

**ARQUITECTURA BASADA EN CONTEXTO PARA EL SOPORTE DEL
SERVICIO DE VOD DE IPTV MÓVIL, APOYADA EN SISTEMAS DE
RECOMENDACIONES Y STREAMING ADAPTATIVO**



M.Sc. GABRIEL ELÍAS CHANCHÍ G.

Tesis de Doctorado en Ingeniería Telemática

**Director:
PhD. José Luis Arciniegas H.**

**Universidad del Cauca
Facultad de Ingeniería Electrónica y Telecomunicaciones
Departamento de Telemática
Línea de investigación en servicios avanzados de telecomunicaciones
Popayán, Noviembre de 2017**

GABRIEL ELÍAS CHANCHÍ GOLONDRINO

**ARQUITECTURA BASADA EN CONTEXTO PARA EL
SOPORTE DEL SERVICIO DE VOD DE IPTV MÓVIL,
APOYADA EN SISTEMAS DE RECOMENDACIONES Y
STREAMING ADAPTATIVO**

**Tesis presentada a la Facultad de Ingeniería
Electrónica y Telecomunicaciones de la
Universidad del Cauca para la obtención del
Título de:**

**Doctor en:
Ingeniería Telemática**

**Director:
PhD. José Luis Arciniegas Herrera**

Popayán, 2017

*A Dios por sus inmensas bendiciones,
A mi familia por su apoyo y comprensión incondicional
A Samuel, Laura y Lina por ser la motivación diaria
A Magaly por su apoyo y amor diarios*

Resumen estructurado

Dentro de los servicios de distribución de contenidos multimedia de IPTV, se destaca por los beneficios económicos brindados y por la amplia difusión en portales como Youtube, Vimeo y Netflix, el servicio por suscripción de VoD, el cual consiste básicamente en una aplicación que permite el acceso al contenido multimedia en el momento exacto que el usuario lo desee. A pesar de las ventajas del servicio de VoD, este tiene asociadas un conjunto de retos relacionados con el acceso ágil y el consumo adecuado del contenido multimedia por parte de los usuarios, las cuales son tomadas como principal foco de estudio en la presente investigación.

Para abordar los anteriores retos, este trabajo planteó el enriquecimiento de la arquitectura del servicio de VoD, teniendo en cuenta el concepto de contexto en telecomunicaciones y sus dimensiones asociadas (usuario, dispositivo, red y servicio). Así, en este trabajo se propuso como objetivo general: Proporcionar una arquitectura basada en contexto para el soporte del servicio de VoD de IPTV móvil, apoyada en sistemas de recomendaciones y *streaming* adaptativo. Para cumplir este objetivo, se definieron los siguientes objetivos específicos: Adaptar e implementar un entorno para la codificación, difusión y consumo de *streaming* adaptativo, soportado en el estándar DASH; Diseñar un filtro de recomendación para el servicio de VoD de IPTV Móvil, teniendo en cuenta un algoritmo de clasificación e inferencia de contenidos multimedia, de acuerdo a información del contexto de usuario; Diseñar e implementar una arquitectura basada en contexto para el soporte del servicio de VoD de IPTV Móvil, teniendo en cuenta sistemas de recomendaciones y *streaming* adaptativo; Evaluar el funcionamiento del sistema de recomendaciones, y del entorno de transmisión de *streaming* adaptativo, mediante un caso de estudio del servicio de VoD de contenidos multimedia musicales.

Para el cumplimiento de los objetivos propuestos, se organizó la metodología del proyecto en seis paquetes de trabajo (WP), relacionados con los resultados esperados de cada objetivo. Los paquetes de trabajo definidos fueron: construcción de un entorno DASH (WP1), dataset de contenidos musicales de video (WP2), sistema de recomendaciones basado en contexto (WP3), arquitectura para el

servicio de VoD (WP4), evaluación de la arquitectura (WP5) y generación de documentación(WP6). WP1 tuvo como fin la construcción de un entorno extremo a extremo para la codificación, difusión y consumo de *streaming* adaptativo DASH. WP2 estuvo enfocado en la construcción de un dataset de contenidos musicales de video, considerando el modelo de emociones de arousal-valence. WP3 se centró en el diseño y construcción de un sistema de recomendaciones basado en contexto para el servicio de VoD de IPTV. WP4 consideró el diseño e implementación de una arquitectura basada en contexto para el servicio de VoD. WP5 se enfocó en la evaluación de los principales componentes de la arquitectura de alto nivel basada en contexto para el servicio de VoD. Finalmente en WP6 se centró en la generación de la documentación del trabajo de doctorado.

Como contribución principal del trabajo de doctorado se destacan el diseño e implementación de una arquitectura basada en contexto para el servicio de video bajo demanda. Esta arquitectura tiene como aporte dos componentes principales: un entorno de *streaming* adaptativo DASH y un sistema de recomendaciones basado en contexto. El entorno de *streaming* adaptativo permite la transmisión extremo a extremo de *streaming* DASH, para lo cual se construyó una herramienta de codificación denominada DASHWebMConverter. El sistema de recomendaciones basado en contexto, permitió por su parte la recomendación de contenidos multimedia de acuerdo a información del contexto de usuario. En este trabajo se consideraron variables del contexto de tipo fisiológico, a partir de las cuales se recomendaron contenidos multimedia previamente clasificados por emociones. Así, el sistema de recomendaciones fue conformado por un dataset de contenidos multimedia musicales basados en emociones, un sistema hardware-software para la captura e inferencia de emociones y un sistema de recomendaciones clásico.

Como principales conclusiones del presente trabajo de doctorado es importante mencionar que la arquitectura basada en contexto permitió enriquecer las características del servicio de VoD desde las dimensiones de usuario, dispositivo y red, teniendo en cuenta los componentes de *streaming* adaptativo y sistemas de recomendaciones basados en contexto. El componente de entorno de *streaming* adaptativo, demostró ser apto para la implementación de servicios soportados en videostreaming, al seguir de manera adecuada los cambios emulados de ancho de banda. Finalmente, el sistema de recomendaciones basado en contexto mostró ser más preciso en las sugerencias con respecto a los sistemas de recomendaciones clásicos, teniendo en cuenta la emoción de entrada del usuario.

Palabras clave: contexto, DASH, IPTV, VoD, *streaming* adaptativo, sistemas de recomendaciones.

Structured abstract

Within the services of distribution of multimedia contents of IPTV, stands out for the economic benefits provided and for the wide diffusion in portals like Youtube, Vimeo and Netflix, the subscription service of VoD, which basically consists of an application that allows the access to multimedia content at the exact moment the user wishes. In spite of the advantages of the VoD service, this one has associated a set of challenges related to the agile access and the adequate consumption of the multimedia content by the users, which are taken as main focus of study in the present investigation.

In order to address the above challenges, this work proposed the enrichment of the VoD service architecture, taking into account the context concept in telecommunications and its associated dimensions (user, device, network and service). Thus, in this work it was proposed as a general aim: Provide a context-based architecture for the support of the mobile IPTV VoD service, supported by adaptive streaming and recommender systems. To achieve this objective, the following specific aims were defined: Adapt and implement an environment for the coding, diffusion and consumption of adaptive streaming, supported in the DASH standard; Design a recommendation filter for the mobile IPTV VoD service, taking into account an algorithm for classification and inference of multimedia contents, according to user context information; Design and implement a context-based architecture for the support of the Mobile IPTV VoD service, taking into account adaptive streaming and recommender systems; Evaluate the performance of the recommender systems, and the adaptive streaming transmission environment, through a case study of the VoD service of musical multimedia content.

To fulfill the objectives, the methodology of the project was organized into six work packages (WP) related to the expected results of each aim: The work packages defined were: construction of a DASH environment (WP1), video musical content dataset (WP2), context-based recommender system (WP3), architecture of the VoD service (WP4), architecture assessment (WP5) and generation of documentation (WP6). WP1 was aimed at building an end-to-end environment for the coding, broadcasting and consumption of adaptive DASH streaming. WP2 was focused on

the construction of a dataset of musical video content, considering the arousal-valence model. WP3 was focused on the design and construction of a context-based recommender system for the IPTV VoD service. WP4 was aimed in the design and implementation of a context-based architecture for the VoD service. WP5 was focused on assessing the key components of the high-level context-based architecture for the VoD service. Finally in WP6 was focused on the generation of documentation of the doctoral work.

The main contribution of doctoral work was the design and implementation of a context-based architecture for the video-on-demand service. This architecture has two main components: an adaptive DASH streaming environment and a context-based recommender system. The adaptive streaming environment allows the end-to-end transmission of DASH streaming, for which a coding tool called DASHWebMConverter was built. The context-based recommender system, allowed the recommendation of multimedia content according to information of the user context. In this work we considered variables of the physiological type, from which multimedia contents previously classified by emotions were recommended. Thus, the recommender system consisted of a musical multimedia content dataset, a hardware-software system for the capture and inference of emotions, and a classic recommender system.

As the main conclusions of the present doctoral work, it is important to mention that the context-based architecture allowed to enrich the VoD service characteristics from the user, device and network dimensions, taking into account adaptive streaming and recommender system components. The adaptive streaming environment component proved to be suitable for the implementation of services supported in videostreaming, by properly following the emulated changes of bandwidth. Finally, the context-based recommender system proved to be more accurate in the suggestions with respect to the classic recommender systems, taking into account the user input emotion.

Palabras clave: adaptive streaming, context, DASH, IPTV, recommender system, VoD.

Contenido

Lista de Figuras	xi
Lista de Tablas.....	xv
Lista de Ecuaciones	xvii
Capítulo 1	19
1. Introducción	19
1.1 Introducción	19
1.2 Marco conceptual	20
1.2.1 IPTV	20
1.2.2 Video Bajo Demanda	23
1.3 Planteamiento del problema.....	24
1.4 Estado actual del conocimiento	29
1.4.1 Arquitecturas basadas en contexto para servicios de IPTV	30
1.4.2 Sistemas de recomendaciones basados en contexto	33
1.4.3 Entornos de streaming adaptativo basados en el estándar DASH.....	38
1.5 Aportes investigativos	40
1.5.1 Brechas existentes	40
1.5.2 Aportes académicos.....	41
1.6 Objetivos	42
1.6.1 Objetivo General.....	42
1.6.2 Objetivos Específicos.....	43
1.7 Enfoque del trabajo de doctorado.....	43
1.8 Metodología del trabajo de doctorado.....	45
1.9 Cronograma de actividades.....	46
1.10 Estructura de la monografía	48
Capítulo 2.....	51

2. Arquitectura de alto nivel servicio de VoD	51
2.1 Introducción	51
2.2 Características funcionales servicio VoD	54
2.3 Arquitectura de alto nivel.....	59
2.3.1 Vista de negocio.....	59
2.3.2 Vista de contexto	61
2.3.3 Vista funcional.....	64
2.3.4 Vista de implementación.....	68
2.4 Comentarios finales arquitectura VoD.....	70
Capítulo 3	73
3. Escenario de streaming adaptativo DASH.....	73
3.1 Introducción	73
3.2 Streaming adaptativo DASH.....	75
3.3 Exploración de herramientas DASH	78
3.3.1 Video.....	78
3.3.2 Códec	78
3.3.3 WebM	79
3.3.4 MP4Box.....	79
3.3.5 Osmo4.....	79
3.3.6 VLC Player.....	79
3.3.7 Red5	80
3.3.8 Darwin	80
3.3.9 Live555.....	80
3.3.10 Apache.....	80
3.3.11 FFmpeg.....	81
3.3.12 Libwebm.....	81
3.3.13 webm-tools	81
3.3.14 webm-dash-javascript	82
3.4 Selección de herramientas DASH	82
3.5 Escenario de streaming adaptativo DASH.....	83
3.6 Herramienta Dash WebM Converter	86
3.6.1 FFmpeg.....	87
3.6.2 Sample_muxer	87
3.6.3 Webm_dash_manifest	88

3.7 Proceso de codificación de streaming adaptativo	88
3.8 Módulos herramienta de codificación.....	91
3.9 Archivo descriptor MPD	93
3.10 Servicio VoD DASH.....	94
Capítulo 4	99
4. Dataset de contenidos de video.....	99
4.1 Introducción	99
4.2 Procedimiento de generación del dataset.....	100
4.3 Características musicales	101
4.3.1 Arousal (Energía).....	102
4.3.2 Valence (Valencia).....	102
4.3.3 Ritmo	102
4.3.4 Tempo	103
4.3.5 Speechiness	103
4.3.6 Liveness.....	104
4.3.7 Acousticness	104
4.3.8 Danceability	104
4.3.9 Modo.....	104
4.4 Emoción	104
4.4.1 Estructura de las emociones.....	105
4.4.2 Modelo circunflejo.....	106
4.4.3 Modelo de Hevner	106
4.4.4 Modelo tridimensional	107
4.4.5 Modelo de plutchick.....	108
4.5 Soluciones comerciales	109
4.5.1 Musicoverly.....	109
4.5.2 Gracenote	110
4.5.3 EchoNest	110
4.6 Conformación dataset de video	111
4.6.1 Modelo de emociones	112
4.6.2 Reconocimiento de emociones del contenido	113
4.6.3 Diagrama de flujo del dataset de contenidos.....	114
4.7 Evaluación del dataset de contenidos.....	120
Capítulo 5	123

5. Sistema de recomendaciones contextual	123
5.1 Introducción	123
5.2 Sistema de medidas fisiológicas	124
5.3 Variables del contexto de usuario	125
5.3.1 Variabilidad de la frecuencia cardíaca	125
5.3.2 Medida de la respuesta galvánica de la piel	130
5.3.3 Procesamiento de la voz.....	132
5.4 Equipos para medidas fisiológicas	134
5.4.1 Pulse sensor cardíaco-arduino	134
5.4.2 Cinturon Zephyr HxM BT	135
5.4.3 Cinturón Polar	135
5.4.4 Galvanic Skin Response (GSR) sensor.....	136
5.4.5 Arduino Yún	137
5.5 Diseño mecanismo hardware-software.....	138
5.6 Concepto de S.R.	140
5.7 Enfoque clásico de los S.R.	141
5.7.1 Enfoque basado en contenido	141
5.7.2 S.R. colaborativos.....	146
5.8 S.R. basados en contexto.....	147
5.9 Diseño del S.R. basado en contexto.....	148
5.9.1 Diseño del clasificador bayesiano	150
5.9.2 Servicio de VoD basado en emociones.....	152
Capítulo 6	157
6. Evaluación y Pruebas	157
6.1 Introducción	157
6.2 Evaluación del entorno de streaming	158
6.2.1 Prueba de consumo de memoria herramienta DASH Webm Converter.....	158
6.2.2 Pruebas de ancho de banda sobre el escenario de streaming	161
6.3 Evaluación del S.R contextual	164
6.3.1 Evaluación S.R. clásico.....	164
6.3.2 Evaluación S.R. contexto.....	167
6.4 Pruebas de carga y estrés	180
6.4.1 Servidor de Procesamiento	181
6.4.2 Servidor de contexto.....	182

6.4.3 Módulo hardware-software (Arduino Yún).....	184
Capítulo 7.....	187
7. Conclusiones y Trabajos Futuros	187
7.1 Conclusiones	187
7.1.1 Arquitectura basada en contexto	187
7.1.2 Entorno de streaming adaptativo.....	189
7.1.3 Servicio de VoD	191
7.1.3 Sistema de recomendaciones.....	192
7.2 Trabajos futuros	195
7.3 Aportes investigativos	197
7.4 Publicaciones y registros software	198
7.4.1 Publicaciones en Congresos.....	198
7.4.2 Publicaciones en Revistas.....	199
7.4.3 Libros.....	201
7.4.4 Registros Software	201
Referencias	203

Lista de Figuras

<i>Figura 1. 1. Cadena de valor IPTV</i>	21
<i>Figura 1. 2. Servicio de video bajo demanda</i>	24
<i>Figura 1. 3. Retos del servicio de VoD</i>	26
<i>Figura 1. 4. Retos y alternativas del servicio de VoD</i>	29
<i>Figura 1. 5. Temáticas de estudio</i>	30
<i>Figura 1. 6. Metodología del trabajo de doctorado</i>	46
<i>Figura 2. 1. Problemáticas del servicio de VoD</i>	52
<i>Figura 2. 2. Problemáticas del servicio de VoD</i>	52
<i>Figura 2. 3. Fases prueba de usabilidad</i>	54
<i>Figura 2. 4. Vistas de la arquitectura</i>	59
<i>Figura 2. 5. Cadena de valor VoD</i>	61
<i>Figura 2. 6. Dimensión del contexto</i>	62
<i>Figura 2. 7. Variables de contexto escogidas</i>	64
<i>Figura 2. 8. Vista funcional de la arquitectura</i>	64
<i>Figura 2. 9. Secuencia funcional de la arquitectura</i>	67
<i>Figura 2. 10. Interacción módulos cliente y servidor</i>	68
<i>Figura 2. 11. Diagrama de implementación</i>	69
<i>Figura 3. 1. Modelo escenario de streaming adaptativo</i>	77
<i>Figura 3.2. Escenario de streaming adaptativa</i>	84
<i>Figura 3. 3. Cliente web de streaming adaptativo DASH</i>	85
<i>Figura 3. 4. Scripts usados en la herramienta de codificación automática</i>	87
<i>Figura 3. 5. Proceso de codificación de streaming adaptativo</i>	88
<i>Figura 3. 6. Interfaz Gráfica de la herramienta Dash WebM Converter</i>	89
<i>Figura 3. 7. Fase de codificación de los contenidos WebM</i>	90
<i>Figura 3. 8. Recursos de la herramienta de codificación automática</i>	91
<i>Figura 3. 9. Diagrama de bloques herramienta de codificación automática</i>	92
<i>Figura 3. 10. Documento de descripción MPD (cabecera)</i>	93
<i>Figura 3. 11. Documento de descripción MPD (pie)</i>	94
<i>Figura 3. 12. Flujo de trabajo del servicio VoD</i>	95
<i>Figura 3. 13. Flujo de trabajo del servicio VoD</i>	96
<i>Figura 3. 14. Servicio VoD DASH</i>	97
<i>Figura 4. 1. Procedimiento de conformación del dataset</i>	101
<i>Figura 4. 2. Ejemplo patencia de audio</i>	103
<i>Figura 4. 3. Modelos de emociones</i>	105

Figura 4. 4. Modelo circunflejo de las emociones adaptado por Russell.....	106
Figura 4. 5. Modelo de las emociones de Hevner.....	107
Figura 4. 6. Modelo tridimensional de plutchick	108
Figura 4. 7. Interfaz de usuario Musicoverly.....	109
Figura 4. 8. Página web oficial de gracernote.....	110
Figura 4. 9. Página web oficial de EchoNest.....	111
Figura 4. 10. Modelo de arousal-valence adaptado a 5 emociones.....	112
Figura 4. 11. Reconocimiento de emociones musicales.....	113
Figura 4. 12. Bloque funcionales herramienta Java	114
Figura 4. 13. Diagrama de flujo del dataset de contenidos multimedia	114
Figura 4. 14. Librerías utilizadas y llave de EchoNest.....	115
Figura 4. 15. Búsqueda de canciones más populares EchoNest.....	115
Figura 4. 16. Obtención de las características del contenido.....	116
Figura 4. 17. Obtención de la emoción del contenido	117
Figura 4. 18. Método para la obtención de url asociada a cada contenido.....	118
Figura 4. 19. Muestra del archivo JSON generado.....	118
Figura 4. 20. Método TinyDB y generación de archivo JSON.....	119
Figura 4. 21. Descarga de contenidos multimedia.....	119
Figura 4. 22. Arquitectura para el despliegue del servicio de VoD.....	120
Figura 4. 23. Interfaz de usuario del servicio de VoD.....	121
Figura 5. 1. Sistema de medidas fisiológicas	124
Figura 5. 2. Variables del contexto.....	125
Figura 5. 3. Variabilidad de la frecuencia cardíaca.....	126
Figura 5. 4. Glándula sudorípara.....	130
Figura 5. 5. Gráfico señal GSR.....	131
Figura 5. 6. Gráfico regresión lineal a conjunto de datos.....	132
Figura 5. 7. Método para cálculo de regresión lineal.....	132
Figura 5. 8. Dispositivo sensor de pulso cardíaco-arduino.....	134
Figura 5. 9. Dispositivo sensor zephyr.....	135
Figura 5. 10. Dispositivo sensor polar	136
Figura 5. 11. Dispositivo sensor GSR.....	136
Figura 5. 12. Tarjeta hardware Arduino Yún.....	137
Figura 5. 13. Módulos funcionales para las variables fisiológicas	139
Figura 5. 14. Medidas de análisis para la VFC. Fuente: propia	139
Figura 5. 15. Gráfico estados de ánimo asociados a un índice de estrés (SI).....	140
Figura 5. 16. Arquitectura básica del S.R	141
Figura 5. 17. Arquitectura de los S.R. basados en contenido.....	142
Figura 5. 18. Diagrama de secuencia de un S.R. basado en contenido.....	143
Figura 5. 19. Diagrama de secuencia de un S.R. basado en filtros colaborativos.....	146
Figura 5. 20. Diagrama de secuencia de un S.R. basado en contexto.....	147
Figura 5. 21. Estructura funcional el S.R. basado en contexto.....	149
Figura 5. 22. Diseño del S.R. basado en contexto	150
Figura 5. 23. Diagrama final del clasificador bayesiano.....	152
Figura 5. 24. Arquitectura general del servicio de VoD.....	153
Figura 5. 25. Interfaz validación de usuario del servicio de VoD.....	154
Figura 5. 26. Interfaz principal de usuario del servicio de VoD.....	154
Figura 6. 1. Porcentaje de CPU vs tiempo (FFMPEG).....	159

<i>Figura 6. 2. Memoria RAM vs tiempo (FFMPEG)</i>	159
<i>Figura 6. 3. Porcentaje de CPU vs tiempo DASH WebM Converter</i>	160
<i>Figura 6. 4. RAM vs tiempo DASH WebM Converter</i>	160
<i>Figura 6. 5. Script para emulación de ancho de banda</i>	162
<i>Figura 6. 6. Servicio de VoD en la nube</i>	162
<i>Figura 6. 7. Emulación de ancho de banda</i>	163
<i>Figura 6. 8. Resumen de predicciones del clasificador bayesiano</i>	165
<i>Figura 6. 9. Matriz de confusión que modela el clasificador bayesiano</i>	166
<i>Figura 6. 10. Diagrama de evaluación S.R</i>	170
<i>Figura 6. 11. Métricas de comparación vs medidas (media aritmética)</i>	173
<i>Figura 6. 12. Desviación estándar vs medidas</i>	177
<i>Figura 6. 13. Análisis métricas de comparación</i>	180
<i>Figura 6. 14. Pruebas de carga, servidor de procesamiento</i>	181
<i>Figura 6. 15. Pruebas de estrés, servidor de procesamiento</i>	182
<i>Figura 6. 16. Pruebas de carga, servidor de contexto</i>	183
<i>Figura 6. 17. Pruebas de estrés, servidor de contexto</i>	184
<i>Figura 6. 18. Pruebas de carga, Arduino Yún</i>	185
<i>Figura 6. 19. Pruebas de estrés, Arduino Yún</i>	185

Lista de Tablas

<i>Tabla 1. 1. Cronograma de actividades</i>	46
<i>Tabla 2. 1. Componentes hardware-software de la arquitectura</i>	70
<i>Tabla 3. 1. Herramientas escenario de streaming adaptativo DASH</i>	82
<i>Tabla 4. 1. Tabla de estados de emoción, según el rango de ángulos</i>	112
<i>Tabla 4. 2. Características musicales para cada video, archivo JSON</i>	120
<i>Tabla 5. 1. Medidas estadísticas y nivel de estrés</i>	129
<i>Tabla 5. 2. Rangos SI asociados a cada estado</i>	130
<i>Tabla 5. 3. Ejemplo del teorema de Naive Bayes</i>	145
<i>Tabla 5. 4. Ejemplo de clasificador bayesiano con valores booleanos</i>	150
<i>Tabla 5. 5. Ejemplo de normalización con un contenido origina</i>	151
<i>Tabla 5. 6. Ejemplo de normalización con un contenido normalizado</i>	152
<i>Tabla 6. 1. Ejemplo matriz de confusión</i>	166
<i>Tabla 6. 2. Abreviaturas de media aritmética para métricas de comparación</i>	171
<i>Tabla 6. 3. Abreviaturas de desviación estándar para métricas de comparación</i>	175

Lista de Ecuaciones

<i>Ecuación 4. 1. Fórmulas de normalización para arousal y valence.</i>	116
<i>Ecuación 5. 1. Fórmula para calcular RMSSD.</i>	127
<i>Ecuación 5. 2. Fórmula para calcular pNN50</i>	127
<i>Ecuación 5. 3. Fórmula para calcular mRR.</i>	127
<i>Ecuación 5. 4. Fórmula para calcular SDNN.</i>	128
<i>Ecuación 5. 5. Formula Índice de estrés mental.</i>	129
<i>Ecuación 5. 6. Fórmula de probabilidad condicionada de A dado B.</i>	143
<i>Ecuación 5. 7. Fórmula de Naive Bayes.</i>	144
<i>Ecuación 5. 8. Probabilidad condicionada, Naive Bayes.</i>	144
<i>Ecuación 5. 9. Fórmula para la estimación de Laplace.</i>	144
<i>Ecuación 5. 10. Fórmulas para normalizar arousal, valence y tempo.</i>	151
<i>Ecuación 6. 1. Distancia Euclidiana.</i>	168
<i>Ecuación 6. 2. Distancia Manhattan.</i>	168
<i>Ecuación 6. 3. Similitud del Coseno.</i>	168
<i>Ecuación 6. 4. Formula parametrización, similitud del coseno.</i>	169
<i>Ecuación 6. 5. Coeficiente de Correlación de Pearson.</i>	169
<i>Ecuación 6. 6. Coeficiente OLS.</i>	170

Capítulo 1

1. Introducción

1.1 Introducción

Los proveedores de servicios de Internet han incrementado continuamente las velocidades de acceso que proporcionan a sus clientes. Así, gracias a las mejoras en el ancho de banda, ha surgido con gran fuerza el servicio basado en la transmisión de tráfico de audio y vídeo sobre internet, denominado flujo multimedia [1]. De acuerdo a [2], en Norte América el consumo de contenidos de entretenimiento representa más del 68% del tráfico en redes de acceso fijo, en donde solo Netflix constituye el 31.6%. De igual manera en Europa el consumo de entretenimiento en tiempo real para redes de acceso fijo supera el 47.4%, como resultado del incremento en la disponibilidad de servicios, sobre la infraestructura pública de internet o servicios de video Over The Top (OTT). Por su parte, el flujo de contenidos de entretenimiento en América latina representa un 50% en cuanto al tráfico de acceso fijo y el 29% del tráfico de acceso móvil. A partir de lo anterior, y teniendo en cuenta que el 35% del tráfico total de la red de internet en América Latina utiliza HTTP en la capa de aplicación, para su sistema de flujo multimedia, se puede establecer HTTP como el principal protocolo en las redes modernas [3].

