fecha de lanzamiento: 23 de enero de 1996 Características: <ul style="list-style-type: none">• Nació como herramienta para los sistemas operativos Solaris, Windows, Mac Os y Linux.• La primera versión estable fue la JDK 1.0.2.
JDK 1.1	Nombre clave: -- Fecha de lanzamiento: 19 de febrero de 1997 Mejoras: <ul style="list-style-type: none">• Reestructuración intensiva del modelo de eventos AWT.• Clases internas.

	<ul style="list-style-type: none">• JavaBeans.• JDBC.• RMI.
J2SE 1.2	<p>Nombre clave: Playground Fecha de lanzamiento: 8 de diciembre de 1998 Nomenclatura:</p> <ul style="list-style-type: none">• Esta y las siguientes versiones fueron recogidas bajo la denominación Java 2 y el nombre J2SE, reemplazó a JDK para distinguir la plataforma base de J2EE y J2ME. <p>Mejoras:</p> <ul style="list-style-type: none">• Reflexión en la programación.• La palabra reservada strictfp.• La API gráfica fue integrada en las clases básicas.• La máquina virtual fue equipada con un compilador JIT por primera vez.• Java Plug-in.• Java IDL, una implementación de IDL para la interoperabilidad con CORBA• Colecciones.
J2SE 1.3	<p>Nombre clave: Kestrel Fecha de lanzamiento: 8 de mayo de 2000 Mejoras:</p> <ul style="list-style-type: none">• Inclusión de la máquina virtual de HotSpot JVM.• RMI fue cambiado para que se basara en CORBA.• JavaSound• Inclusión del JNDI en el paquete de librerías principales.• JPDA
J2SE 1.4	<p>Nombre clave: Merlin Fecha de lanzamiento: 6 de febrero de 2002 Mejoras:</p> <ul style="list-style-type: none">• Palabra reservada assert.• Expresiones regulares modeladas al estilo de las expresiones regulares Perl.• Encadenación de excepciones Permite a una excepción encapsular la excepción de bajo nivel original.• Non-blocking.• Logging API.• API para la lectura y escritura de imágenes en formatos como JPEG o PNG.

	<ul style="list-style-type: none">• Parser XML integrado y procesador XSLT.• Seguridad integrada y extensiones criptográficas.• Java Web Start.
Java SE 6	<p>Nombre clave: Mustang</p> <p>Fecha de lanzamiento: 11 de diciembre de 2006</p> <p>Nomenclatura:</p> <ul style="list-style-type: none">• Sun cambió el nombre J2SE por Java SE y eliminó el ".0" del número de versión <p>Mejoras:</p> <ul style="list-style-type: none">• Incluye un nuevo marco de trabajo y APIs que hacen posible la combinación de Java con lenguajes dinámicos como PHP, Python, Ruby y JavaScript.• Incluye el motor Rhino, de Mozilla, una implementación de Javascript en Java.• Incluye un cliente completo de Servicios Web y soporta las últimas especificaciones para Servicios Web.• Mejoras en la interfaz gráfica y en el rendimiento.
Java SE 7	<p>Nombre clave: Dolphin</p> <p>Fecha de lanzamiento: julio de 2011</p> <p>Mejoras:</p> <ul style="list-style-type: none">• Soporte para XML dentro del propio lenguaje.• Un nuevo concepto de super paquete.• Soporte para closures.• Introducción de anotaciones estándar para detectar fallos en el software.• Java Module System.• Java Kernel.• Nueva API para el manejo de Días y Fechas, la cual reemplazara las antiguas clases Date y Calendar.• Posibilidad de operar con clases BigDecimal usando operadores.

Los componentes de Java han ido cambiando a lo largo de su historia, pero en la versión actual, la Java SE 7, estos son los por los que contiene:

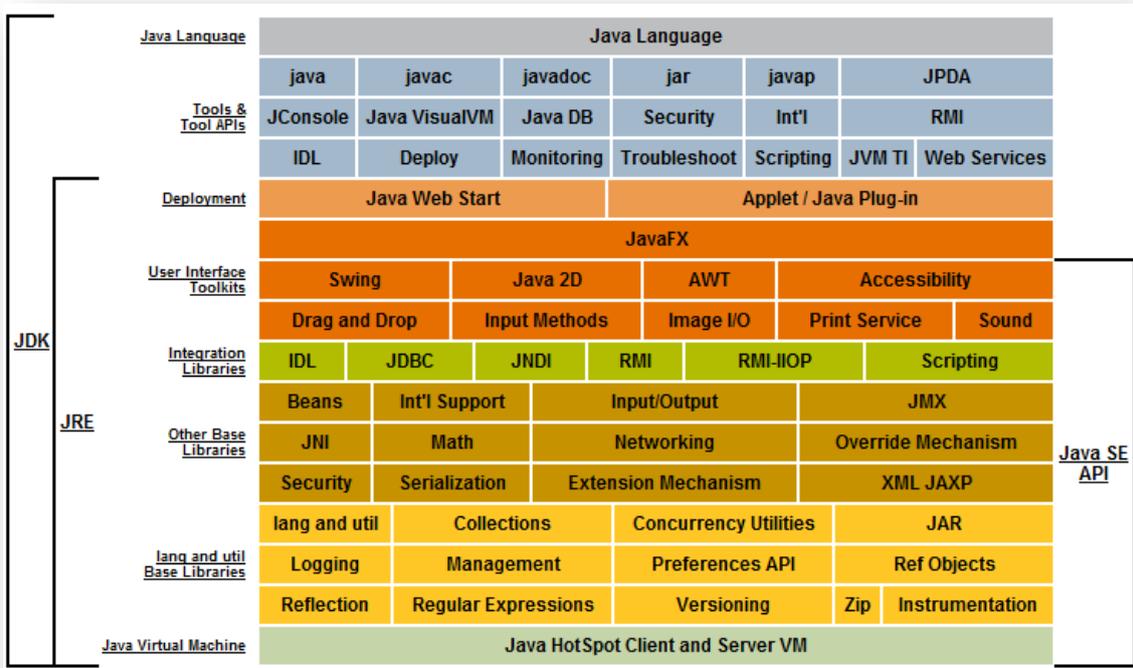


Ilustración 14. Componentes de Java SE 7

3.2.6 Distribuciones de Java

Existen cuatro plataformas del lenguaje de programación Java:

- Java Platform, Standard Edition (Java SE)
- Java Platform, Enterprise Edition (Java EE)
- Java Platform, Micro Edition (Java ME)
- JavaFX

Cada plataforma Java proporciona una máquina virtual y una interfaz de programación de aplicaciones, esto permite que las aplicaciones escritas en una plataforma puedan ejecutarse en cualquier sistema compatible con todas las ventajas del lenguaje de programación Java: independencia de la plataforma, potencia, estabilidad, facilidad de desarrollo y seguridad.

Java SE

Es la base para las otras distribuciones Java. Su principal componente es el API, el cual proporciona la funcionalidad básica del lenguaje de programación Java. Define todo lo necesario, desde los tipos básicos hasta clases que se utilizan para la creación de redes, seguridad, acceso a base de datos, interfaz gráfica de usuario (GUI) desarrollo y análisis XML.



Además de la API, la plataforma Java SE dispone de una máquina virtual, herramientas de desarrollo, tecnologías de implementación, y otras bibliotecas de clases y herramientas usadas en aplicaciones de tecnología Java.

Java EE

La plataforma Java EE está construida sobre la plataforma Java SE. La plataforma Java EE proporciona una API y un entorno de ejecución para desarrollar y ejecutar aplicaciones de red a gran escala, a varios niveles y escalables.

Java ME

La plataforma Java ME proporciona un API y una máquina virtual de pequeñas dimensiones para la ejecución de aplicaciones en dispositivos pequeños como los móviles. Este API es un subconjunto de la API de Java SE, junto con las bibliotecas especiales de clases útiles para el desarrollo de aplicación en dispositivos pequeños. Las aplicaciones JavaME a menudo son clientes de servicios de la plataforma Java EE.

Java FX

JavaFX es una plataforma para la creación de aplicaciones dinámicas de Internet usando una interfaz de usuario ligera. Las aplicaciones de JavaFX utilizan gráficos de aceleración hardware y motores de comunicación para aprovechar los clientes de mayor rendimiento y además aporta un moderno look-and-feel y APIs de alto nivel para la conexión de fuentes de datos en red. Aplicaciones JavaFX pueden ser clientes de servicios de la plataforma Java EE.

3.3 Android

3.3.1 Introducción a Android

Android es un sistema operativo desarrollado originalmente para dispositivos móviles basado en el núcleo Linux. Es una plataforma de código abierto que permite controlar dispositivos por medio de bibliotecas desarrolladas o adaptados por Google mediante el lenguaje de programación Java.

La arquitectura interna de la plataforma Android, está básicamente formada por cinco componentes: Applications, Applications Framework, Libraries, Android Runtime and Linux Kernel. Cada aplicación Android se ejecuta en su propio proceso, con su propia instancia de la máquina virtual de Dalvik.

En cuanto a la funcionalidad que Android nos puede aportar a nuestro proyecto, hemos tenido en cuenta las características que aparecen en la Tabla 4.

Características Android	
Amplia variedad de diseños	VGA, librerías de gráficos 2D, 3D, etc.
Almacenamiento	SQLite, una base de datos usada para propósitos de almacenamiento de datos.
Conectividad	Soporta las siguientes tecnologías de conectividad: GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE y WiMAX.
Mensajería	SMS, MMS
Soporte Java	El bytecode Java no es ejecutado, sino que primero se compila en un ejecutable Dalvik y corre en la Máquina Virtual Dalvik.
Máquina Virtual	Dalvik es una máquina virtual especializada, diseñada para Android y optimizada para dispositivos móviles con memoria y procesador limitados.
Soporte de formatos	MPEG-4, H.264, WAV, MP3, AAC, OGG, AMR, JPEG, PNG, GIF
Soporte para hardware adicional	Cámaras de vídeo, pantallas táctiles, GPS, acelerómetros, etc.
Entorno de desarrollo	Incluye un emulador de dispositivos, herramientas para depuración de memoria y análisis del rendimiento del software.

Tabla 4. Características Android

3.3.2 Arquitectura

Android es un sistema operativo diseñado por capas como se puede apreciar en la siguiente ilustración. El núcleo es una versión modificada del kernel de Linux 2.6. A continuación se explican brevemente cada una de las diferentes capas que componen el sistema.

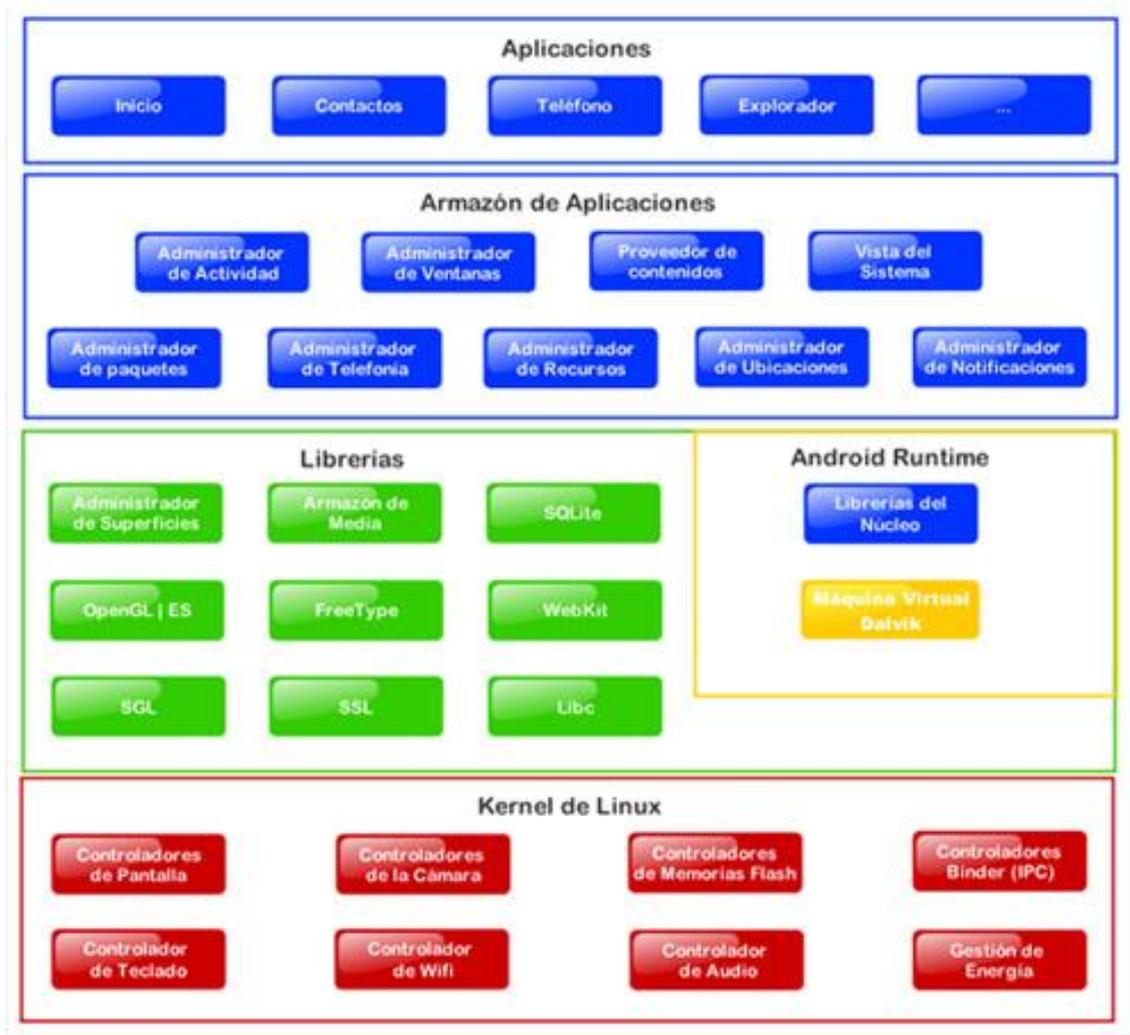


Ilustración 15. Arquitectura Android

A continuación, se detallarán los diferentes componentes principales de la arquitectura de Android:

- **Aplicaciones:** estas aplicaciones son los programas que pueden utilizar el usuario o el sistema. Incluyen aquellas que ya se integran por defecto en el sistema operativo y también las que el usuario decida instalar, creadas por desarrolladores. Las aplicaciones base incluirán un cliente de email,

programa de SMS, calendario, mapas, navegador, contactos, y otros. Todas las aplicaciones están escritas en el lenguaje de programación Java.

- Framework de aplicaciones: Los desarrolladores tienen acceso completo a las APIs del framework usado por las aplicaciones base. La arquitectura está diseñada para simplificar la reutilización de componentes; cualquier aplicación puede publicar sus capacidades y cualquier otra aplicación puede luego hacer uso de esas capacidades (sujeto a reglas de seguridad del framework). Éste mismo mecanismo permite que los componentes sean reemplazados por el usuario. Este framework está formado por:
 - Un extenso conjunto de Vistas tales como listas, cajas de texto, botones, etc.
 - Content Providers que permiten a las aplicaciones acceder a información de otras aplicaciones o compartir su propia información.
 - Resource Manager, que proporciona acceso a recursos que no son código como pueden ser gráficos, cadenas de texto...
 - Notification Manager que permite a las aplicaciones mostrar alarmas personalizadas en la barra de estado.
 - Activity Manager, que gestiona el ciclo de vida de las aplicaciones.

- Librerías: Android incluye un set de librerías C/C++ usadas por varios componentes del sistema. Estas capacidades se exponen a los desarrolladores a través del framework de aplicaciones de Android, el cual interactúa con las librerías mediante JNI (Java Native Interface). Algunas son: System C library (implementación de librería C estándar), librerías de medios, librerías de gráficos, 3d, SQLite, entre otras.

Nombre de la librería	Descripción
System C library	Se trata de una implementación de la librería estándar de C (libc) optimizada para utilizarse en dispositivos móviles.
Media libraries	Es la librería encargada de reproducir y grabar contenido audiovisual, es decir, música, videos e imágenes. Soporta los formatos más utilizados como el MPEG4, el MP3 o el JPG. sin embargo, se han encontrado algunas limitaciones como el hecho de que aun no soporte imágenes vectoriales en formato SVG,
Surface Manager	Esta librería permite gestionar el acceso al sistema de visualización y se encarga de la composición de capas gráficas tanto en 2D como en 3D.
LibWebCore	Es el motor de navegación web integrado en el sistema. Lo utiliza el navegador web en cualquier aplicación mediante las llamadas correspondientes.
SGL	Se trata del motor básico de gráficos en dos dimensiones.
3D libraries	Es una implementación de librerías en tres dimensiones basadas en OpenGL. Dado que el hardware de los teléfonos Android puede diferir bastante, la librería se encarga de detectar si existe un elemento hardware capaz de acelerar la reproducción en 3D o si por el contrario debe recurrir al software destinado a ello, si no existe hardware 3D.
FreeType	Esta librería está destinada a renderizar fuentes de texto y mapas de bits.
SQLite	Se trata de un gestor de bases de datos relacionales que permite a las diferentes aplicaciones almacenar información estructurada. Es bastante potente y ligero.

Tabla 5. Librerías Android

- Runtime de Android: Android incluye un set de librerías base que proveen la mayor parte de las funcionalidades disponibles en las librerías base del lenguaje de programación Java. Cada aplicación Android corre su propio proceso, con su propia instancia de la máquina virtual Dalvik. Dalvik ha sido escrito de forma que un dispositivo puede correr múltiples máquinas virtuales de forma eficiente. Dalvik ejecuta archivos en el formato Dalvik Executable (.dex), el cual está optimizado para memoria mínima.

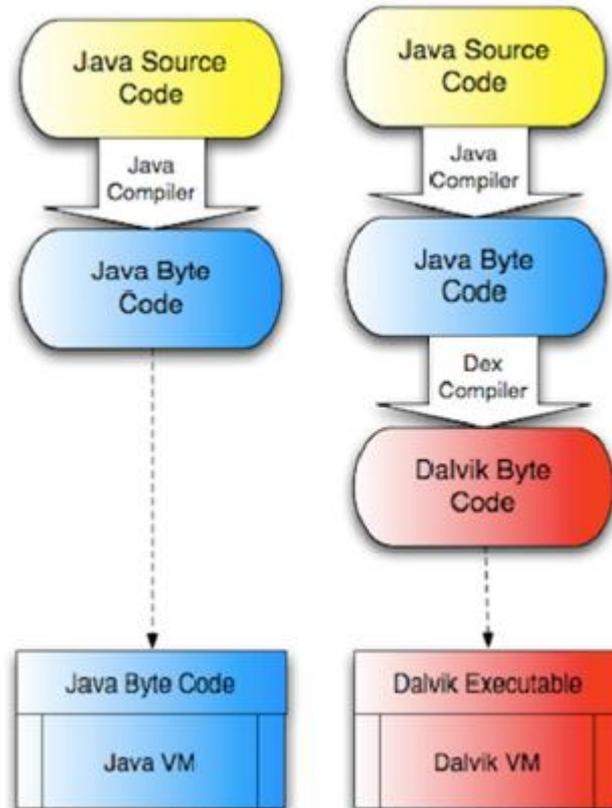


Ilustración 16. Java vs Dalvik

- Núcleo - Linux: Android depende de un Linux versión 2.6 para los servicios base del sistema como seguridad, gestión de memoria, gestión de procesos, pila de red, y modelo de drivers. El núcleo también actúa como una capa de abstracción entre el hardware y el resto de la pila.

3.3.3 Ciclo de vida de las actividades

Como se ha dicho con anterioridad, en Android cada aplicación se ejecuta en su propio proceso. Esto es debido, principalmente, a motivos de seguridad y protección ante errores de memoria, pero también conlleva que el sistema tenga que gestionar la creación y destrucción de una manera ordenada y clara de dichos procesos.

El sistema reacciona ante los cambios de aplicaciones y actividades que realiza el usuario almacenando las actividades que son utilizadas en una pila, de esta manera podemos deshacer nuestras acciones y volver a la actividad anterior sin más que pulsar uno de los cuatro botones que tienen todos los teléfonos Android, la tecla de volver. En general el ciclo de vida de una actividad [25] se muestra en la Ilustración 17 y se compone de los siguientes estados:

- Activo: La actividad se encuentra en este estado si está visible en la pantalla (estará en la parte superior de la pila). La actividad tiene, por tanto, el foco para las acciones del usuario.
- En pausa: Una actividad que se encuentre visible pero que haya perdido el foco se encontrará en este estado. Para que esto ocurra, dicha actividad tiene que estar parcialmente oculta por otra. Las actividades en estado pausado se encuentran activas pero podrían ser terminadas por el sistema en caso de que el mismo se encontrase con muy poca memoria disponible.
- Parado: La actividad se encontrará parada si está por completo oculta de la pantalla. A pesar de ello, guardará toda la información de la que estaba haciendo uso, pero al no ser visible es muy probable que el sistema la cierre cuando se necesite memoria.



Capítulo 4

Análisis y diseño

*"Nada os pertenece en propiedad más que
vuestros sueños."*

Friedrich Nietzsche (1844-1900)

4.1 Introducción

El objetivo de esta sección es explicar la metodología de desarrollo seguida a lo largo del proyecto y la especificación de la funcionalidad a través de los casos de uso y los requisitos. Además, se comentará el diseño del sistema mediante su arquitectura y su diagrama de clases.

4.2 Metodología de desarrollo

Durante la creación de este proyecto se ha seguido como metodología de desarrollo software una versión adaptada de Métrica 3. Los procesos de la estructura principal de Métrica Versión 3 son los siguientes:

- Planificación de Sistemas de Información.
- Desarrollo de Sistemas de Información.
- Mantenimiento de Sistemas de Información.

Métrica v.3 ha sido concebida para abarcar el desarrollo completo de Sistemas de Información sea cual sea su complejidad y magnitud, por lo cual su estructura responde a desarrollos máximos. La ventaja de esta metodología es que puede adaptarse y dimensionarse en cada momento de acuerdo a las características particulares de un proyecto. Es por ello que para el desarrollo de nuestro sistema nos hemos centrado únicamente en los dos primeros procesos, descartando la fase de mantenimiento de Sistemas de Información porque entendemos esta fase como un proceso a realizar en el futuro.

4.3 Casos de Uso

En este apartado se expondrán los casos de uso de nuestro sistema. Un caso de uso representa un uso típico del sistema que permite identificar y definir los requisitos que una aplicación debe cumplir así como la interacción que existe entre el usuario y el sistema. Es una técnica muy utilizada para mostrar la futura capacidad del sistema de una forma simple e intuitiva.

4.3.1 Casos de uso de Gestión de Usuarios

En la siguiente ilustración se muestran los casos de usos de la aplicación relacionados con la gestión de usuarios y sesiones.

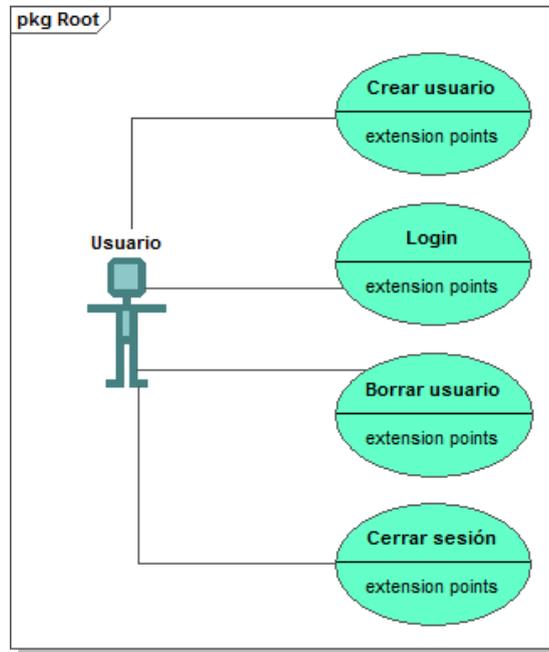


Ilustración 18. Casos de uso de Gestión de Usuarios

Identificador	CU-01
Nombre	Crear usuario
Actores	Usuario sin registrar
Objetivos	El usuario crea un perfil y se guarda la información del usuario.
Precondiciones	El usuario debe acceder a la aplicación y pulsar el botón de “Nuevo Usuario”.
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario introduce el nombre de usuario y la contraseña (dos veces) y pulsa el botón “Siguiente”. 2. Se envían los datos al servidor junto con un SPL y un tiempo por defecto. 3. Se comprueba que no exista un usuario con ese nombre de usuario. 4. Se almacena el nuevo usuario en la base de datos del servidor. 5. Se envía una respuesta afirmativa al dispositivo móvil. 6. Se guarda en la base de datos del móvil. 7. Se muestra un mensaje por pantalla informando al usuario que su perfil ha sido creado correctamente.
Flujo Alternativo	<ol style="list-style-type: none"> 1. El usuario pulsa el botón “Atrás” y se muestra la pantalla de login.
Flujo Alternativo	<ol style="list-style-type: none"> 4. Se comprueba que no exista un usuario con ese nombre de usuario, pero previamente ya ha sido creado. 5. Se envía una respuesta de error al dispositivo móvil.

	6. Se muestra un mensaje por pantalla informando al usuario que su perfil ha sido creado correctamente.
Postcondiciones	El usuario ha sido registrado en el sistema. Se crea la jerarquía de carpetas de usuario en el dispositivo móvil.

Tabla 6. CU-01

Identificador	CU-02
Nombre	Login
Actores	Usuario registrado y sin registrar
Objetivos	Identificar al usuario y permitir el acceso al sistema.
Precondiciones	El usuario debe acceder al sistema.
Flujo Normal	<ol style="list-style-type: none"> 1. Se muestra al usuario una pantalla para que introduzca su usuario y contraseña. 2. El usuario escribe los datos y pulsa el botón “Entrar”. 3. Se envían los datos al servidor. 4. Se comprueba que exista un usuario con ese nombre de usuario y contraseña. 5. Se envía una respuesta afirmativa al dispositivo móvil. 6. Se muestra un mensaje por pantalla informando al usuario que el login es correcto.
Flujo Alternativo	<ol style="list-style-type: none"> 3. No se pueden enviar los datos al servidor porque se produce un fallo en la conexión. 4. Se muestra un mensaje por pantalla informando al usuario que no es posible realizar la conexión.
Flujo Alternativo	<ol style="list-style-type: none"> 7. Se comprueba el usuario y contraseña y no coincide. 8. Se envía una respuesta de error al dispositivo móvil. 9. Se muestra un mensaje por pantalla informando al usuario que no existe ese usuario y contraseña en el sistema.
Postcondiciones	Se crea la jerarquía de carpetas de usuario en el dispositivo móvil, si no existe. Se accede a la pantalla principal.

Tabla 7. CU-02

Identificador	CU-03
Nombre	Borrar usuario.
Actores	Usuario registrado.
Objetivos	Eliminar el usuario del sistema.
Precondiciones	El usuario debe haber accedido a la pantalla de configuración.
Flujo Normal	<ol style="list-style-type: none"> 1. Se muestra un mensaje por pantalla preguntando al usuario la confirmación de la acción. 2. El usuario pulsa el botón “Sí”. 3. Se envía al servidor el nombre de usuario 4. El servidor busca el nombre de usuario en la base de datos y elimina su registro y su jerarquía de carpetas. 5. El servidor envía una respuesta afirmativa. 6. El dispositivo móvil busca el nombre de usuario en la base de datos y elimina su registro y su jerarquía de carpetas.
Flujo Alternativo	<ol style="list-style-type: none"> 2. El usuario pulsa el botón “No”. 3. Se muestra la pantalla de configuración.

Postcondiciones	Se muestra la pantalla de login.
------------------------	----------------------------------

Tabla 8. CU-03

Identificador	CU-04
Nombre	Cerrar sesión.
Actores	Usuario registrado.
Objetivos	Cerrar sesión de usuario y salir del sistema.
Precondiciones	El usuario debe acceder al Menú de la pantalla principal y pulsar sobre el botón “Cerrar sesión”.
Flujo Normal	<ol style="list-style-type: none"> 1. Se muestra un mensaje por pantalla preguntando al usuario la confirmación de la acción. 2. El usuario pulsa el botón “Sí”.
Flujo Alternativo	<ol style="list-style-type: none"> 2. El usuario pulsa el botón “No”. 3. Se muestra la pantalla principal.
Postcondiciones	Se muestra la pantalla de Login.

Tabla 9. CU-04

4.3.2 Casos de uso de Gestión de Configuración de los valores de grabación

En la ilustración de más abajo se muestran los casos de usos relacionados con la gestión de los valores de SPL y tiempo, utilizados en la grabación de somniloquias.

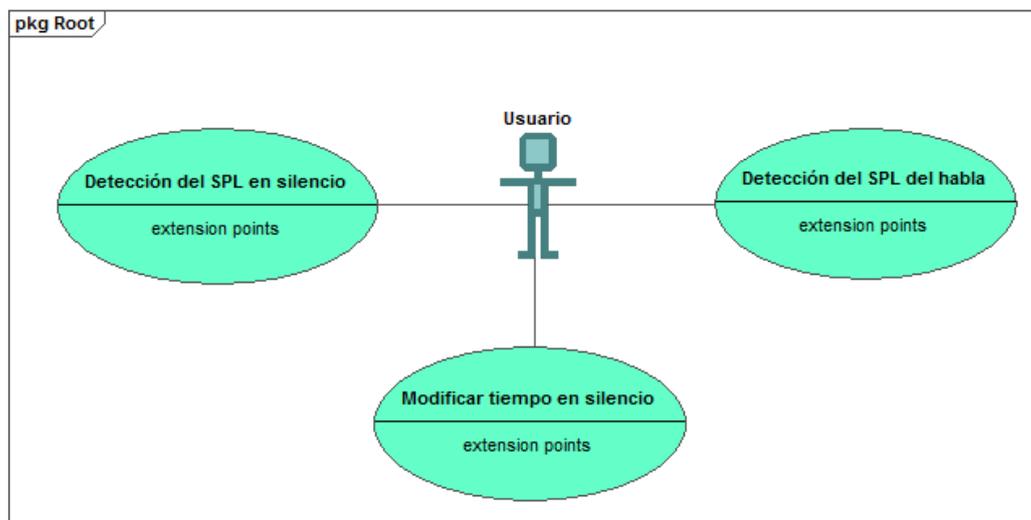


Ilustración 19. Casos de uso de configuración de valores de grabación

Identificador	CU-05
Nombre	Detección del SPL en silencio
Actores	Usuario registrado
Objetivos	Calcular el valor del SPL cuando no hay ruido ambiental
Precondiciones	El usuario debe pulsar el botón “OK” tras haber creado su perfil o

	acceder a la pantalla de configuración.
Flujo Normal	<ol style="list-style-type: none"> 1. Se muestra al usuario la pantalla de detección en silencio con un texto informativo. 2. El usuario pulsa el botón de detección. 3. El usuario permanece callado para que el sistema pueda realizar los cálculos del SPL en silencio. 4. El usuario pulsa el botón “→”.
Flujo Alternativo	<ol style="list-style-type: none"> 4. El usuario pulsa de nuevo el botón de detección. 5. El usuario permanece callado para que el sistema pueda realizar los cálculos del SPL en silencio. 6. El usuario pulsa el botón “→”.
Postcondiciones	Se guarda el valor del SPL internamente para usos posteriores.

Tabla 10. CU-05

Identificador	CU-06
Nombre	Detección del SPL del habla
Actores	Usuario registrado
Objetivos	Calcular el valor del SPL cuando el usuario está hablando.
Precondiciones	El usuario debe pulsar el botón “→” en la pantalla de detección del SPL en silencio.
Flujo Normal	<ol style="list-style-type: none"> 1. Se muestra al usuario la pantalla de detección del habla con un texto informativo. 2. El usuario pulsa el botón de detección. 3. El usuario habla durante un tiempo determinado para que el sistema pueda realizar los cálculos del SPL con su voz. Simultáneamente el sistema graba un fichero de audio para el posterior análisis de la emoción neutra del usuario. 4. El sistema envía el SPL calculado al servidor junto con el nombre de usuario. 5. El servidor busca el registro del usuario y lo actualiza. 6. El servidor envía una respuesta afirmativa. 7. El sistema envía el fichero de audio al servidor junto con el nombre de usuario. 8. El servidor realiza un análisis del archivo, obtiene sus valores prosódicos y los almacena. 9. El servidor envía una respuesta afirmativa. 10. El usuario pulsa el botón “→”.
Flujo Alternativo	10. El usuario pulsa de nuevo el botón de detección y se realiza el flujo normal de nuevo.
Postcondiciones	<p>El SPL se almacena en la base de datos del servidor y del dispositivo móvil.</p> <p>Se muestra la pantalla principal al usuario.</p>

Tabla 11. CU-06

Identificador	CU-07
Nombre	Modificar tiempo en silencio
Actores	Usuario registrado.
Objetivos	Modificar el valor del tiempo de silencio entre archivos.
Precondiciones	El usuario debe haber accedido a la pantalla de configuración.

Flujo Normal	<ol style="list-style-type: none"> 1. El usuario escribe el tiempo en el cuadro de texto correspondiente. 2. El usuario pulsa el botón de guardar. 3. El sistema comprueba el formato del tiempo introducido. 4. Si el formato es correcto establece una conexión con el servidor y le envía el valor junto con el nombre de usuario 5. El servidor busca el registro del usuario y lo actualiza. 6. El servidor envía una respuesta afirmativa.
Flujo Alternativo	<ol style="list-style-type: none"> 3. Si el formato es incorrecto se muestra un mensaje por pantalla informando al usuario de que el formato no es correcto. 4. El usuario pulsa el botón OK. 5. Se muestra la pantalla de configuración.
Postcondiciones	El SPL se almacena en la base de datos del servidor y del dispositivo móvil.

Tabla 12. CU-07

4.3.3 Casos de uso de la funcionalidad principal

En la siguiente ilustración se muestran los casos de usos de la aplicación relacionados con la principal funcionalidad del sistema.

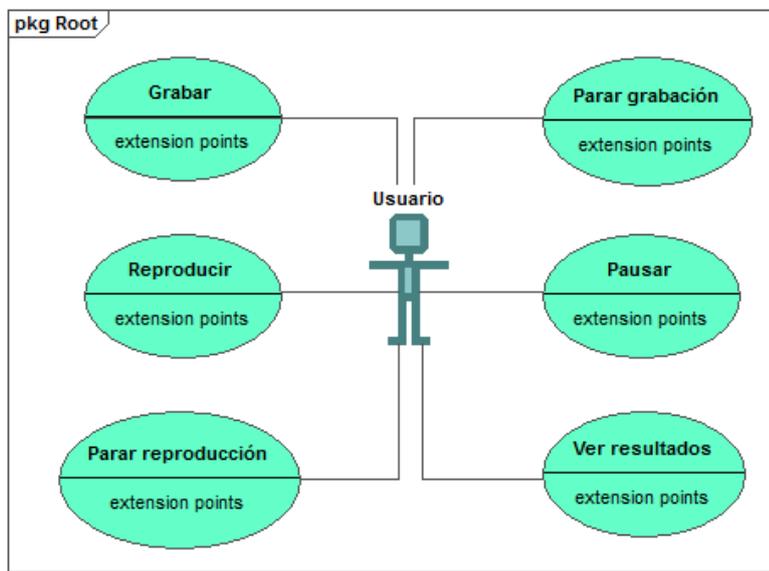


Ilustración 20. Casos de uso de la funcionalidad principal

Identificador	CU-08
Nombre	Grabar
Actores	Usuario registrado.
Objetivos	Grabar las somniaquias del usuario.
Precondiciones	El usuario debe acceder a la pestaña de grabación de la pantalla principal y pulsar el botón de grabación.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema inicia el proceso de grabación de somniaquias. 2. La aplicación crear archivos de audio tras detectar sonidos que superen el umbral de SPL del usuario.

	<ol style="list-style-type: none"> 3. Para cada somniaquia: <ol style="list-style-type: none"> a. Se establece una comunicación el servidor y se envía el fichero. b. El servidor lo procesa y elimina los ruidos c. El servidor envía el fichero post-procesado al dispositivo móvil. d. El dispositivo móvil realiza la petición para la obtención de resultados. e. El servidor realiza un análisis acústico y una transcripción a texto. f. El servidor envía los resultados al dispositivo móvil.
Postcondiciones	<p>Se almacenan los archivos en sus correspondientes rutas del servidor y del dispositivo móvil.</p> <p>Se desactiva el botón de reproducción.</p>

Tabla 13. CU-08

Identificador	CU-09
Nombre	Parar grabación.
Actores	Usuario registrado.
Objetivos	Parar la grabación de las somniaquias del usuario.
Precondiciones	El sistema debe haber iniciado el proceso de grabación.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema termina la grabación de las somniaquias. 2. El sistema almacena el archivo de grabación en curso. 3. Se comprueba si hay somniaquias sin enviar y para cada una de ellas: <ol style="list-style-type: none"> a. Se establece una comunicación el servidor y se envía el fichero. b. El servidor lo procesa y elimina los ruidos c. El servidor envía el fichero post-procesado al dispositivo móvil. d. El dispositivo móvil realiza la petición para la obtención de resultados. e. El servidor realiza un análisis acústico y una transcripción a texto. f. El servidor envía los resultados al dispositivo móvil.
Postcondiciones	<p>Finaliza el proceso de grabación.</p> <p>Se almacenan los archivos en sus correspondientes rutas del servidor y del dispositivo móvil.</p> <p>Se activa el botón de reproducción.</p>

Tabla 14. CU-09

Identificador	CU-10
Nombre	Reproducir
Actores	Usuario registrado.
Objetivos	Reproducir un archivo de audio propio.
Precondiciones	El usuario debe haber accedido a la pestaña de grabación de la pantalla principal y haber pulsado el botón de reproducción.
Flujo Normal	<ol style="list-style-type: none"> 1. El sistema muestra al usuario un listado de sus carpetas.

	<ol style="list-style-type: none"> 2. El usuario accede a una de las carpetas y selecciona un archivo de audio. 3. El sistema reproduce el archivo permitiendo la pausa del mismo.
Postcondiciones	<p>La reproducción del archivo finaliza al terminar de leer el fichero de audio.</p> <p>El sistema activa el botón de pausa y desactiva el botón de grabación.</p>

Tabla 15. CU-10

Identificador	CU-11
Nombre	Pausar
Actores	Usuario registrado.
Objetivos	Pausar un archivo de audio propio.
Precondiciones	El sistema debe estar reproduciendo un archivo de audio.
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario pulsa el botón de pausar. 2. La reproducción se detiene.
Postcondiciones	El sistema activa el botón de reproducción.

Tabla 16. CU-11

Identificador	CU-12
Nombre	Parar reproducción
Actores	Usuario registrado.
Objetivos	Pausar un archivo de audio propio.
Precondiciones	El sistema debe estar reproduciendo un archivo de audio.
Flujo Normal	<ol style="list-style-type: none"> 1. El usuario pulsa el botón de pausar. 2. La reproducción se detiene.
Postcondiciones	El sistema activa el botón de reproducción y activa el botón de grabación.

Tabla 17. CU-12

Identificador	CU-13
Nombre	Ver resultados
Actores	Usuario registrado.
Objetivos	Mostrar al usuario los resultados calculados para un archivo de audio: frecuencia fundamental, transcripción y análisis de emociones.
Precondiciones	El usuario debe haber accedido a la pestaña de resultados de la pantalla principal.
Flujo Normal	<ol style="list-style-type: none"> 1. Si se trata del primer acceso a esta pantalla, el sistema muestra al usuario un listado de sus archivos de resultados. 2. El usuario selecciona un archivo. 3. Se muestra una pantalla con los resultados del archivo y un botón "Cambiar" para cambiar de archivo.
Flujo Alternativo	<ol style="list-style-type: none"> 1. Si no se trata del primer acceso a esta pantalla, el sistema muestra al usuario los resultados de la selección anterior. 2. El usuario pulsa el botón "Cambiar". 3. El sistema muestra al usuario un listado de sus archivos de resultados. 4. El usuario selecciona un archivo.

	5. Se muestra una pantalla con los resultados del archivo y un botón “Cambiar” para cambiar de archivo.
Postcondiciones	Se muestran al usuario los resultados obtenidos de una somniloquia.

Tabla 18. CU-13

4.3.4 Casos de uso navegación

La siguiente ilustración refleja un esquema con los casos de usos del sistema relacionados a la navegación que el usuario puede realizar en la aplicación.

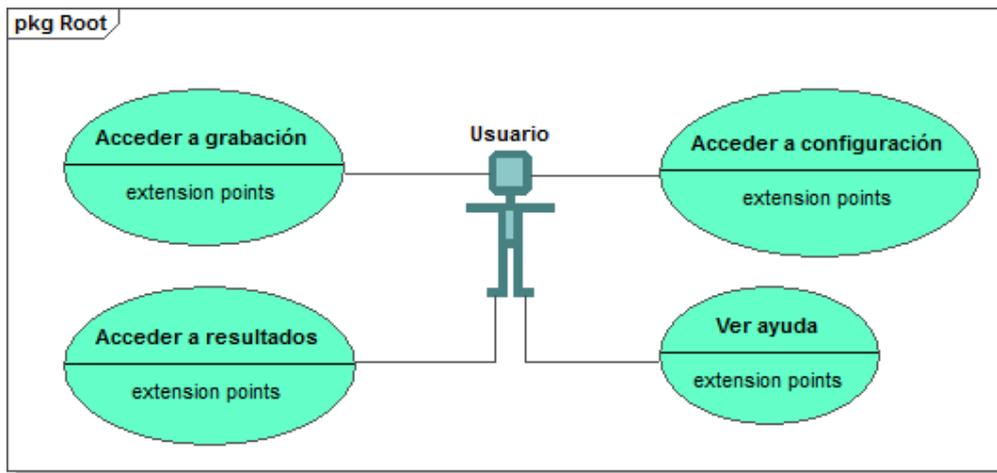


Ilustración 21. Casos de uso de navegación

Identificador	CU-14
Nombre	Acceder a grabación
Actores	Usuario registrado.
Objetivos	Acceder a la pantalla de grabación dentro de la pantalla principal.
Precondiciones	El usuario debe acceder a la pantalla principal.
Flujo Normal	1. El usuario pulsa la pestaña de grabación.
Postcondiciones	Se muestra la pantalla de grabación.

Tabla 19. CU-14

Identificador	CU-15
Nombre	Acceder a configuración
Actores	Usuario registrado.
Objetivos	Acceder a la pantalla de configuración dentro de la pantalla principal.
Precondiciones	El usuario debe acceder a la pantalla principal.
Flujo Normal	1. El usuario pulsa la pestaña de configuración.
Postcondiciones	Se muestra la pantalla de configuración.

Tabla 20. CU-15

Identificador	CU-16
Nombre	Acceder a resultados
Actores	Usuario registrado.
Objetivos	Acceder a la pantalla de resultados dentro de la pantalla principal.
Precondiciones	El usuario debe acceder a la pantalla principal.
Flujo Normal	1. El usuario pulsa la pestaña de resultados.
Postcondiciones	Se muestra la pantalla de resultados.

Tabla 21. CU-16

Identificador	CU-17
Nombre	Ver ayuda.
Actores	Usuario registrado.
Objetivos	Mostrar una pantalla de ayuda al usuario.
Precondiciones	El usuario debe acceder al Menú de la pantalla principal.
Flujo Normal	1. El usuario pulsa sobre el botón “Ayuda”.
Postcondiciones	El sistema muestra la pantalla de Ayuda con un texto informativo.

Tabla 22. CU-17

4.4 Requisitos

4.4.1 Requisitos de Usuario

La plantilla que se utilizará para especificar los requisitos de usuario consta de los siguientes campos que se pasan a describir a continuación:

- **Identificador:** Representa unívocamente a cada requisito. La nomenclatura utilizada será:
 - RUC-AXX: para los requisitos de usuario de capacidad Android.
 - RUC-WXX para los requisitos de usuario de capacidad del servicio web.
 - RUR-AXX para los requisitos de usuario de restricción Android.
 - RUR-WXX para los requisitos de usuario de restricción del servicio web.

XX será un número desde el 01 hasta el 99.

- **Nombre:** Nombre representativo del requisito.
- **Descripción:** Fragmento en el que se explica brevemente en qué consiste el requisito.
- **Fuente:** Hace referencia a la procedencia del requisito.
- **Prioridad:** Grado de urgencia con el que debe cumplirse el requisito. Puede ser Alta, Media o Baja.



Requisitos de Usuario de Capacidad

Android

Identificador	RUC-A01
Nombre	<u>Crear un nuevo usuario</u>
Descripción	Dar de alta en el sistema a un nuevo usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 23. RUC-A01

Identificador	RUC-A02
Nombre	<u>Eliminar a un usuario registrado</u>
Descripción	Eliminar a un usuario del sistema, así como toda la información asociada a su cuenta de usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 24. RUC-A02

Identificador	RUC-A03
Nombre	<u>Modificar usuario registrado</u>
Descripción	Un usuario podrá modificar algunos valores de su cuenta.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 25. RUC-A03

Identificador	RUC-A04
Nombre	<u>Iniciar sesión</u>
Descripción	Iniciar sesión por parte de los usuarios registrados para hacer uso de las funcionalidades del sistema.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 26. RUC-A04

Identificador	RUC-A05
Nombre	<u>Cerrar sesión</u>
Descripción	Cerrar sesión cuando el usuario registrado desee abandonar la aplicación.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 27. RUC-A05



Identificador	RUC-A06
Nombre	<u>Grabar</u>
Descripción	El usuario podrá grabar somniaquias.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 28. RUC-A06

Identificador	RUC-A07
Nombre	<u>Parar grabación</u>
Descripción	El usuario podrá parar la grabación.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 29. RUC-A07

Identificador	RUC-A08
Nombre	<u>Reproducir</u>
Descripción	El usuario podrá reproducir una grabación.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 30. RUC-A08

Identificador	RUC-A09
Nombre	<u>Pausar</u>
Descripción	El usuario podrá pausar la reproducción de un audio.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 31. RUC-A09

Identificador	RUC-A10
Nombre	<u>Parar reproducción</u>
Descripción	El usuario podrá parar la reproducción de un audio.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 32. RUC-A10

Identificador	RUC-A11
Nombre	<u>Acceder grabación</u>
Descripción	El usuario podrá acceder a las grabaciones realizadas
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 33. RUC-A11



Identificador	RUC-A12
Nombre	<u>Navegación</u>
Descripción	El usuario podrá navegar entre las pantallas y pestañas del programa.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 34. RUC-A12

Identificador	RUC-A13
Nombre	<u>Envío de archivos</u>
Descripción	La aplicación móvil podrá enviar los archivos de audio generados a la plataforma web.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 35. RUC-A13

Identificador	RUC-A14
Nombre	<u>Creación de jerarquías</u>
Descripción	La aplicación móvil podrá crear la jerarquía de directorios para cada usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 36. RUC-A14

Identificador	RUC-A15
Nombre	<u>Comunicación</u>
Descripción	El dispositivo móvil podrá comunicarse con la plataforma web.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 37. RUC-A15

Identificador	RUC-A16
Nombre	<u>Almacenamiento de archivos</u>
Descripción	La aplicación móvil almacenará los archivos asociados a cada usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 38. RUC-A16

Identificador	RUC-A17
Nombre	<u>Obtención del nivel de presión sonora de forma automática</u>
Descripción	El sistema calculará el nivel de presión sonora de un usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 39. RUC-A17

Identificador	RUC-A18
Nombre	<u>Ver resultados</u>
Descripción	El usuario podrá visualizar los resultados que devuelve el servidor.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 40. RUC-A18

Identificador	RUC-A19
Nombre	<u>Ayuda</u>
Descripción	La aplicación dispondrá de un tutorial de ayuda para el usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 41. RUC-A19

Identificador	RUC-A20
Nombre	<u>Generación de archivos de audios</u>
Descripción	Se generará un nuevo archivo de audio tras cumplirse un tiempo de espera sin detectar sonidos en una misma grabación.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 42. RUC-A20

Identificador	RUC-A21
Nombre	<u>Recibir archivos</u>
Descripción	La aplicación móvil podrá recibir archivos enviados desde el servidor.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 43. RUC-A21

Identificador	RUC-A22
Nombre	<u>Modificar SPL</u>
Descripción	El usuario podrá configurar tantas veces como quiera el valor del SPL a través de la detección automática.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 44. RUC-A22

Identificador	RUC-A23
Nombre	<u>Modificar tiempo de silencio entre grabaciones</u>
Descripción	El usuario podrá modificar el tiempo de silencio entre grabaciones.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 45. RUC-A24