Uno de los servicios que está relacionado con las cifras de consumo antes mencionadas, es la televisión IP o IPTV, la cual según la ITU-T (Unión Internacional de Telecomunicaciones) es definida como el envío de servicios multimedia (televisión, video, audio, texto, gráficos y datos) de alta calidad a través de redes IP de banda ancha, haciendo uso de protocolos y/o tecnologías de *streaming*¹. Dentro de la gama de prestaciones que ofrece el servicio de IPTV, se destaca por su gran difusión en internet el servicio de video bajo demanda (VoD), el cual a pesar de sus

¹ El *streaming* es una técnica de transmisión a través de redes de medios continuos (audio y video), en la que se envía información a través de la red y el cliente la reproduce en tiempo real al recibirla.

ventajas, presenta un conjunto de dificultades relacionadas con el acceso ágil y el consumo adecuado del contenido multimedia por parte de los usuarios, las cuales son tomadas como principal foco de estudio en la presente investigación. Dentro del presente trabajo de doctorado se busca aportar a la solución de estos retos del servicio de VoD, considerando el concepto de contexto en el ámbito de las telecomunicaciones y particularmente en lo que se refiere a IPTV [4].

En este capítulo se presentan en primera instancia, un conjunto de conceptos que fueron tenidos en cuenta para el planteamiento y desarrollo del presente trabajo. Dentro de estos se encuentran: IPTV, Video bajo demanda (VoD), contexto. Posteriormente se aborda el planteamiento del problema, en donde se mencionan los principales retos del servicio de VoD y las problemáticas asociadas a esos retos, las cuales sirvieron de base para la formulación de la pregunta de investigación y la hipótesis que motivaron el desarrollo del presente trabajo de doctorado. De igual manera se describen un conjunto de trabajos relacionados con la temática de investigación, los cuales han sido catalogados en 3 áreas temáticas, como son: arquitecturas basadas en contexto para escenarios de televisión, *streaming* adaptativo DASH y sistemas de recomendaciones basados en contexto. Posteriormente y como conclusión de la revisión del estado del arte, se describen los aportes investigativos del presente trabajo, lo cual incluye las brechas de investigación identificadas y los aportes que responden a dichas brechas. Seguidamente, se presentan los objetivos planteados para el desarrollo de este trabajo de doctorado. A continuación se muestran las actividades de investigación asociadas a los objetivos. Finalmente se describe la estructura de la presente monografía.

1.2 Marco conceptual

En esta sección se presentan los conceptos relacionados con IPTV, Video Bajo Demanda, los cuales se tuvieron en cuenta para la formulación del presente trabajo.

1.2.1 IPTV

IPTV es definida por la ITU como el envío de servicios multimedia (televisión, video, audio, texto, gráficos y datos) de alta calidad a través de redes IP de banda ancha, haciendo uso de protocolos y/o tecnologías de *streaming*. La ITU establece que dichos servicios deben ser gestionados para soportar el nivel de calidad de servicio (QoS, Quality of Service), calidad de la experiencia (QoE, Quality of Experience), seguridad, interactividad y confiabilidad [5]. Por su parte, El Foro

Abierto de IPTV (OIPF, Open IPTV Forum²), establece que las redes IP sobre las cuales se entrega el servicio de IPTV pueden ser redes gestionadas extremo a extremo o no gestionadas, es decir sin garantías de calidad de servicio [6]. En el presente trabajo se considerará la definición aportada por el Open IPTV Forum, teniendo como mecanismos de gestión los provistos por el *streaming* adaptativo DASH (Dinamic Adaptive Streaming over HTTP) en el lado del cliente y en el lado del servidor. Adicionalmente, desde la perspectiva del usuario, en cambio, IPTV es otro modelo de negocio de la televisión y por ende otro tipo de televisión por suscripción [7].

Los organismos de estandarización³ de IPTV coinciden en que la cadena de valor de IPTV está formado por cuatro eslabones o dominios: dominio del proveedor de contenidos, dominio del proveedor del servicio IPTV, dominio del proveedor de red y dominio del consumidor (ver figura 1.1). En el dominio del proveedor de contenidos se realiza la producción, edición y metadatos del contenido (películas, series, eventos, documentales, noticias, etc.). El actor de este dominio es el proveedor de contenidos y su rol es desempeñado por una compañía de producción o un distribuidor que tiene licencia para vender contenido y por lo tanto tiene los derechos del recurso audiovisual. De acuerdo a la Figura 1, el proveedor de contenidos le permite el acceso a los contenidos multimedia al dominio del proveedor del servicio IPTV [6].

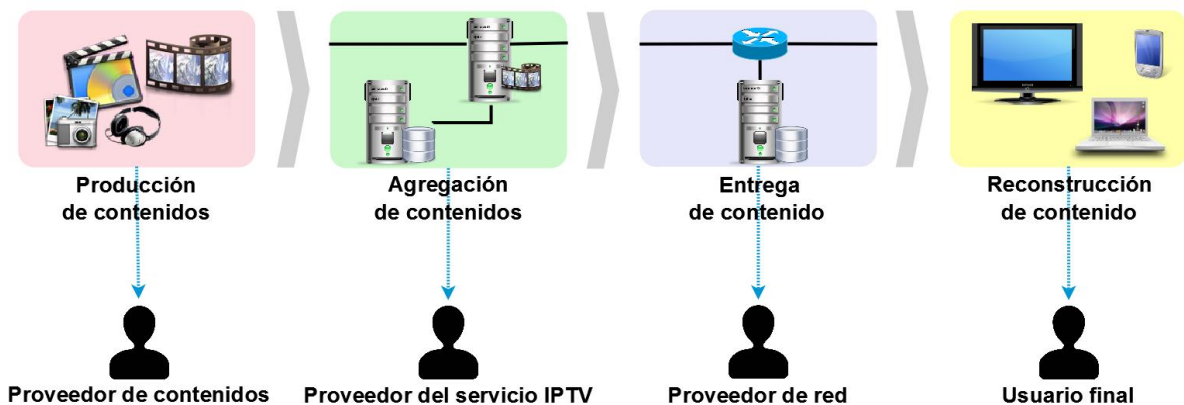


Figura 1. 1. Cadena de valor IPTV. Fuente: propia.

En el dominio del proveedor del servicio, se realiza la agregación de contenido, es decir, se preparan los contenidos para ser enviados hacia el usuario final (codificación y adaptación de los contenidos multimedia), y se proporcionan los servicios de plataforma (por ejemplo, tarificación, autenticación de usuario, etc.) y los complementos para enriquecer el servicio de televisión (por ejemplo, servicios de comunicación, servicios interactivos, publicidad, etc.). Los diferentes tipos de

² OIPF es una asociación abierta conformada por las industrias de las comunicaciones y el entretenimiento (operadores de redes, proveedores de contenido, proveedores de servicios, fabricantes de electrodomésticos y proveedores de dispositivos de infraestructura de red) que tiene como objetivo desarrollar especificaciones extremo a extremo para llevar a la próxima generación de IPTV al mercado masivo (ver <http://www.oipf.tv/>).

³ UIT, TISPAN y OIPF.

servicios de IPTV se clasifican de acuerdo a las características del servicio desde la perspectiva del usuario final en: servicios de distribución de contenido, servicios interactivos, servicios de comunicación y otros servicios [8].

En primer lugar, los servicios de distribución de contenido son los orientados a la difusión de contenido y se clasifican en: servicios de difusión, servicios bajo demanda, servicios con desplazamiento en tiempo y lugar, y servicios de contenido complementario. En los servicios de difusión, los usuarios tienen un control limitado sobre el contenido multimedia que visualizan, ejemplos de estos son: televisión lineal, servicio de pago por ver, televisión lineal con servicio multivisión, etc. En los servicios de video bajo demanda por su parte, el usuario puede acceder al contenido multimedia en el momento que lo desee. En los servicios con desplazamiento en tiempo y lugar, el usuario puede acceder y controlar el contenido multimedia (pausar, rebobinar, adelantar, etc.) suscrito en cualquier lugar y sin limitaciones de tiempo. Finalmente los servicios de contenido complementario, se refieren a mecanismos que pueden ser accedidos de manera opcional por los usuarios a la hora de consumir los contenidos multimedia, con el fin de enriquecer la descripción de los contenidos multimedia, ejemplos de estos mecanismos son: subtítulos y leyendas, descripción de audio para personas con limitaciones visuales, interpretación de lenguaje de señas, etc. [8].

En segundo lugar se encuentran los servicios interactivos que le permiten al usuario enviar diferentes tipos de peticiones al proveedor de servicios y recibir una realimentación, aprovechando las ventajas de la interactividad de IPTV, la cual es entendida como el despliegue de diversos tipos de aplicaciones (administración, salud, educación, entretenimiento, etc.) a usuarios finales, con la particularidad de que el propio usuario puede hacer parte activa del servicio, decidiendo si accede o no a dichas aplicaciones interactivas [9]. Dentro de esta categoría se encuentran servicios de información que soportan varios tipos de contenido (noticias, estados del tiempo, etc.), servicios comerciales, servicios de entretenimiento diseñados para ofrecer diversión al usuario final (juegos, karaoke, blogs, etc.), servicios de aprendizaje para entregar contenido educativo, servicios médicos, servicios de monitoreo y publicidad interactiva [8].

En tercer lugar están los servicios de comunicación que facilitan la comunicación del usuario final con otros usuarios IPTV por medio de mensajería, telefonía, video llamadas o videoconferencias. En cuarto y último lugar se encuentran los servicios que no pueden ser clasificados en ninguna de las categorías anteriores, estos son: servicios de interés público (apoyo a usuarios con discapacidad, comunicación de emergencias, etc.), servicios de alojamiento (por ejemplo: alojamiento de contenido) y servicios de presencia (manejo del estado del usuario: “viendo la televisión”, “viendo un partido de fútbol”, etc.) [8].

Continuando con la cadena de valor, el dominio del proveedor de red se encarga de entregar el contenido multimedia al dominio del consumidor y proporcionar la comunicación entre el dominio del proveedor del servicio IPTV y el dominio del consumidor [6]. A diferencia de la televisión no difundida a través de redes IP (televisión analógica, televisión digital, etc.) donde el proveedor de red transmite al mismo tiempo todos los canales hacia el usuario final, IPTV solo admite el envío del canal que el usuario solicita, conservando el ancho de banda en la red. Cada canal de televisión solicitado hace parte de un grupo *multicast*⁴, de manera que si el usuario desea ir a un nuevo canal, el dispositivo de usuario debe cambiar al grupo asociado al nuevo canal, para que el proveedor compruebe si el usuario está autorizado para acceder al canal requerido y añadir su terminal a la lista de distribución. Así, es posible enviar simultáneamente el servicio de televisión IPTV hacia diversos abonados. De igual forma, existe la necesidad de enviar servicios a un único usuario, debido a que IPTV soporta el flujo de datos *unicast*⁵ [7].

Finalmente, el dominio del consumidor está compuesto por los terminales y dispositivos necesarios para el consumo del servicio IPTV (televisor, computador, decodificador, *Set-Top-Box* STB⁶, dispositivo móvil, etc.), dichos dispositivos permiten recibir y reconstruir el contenido a un formato adecuado que pueda ser desplegado por el dispositivo de usuario final, así como consumir los diferentes servicios asociados al contenido multimedia [6].

Los anteriores dominios no definen un modelo comercial para IPTV, pues los proveedores pueden abarcar varios dominios cuando ofrecen un determinado servicio. Así, de acuerdo al OIPF, para garantizar al usuario final el suficiente ancho de banda para la señal y una alta calidad de servicio en una red gestionada, el actor de los dominios de proveedor de servicio y proveedor de red, debe ser el mismo, así es posible priorizar el tráfico de video para evitar retrasos o fragmentación de la señal [7]. Por el contrario, en un modelo no gestionado, la relación entre los proveedores de servicio y red no está definida necesariamente, es decir, los roles de cada dominio son desempeñados por diferentes proveedores, de manera que para garantizar calidad de servicio en IPTV es necesario coordinar esfuerzos entre los diferentes proveedores y operadores [6].

1.2.2 Video Bajo Demanda

Uno de los servicios más representativos de IPTV, es el servicio de VoD el cual según [1], es entendido como una aplicación que espera, procesa y sirve peticiones

⁴ Es un método de envío simultáneo de paquetes (a nivel de IP) que tan solo serán recibidos por un determinado grupo de receptores, que están interesados en los mismos.

⁵ Es un método de envío de paquetes (a nivel de IP) desde un único emisor a un único receptor.

⁶ Un *Set-Top-Box* es un dispositivo encargado de recibir, demodular y enviar las señales digitales de televisión al televisor. También tiene la funcionalidad de verificar si el televidente tiene el permiso para visualizar la señal y en algunos casos está equipado con mediadores para ejecutar aplicaciones interactivas.

de uno, o varios clientes. La petición, contiene un comando mediante el cual el cliente solicita el vídeo que desea recibir. Cuando el servidor ha recibido el comando de reproducción, empieza a transmitir el vídeo. Así mismo, de acuerdo a [10] este servicio se puede definir como una aplicación que le permite al usuario obtener un contenido multimedia en el momento exacto que lo desee. En la figura 1.2 se presenta un ejemplo de servicio de video bajo demanda obtenido a partir del portal de contenidos multimedia de Youtube. De acuerdo a la figura 2, se pueden identificar 6 componentes fundamentales del servicio: componente de reproducción (1), contador de reproducciones (2), valoración de contenidos (3), descripción del contenido (4), buscador de contenidos (5) y listado de recomendaciones o sugerencias (6). Dentro de los componentes antes mencionados, es importante destacar el de sugerencias o recomendaciones, el cual genera un listado de videos personalizados que tiene como propósito agilizar el proceso de escogencia de un contenido multimedia a partir del catálogo de videos del servicio de VoD.

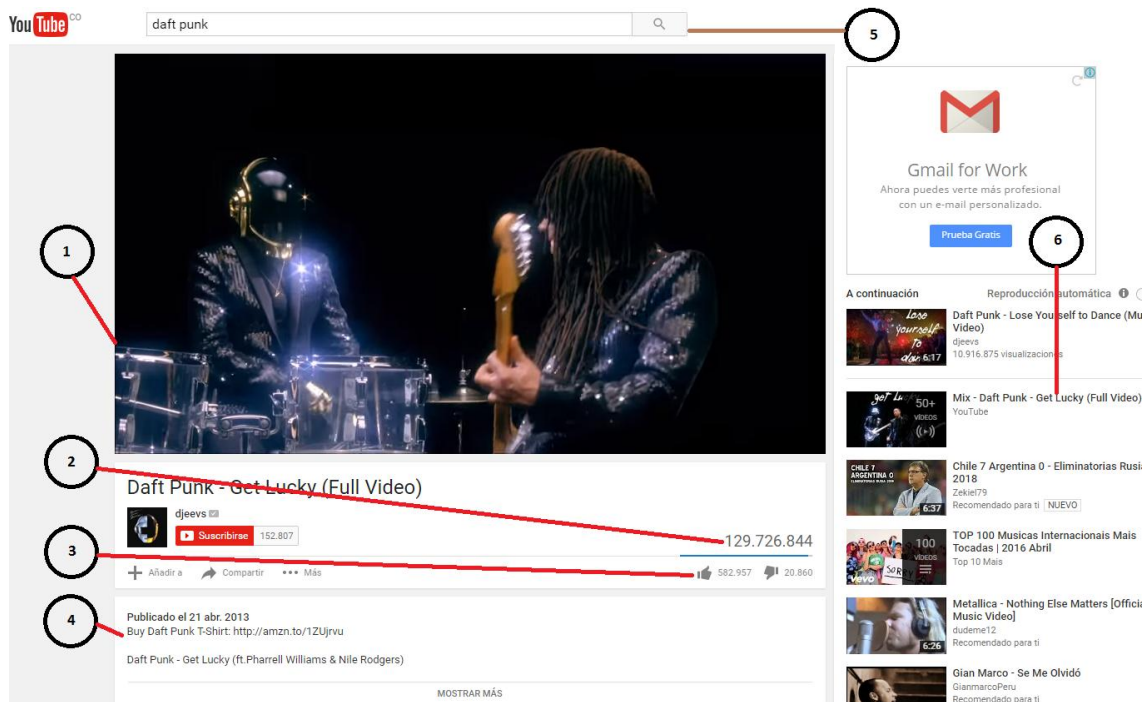


Figura 1. 2. Servicio de video bajo demanda. Fuente: propia.

1.3 Planteamiento del problema

En los últimos años, los proveedores de servicios de Internet han incrementado continuamente las velocidades de acceso a sus suscriptores. Estas mejoras en el ancho de banda, han propiciado el aprovechamiento de estas redes para el consumo de servicios basados en transmisión de audio y video, como es el caso del *streaming* o flujo multimedia [1]. A través del uso del *streaming* no es necesario

que el cliente descargue completamente toda la información para consumirla, ya que esta se va almacenando en un buffer y se va ejecutando al mismo tiempo que se trasmite por la red [11].

Tradicionalmente el *streaming* en internet se ha llevado a cabo por medio de protocolos como Real-time Transport Protocol (RTP) y Real Time Streaming Protocol (RTSP). El protocolo RTP trabaja sobre UDP, por lo que no garantiza que todos los paquetes lleguen a su destino, por lo que el servidor debe gestionar una sesión diferente para cada cliente, además de coordinar la entrega de paquetes por medio de otros protocolos como son: RTSP y RTCP [12], incrementando la información que se transmite por la red. Una solución a lo anterior es la técnica de descarga progresiva usando el protocolo HTTP, lo que permite que el cliente extraiga los datos desde el servidor, ya que la sesión no tiene estado. Lo anterior tiene como ventaja la explotación plena de la infraestructura existente de Internet, aunque HTTP añade una sobrecarga en la transmisión, la cual representa aproximadamente el doble de la tasa de bits de los medios de comunicación [13].

Uno de los servicios cuya transmisión está soportada en *streaming* es la televisión IP o IPTV, la cual es definida por la ITU-T como el envío de servicios multimedia (televisión, video, audio, texto, gráficos y datos) de alta calidad a través de redes IP de banda ancha usando protocolos de *streaming*. Los servicios de IPTV se clasifican desde el punto de vista del usuario final en: servicios de distribución de contenido, servicios interactivos, servicios de comunicación y otros servicios [14]. Dentro de los servicios de distribución de contenidos multimedia, se destaca por los beneficios económicos brindados y por la amplia difusión en portales como Youtube, Vimeo y Netflix, el servicio por suscripción de VoD [15], el cual consiste básicamente en una aplicación que espera, procesa y sirve peticiones de uno, o varios clientes. La petición, contiene un comando mediante el cual el cliente solicita el vídeo que desea recibir. Una vez el servidor ha recibido el comando de reproducción, empieza a transmitir el vídeo. Para facilitar el acceso a los contenidos multimedia, este servicio provee un catálogo de contenidos, a través del cual el televidente navega, escoge, reproduce y en ocasiones valorar cada uno de los contenidos multimedia [1].

A pesar de las ventajas del servicio de VoD, existen un conjunto de problemas que dificultan la experiencia e interacción del usuario en el entorno de televisión, dentro de los cuales se encuentran: **el crecimiento de los catálogos de contenidos multimedia, el tiempo que puede emplear un usuario en la navegación por los catálogos de contenidos multimedia, los limitados métodos de entrada (control remoto) para navegar a través de los catálogos de contenidos, la fluctuación del ancho de banda en el momento de reproducir el contenido multimedia y las diferentes características de los dispositivos que acceden al servicio de IPTV (colores, codecs, resolución, entre otros) [4] [16] [17].** Las anteriores problemáticas pueden ser clasificadas dentro de dos retos principales

para el servicio de VoD: **facilitar el acceso ágil y el permitir consumo adecuado** del contenido multimedia en el entorno de televisión (ver figura 1.3).

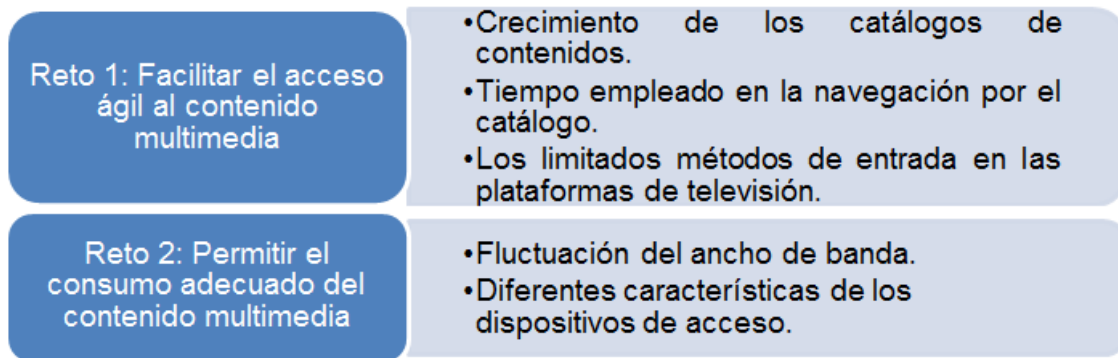


Figura 1. 3. Retos del servicio de VoD. Fuente: propia.

Con respecto al primer reto del servicio de VoD, correspondiente a **facilitar el acceso ágil** al contenido multimedia, una de las posibles opciones para abordarlo son los sistemas de recomendaciones [4] [16], los cuales son entendidos como agentes encargados de identificar los gustos y preferencias de un usuario, para guiarlo de forma personalizada en el proceso de escogencia de un ítem a partir de muchas opciones [18] [19]. De acuerdo a la literatura, estos se dividen en 2 enfoques principales [20] [21] [22] [18] [19] [23] [24]: los basados en filtros de contenidos y los basados en filtros colaborativos.

En el primer enfoque, se tiene en cuenta la información del perfil de usuario y de los contenidos valorados por este sin considerar la información de otros usuarios [20] [21] [22], razón por la cual presentan problemas a la hora de recomendar nuevos contenidos. En el segundo enfoque, las recomendaciones son generadas teniendo en cuenta la similitud del usuario con sus vecinos [20] [21] [22]. Así, los sistemas colaborativos corrigen parcialmente el problema de recomendación de nuevos contenidos, siempre y cuando existan las valoraciones necesarias. Teniendo en cuenta lo anterior, las aproximaciones colaborativas y las basadas en contenido, son con frecuencia integradas para actuar en conjunto.

Uno de los principales problemas de los dos enfoques anteriores, es la dificultad para sugerir contenidos cuando el sistema se encuentra en estados preliminares (no hay suficientes contenidos valorados por los usuarios), problema conocido como **arranque en frío** [18] [17]. Como alternativa a este problema, se suelen utilizar estrategias híbridas que integran sistemas de dominio de conocimiento a los enfoques mencionados [20] [18] [23]. Los sistemas de basados en conocimiento infieren información de los ítems cuando el sistema se encuentra en estados preliminares, para lo cual requieren un conjunto de especialistas que definan de manera detallada las características del modelo de negocio. Dentro de esta categoría de sistemas de recomendaciones, se destacan los basados en contexto o sensibles al contexto, los cuales son foco reciente de investigación como

alternativa para abordar el problema de arranque en frío, haciendo uso para esto de variables del contexto para inferir información de interés en estados preliminares.

De acuerdo a [25] [26], el contexto puede ser entendido como cualquier información que puede usarse para caracterizar el estado de una entidad en un caso específico, siendo una entidad una persona, un lugar, o un objeto relevante para cualquier tipo de interacción entre el usuario y la aplicación. Las actuales aproximaciones, en cuanto al uso de información del contexto con técnicas de recomendación tradicionales, se enmarcan en el ámbito de distribución de contenidos multimedia musicales en escenarios fijos o móviles. En ambos casos, con ayuda de sensores se obtienen variables de contexto tales como: temperatura, ruido ambiente, ubicación, luminosidad, hora, situación geográfica, entre otras. A partir de estas variables, es probable determinar el estado de ánimo o la actividad diaria asociada a una persona en un determinado momento del día, usando para ello métodos de inferencia. Así, de acuerdo a la información inferida, se recomiendan un conjunto de contenidos multimedia que han sido previamente clasificados (según el estado de ánimo o la actividad diaria). Dada la necesidad de variables del contexto desde diferentes fuentes, los entornos móviles se adecuan de mejor forma al despliegue de los sistemas de recomendaciones basados en contexto [27] [28] [29] [30].

En lo que respecta al segundo reto del servicio de VoD, correspondiente a **permitir el consumo adecuado** del contenido multimedia, a partir de los problemas de fluctuación del ancho de banda y acceso al servicio desde diferentes dispositivos, una de las posibles opciones es la técnica conocida como *streaming* adaptativo, especificada en 3GPP como Adaptive HTTP Streaming (AHS) [31], la cual consiste en cortar un archivo multimedia en segmentos de igual duración que pueden ser codificados en diferentes resoluciones y tasas de bits, que son suministrados a un servidor web para ser descargados vía HTTP. Con el fin de establecer la relación entre tasas de bits, segmentos y el orden de los mismos, AHS hace uso del archivo Media Presentation Description (MPD), el cual contiene una descripción formal sobre una colección de datos, que representan las características técnicas del contenido multimedia [32]. Cada cliente primero solicita el MPD y a partir de esa información se indica a los segmentos individuales que se ajusten mejor a sus necesidades. Así, el control está del lado del cliente, de tal manera que cada segmento puede cambiar a una velocidad de bits en función del ancho de banda disponible, o de la resolución del dispositivo de acceso.

La industria ha desplegado varias soluciones propietarias de *streaming* adaptativo, dentro de las que se encuentran: Microsoft Smooth Streaming [33], Apple HTTP Live Streaming [34] y Adobe Dynamic Streaming HTTP [35]. Estas plataformas de *streaming* suelen utilizar como estándar de video digital MPEG-4 y H.264, mientras que como estándar de audio MPEG Advanced Audio Coding (AAC). Dichas plataformas tienen como inconveniente el empleo de técnicas propias de

segmentación, generación de secuencia de tiempo y creación de formatos MPD [36]. Con el propósito de lograr la entrega eficiente de contenidos multimedia utilizando HTTP en sus diferentes formas: adaptativa, progresiva, *streaming*/descarga, así como garantizar la interoperabilidad entre las soluciones propietarias, MPEG (Moving Picture Expert Group) desarrolló Dynamic Adaptive Streaming over HTTP (DASH), el cual se convirtió en un estándar internacional en 2012 (ISO/IEC 23009-1:2012) [32]. De acuerdo con [37], DASH se puede definir como un sistema por el cual se proporcionan formatos, que habilitan la entrega eficiente y de alta calidad de servicios de *streaming*, permitiendo las siguientes ventajas [38]: reutilización de la tecnología existente (contenedores, códecs, DRM⁷), despliegue sobre la infraestructura actual de las redes de distribución HTTP, mejor percepción del servicio por parte del usuario (menor tiempo de arranque, sin buffer), selección de la calidad en función de la red y de las características del dispositivo, cambios entre calidades transparentes al usuario y coexistencia con tecnologías propietarias existentes.

Tal como se mencionó anteriormente, para abordar los dos retos del servicio de VoD, es posible hacer uso de variables de contexto. En el caso del primer reto correspondiente a **facilitar el acceso ágil** al contenido multimedia, variables del contexto de usuario (luminosidad, ubicación, ruido, etc.) pueden ser utilizadas para inferir información (estado de ánimo, actividad diaria) que contribuyan con el problema del arranque en frío de los sistemas de recomendaciones, mejorando así la precisión de los mismos. En lo que respecta al reto correspondiente a **permitir el consumo adecuado** del contenido multimedia, variables como el ancho de banda y la resolución del dispositivo de acceso pueden considerarse a través del *streaming* adaptativo, con el fin de mejorar los problemas de fluctuación de ancho de banda y acceso al servicio desde diferentes dispositivos. Lo anteriormente expuesto está de acuerdo con la definición de contexto para IPTV propuesta en [4], en donde contexto es entendido como cualquier información, que puede usarse para caracterizar la situación de una entidad relacionada con el servicio. Así, de acuerdo a [4], el servicio de IPTV cuenta con 4 entidades, relacionados directamente con los eslabones de la cadena de valor: usuario, dispositivo de acceso, red y servicio.

De acuerdo a lo anterior, los dos retos principales pueden ser abordados desde las dimensiones del contexto de la siguiente forma: a través de los sistemas de recomendaciones basados en contexto se pretende considerar el contexto de usuario, aportando así al reto de facilitar el acceso ágil al contenido multimedia. De igual forma, mediante el *streaming* adaptativo busca abordar el contexto de red y dispositivo, aportando al reto de permitir el consumo adecuado de contenido

⁷ Digital Rights Management (DRM), Gestión digital de derechos, es un término que se refiere a las tecnologías de control de acceso usadas por editoriales y titulares de derechos de autor para limitar el uso de medios digitales.

multimedia (ver figura 1.4). De esta manera, es importante reunir y procesar las variables de contexto más adecuadas para cada dimensión, de tal forma que se pueda aportar a los dos retos principales del servicio de VoD. Dadas las ventajas de los entornos de movilidad para proveer un conjunto de datos a partir de diferentes sensores, este es un escenario propicio para integrar las ventajas del contexto dentro de IPTV. A partir de los anteriores retos y oportunidades, el presente trabajo de doctorado busca responder a la siguiente pregunta de investigación principal: ¿Cómo mejorar el acceso y facilitar el consumo de contenidos multimedia del servicio de video bajo demanda de IPTV móvil, teniendo en cuenta información del contexto?.

IPTV - VoD			
Retos VoD	Acceso ágil al contenido multimedia.	Consumo adecuado del contenido multimedia.	
Alternativas	S.R contexto.	Streaming adaptativo DASH.	
Dimensión Contexto	Contexto de usuario.	Contexto de red.	Contexto de dispositivo.

Figura 1. 4. Retos y alternativas del servicio de VoD. Fuente: propia.

El presente trabajo de doctorado, busca responder a la anterior pregunta de investigación, mediante el planteamiento de una arquitectura de alto nivel basada en contexto para el servicio de VoD, teniendo en cuenta sistemas de recomendaciones y *streaming* adaptativo. A través de esta arquitectura de alto nivel, se pretende integrar las ventajas de los sistemas de recomendaciones y el *streaming* adaptativo, considerando en ambos casos la información del contexto, con el fin de contribuir a las problemáticas de **agilizar el acceso** y **permitir el consumo adecuado** del contenido multimedia. Así, la hipótesis principal del presente trabajo de doctorado es: la arquitectura de alto nivel propuesta, contribuirá a mejorar el acceso y facilitar el consumo de contenidos multimedia del servicio de VoD de IPTV, teniendo en cuenta información del contexto.

1.4 Estado actual del conocimiento

En esta sección se describen los trabajos actuales que han sido tenidos en cuenta para el planteamiento y formulación del presente trabajo de doctorado. Dentro de este trabajo de abarcan 3 temáticas principales: arquitecturas basadas en contexto para servicios de IPTV, sistemas de recomendaciones basados en

contexto para contenidos multimedia, y entornos de *streaming* adaptativo basados en el estándar DASH (ver figura 1.5).

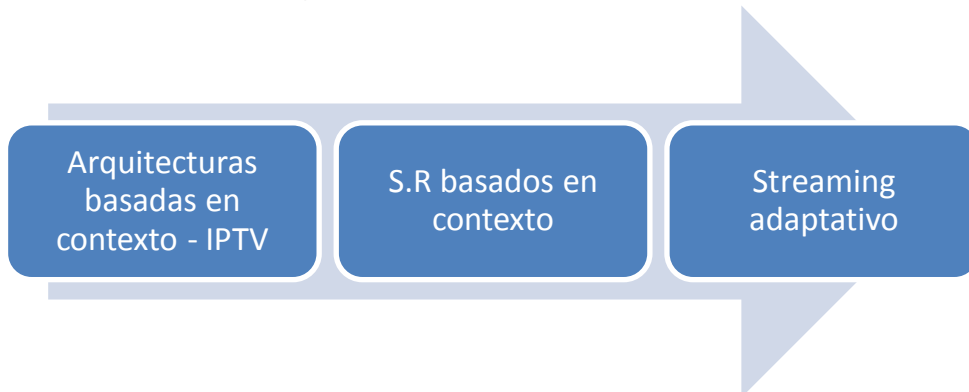


Figura 1. 5. Temáticas de estudio. Fuente: propia.

1.4.1 Arquitecturas basadas en contexto para servicios de IPTV

En esta sección se presentan un conjunto de trabajos que se tuvieron en cuenta con respecto a la temática de Arquitecturas basadas en contexto para servicios de IPTV.

PersonalTVware: A Proposal of Architecture to Support the Context-aware Personalized Recommendation of TV Programs [39] . En este artículo se presenta una arquitectura para el soporte de recomendaciones personalizadas basadas en contexto, sobre entornos de Televisión Digital Interactiva soportados en los mediadores DVB-MHP y GINGA. El sistema de recomendaciones propuesto pretende mejorar las falencias del servicio de guía interactiva EPG, facilitando el acceso a los contenidos adecuados a partir de la gran cantidad de contenidos, ofrecidos por la televisión digital interactiva. La información del contexto es obtenida de manera implícita a través del historial de usuario, y es usada para generar un esquema de metadatos XML, a partir del cual se busca responder de manera constante a las preguntas: ¿Quién?, ¿Cómo?, ¿Cuándo?, ¿Qué? y ¿Dónde?. Así mismo, mediante reglas de inferencia aplicadas a la información del contexto, es posible definir el tipo de contenido a recomendar de acuerdo a la situación particular del usuario en determinado momento. Este artículo no considera entornos de movilidad, ni tampoco escenarios de IPTV, así mismo este trabajo no tiene en cuenta para la definición del contexto, el tipo de dispositivo desde el cual se accede, ni tampoco las características de la red de acceso desde la que se consume el servicio. Este artículo aporta ideas importantes, en cuanto a la descripción de contenidos multimedia usando la especificación TV Anytime y la generación del contexto usando esquemas de metadatos XML.

A context-aware architecture for IPTV services personalization [4]. Este artículo presenta una arquitectura para el soporte de servicios personalizados en el ámbito

de IPTV, considerando características del contexto del usuario, del dispositivo de acceso, de la red y de los servicios interactivos. La arquitectura propuesta busca responder a la necesidad de servicios personalizados, como una alternativa al crecimiento de los catálogos de contenidos multimedia en entornos de IPTV. Dentro de la arquitectura presentada se destaca un sistema de recomendaciones de contenidos multimedia, que tiene en cuenta el perfil del usuario, así como la información del contexto (ubicación, hora, distancia al STB). Para el acceso a los servicios personalizados, en este trabajo se hace uso de servicios web soportados en el estilo arquitectónico REST. Este trabajo no centra su atención en movilidad para IPTV, por lo que solo se tienen en cuenta un conjunto restringido de variables de contexto en el dispositivo de acceso. En cuanto a la obtención de características de la red, si se quisiera migrar esta arquitectura a entornos de movilidad, sería necesario contar con sensores software de medición a lo largo de toda la red. Además de lo anterior, cabe resaltar que el sistema de recomendaciones propuesto, no tiene en cuenta el perfil de usuarios vecinos, por lo que podría mejorarse aún más la precisión de la información recomendada. Este trabajo aporta ideas importantes en cuanto al consumo de servicios interactivos personalizados, haciendo uso del estilo arquitectónico REST.

Context Aware Personalization Ad Insertion in an Interactive TV Environment [25]. Este artículo presenta una arquitectura para la selección e inserción de anuncios publicitarios basados en el contexto, sobre escenarios de televisión digital interactiva. El objetivo de la arquitectura es permitir la personalización de los anuncios, de acuerdo a los gustos y preferencias de un usuario, estos últimos obtenidos mediante reglas de inferencia aplicadas a las variables del contexto. En el presente trabajo, el contexto de usuario es dividido en 2 partes: el contexto pasado y el contexto actual, el contexto pasado se obtiene mediante la extracción de etiquetas o palabras representativas a partir del historial de usuario, por su parte el contexto actual se obtiene mediante la adquisición de las mismas etiquetas, a partir de interacción en tiempo presente con el entorno de televisión (páginas visitadas, contenidos visualizados, compras hechas, uso de la guía interactiva). El servicio de inserción de anuncios personalizados es consumido desde el STB a través del canal de retorno, siendo embebido en el flujo Broadcast, para que el usuario lo acceda mientras interactúa con el sistema de televisión. Este artículo no considera dentro del contexto variables referentes a la red o a las características del dispositivo, así mismo no considera entornos de movilidad. Si bien este trabajo no aborda como tal un sistema de recomendaciones, aporta importantes conceptos en cuanto al manejo de reglas de inferencia para la obtención del contexto.

IPTV Services Personalization Using Context-Awareness [16]. Este artículo presenta una arquitectura para servicios personalizados de IPTV, la cual se encuentra enmarcada dentro del dominio NGN IPTV. La arquitectura planteada busca responder a la necesidad de filtrar el contenido y los servicios adecuados de

acuerdo al perfil del usuario y considerando 4 diferentes tipos de contexto (usuario, terminal, red y servicio). Este trabajo pretende entonces aportar 3 elementos fundamentales: Personalización de la guía interactiva (EPG), adaptación del contenido multimedia de acuerdo a las características del dispositivo STB (Set Top Box) y recomendación de contenidos multimedia de acuerdo a las preferencias del usuario. Para el manejo del contexto de la red, se obtiene la información arrojada (perdida de paquetes, jitter, delay) por el protocolo RTCP (Real Time Control Protocol). En lo que respecta al contexto del usuario, se obtiene información básica a través del STB (hora, localización, historial) e información de la proximidad del usuario a través de sensores RFID. En cuanto al contexto del dispositivo de acceso, se obtienen las características del STB (codecs soportados) con el fin de adecuar los contenidos multimedia a esas características. Finalmente el contexto del servicio depende del modelo del negocio y de la funcionalidad del mismo. Este trabajo no tiene en cuenta entornos de movilidad para IPTV, por lo cual solo se escoge un conjunto reducido de características desde el dispositivo de acceso. Así mismo, el recomendador propuesto solo considera la información del perfil del usuario y las características del dispositivo, sin incluir el perfil de usuarios similares. Este trabajo aporta la definición de características importantes del contexto de la red, que pueden ser tenidas en cuenta en la formulación de la presente propuesta de Doctorado.

A partir de las arquitecturas presentadas anteriormente, es importante precisar que en ninguna de ellas se consideran entornos de movilidad para el despliegue el servicio de IPTV. Así mismo, solo en [4] y [16] se define el contexto de IPTV desde 4 dimensiones (usuario, red, dispositivo de acceso, servicio) mientras que en los otros trabajos solo se aborda parcialmente algunas de estas dimensiones. En cuanto al contexto desde la dimensión de usuario, dado que no se tienen en cuenta entornos de movilidad (diferentes sensores incluidos en un dispositivo móvil), estos trabajos consideran como información del contexto: la hora, la distancia al STB (Set Top Box) y el historial del usuario [4] [16] [25] [39], lo cual limita la precisión de los contenidos presentados por los sistemas de recomendaciones vinculados a la arquitectura. Así, éstas arquitecturas, no consideran como variables de contexto, variables de tipo biométrico o fisiológico. Con respecto a la dimensión de red, esta es abordada en [4] y [16], a través de un conjunto de sensores software distribuidos sobre la red, de tal forma que en ninguno de los 2 casos se hace uso de la técnica de *streaming* adaptativo DASH. En lo que se refiere al contexto desde la dimensión del dispositivo de acceso, tanto en [4] como en [16] se tienen en cuenta características básicas del STB, con el fin de determinar los codecs adecuados para la reproducción del contenido multimedia. Finalmente en cuanto al contexto del servicio, cada una de los trabajos revisados presenta modelos de negocio semejantes con respecto al modo de distribuir el contenido multimedia.

1.4.2 Sistemas de recomendaciones basados en contexto

En esta sección se presentan un conjunto de trabajos que se tuvieron en cuenta con respecto a la temática de sistemas de recomendaciones basados en contexto.

Context-Aware Mobile Music Recommendation for Daily Activities (2012) – CAMRS [28]. Este artículo presenta un sistema de recomendaciones de contenidos musicales para entornos móviles basado en filtros colaborativos, el cual hace uso de información obtenida a partir del contexto (clima, posición, hora), que es usada para filtrar el contenido multimedia a recomendar. Para acceder a la información del contexto, se hace uso de los diferentes sensores con los que cuenta un dispositivo móvil (posicionamiento, acelerómetro, ruido ambiente, clima, etc.), y a partir de esta información en conjunto con reglas de inferencia se logra deducir la probable acción que está desarrollando el usuario en un momento determinado (trabajando, estudiando, caminando, durmiendo). Esta información es cotejada con una base de datos de canciones que han sido previamente etiquetadas de acuerdo a las diferentes acciones de una persona durante el día. Este trabajo no tiene en cuenta las características hardware del dispositivo de acceso (contexto hardware), ni las características de la red desde la que se consume el servicio (contexto de la red). Este trabajo aporta como ideas importantes, el manejo de sensores en el dispositivo móvil para la obtención de la información del contexto y la definición de reglas de inferencia a partir de la información del contexto.

A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory [27]. Este artículo presenta un sistema de recomendaciones híbrido para contenidos multimedia musicales, el cual tiene en cuenta información del contexto de usuario, obtenida a partir de sensores y diversas fuentes de internet. Dentro de las variables obtenidas a partir de estos sensores se encuentran: Temperatura, Humedad, Ruido, Clima, Hora, Estación, entre otras. Esta información en conjunto con los datos del perfil del usuario (al ser tanto discreta como continua), es procesada por un sistema de lógica difusa y por una red bayesiana, generando mediante inferencia un contexto de usuario, a partir del cual se obtienen ponderaciones para los contenidos multimedia musicales. Este trabajo no considera dentro del contexto, la información asociada al tipo de red o al dispositivo de acceso al contenido multimedia. Así mismo, el sistema de recomendaciones planteado, no ha sido diseñado ni evaluado en entornos de movilidad, sin embargo el algoritmo usado para inferir el contexto a partir del perfil del usuario y de la información obtenida de los sensores, es un aporte que puede ser tenido en cuenta para el desarrollo del presente trabajo de Doctorado.

SVR-based Music Mood Classification and Context-based Music Recommendation [29]. Este artículo presenta un sistema de recomendaciones basado en contexto para contenidos multimedia musicales, teniendo en cuenta el

estado de ánimo del usuario. El sistema de recomendaciones provee dos modos diferentes de respuesta: por un lado entrega una recomendación basada en filtros colaborativos centrada en los ítems valorados, y por otro lado una recomendación basada en inferencia (a partir de una ontología), la cual tiene relación el estado de ánimo del usuario con contenidos multimedia musicales previamente clasificados. El estado de ánimo del usuario es determinado a partir de variables del contexto, mientras que la clasificación del contenido multimedia musical, es obtenida a partir del cálculo del factor musical de Arousal/Valence (a partir de características como tonalidad, ritmo, intensidad, escala, etc.) y del uso del método de regresión de vector de soporte (SVR). A partir de lo anterior cada canción es clasificada dentro de 11 categorías posibles de estado de ánimo (enojado, aburrido, calmado, emocionado, feliz, nervioso, tranquilo, contento, relajado, triste y soñoliento). Este artículo no considera dentro de las variables del contexto, las características de la red de acceso, ni las características del dispositivo de consumo de los servicios multimedia, puesto que no se tienen en cuenta entornos de movilidad. Así mismo este trabajo no incluye dentro del sistema de recomendaciones los filtros basados en contenido, lo cual puede mejorar aún más la precisión del recomendador. Este trabajo aporta importantes ideas en cuanto a la clasificación del contenido musical en estados de ánimo que pueden ser tenidas en cuenta para el diseño e implementación del recomendador propuesta en el trabajo de Doctorado.

Music Recommendation Using Context and Context Information Mining [30].

Este artículo propone un Sistema de recomendaciones ubicuo, para contenidos multimedia musicales, el cual tiene en cuenta información del contexto de usuario y de los demás usuarios del servicio de música bajo demanda. El recomendador propuesto parte de la idea de los filtros colaborativos, la cual consiste en seleccionar usuarios con perfiles semejantes para recomendar nuevos contenidos, con la diferencia de que en el presente trabajo, se agrupan usuarios con un contexto similar al del usuario en cuestión, para después buscar similitudes entre los contenidos multimedia del contexto y los del usuario. Las similitudes mencionadas, son obtenidas mediante comparación a bajo nivel de ciertas características específicas de los contenidos multimedia musicales, tales como: ritmo, tono, tempo, timbre. A partir de las anteriores características se obtienen los denominados patrones perceptivos, los cuales son usados en el cálculo de la similitud y generación del listado de recomendaciones. El recomendador propuesto por los autores usa procesamiento fuera de línea y en línea para la generación del listado de recomendaciones. En el procesamiento fuera de línea se obtienen los patrones perceptivos de todos los ítems o contenidos de la base de datos, mientras que en el procesamiento en línea se realizan los cálculos de similitud a partir de las variables del contexto (ubicación, hora, temperatura). Este trabajo no considera dentro de las variables del contexto, las características de la red y de los dispositivos de acceso. Este artículo aporta ideas importantes en cuanto al manejo de patrones perceptivos, para el cálculo de similitud de contenidos multimedia y la

posterior obtención de recomendaciones colaborativas.

Saturday Night or Fever? Context Aware Music Playlists [40] . Este artículo propone un sistema basado en contexto, para la generación de listas automáticas de reproducción de contenidos multimedia musicales. La información del contexto es obtenida a partir de dos fuentes principales: el usuario y el entorno donde el usuario escucha la música. Las variables de contexto de usuario analizadas son: conductividad de la piel, ritmo cardiaco y cantidad de movimiento corporal; por su parte las variables de contexto del entorno del usuario son: temperatura, cantidad de luz y condiciones climáticas. A partir de las anteriores variables y utilizando lógica difusa, los autores definen 9 diferentes tipos de estados emocionales del usuario, según los cuales se genera uno u otro tipo de contenido multimedia musical previamente clasificado. Las variables de contexto de usuario son obtenidas mediante sensores externos al dispositivo móvil, mientras que las variables de contexto del entorno del usuario se obtienen mediante los sensores incluidos en el dispositivo móvil. Este artículo no incluye dentro las variables del contexto, las características de la red o las características del contenido multimedia. Así mismo, este trabajo no considera los filtros de contenidos o filtros colaborativos usados en los sistemas de recomendaciones tradicionales, lo cual podría brindar una mayor precisión en la generación de las listas de reproducción multimedia. La definición de las reglas de inferencia a partir de lógica difusa, son un aporte importante de este artículo, que puede ser tenido en cuenta en el diseño e implementación del sistema de recomendaciones del presente trabajo.