Servicio Web

Identificador	RUC-W01
Nombre	<u>Insertar a un nuevo usuario</u>
Descripción	Insertar en el servicio web a un nuevo usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 46. RUC-W01

Identificador	RUC-W02
Nombre	<u>Eliminar a un usuario registrado</u>
Descripción	Eliminar a un usuario del sistema, así como toda la información asociada a su cuenta de usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 47. RUC-W02

Identificador	RUC-W03
Nombre	<u>Modificar usuario registrado</u>
Descripción	Modificar los datos de un usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 48. RUC-W03

Identificador	RUC-W04
Nombre	<u>Almacenar archivos</u>
Descripción	La plataforma web podrá almacenar archivos recibidos o generados asociándolos a cada usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 49. RUC-W04

Identificador	RUC-W05
Nombre	<u>Transcripción</u>
Descripción	La plataforma web será capaz de transcribir los archivos de audio a texto.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 50. RUC-W05



Identificador	RUC-W06
Nombre	<u>Análisis de emociones</u>
Descripción	El sistema se encargará de realizar un análisis emocional de los archivos de audio.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 51. RUC-W06

Identificador	RUC-W07
Nombre	<u>Comunicación</u>
Descripción	La plataforma web podrá comunicarse con el dispositivo móvil.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 52. RUC-W07

Identificador	RUC-W08
Nombre	<u>Procesar peticiones</u>
Descripción	El sistema se encargará de procesar las peticiones recibidas.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 53. RUC-W08

Identificador	RUC-W09
Nombre	<u>Enviar ficheros de audio</u>
Descripción	El servicio web podrá enviar audios al dispositivo móvil.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 54. RUC-W09

Identificador	RUC-W10
Nombre	<u>Enviar ficheros de texto</u>
Descripción	El servicio web podrá enviar archivos de texto al dispositivo móvil.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 55. RUC-W10

Identificador	RUC-W11
Nombre	<u>Crear jerarquía</u>
Descripción	El servicio web podrá crear jerarquías de directorios asociadas a un usuario para almacenar archivos.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 56. RUC-W11



Identificador	RUC-W12
Nombre	<u>Eliminar ruidos</u>
Descripción	El servicio web podrá eliminar los ruidos y ronquidos de las somniloquias.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 57. RUC-W12

Identificador	RUC-W13
Nombre	<u>Actualizar SPL</u>
Descripción	El servlet podrá actualizar el valor del SPL.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 58. RUC-W13

Identificador	RUC-W14
Nombre	<u>Actualizar tiempo de silencio entre grabaciones</u>
Descripción	El servlet podrá actualizar el valor del tiempo de silencio entre grabaciones.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 59. RUC-W14

Identificador	RUC-W15
Nombre	<u>Obtener características fundamentales de un audio</u>
Descripción	El servlet podrá realizar análisis acústico y obtener las características fundamentales de un audio.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 60. RUC-W15

Identificador	RUC-W16
Nombre	<u>Insertar características fundamental de un archivo neutro</u>
Descripción	El servlet podrá almacenar las características fundamentales de un archivo neutro.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 61. RUC-W16

Requisitos de Usuario de Restricción

Android

Identificador	RUR-A01
Nombre	<u>Crear una interfaz de usuario sencilla e intuitiva.</u>
Descripción	Crear una interfaz de usuario lo más sencilla e intuitiva posible.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 62. RUR-A01

Identificador	RUR-A02
Nombre	<u>Campos de login</u>
Descripción	Requerir un nombre de usuario único y una contraseña para iniciar sesión en el sistema.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 63. RUR-A02

Identificador	RUR-A03
Nombre	<u>Ocultación de contraseñas</u>
Descripción	La aplicación no mostrará el texto de la contraseña de un usuario por motivos de seguridad.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 64. RUR-A03

Identificador	RUR-A04
Nombre	<u>Enviar contraseñas cifradas</u>
Descripción	Por motivos de seguridad la contraseña se enviará cifrada al servidor.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 65. RUR-A04

Identificador	RUR-A05
Nombre	<u>Grabación inteligente.</u>
Descripción	La aplicación graba de forma inteligente, almacenando sólo aquellos fragmentos en los que el usuario hable.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 66. RUR-A05



Identificador	RUR-A06
Nombre	<u>Campos de crear usuario</u>
Descripción	Se pedirá un nombre de usuario único y dos veces una contraseña para crear un usuario en el sistema.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 67. RUR-A06

Identificador	RUR-A07
Nombre	<u>Valor por defecto el tiempo de espera entre audios</u>
Descripción	El tiempo que la aplicación espera en los momentos de silencio tendrá un valor por defecto de 5 minutos.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 68. RUR-A07

Identificador	RUR-A08
Nombre	<u>Rango de valores del tiempo de espera entre audios</u>
Descripción	El valor mínimo del tiempo en silencio entre audios será 0 segundos y el valor máximo será 60 minutos.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 69. RUR-A08

Identificador	RUR-A09
Nombre	<u>Dispositivo móvil</u>
Descripción	La aplicación se podrá ejecutar en dispositivos móviles con características similares al Samsung Galaxy S (I9000).
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 70. RUR-A09

Identificador	RUR-A10
Nombre	<u>Base de datos</u>
Descripción	La aplicación dispondrá de una base de datos (SQLite) para el almacenamiento de información.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 71. RUR-A10

Identificador	RUR-A11
Nombre	<u>Sistema operativo</u>
Descripción	La aplicación será compatible con el sistema operativo para móviles Android.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 72. RUR-A11

Identificador	RUR-A12
Nombre	<u>Versión de Android</u>
Descripción	La aplicación se podrá ejecutar en cualquier dispositivo móvil con versión de la plataforma Android 2.3.3 (API 10) y superiores.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 73. RUR-A12

Identificador	RUR-A13
Nombre	<u>Protocolo de comunicación</u>
Descripción	Las comunicaciones con el servidor se harán mediante el protocolo HTTP.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 74. RUR-A13

Identificador	RUR-A14
Nombre	<u>Formato de archivos de somniaquias originales</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 75. RUR-A14

Identificador	RUR-A15
Nombre	<u>Formato de archivos de somniaquias procesadas</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 76. RUR-A15

Identificador	RUR-A16
Nombre	<u>Formato de archivos neutros</u>
Descripción	Los archivos de audio tendrán el siguiente formato: Neutro-usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 77. RUR-A16

Identificador	RUR-A17
Nombre	<u>Formato de archivos de resultados</u>
Descripción	Los archivos de texto con los resultados tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-R, seguido de la extensión “.txt”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 78. RUR-A17

Identificador	RUR-A18
Nombre	<u>Formato de jerarquía</u>
Descripción	El formato .de la jerarquía contiene una carpeta por cada usuario y dentro los archivos de audio y texto organizados por tipo.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 79. RUR-A18

Identificador	RUR-A19
Nombre	<u>Idioma</u>
Descripción	La aplicación mostrará los textos en español.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 80. RUR-A19

Identificador	RUR-A20
Nombre	<u>Configuración inicial del valor del nivel de presión sonora</u>
Descripción	Una de las fases de la creación de usuario es la detección del valor del SPL.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 81. RUR-A20

Servicio Web

Identificador	RUR-W01
Nombre	<u>Emociones primarias</u>
Descripción	Las emociones que se clasificarán serán primarias, distinguiendo entre: alegría, enfado, tristeza y miedo.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 82. RUR-W01

Identificador	RUR-W02
Nombre	<u>Base de datos plataforma web</u>
Descripción	Se dispondrá de una base de datos MySQL.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 83. RUR-W02

Identificador	RUR-W03
Nombre	<u>Servidor</u>
Descripción	Se dispondrá de un servidor web Apache Tomcat.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 84. RUR-W03

Identificador	RUR-W04
Nombre	<u>Formato de archivos de somniaquias originales</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 85. RUR-W04

Identificador	RUR-W05
Nombre	<u>Formato de archivos de somniaquias procesadas</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 86. RUR-W05

Identificador	RUR-W06
Nombre	<u>Formato de archivos neutros</u>
Descripción	Los archivos de audio tendrán el siguiente formato: Neutro-usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 87. RUR-W06

Identificador	RUR-W07
Nombre	<u>Formato de archivos de resultados</u>
Descripción	Los archivos de texto con los resultados tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-R, seguido de la extensión “.txt”.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 88. RUR-W07

Identificador	RUR-W08
Nombre	<u>Formato de jerarquía de directorios</u>
Descripción	La jerarquía estará compuesta por cuatro carpetas principales y dentro de ellas se organizan los archivos por usuario.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 89. RUR-W08

Identificador	RUR-W09
Nombre	<u>Transcripción</u>
Descripción	La transcripción se realizará mediante peticiones a un servidor externo que procese el audio y devuelva un resultado. Este resultado se parseará para obtener únicamente el texto que interese.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 90. RUR-W09

Identificador	RUR-W10
Nombre	<u>Resultado del análisis emocional</u>
Descripción	El resultado del análisis de emociones de un archivo se almacenará en un archivo de texto.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 91. RUR-W10

Identificador	RUR-W11
Nombre	<u>Eliminar fragmentos de grabación</u>
Descripción	La aplicación eliminará fragmentos con ruidos como llantos o ronquidos mediante análisis acústico de las somniaquias.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 92. RUR-W11

Identificador	RUR-W12
Nombre	<u>Tipo de información</u>
Descripción	Cada tipo de información que se almacene en el sistema tendrá un formato adecuado.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 93. RUR-W12

Identificador	RUR-W13
Nombre	<u>Comunicación</u>
Descripción	La comunicación con el dispositivo móvil se hará por HTTP.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 94. RUR-W13

Identificador	RUR-W14
Nombre	<u>Frecuencias de la voz</u>
Descripción	El filtrado de audios se hará para las frecuencias de la voz humana.
Fuente	Equipo de desarrollo
Prioridad	Alta

Tabla 95. RUR-W14

4.4.2 Requisitos de Software

La plantilla que se utilizará para especificar los requisitos software consta de los siguientes campos que se pasan a describir a continuación:

- **Identificador:** Representa unívocamente a cada requisito. La nomenclatura utilizada será:
 - RSF-AXX: para los requisitos de software funcionales de Android.
 - RSF-WXX: para los requisitos de software funcionales del servicio web.
 - RSNF-AXX: para los requisitos de software no funcionales de Android.
 - RSNF-WXX: para los requisitos de software no funcionales del servicio web.

XX será un número desde el 01 hasta el 99.

- **Nombre:** Nombre representativo del requisito.
- **Descripción:** Fragmento en el que se explica brevemente en qué consiste el requisito.
- **Fuente:** Hace referencia a la procedencia del requisito.
- **Prioridad:** Grado de urgencia con el que debe cumplirse el requisito. Puede ser Alta, Media o Baja.

Requisitos de Software Funcionales

Android

Identificador	RSF-A01
Nombre	<u>Insertar un nuevo usuario</u>
Descripción	El sistema creará un nuevo usuario y almacenará sus datos: nombre de usuario, contraseña, SPL y tiempo de silencio entre grabaciones.
Fuente	RUC-A01
Prioridad	Alta

Tabla 96 RSF-A01

Identificador	RSF-A02
Nombre	<u>Borrar un usuario</u>
Descripción	El sistema podrá eliminar al usuario de la base de datos, además de toda su información asociada y su jerarquía de directorios.
Fuente	RUC-A02
Prioridad	Alta

Tabla 97 RSF-A02

Identificador	RSF-A03
Nombre	<u>Modificar los datos de usuario</u>
Descripción	El sistema permitirá al usuario modificar el valor del SPL y del tiempo de silencio entre grabaciones.
Fuente	RUC-A03
Prioridad	Alta

Tabla 98 RSF-A03

Identificador	RSF-A04
Nombre	<u>Login</u>
Descripción	El sistema contará con un servicio para que un usuario pueda loguearse a través de su nombre de usuario y contraseña. Para ello el sistema comprobará que los datos introducidos sean correctos antes de dar acceso.
Fuente	RUC-A04
Prioridad	Alta

Tabla 99 RSF-A04

Identificador	RSF-A05
Nombre	<u>Cierre de sesión</u>
Descripción	El sistema permitirá el cierre de sesión de un usuario que se encuentre actualmente logueado.
Fuente	RUC-A05
Prioridad	Alta

Tabla 100 RSF-A05

Identificador	RSF-A06
Nombre	<u>Grabar</u>
Descripción	El sistema será capaz de grabar distintas somniloquias a través de un mecanismo de detección del nivel de presión sonora.
Fuente	RUC-A06
Prioridad	Alta

Tabla 101 RSF- A06

Identificador	RSF-A07
Nombre	<u>Parar grabación</u>
Descripción	El sistema permitirá parar una grabación que previamente haya sido iniciada.
Fuente	RUC-A07
Prioridad	Alta

Tabla 102 RSF- A07

Identificador	RSF-A08
Nombre	<u>Reproducir grabación</u>
Descripción	El sistema proporcionará al usuario la opción de reproducir sus archivos de audio: somniloquias, neutros y post-procesados.
Fuente	RUC-A08
Prioridad	Alta

Tabla 103 RSF- A08

Identificador	RSF-A09
Nombre	<u>Pausar reproducción</u>
Descripción	El sistema proporcionará al usuario la opción de pausar la reproducción actual.
Fuente	RUC-A09
Prioridad	Alta

Tabla 104 RSF- A09

Identificador	RSF-A10
Nombre	<u>Parar reproducción</u>
Descripción	El sistema proporcionará al usuario la opción de parar la reproducción actual.
Fuente	RUC-A10
Prioridad	Alta

Tabla 105 RSF- A10

Identificador	RSF-A11
Nombre	<u>Acceder a una grabación</u>
Descripción	Se podrá acceder a las grabaciones realizadas a través de la interfaz de la aplicación.
Fuente	RUC-A11
Prioridad	Alta

Tabla 106 RSF- A11

Identificador	RSF-A12
Nombre	<u>Navegación</u>
Descripción	La aplicación contará con pantallas y pestañas para que el usuario pueda navegar entre ellas.
Fuente	RUC-A12
Prioridad	Alta

Tabla 107 RSF- A12

Identificador	RSF-A13
Nombre	<u>Envío de archivos</u>
Descripción	El envío de archivos se hará de manera transparente al usuario, produciéndose cada vez que se genere un nuevo audio. Si se ha terminado de grabar las somniaquias y quedan archivos sin enviar, se esperará hasta que se manden todos.
Fuente	RUC-A13
Prioridad	Alta

Tabla 108 RSF-A13

Identificador	RSF-A14
Nombre	<u>Creación de jerarquías</u>
Descripción	El sistema creará de manera transparente al usuario las jerarquías. Este proceso se asocia con la creación de un nuevo usuario.
Fuente	RUC-A14
Prioridad	Alta

Tabla 109 RSF-A14

Identificador	RSF-A15
Nombre	<u>Comunicación</u>
Descripción	La aplicación móvil se comunicará con el servicio web a través de un protocolo basado en peticiones y respuestas.
Fuente	RUC-A15
Prioridad	Alta

Tabla 110 RSF-A15

Identificador	RSF-A16
Nombre	<u>Almacenamiento de audio</u>
Descripción	La aplicación almacenará las somniaquias generadas en el sistema de ficheros del dispositivo.
Fuente	RUC-A16
Prioridad	Alta

Tabla 111 RSF-A16

Identificador	RSF-A17
Nombre	<u>Obtención del nivel de presión sonora de forma automática en silencio</u>
Descripción	El sistema calculará el nivel de presión sonora del ruido ambiental en el que se encuentra el usuario, para tomarla como referencia en el cálculo del valor del SPL.
Fuente	RUC-A17
Prioridad	Alta

Tabla 112 RSF-A17

Identificador	RSF-A18
Nombre	<u>Obtención del nivel de presión sonora con voz.</u>
Descripción	El sistema calculará el nivel de presión sonora de la voz del usuario, para tomarla como referencia en el cálculo del valor del SPL.
Fuente	RUC-A17
Prioridad	Alta

Tabla 113 RSF-A18

Identificador	RSF-A19
Nombre	<u>Cálculo del SPL</u>
Descripción	El sistema calculará el SPL a través de un algoritmo que tendrá como valores de entrada los resultados de la detección en silencio y con voz.
Fuente	RUC-A17
Prioridad	Alta

Tabla 114 RSF-A19

Identificador	RSF-A20
Nombre	<u>Ver resultados</u>
Descripción	La aplicación dispondrá de una opción para visualizar los resultados que devuelve el servidor.
Fuente	RUC-18
Prioridad	Alta

Tabla 115 RSF-A20

Identificador	RSF-A21
Nombre	<u>Ver ayuda</u>
Descripción	La aplicación dispondrá de un tutorial de ayuda para el usuario. Este manual explica la funcionalidad principal del sistema.
Fuente	RUC-19
Prioridad	Alta

Tabla 116 RSF-A21

Identificador	RSF-A22
Nombre	<u>Generación de archivos de audios</u>
Descripción	La aplicación generará un nuevo archivo de audio tras cumplirse un tiempo de espera sin detectar sonidos en una misma grabación.
Fuente	RUC-A20
Prioridad	Alta

Tabla 117 RSF-A22

Identificador	RSF-A23
Nombre	<u>Recibir archivos de audio</u>
Descripción	La aplicación móvil podrá recibir archivos de audio procesados en la plataforma web, además los almacena en la jerarquía de ficheros del usuario.
Fuente	RUC-A21, RUC-A16
Prioridad	Alta

Tabla 118. RSF-A23

Identificador	RSF-A24
Nombre	<u>Recibir archivos de texto</u>
Descripción	La aplicación móvil podrá recibir archivos de texto con los resultados generados en la plataforma web, además los almacena en la jerarquía de ficheros del usuario.
Fuente	RUC-A21, RUC-A16
Prioridad	Alta

Tabla 119. RSF-A24

Identificador	RSF-A25
Nombre	<u>Modificar SPL</u>
Descripción	El sistema ofrecerá la posibilidad de calcular el SPL tantas veces como desee el usuario. El nuevo valor se actualizará en la base de datos.
Fuente	RUC-A22
Prioridad	Alta

Tabla 120. RSF-A25

Identificador	RSF-A26
Nombre	<u>Modificar tiempo de silencio entre grabaciones</u>
Descripción	El sistema ofrecerá la posibilidad de modificar el tiempo de silencio entre grabaciones. El nuevo valor se actualizará en la base de datos.
Fuente	RUC-A23
Prioridad	Alta

Tabla 121. RSF-A26

Servicio Web

Identificador	RSF-W01
Nombre	<u>Insertar un nuevo usuario</u>
Descripción	El servlet insertará en la base de datos un nuevo usuario con estos campos: <ul style="list-style-type: none">• Nombre de usuario.• Contraseña.• SPL.• Tiempo de silencio entre grabaciones.
Fuente	RUC-W01
Prioridad	Alta

Tabla 122 RSF-W01

Identificador	RSF-W02
Nombre	<u>Dar de baja a un usuario registrado</u>
Descripción	El servlet eliminará a un usuario previamente creado de la base de datos, así como su jerarquía.
Fuente	RUC-W02
Prioridad	Alta

Tabla 123 RSF-W02

Identificador	RSF-W03
Nombre	<u>Modificar usuario</u>
Descripción	El servidor podrá actualizar los campos de un usuario en la base de datos.
Fuente	RUC-W03
Prioridad	Alta

Tabla 124 RSF-W02

Identificador	RSF-W04
Nombre	<u>Almacenamiento de audio</u>
Descripción	El sistema guardará automáticamente, en la jerarquía del usuario, todos sus archivos de audio recibidos o procesados. Cada archivo estará asociado únicamente a un usuario.
Fuente	RUC-W04
Prioridad	Alta

Tabla 125 RSF-W02

Identificador	RSF-W05
Nombre	<u>Almacenamiento de texto</u>
Descripción	El sistema guardará automáticamente, en la jerarquía del usuario, todos sus archivos de resultados generados. Cada archivo estará asociado únicamente a un usuario.
Fuente	RUC-W04
Prioridad	Alta

Tabla 126 RSF-W05

Identificador	RSF-W06
Nombre	<u>Transcripción</u>
Descripción	La aplicación será capaz de transcribir a texto un archivo de audio almacenado. Para ello es necesario que el idioma empleado sea el castellano.
Fuente	RUC-W05
Prioridad	Alta

Tabla 127 RSF-W06

Identificador	RSF-W07
Nombre	<u>Análisis de emociones</u>
Descripción	El servicio web contará con un módulo encargado del análisis emocional que consistirá en la detección de las siguientes emociones básicas: alegría, enfado, tristeza y miedo.
Fuente	RUC-W06
Prioridad	Alta

Tabla 128. RSF-W07

Identificador	RSF-W08
Nombre	<u>Recibir peticiones</u>
Descripción	El servlet podrá recibir peticiones del dispositivo móvil.
Fuente	RUC-W07
Prioridad	Alta

Tabla 129. RSF-W08

Identificador	RSF-W09
Nombre	<u>Procesar peticiones en servlet simple</u>
Descripción	El servlet simple procesará las peticiones a través de la lectura del valor de un parámetro.
Fuente	RUC-W08
Prioridad	Alta

Tabla 130. RSF-W09

Identificador	RSF-W10
Nombre	<u>Procesar peticiones en servlet fichero</u>
Descripción	El servlet fichero procesará una petición a través de la lectura de su cabecera, distinguiendo entre “neutro”, “archivo” y “procesado”.
Fuente	RUC-W08
Prioridad	Alta

Tabla 131. RSF-W10

Identificador	RSF-W11
Nombre	<u>Enviar ficheros de audio</u>
Descripción	El servlet podrá mandar ficheros de audio al dispositivo móvil. El envío se realizará en la respuesta de una petición.
Fuente	RUC-W09
Prioridad	Alta

Tabla 132. RSF-W11

Identificador	RSF-W12
Nombre	<u>Enviar ficheros de texto</u>
Descripción	El servlet podrá mandar ficheros de texto al dispositivo móvil. El envío se realizará en la respuesta de una petición.
Fuente	RUC-W10
Prioridad	Alta

Tabla 133. RSF-W12

Identificador	RSF- W13
Nombre	<u>Enviar cadenas de texto</u>
Descripción	El servlet podrá mandar cadenas de texto al dispositivo móvil. El envío se realizará en la respuesta de una petición.
Fuente	RUC-W07
Prioridad	Alta

Tabla 134. RSF-W13

Identificador	RSF-W14
Nombre	<u>Crear jerarquía</u>
Descripción	El servicio web podrá crear jerarquías de directorios asociadas a un usuario para almacenar archivos. La creación de la jerarquía es una fase de la creación de nuevos usuarios en el sistema.
Fuente	RUC-W11
Prioridad	Alta

Tabla 135. RSF-W14

Identificador	RSF-W15
Nombre	<u>Eliminar ruidos</u>
Descripción	El servicio web podrá procesar somniaquias y eliminar los ruidos y ronquidos, a través de un análisis acústico de las mismas.
Fuente	RUC-W12
Prioridad	Alta

Tabla 136. RSF-W15

Identificador	RSF-W16
Nombre	<u>Modificar SPL</u>
Descripción	El servlet podrá actualizar el valor del SPL de un usuario en la base de datos.
Fuente	RUC-W13
Prioridad	Alta

Tabla 137. RSF-W16

Identificador	RSF-W17
Nombre	<u>Modificar tiempo de silencio entre grabaciones</u>
Descripción	El servlet podrá actualizar el valor del tiempo de silencio entre grabaciones asociado a un usuario en la base de datos.
Fuente	RUC-W16
Prioridad	Alta

Tabla 138. RSF-W17

Identificador	RSF-W18
Nombre	<u>Obtener características fundamentales de un audio</u>
Descripción	El servlet podrá realizar análisis acústicos de los ficheros con emoción neutra y los archivos procesados para obtener sus características fundamentales.
Fuente	RUC-W17
Prioridad	Alta

Tabla 139. RSF-W18

Identificador	RSF-W19
Nombre	<u>Insertar características fundamental de un archivo neutro</u>
Descripción	El servlet podrá almacenar en la base de datos las características fundamentales de un archivo neutro.
Fuente	RUC-W18
Prioridad	Alta