Building an IPTV VoD Recommender System: An Experience Report [15]. Este artículo propone un sistema de recomendaciones para contenidos multimedia, del servicio de video bajo demanda (VoD) de IPTV. Una de las motivaciones para el desarrollo de este recomendador, es el hecho de facilitar el acceso al gran número de contenidos multimedia proporcionados por IPTV, dadas las restricciones para la navegación que posee el control remoto disponible en el sistema de televisión. Así mismo, los autores destacan la importancia de estudiar e investigar el servicio de video bajo demanda, teniendo en cuenta la popularidad del mismo y la rentabilidad que este ofrece, siendo el que más obtiene beneficios económicos después del servicio básico de difusión. Este trabajo presenta 3 aportes importantes: la arquitectura software para el sistema de recomendaciones del servicio de video bajo demanda, un algoritmo híbrido (filtro de contenidos y filtro colaborativo) centrado en los ítems para la recomendación de contenidos multimedia y una evaluación de rendimiento al sistema de recomendaciones propuesto. El sistema de recomendaciones propuesto fue implementado a través de servicios web y usando la tecnología JEEE, en conjunto con el servidor de aplicaciones GlassFish. Este trabajo no tiene en cuenta dentro del sistema de recomendaciones el contexto hardware y de la red, así mismo no considera entornos de movilidad. Finalmente, este artículo aporta importantes ideas en cuanto a la definición de una arquitectura

para el servicio de video bajo demanda, y la presentación de un algoritmo híbrido para la obtención de las recomendaciones.

Personalized TV Program Recommendation based on TV-Anytime Metadata [41]. Este artículo presenta un sistema de recomendaciones, para contenidos multimedia de televisión digital, los cuales son descritos mediante el esquema de metadatos TV-Anytime. El mencionado esquema define un conjunto de características (título, género, sinopsis, actores, directores, etc.) para cada uno de los contenidos multimedia a ser difundidos, así mismo, TV-Anytime permite manejar el historial de contenidos multimedia visualizados, lo cual le permite al recomendador propuesto, obtener por inferencia las preferencias de usuario, usando para ello un sistema de lógica difusa. El sistema de recomendaciones desarrollado en este trabajo, ha sido evaluado en un entorno de televisión digital, soportado en la arquitectura del mediador DVB MHP. Este recomendador parte de la necesidad de mejorar los problemas de navegación y acceso al contenido del servicio de guía interactiva (EPG), los cuales son producto del incremento en el número de contenidos en los catálogos de televisión digital. En este artículo no se abordan escenarios de IPTV para la evaluación del sistema de recomendaciones propuesto, de igual forma, tampoco considera el contexto, ni la vecindad del usuario para generar las recomendaciones. Este trabajo hace un aporte importante en cuanto a la definición de un método de inferencia (basado en lógica difusa) para la obtención de las preferencias de usuario, el cual puede ser considerado en el diseño del sistema de recomendaciones.

Recomender System for Interactive TV [17]. En este trabajo se presentan los principales problemas de la televisión digital interactiva, principalmente los relacionados con el número de canales existentes y el tiempo de salto de un canal a otro. Además de lo anterior, se plantean los sistemas de recomendaciones de contenidos como posible solución a estas falencias, proponiendo una arquitectura base para el diseño estos sistemas en entornos de televisión interactiva. Dentro de los sistemas de recomendaciones, se describen los basados en filtros de contenidos y filtros colaborativos, así como un conjunto de algoritmos para la implementación de estos. Finalmente, se muestra la implementación de un sistema de recomendaciones en el marco de IPTV y se evalúan algunas medidas de rendimiento para los diversos filtros existentes. Este trabajo no considera la información del contexto dentro del sistema de recomendaciones desarrollado, así mismo, tampoco se consideran entornos de movilidad para el despliegue del recomendador. Este artículo brinda un aporte importante en cuanto a la integración de componentes de recomendaciones (filtros y algoritmos) dentro de la arquitectura de IPTV, los cuales pueden ser tenidos en cuenta para el diseño e implementación del sistema de recomendaciones y el servicio de video bajo demanda, propuestos en este trabajo de Doctorado.

Content-Boosted Collaborative Filtering for Improved Recommendations [24].

En este trabajo se presentan las ventajas y desventajas de los sistemas de recomendaciones basados en las dos aproximaciones clásicas: filtros de contenidos y en filtros colaborativos. A partir de las ventajas mencionadas, en este artículo se diseña e implementa un sistema de recomendaciones híbrido, el cual combina la técnica de predicción probabilística del clasificador de Naive Bayes (complementa la función del filtro basado en contenidos), con el algoritmo del vecindario usado para el filtro colaborativo. Este recomendador es usado para mejorar el acceso al contenido multimedia, de un servicio de video bajo demanda propuesto por los autores. Dentro de este artículo no se tienen en cuenta las variables del contexto para enriquecer la precisión de las recomendaciones. Este trabajo tampoco considera entornos de movilidad para el despliegue del sistema de recomendaciones, sin embargo este artículo aporta ideas importantes acerca del diseño e implementación de un recomendador híbrido en conjunto con un servicio de video bajo demanda, lo cual es un aspecto importante a tener en cuenta en el presente trabajo de Doctorado.

Algorithms and Methods in Recommender Systems [42].

Este artículo hace un recuento de las técnicas, métodos, algoritmos y problemáticas, tenidas en cuenta en el diseño e implementación de sistemas de recomendaciones tradicionales y modernos. Dentro de los sistemas de recomendaciones tradicionales se presentan: los basados en filtros de contenido, los basados en filtros colaborativos y los sistemas de recomendaciones híbridos que combinan los dos métodos anteriores. En cuanto a los sistemas de recomendaciones modernos se presentan: los que tienen en cuenta el contexto, los basados en semántica, los que consideran dominios cruzados de información, los que se basan en métodos peer to peer y los que tienen en cuenta recomendaciones multilingües. Así mismo este trabajo presenta los problemas más comunes de los sistemas de recomendaciones, como son: arranque en frío, escalabilidad, escasez de valoraciones, entre otros. Este trabajo aporta conceptos importantes en cuanto a los métodos y algoritmos usados en sistemas de recomendaciones basados en filtros de contenidos, en filtros de colaborativos y en contexto, los cuales se tendrán en cuenta para el diseño e implementación del sistema de recomendaciones propuesto en el presente trabajo.

Esquema de servicios para Televisión Digital Interactiva basados en el protocolo REST-JSON [43].

Este artículo presenta un esquema para el consumo de servicios interactivos de la Web 2.0 (Foros, Micro-Blog, Chat), sobre diversos entornos de TDi (Televisión Digital Terrestre, Televisión Móvil, IPTV), propuestos por el proyecto ST-CAV de la Universidad del Cauca, cuyo objetivo fue el despliegue de comunidades académicas virtuales (CAV) en entornos de televisión. El esquema de consumo de servicios, está basado en el estilo arquitectónico REST-JSON, el cual permite el acceso a un conjunto de recursos independientes, a través de peticiones HTTP simplificadas (GET/PUT), mediante las cuales se intercambian

mensajes livianos en formato JSON. Estos mensajes contienen información de la lógica funcional de cada servicio, de tal forma que al ser recibidos por los diversos clientes de televisión, cada uno implementa la lógica de presentación para cada servicio. Este trabajo aporta un esquema para el consumo de servicios interactivos, que puede ser tenido en cuenta en el presente trabajo para el diseño e implementación del sistema de recomendaciones.

A modo de conclusión, los trabajos anteriores presentan tanto aproximaciones clásicas de sistemas de recomendaciones (filtros de contenidos y filtros colaborativos) de contenidos multimedia en entornos de televisión [15] [24] [17] [41] [42], como recomendadores basados en contexto para contenidos multimedia musicales [28] [27] [29] [30]. Con respecto al primer caso, una de las limitaciones para no vincular el contexto a las recomendaciones es la dificultad para acceder a variables externas, por no considerar entornos de movilidad. Por su parte, en el segundo caso, los recomendadores considerados no están enmarcados en entornos de televisión, ni trabajan con contenidos multimedia de video. Así mismo estos sistemas de recomendaciones no consideran dentro del contexto variables de tipo biométrico o fisiológico. A pesar de esto, los métodos usados en cuanto a clasificación e inferencia de contenidos multimedia a partir de variables locales (ubicación, hora, temperatura, etc.), pueden ser tenidos en cuenta en el manejo de variables del contexto para contenidos multimedia de video sobre escenarios de IPTV. Adicionalmente en [43] se propone un esquema para el consumo de servicios basados en sistemas de recomendaciones para entornos de televisión, el cual puede ser tenido en cuenta para el presente trabajo de doctorado. Finalmente es importante resaltar que en ninguno de estos trabajos, el entorno usado para la difusión de contenidos multimedia, tiene en cuenta las características de la red y las características del dispositivo de acceso.

1.4.3 Entornos de streaming adaptativo basados en el estándar DASH

En esta sección se presentan un conjunto de trabajos que se tuvieron en cuenta con respecto a la temática de entornos de *streaming* adaptativo basado en el estándar DASH.

Light Weight Content Fingerprinting for Video Playback Verification in MPEG-DASH [44]. Este artículo describe el modelo de negocio y distribución de contenidos multimedia, de los proveedores de contenido que hacen uso del estándar DASH. Dentro del modelo descrito, se destaca el servicio de anuncios publicitarios, debido a la importancia de este como principal fuente de ingresos económicos en el contexto multimedia. Los anuncios publicitarios tienen una duración promedio de 15 segundos y suelen aparecer al inicio, al final o en el intermedio de los contenidos multimedia difundidos mediante la tecnología de *streaming*, dichos servicios se

encuentran descritos en el archivo MPD, el cual es un documento en formato XML, que es proporcionado a los clientes DASH al iniciar la transmisión y que normalmente solo describe la información de los contenidos multimedia. Si bien el artículo no aborda la temática de los sistemas de recomendaciones, proporciona bases esenciales para conocer las diversas capacidades y la flexibilidad que ofrece el estándar MPEG-DASH, para vincular servicios adicionales dentro del archivo MPD, lo cual puede ser usado para vincular el recomendador dentro del servicio de video bajo demanda, propuesto en el presente trabajo de Doctorado.

Open Source Column: Dynamic Adaptive Streaming over HTTP Toolset [45].

Este artículo menciona los actuales y principales proveedores de contenidos multimedia adaptativo: Netflix y YouTube, los cuales no despliegan su propia arquitectura de *streaming*, pero ofrecen sus servicios OTT (Over-The-Top), basándose en el uso de la infraestructura de HTTP. Además de lo anterior, este escrito recoge un conjunto de herramientas libres, desarrolladas para propósitos de evaluación, despliegue e implementación del estándar MPEG-DASH. Dentro de estas herramientas se encuentran, en el lado del cliente Libdash, la cual es una biblioteca para el control de *streaming* DASH y el acceso a los segmentos multimedia descargables; otra de las herramientas del lado del cliente es el plugin dash de VLC: “media player”, el cual es necesario para la recepción de contenido multimedia en formato DASH (disponible también en Android); por último se tiene el cliente basado en JavaScript (DASH-JS), que integra perfectamente DASH en la Web, haciendo uso de las extensiones para manejo de medios en HTML5 propias del navegador Google Chrome. Con respecto a las herramientas del lado del servidor, este artículo presenta una base de datos DASH, que contiene diferentes secuencias de películas, disponibles en diferentes longitudes de segmentos, diferentes tasas de bit y resoluciones, además de un conjunto de contenidos codificados usando DASHEncoder, la cual es una herramienta de generación de contenido DASH, recomendada especialmente para el servicio de video bajo demanda (el usuario no necesita codificar y multiplexar por separado cada nivel de calidad del contenido DASH final) y usada para codificación por x264 y Mp4box de GPAC. Si bien este artículo no aborda la temática de sistema de recomendaciones, describe un conjunto de herramientas tanto del lado del servidor como del lado del cliente, para la codificación y recepción de contenidos multimedia adaptativos, soportados en el estándar MPEG-DASH, lo cual es un aporte importante para el diseño e implementación del servicio de video bajo demanda de contenidos multimedia adaptativos, requerido en el presente trabajo de Doctorado.

Dynamic Adaptive Streaming over HTTP: From Content Creation to Consumption [46].

En este artículo se presenta una guía sobre el despliegue de la tecnología de *streaming* dinámico adaptativo a través de HTTP, teniendo en cuenta la creación de contenidos, la entrega, el consumo y la evaluación de los mismos, con respecto a las soluciones de la industria de la competencia. En cuanto a la

creación de contenidos se incluyen temas relacionados con el formato adecuado de codificación, tasas de bits, resolución, velocidad de fotogramas, la longitud del segmento y la calidad actual del contenido multimedia. Además, se tiene en cuenta si las diferentes modalidades de contenido como audio, vídeo, texto son multiplexados o proveídos como tramas individuales. Respecto a la entrega de contenido usando DASH, la guía expone una de las ventajas más importantes del estándar; la utilización de la infraestructura actual de distribución de contenidos (proxys, caches, etc.). Así mismo, se menciona que los patrones de tráfico estudiados en *streaming* DASH difieren de los tráficos conocidos, de tal manera que los efectos positivos y negativos aún no se han evaluado. En cuanto al consumo de contenidos usando DASH, la guía expone que a diferencia de los métodos antiguos de *streaming*, los clientes en lugar de los servidores toman las decisiones oportunas y se encargan de la adaptación. En éste punto se involucran las aplicaciones de cliente, la lógica de adaptación, la estimación del ancho de banda, y la programación de solicitud de segmentos; en consecuencia, las decisiones están influenciadas por las capacidades del dispositivo y tienen en cuenta la satisfacción del usuario. Este trabajo menciona la necesidad de vincular el consumo de servicios interactivos dentro del estándar DASH, así mismo indica las falencias en cuanto a la existencia de herramientas automáticas para la creación de contenidos y su consumo de una manera eficiente. Si bien este artículo no aborda la temática de sistemas de recomendaciones, proporciona una base teórica importante para el diseño e implementación del servicio de video bajo demanda, requerido en el presente trabajo de doctorado, aportando así un conjunto de herramientas y métodos para el manejo de *streaming* adaptativo DASH.

A modo de conclusión, los anteriores trabajos presentan en cada caso un conjunto de herramientas para la codificación, difusión y consumo de *streaming* adaptativo, basado en el estándar DASH. A pesar de lo anterior, en ninguno de estos trabajos se plantea un servicio de video bajo demanda soportado en la tecnología de *streaming* adaptativo DASH. De otra parte, aunque el estándar de *streaming* adaptativo solo define las características de los contenidos multimedia en su archivo de configuración MPD, en [44] se propone la vinculación de aplicaciones sobre el descriptor DASH, lo cual puede tenerse en cuenta para la implementación sistema de recomendaciones que hace parte de la arquitectura a proponer en el presente trabajo de doctorado.

1.5 Aportes investigativos

1.5.1 Brechas existentes

A continuación se presentan las brechas identificadas a partir de la revisión

literaria:

- De la revisión teórica realizada, se encontró que las arquitecturas basadas en contexto para el soporte servicios de IPTV, no consideran en los contexto de red y de dispositivo, la tecnología de *streaming* adaptativo DASH como medio para adaptar el contenido multimedia a las características de la red y del dispositivo de acceso. Estas arquitecturas hacen uso en los contextos de red y de dispositivo de un conjunto de sensores combinados con métodos no estandarizados.
- Según la revisión teórica efectuada, se encontró que las arquitecturas para el servicio de VoD exploradas, consideran un conjunto restringido de variables del contexto de usuario, las cuales son obtenidas a partir del historial de usuario y de las características del STB. Así, dentro de las variables del contexto, estos servicios no consideran variables de tipo biométrico o fisiológico. Es por lo anterior, que los sistemas de recomendaciones asociados a estas arquitecturas, generan un conjunto limitado de sugerencias. Una posible a lo anterior, es que estas arquitecturas no han sido desplegadas en entornos de movilidad, ni consideran variables del contexto externas.
- A partir de la revisión teórica realizada, se encontró que los sistemas de recomendaciones basados en contexto y desplegados en entornos de movilidad, están asociados a servicios de audio bajo demanda, sin considerar contenidos multimedia de video. Así mismo, estos tipos de sistemas de recomendaciones no consideran como variables de contexto, variables de tipo biométrico o fisiológico.
- Según la revisión teórica realizada, no se evidenció la existencia de un servicio de VoD soportado en *streaming* adaptativo DASH, lo anterior considerando que DASH solo define los procesos de codificación, difusión y despliegue de *streaming* adaptativo, más no define la lógica de implementación del servicio de VoD.
- El estándar de *streaming* adaptativo DASH, no define el consumo de servicios interactivos, es decir el descriptor MPD solo incluye la información de las características de los contenidos multimedia. Lo anterior implica la necesidad de adaptar un esquema de consumo de servicios, que permita vincular características de interactividad al servicio de VoD.

1.5.2 Aportes académicos

- Una arquitectura basada en contexto para el soporte del servicio de video bajo demanda de IPTV, apoyada en sistemas de recomendaciones y *streaming* adaptativo. Esta arquitectura se enmarca en la definición del contexto propuesta en [4], en el sentido que considera las dimensiones de red, dispositivo y usuario. Las dimensiones de red y dispositivo son abordadas mediante la tecnología de

streaming adaptativo DASH, mientras que la dimensión de usuario es abordada mediante un sistema de recomendaciones basado en contexto, que considera variables del contexto de usuario.

- Un entorno para la codificación, difusión y consumo de *streaming* adaptativo, soportado en el estándar DASH. Este entorno constituye la infraestructura base para la construcción de un servicio de VoD basado en *streaming* adaptativo, permitiendo una evolución en la infraestructura de *streaming* tradicional (RTP/RTSP) del servicio de VoD.
- Un servicio de video bajo demanda para contenidos multimedia de IPTV Móvil, teniendo en cuenta el estándar de *streaming* adaptativo DASH. Este servicio implementa la lógica del negocio sobre la infraestructura de un entorno de transmisión de *streaming* adaptativo. Adicionalmente, este servicio permite el consumo de contenidos musicales de video que han sido previamente clasificados según el modelo de emociones de arousal-valence.
- Un método computacional para la inferencia de emociones a partir de variables del contexto de usuario de tipo biométrico o fisiológico. Este método computacional permite obtener una emoción de entrada al sistema de recomendaciones clásico de contenidos multimedia emocionales.
- Un dataset de contenidos musicales de video, clasificados según el modelo de emociones de arousal-valence. Este dataset constituye la base del catálogo de contenidos multimedia emocionales de video, difundidos mediante el servicio de VoD.
- Un sistema de recomendaciones basado en contexto, para contenidos multimedia del servicio de video bajo demanda de IPTV Móvil. Este sistema de recomendaciones toma en cuenta un conjunto de variables del contexto de tipo fisiológico o biométrico, con el fin de mejorar de obtener información más precisa para la recomendación de contenidos multimedia de video.
- Un esquema para el consumo de servicios interactivos, vinculado al servicio de video bajo demanda de contenidos multimedia de IPTV Móvil. Lo anterior considerando que DASH solo define los procesos de codificación, difusión y despliegue de *streaming* adaptativo, más no define pautas para la construcción de servicios interactivos asociados al servicio de VoD.

1.6 Objetivos

1.6.1 Objetivo General

Proporcionar una arquitectura basada en contexto para el soporte del servicio de VoD de IPTV móvil, apoyada en sistemas de recomendaciones y *streaming* adaptativo.

1.6.2 Objetivos Específicos

- Adaptar e implementar un entorno para la codificación, difusión y consumo de *streaming* adaptativo, soportado en el estándar DASH⁸.
- Diseñar un filtro de recomendación para el servicio de VoD de IPTV Móvil, teniendo en cuenta un algoritmo de clasificación en inferencia de contenidos multimedia, de acuerdo a información del contexto de usuario.
- Diseñar e implementar una arquitectura basada en contexto para el soporte del servicio de VoD de IPTV Móvil, teniendo en cuenta sistemas de recomendaciones y *streaming* adaptativo⁹.
- Evaluar el funcionamiento del sistema de recomendaciones, y del entorno de transmisión de *streaming* adaptativo, mediante un caso de estudio del servicio de VoD de contenidos multimedia musicales¹⁰.

1.7 Enfoque del trabajo de doctorado

Actualmente la temática del contexto es una tendencia en el área de las telecomunicaciones, con el propósito de enriquecer las características y las prestaciones de los servicios. En el caso del servicio de VoD de IPTV, sus principales retos resumidos en: facilitar el acceso ágil y permitir el consumo adecuado del contenido multimedia, pueden ser abordados a partir de la definición de contexto para IPTV. De acuerdo a [4], el contexto en IPTV puede ser entendido como cualquier información que puede ser usada para caracterizar el estado de una entidad, siendo una entidad aquella que incide en la interacción entre el usuario y el servicio. En el caso de IPTV, se distinguen cuatro entidades o dimensiones del contexto a saber (asociadas con la cadena de valor del servicio): usuario, dispositivo de acceso, red y servicio.

En el presente trabajo de doctorado se pretende aportar a la solución de los retos del servicio de VoD, considerando las cuatro dimensiones del contexto, mediante las temáticas de *streaming* adaptativo DASH y sistemas de recomendaciones basados en contexto, enriqueciendo de esta forma la arquitectura del servicio de VoD de IPTV. A través de la temática de *streaming* adaptativo DASH, se busca aportar en las dimensiones de red y dispositivo del servicio de VoD, atacando los

⁸ Es de aclarar que la idea del entorno es integrar un conjunto de herramientas necesarias que permitan la transmisión de contenidos multimedia usando *streaming* adaptativo. A partir de este entorno será entonces posible el diseño e implementación de un servicio de VoD. Lo anterior teniendo en cuenta que el *streaming* adaptativo es la tecnología, mientras que VoD es el servicio.

⁹ La arquitectura del servicio de video bajo demanda, incluye como apoyo al servicio de video bajo demanda un sistema de recomendaciones, tal como lo hace Youtube o Netflix.

¹⁰ El caso de estudio de contenidos multimedia de video musicales, permite vincular de mejor manera el algoritmo de clasificación e inferencia, adaptando las ideas de los contenidos multimedia de audio presentadas en [23] y [24].

problemas de fluctuación de ancho de banda y acceso al servicio desde diferentes dispositivos. Así mismo, mediante la temática de sistemas de recomendaciones basados en contexto, se busca aportar en la dimensión de usuario, atacando los problemas relacionados con el tiempo de navegación por los catálogos de contenidos y los limitados métodos de entrada desde el entorno de televisión.

En cuanto al *streaming* adaptativo DASH, el aporte investigativo fundamental de este trabajo de doctorado, reside en el diseño e implementación de un entorno para la codificación, transmisión y despliegue de contenidos multimedia adaptativos. Este entorno de *streaming* adaptativo, constituye la infraestructura software base del servicio de VoD, en cuanto a la adecuación, difusión y despliegue del contenido multimedia. Así, se pretende adecuar la arquitectura tradicional del servicio de VoD, migrando la infraestructura clásica de *streaming* RTSP/RTP a una infraestructura soportada en *streaming* adaptativo DASH.

En cuanto a los sistemas de recomendaciones basados en contexto, el aporte de investigación principal de este trabajo de doctorado, está enfocado en la propuesta de un sistema de recomendaciones basado en contexto, que considere variables de contexto diferentes a las tradicionales, como es el caso de las variables de tipo fisiológico o biométrico. Lo anterior con el fin de obtener información más precisa para la recomendación de contenidos multimedia del servicio de VoD de IPTV. A partir de la información del contexto capturada, el sistema de recomendaciones infiere una posible emoción de usuario y le recomienda un conjunto de contenidos multimedia asociados a dicha emoción, que han sido previamente clasificados y consolidados en un dataset de contenidos multimedia musicales. Tradicionalmente cuando se habla del servicio de VoD, se hace referencia también a sus aplicaciones asociadas, como es el caso de los sistemas de recomendaciones. Así la definición de un sistema de recomendaciones, influye en la arquitectura base del servicio de video bajo demanda.

De acuerdo a lo anterior, la arquitectura de alto nivel propuesta en este trabajo de doctorado, tiene como principales aportes y/o componentes: un entorno extremo a extremo para la codificación, difusión y consumo de *streaming* adaptativo, y un sistema de recomendaciones basado en contexto. Mediante el primer componente se busca contribuir a las dimensiones de contexto de red y dispositivo, mientras que con el segundo componente se pretende aportar a la dimensión de usuario. Así, la arquitectura de alto nivel propuesta en este trabajo, aborda las diferentes dimensiones del contexto con el fin de facilitar el acceso ágil y permitir el consumo adecuado del contenido multimedia del servicio de VoD de IPTV.

1.8 Metodología del trabajo de doctorado

Para el cumplimiento del objetivo general y los objetivos específicos planteados en este trabajo de doctorado, se ha organizado la metodología en seis paquetes de trabajo (WP), directamente relacionados con los resultados esperados de cada objetivo específico. Así, los paquetes de trabajo definidos son: construcción de un entorno DASH, dataset de contenidos musicales de video, sistema de recomendaciones basado en contexto, arquitectura para el servicio de VoD, evaluación de la arquitectura y generación de documentación (ver figura 1.6). A continuación se describen brevemente cada uno de estos paquetes de trabajo.