Tabla 140. RSF-W19

Requisitos de Software No Funcionales

Android

Identificador	RSNF-A01
Nombre	<u>Crear una interfaz de usuario sencilla e intuitiva.</u>
Descripción	La interfaz de usuario del dispositivo móvil se adapta a las características de la pantalla. Será sencilla y poco sobrecargada, ofreciendo tanto comodidad como eficiencia. Utilizará colores contrastados para facilitar la lectura.
Fuente	RUR-A01
Prioridad	Alta

Tabla 141. RSNF-A01

Identificador	RSNF-A02
Nombre	<u>Pantalla de login</u>
Descripción	El sistema contará con una pantalla destinada al login que muestre los siguientes campos a rellenar: <ul style="list-style-type: none"> • Nombre de usuario. • Contraseña
Fuente	RUR-A02
Prioridad	Alta

Tabla 142 RSNF-A02

Identificador	RSNF-A03
Nombre	<u>Formato campos de login</u>
Descripción	<ul style="list-style-type: none"> • Nombre de usuario: Conjunto de caracteres alfanuméricos que definen unívocamente al usuario. • Contraseña: Conjunto de caracteres alfanuméricos.
Fuente	RUR-A02
Prioridad	Alta

Tabla 143 RSNF-A03

Identificador	RSNF-A04
Nombre	<u>Ocultación de contraseñas</u>
Descripción	La aplicación no mostrará el texto de la contraseña, en vez de eso aparecerá un asterisco “*” por cada letra.
Fuente	RUR-A03
Prioridad	Alta

Tabla 144. RSNF-A04

Identificador	RSNF-A05
Nombre	<u>Contraseñas cifradas</u>
Descripción	Por motivos de seguridad la contraseña se enviará cifrada al servidor. La contraseña se cifrará con el algoritmo DES.
Fuente	RUR-A04
Prioridad	Alta

Tabla 145. RSNF-A05

Identificador	RSNF-A06
Nombre	<u>Grabación inteligente mediante SPL</u>
Descripción	La aplicación móvil detectará cuando el usuario hable mediante el cálculo del nivel de presión sonora (SPL). El sistema almacenará aquellos fragmentos de audio que hayan sido detectados tras superar el SPL configurado por el usuario.
Fuente	RUR-A05
Prioridad	Alta

Tabla 146. RSNF-A06

Identificador	RSNF-A07
Nombre	<u>Pantalla para crear un nuevo usuario</u>
Descripción	La interfaz de la pantalla de creación de un nuevo usuario mostrará los siguientes campos: <ul style="list-style-type: none"> • Nombre de usuario. • Contraseña. • Repetir contraseña.
Fuente	RUR-A06
Prioridad	Alta

Tabla 147 RSNF-A07

Identificador	RSNF-A08
Nombre	<u>Insertar un nuevo usuario</u>
Descripción	El sistema impedirá la creación de cuentas con nombre de usuario repetido. Para ello comprobará en el servidor que el nombre de usuario no exista.
Fuente	RUC-A01
Prioridad	Alta

Tabla 148 RSNF-A08

Identificador	RSNF-A09
Nombre	<u>Formato campos de nuevo usuario</u>
Descripción	<ul style="list-style-type: none"> • Nombre de usuario: Conjunto de caracteres alfanuméricos que definen unívocamente al usuario. • Contraseña: Conjunto de caracteres alfanuméricos. • Repetir contraseña: mismo formato que contraseña.
Fuente	RUR-A06
Prioridad	Alta

Tabla 149 RSNF-A09

Identificador	RSNF-A10
Nombre	<u>Valor por defecto el tiempo de espera entre audios</u>
Descripción	El tiempo que la aplicación espera en los momentos de silencio tendrá un valor por defecto de 5 minutos.
Fuente	RUR-A07
Prioridad	Alta

Tabla 150. RSNF-A10

Identificador	RSNF-A11
Nombre	<u>Rango de valores del tiempo de espera entre audios</u>
Descripción	El valor mínimo del tiempo en silencio entre audios será 0 segundos y el valor máximo será 60 minutos.
Fuente	RUR-A08
Prioridad	Alta

Tabla 151. RSNF-A11

Identificador	RSNF-A12
Nombre	<u>Cierre de sesión</u>
Descripción	Se saldrá del sistema pulsando el botón “Cerrar Sesión” dentro del menú de Opciones. Para eso el usuario registrado tendrá que tener la sesión activa.
Fuente	RUC-A05
Prioridad	Alta

Tabla 152 RSNF-A12

Identificador	RSNF-A13
Nombre	<u>Iniciar grabación</u>
Descripción	Una grabación comienza cuando el usuario pulsa el botón “record” en la pestaña de “Grabar y Reproducir”. Cuando el usuario pulse el botón se creará una carpeta con el nombre correspondiente al día actual, en caso de que no estuviese creada con anterioridad.
Fuente	RUR-A05
Prioridad	Alta

Tabla 153 RSNF-A13

Identificador	RSNF-A14
Nombre	<u>Parar grabación</u>
Descripción	El sistema permitirá parar la grabación dentro de la pestaña “Grabar y Reproducir” pulsando sobre el botón “stop”. Previamente se tiene que haber iniciado la grabación.
Fuente	RUC-A07
Prioridad	Alta

Tabla 154 RSNF-A14

Identificador	RSNF-A15
Nombre	<u>Reproducir grabación</u>
Descripción	El sistema permitirá reproducir una grabación dentro de la pestaña “Grabar y Reproducir” pulsando sobre el botón “play”.
Fuente	RUC-A08
Prioridad	Alta

Tabla 155 RSNF-A15

Identificador	RSNF-A16
Nombre	<u>Pausar reproducción</u>
Descripción	El sistema permitirá pausar la reproducción de una grabación dentro de la pestaña “Grabar y Reproducir” pulsando sobre el botón “pause”. Previamente se tiene que haber iniciado la reproducción.
Fuente	RUC-A09
Prioridad	Alta

Tabla 156 RSNF-A16

Identificador	RSNF-A17
Nombre	<u>Parar reproducción</u>
Descripción	El sistema permitirá parar una reproducción dentro de la pestaña “Grabar y Reproducir” pulsando sobre el botón “stop”. Previamente se tiene que haber iniciado la reproducción.
Fuente	RUC-A10
Prioridad	Alta

Tabla 157 RSNF-A17

Identificador	RSNF-A18
Nombre	<u>Pantalla de detección del nivel de presión sonora en silencio.</u>
Descripción	Se proporcionará una pantalla para la detección automática del nivel de presión sonora en silencio.
Fuente	RUR-A20, RUC-A17
Prioridad	Alta

Tabla 158 RSNF-A18

Identificador	RSNF-A19
Nombre	<u>Pantalla de detección del nivel de presión sonora con voz.</u>
Descripción	Se proporcionará una pantalla para la detección automática del nivel de presión sonora con voz.
Fuente	RUR-A20, RUC-A17
Prioridad	Alta

Tabla 159 RSNF-A19

Identificador	RSNF-A20
Nombre	<u>Pantalla de ayuda</u>
Descripción	Se contará con una pantalla de ayuda donde se explican las principales funciones de la aplicación.
Fuente	RUC-A19
Prioridad	Alta

Tabla 160 RSNF-A20

Identificador	RSNF-A20
Nombre	<u>Pestaña de configuración</u>
Descripción	El sistema contará con una pestaña para la configuración de datos del usuario. Esta pestaña tendrá tres funcionalidades: <ul style="list-style-type: none"> • Detección automática del SPL. • Tiempo en silencio. • Borrar usuario.
Fuente	RUC-A23
Prioridad	Alta

Tabla 161. RSNF-A20

Identificador	RSNF-A21
Nombre	<u>Eliminar un usuario registrado</u>
Descripción	En el apartado de Configuración existirá un botón para eliminar un usuario. Cuando el usuario lo pulse, y tras aceptar confirmación, se elimina de la base de datos así como su jerarquía de ficheros.
Fuente	RUC-A02
Prioridad	Alta

Tabla 162 RSNF-A21

Identificador	RSNF-A22
Nombre	<u>Tiempo de silencio entre audios</u>
Descripción	El tiempo de silencio entre audio tiene que seguir el siguiente formato: mm:ss. Además existirá el botón “Guardar” que permite almacenar en la base de datos el tiempo de silencio.
Fuente	RUR-A07, RUR-A08
Prioridad	Alta

Tabla 163. RSNF-A22

Identificador	RSNF-A23
Nombre	<u>Pestaña de resultados</u>
Descripción	El sistema contará con una pestaña para visualizar los resultados obtenidos en el servidor. Tendrá un botón para cambiar el archivo que se desee mostrar.
Fuente	RUC-A18
Prioridad	Alta

Tabla 164. RSNF-A23

Identificador	RSNF-A23
Nombre	<u>Dispositivo móvil</u>
Descripción	La aplicación se podrá ejecutar en dispositivos móviles con características similares al Samsung Galaxy S (I9000).
Fuente	RUR-A09
Prioridad	Alta

Tabla 165. RSNF-A23

Identificador	RSNF-A24
Nombre	<u>Base de datos</u>
Descripción	La aplicación dispondrá de una base de datos (SQLite) para el almacenamiento de información.
Fuente	RUR-A10
Prioridad	Alta

Tabla 166. RSNF-A24

Identificador	RSNF-A25
Nombre	<u>Sistema operativo</u>
Descripción	La aplicación será compatible con el sistema operativo para móviles Android.
Fuente	RUR-A11
Prioridad	Alta

Tabla 167. RSNF-A25

Identificador	RSNF-A26
Nombre	<u>Versión de Android</u>
Descripción	La aplicación se podrá ejecutar en cualquier dispositivo móvil con versión de la plataforma Android 2.3.3 (API 10) y superiores.
Fuente	RUR-A12
Prioridad	Alta

Tabla 168. RSNF-A26

Identificador	RSNF-A27
Nombre	<u>Protocolo de comunicación</u>
Descripción	Las comunicaciones con el servidor se harán mediante el protocolo HTTP.
Fuente	RUR-A13
Prioridad	Alta

Tabla 169. RSNF-A27

Identificador	RSNF-A28
Nombre	<u>Formato de archivos de somniaquias originales</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	RUR-A14
Prioridad	Alta

Tabla 170. RSNF-A28

Identificador	RSNF-A29
Nombre	<u>Formato de archivos de somniaquias procesadas</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	RUR-A15
Prioridad	Alta

Tabla 171. RSNF-A29

Identificador	RSNF-A30
Nombre	<u>Formato de archivos neutros</u>
Descripción	Los archivos de audio tendrán el siguiente formato: Neutro-usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	RUR-A16
Prioridad	Alta

Tabla 172. RSNF-A30

Identificador	RSNF-A31
Nombre	<u>Formato de archivos de resultados</u>
Descripción	Los archivos de texto con los resultados tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-R, seguido de la extensión “.txt”.
Fuente	RUR-A17
Prioridad	Alta

Tabla 173 RSNF- A31

Identificador	RSNF-A32
Nombre	<u>Formato de jerarquía</u>
Descripción	<p>El formato .de la jerarquía contiene una carpeta por cada usuario identificada por su nombre de usuario. Dentro de esta carpeta principal existirán cuatro carpetas:</p> <ul style="list-style-type: none"> • Neutro: incluye el archivo con la emoción neutra. • Somniloquias originales: se organiza por fecha y contiene las generadas. • Somniloquias sin ruidos: se organiza por fecha y contiene los archivos post-procesados. • Resultados: se organiza por fecha y contiene los archivos de resultados.
Fuente	RUR-A18
Prioridad	Alta

Tabla 174 RSNF- A32

Identificador	RSNF-A33
Nombre	<u>Idioma</u>
Descripción	La aplicación mostrará los textos en español.
Fuente	RUR-A19
Prioridad	Alta

Tabla 175. RSNF-A33

Identificador	RSNF-A34
Nombre	<u>Configuración inicial del valor del nivel de presión sonora</u>
Descripción	Una de las fases de la creación de usuario es la detección del valor del SPL.
Fuente	RUR-A20
Prioridad	Alta

Tabla 176. RSNF-A34

Servicio Web

Identificador	RSNF-W01
Nombre	<u>Emociones primarias</u>
Descripción	El sistema detectará las emociones primarias de alegría, enfado, tristeza y miedo en los archivos de audio a partir de las características fundamentales.
Fuente	RUR-W01
Prioridad	Alta

Tabla 177 RSNF-W01



Identificador	RSNF-W02
Nombre	<u>Base de datos plataforma web</u>
Descripción	El sistema contará con una base de datos MySQL.
Fuente	RUR-W02
Prioridad	Alta

Tabla 178 RSNF-W02

Identificador	RSNF-W03
Nombre	<u>Servidor Web</u>
Descripción	El sistema contará con un Servidor Web para responder las peticiones de los clientes. El servidor web empleado será un Apache Tomcat.
Fuente	RUR-W03
Prioridad	Alta

Tabla 179 RSNF-W03

Identificador	RSNF-W04
Nombre	<u>Formato de archivos de somniaquias originales</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss, seguido de la extensión “.wav”.
Fuente	RUR-W04
Prioridad	Alta

Tabla 180 RSNF-W04

Identificador	RSNF-W05
Nombre	<u>Formato de archivos de somniaquias procesadas</u>
Descripción	Los archivos de audio tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	RUR-W05
Prioridad	Alta

Tabla 181 RSNF-W05

Identificador	RSNF-W06
Nombre	<u>Formato de archivos neutros</u>
Descripción	Los archivos de audio tendrán el siguiente formato: Neutro-usuario-dd_mm_yy-hh_mm_ss-SR, seguido de la extensión “.wav”.
Fuente	RUR-W06
Prioridad	Alta

Tabla 182 RSNF-W06

Identificador	RSNF-W07
Nombre	<u>Formato de archivos de resultados</u>
Descripción	Los archivos de texto con los resultados tendrán el siguiente formato: usuario-dd_mm_yy-hh_mm_ss-R, seguido de la extensión “.txt”.
Fuente	RUR-W07
Prioridad	Alta

Tabla 183 RSNF-W07

Identificador	RSNF-W08
Nombre	<u>Formato de jerarquía de directorios</u>
Descripción	La jerarquía estará compuesta por cuatro carpetas principales: “Neutro”, “Resultado”, “Somniquias_originales” y “Somniquias_sin_ruidos”. Cada usuario registrado posee dentro de ellas una carpeta con su nombre y los ficheros relacionados.
Fuente	RUR-W08
Prioridad	Alta

Tabla 184 RSNF-W08

Identificador	RSNF-W09
Nombre	<u>Transcripción</u>
Descripción	El servicio web realizará este proceso de manera automática. La transcripción se realiza mediante peticiones a una URL que procesa el audio y devuelve un resultado. Este resultado se parseará para obtener únicamente el texto que nos interesa.
Fuente	RUR-W09
Prioridad	Alta

Tabla 185 RSNF-W09

Identificador	RSNF-W10
Nombre	<u>Resultado del análisis emocional</u>
Descripción	<p>El resultado del análisis de emociones de un archivo se almacenará en un archivo de texto que contendrá los siguientes campos:</p> <ul style="list-style-type: none">• Emoción• Transcripción• Energía• Pitch• Rango del pitch• SPL• Rango del SPL• Media• Varianza• Velocidad
Fuente	RUR-W10
Prioridad	Alta

Tabla 186 RSNF-W10

Identificador	RSNF-W11
Nombre	<u>Post-procesado de la grabación</u>
Descripción	<p>La aplicación realizará un post-procesado de los archivos de audio para eliminar los fragmentos que no correspondan al habla del usuario, como ronquidos o llantos. Para ello hará un análisis acústico de la energía y los cruces por cero de la señal.</p>
Fuente	RUR-W11
Prioridad	Alta

Tabla 187 RSNF-W11

Identificador	RSNF-W12
Nombre	<u>Almacenamiento</u>
Descripción	<p>Toda la información de los usuarios y los valores neutros se almacenarán en la base de datos del servicio web. De los usuarios se guardará:</p> <ul style="list-style-type: none"> • Nombre de usuario • Contraseña • SPL • Tiempo de silencio entre somniloquias • De los archivos neutros se guardará: <ul style="list-style-type: none"> • Nombre de usuario • Pitch • Rango del pitch • Energía • SPL Rango del SPL • Media • Varianza • Velocidad
Fuente	RUR-W12
Prioridad	Alta

Tabla 188. RSNF-W12

Identificador	RSNF-W13
Nombre	<u>Tipo de información</u>
Descripción	<p>Cada tipo de información que se almacene en el sistema tendrá un formato adecuado. A continuación se especifica el formato:</p> <ul style="list-style-type: none"> • Nombre de usuario: cadena de caracteres. • Contraseña: cadena de caracteres • SPL: double. • Tiempo de silencio entre somniloquias: double. • Pitch: double. • Rango del pitch: double. • Energía: double. • SPL: double. • Rango del SPL: double. • Media: double. • Varianza: double. • Velocidad: double.
Fuente	RUR-W12
Prioridad	Alta

Tabla 189. RSNF-W13

Identificador	RSNF-W14
Nombre	<u>Comunicación</u>
Descripción	La comunicación con el dispositivo móvil se hará por HTTP. El servlet contestará a las peticiones que reciba.
Fuente	RUR-W13
Prioridad	Alta

Tabla 190. RSNF-W14

Identificador	RSNF-W15
Nombre	<u>Filtrar por frecuencias</u>
Descripción	El filtrado de audios por frecuencias se hará para frecuencias mayores de 82 Hz y menores a 1056 Hz, debido a que este es el rango de la voz humana.
Fuente	RUR-W14
Prioridad	Alta

Tabla 191. RSNF-W15

4.5 Arquitectura

La arquitectura cliente-servidor es un modelo de aplicación distribuida entre múltiples procesadores donde hay clientes que solicitan servicios y servidores que los proporcionan. Normalmente se utiliza para separar los servicios, situando cada uno en su plataforma más adecuada.

Nuestro sistema se ajusta a esta arquitectura, debido a que el dispositivo móvil se comporta como el cliente, realizando peticiones al servicio web que es el que las procesa y emite las respuestas.

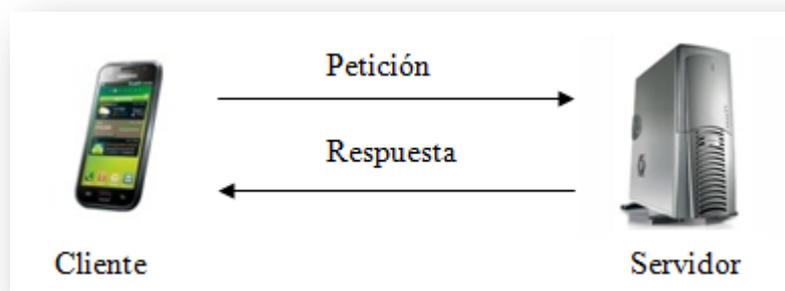


Ilustración 22. Arquitectura cliente-servidor

4.6 Diagrama de clases

En esta sección se muestra la estructura del sistema a partir de sus clases. La vista que proponemos no es la clásica, con todos los atributos y métodos por cada clase, debido a que nos extenderíamos en exceso con ello, al contar con una gran cantidad de clases en el proyecto. Los diagramas únicamente incluyen las clases y la relación que poseen con el resto de módulos del sistema, es decir, si una clase del módulo que se explica se asocia con otra clase de otro módulo, habrá una relación en el diagrama clase que lo represente. La descripción de los módulos se describe a continuación en dos bloques separados: plataforma móvil y servicio web.

Plataforma Móvil

4.6.1 Módulo Core Functionality

En el siguiente diagrama de clases se muestra el módulo encargado de la funcionalidad principal de la aplicación móvil.

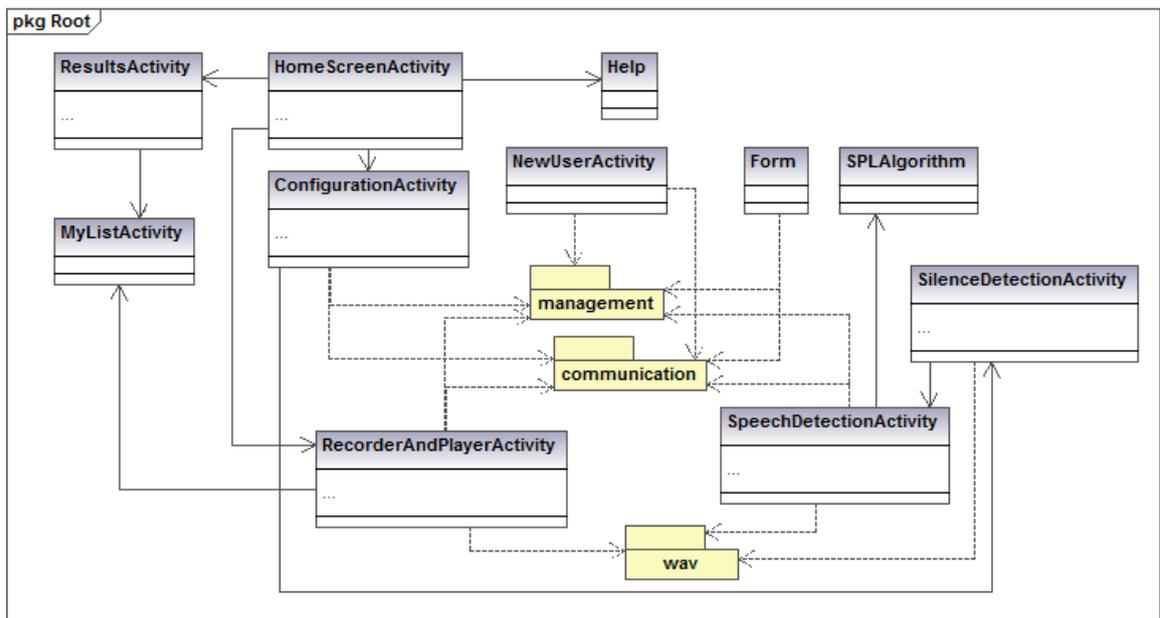


Ilustración 23. Diagrama de clases de Core Functionality

Descripción de las clases:

- **RecorderAndPlayerActivity:** Esta clase implementa la funcionalidad relacionada con la grabación y reproducción de archivos de audio mediante una interfaz de usuario.
 - Asociación externa: Se comunica con el módulo “wav” a través de las clases “WavAudioPlayer” y “WavAudioRecorder”. Se comunica con el módulo “management” a través de las clases “FileManagement”, “User” y “UserDataManagement”. Por último se relaciona con el módulo “communication” a través de la clase “Communication”:

- **ConfigurationActivity:** Esta clase contiene la interfaz dedicada a la ventana de configuración de usuario. En ella se implementan las funcionalidades siguientes: detección automática del SPL, modificar el tiempo en silencio y borrar usuario.
 - Asociación externa: Se comunica con el módulo “management” a través de las clases “FileManagement” y “UserDataManagement”, y con el módulo “communication”, a través de la clase “Communication”.

- **NewUserActivity:** Esta clase implementa la funcionalidad dedicada a la creación de un nuevo usuario en la aplicación mediante la introducción de un usuario y contraseña.
 - Asociación externa: Se comunica con el módulo “management” a través de “FileManagement”, “UserDataManagement” y “StringEncrypter” y con el módulo “communication”, a través de “Communication”.

- **Form:** Esta clase contiene la interfaz que permite realizar el *login* de un usuario mediante un usuario y contraseña.
 - Asociación externa: Se comunica con el módulo “communication” a través de la clase “Communication” y con el módulo “management” a través de “FileManagement”, “User”, “UserDataManagement” y “StringEncrypter”.

- **SilenceDetectionActivity:** Esta clase implementa la funcionalidad relacionada con la detección del SPL del ruido ambiental. El resultado se almacena para después tenerlo en cuenta en el cálculo del SPL de activación del usuario.
 - Asociación externa: Se comunica con el módulo “wav” a través de la clase “WavAudioRecorderWithoutFile”.
- **SpeechDetectionActivity:** Esta clase implementa la funcionalidad relacionada con la detección del SPL que se registra durante una locución del usuario en voz alta. Además se realiza la grabación del habla del usuario en un archivo de audio.
 - Asociación externa: Se comunica con el módulo “wav” a través de “WavAudioRecorder”. Se comunica con el módulo “management” a través de “FileManagement”, y “UserDataManagement”, y con el módulo “communication” a través de la clase “Communication”.
- **SPLAlgorithm:** Esta clase implementa un algoritmo que calcula el SPL de activación del usuario a partir de las intensidades captadas por el micrófono en la fase de silencio y de detección del habla.
- **Help:** Esta clase contiene la interfaz dedicada a mostrar la información de ayuda al usuario.
- **MyListActivity:** Esta clase contiene la interfaz dedicada a mostrar un listado de archivos y directorios. Permite seleccionar un archivo para su futura lectura.
- **HomeScreenActivity:** Esta clase implementa la interfaz de la pantalla principal de usuario, que contiene las pestañas de navegación.
- **ResultsActivity:** Esta clase contiene la interfaz que muestra los resultados de un archivo de audio generados en el servidor.

4.6.2 Módulo Wav

En el siguiente diagrama de clases se muestra el módulo encargado del tratamiento de los archivos de audio con formato wav.

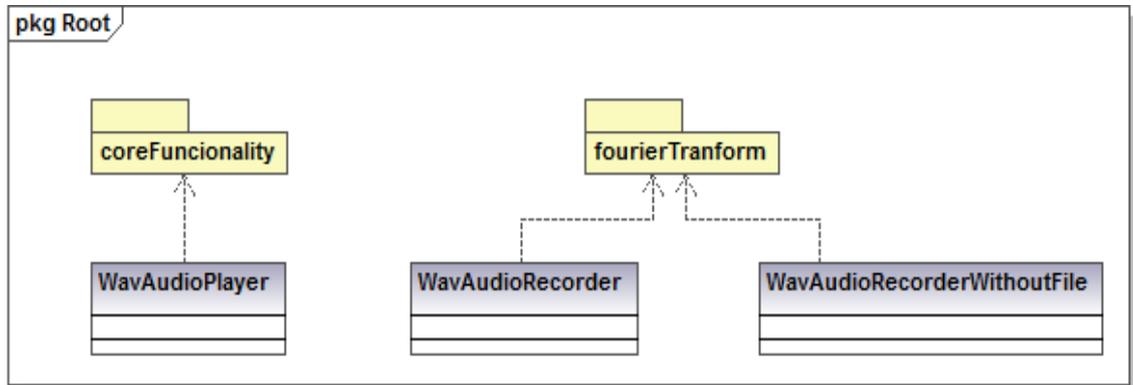


Ilustración 24. Diagrama de clases de Wav

Descripción de las clases:

- **WavAudioRecorder:** Esta clase se encarga de grabar los sonidos que el micrófono del dispositivo capta en archivos de audio que siguen el formato wav. Además prepara y escribe los archivos de este formato. Calcula los campos relacionados con la cabecera antes de crear archivos de audio wav.
 - Asociación externa: Se comunica con el módulo “fourierTransform” a través de las clases “FFT” y “Complex”.
- **WavAudioRecorderWithoutFile:** Esta clase realiza una funcionalidad parecida a la clase “WavAudioRecorder”, pero no realiza la escritura del fichero de audio. Se utiliza para analizar los sonidos captados por el micrófono del dispositivo móvil.
 - Asociación externa: Se comunica con el módulo “fourierTransform” a través de las clases “FFT” y “Complex”.
 -
- **WavAudioPlayer:** Esta clase se encarga de reproducir archivos wav. En primer lugar lee los valores de la cabecera. Después instancia un objeto *AudioTrack* para reproducir los datos que va leyendo de fichero de entrada.
 - Asociación externa: Se comunica con el módulo “coreFuncionalidad” a través de la clase “RecorderAndPlayerActivity”.

4.6.3 Módulo Fourier Transform

En el siguiente diagrama de clases se muestra el módulo encargado del cálculo de la transformada de Fourier.

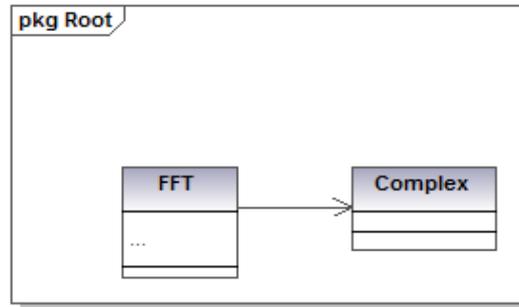


Ilustración 25. Diagrama de clases de Fourier Transform

Descripción de las clases:

- **FFT**: Esta clase se encarga de calcular la transformada rápida de Fourier para pasar al dominio de la frecuencia.
- **Complex**: Esta clase representa un número complejo, necesario para realizar los cálculos de la transformada de Fourier.

4.6.4 Módulo Management

En el siguiente diagrama de clases se muestra el módulo encargado de la gestión de los datos de usuario en la base de datos y la jerarquía de ficheros.

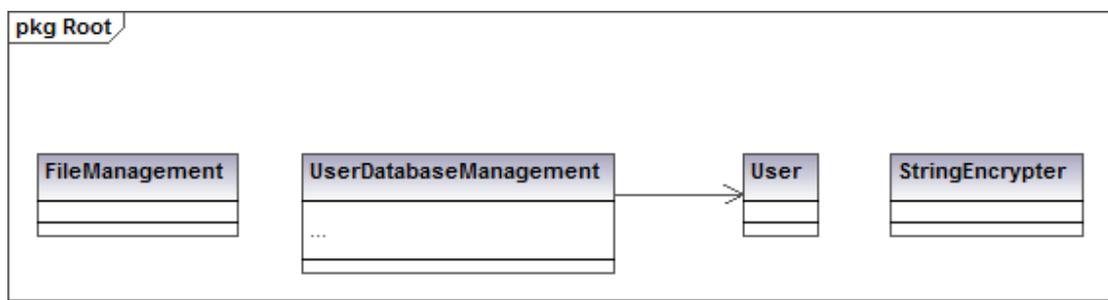


Ilustración 26. Diagrama de clases de Management

Descripción de las clases:

- **FileManagement:** Esta clase se encarga de gestionar la jerarquía de ficheros asociada a un usuario.
- **UserDataManagement:** Esta clase se encarga de gestionar la base de datos con los datos del usuario en el dispositivo móvil.
- **User:** Esta clase gestiona los datos del usuario actual.
- **StringEncrypter:** Esta clase se encarga de encriptar y desencriptar cadenas de texto. Se utiliza para cifrar la contraseña.

4.6.5 Módulo Communication

En el siguiente diagrama de clases se muestra la relación del módulo encargado de la comunicación con el servidor.

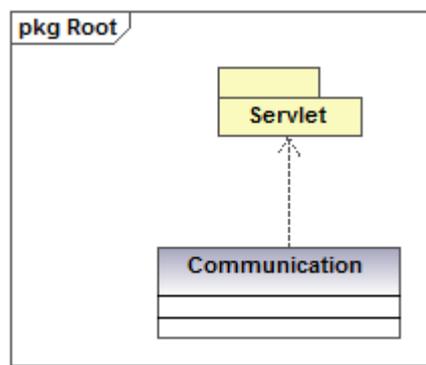


Ilustración 27. Diagrama de clases de Communication

Descripción de las clases:

- **Communication:** esta clase es la encargada de gestionar las peticiones que se hacen al servlet, además de recibir las respuestas.
 - Asociación externa: Se comunica con el módulo “Servlet” a través de las clases “FileServlet” y “SimpleServlet”.

Servicio Web

4.6.6 Módulo Servlet

El siguiente diagrama UML representa la interacción de las clases del módulo “Servlet”.

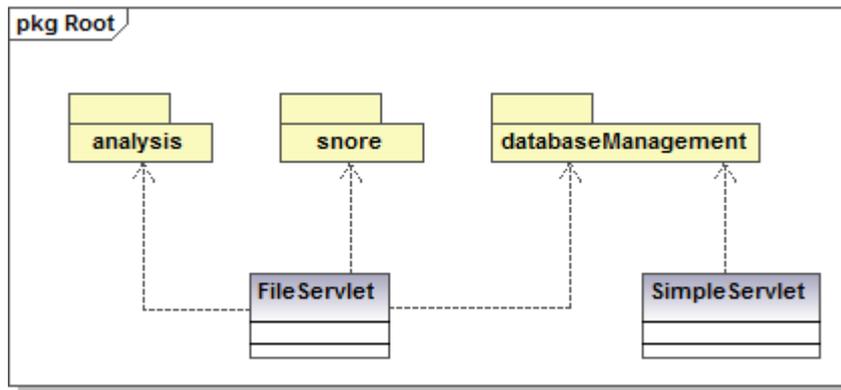


Ilustración 28. Diagrama de clases de Servlet

Descripción de las clases:

- **FileServlet:** es la clase que gestiona todas las peticiones relacionadas con el procesado o envío de ficheros.
 - Asociación externa: Se comunica con el módulo “Analysis” a través de las clases “NeutralAnalyzer” y “EmotionAnalyzer”. Utiliza la clase “PostProcessing” del módulo “Snore”.
- **SimpleServlet:** es la clase que gestiona todas las peticiones relacionadas con el procesado de cadenas de texto.
 - Asociación externa: Se comunica con el módulo “Database Management” mediante las clases “Helper” y “User”.

4.6.7 Módulo Analysis

El siguiente diagrama UML representa la interacción de las clases del módulo “Analysis” entre ellas y con otros módulos del sistema.

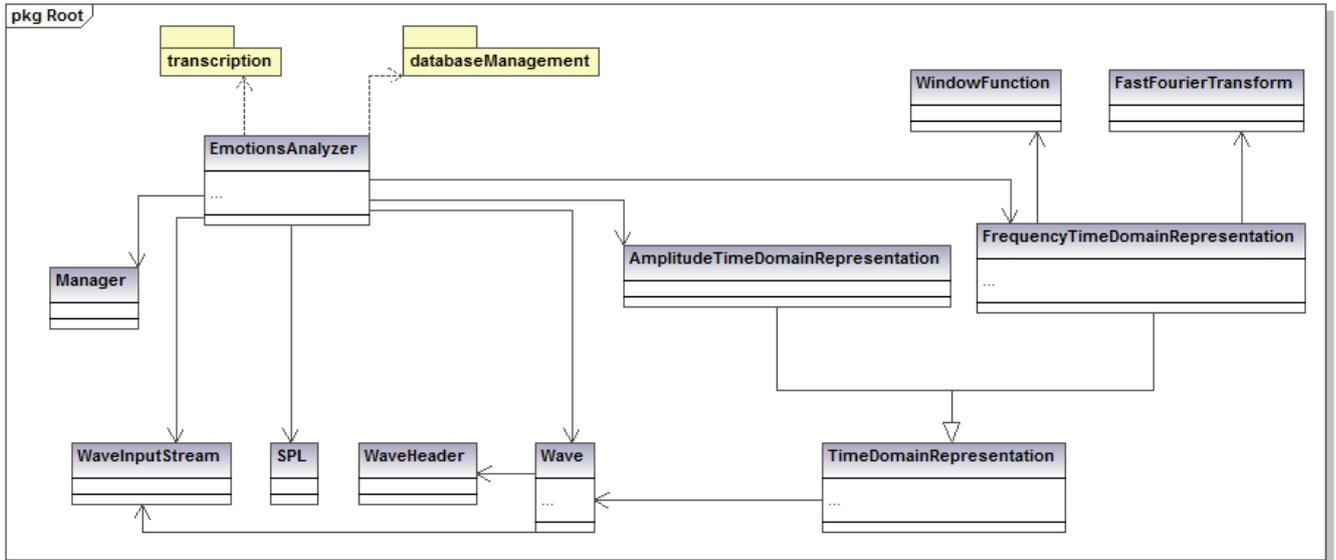


Ilustración 29. Diagrama de clases de Analysis

Descripción de las clases:

- **EmotionsAnalyzer:** es la clase principal del módulo, cuyo propósito es la gestión del proceso de análisis acústico de los ficheros.
 - Asociación externa: Se comunica con el módulo “Transcription” a través de las clases “Transcription” y “Parser”. También accede a la clase “Emotion” del paquete “DatabaseManagement”.
- **Manager:** es la clase que gestiona el reconocimiento de emociones.
- **WaveHeader:** es la clase encargada de determinar si la cabecera del fichero “wav” es correcta o no.
- **WaveInputStream:** es la clase encargada de leer los datos del InputStream de entrada, a excepción de los primeros 44 bytes de los datos, que son los pertenecientes a la cabecera del archivo “wav”.
- **Wave:** Es la clase que gestiona la lectura de los ficheros de tipo “wav”



- **TimeDomainRepresentation**: es la clase encargada de la representación en el dominio del tiempo y ejerce de clase padre de la cual heredan “FrequencyTimeDomainRepresentation” y “AmplitudeTimeDomainRepresentation”.
- **AmplitudeTimeDomainRepresentation**: Esta clase permite manejar los datos de la señal en el dominio de amplitud-tiempo. Su principal función es la de normalizar las amplitudes y devolvérselas al usuario para que pueda trabajar con ellas.
- **FrequencyTimeDomainRepresentation**: es la clase encargada de manejar los datos del audio en el dominio de la frecuencia y de construir el espectrograma de la señal.
- **WindowsFuction**: esta clase permite generar y usar diferentes tipos de ventanas: rectangular, Bartlett, Hanning, Hamming y Blackman.
- **FastFourierTransform**: es la clase encargada de la realización de la Transformada Rápida de Fourier
- **SPL**: es la clase encargada de obtener el SPL y el rango del SPL.

4.6.8 Módulo Transcription

El siguiente diagrama UML representa la interacción de las clases del módulo “Transcription”.

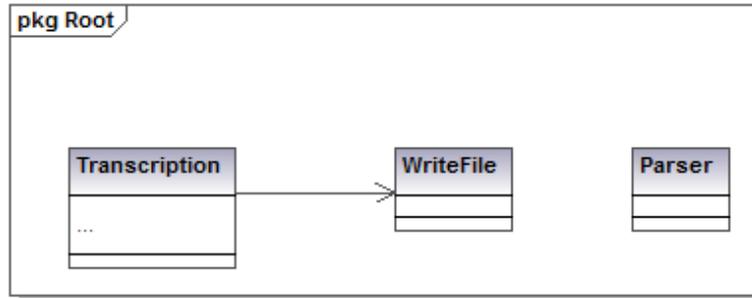


Ilustración 30. Diagrama de clases de Transcription

Descripción de las clases:

- **Transcription:** es la clase principal del módulo, cuyo propósito es la transcripción a texto de un fichero de audio.
- **WriteFile:** es la clase encargada de la escritura de ficheros.
- **Parser:** es la clase encargada de analizar el texto de entrada de un fichero para obtener las partes importantes.

4.6.9 Módulo Snore

El siguiente diagrama UML representa el módulo encargado de la eliminación de ruidos y ronquidos.

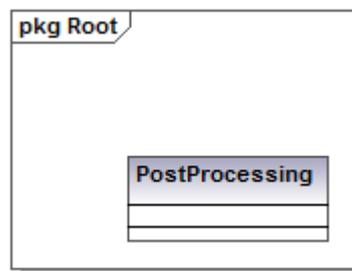


Ilustración 31. Diagrama de clases de Snore

Descripción de las clases:

- **PostProcessing:** Esta clase se encarga de analizar acústicamente las somniaquias del usuario, detecta los fragmentos de ruidos y ronquido y los elimina, generando un nuevo archivo de audio.

4.6.10 Módulo Database Management

El siguiente diagrama UML representa la interacción de las clases del módulo “Database Management”.

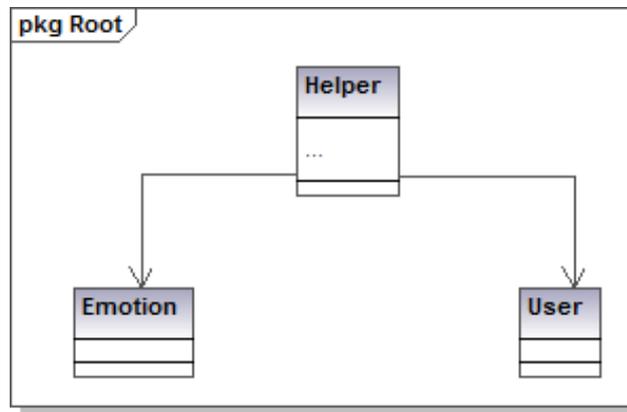


Ilustración 32. Diagrama de clases de Database Management

Descripción de las clases:

- **Helper**: es la clase encargada de gestionar la base de datos y los usuarios del sistema.
- **Emotion**: esta clase representa a una emoción y posee todas las características fundamentales necesarias para el reconocimiento.
- **User**: esta clase representa a un usuario del sistema.





Capítulo 5

Implementación

"Crear es tan difícil como ser libre."

Elsa Triolet (1896-1970)

5.1 Dispositivo móvil

5.1.1 Módulo de funcionalidad principal

Este módulo recoge las principales funciones de la aplicación Android: módulo de grabación y módulo de detección del SPL.

Módulo de grabación

En esta sección se van a explicar los pasos que se han seguido para implementar la grabación de audio mediante la detección del nivel de presión sonora a tiempo real.

El primer paso fue diseñar un ejemplo de una aplicación simple capaz de grabar audio a través del hardware del dispositivo móvil. Para ello era necesario utilizar una de las siguientes clases:

AudioRecord:

Esta clase gestiona los recursos de audio para aplicaciones Java que necesiten grabar la entrada de audio desde el hardware de la plataforma utilizada. Esto se consigue mediante la lectura de los datos del objeto *AudioRecord*, el cual se encarga de extraer en tiempo real los datos capturados desde el micrófono con uno de estos tres métodos:

- *read(byte[], int, int)*
- *read(short[], int, int)*
- *read(ByteBuffer, int)*

La elección del método a emplear se basa en el formato de almacenamiento de datos de audio que se quiera utilizar.

Tras su creación, un objeto *AudioRecord* inicializa el buffer de audio asociado que irá completando con los datos nuevos de audio. El tamaño de este buffer, especificado durante la construcción, determina la cantidad de información que un *AudioRecord* puede grabar antes de sobrescribir los datos que no han sido leídos todavía.

Para su uso es necesario configurar el objeto *AudioRecord* con los siguientes parámetros:

- Frecuencia de muestreo o sample rates.
- Número de canales:
 - 1 canal: “mono”
 - 2 canales: “stereo”
- Bits por muestra.

MediaRecorder.

Se utiliza para grabar audio y video. El control de grabación se basa en la siguiente máquina de estados:

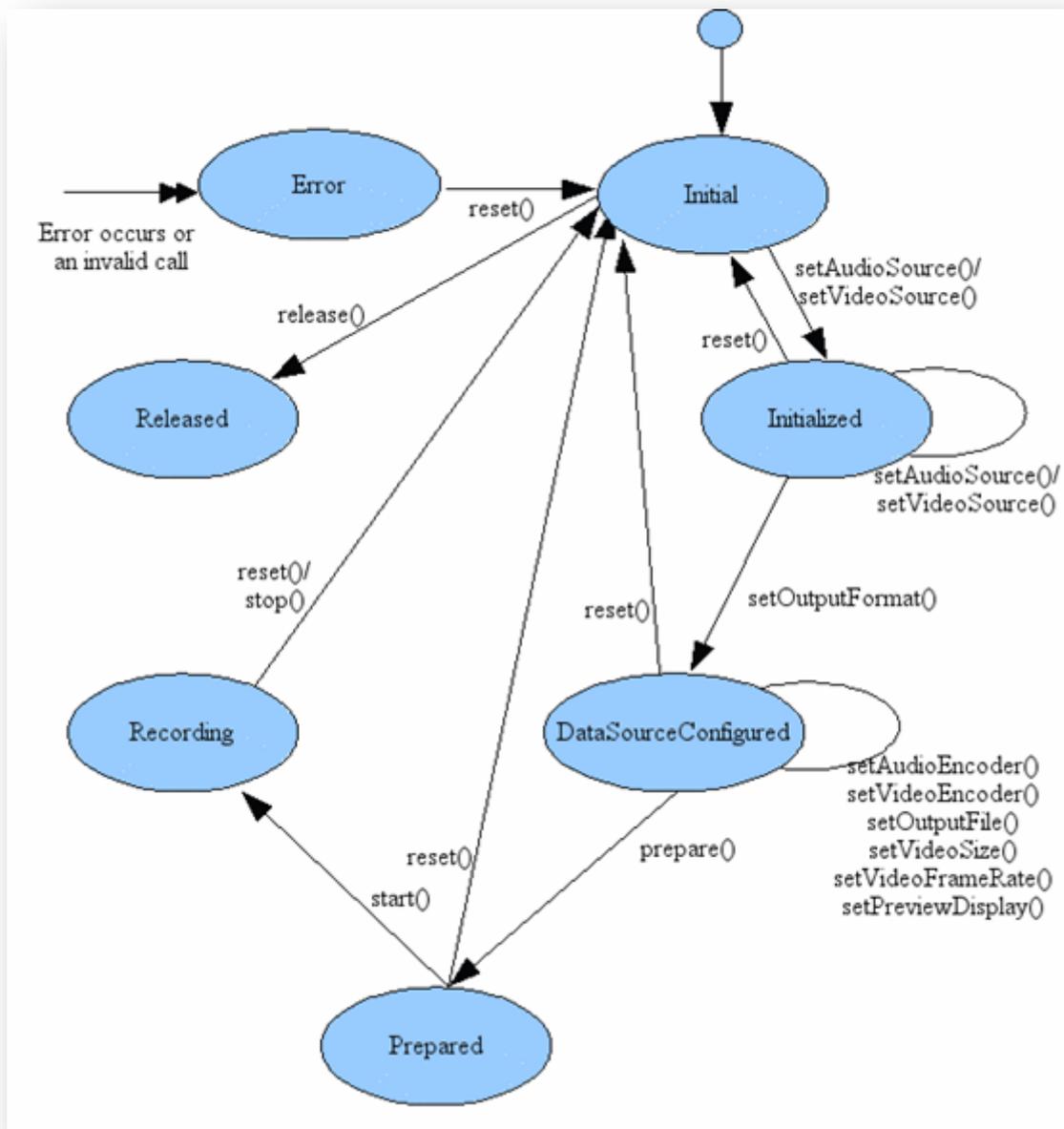


Ilustración 33. Diagrama de estado del MediaRecorder

Para su uso es necesario crear un objeto de tipo *MediaRecorder* y asignarle la fuente de entrada, en este caso *AudioSource.MIC*, para el micrófono del dispositivo. También es necesario especificar el formato de salida, el nombre del archivo y la codificación de audio.

Tras analizar las características de ambas clases de grabación de audio, nos decantamos por *AudioRecord*. Esto es debido a la mayor flexibilidad de configuración que proporciona esta clase frente a *MediaRecorder*. *AudioRecord* permite establecer el tipo de calidad de audio, elegir el formato de codificación y la configuración de canales.

Para realizar la grabación correctamente se tuvieron que escoger aquellos parámetros que permitieran la grabación en el dispositivo móvil en cuestión. No todos los dispositivos graban con la misma frecuencia de muestreo. La solución que se tomó fue probar en cada ejecución distintas configuraciones hasta encontrar aquella que fuese adecuada.

Por lo tanto, en la implementación final, se tiene una matriz de frecuencias de muestreo con los siguientes valores *sampleRates* = {44100, 22050, 11025, 16000} y se ejecuta el siguiente código:

```
int i=0;
do{
    result = new WavAudioRecorder(
        true,
        AudioSource.MIC,
        sampleRates[i],
        AudioFormat.CHANNEL_CONFIGURATION_STEREO,
        AudioFormat.ENCODING_PCM_16BIT);

    while(++i<sampleRates.length) &
        !(result.getState() ==
        ExtAudioRecorder.State.INITIALIZING));
```

El siguiente paso consistía en conseguir que la aplicación grabase tras detectar sonido por encima de un determinado volumen. Para ello nos centramos en diseñar un algoritmo capaz de calcular el *nivel de presión sonora* ó *SPL* en tiempo real.

La idea inicial del algoritmo de detección y grabación seguía la siguiente estructura:

1. Crear un objeto AudioRecord
2. Empezar a escuchar, calcular el SPL en cada muestra y comparar con un umbral
3. Si el SPL supera el umbral:
 - a. Iniciar grabación
4. Si no, seguir escuchando

Se utilizó la clase “FFT” - puede encontrarse aquí [26] - que implementa la transformada rápida de Fourier para pasar la señal del dominio del tiempo al dominio de la frecuencia. La transformada es necesaria para calcular el SPL de una muestra. Este cálculo se realiza mediante la ejecución de un código adaptado de la aplicación SPLMeter. Se puede consultar en [11].