WP1. Construcción de un entorno DASH: Este paquete de trabajo tiene por objetivo la construcción de un entorno extremo a extremo para la codificación, difusión y consumo de *streaming* adaptativo DASH. Este entorno de *streaming* adaptativo es la base para el diseño e implementación del servicio de video bajo demanda objeto de estudio del presente trabajo de doctorado.

WP2. Dataset de contenidos musicales de video: Este paquete de trabajo tiene por objetivo la construcción de un dataset de contenidos musicales de video, considerando el modelo de emociones de arousal-valence. Este dataset de contenidos pretende servir de base para la recomendación de contenidos multimedia emocionales, a partir de las emociones inferidas por el método de inferencia propuesto en el paquete de trabajo WP3.

WP3. Sistema de recomendaciones basado en contexto: Este paquete de trabajo se centra en el diseño y construcción de un sistema de recomendaciones basado en contexto para el servicio de VoD de IPTV. Este sistema de recomendaciones parte de un recomendador clásico bajo el enfoque basado en contenido, el cual es enriquecido mediante el uso de variables biométricas del contexto, las cuales una vez capturadas, permiten inferir emociones que son relacionadas directamente con los contenidos clasificados a través del dataset generado en WP2.

WP4. Arquitectura para el servicio de VoD: Este paquete se enfoca en el diseño de alto nivel de una arquitectura basada en contexto para el servicio de VoD. Dentro de la arquitectura de alto nivel se destacan como componentes principales el entorno de *streaming* adaptativo DASH obtenido en WP1 y el sistema de recomendaciones basado en contexto propuesto en WP3.

WP5. Evaluación de la arquitectura: En este paquete de trabajo, se realiza la evaluación de los principales componentes de la arquitectura de alto nivel basada en contexto para el servicio de VoD. Dentro de los componentes a evaluar se

WP6	WP6.1	Generación y envío de por lo menos un artículo de investigación a una revista indexada de Colciencias, categoría B.												
	WP6.2	Elaboración de la Monografía del Trabajo de Doctorado												

1.10 Estructura de la monografía

La monografía del presente trabajo de doctorado, está estructurada en 7 capítulos a saber.

Capítulo 1 – Introducción: En este capítulo se muestra la descripción detallada de la propuesta de trabajo de doctorado. De esta manera en primer lugar se presenta el conjunto de conceptos que se tuvieron en cuenta para el desarrollo del trabajo de doctorado. Luego de lo anterior, se describen el planteamiento del problema, la pregunta de investigación y la hipótesis del presente trabajo. Igualmente, se muestra la revisión de los trabajos relacionados con el presente trabajo, así como las brechas y aportes obtenidos de la revisión literaria. Finalmente se plantean los objetivos del trabajo y las actividades asociadas a cada objetivo específico.

Capítulo 2 – Arquitectura basada en contexto para el servicio de VoD: En este capítulo se presenta la descripción de los diferentes módulos de la arquitectura de alto nivel para el soporte del servicio de VoD de IPTV. Así, primero se presentan los requisitos funcionales, no funcionales y los casos de uso asociados al servicio de VoD de IPTV. Posteriormente el diseño de la arquitectura basado en contexto para el servicio de VoD, es presentado de manera detallada, mediante diferentes vistas. Dentro de las vistas de la arquitectura de alto nivel, se destacan los módulos de sistemas de recomendaciones y de *streaming* adaptativo DASH, los cuales contribuyen a la definición de contexto dentro en las dimensiones de usuario, red y dispositivo.

Capítulo 3 – Entorno de streaming adaptativo DASH: En este capítulo se presentan los diferentes componentes del entorno para la codificación, difusión y consumo de *streaming* adaptativo DASH. Así, en primera instancia se describen las herramientas libres consideradas para las fases de codificación, difusión y consumo de *streaming* adaptativo. Posteriormente se describe la estructura de la herramienta DASH WebM Converter, la cual permite automatizar los procesos de codificación de *streaming* adaptativo. Finalmente se presenta la estructura modular extremo a extremo del entorno de *streaming* adaptativo desarrollado.

Capítulo 4 – Dataset de contenidos emocionales de video: Este capítulo tiene por objetivo presentar los pasos utilizados para la conformación del dataset de

contenidos emocionales de video. Así, en primera instancia se muestran un conjunto de propiedades o características musicales consideradas para la clasificación de contenidos por emociones. Posteriormente, se describen un conjunto de modelos y soluciones comerciales exploradas para la clasificación de contenidos musicales. Finalmente se presentan los diferentes pasos usados para la conformación del dataset de contenidos emocionales de video, teniendo en cuenta la API de youtube y la API de echonest.

Capítulo 5: Sistema de recomendaciones basado en contexto: En este capítulo se describen los diferentes componentes del sistema de recomendaciones basado en contexto. De este modo, en primera instancia se presenta el marco conceptual de las aproximaciones clásicas de los sistemas de recomendaciones. Posteriormente, se describe el diseño de un recomendador clásico basado en contenidos para la recomendación de contenidos multimedia emocionales. Finalmente, partiendo del sistema de recomendaciones clásico diseñado, se presenta el diseño del sistema de recomendaciones basado en contexto, considerando un conjunto de variables fisiológicas.

Capítulo 6: Pruebas y Resultados: En este capítulo se presentan las pruebas y resultados obtenidos tras la evaluación de los diferentes componentes de la arquitectura de alto nivel para el servicio de VoD de IPTV. Dentro de estas evaluaciones se destacan las pruebas de variación de ancho de banda sobre el escenario de *streaming* adaptativo y las pruebas realizadas sobre el sistema de recomendaciones basado en contexto. Así mismo, se incluyen las pruebas sobre los componentes de captura de variables de contexto y sobre los servicios interactivos asociados al servicio de VoD.

Capítulo 7: Conclusiones y Trabajos Futuros: En este capítulo se presentan las conclusiones, los trabajos futuros y las conclusiones obtenidas con el desarrollo del presente trabajo de doctorado.

Capítulo 2

2. Arquitectura de alto nivel servicio de VoD

2.1 Introducción

IPTV es definida por la ITU como el envío de servicios multimedia (televisión, video, audio, texto, gráficos y datos) de alta calidad a través de redes IP de banda

ancha, haciendo uso de protocolos y/o tecnologías de *streaming* [5]. Uno de los servicios más representativos de IPTV, es el servicio de VoD el cual según [1], es definido como una aplicación que espera, procesa y sirve peticiones de uno, o varios clientes. La petición, contiene un comando mediante el cual el cliente solicita el vídeo que desea recibir. Cuando el servidor ha recibido el comando de reproducción, empieza a transmitir el vídeo. De igual manera, de acuerdo a [10] este servicio se puede definir como una aplicación que le permite al usuario obtener un contenido multimedia en el momento exacto que lo desee. Este servicio en la actualidad ha sido ampliamente difundido a través de portales y/o servicios en línea como Youtube, Vimeo y Netflix, los cuales permiten la visualización en tiempo real del contenido escogido por el usuario a partir de un catálogo, en el momento que este lo desee.

Tal como se mencionó en el capítulo uno del presente trabajo de doctorado, a pesar de los beneficios del servicio de VoD evidenciado en el éxito en los portales y/o servicios ya mencionados, aún existen un conjunto de problemáticas que dificultan la experiencia de un usuario en el entorno de televisión, dentro de los cuales se pueden enumerar: el crecimiento de los catálogos de contenidos multimedia, el tiempo que puede emplear un usuario en la navegación por los catálogos de contenidos multimedia, los limitados métodos de entrada (control remoto) para navegar a través de los catálogos de contenidos, la fluctuación del ancho de banda en el momento de reproducir el contenido multimedia y las diferentes características de los dispositivos que acceden al servicio de IPTV (colores, codecs, resolución,

entre otros) [4] [16] [17]. Estas problemáticas pueden ser clasificadas dentro de dos retos principales para el servicio de VoD: facilitar el acceso ágil y el permitir consumo adecuado del contenido multimedia en el entorno de televisión (ver Figura 2.1).

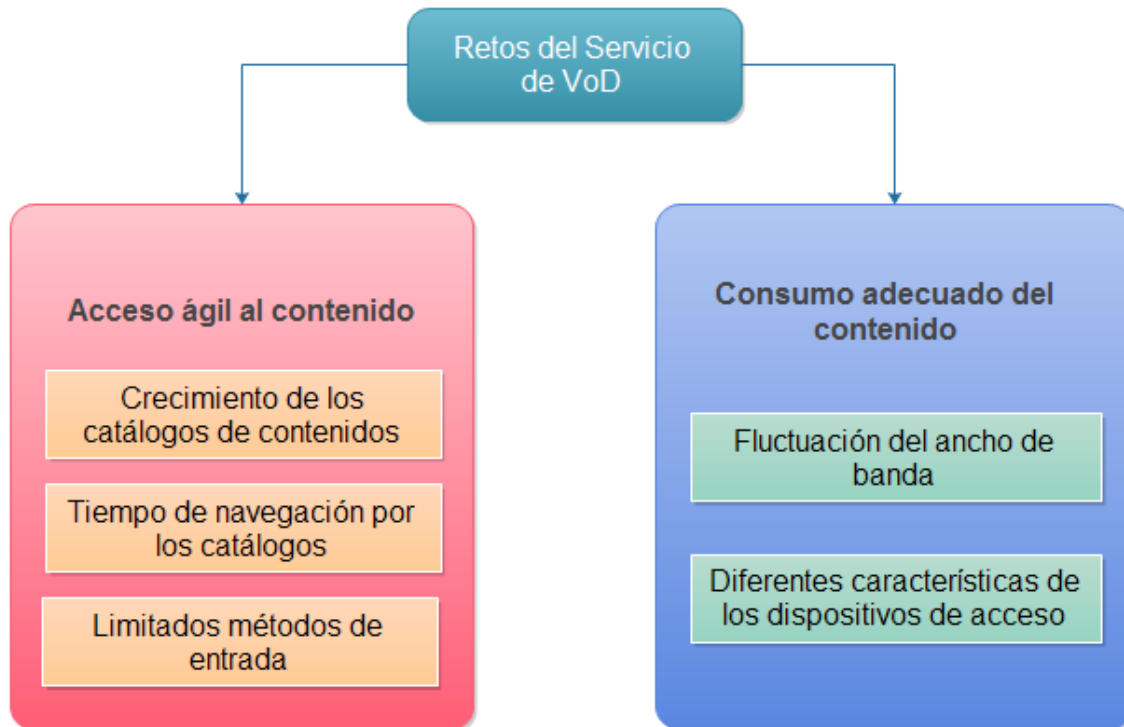


Figura 2. 1. Problemáticas del servicio de VoD. Fuente: propia.

El objetivo del presente trabajo de doctorado es abordar los anteriores retos del servicio de VoD, a través de la propuesta de una arquitectura de alto nivel que permita enriquecer las características del servicio de VoD, teniendo en cuenta la definición del contexto para telecomunicaciones. Como se mencionó en el capítulo uno, el contexto puede ser entendido como cualquier información que puede ser utilizada para caracterizar el estado de una entidad, siendo una entidad una persona, un lugar o un objeto relevante para la interacción entre el usuario y el servicio de VoD [4]. Así, en el caso del servicio de VoD se distinguen cuatro dimensiones del contexto a saber: usuario, dispositivo, red y servicio. Cada dimensión de este contexto tiene a su vez asociadas un conjunto de variables que pueden incidir en la percepción que tienen un usuario del servicio de VoD (ver Figura 2.2).



Figura 2. 2. Problemáticas del servicio de VoD. Fuente: propia.

Según la literatura, en el caso del contexto del usuario algunas de las variables que pueden incidir en la interacción entre el usuario y el dispositivo pueden ser la luz, la humedad, la temperatura, el nivel de ruido, entre otras, las cuales pueden determinar el contenido multimedia que un usuario desea ver en determinado momento. Así mismo para el contexto de dispositivo algunas de las variables que pueden determinar la experiencia de un usuario en cuanto a la visualización de los contenidos son la resolución del dispositivo, el soporte de los códecs de video, entre otras. Para el caso del contexto de red una de las variables que puede incidir en el consumo del contenido multimedia es el ancho de banda. Finalmente en el caso del contexto de usuario, las variables de contexto dependen del modelo de negocio en el cual se enmarque el servicio de VoD.

Así, en el presente trabajo de doctorado, se plantea abordar en específico con 3 dimensiones del contexto: usuario, dispositivo y red, lo anterior con el fin de contribuir con los retos de facilitar el acceso ágil y el permitir consumo adecuado del contenido multimedia. En cuanto al reto de facilitar el acceso ágil al contenido multimedia se pretende aportar desde la dimensión de usuario, mediante el uso de herramientas que permitan acceder de manera precisa y ágil al contenido multimedia a partir de variables de contexto de usuario, como es el caso de los sistemas de recomendaciones basados o sensibles al contexto. En lo que respecta al reto de permitir el consumo adecuado del contenido multimedia, se busca aportar desde la dimensión de red y dispositivo, mediante el uso del estándar de *streaming* adaptativo DASH, el cual permite que el consumo del contenido multimedia tenga en cuenta las características del dispositivo y el ancho de banda disponible del lado del cliente. Así a partir de las dimensiones de usuario, dispositivo y red, la arquitectura de alto nivel para VoD busca enriquecer las características del servicio, aportando a la solución de los dos principales retos del servicio y buscando enriquecer la experiencia de un usuario en un entorno de televisión.

De esta manera, en este capítulo se presentan las diferentes dimensiones de diseño de la arquitectura de alto nivel para el servicio de VoD, la cual busca contribuir a partir de las dimensiones de usuario, dispositivo y red a los dos principales retos del servicio de VoD. El resto del capítulo está organizado de la siguiente manera: en la sección 2.2 se presentan las características del servicio de VoD, los cuales fueron obtenidas a partir de las problemáticas del servicio de VoD evidenciadas en la literatura y teniendo en cuenta pruebas de usabilidad realizadas sobre servicios convencionales de VoD. En la sección 2.3 se presenta la arquitectura de alto nivel para el servicio de VoD desde 4 diferentes vistas a saber: vista del contexto, vista del negocio, vista funcional y vista de implementación. Finalmente en la sección 2.4 se presentan un conjunto de comentarios finales obtenidos a partir del diseño de la arquitectura de alto nivel del servicio de VoD.

2.2 Características funcionales servicio VoD

En esta sección se presentan un conjunto de características funcionales para el diseño del servicio de VoD, las cuales fueron obtenidas a partir del análisis de los principales retos del servicio según la literatura, y a partir de los resultados obtenidos en una prueba de usabilidad realizada sobre un servicio convencional de VoD. Esta última fue realizada en el marco de una estancia de investigación doctoral desarrollada en el grupo UseCV de la Escuela de Ingeniería Informática de la Pontificia Universidad Católica de Valparaíso – Chile.

Para la ejecución de esta prueba se escogió una aplicación en línea con las características básicas del servicio de VoD, en la cual los participantes de la prueba no estuvieran familiarizados. De esta manera, se seleccionó el portal web peliculashoy.com, el cual permite la visualización de películas en línea de manera gratuita que han sido agrupadas en varios catálogos de contenidos por letra inicial, año, género y país. Las películas con las que cuenta este portal, son embebidas a partir de diferentes servidores gratuitos que publican sus contenidos multimedia en línea. El objetivo de la prueba de usabilidad fue evaluar características como: la búsqueda de los contenidos, la navegación por los catálogos de contenidos, la reproducción de los contenidos y la calidad de los contenidos multimedia reproducidos.

La prueba de usabilidad sobre el portal web peliculashoy.com fue desarrollada en el laboratorio de usabilidad de la Pontificia Universidad Católica de Valparaíso con estudiantes del programa de Ingeniería Informática y se diseñó teniendo en cuenta las siguientes fases: acuerdo de confidencialidad, cuestionario pre-test, listado de tareas, cuestionario pos-test y finalmente análisis de resultados (ver Figura 2.3). La estructura en detalle de la prueba es presentada en el Anexo I.

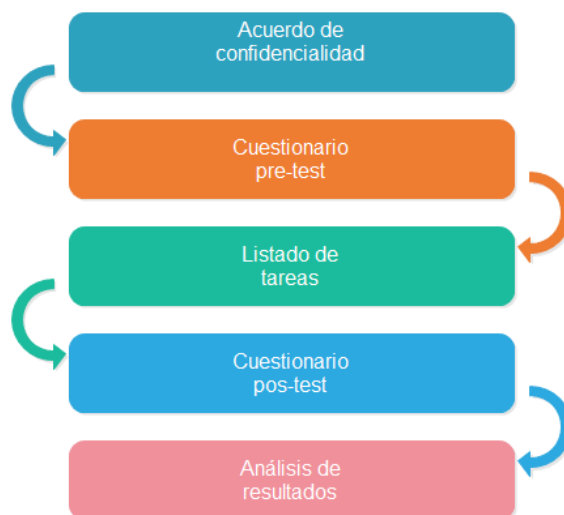


Figura 2. 3. Fases prueba de usabilidad. Fuente: propia.

El acuerdo de confidencialidad se utilizó con el fin de garantizar al usuario el propósito académico de la prueba y la reserva en el uso de los datos obtenidos en la prueba. En el cuestionario pre-test por su parte buscó identificar el perfil y la experiencia previa de los usuarios en portales de contenidos multimedia y/o películas. La fase de listado de tareas incluyó el conjunto de tareas a desarrollar por el usuario dentro del portal *peliculashoy.com*, las cuales se centraban en evaluar la navegación, la búsqueda, la reproducción y la calidad de los contenidos. Finalmente la fase de análisis de resultados abordó el estudio de los datos cualitativos y cuantitativos obtenidos con la prueba.

Como resultados generales de la prueba se pudo concluir que los usuarios indicaron la necesidad de mejorar la navegación por los catálogos de contenidos multimedia, dada la gran cantidad de contenidos existentes y la limitación de las opciones de búsqueda. Así mismo, el experimento arrojó como resultado la importancia de mejorar la calidad de los contenidos multimedia y buscar alternativas para permitir el flujo continuo del contenido multimedia. De igual forma según los usuarios, es necesario mejorar los criterios de búsqueda del portal, dada la limitación de los criterios y resultados de la búsqueda. Finalmente, los usuarios indicaron que el portal no cuenta con un diseño minimalista, que les permita centrarse en la visualización de los contenidos multimedia. Así de acuerdo a lo anterior, la prueba de usabilidad realizada permitió verificar la existencia de los retos de acceso ágil al contenido multimedia y consumo adecuado del contenido multimedia.

De este modo, a partir del análisis de los dos principales retos del servicio de VoD y teniendo en cuenta los resultados de la prueba de usabilidad realizada, se definieron un conjunto de características a tener en cuenta para el diseño del servicio, las cuales se presentan a continuación:

- La arquitectura de alto nivel para el servicio de VoD debe considerar la definición del contexto en sus diferentes dimensiones, con el fin de enriquecer y mejorar la percepción del usuario con respecto al servicio. Así desde la dimensión de red y dispositivo se debe aportar al reto de permitir el consumo adecuado del contenido, teniendo en cuenta el estándar de *streaming* adaptativo DASH. Del mismo modo, desde la dimensión de usuario se debe aportar al reto de garantizar el acceso ágil al contenido multimedia, teniendo en cuenta los sistemas de recomendaciones basados en contexto.
- Dada las mejoras en el ancho de banda de las redes modernas y considerando las tendencias en el uso de HTTP como principal protocolo para el transporte de contenido multimedia, la arquitectura de alto nivel del servicio de VoD en cuanto a la distribución del contenido debe soportarse en el uso de la tecnología de *streaming* sobre el protocolo HTTP.
- En cuanto al consumo de contenido multimedia transportado sobre el protocolo HTTP, la arquitectura basada en contexto desde la dimensión de red y

dispositivo, debe posibilitar la adaptación del contenido multimedia a la fluctuación del ancho de banda, así como a las características del dispositivo.

- Dado que tanto la distribución del contenido multimedia adaptativo DASH, como la lógica del servicio de VoD operan sobre el protocolo HTTP, la arquitectura de alto nivel del servicio debe estar soportada en el modelo cliente servidor y por lo tanto contar con un servidor web que posibilite la implementación de la lógica del negocio y la distribución del contenido multimedia.
- De acuerdo a la literatura explorada en el capítulo uno del presente trabajo, el contexto es utilizado para enriquecer los servicios de distribución de contenidos multimedia, con el fin de identificar preferencias de usuario más precisas que permitan sugerir contenidos más adecuados a las necesidades de este. Así, en estos escenarios, el contexto es relacionado con una actividad o una emoción que realiza o experimenta un usuario en un determinado momento, la cual es usada como variable de entrada al servicio de distribución de contenidos multimedia, para mejorar la personalización del mismo. De este modo la arquitectura de alto nivel para el servicio de VoD, debe emplear el contexto de usuario para la inferencia de emociones, con el fin de enriquecer las recomendaciones de contenidos del servicio y aportar a la recomendación de contenidos cuando el sistema se encuentra en estados preliminares (problema de arranque en frío).
- Los sistemas de recomendaciones clásicos asociados a los servicios convencionales de VoD presentan el problema de arranque en frío, es decir tienen la incapacidad para recomendar contenidos multimedia cuando el sistema se encuentra en un estado preliminar. Así la arquitectura de alto nivel para el servicio de VoD debe considerar el contexto de usuario como medio para enriquecer las sugerencias de contenido multimedia y aportar a la solución del problema de arranque en frío.
- Dado que los sistemas de recomendaciones clásicos enmarcados en entornos de distribución de contenidos multimedia, basan la generación de recomendaciones en el uso de las valoraciones hechas por los usuarios, es necesario enriquecer estas recomendaciones considerando las propiedades musicales de los contenidos multimedia. Así, el sistema de recomendaciones clásico en el que se basa el servicio de VoD, debe soportarse en un clasificador cuyos atributos incluyan las propiedades musicales del contenido multimedia.
- De acuerdo a los trabajos relacionados encontrados en la temática de sistemas de recomendaciones basados en contexto para contenidos multimedia, los contenidos multimedia musicales son los más adecuados para su vinculación con variables de contexto de usuario, dada la relación existente entre de las características musicales y los modelos emocionales. Así, la arquitectura de alto nivel para considerar el contexto de usuario, debe enfocarse en la distribución de contenidos multimedia asociados a videos musicales.
- Los sistemas de recomendaciones basados en contexto deben hacer uso de variables de contexto de usuario, que permitan obtener adecuarse a las

necesidades y preferencias reales de usuario. De este modo, las variables de tipo fisiológico son perfiladas como las más adecuadas para identificar las necesidades más precisas de un usuario, por lo cual la arquitectura de alto nivel para el servicio de VoD, debe vincular un sistema de recomendaciones basado en contexto que haga uso de variables de contexto de tipo fisiológico.

- La arquitectura de alto nivel del servicio de VoD debe permitir la integración del sistema de recomendaciones clásico con las variables del contexto de usuario, de tal modo que se puedan enriquecer la precisión de las recomendaciones y se contribuya a la solución al problema de arranque en frío. Así a partir de la integración del sistema de recomendaciones clásico y las variables del contexto, se conforma el sistema de recomendaciones basado en contexto.
- La arquitectura de alto nivel del servicio de VoD debe posibilitar la respuesta en tiempo real a las variaciones de cualquiera de las variables asociadas a las diferentes dimensiones del contexto. Así, en las dimensiones de red y del dispositivo, el servicio de VoD debe adaptar el contenido multimedia a las variaciones de ancho de banda en el lado del cliente. Del mismo modo en la dimensión de usuario, el servicio de VoD debe actualizar el listado de recomendaciones a los cambios emocionales obtenidos al analizar las variables de contexto de tipo fisiológico.
- Dada la necesidad de la arquitectura de alto nivel del servicio de VoD por vincular variables del contexto de tipo fisiológico, se debe considerar las oportunidades ofrecidas por internet de las cosas a través de los dispositivos *wereables*¹¹, los cuales permiten obtener datos del espacio corporal del usuario, con el fin de obtener información útil de las preferencias de un usuario.
- La arquitectura debe contar con un módulo hardware software que permita la captura de variables fisiológicas de usuario, a partir de las cuales sea posible la inferencia de emociones. En concordancia con lo anterior, el módulo hardware-software debe tener articulados un conjunto de sensores para la adquisición de estas variables, permitiendo la portabilidad y movilidad, tal como lo hacen los dispositivos *wereables* de internet de las cosas.
- El componente de recomendaciones de la arquitectura de alto nivel, debe considerar variables de contexto de tipo fisiológico, dado que de acuerdo a la literatura estas variables pueden ser asociadas directamente con las emociones de usuario. Por lo anterior, conviene hacer uso de las variables de contexto más adecuadas para la inferencia de emociones de usuario.
- La arquitectura de alto nivel para el servicio de VoD debe contar con un método matemático o computacional que permita la inferencia de emociones a partir del análisis de las variables fisiológicas de usuario.
- En el lado del cliente, se debe contar con un componente de estimación de ancho de banda, el cual permita obtener en tiempo real el ancho de banda

¹¹ Dispositivos que se adhieren al espacio personal del usuario y permiten obtener información relevante del mismo.

disponible en el lado del cliente, que permita la conmutación al segmento de contenido multimedia adaptativo asociado al ancho de banda estimado.