En las primeras versiones del código, el algoritmo calculaba continuamente el SPL en cada muestra, pero pronto nos dimos cuenta de que esta forma de operar consumía demasiados recursos en el teléfono móvil. Además no era eficaz ya que algunas muestras se perdían. Por eso se definió un nuevo parámetro: el *tiempo en silencio*. Este atributo define el tiempo que tiene que transcurrir desde que se detectó sonido hasta el próximo cálculo del nivel de presión sonora. Pasado ese tiempo, si se detecta un volumen por encima del umbral definido se continúa la grabación durante el tiempo en silencio definido. En caso contrario, se verifica el SPL durante una serie de muestras para que sea evidente que no existe información auditiva a almacenar.

El módulo de grabación se activa cuando el usuario pulsa el botón de grabar en la interfaz del dispositivo móvil. El algoritmo implementado en este módulo es el siguiente:

1. Se cambia el estado actual a “Grabando” y se crea un directorio cuyo nombre es la fecha actual con el siguiente formato: *dd_mm_aaaa*.
2. A continuación se crea un objeto del tipo *WavAudioRecorder* y se le asigna un *Listener*.
3. Después se llama al método *start*: aquí se calcula el tamaño del buffer de grabación y comienza la grabación. Cada vez que hay datos por leer se activa el *Listener* y se hace lo siguiente:
 - 3.1 Se leen los datos de entrada y se guardan en un buffer.
 - 3.1.1 Si no se están guardando los datos en fichero.
 - Se calcula la intensidad o *SPL* actual mediante el método *measure*.
 - Si la intensidad supera el umbral de decibelios de grabación entonces se crea un nuevo fichero cuyo formato es el siguiente:
$$\text{nombre_usuario-}dd_mm_aaaa_hh_mm_ss.wav$$
 - Se escribe en fichero la cabecera relacionada con el formato *wav* con los datos correspondientes a la configuración actual. La configuración por defecto es la siguiente:
 - *Codificación de audio: PCM, 2 Channels, Stereo, uncompressed, SampleRates = 16000, bit per Sample = 16.*
 - Se calcula el periodo de captura de muestras:
$$\text{framePeriod} = \text{bufferSize} / (2 * \text{bSamples} * \text{nChannels} / 8)$$
 - Se escribe en el fichero el primer buffer y se actualiza el *payloadSize*.
 - Se inicializa el contador del tiempo tomando el tiempo actual.
 - 3.1.2 Si ya se están guardando datos en fichero.
 - Se escribe un nuevo buffer y se actualiza el *payloadSize*.
 - Se toma el tiempo actual y se comprueba si ha pasado el tiempo de silencio.
 - Se calcula la intensidad o *SPL* actual.
 - 3.1.3 Si la intensidad supera el umbral de decibelios de grabación entonces el contador inicial de tiempo toma se inicializa de nuevo.
 - 3.1.4 Sino la grabación se para y se hace lo siguiente:
 - Se escribe en la cabecera el número total de bytes del archivo y el número total de bytes de audio y se cierra el archivo.
 - Se vuelve al paso 2.

Si el usuario pulsa el botón de “stop” de grabación, la aplicación detiene la grabación actual y escribe en la cabecera del fichero de audio wav el número total de bytes del archivo y el número total de bytes de audio. Finalmente se cierra el archivo.

Durante la grabación, cada vez que se genera un archivo de audio completo se ejecuta un hilo que se encarga de enviar el archivo al servidor. A partir de aquí se realizan distintos intercambios de información entre el dispositivo móvil y el servidor.

1. El dispositivo móvil envía el archivo de audio generado.
2. El servidor le contesta con “Archivo OK\r\n”
3. El dispositivo móvil le pide al servidor el archivo de audio post-procesado.
4. El servidor genera y envía el archivo de audio post-procesado.
5. El dispositivo móvil pide al servidor el fichero de resultados.
6. El servidor genera y envía el fichero de resultados.

En cuanto al formato de audio utilizado, en un principio los datos se almacenaban en bruto. Los archivos seguían únicamente el formato de codificación digital “*Pulse Code Modulation*” o *PCM*, es decir, los datos no se comprimen y no existen “pérdidas”. La calidad de los archivos de audio es muy buena, ya que los datos no van comprimidos. Sin embargo, más tarde surgieron problemas de compatibilidad con este tipo de formato, al no contener ningún tipo de cabecera de datos. Por ello se recurrió al formato wav, el cual sigue la codificación PCM y además añade a los archivos una cabecera con información relevante sobre la codificación del audio.

El algoritmo realiza el envío de los archivos de audio generados al servidor cada vez que se genera un nuevo fichero de audio. El proceso se ejecuta tras crear un nuevo hilo encargado de realizar el envío del fichero de audio al servidor y de obtener los resultados que el servidor genera.

Módulo de Detección del SPL.

En esta sección se van a explicar los pasos que se han seguido para implementar el módulo encargado de detectar de forma automática el nivel de presión sonora.

En primer lugar hay que tener en cuenta que el nivel de presión sonora ó *SPL* toma valores distintos en espacios distintos, en función del ruido ambiental. De la misma manera ocurre que el *SPL* varía entre locuciones de diferentes personas.

Como se explicó en el módulo de grabación, la aplicación debe activarse tras detectar un cierto nivel de presión sonora. Por ello se consideró de vital importancia el desarrollo de un módulo capaz de calcular automáticamente el valor o umbral de activación del SPL para cada usuario.

Desde un principio el diseño de este módulo se dividió en tres sub-módulos distintos pero relacionados:

- Detección del SPL del ruido ambiente (el usuario debe permanecer en silencio).
- Detección del SPL del habla del usuario.
- Cálculo del SPL de activación.

En la figura siguiente se puede ver el esquema que sigue el módulo de detección del SPL:

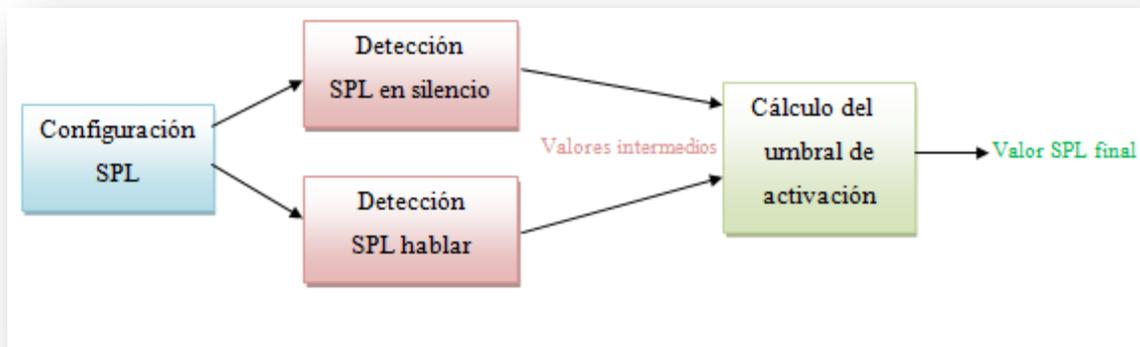


Ilustración 34 Esquema del módulo de detección del SPL.

Partiendo del código utilizado en el módulo de grabación que calcula el SPL en cada muestra, se optó por utilizar este código en la detección del silencio y en la detección del habla.

En el primer sub-módulo se pide al usuario que permanezca en silencio y durante cinco segundos se toman un número predefinido de muestras. Se realiza la transformada de la señal al dominio de la frecuencia y se calcula el SPL en cada muestra.

En el segundo sub-módulo se realiza el mismo proceso, pero esta vez pidiendo al usuario que hable.

Los valores de ambos sub-módulos forman la entrada del módulo encargado de calcular el SPL de activación. El algoritmo que implementa esta funcionalidad se puede ver a continuación:

1. Se calcula la media aritmética del nivel de presión sonora de todas las muestras en silencio → $mediaSPLSilencio$
2. Se calcula la media aritmética del nivel de presión sonora de todas las muestras del habla → $mediaSPLHablar$
3. Para cada una de las muestras en silencio:
 - a. Si el valor del SPL $>$ $mediaSPLSilencio$
 - i. Se elimina el valor
4. Se calcula una nueva media con los valores del SPL de las muestras en silencio no eliminados.
5. Para cada una de las muestras del habla:
 - a. Si el valor del SPL $>$ $mediaSPLHablar$
 - i. Se elimina el valor
6. Se calcula una nueva media con los valores del SPL de las muestras del habla no eliminados.
7. Se calcula la media aritmética de ambos valores.

En un principio el algoritmo diseñado de obtención del umbral de activación era más sencillo. Su función era calcular la media aritmética del SPL en silencio y del SPL del habla. El algoritmo no eliminaba muestra alguna y finalmente calculaba la media aritmética de ambos valores.

Sin embargo, nos dimos cuenta de que se consigue una mayor precisión cuando el algoritmo elimina muestras que podemos considerar erróneas. Por ejemplo:

- Sonidos o ruidos imprevistos en la etapa de detección del silencio.
- Fragmentos con pausas o silencios en la detección del SPL del habla

Muestras de este tipo influyen negativamente en el cálculo del umbral del SPL de activación.

5.1.2 Módulo de Gestión

Gestión de Usuario en la base de datos

Para la gestión de los usuarios y sus datos se hace uso de una base de datos SQLite para Android que, de forma local, guarda su configuración. Además se hace uso del almacenamiento de la *sdcard* del dispositivo para guardar de forma organizada los archivos de audio y texto relacionados con la aplicación.

La clase *GestionUsuariosBDD* contiene la clase anidada *DataBaseHelper*, la cual extiende de *SQLiteOpenHelper*. Esta clase es la encargada de gestionar la base de datos. En ella se crea la tabla de usuarios y se definen los campos que contendrá dicha tabla:

Usuarios.db	
<i>Campo</i>	<i>Valor</i>
TABLE_KEY_ID	_id
USER	user
PASSWORD	pwd
SPL	spl
TSILENCIO	tsilencio

Tabla 192. Tabla de Usuario en SQLite

Además se le asigna un nombre y una ruta a dicha tabla. Primero se comprueba si existe la base de datos y si no existe la crea. Después se abre la base de datos para poder operar con ella.

Es en esta clase donde se gestiona la información de usuario en la base de datos del dispositivo. Las funciones principales son:

- Insertar un nuevo usuario con todos sus datos. Si en principio no se cuenta con todos los datos, se ponen por defecto.
- Configurar Usuario: introduce el spl dado por la detección ambiental.
- Actualizar SPL y Tiempo de silencio.
- Borrar usuario. Elimina a un usuario de la base de datos.

Como se ha comentado, la base de datos se guarda en la *sdcard* del dispositivo móvil. Para acceder a esta información desde cualquier clase, se debe instanciar un objeto de tipo `GestionUsuariosBBDD`, y desde ahí abrir la base de datos, o crearla en caso de que no exista, y acceder al contenido.

Como herramienta de ayuda para la comprobación de la gestión de la base de datos se ha utilizado la aplicación *SQLite Database Browser 2.0 b1* [27].

El Explorador de base de datos SQLite es una utilidad gráfica que se utiliza para crear, editar y diseñar archivos de base de datos compatibles con SQLite. Está diseñado para ser utilizado para los usuarios y los desarrolladores que desean crear bases de datos, edición y datos de búsqueda utilizando una interfaz de hoja de cálculo como familiar, sin necesidad de aprender complicados comandos SQL.

Gestión de Ficheros

La clase GestionFicheros se encarga de crear los directorios necesarios para el almacenamiento organizado de los archivos de cada usuario.

La estructura de ficheros que sigue la aplicación se puede observar en el siguiente gráfico:

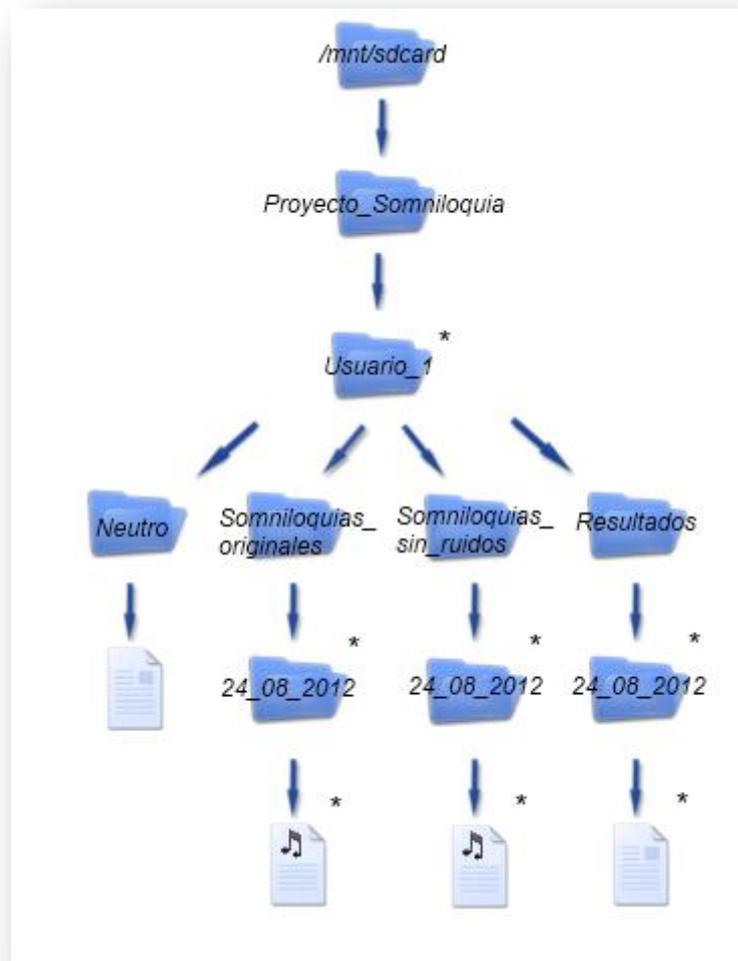


Tabla 193. Jerarquía de carpetas

El símbolo * indica que el número de carpetas en este nivel se encuentra en el intervalo $[0, X)$, donde X es el número máximo de carpetas que la memoria del teléfono puede almacenar.

A cada usuario se le asigna una carpeta distinta. Dado que el nombre de usuario es único se evitan los problemas de sobrescritura de directorios.

Como se puede observar en la figura anterior, cada usuario cuenta con 4 subdirectorios distintos:

- **Neutro:** Contiene un archivo de audio con la grabación de voz del usuario en la etapa de configuración de la intensidad del habla.
- **Somniaquias_originales:** Contiene las somniaquias que han sido grabadas originalmente por el usuario.
- **Somniaquias_sin_ruidos:** Contiene las somniaquias que han sido procesadas en el servidor para la eliminación de ruidos y ronquidos.
- **Resultados:** Contiene los archivos que han sido creados en el servidor con la información relevante de cada archivo.

5.1.3 Módulo de comunicación

Este módulo se encarga de interactuar con el servlet. Ha sido desarrollado por Virginia Aparicio Paniagua y la información referente a esta sección puede encontrarla en su memoria [28].

5.2 Servidor

5.2.1 Módulo Servlet

Este módulo recoge la funcionalidad principal del servicio web que recibe las peticiones del dispositivo móvil. Ha sido desarrollado por Virginia Aparicio Paniagua y la información referente a esta sección puede encontrarla en su memoria [28].

5.2.2 Módulo de gestión de la base de datos

Este módulo se encarga de la gestión de la base de datos MySQL del servidor. Ha sido desarrollado por Virginia Aparicio Paniagua y la información referente a esta sección puede encontrarla en su memoria [28].

5.2.3 Módulo de eliminación de ruidos

El módulo de “post-procesado” de los archivos de audio es un módulo encargado de eliminar aquellos fragmentos de audio en los que no exista habla humana. El algoritmo utilizado se ejecuta en el servidor tras recibir éste una somniloquia original.

A continuación se explican las etapas por las que ha ido pasando el análisis de la definición del algoritmo:

1. Aplicación de filtros en el dominio de la frecuencia para eliminar aquellas frecuencias que no se encontrasen en el rango de de la voz humana [82-1056 Hz].

Aplicar filtros de este tipo no fue una tarea difícil. Utilizar la transformada rápida de Fourier permite pasar al dominio de la frecuencia para después eliminar aquellas frecuencias que no fueran de interés.

Sin embargo, el proceso inverso, necesario para calcular la energía y el ZCR en el dominio del tiempo, nos resultó mucho más complicado. Para convertir la señal filtrada (en el dominio de la frecuencia) al dominio del tiempo, era necesario utilizar la transformada inversa de Fourier. Pero recuperar la señal original, habiendo eliminado las frecuencias filtradas, requería un procesamiento complejo. Aunque este proceso facilita la labor de la eliminación de ruidos con frecuencias distintas al rango de frecuencias

de la voz humana, no elimina, sin embargo, otros ruidos como los ronquidos.

2. Se pensó en aplicar el siguiente algoritmo descrito en [28], que explica lo siguiente:

Calcula la potencia del espectro entre 100–800 Hz. Si excede en este rango por encima del 50% del total, se identifica el acontecimiento como ronquido.

Pero en este rango de frecuencias también se encuentra el habla humana, por lo que descartamos el algoritmo al no ser lo suficientemente discriminante.

3. Se crearon distintas grabaciones de voz y ronquidos de individuos para analizar algunas características en la señal auditiva que permitieran diferenciar entre ambos eventos. También era de nuestro interés captar cualquier otro tipo distinto de ruidos. Debíamos analizar los parámetros a estudiar en las señales obtenidas para después seleccionar los más influyentes en la tarea de detección de ruidos y ronquidos.
4. Para conocer cuáles son las características que definen un ronquido y cuáles permiten discriminar entre el ronquido y el habla, se analizaron distintos ejemplos de señales. El análisis se centró en dos variables:
 - **Energía.**
 - **Cruces por Cero.**

Teniendo en cuenta que el número de cruces por cero varía entre muestras con ruidos o ronquidos y muestras con habla, se analizaron algunos ejemplos de señales con ambos eventos. Cada ejemplo consiste en un archivo de audio que contiene la grabación generada por un individuo. Este archivo contiene fragmentos en los que la persona habla y fragmentos en los que ronca. Antes de mostrar los resultados se explicarán algunos aspectos a tener en cuenta:

Para el análisis de las características de las señales de audio obtenidas se utilizaron distintos tamaños de muestra. Era necesario elegir un tamaño de muestra que cumpliera las siguientes características:

- Si el tamaño de muestra es grande, aumenta la probabilidad de que, en cada muestra, se produzca un mayor número de cruces por cero. Por el contrario, si el tamaño de las muestras no es lo suficientemente grande el ZCR será próximo entre muestras con habla y con ronquido, lo que hará más complicada la discriminación entre unas y otras muestras.
- Muestras demasiado grandes disminuyen la precisión de detectar dónde empiezan y dónde acaban fragmentos de habla y de ruido.

Por eso era necesario elegir un tamaño de muestra que tuviera en cuenta estas dos consideraciones y su valor se encontrase entre estos dos márgenes. Por otra parte, la energía muestra de forma muy gráfica las intervenciones del habla y de ronquidos o de cualquier sonido que se produzca. Este resultado útil para determinar donde empezaba cada fragmento. También había que tener en cuenta el umbral de la energía a partir del cual se consideraba sonido audible o no, es decir encontrar donde empieza y acaba un fragmento de audio.

La distancia existente entre fragmentos contiguos también ayuda a conocer si se trata de un ronquido o de una locución hablada. En los periodos con ronquidos se aprecia una cierta periodicidad. En la siguiente Ilustración se muestran tres gráficas con la representación de la amplitud en tiempo, y de la energía y ZCR en frecuencia de varios ronquidos:

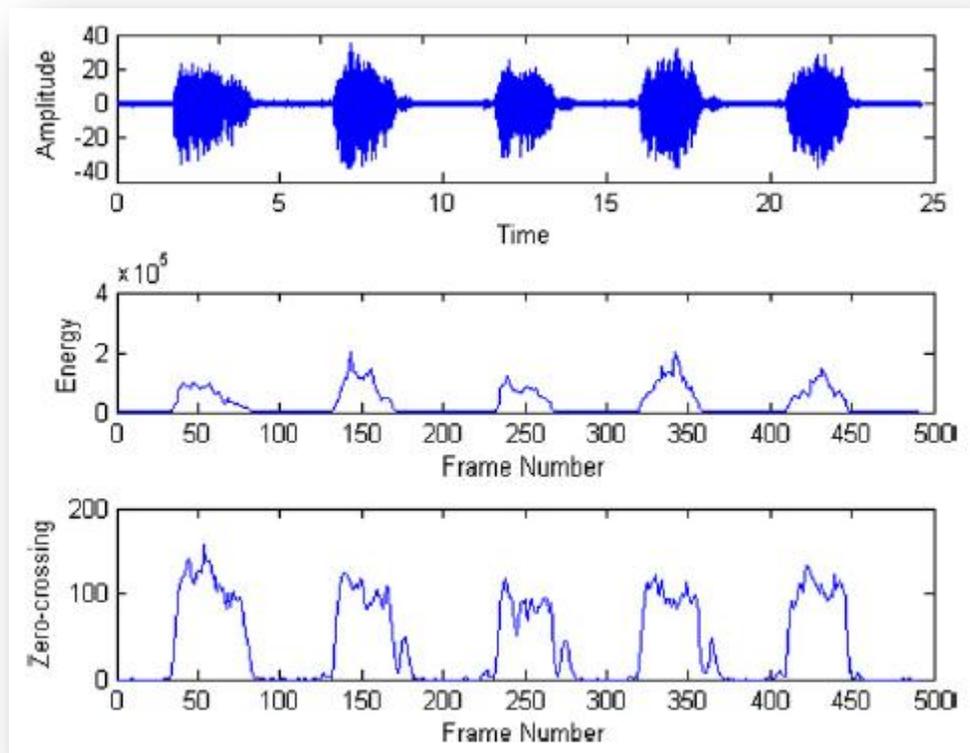


Ilustración 35. Representación en tiempo de la señal de un ronquido, su representación en energía y su representación en ZCR.

Durante la fase de análisis de la eliminación de ruidos y ronquidos se realizaron seis pruebas diferentes. En ellas se grabaron intervenciones de seis individuos distintos, con el objetivo de extraer las características principales de las señales resultantes de la grabación de habla y ronquido. Se debe tener en cuenta que a las dificultades inherentes a una señal acústica en la recogida y en el análisis, se añaden las propias derivadas de los estudios de sueño: señal recogida en una noche de estudio mezclada con ruido de orígenes distintos procedentes del entorno o del propio sujeto. A continuación se presentan dos ejemplos realizados con grabaciones de dos individuos distintos.

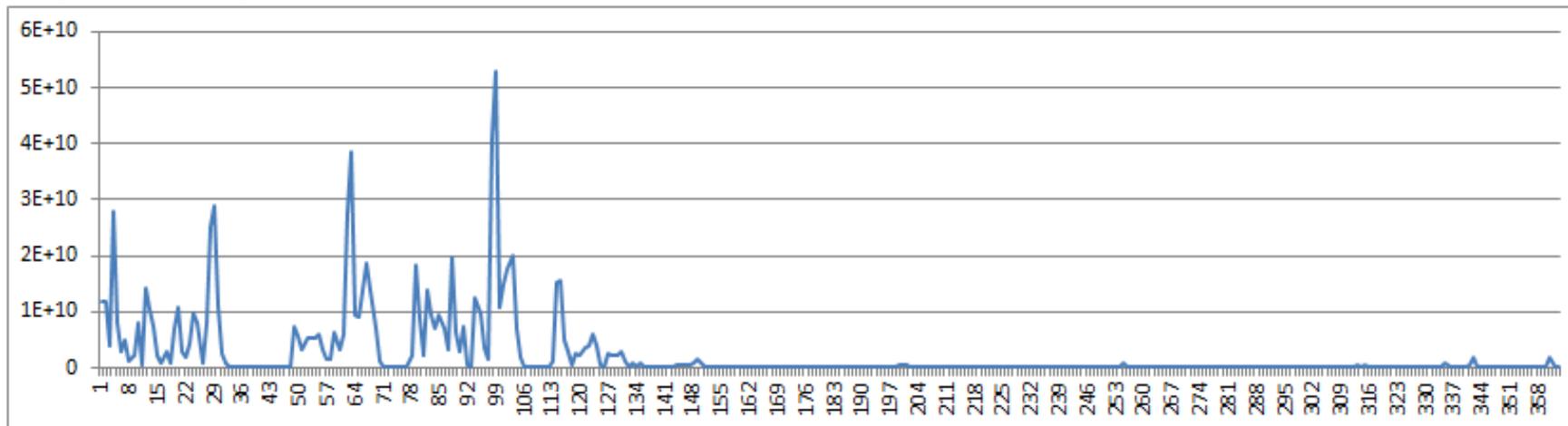
Ejemplo 1:

Ejemplo 1	
Nombre del participante	Carlos Aparicio
Fecha	13/Junio/2012
Frase	“Soy Carlos Aparicio Paniagua, tengo veintiún años, estudio en la Rey Juan Carlos, Grado en Ingeniería informática”
Número de ronquidos	4

Tabla 194. Ejemplo 1, análisis de ronquidos

A continuación se observa la gráfica de la energía de la señal y una tabla con la información del ZCR de cada fragmento.

Gráfica de la Energía:



El eje horizontal representa el número de muestras en el tiempo. El eje vertical representa el valor de la energía en cada muestra.

Ilustración 36. Ejemplo 1, análisis de ronquidos. Gráfica de la Energía

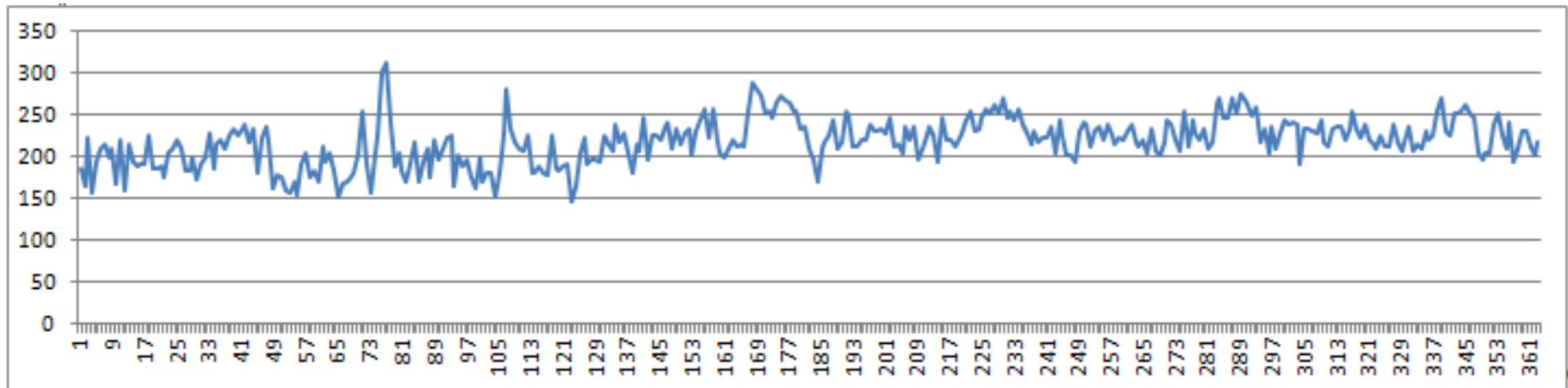
Fragmento	Tipo	Inicio (número de muestra)	Fin (número de muestra)	Varianza del ZCR
Fragmento 1	Habla	0	32	21,14
Fragmento 2	Habla	48	70	23,34
Fragmento 3	Habla	77	106	27,1
Fragmento 4	Habla	112	135	22,03
Fragmento 5	Ronquido	143	158	15,39
Fragmento 6	Ronquido	195	205	13,16
Fragmento 7	Ronquido	222	227	12,65
Fragmento 8	Ronquido	253	263	9,19
Fragmento 9	Ronquido	310	321	11,82

Tabla 195. Resultados ejemplo 1, análisis de ronquidos

Como puede apreciarse en la gráfica de la energía, se distinguen de forma clara los fragmentos de audio en los que el participante habla, de los fragmentos donde se producen los ronquidos.

Además, si observamos los valores del ZCR podemos darnos cuenta de que la varianza es mayor en fragmentos con habla que en fragmentos con ronquido. En la siguiente imagen podemos observar la gráfica de los cruces por cero de la señal.

Gráfica de los Cruces por cero:



El eje horizontal representa el número de muestras en el tiempo. El eje vertical representa el ZCR (Zero crossing rate).

Ilustración 37. Ejemplo 1, análisis de ronquidos. Gráfica del ZCR



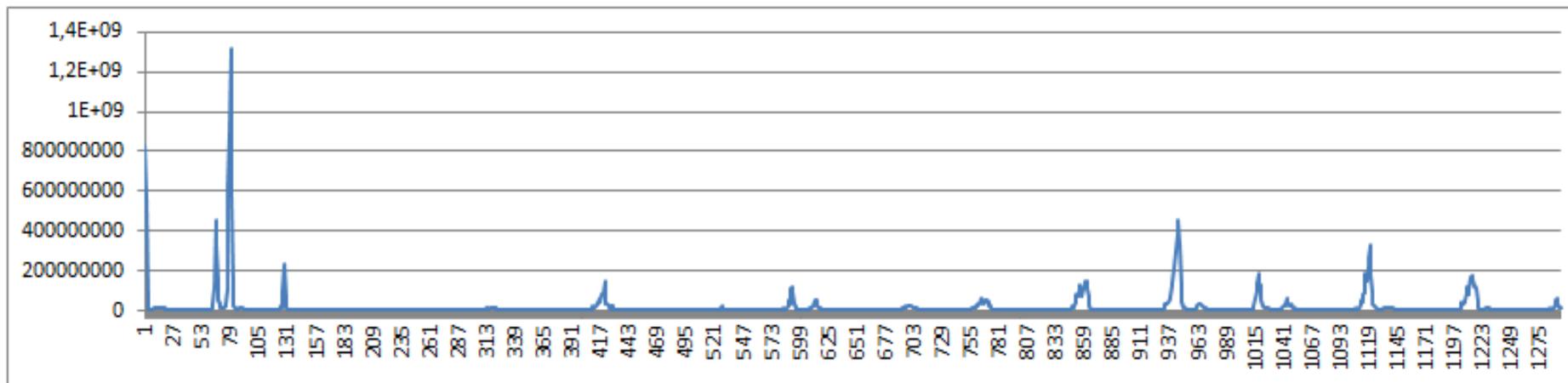
Ejemplo 2:

Ejemplo 2	
Nombre del participante	Valentín Pérez
Fecha	14/Junio/2012
Número de ronquidos	11

Tabla 196. Ejemplo 2, análisis de ronquidos.

A continuación se observa la gráfica de la energía de la señal y una tabla con la información del ZCR de cada fragmento.

Gráfica de la Energía:



El eje horizontal representa el número de muestras en el tiempo. El eje vertical representa el valor de la energía en cada muestra.

Ilustración 38. Ejemplo 2, análisis de ronquidos. Gráfica de la energía

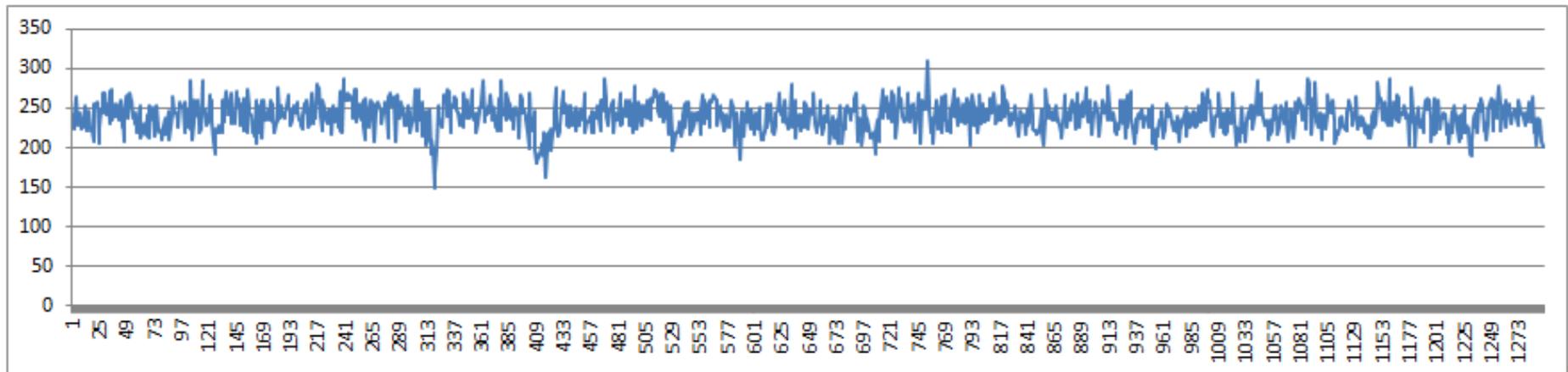
Fragmento	Tipo	Inicio (número de muestra)	Fin (número de muestra)	Varianza del ZCR
Fragmento 1	Ronquido	75	80	6,43
Fragmento 2	Ronquido	851	862	17,9
Fragmento 3	Ronquido	937	946	6,8
Fragmento 5	Ronquido	1113	1121	9,8
Fragmento 6	Ronquido	1207	1216	15,1

Tabla 197. Resultados Ejemplo 2, análisis de ronquidos

En la gráfica de la energía se aprecia perfectamente los momentos en los que hay sonido. Estos sonidos corresponden a los ronquidos del participante.

Además, si observamos los valores del ZCR podemos darnos cuenta de que la varianza del ZCR es baja en comparación con los fragmentos de habla del participante anterior. En la siguiente imagen podemos observar la gráfica de los cruces por cero de la señal.

Gráfica de los Cruces por cero:



El eje horizontal representa el número de muestras en el tiempo. El eje vertical representa el ZCR (Zero crossing rate).

Ilustración 39. Ejemplo 2, análisis de ronquidos. Gráfica del ZCR

Tras estudiar los resultados presentados, se especifica a continuación el algoritmo diseñado:

1. Se recoge una muestra de $t1$ bytes.
2. Se realiza la transformada al dominio de la frecuencia de la señal y se calcula la energía y cruces por cero.
3. Trasladar a buffer de $t2$ bytes para después poder reproducir el archivo de audio resultante.
4. Detectar segmento:
 - a. Se detecta cuando se ha superado el umbral de energía.
 - b. Se busca el final y se comprueba su longitud
 - c. En caso de encontrar un valor inferior se comprueba el siguiente valor:
 - i. Si es inferior: el segmento finaliza.
 - ii. Si es superior: el segmento continúa.
5. Detectar ronquidos:
 - a. Se calcula la distancia al siguiente fragmento.
 - i. Si la distancia es suficientemente grande (como para que dé tiempo a una expiración) o la longitud del ronquido es menor al umbral predefinid, el segmento se clasifica en función de su desviación estándar de cruces por cero.
 1. Menor que 20: hablar → Se conserva.
 2. Mayor que 20: roncar → Se elimina.
6. Se vuelve al paso 4 mientras no finalice el archivo.

* Donde $t1 = 2972$ bytes y $t2 = 1024$ bytes.

El algoritmo se ha diseñado teniendo en cuenta lo siguiente:

Es más importante conservar fragmentos con "falsos ronquidos" que eliminar "verdaderos fragmentos de habla". No solo se detecta ronquido en función de la variación de cruces por cero sino también por la distancia y la longitud de cada segmento.



5.2.4 Módulo de Transcripción

En esta sección se explican los pasos que se han seguido para realizar la transcripción de audio a texto por medio de un sistema de reconocimiento del habla. Ha sido desarrollado por Virginia Aparicio Paniagua y la información referente a esta sección puede encontrarla en su memoria [28].

5.2.5 Módulo de análisis de emociones

En este apartado, se describe el proceso seguido para construir el módulo software encargado de la obtención de las emociones presentes en archivos de audios. Ha sido desarrollado por Virginia Aparicio Paniagua y la información referente a esta sección puede encontrarla en su memoria [28].





Capítulo 6

Presupuesto

*"No estimes el dinero en más ni en menos de lo
que vale, porque es un buen siervo y un mal
amo."*

Alejandro Dumas (1803-1870)

6.1 Diagrama de tareas

Este proyecto ha sido realizado en un tiempo aproximado de un año, durante el cual los proyectantes lo han compaginado con estudios y becas. La elaboración del proyecto se llevó a cabo en diferentes fases, empezando con un planteamiento de la idea principal del proyecto, seguido de la realización del diseño del sistema, para continuar con el desarrollo. Una vez terminada la implementación, se probó el sistema y se realizó la documentación pertinente.

Se muestra a continuación una lista detallada de tareas y su correspondiente diagrama de Gantt.

	Nombre de tarea	Duración
	1 <input type="checkbox"/> Análisis del problema	40 días
	2 Recopilación de información sobre somniloquias	7 días
	3 Estudio de aplicaciones Android	10 días
	4 Estudio de herramientas de transcripción	13 días
	5 Recopilación de información sobre análisis de emociones	10 días
	6 Definición del alcance	2 días
	7 <input type="checkbox"/> Diseño	14 días
	8 Diseño arquitectónico	7 días
	9 Diseño detallado	7 días
	10 <input type="checkbox"/> Implementación	177 días
	11 Aprender Android	7 días
	12 Módulo aplicación	170 días
	13 Módulo detección SPL	63 días
	14 Módulo gestión bases de datos Android	10 días
	15 Módulo eliminación ruidos	40 días
	16 Módulo de transcripción	44 días
	17 Módulo de análisis de emociones	83 días
	18 Aprender J2EE	9 días
	19 Módulo de comunicación	22 días
	20 Módulo Servlet	30 días
	21 Módulo de gestión de bases de datos Servidor	7 días
	22 Pruebas	2 días
	23 Documentación	28 días

Ilustración 40. Tareas del proyecto

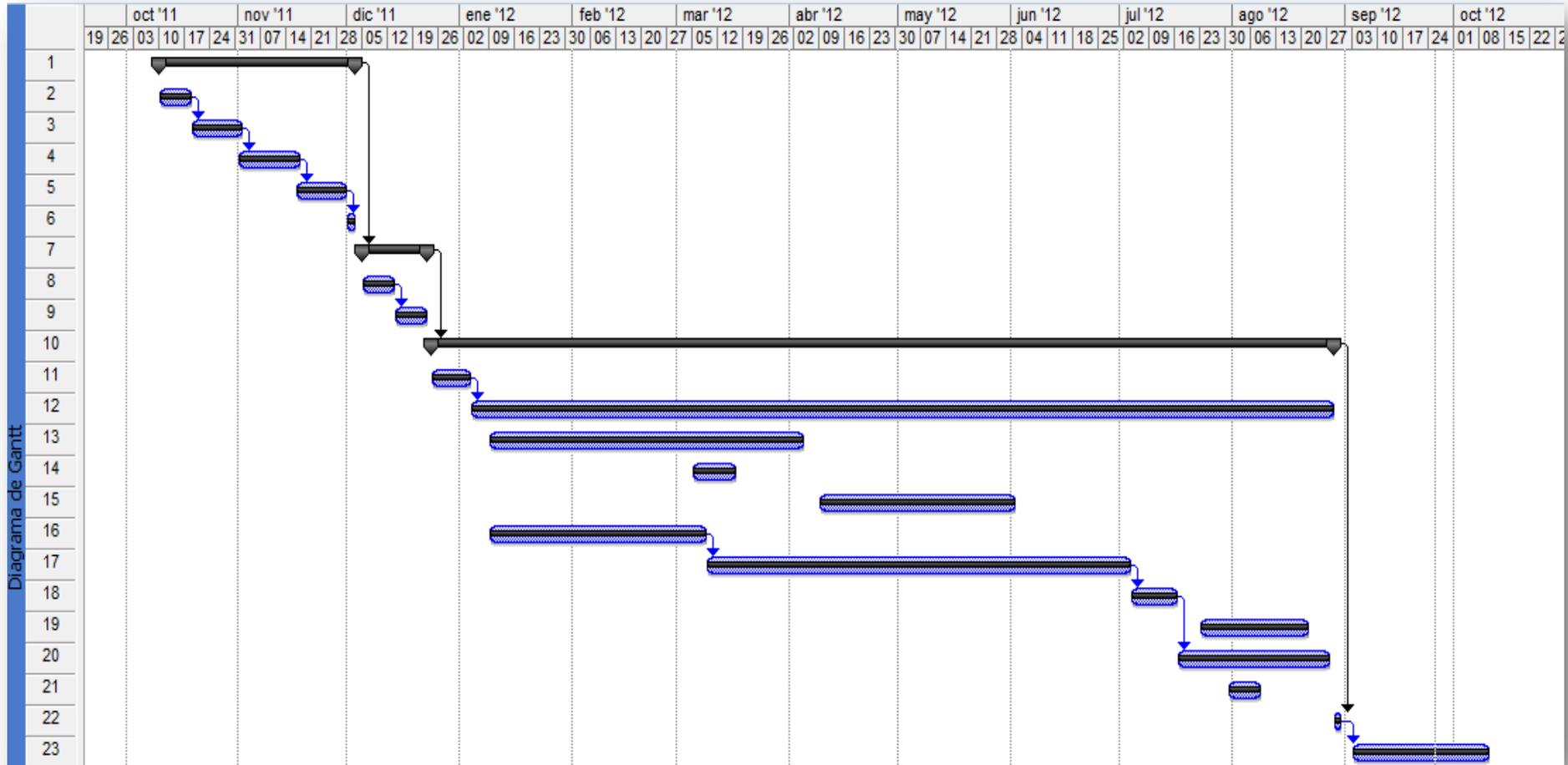


Ilustración 41. Diagrama de Gantt

6.2 Desglose de costes

Como se comentó anteriormente, la duración del proyecto ha sido de aproximadamente 12 meses, con una media de trabajo de cuatro horas diarias. Si contamos los días laborales de este periodo nos salen 251 días que si se multiplican por las horas de trabajo al día y el número de proyectantes nos sale un total de **2008 horas** de trabajo.

6.2.1 Personal

Los costes del personal para el proyecto ascienden a **24.096 €** (veinticuatro mil noventa y seis euros), calculados a partir del coste de la hora para un ingeniero en informática sin experiencia, situado en 12 €/h (doce euros/hora), multiplicado por las 2008 h. (dos mil ocho horas) dedicadas a la elaboración del proyecto. Para el cálculo anterior no se han tenido en cuenta los gastos derivados de la contratación de personal como puede ser las cantidades a pagar al sistema de Seguridad Social español.

6.2.2 Material

En este apartado se detalla el coste de los materiales empleados para la realización del proyecto. Estos costes se pueden dividir en equipos y licencias.

Equipos

Equipo	Precio
Ordenador portátil Toshiba Satellite L655	599,99 €
Ordenador portátil Toshiba Satellite A300	699,99 €
Teléfono móvil Samsun Galaxy i9000	292,72 €
Micrófono ordenador	5,70 €
Total	1598,40 €

Tabla 198. Coste de los equipos

El total gastado en equipos asciende a la cantidad de 1598,40 € (mil quinientos noventa y ocho euros y cuarenta céntimos).

Licencias

Concepto	Unidades	Precio/unidad	Coste
Windows 7 Professional (*)	1	0,00 €	0,00 €
Windows 7 Home Premium	1	199,99 €	199,99 €
Microsoft Office Word 2007	2	229 €	458 €
Microsoft Excel 2007 (*)	2	0,00 €	0,00 €
Microsoft Office Project 2007 (*)	1	0,00 €	0,00 €
Microsoft Office PowerPoint 2007 (*)	2	0,00 €	0,00 €
SoX	1	0,00 €	0,00 €
Eclipse Helios	2	0,00 €	0,00 €
Eclipse Índigo IDE	2	0,00 €	0,00 €
Android SDK	2	0,00 €	0,00 €
MySQL 5.5	1	0,00 €	0,00 €
SQLite	1	0,00 €	0,00 €
Apache Tomcat	2	0,00 €	0,00 €
Java Platform 7	2	0,00 €	0,00 €
Total			657,99 €

Tabla 199. Coste de las licencias

(*) Las licencias de estos productos *Microsoft* se obtuvieron gracias al acuerdo MSDNAA entre *Microsoft Corp.* y la *Universidad Carlos III de Madrid*.

El total gastado en licencias asciende a la cantidad de **657,99 €** (seiscientos cincuenta y siete euros con noventa y nueve céntimos).

6.1.1 Transporte

Para acudir al lugar de trabajo se empleó el transporte público, para ello se adquirieron abonos de transporte mensuales tipo B1. El coste del abono ha variado en el transcurso del proyecto pasando de 55,50 € (cincuenta y cinco euros y cincuenta céntimos) a 60,90 € (sesenta euros y noventa céntimos) el 1 de septiembre de 2012, por lo que contabilizaremos once meses con el primer precio y uno con el segundo. El total asciende a **1342,8 €** (mil trescientos cuarenta y dos euros y ochenta céntimos).

6.1.2 Resumen de costes

Calculando la suma económica necesaria para cubrir los costes derivados de los apartados anteriores, se necesitan **27.695,19 €** (veintisiete mil seiscientos noventa y euros y diecinueve céntimos) para realizar el proyecto. Sin embargo, esta cantidad no incluye el impuesto sobre valor añadido ni los márgenes de beneficio y riesgo asociados a este tipo de proyectos.

Concepto	Cantidad
Personal	24.096 €
Equipos	1.598,40 €
Licencias	657,99 €
Transporte	1.342,8 €
Total	27.695,19 €

Tabla 200. Resumen de los costes

6.1.3 Totales

Para terminar se muestra el precio final del proyecto.

Concepto	Cantidad
Coste total	27.695,19 €
Riesgo (20%)	5.539,04 €
Beneficio (20%)	5.539,04 €
Total sin I.V.A.	38.773,27 €
I.V.A. (21%)	8.142,39 €
Total	46.915,66 €

Tabla 201. Coste total



Teniendo en cuenta los costes desglosados en los apartados anteriores, el presupuesto total de este proyecto asciende a la cantidad de **46.915,66 € (cuarenta y seis mil novecientos quince euros y sesenta y seis céntimos)**.

Leganés, Octubre de 2012

Los ingenieros proyectistas Virginia Aparicio Paniagua y Jorge Pérez Muñoz.

Fdo.

Virginia Aparicio Paniagua

Jorge Pérez Muñoz





Capítulo 7

Conclusiones

*"La llave del éxito es el conocimiento del valor de
las cosas."*

John Boyle O'Reilly (1844-1890)

En este punto nos gustaría comentar una serie de conclusiones que hemos obtenido con la realización del presente proyecto:

- El sistema ha sido desarrollado de manera modular, de forma que cada una de las funcionalidades cuenta con un paquete propio. Esto favorece a la implantación de nuevas funcionalidades y a la eliminación de algún módulo ya existente.
- Para la grabación de somniaquias se ha creado un modelo de grabación inteligente que obtiene únicamente las palabras que una determinada persona haya dicho durante el proceso del sueño. Esto aporta mejoras frente a la grabación tradicional, ya que permite ahorrar en tamaño de almacenamiento, debido a que se descartan los periodos de silencio.
- Se han conseguido eliminar los ruidos que no corresponden a palabras. Junto a la grabación inteligente, esto permite reducir la duración de los audios, por lo que el usuario que desee oír sus grabaciones perderá menos tiempo en este proceso.
- El sistema dispone de un módulo de análisis acústico que funciona de manera eficiente, debido a que agiliza el proceso, eliminando aquellas frecuencias que no corresponden a la voz humana.
- Los resultados obtenidos en el reconocimiento automático de emociones sitúan a la emoción primaria tristeza como la emoción que mejor se diferencia con un resultado de un 80% de aciertos, seguido del enfado con un 64,71%, terminando con la alegría y el miedo con un 35,29%.
- Para utilizar el sistema se debe disponer de un dispositivo móvil con el sistema operativo Android. El problema es que en estos momentos el servidor es local, por lo que solamente funciona si el móvil está conectado a la misma red que el servidor.
- Si se desea implantar la parte del servicio web en un servidor externo, se pide como requisito que funcione con Apache Tomcat y que cuente con una base de datos MySQL.
- La transcripción funciona en estos momentos para sistemas operativos Windows, siendo muy fácil su implantación en Linux, debido a que únicamente deberíamos instalar el software de SoX.



- Para el reconocimiento de emociones no se dispone de gran cantidad de herramientas que se dediquen a esta labor. Además las herramientas existentes no son de código libre por lo que la implementación en esta área está muy acotada.
- El sistema puede servir a profesionales del área de la psicología como herramienta de ayuda para diagnosticar ciertas patologías, como la ansiedad o el estrés.