- En el lado del cliente, se debe contar con un componente que posibilite la interpretación del archivo descriptor del contenido multimedia adaptativo (MPD), de tal modo que al presentarse una fluctuación en el ancho de banda, se pueda acceder al segmento de contenido multimedia asociado al ancho de banda estimado.
- El servicio de VoD debe vincular un componente de valoración de contenidos multimedia, que permita realimentar las sugerencias generadas por el sistema de recomendaciones clásico asociado al servicio.
- El servicio de VoD debe contar con un componente de seguimiento de ancho de banda, que permita visualizar el comportamiento del ancho de banda de la red en el lado del cliente y la forma como el contenido multimedia se adapta al mismo en tiempo real.
- El servicio de VoD debe estar soportado en un escenario de *streaming* adaptativo que permita realizar los procesos de codificación, segmentación, generación de los descriptores del contenido, distribución y consumo del contenido multimedia.
- El escenario de *streaming* adaptativo en el cual se soporta el servicio de VoD, debe contar con una herramienta de codificación que permita realizar los procesos de codificación, segmentación y generación de los archivos descriptores asociados a los contenidos codificados, de acuerdo al estándar de *streaming* adaptativo DASH.
- Para la recomendación de contenidos multimedia según las emociones, es necesario disponer de un catálogo de contenidos que clasifique los contenidos multimedia musicales según el modelo emocional adaptado en el presente trabajo.
- La arquitectura de alto nivel para el servicio de VoD debe proveer un componente software que permita la extracción de las características musicales a partir del contenido multimedia musical. Dado que de acuerdo a la literatura, las propiedades de arousal y valence permiten la clasificación de un contenido multimedia musical dentro de los modelos emocionales, el componente software debe obtener al menos estas dos características.
- El catálogo de contenidos multimedia debe incluir las propiedades o características musicales de cada contenido musical que permitan clasificar el contenido musical según el modelo emocional adaptado. De acuerdo a la literatura, las características musicales de arousal y valence posibilitan la clasificación de un contenido multimedia musical, por lo que el catálogo debe contener al menos dichas propiedades por cada contenido multimedia.

2.3 Arquitectura de alto nivel

En esta sección se presenta la descripción de la arquitectura de alto nivel para el servicio de VoD. Para lo anterior esta arquitectura se describe desde cuatro vistas a saber: vista de del contexto, vista del negocio, vista funcional y vista de implementación (ver Figura 2.4). En la vista del contexto se muestra como la arquitectura desde las diferentes dimensiones de la definición del contexto aporta a la solución de los dos principales retos del servicio de VoD. En la vista del negocio se presenta como la cadena de valor del servicio de VoD es modificada a partir de la vinculación del contexto a la arquitectura tradicional del servicio. En la vista funcional se describe la estructura modular del servicio de VoD, mediante la cual se ilustran los diferentes componentes funcionales de la arquitectura de alto nivel del servicio. Finalmente en la vista de implementación se presentan los diferentes componentes hardware y software escogidos para la construcción de la arquitectura de alto nivel del servicio de VoD.

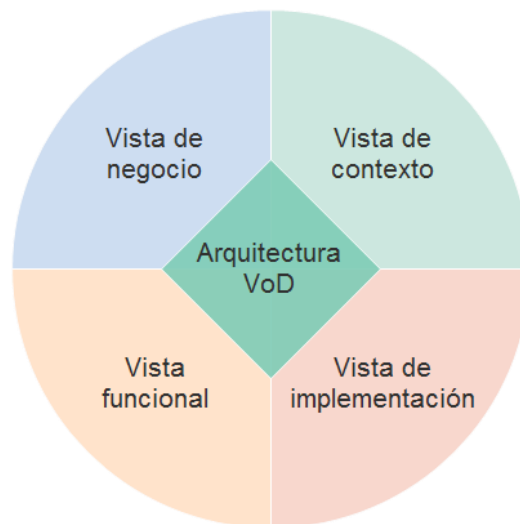


Figura 2. 4. Vistas de la arquitectura. Fuente: propia.

2.3.1 Vista de negocio

Tal como se mostró en el capítulo uno de la presente monografía, la cadena de valor para el servicio de IPTV está formado por cuatro eslabones o dominios a saber: dominio del proveedor de contenidos, dominio del proveedor del servicio, dominio del proveedor de red y dominio del consumidor (ver Figura 1.1). En el dominio del proveedor de contenidos se realiza la producción, edición y metadatos del contenido (películas, series, eventos, documentales, noticias, etc.). En el dominio del proveedor del servicio, se realiza la agregación de contenido, es decir,

se preparan los contenidos que serán enviados al usuario final (codificación y adaptación de los contenidos multimedia), y se proporcionan los servicios de plataforma (por ejemplo, tarificación, autenticación de usuario, etc.). El dominio del proveedor de red por su parte, se encarga de entregar el contenido multimedia al dominio del consumidor y proporcionar la comunicación entre el dominio del proveedor del servicio IPTV y el dominio del consumidor. Finalmente, el dominio del consumidor está compuesto por los terminales necesarios para el consumo del servicio IPTV (televisor, computador, decodificador, STB, dispositivo móvil, etc.), dichos dispositivos permiten consumir el contenido a un formato adecuado que pueda ser desplegado por el terminal o dispositivo de usuario final, así como consumir los diferentes servicios asociados al contenido multimedia [6].

En el caso del presente trabajo de doctorado, la vinculación del contexto al servicio de VoD en sus diferentes dimensiones: usuario, red y dispositivo, implica un cambio en las tareas realizadas en los eslabones de la cadena de valor del servicio, específicamente en los dominios de proveedor de servicio y consumidor. En la figura 2.5 se presentan cada una de las tareas desarrolladas en cada uno de los eslabones de la cadena de valor del servicio de VoD, incluyendo aquellas relacionadas con la vinculación de las diferentes dimensiones del contexto al servicio tradicional de VoD.

De esta manera en el dominio del proveedor de contenido se conservan las tareas de producción y edición del contenido multimedia, así como la generación de los metadatos asociados a cada contenido multimedia. En el dominio del proveedor del servicio por su parte se incluyen las tareas relacionadas con la generación del contenido multimedia adaptativo DASH (codificación, segmentación y generación de descriptores), así como las tareas de análisis e inferencia de emociones a partir de las variables de contexto capturadas en el lado del cliente y la recomendación de contenidos multimedia de acuerdo a la emoción inferida según las variables de contexto procesadas. En lo que respecta al dominio de proveedor de red se conservan las tareas de distribución del contenido multimedia y gestión de la comunicación extremo a extremo. Finalmente en el dominio del consumidor se incluyen las tareas de captura de las variables del contexto de usuario, las cuales permiten el análisis e inferencia de emociones en el dominio del proveedor del servicio. Así mismo en este dominio se realizan las tareas de estimación del ancho de banda, interpretación de los descriptores asociados a cada contenido multimedia musical y reproducción del contenido multimedia. De este modo, en el lado del cliente ante cualquier variación del ancho de banda, se puede acceder a un segmento determinado de contenido multimedia codificado según el estándar DASH en el dominio del proveedor del servicio.



Figura 2. 5. Cadena de valor VoD. Fuente: propia.

A modo de conclusión, aunque los eslabones de la cadena de valor del servicio de VoD se mantienen, la vinculación del contexto a la arquitectura tradicional del servicio implica la captura, análisis y aprovechamiento de las variables de contexto asociadas a las dimensiones de usuario, red y dispositivo.

2.3.2 Vista de contexto

Tal como se ha mencionado previamente, el concepto de contexto ha aparecido en los últimos años ligado a los servicios de telecomunicaciones, con el fin de enriquecer la interacción entre el usuario y el servicio, de tal modo que este último pueda responder a necesidades más precisas del usuario, permitiendo una personalización más adecuada. De este modo, en el ámbito de las telecomunicaciones, el contexto puede definirse como cualquier información que puede usarse para caracterizar el estado de una entidad relacionada con el servicio, siendo una entidad una persona, un lugar o un objeto relevante para la interacción del usuario [4].

En el presente trabajo de doctorado, se ha optado por considerar la definición del contexto desde 3 dimensiones: usuario, red y dispositivo. Lo anterior con el fin de contribuir a la solución de los dos principales retos del servicio de VoD, como son: facilitar el acceso ágil y el permitir consumo adecuado del contenido multimedia en el entorno de televisión. El primer reto del servicio está relacionado con la necesidad de mejorar los tiempos de navegación por el catálogo de contenidos multimedia, dada la gran cantidad de contenidos existentes y los métodos limitados de entrada de los que dispone un usuario en un entorno de televisión. El segundo reto del servicio está relacionado con la necesidad de permitir el consumo adecuado del contenido multimedia, teniendo en cuenta los problemas de fluctuación de ancho de banda y las diferentes características de los dispositivos que acceden al servicio.

Teniendo en cuenta las problemáticas del servicio de VoD mencionadas anteriormente, la arquitectura de alto nivel basada en contexto, pretende abordar los dos principales retos del servicio de VoD, mediante el uso de los componentes arquitectónicos de sistemas de recomendaciones y *streaming* adaptativo DASH, los cuales se enmarcan en las dimensiones del contexto de usuario, red y dispositivo (ver Figura 2.6).

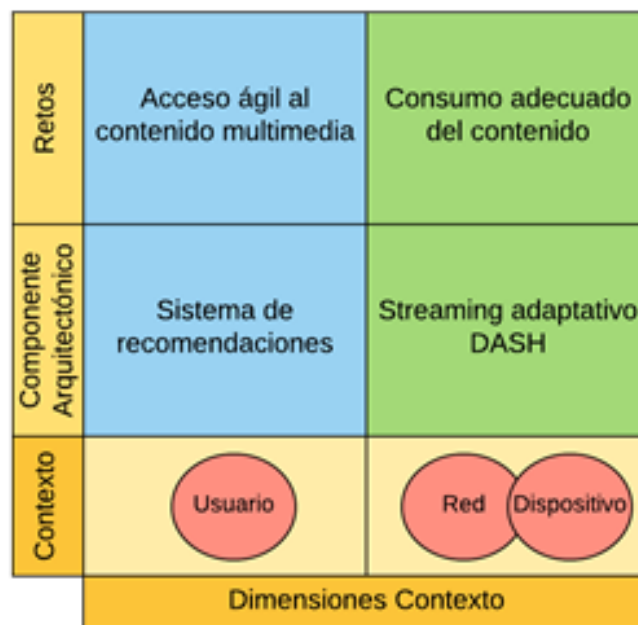


Figura 2. 6. Dimensión del contexto. Fuente: propia.

Mediante el componente de sistema de recomendaciones, el cual se enmarca en la dimensión del contexto de usuario, la arquitectura de alto nivel pretende aprovechar variables de tipo fisiológico del usuario, para a partir de ellas inferir una posible emoción de entrada, la cual pueda ser usada para recomendar un conjunto de contenidos multimedia musicales de video que han sido previamente

clasificados según un modelo emocional. Así, se busca mejorar la precisión de las sugerencias de contenidos aportadas por los sistemas de recomendaciones clásicos, las cuales se centran en las valoraciones de los usuarios, de tal modo que a partir del análisis de variables de tipo fisiológico se puedan generar recomendaciones más adecuadas a las necesidades emocionales de un usuario. Del mismo modo la vinculación de variables del contexto tiene por objetivo aportar a la solución del problema de arranque en frío, el cual se presenta cuando el sistema se encuentra en estados preliminares. Por su parte, los contenidos multimedia musicales de video fueron escogidos, teniendo en cuenta la facilidad que tiene la música a través de sus propiedades musicales (arousal, valence, tempo, etc) para ser asociada con los modelos de emociones.

Mediante el componente de *streaming* adaptativo DASH, el cual se enmarca en las dimensiones del contexto de red y dispositivo, la arquitectura de alto nivel pretende aprovechar las ventajas provistas por el estándar de *streaming* adaptativo DASH, para distribuir contenidos multimedia que se adapten a las características de la red y del dispositivo de acceso. De esta manera, los contenidos multimedia se codifican, segmentan y describen mediante un archivo MPD en el lado del servidor, de modo que ante cualquier variación en el ancho de banda detectado en el lado del cliente, se pueda acceder a un determinado segmento del contenido multimedia asociado al ancho de banda estimado.

Así, desde el contexto de usuario, la arquitectura de alto nivel para el servicio de VoD busca aportar al reto de facilitar el acceso ágil al contenido multimedia, mediante el componente de sistemas de recomendaciones basado en contexto. Del mismo modo, la arquitectura de alto nivel para el servicio de VoD busca contribuir al reto de permitir el consumo adecuado del contenido multimedia, mediante el componente de *streaming* adaptativo DASH (ver Figura 2.6).

Para cada dimensión del contexto se han escogido un conjunto de variables que buscan enriquecer la interacción del usuario con el servicio de VoD. Así, en el contexto de usuario se han escogido en el presente trabajo las variables fisiológicas: voz, variabilidad de la frecuencia cardiaca (VFC) y conductividad de la piel (GSR). Mediante estas variables se infiere una emoción de entrada al sistema, que permite recomendar un conjunto de contenidos multimedia asociados a las necesidades emocionales de un usuario. En el contexto de dispositivo se ha escogido como variable asociada la resolución del dispositivo, con el fin de adaptar el contenido multimedia a las características del dispositivo de acceso. Finalmente en el contexto de red se ha escogido como variable asociada el ancho de banda, con el fin de que el contenido en el lado del cliente responda a los cambios en el ancho de banda (ver Figura 2.7).

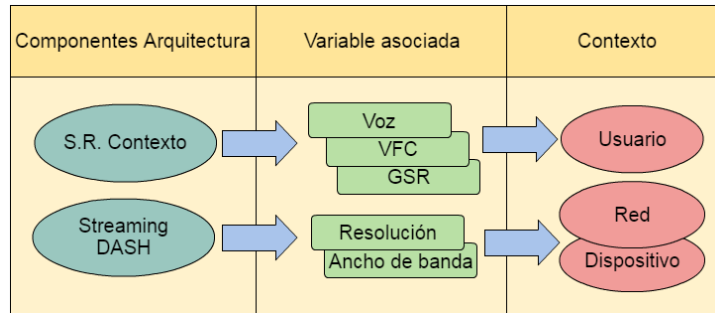


Figura 2. 7. Variables de contexto escogidas. Fuente:propia.

2.3.3 Vista funcional

En esta sección se presentan los diferentes componentes funcionales de la arquitectura de alto nivel de VoD, los cuales fueron obtenidos a partir de la vinculación de las dimensiones del contexto de usuario, dispositivo y red al servicio (ver Figura 2.8). Esta vista funcional de la arquitectura está basada en el modelo cliente-servidor, teniendo en cuenta que la distribución de los contenidos se basa en el *streaming* adaptativo DASH, es decir que tanto la entrega de los contenidos como la lógica del negocio del servicio están soportados en el protocolo HTTP. Así, en la vista funcional de la Figura 2.8, se destacan los módulos de cliente y servidor que a continuación se describen.

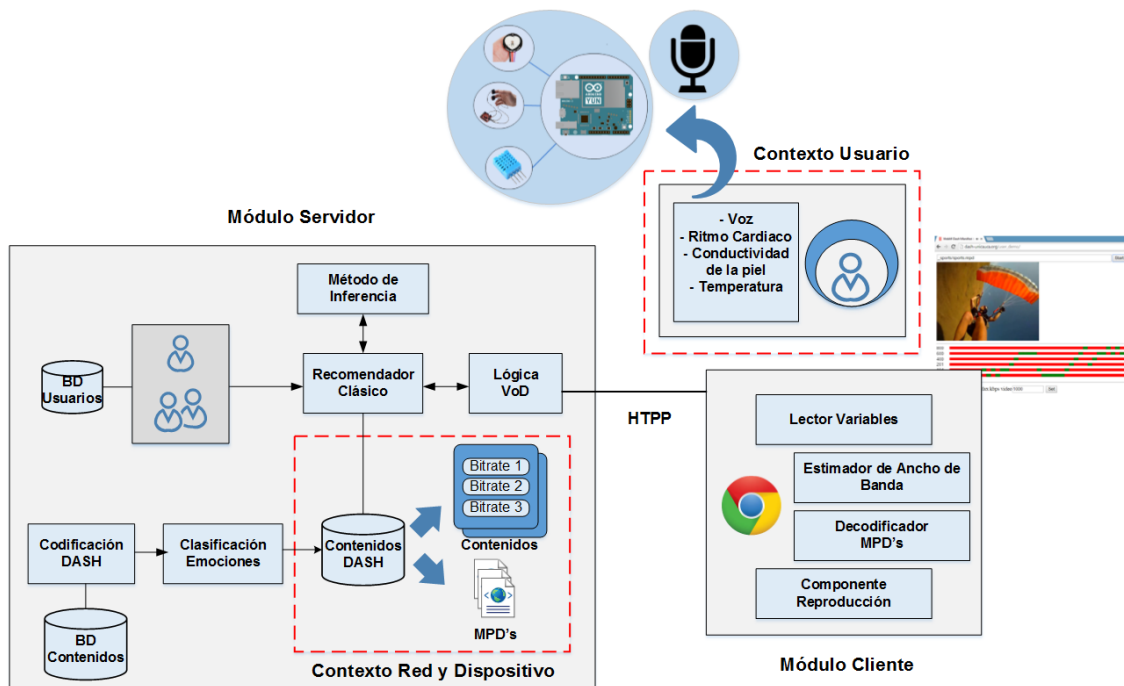


Figura 2. 8. Vista funcional de la arquitectura. Fuente: propia.

2.3.3.1 Módulo del cliente

El módulo del cliente tiene asociado un sub-módulo denominado: “Lector de variables”, el cual tiene asociado a un dispositivo *wearable* abierto de captura. Este sub-módulo se encarga de obtener desde el dispositivo cliente, las variables de tipo fisiológico del usuario, las cuales son usadas en el lado del servidor para inferir una emoción de entrada, que a su vez posibilita la recomendación de contenidos multimedia musicales de video previamente clasificados según un modelo de emociones. Dentro de las variables consideradas para el análisis emocional, en este trabajo de doctorado se consideraron: la voz, la variación de la frecuencia cardiaca y la conductividad de la piel, aunque pudieron haberse considerado también otras variables usadas comúnmente como son la temperatura, la luminosidad, el nivel de ruido ambiente, entre otras. Adicionalmente, en el lado del cliente se cuenta con el sub-módulo “Estimador de ancho de banda”, el cual cumple la función de obtener el ancho de banda disponible en el lado del cliente, para a partir de este acceder al segmento de contenido multimedia codificado que corresponde al ancho de banda estimado. La información de los segmentos y su relación con los diferentes anchos de banda, se encuentra consignada en el archivo descriptor MPD de cada contenido, razón por la cual el sub-módulo “Decodificador de MPDs”, interpreta estos descriptores, para así obtener el segmento de contenido adecuado para el ancho de banda estimado y proceder con su posterior reproducción. Finalmente es importante mencionar que tanto los procesos de captura de variables del contexto y estimación de ancho de banda, se realizan de manera constante con el fin de actualizar el listado de recomendaciones y adaptar el contenido multimedia a los cambios de la red respectivamente.

2.3.3.2 Módulo del servidor

El módulo del servidor tiene asociado un sub-módulo denominado: “Lógica VoD”, el cual se encarga de gestionar la lógica del servicio de VoD. Dentro de las funciones de este sub-módulo, se encuentra la recepción de las variables del contexto a partir del cliente, las cuales son enviadas al sub-módulo “Método de Inferencia” para la obtención de una emoción de entrada. Esta emoción de entrada es usada para realimentar las sugerencias generadas por el sub-módulo “Recomendador Clásico”, que a su vez son obtenidas por un clasificador bayesiano que hace uso del perfil de usuario, así como de las valoraciones y las características musicales de los contenidos multimedia. De este modo, el sub-módulo de “Recomendador Clásico” en conjunto con el sub-módulo “Método de Inferencia” conforman el recomendador basado en contexto.

A partir del listado de sugerencias generadas por el recomendador basado en contexto, el usuario puede escoger un contenido multimedia musical de video del catálogo de contenidos. Cada contenido del catálogo ha sido previamente

adecuado al formato del estándar DASH y clasificado según un modelo de emociones, por sus características musicales. Para realizar la adecuación del contenido multimedia al formato DASH, se hace uso del submódulo “Codificación DASH”, el cual se encarga de realizar las tareas de codificación, segmentación y generación de descriptores para cada contenido multimedia. En lo que respecta a la clasificación del contenido, el sub-módulo “Clasificación Emociones”, cumple con la función de extraer las características musicales de los contenidos multimedia musicales, para a partir de estas clasificar el contenido según un modelo de emociones. Cabe mencionar, que en el presente trabajo se hizo uso de las características musicales de arousal y valence para clasificar el contenido multimedia dentro de una adaptación del modelo emocional de Russell. Finalmente, los contenidos multimedia que pasan el proceso de codificación y clasificación, son dispuestos en un catálogo o dataset de contenidos multimedia emocionales, el cual es descrito en detalle el capítulo cuatro. Una vez descritas las funciones y sub-módulos que conformar el módulo de cliente y el módulo de servidor, en la figura 2.9 se presenta un diagrama de flujo que representa la secuencia funcional de la arquitectura de alto nivel para el servicio de VoD.

En primera instancia el usuario accede al servicio de VoD (1), posteriormente en el lado del cliente se obtienen las variables del contexto de tipo fisiológico (2) a partir de las cuales en el módulo del servidor se analiza e infiere el estado emocional del usuario (3). A partir del estado emocional inferido, el sistema de recomendaciones clásico, genera un conjunto de recomendaciones de contenidos multimedia (4), considerando las valoraciones y los parámetros musicales de cada contenido multimedia (arousal, valence, tempo, etc.). De acuerdo al listado de recomendaciones obtenido, el sistema de recomendaciones basado en contexto, se encarga de filtrar el listado de sugerencias, considerando la emoción de entrada obtenida, para así generar un nuevo listado de recomendaciones basadas en contexto (5). A partir de este listado, el usuario escoge un contenido multimedia (6) para que sea reproducido en la interfaz del servicio, luego de lo cual se procede de manera simultánea y repetitiva a realizar un seguimiento sobre el ancho de banda disponible en el lado del cliente (7) y sobre el cambio en la emoción de entrada a partir del estudio de las variables del contexto de usuario (2). De este modo al hacer el seguimiento sobre las variables del contexto se repiten de manera continua los pasos del 2 al 5. Por su parte, en lo que respecta al seguimiento del ancho de banda, en primer lugar se estima el ancho de banda disponible (7), luego se continúa con la descarga y la consulta del archivo descriptor MPD asociado al contenido multimedia escogido (8), el cual contiene la información del segmento de contenido multimedia DASH codificado correspondiente al ancho de banda estimado. Luego de identificarse el segmento de contenido multimedia DASH, se solicita dicho segmento en el lado del servidor (9), para lo cual se consulta el dataset de contenidos multimedia emocional (10), que almacena los contenidos musicales clasificados por emociones y codificados según el estándar de *streaming* adaptativo

DASH. Después de consultado el segmento de contenido, este se envía al lado del cliente y se inicia la reproducción del mismo (11). Al mismo tiempo que el contenido multimedia es reproducido, se realiza la comprobación en tiempo real del ancho de banda en el lado del cliente (12), para verificar si hubo un cambio con respecto al estimado inicialmente. En caso de que el ancho de banda no haya cambiado, se da continuidad a la reproducción del contenido multimedia, mientras que si se evidencia un cambio en el ancho de banda, se ejecutan nuevamente las tareas descritas en los pasos 9,10 y 11. Finalmente el usuario puede cerrar la sesión del servicio en el momento que lo desee (13).