Capítulo 8

Líneas futuras

*"Sólo cerrando las puertas detrás de uno se abren
ventanas hacia el porvenir."*

Françoise Sagan (1935-2004)

A continuación se expondrán las ideas principales que podrían ser aplicadas en un futuro al sistema desarrollado en este proyecto.

En primer lugar, es importante resaltar que la funcionalidad que ofrece el sistema propuesto en este proyecto es abundante, siendo receptor de un sinfín de funcionalidades más. Las opciones que ofrece la aplicación móvil se ven enormemente potenciadas por las que presenta la plataforma web.

El diseño de la arquitectura cliente-servidor requiere una comunicación constante y necesaria entre ambas partes. Esto implica que, actualmente, ambos componentes sean dependientes uno del otro, por lo que si el servidor no se encuentra activo, la aplicación móvil no podrá realizar funcionalidad alguna. Como línea futura, se propone la opción de trabajar “*offline*”, de forma que la aplicación móvil grabe las somniaquias, y que la recepción de los archivos procesados por el servidor se realice una vez éste se encuentre activo.

Unido a lo anterior, se podría ofrecer al usuario la posibilidad de decidir qué archivos desea que sean procesados en el servidor y cuáles no. Tal y como se ha comentado en algunos puntos del documento, el envío y posterior procesado de las somniaquias se realiza automáticamente tras la creación de un nuevo archivo de audio. En lugar de esto, sería interesante ofrecer al usuario la elección de los archivos a procesar en el servidor.

También podrían mejorarse algunas opciones que el sistema ya presenta:

- Se podría dar la posibilidad al usuario de elegir el formato de sus archivos de audio.
- La reproducción de los ficheros de audio podría presentar una línea de tiempo para que el usuario pudiese trasladarse a cualquier punto dentro del archivo.

Por otra parte, sería conveniente que la visualización de los resultados generados por el servidor se pudiese realizar desde una página web. Es decir, como funcionalidad añadida a la existente, se podría crear una página web donde cada usuario pudiese entrar en ella con su nombre de usuario y contraseña. Dentro de su perfil podría visualizar los resultados de una forma más limpia, ordenada y con opciones adicionales, como por ejemplo:

- Descargar las somniaquias y los archivos post-procesados a su disco duro
- Descargar los archivos de resultados que genera el servidor.
- Acceder a datos avanzados resultantes del análisis acústico de sus archivos de audio.



Por último, en una versión posterior del sistema, se podría crear un nuevo perfil de usuario asociado a un profesional de la psicología. Teniendo en cuenta que el sistema propuesto puede ser de gran ayuda a un psicólogo en su labor de ayuda a sus pacientes, este perfil podría acceder a toda la información de éstos.





Capítulo 9

Glosario de términos y acrónimos

Daría todo lo que sé, por la mitad de lo que ignoro.

René Descartes (1596-1650)

A continuación se explican términos y acrónimos que aparecen o se utilizan en el presente proyecto.

9.1 Glosario

Término	Definición
Android	Sistema operativo basado en Linux para dispositivos móviles tales como los teléfonos inteligentes o tablets.
Base de datos	Una base de datos es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso.
Contenedor de servlet	Entorno de ejecución para componentes web que incluye seguridad, concurrencia, gestión del ciclo de vida, procesamiento de transacciones, despliegue y otros servicios.
Eclipse	Entorno de desarrollo integrado de código abierto multiplataforma que cuenta con una plataforma de programación, desarrollo y compilación de elementos.
Framework	Estructura conceptual y tecnológica de soporte definida, normalmente con artefactos o módulos de software concretos, con base en la cual otro proyecto.
Java	Lenguaje de programación orientado a objetos, desarrollado por Sun Microsystems.
Parasomnia	Trastorno de la conducta durante el sueño asociado con episodios breves o parciales de despertar, sin que se produzca una interrupción importante del sueño ni una alteración del nivel de vigilia diurno.
Servidor	Es una computadora que, formando parte de una red, provee servicios a otras computadoras denominadas clientes.
Servlet	Los servlets son objetos que corren dentro y fuera del contexto de un contenedor de servlets y extienden su funcionalidad.
Sistema Operativo	Es un conjunto de programas que en un sistema informático gestiona los recursos de hardware y provee servicios a los programas de aplicación.

Somniquia	Parasomnia referida al hablar en voz alta durante el sueño. Puede ser bastante fuerte, variar entre simples sonidos hasta largos discursos, y ocurrir una o varias veces durante el sueño.
Transcripción	En sentido lingüístico, es la representación sistemática de la lengua en forma escrita

Tabla 202. Glosario

9.2 Acrónimos

Acrónimo	Significado
API	Application Programming Interface. Conjunto de funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
ASR	Automatic Speech Recognition. Es una parte de la Inteligencia Artificial que tiene como objetivo permitir la comunicación hablada entre seres humanos y computadoras.
AWT	Abstract Windowing Toolkit. Es un kit de herramientas de gráficos, interfaz de usuario, y sistema de ventanas independiente de la plataforma original de Java.
CGI	Common Gateway Interface. Es una importante tecnología de la World Wide Web que permite a un cliente solicitar datos de un programa ejecutado en un servidor web.
DDR2	Double Data Rate. Es un tipo de memoria RAM.
GB	Un gigabit es una unidad de medida de almacenamiento informático normalmente abreviada como Gb, que equivale a 10^9 bits.
GHZ	El Gigahercio es un múltiplo de la unidad de medida de frecuencia hercio y equivale a 10^9 hercios.
HMM	Hidden Markov Model. Es un modelo estadístico en el que se asume que el sistema a modelar es un proceso de Márkov de parámetros desconocidos.
HNR	Harmonics to Noise Ratio. Razón-armónico-ruído



HTTP	Hypertext Transfer Protocol. Es el protocolo usado en cada transacción de la World Wide Web.
IDE	Integrated Development Environment. Es un programa informático compuesto por un conjunto de herramientas de programación.
IOS	(Anteriormente denominado iPhone OS) es un sistema operativo móvil de Apple.
J2EE	Java 2 Platform, Enterprise Edition. La plataforma proporciona una API y entorno de ejecución para el desarrollo y ejecución de software empresarial.
JDBC	Java Database Connectivity. Es una API que permite la ejecución de operaciones sobre bases de datos desde el lenguaje de programación Java.
JDK	Java Development Kit. Es un software que provee herramientas de desarrollo para la creación de programas en Java. Puede instalarse en una computadora local o en una unidad de red.
JVM	Java Virtual Machine. Es una máquina virtual de proceso nativo, capaz de interpretar y ejecutar instrucciones expresadas en bytecode de Java.
JRE	Java Runtime Environment. Es un conjunto de utilidades que permite la ejecución de programas Java.
NHR	Noise-to-Harmonics-Ratio. Razón-ruido-armónico
NREM	Non-Rapid Eye Movement. Engloba las fases 1-4 del sueño y es conocido como sueño lento.
RAH	Reconocimiento Automático del Habla. Es una parte de la Inteligencia Artificial que tiene como objetivo permitir la comunicación hablada entre seres humanos y computadoras.
RAM	Random-Access Memory. se utiliza como memoria de trabajo para el sistema operativo, los programas y la mayoría del software.
REM	Rapid Eye Movements. Es la quinta fase del sueño y se caracteriza por un rápido y movimiento de los ojos.



RMI	Remote Method Invocation. Mecanismo ofrecido por Java para invocar métodos de manera remota. Se utiliza para la comunicación entre servidores y aplicaciones distribuidas basadas en Java.
SDK	Software Development Kit. Es un conjunto de herramientas de desarrollo de software que le permite al programador crear aplicaciones.
SPL	Sound Pressure Level. Se define como la diferencia de presión instantánea y la presión atmosférica estática.
UML	Lenguaje Unificado de Modelado. Es el lenguaje de modelado de sistemas software más conocido y utilizado en la actualidad.
XML	eXtensible Markup Language. Es un lenguaje de marcas y se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas.
WAV	WAVEform audio file format. Formato de fichero de audio con forma de onda.
ZCR	Zero Crossing Rate. Tasa de cruces por cero.

Tabla 203. Acrónimos





Capítulo 10

Bibliografía y referencias

*"De razones vive el hombre, de sueños sobrevive."
Miguel de Unamuno (1864-1936)*



[1] AASM. *The International Classification of Sleep Disorders. Diagnostic and coding manual*. 2. AASM, Westchester, 2005.

[2] A.M. Arkin, M.F. Toth, J. Barker and J.M. Haste, *The frequency of sleeptalking in the laboratory among chronic sleeptalked and good dream recallers*. J. Nerv. Ment. 1970.

[3] A. Rechtschaffen, D.R. Goodnough and A. Shapiro, *Patterns of sleeptalking*. Arch. Gen. Psychiatr., 1962.

[4] G. Andriani, *Fisiología psicológica del sonniloquio*. Ann. Neurol, 1892.

[5] C. Darwin, *The Expression of the Emotions in Man and Animals*, 1872.

[6] Ernst Heinrich Weber, Gustav Theodor Fechner. *Elementos de Psicofísica*, 1860

[7] Instituto de Salud Pública. Gobierno de Chile. *Instructivo para la aplicación del D. S N° 594/99 del Minsal, Título IV, Parrafo 3° Agentes Físicos – Ruido*, 1999.

[8] Federico Miyara, *La voz humana*, 2001

[9] Ricardo Alzate Castaño, *Estimación de contornos del pitch en línea sobre DSP*, 2003

[10] F. Alexander Sepúlveda, Germán Castellanos, *Estimación de la frecuencia fundamental de señales de voz usando transformada wavelet*, 2004

[11] Código fuente SPL Meter. Disponible en: <http://code.google.com/p/splmeter/>
(En línea. Accedido el 8 de Octubre de 2012)

[12] Juan L. Rojas, Luis F. Crespo, Daniel Sánchez Morillo, Rüdiger Daub, *Sensores y señales para la detección de la apnea del sueño*, 2004. Disponible en: http://www.aurova.ua.es:8080/ja2005/comu/4601-JLR_JA05_alicante.pdf (En línea. Accedido el 8 de Octubre de 2012)

[13] Wilson K, Stoohs RA, Mulrooney TF, Johnson LJ, Guilleminault CH, Huang Z. *The snoring spectrum. Acoustic assessment of snoring sound intensity in 1139 individuals undergoing polysomnography*, 1999.

[14] Fiz JA, Abad J, Jané R, Riera M, Mañanas MA, Caminal P, et al, *Acoustic analysis of snoring sound in patients with simple snoring and obstructive sleep apnoea*, 1996.



- [15] Miyazaki S, Itasaka Y, Ishikawa K, Togawa K. *Acoustic analysis of snoring and the site of airway obstruction in sleep related respiratory disorders*, 1998.
- [16] Beck R, Odeh M, Oliven A, Gavriely N. *The acoustic properties of snores*, 1995.
- [17] Meslier N, Auregan Y, Badatcheff A, Depollier C, Racineux JL. *Spectral analysis of snores in patients with obstructive sleep apnea syndrome*, 1990.
- [18] Pérez-Padilla JR, Slawinski E, Difrancesco LM, Feige RR, Remmers JE, Whitelaw WA. *Characteristics of the snoring noise in patients with and without occlusive sleep apnea*, 1993.
- [19] Sola` -Soler J, Jané R, Fiz JA, Morera J. *Spectral envelope analysis in snoring signals from simple snorers and patients with obstructive sleep apnea*, 2003.
- [20]Stoohs R, Guilleminault CH. *Mesam 4: An ambulatory device for the detection of patients at risk of obstructive sleep apnea syndrome (OSAS)*, 1992.
- [21] Perez-Padilla, J.R. West, P. Kryger, M. *Snoring in normal young adults: prevalence in sleep stages and associated changes in oxygen saturation, heart rate, and breathing pattern*, 1987.
- [22] Hoffstein, V. Mateika, S. Anderson D. *Snoring: Is it in the ear of the beholder?*, 1994.
- [23] Wilson, K. Stoohs, R.A. Mulrooney, T.F. Johanson LJ, Guilleminault C, Huang Z. *The snoring spectrum*, 1999.
- [24] Oracle. *Página de la documentación de Java*. Disponible en: <http://www.oracle.com/technetwork/java/javase/documentation/index.html> (En línea. Accedido el 8 de Octubre de 2012)
- [25] Google, Activity lifecycle. Disponible en: <http://developer.android.com/guide/topics/fundamentals.html#actlife>. (En línea. Accedido el 8 de Octubre de 2012)
- [26] FFT.java from §9.7 Data Analysis. Disponible en: <http://introcs.cs.princeton.edu/java/97data/FFT.java.html> (En línea. Accedido el 8 de Octubre de 2012)
- [27] SQLite Data Browser. Disponible en <http://sqlitebrowser.sourceforge.net/> (En línea. Accedido el 8 de Octubre de 2012)
- [28] V. Aparicio Paniagua. *Memoria*. Leganés, Octubre 2012.
- [29] GONZÁLES, M. J y T. Cervera, *Análisis acústico de la voz: Fiabilidad de un conjunto de parámetros multidimensionales*, 2002.



[30] Coifman, R. and Wickerhauser, M., *Entropy based algorithms for best bases selection, Theory*, 1992.

[31] Isabel Carmona Jover, *Ecuaciones Diferenciales*, 1996.





Capítulo 11

Anexos

*"Allí donde el agua alcanza su mayor
profundidad, se mantiene más en calma."*

William Shakespeare (1564-1616)

11.1 Anexo 1

En esta sección se incluye el manual de usuario de la aplicación desarrollada en el presente proyecto.

Introducción

Este manual le permitirá aprender a utilizar las funcionalidades básicas de la aplicación “Somniloquia”.

¿Cómo acceder a Somniloquia?

Instale la aplicación utilizando el archivo *somniloquia.apk* en un dispositivo móvil con sistema operativo Android 2.3.3 ó superior.

Una vez instalada la aplicación, pulse sobre el icono para acceder a la misma.

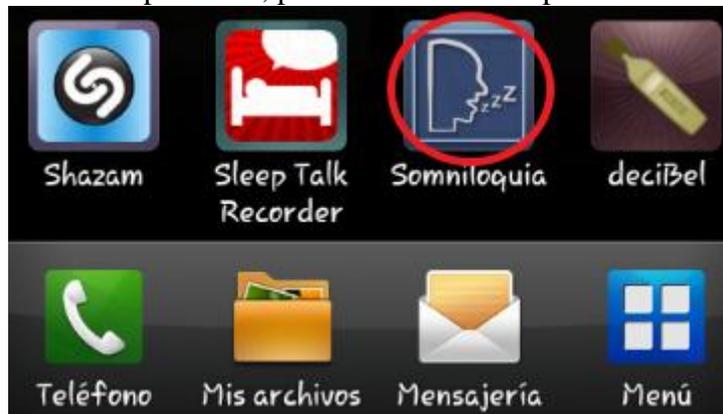


Ilustración 42. Acceder a la aplicación

La primera pantalla que va a aparecer en la aplicación es la pantalla de *Iniciar sesión*.

Iniciar Sesión

En esta pantalla se tienen dos campos de texto donde podrá introducir su nombre de usuario y su contraseña. Asegúrese de que está conectado a internet y pulse el botón “Entrar”. Si los datos son correctos, la aplicación le llevará a la pantalla principal.

En caso de que usted no se haya registrado en la aplicación podrá hacerlo pulsando el botón “Nuevo Usuario”.



Ilustración 43. Login

Nuevo Usuario

En esta pantalla se observan tres campos de texto a rellenar por el usuario. El nombre de usuario debe ser único y las contraseñas deben coincidir.

Debe pulsar el botón “Siguiente” para continuar. En la próxima pantalla podrá configurar el nivel de presión sonora adecuado para usted de forma automática.



Ilustración 44. Nuevo Usuario

Detección del nivel de presión sonora

Siga las instrucciones que aparecen en pantalla. Usted deberá permanecer en silencio y pulsar el botón que aparece centrado en la pantalla.

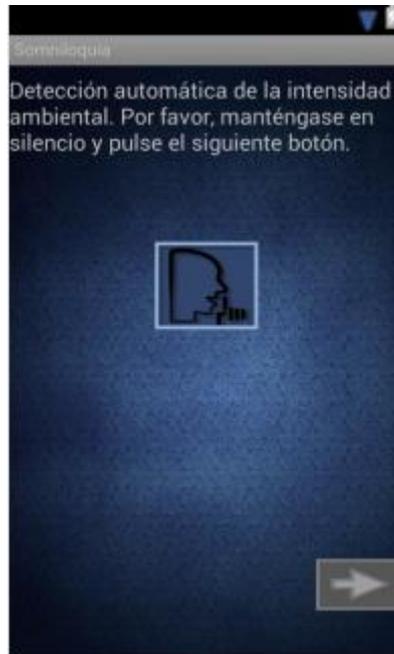


Ilustración 45. Detección del SPL en silencio

Tras cinco segundos el sistema le avisará de que puede continuar. Pulse la flecha que aparece en la parte inferior de la pantalla para proceder a la detección del nivel de presión sonora del habla.



Ilustración 46. Detección del SPL del habla

Siga las instrucciones que aparecen en pantalla. Usted deberá comenzar a hablar, pulsar el botón que aparece centrado en la pantalla y continuar hablando durante cinco segundos. Pare de hablar cuando aparezca el siguiente mensaje en pantalla:



Ilustración 47. Enviando SPL al servidor

Transcurrido el tiempo de detección el sistema procesará la información. Espere unos segundos y pulse la flecha que aparece en la parte inferior de la pantalla para acceder a la pantalla principal.

Pantalla principal

Esta pantalla contiene tres pestañas distintas en las que se encuentra la funcionalidad principal de la aplicación.

Pestaña de Grabación y Reproducción

En la primera pestaña el usuario puede pulsar el botón de la parte superior de la pantalla para comenzar la grabación de las somniloquias. El sistema detectará los momentos de sonido y comenzará la grabación, creando distintos ficheros de audio.



Ilustración 48. Pestaña de Grabación y Reproducción

Cuando desee parar la grabación podrá hacerlo pulsando el primer botón de “stop”.



Ilustración 49. Parar la grabación

Si quiere reproducir una de sus somniloquias tendrá que pulsar el botón que aparece en el medio de la pantalla. La aplicación mostrará un listado de sus carpetas:



Ilustración 50. Carpetas de usuario

Si accede a una de esas carpetas y selecciona un archivo, comenzará la reproducción del mismo.



Ilustración 51. Seleccionar archivo

Podrá pausar la reproducción en el botón del medio “pause”.

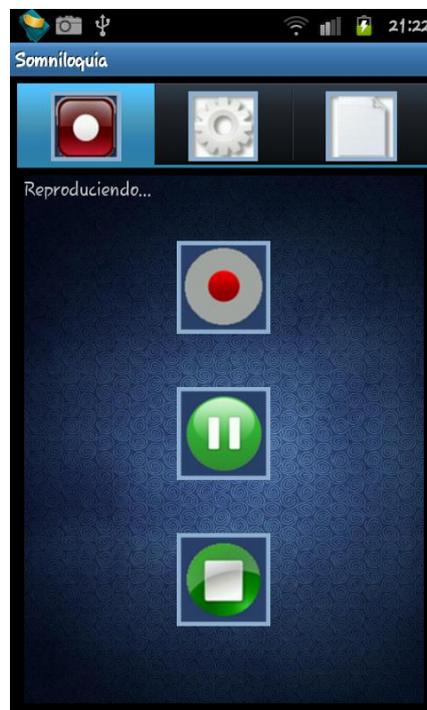


Ilustración 52. Pausar la reproducción

Si, por el contrario, quiere parar la reproducción, deberá presionar el último botón que aparece en pantalla.

Pestaña de Configuración

En la pestaña de configuración el usuario podrá realizar las siguientes acciones:

- Realizar la detección automática del SPL.
- Configurar el tiempo en silencio, utilizado para la duración de los tiempos de silencio en los ficheros de audio generados en la grabación.
- Borrar el usuario.



Ilustración 53. Pestaña de Configuración

Si pulsa sobre el botón de “Detección Automática”, la aplicación le llevará a la pantalla de detección de silencio para realizar de nuevo la configuración del SPL.

También podrá introducir el tiempo en silencio en el cuadro de texto que aparece en pantalla. Es necesario que siga el formato que indica, es decir, dos dígitos para los minutos y dos dígitos para los segundos. El valor que introduzca tendrá que encontrarse entre 0 y 60 minutos.

Si pulsa el botón guardar, se actualizará la información.

Por último, en esta pantalla tiene la opción de borrar su perfil de usuario. Si pulsa el botón “Borrar Usuario”, la aplicación le mostrará el siguiente mensaje:

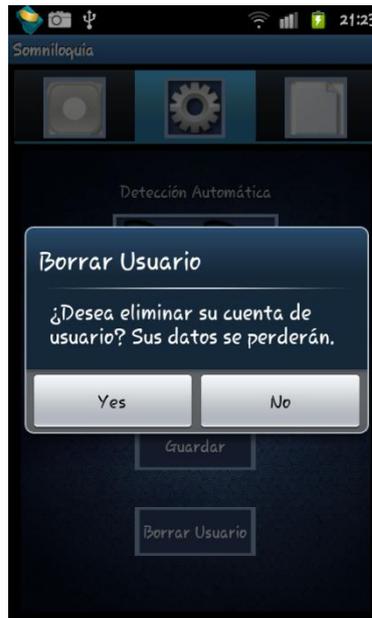


Ilustración 54. Borrar Usuario

Si pulsa “Sí”, saldrá de la aplicación, sus datos de usuario se perderán y no podrá volver a iniciar sesión con esta cuenta. Si pulsa “No” la aplicación continuará con normalidad en la pestaña de “Configuración”.

Pestaña de Resultados

En la pantalla de resultados el usuario podrá acceder al listado de archivos de texto que el servidor ha generado a partir de los archivos de audio originales. Si selecciona un fichero podrá observar la siguiente información:

- Nombre del archivo
- Frecuencia fundamental
- Emoción predominante
- Transcripción

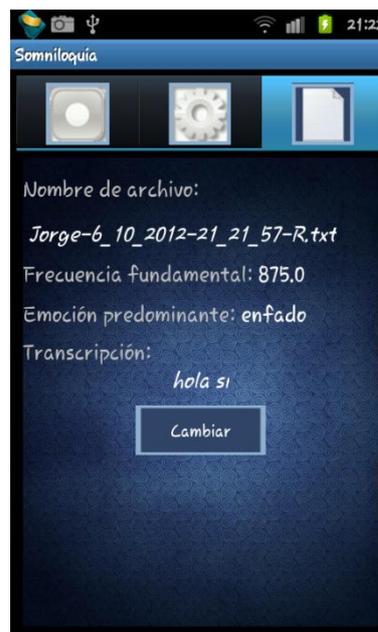


Ilustración 55. Pestaña de Resultados

Podrá cambiar a otro archivo pulsando el botón de “Cambiar”.

Menú de opciones

Si el usuario pulsa sobre el botón de opciones podrá realizar alguna de estas dos opciones: Cerrar sesión y Ayuda. En la siguiente pantalla se puede ver el Menú de Opciones en la parte inferior de la pantalla.



Ilustración 56. Menú Opciones

Si pulsa el botón de “Cerrar Sesión” la aplicación mostrará el siguiente mensaje:



Ilustración 57. Cerrar Sesión

Si pulsa el botón “Sí”, saldrá de la aplicación y volverá a la pantalla de “Inicio de Sesión”. Si pulsa “No” la aplicación continuará con normalidad en la pantalla principal.

Si pulsa el botón de “Ayuda” del menú de opciones, se mostrará la siguiente pantalla con la información de ayuda al usuario.

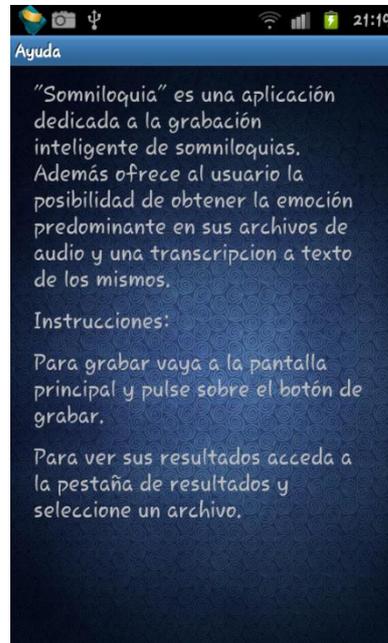


Ilustración 58. Pantalla de Ayuda