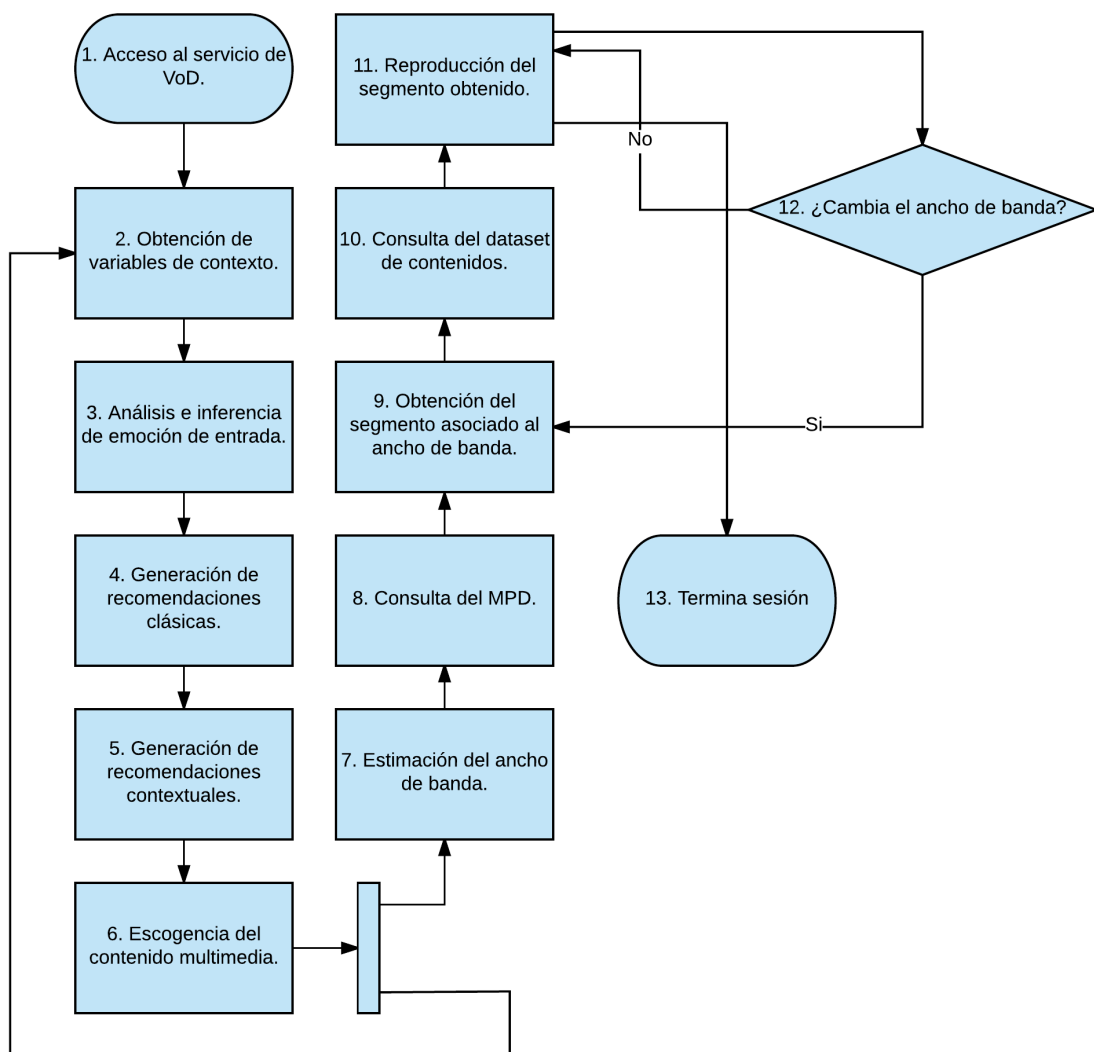


Figura 2. 9. Secuencia funcional de la arquitectura. Fuente: propia.

Finalmente como complemento a la secuencia funcional de la arquitectura de alto nivel del servicio de VoD presentada en la figura 2.9, en la figura 2.10 se presenta la forma en la que estas tareas son distribuidas entre el módulo cliente y el módulo servidor. Así mientras en el lado del cliente se realizan principalmente las tareas de captura de variables del contexto, estimación del ancho de banda y reproducción

del contenido multimedia DASH, en el lado del servidor se ejecutan principalmente las tareas de inferencia de emociones, generación del listado de recomendaciones contextuales y consulta del dataset de contenidos multimedia musicales de video.

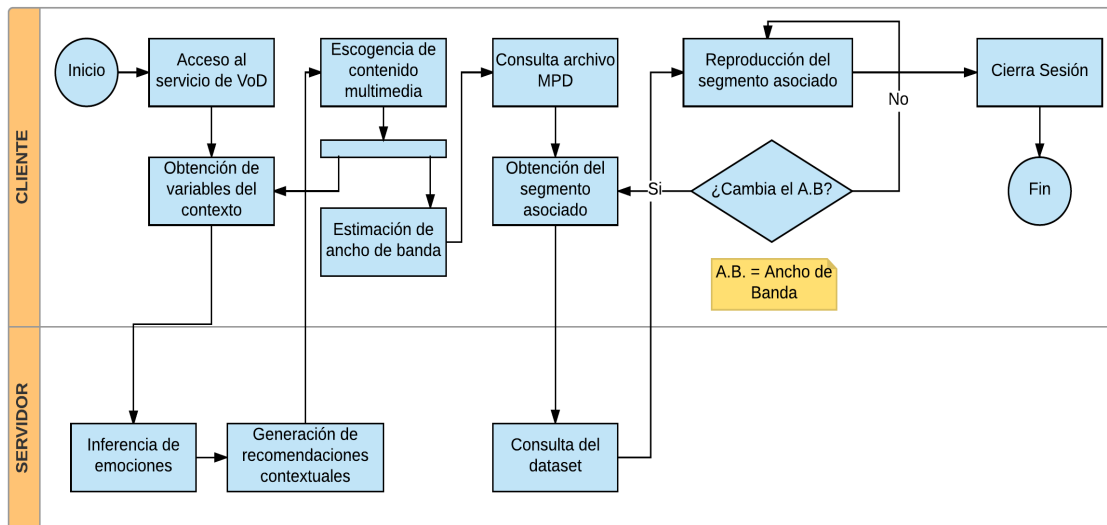


Figura 2. 10. Interacción módulos cliente y servidor. Fuente: propia.

2.3.4 Vista de implementación

A nivel de la vista de implementación, en esta sección se presentan los componentes hardware y software escogidos para la implementación de la arquitectura de alto nivel para el servicio de VoD (ver Figura 2.11), los cuales están enmarcados en una arquitectura cliente-servidor. En el módulo del cliente, el usuario tiene adherido a su espacio corporal un conjunto de dispositivos *wereables*, los cuales permiten la obtención de variables del contexto que son empleadas para inferir una emoción de entrada al sistema, así como realizar un seguimiento sobre los cambios emocionales de un usuario. En el caso de la presente arquitectura, la información del contexto es obtenida haciendo uso de la plataforma de internet de las cosas Arduino Yun, en conjunto con los sensores de obtención de frecuencia cardíaca y conductividad de la piel compatibles con esta plataforma. De igual manera, en el lado del cliente se cuenta con un conjunto de dispositivos de acceso desde los cuales se accede al servicio de VoD. Lo anterior es posible teniendo en cuenta que el servicio ha sido desplegado en un escenario web y cuenta con el soporte del estándar DASH.

En el módulo del servidor por su parte, se cuenta con un servidor web el cual se encarga de recibir y procesar las peticiones por parte del usuario y enviarlas al servidor lógica que cumple la función de gestionar la lógica del servicio de VoD. Tanto el servidor web, como el servidor de lógica son componentes conceptuales

que están implementados a través del servidor web Apache. El servidor de lógica hace uso de un servidor de contexto, el cual se encarga de procesar la información de contexto obtenida en el módulo del cliente, para inferir la emoción percibida por el usuario en un determinado momento. El servidor de contexto fue implementado mediante el framework web Flask, el cual permite el rápido despliegue de servicios en línea mediante el lenguaje Python. De la misma manera, el servidor de lógica hace uso de un servidor de procesamiento, el cual cumple con la función de generar el listado de recomendaciones basadas en contexto a partir de la emoción inferida por el servidor de contexto. Para realizar el proceso anterior, el servidor de procesamiento, hace uso del framework web Spark Java y la API de la herramienta de minería de datos Weka. El framework Spark de Java, el cual brinda la facilidad para la construcción y despliegue rápido de servicios de telecomunicaciones, tal como lo hace el framework Node.js de javascript. De igual modo, para la generación del listado de recomendaciones, el servidor de procesamiento consulta la base de datos de usuario y el dataset de contenidos multimedia, los cuales son gestionados por el servidor de base de datos MySQL. El dataset de contenidos multimedia también fue generado en un formato portable como JSON, mediante el uso de la base de datos no-sql TinyDB. Este dataset contiene el listado de contenidos multimedia disponibles y el conjunto de propiedades musicales asociadas a cada uno de ellos, mediante las cuales es posible relacionar un contenido multimedia con una emoción.

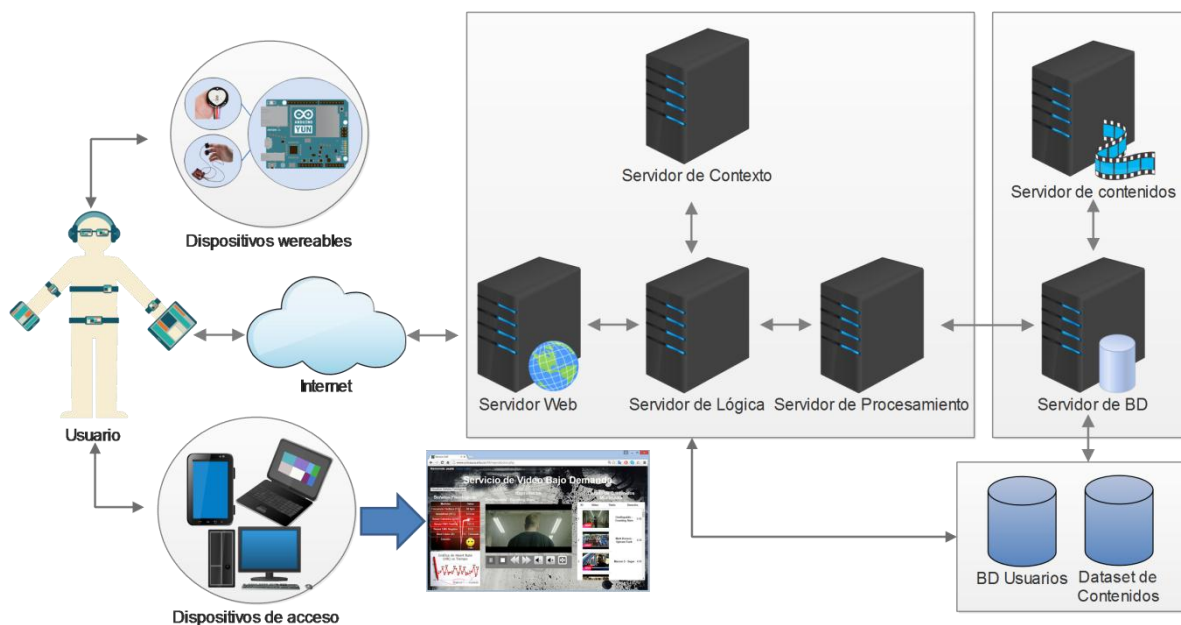


Figura 2. 11. Diagrama de implementación. Fuente: propia.

Finalmente, el servidor de contenidos multimedia es un componente conceptual que está enmarcado dentro del servidor web Apache y cuyo propósito es almacenar los contenidos multimedia que son reproducidos en el lado del cliente. Estos contenidos, son codificados en diferentes calidades de acuerdo al estándar DASH

y tienen asociado un descriptor MPD, con la relación de segmentos de contenido y anchos de banda. Cabe mencionar que este componente también almacena la herramienta de codificación DASH WebM Converter, la cual permite realizar los procesos de codificación, segmentación y generación del archivo descriptor MPD de manera off-line.

A modo de resumen de los componentes presentados en la figura 2.11, en la tabla 2.1 se presenta la información detallada de los componentes software y hardware empleados para la implementación de la arquitectura de alto nivel para el servicio de VoD.

Tabla 2. 1. Componentes hardware-software de la arquitectura. Fuente: propia.

Componentes de la arquitectura	
Hardware	<ul style="list-style-type: none"> • Plataforma de IoT “Arduino Yún”. • Sensor de frecuencia cardiaca “Pulse Sensor”. • Sensor de conductividad de la piel “Grove” GSR.
Software	<ul style="list-style-type: none"> • Servidor web Apache. • Componente servidor implementado mediante el framework Flask de Python. • Componente servidor implementado mediante el framework Spark de Java. • API de la herramienta de minería de datos Weka. • Gestor de base de datos MySQL. • Gestor de base de datos no-sql TinyDB • Herramienta de codificación libre propuesta DASH Web Converter.

2.4 Comentarios finales arquitectura VoD

En este capítulo se presentó el diseño de alto nivel de una arquitectura para el servicio de VoD desde cuatro vistas a saber: vista de negocio, vista de contexto, vista funcional y vista de implementación, las cuales ilustran la manera en la que la definición del contexto para telecomunicaciones, puede contribuir a la solución de los dos principales retos del servicio de VoD.

En concordancia con lo anterior, las vistas de la arquitectura de alto nivel propuesta, consideran el contexto de usuario, dispositivo y red. Desde la dimensiones de dispositivo y red, se busca aportar a la solución del reto de consumo adecuado del contenido multimedia, mediante el uso del estándar de *streaming* adaptativo DASH. En lo que respecta a la dimensión de usuario, se busca aportar a la solución del reto de acceso ágil al contenido multimedia, mediante el uso de un sistema de recomendaciones basado en contexto, estando el contexto asociado a las variables

fisiológicas de usuario. De acuerdo con lo anterior, los principales aportes a nivel funcional de la presente arquitectura se pueden resumir en:

- Un sistema de recomendaciones basado en contexto para contenidos multimedia musicales, el cual posibilita la recomendación de contenidos musicales de video a partir de variables de contexto de tipo fisiológico.
- Un sistema de recomendaciones clásico, soportado en un nuevo clasificador para contenidos multimedia musicales, el cual considera como atributos parámetros como el arousal y la valence de los contenidos multimedia musicales.
- Un dataset de contenidos multimedia musicales, el cual clasifica los contenidos de acuerdo al modelo de 5 emociones propuesto en el presente trabajo y teniendo en cuenta las propiedades musicales de arousal y valence.
- Un método computacional para la inferencia de emociones a partir de variables de contexto de tipo fisiológico, el cual permitió la asociación de emociones de entrada con los contenidos multimedia musicales clasificados y agrupados en el dataset generado.
- Un escenario de *streaming* adaptativo que permite el soporte de las tareas de codificación, segmentación, difusión y consumo de contenidos multimedia adaptativos DASH.

Finalmente, la presente monografía aborda las dimensiones del contexto del servicio en los demás capítulos de la siguiente manera: en el capítulo tres se aborda el contexto de red y de dispositivo, mediante la presentación de la estructura detallada del entorno de *streaming* adaptativo DASH, en el cual se soporta el servicio de VoD; por su parte en los capítulos cuatro y cinco se abordan el contexto de usuario, mediante la presentación respectiva del dataset de contenidos multimedia musicales y el sistema de recomendaciones basado en contexto que hace uso de este último.

Capítulo 3

3. Escenario de streaming adaptativo DASH

3.1 Introducción

El *streaming* es una técnica para la distribución de contenido multimedia en internet, en la cual no es necesario que el cliente descargue completamente ésta información para consumirla, ya que se va almacenando en un buffer y se va reproduciendo al mismo tiempo que se trasmite por la red [11]. Tradicionalmente el *streaming* en internet se llevaba a cabo por medio de protocolos como Real-time Transport Protocol (RTP) y Real Time Streaming Protocol (RTSP). El protocolo RTP trabaja sobre UDP, por lo cual no garantiza que todos los paquetes lleguen a su destino, es por esto que el servidor debe gestionar una sesión diferente para cada cliente, además de coordinar la entrega de paquetes por medio de otros protocolos como son: RTSP y RTCP [12], por lo tanto se incrementa la información que se transmite por la red.

Una solución a los inconvenientes del *streaming* tradicional es la técnica de descarga progresiva. A diferencia de RTP, ésta hace uso del protocolo HTTP, razón por la cual, la sesión no tiene estado y el cliente extrae los datos desde el servidor. Lo anterior tiene como ventaja la explotación plena de la infraestructura existente de Internet, sin embargo este enfoque tiene como desventaja que el protocolo HTTP añade una sobrecarga significativa en la transmisión [13]. Por lo anterior, la descarga básica progresiva sobre HTTP no se adecuaba a las necesidades de los entornos de movilidad, teniendo en cuenta el problema frecuente de las fluctuaciones en el ancho de banda, generando así una nueva necesidad tecnológica, la cual sugiere que el flujo de vídeo se adapte a las capacidades de ancho de banda y a las características del dispositivo de acceso, con el fin de

ofrecer al usuario una secuencia de vídeo continua, con la mejor calidad posible en la recepción.

Una alternativa a los problemas anteriores, es la técnica conocida como *streaming* adaptativo, la cual consiste en cortar y codificar un archivo multimedia en segmentos de igual duración en diferentes resoluciones y tasas de bits, los cuales se suministran a un servidor web para ser descargados a través de peticiones HTTP estándar, de acuerdo a las variaciones de ancho de banda detectadas en el lado del cliente. Así a partir de la definición de contexto presentada en el capítulo 2, con el uso de la tecnología de *streaming* adaptativo se pretende aportar al contexto de red, contribuyendo así a la solución de los problemas de fluctuación de ancho de banda y diferentes características de los dispositivos que acceden al servicio de VoD. En este capítulo se presenta un escenario extremo a extremo de *streaming* adaptativo, el cual fue configurado haciendo uso de herramientas libres. El escenario propuesto tiene como componente principal en el lado del servidor la herramienta Dash WebM Converter, cuyo objetivo es automatizar el proceso de generación de contenido multimedia, según lo establecido por el estándar DASH. La herramienta propuesta fue desarrollada en el lenguaje Python, y permite la integración y ejecución secuencial de las tareas de codificación, segmentación y creación de los archivos descriptores de contenido. Para ejecutar las anteriores tareas, esta herramienta usa en segundo plano las librerías: libwebm, webm-tools, así como la herramienta FFmpeg.

El resto del capítulo está organizado de la siguiente manera: en la sección 3.2 se presentan los conceptos básicos del *streaming* adaptativo DASH. En la sección 3.3 se describen un conjunto de herramientas que fueron exploradas de cara a la conformación del escenario extremo a extremo de *streaming* adaptativo DASH. En la sección 3.4 se muestran las herramientas seleccionadas para la conformación del escenario DASH, las cuales fueron escogidas a partir de las herramientas presentadas en la sección 3.3. En la sección 3.5 se describen los módulos de codificación, difusión y recepción del escenario de *streaming* adaptativo DASH. En la sección 3.6 se presentan las herramientas que fueron consideradas para el diseño e implementación de la herramienta DASH WebM Converter. En la sección 3.7 se detallan las diferentes etapas del proceso de codificación de *streaming* adaptativo, llevadas a cabo por la herramienta Dash WebM Converter. En la sección 3.8 se describen los módulos funcionales a nivel de implementación que componen la herramienta DASH WebM Converter. En la sección 3.9 se describe la estructura del archivo descriptor de *streaming* adaptativo DASH. Finalmente en la sección 3.10 se presenta la estructura funcional del servicio de VoD implementado a partir del escenario de *streaming* adaptativo DASH.

3.2 Streaming adaptativo DASH

El *streaming* es una técnica para la distribución de contenido multimedia en internet, en la cual no se necesita que el cliente descargue completamente la información para consumirla, ya que esta se va almacenando en un buffer y se va reproduciendo al mismo tiempo que se transmite por la red. Tradicionalmente esta técnica de distribución de contenidos, ha sido soportada en los protocolos tradicionales de *streaming* como son: RTSP y RTP [11] [12]. A pesar de que estos protocolos han sido clásicamente usados para la transmisión y recepción de *streaming*, cuentan con algunos inconvenientes, dentro de los que se destacan [47]:

- En la actualidad existen muchas redes de distribución de contenidos o CDN (Content Delivery Networks) que no soportan el protocolo RTSP.
- En concordancia con lo anterior, por lo general los firewalls distribuidos a lo largo de la red no soportan este tipo de paquetes.
- Tanto en el caso de los protocolos RTP y RTMP, el servidor de *streaming* debe establecer una sesión diferente por cada cliente.
- Protocolos como RTP requieren trabajar en conjunto con otros protocolos como es el caso de los protocolos RTCP (Real Time Control Protocol) y RTSP. Lo anterior hace que se incremente la información a transmitir por la red.
- RTP y RTSP requieren el uso de un servidor dedicado de *streaming* que permita el manejo de estos protocolos, lo cual representa infraestructura adicional a la considerada para la lógica del servicio.

A partir de los anteriores inconvenientes asociados a los protocolos de *streaming* tradicional, ha surgido la necesidad de que la tecnología de *streaming* esté soportada en el protocolo HTTP, buscando así aprovechar la infraestructura existente de red basada en este protocolo. A pesar de lo anterior, el protocolo HTTP añade una sobrecarga significativa en la transmisión [13]. Por lo anterior, es necesario considerar dentro del *streaming* apoyado en HTTP, que el flujo de video se adapte a las capacidades de ancho de banda y a las características del dispositivo de acceso, considerando las problemáticas de fluctuación de ancho de banda. De esta forma, se busca que en el lado del cliente, se cuente con una secuencia de vídeo continua, con la mejor calidad posible en la recepción.

De esta manera, a partir de los retos del *streaming* sobre HTTP, surge la técnica de *streaming* adaptativo, la cual ha sido especificada por 3GPP como Adaptive HTTP Streaming (AHS) y consiste en cortar un archivo multimedia en segmentos de igual duración que pueden ser codificados en diferentes resoluciones y tasas de bits, los cuales son suministrados a un servidor web para ser descargados mediante peticiones HTTP estándar. Con el fin de establecer la relación entre tasas de bits, segmentos y el orden de los mismos, AHS hace uso del archivo Media Presentation Description (MPD), el cual contiene una descripción formal sobre una colección de

datos, que representan las características técnicas del contenido multimedia [32]. Cada cliente primero solicita el MPD y a partir de dicha información, obtiene a partir del servidor los segmentos individuales que mejor se ajusten mejor a las necesidades. Así entonces, el control está del lado del cliente, de tal manera que cada segmento puede cambiar a una velocidad de bits en función de la variación del ancho de banda disponible en el lado del cliente.

La industria ha desplegado varias soluciones propietarias de *streaming* adaptativo, siendo las más representativas: Microsoft SmoothStreaming [33], Apple HTTP Live Streaming [34] y Adobe DynamicStreaming HTTP [35]. Estas plataformas de *streaming* suelen utilizar el estándar de video digital MPEG-4 H.264, junto con MPEG Advanced Audio Coding (AAC). H.264 permite lograr una mayor calidad de imagen de video para una frecuencia de bits determinada, mientras que AAC se encarga de la distribución eficiente de sonido a través de conexiones de ancho de banda moderada. Sin embargo, el inconveniente radica en que cada uno emplea sus propias técnicas de segmentación, secuencia de tiempo y formatos de MPD. Por lo anterior, si un dispositivo quisiese reproducir el contenido multimedia de un servidor de cada plataforma, debe soportar el protocolo cliente propietario, lo cual es un inconveniente [36].

Así, para solucionar los problemas anteriores y con el fin de lograr la entrega eficiente de contenidos multimedia utilizando HTTP en sus diferentes formas: adaptativa, progresiva, *streaming*/descarga y además de garantizar la interoperabilidad entre las soluciones propietarias, MPEG (Moving Picture ExpertGroup) desarrolló Dynamic Adaptive Streaming over HTTP (DASH), el cual se convirtió en un estándar internacional en noviembre de 2011, siendo publicado como ISO/IEC 23009-1:2012 en abril de 2012 [32]. En la figura 3.1 se puede apreciar un escenario simple de *streaming* adaptativo, donde se distinguen principalmente dos módulos funcionales: un servidor HTTP y un cliente DASH. El contenido multimedia codificado en el formato del estándar DASH es almacenado en el servidor web HTTP y es entregado al cliente mediante este protocolo. Así, el proceso que debe realizar el cliente para reproducir el contenido multimedia es: primero se descarga en el lado del cliente el archivo descriptor MPD, posteriormente se estima el ancho de banda disponible, una vez estimado el ancho de banda se consulta en el archivo MPD el segmento más adecuado al ancho de banda estimado, luego de lo anterior se obtienen via HTTP los segmentos (consultados en el archivo MPD) al servidor web y se empieza a reproducir el contenido multimedia en el lado del cliente. Ante cualquier cambio en el ancho de banda en el lado del cliente, se repite el proceso anterior.

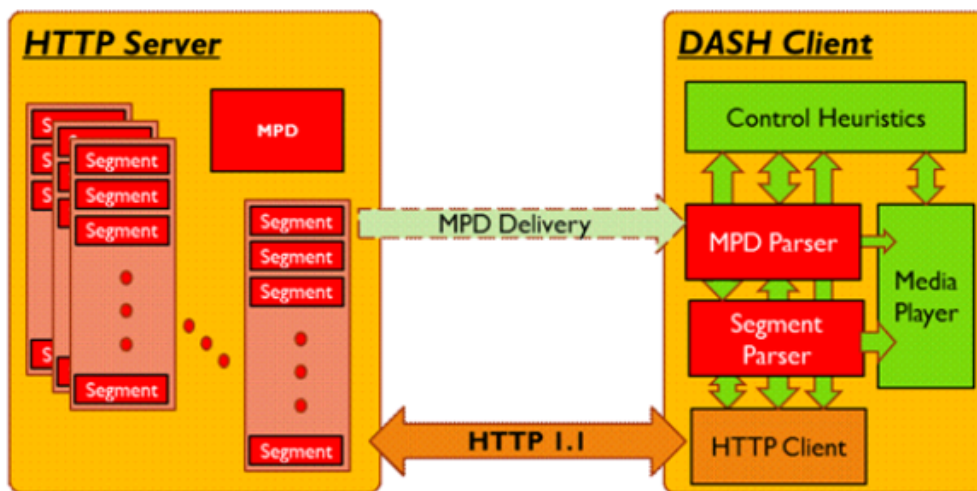


Figura 3. 1. Modelo escenario de streaming adaptativo.

De acuerdo con [37], DASH se puede definir como un sistema por el cual se proporcionan formatos, que habilitan la entrega eficiente y de alta calidad de servicios de *streaming*, que cuenta con las siguientes ventajas [38]:

- Reúso de la tecnología existente de contenedores, códec, DRM (Gestión digital de derechos), etc.
- Puede ser desplegado sobre la infraestructura actual de las redes de distribución HTTP, lo cual implica que no se requiere el uso de servidor de *streaming*, como en el caso de la infraestructura de distribución basada en los protocolos RTP y RTSP.
- El usuario percibe un mejor servicio: menor tiempo de arranque, no se requiere buffer.
- Permite seleccionar la calidad en función de la fluctuación del ancho de banda de la red y las capacidades del dispositivo cliente
- Los cambios entre calidades son automáticos y transparentes al usuario.
- Puede coexistir con tecnologías propietarias existentes de *streaming* adaptativo.

A pesar de las ventajas que provee el estándar DASH, en cuanto a la transmisión de contenidos multimedia sobre HTTP, de tal manera que el flujo se adapte a los problemas de fluctuación de ancho de banda en el lado del cliente, el proceso de generación de contenido multimedia DASH requiere la ejecución de múltiples operaciones secuenciales como son: codificación, segmentación y creación del archivo descriptor MPD. En el caso de los contenidos multimedia en formato Webm, los cuales fueron escogidos en el presente trabajo por su flexibilidad para el despliegue en entornos web, las operaciones de segmentación, agregación y generación del archivo MPD pueden ser ejecutadas de manera separada por herramientas libres como FFmpeg, simple_muxer de la librería libwebm y webm_dash_manifest de la librería webm-tools.

Lo anterior muestra que aunque existen un conjunto de herramientas libres que permiten adecuar el contenido multimedia al formato establecido por el estándar DASH, este proceso no ha sido automatizado, razón por la cual el despliegue del contenido multimedia requiere un esfuerzo mayor del lado del servidor. De acuerdo a lo anterior, en el escenario de *streaming* propuesto en este capítulo, se incluye la herramienta DASH WebM Converter, la cual fue construida con el fin de realizar los procesos de codificación, segmentación y creación del archivo descriptor MPD.

3.3 Exploración de herramientas DASH

En esta sección son descritos los conceptos, herramientas y tecnologías claves, que fueron exploradas para el soporte del *streaming* adaptativo DASH. A partir de las herramientas exploradas, se seleccionaron un conjunto de estas para la conformación de un escenario de *streaming* adaptativo DASH. Dichas herramientas permiten realizar los procesos de la codificación, difusión y consumo de contenidos multimedia DASH.

3.3.1 Video

Básicamente es una secuencia de imágenes, que formando un encadenamiento contiene escenas en movimiento [48]. La generación de video requiere dispositivos electrónicos que permitan la captura, procesamiento, almacenamiento y reconstrucción de dichas imágenes. Para el almacenamiento de video digital es requerido el uso de algún tipo de software llamado códec. Generalmente es comprimido en algún tipo de formato (MPEG-4, VP8, H.264, WMV, etc.) para facilitar su almacenamiento y transmisión, debido al ancho de banda limitado de algunas redes de datos. Una característica propia del video digital es la resolución, la cual es medida por la cantidad de píxeles verticales y horizontales que componen la imagen.

3.2.2 Códec

Es un software que contiene un conjunto de algoritmos para codificar y decodificar video y audio digital [49]. Una de las principales funciones de un códec es reducir el tamaño de video para facilitar su procesamiento y almacenamiento, encapsulando este mismo en algún tipo de formato. La visualización de un determinado formato de video requiere del codec adecuado.

3.3.3 WebM

Es un formato contenedor multimedia libre, desarrollado por Google Inc. para ser utilizado en la web con HTML5 [50]. WebM utiliza los formatos Vorbis para audio y VPx para video. Vorbis es un codec libre desarrollado por Xiph.org, de igual manera la serie de versiones del codec VPx actualmente adquirido por Google son open source. Un archivo WebM contiene agrupaciones separadas por fotogramas clave ubicados al inicio de estas, los cuales permiten una búsqueda rápida para el cliente.

3.3.4 MP4Box

Es un paquete de herramientas desarrolladas por GPAC [51], destinadas para trabajar sobre diversos contenidos multimedia, particularmente los de la especificación ISO multimedia (MP4, 3GP). MP4Box es ejecutado mediante línea de comandos y es utilizado para cifrar, codificar y multiplexar flujos de video y audio. Además permite adaptar contenidos MP4 al estándar DASH, este proceso consiste en la generación de múltiples segmentos de contenido en formato .m4s a partir de un archivo multimedia MP4 fuente, dichos segmentos son descritos en el archivo MPD especificado por el estándar. MP4Box trabaja sobre plataformas Windows, Linux, Mac Os X, Android, iOS, entre otras.

3.3.5 Osmo4

Es un reproductor multimedia de GPAC el cual permite el consumo de contenidos multimedia DASH en formato MP4, está apoyado en la mayoría de protocolos de entrega existentes haciendo posible la reproducción de cualquier tipo de contenido audiovisual. Está enfocado a los gráficos, animaciones y tecnologías de interactividad, además reproduce contenidos 2D y 3D [52]. Esta herramienta puede ejecutarse como una aplicación con interfaz gráfica (GUI) o como una consola de comandos, funciona además como un plugin embebido en navegadores web como Firefox, Opera o Internet Explorer.

3.3.6 VLC Player

Es un reproductor multimedia libre y multiplataforma (Windows, Linux, Solaris, Mac OS X, etc.), soporta múltiples formatos de video y audio (MPEG-1, MPEG-2, MPEG-4, DVD, VCD, CD-ROM, archivos DivX y 3GP, etc.), además soporta transmisiones *multicast* a través de IPV4 o IPV6 y *streaming* adaptativo del

estándar DASH [13]. VLC permite la recepción de contenido bajo demanda usando los protocolos RTP, RSTP y RTSP.

3.3.7 Red5

Es una herramienta libre utilizada para realizar *streaming* de video bajo demanda, en vivo, video chat y conferencia. Este servidor está desarrollado en el lenguaje Java y puede detectar la velocidad de conexión de los clientes además de proporcionar la mejor secuencia de video a los mismos. Red5 soporta los formatos de video FLV, MP4, 3GP y F4V, también permite la grabación de *streaming* proveniente de sus clientes en formato FLV y AVC-ACC [53].

3.3.8 Darwin

Es un servidor libre de *streaming* multimedia, que utiliza los protocolos RTP/RSTP, el cual fue desarrollado por Apple en 1999. Es una herramienta que proporciona un alto nivel de personalización, utilizada para realizar *streaming* de video bajo demanda, en vivo y video chat [53]. Darwin funciona sobre plataformas Mac OS, Linux, Windows, FreeBSD, Tru64 Unix y Solaris. Además soporta múltiples formatos multimedia incluyendo H.264/MPEG-4 AVC y 3GP.

3.3.9 Live555

Es un servidor de *streaming* multimedia, el cual utiliza los protocolos RTP, RTSP y SDP (Session Description Protocol) [54], sin embargo no soporta *streaming* adaptativo. Live555 es libre y de código abierto, lo cual significa que puede modificarse y configurarse de acuerdo a requerimientos propios. Algunos de los formatos multimedia soportados por este servidor están: MPEG-2 (.mpeg), MPEG-4 (.m4e), MPEG Transport Stream (.ts), WebM (.webm), H264 (.264), MP3 (.mp3), AAC (.aac), etc.

3.3.10 Apache

Es un servidor web HTTP de código abierto, destacado por su seguridad (SSL y TLS), rendimiento (capaz de manejar más de un millón de visitas por día) y por ser muy flexible y altamente configurable con módulos como PHP, Python, Perl, Cband, Ruby, entre otros [55]. Apache es el servidor web por defecto en muchas distribuciones Linux. Su versión local es muy útil para los desarrolladores de

aplicaciones web. Su archivo `apache2.conf` es el más importante, puesto que es el que define el comportamiento general del servidor y es el encargado de acceso a los distintos módulos del mismo.

3.3.11 FFmpeg

Es una herramienta de línea de comandos perteneciente al proyecto libre FFMPEG, cuyas funciones principales son: convertir archivos multimedia entre diferentes formatos, cambiar el tamaño de la imagen de video, generar archivos multimedia a diferentes frecuencias de muestreo y velocidad de fotogramas, entre otras. FFMPEG contiene múltiples bibliotecas útiles para codificar, decodificar, mux y demux, prácticamente cualquier tipo de archivo multimedia [56]. Dentro del presente trabajo esta herramienta fue utilizada para cambiar el formato original de diferentes contenidos multimedia al formato WebM. Así mismo FFMPEG es utilizada en segundo plano por la herramienta de codificación automática, con el fin de obtener múltiples flujos de audio y video a diferentes tasas de bits, a partir de un archivo multimedia WebM base.

3.3.12 Libwebm

Es una biblioteca de código abierto asociada al proyecto WebM de Google Inc., la cual contiene el *script*¹² `sample_muxer`, escrito en lenguaje C++. Este es utilizado para agregar índices de búsqueda para cada fotograma clave y para alinear puntos de sincronización en archivos multimedia WebM. Esta biblioteca provee además software para el control de *streaming* DASH y el acceso a los segmentos multimedia descargables. El *script* `sample_muxer` es utilizado en segundo plano por la herramienta propuesta en el presente trabajo, con el fin de agregar los índices de búsqueda a los contenidos multimedia previamente codificados en el formato WebM [57].

3.3.13 webm-tools

Es otra biblioteca de código abierto asociada al proyecto WebM de Google Inc., la cual contiene *scripts* escritos en el lenguaje C++, útiles para cifrar flujos de video y generar el archivo de manifiesto XML DASH, usado para describir el contenido multimedia del *streaming* adaptativo. El *script* `webm_dash_manifest` de la biblioteca `webm-tools` es utilizado en segundo plano por la herramienta propuesta en este

¹² Un script es un conjunto de órdenes guardadas en un archivo de texto, el cual es ejecutado por lotes o línea a línea, en tiempo real por un intérprete.

trabajo, para la generación del archivo de manifiesto XML DASH de los contenidos multimedia previamente codificados e indexados [57] .

3.3.14 webm-dash-javascript

Es una librería basada en JavaScript [50], la cual permite la integración del estándar de *streaming* adaptativo DASH en la Web, para lo cual hace uso de las extensiones para manejo de medios en HTML5 propias del navegador Google Chrome (Media Source API). Gracias a sus componentes `bandwidth_manager.js`, `dash_player.js`, `dash_parser.js`, `webm_parser.js`, entre otros, es posible realizar el consumo de segmentos de medios basados en WebM y segmentos basados en el formato de archivo ISO Base Media. El presente trabajo hizo uso de las anteriores librerías javascript y del componente HTML5 de reproducción, para el consumo del contenido multimedia en formato DASH desde una página web en el navegador Google Chrome.

3.4 Selección de herramientas DASH

A partir de las herramientas y tecnologías descritas en la sección 3.2, se seleccionaron un conjunto de estas para los procesos de codificación, difusión y recepción de *streaming* adaptativo, considerando la flexibilidad y capacidad de integración de estas tecnologías con la web. Así, en la tabla 3.1 se resumen las herramientas seleccionadas para la implementación del escenario de *streaming* adaptativo DASH planteado en este presente trabajo.

Tabla 3. 1. Herramientas escenario de streaming adaptativo DASH. Fuente: propia.

Tarea	Herramientas	Descripción
Codificación	FFmpeg	Esta herramienta permite la codificación y adaptación previa de los contenidos multimedia WebM, de tal manera que estos puedan ser adecuados al formato del estándar DASH.
	Libwebm	Sobre los archivos WebM codificados y adecuados por FFmpeg, esta herramienta permite agregar índices de búsqueda para fotogramas clave de los flujos de video y audio del formato WebM, de tal manera que puedan ser soportados por el estándar DASH.

	Webm-tools	Esta herramienta permite generar el archivo descriptor MPD para los flujos de video y audio WebM codificados y adecuados en formato DASH por las herramientas FFmpeg y Libwebm. Este archivo contiene información detallada de los diferentes segmentos del archivo WebM, dentro de la que se destaca: el ancho de banda y la resolución asociadas al segmento.
Difusión	Apache	Este servidor permite la gestión de peticiones de tipo HTTP, el almacenamiento de los recursos multimedia y la implementación de la lógica del negocio de servicios como el de VoD. Dentro de los recursos multimedia se encuentran los archivos WebM codificados y adecuados en el formato del estándar DASH, los cuales pueden ser accedidos vía HTTP.
Recepción	VLC player	Esta herramienta permite la reproducción de contenidos multimedia WebM en formato DASH.
	Webm-dash-javascript	Esta librería basada en javascript permite la estimación del ancho de banda en el lado del cliente, así como el acceso y consumo de los segmentos DASH WebM asociados al ancho de banda estimado.

3.5 Escenario de streaming adaptativo DASH

En esta sección se presenta el diagrama extremo a extremo del escenario de *streaming* adaptativo DASH, el cual fue constituido con ayuda de las herramientas y tecnologías presentadas en la tabla 3.1. El escenario de *streaming* está formado por tres módulos principales (ver figura 3.1): módulo de codificación, módulo de difusión y módulo de recepción.

Módulo de codificación: Este módulo es el responsable de realizar los procesos de codificación, segmentación y generación del archivo descriptor del contenido multimedia adaptativo MPD. Las funciones de codificación son realizadas por la herramienta FFmpeg, las operaciones de segmentación y agregación de los

encabezados son ejecutadas por la herramienta `sample_muxer` de la librería `libwebm` y la tarea de creación del archivo de manifiesto MPD es desarrollada por la herramienta `webm_dash_manifest` de la librería `webm-tools` (ver tabla 3.1). Así, considerando que las funciones realizadas por el módulo de codificación, son ejecutadas por múltiples herramientas de manera secuencial, se desarrolló la herramienta `DashWebMConverter` cuyo objetivo es automatizar el proceso de generación de contenidos multimedia DASH. Así, dentro de este módulo se encuentran los contenidos multimedia WebM a codificar y la herramienta automática de codificación `DashWebMConverter`, en conjunto con las librerías usadas en segundo plano (`FFmpeg`, `libwebm` y `webm-tools`). Los anteriores componentes fueron desplegados sobre el sistema operativo Linux OpenSUSE, el cual facilita la configuración e integración de las herramientas que corren en segundo plano, así como el seguimiento de la herramienta de codificación propuesta. Así, dentro de este módulo se encuentran los contenidos multimedia WebM a codificar y la herramienta automática de codificación `DashWebMConverter`, en conjunto con las librerías y herramientas usadas en segundo plano (`FFmpeg`, `libwebm` y `webm-tools`). Los anteriores componentes fueron desplegados en el presente trabajo a través del sistema operativo Linux OpenSUSE, el cual facilita la configuración e integración de las herramientas que corren en segundo plano, así como el monitoreo de la herramienta de codificación propuesta, ver Figura 3.2.

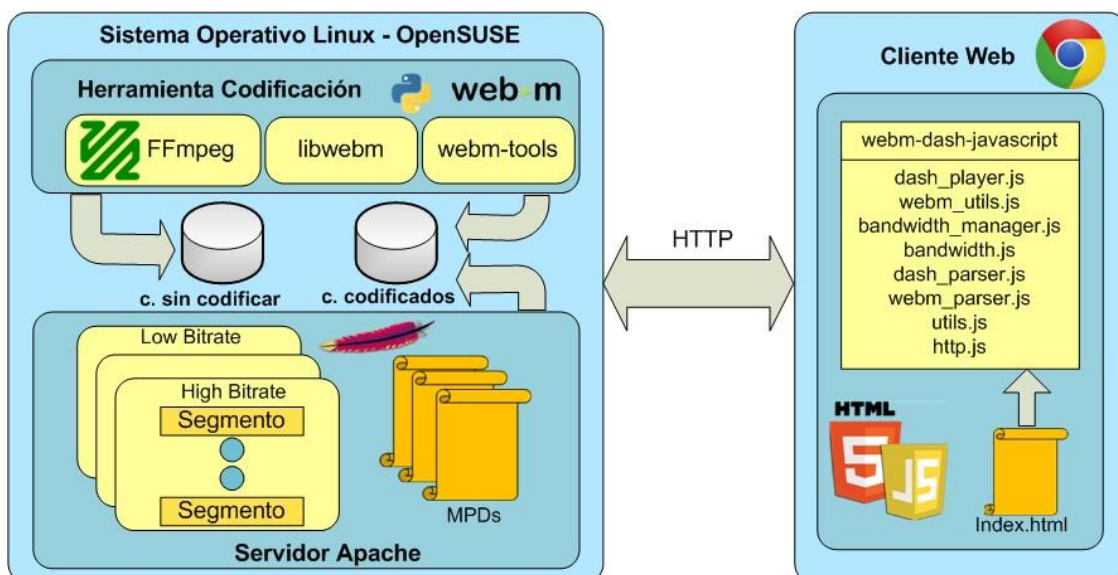
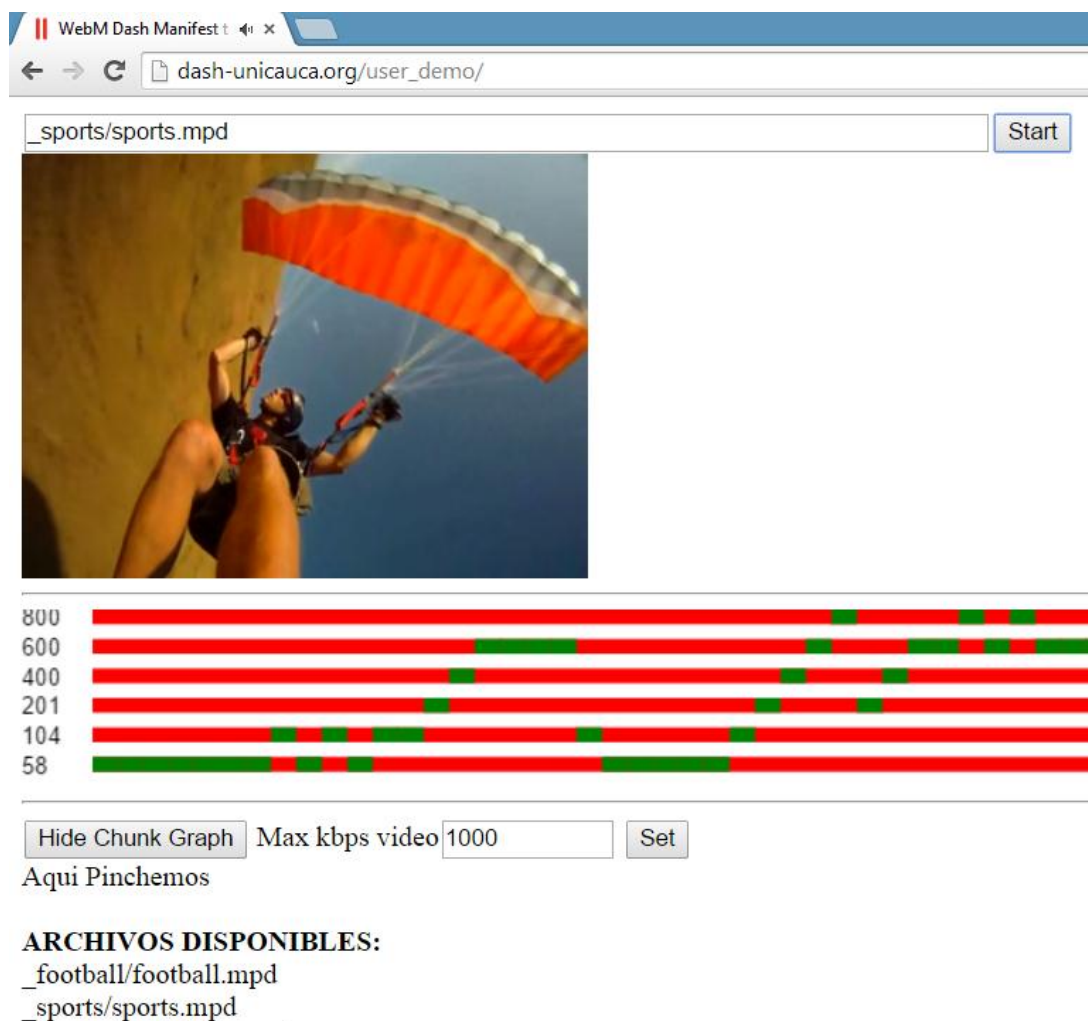


Figura 3.2. Escenario de streaming adaptativa. Fuente: propia.

Módulo de difusión: Este módulo es el encargado de almacenar y transmitir los contenidos multimedia WebM DASH solicitados por los clientes de *streaming* web mediante peticiones HTTP. Cada uno de los contenidos WebM está codificado en

diferentes calidades o tasas de bits y tiene asociado un archivo descriptor MPD, el cual incluye características de los flujos de video y audio tales como: URL de los flujos, tipo de contenedor, codecs de audio y video, tasa de bits, resolución de pantalla del video, tasa de muestreo del audio, entre otros. Los contenidos multimedia y sus archivos de descripción MPD se encuentran alojados dentro de un servidor web, lo cual es requisito fundamental para la recepción de peticiones desde los clientes web y para la transmisión de *streaming* adaptativo basado en el protocolo HTTP. A nivel de implementación, en este trabajo se hizo uso del servidor web Apache, corriendo sobre el sistema operativo Linux OpenSUSE.

Módulo de recepción: Este módulo está formado por el cliente Web de *streaming*, el cual solicita mediante peticiones HTTP, el contenido multimedia desde el servidor de *streaming* adaptativo DASH y lo reproduce en una página Web, ver Figura 3.3.



WebM Dash Manifest t x

dash-unicauca.org/user_demo/

_sports/sports.mpd Start

800
600
400
201
104
58

Hide Chunk Graph Max kbps video 1000 Set

Aqui Pinchemos

ARCHIVOS DISPONIBLES:
_football/football.mpd
_sports/sports.mpd

Figura 3. 3. Cliente web de streaming adaptativo DASH. Fuente: propia.

Durante el proceso de solicitud y reproducción, el cliente web se encarga de estimar periódicamente el ancho de banda disponible, de tal manera que según esta información y la descrita en el archivo de manifiesto MPD asociado a cada

contenido, se puedan reproducir los segmentos de contenidos multimedia adecuados. Para estimar el ancho de banda y decodificar el archivo de manifiesto MPD del contenido multimedia adaptativo, se hace uso de la librería javascriptwebm-dash-javascript. Con respecto a la reproducción del contenido multimedia adaptativo, esta librería incluye un cliente web demo (dash-player.html), que permite la reproducción del contenido multimedia adaptativo, mediante la integración del componente de video de HTML5 y las funcionalidades de estimación ancho de banda y decodificación del manifiesto MPD. Este cliente muestra además la variación del ancho de banda durante el proceso de reproducción, mediante franjas de color rojo, las cuales representan las diferentes tasas de bits en las que ha sido codificado el contenido multimedia adaptativo (ver figura 3.3). Finalmente es importante mencionar que para lograr el cambio eficiente entre segmentos multimedia DASH, es necesario utilizar el navegador Chrome en su versión 18 o superior.

3.6 Herramienta Dash WebM Converter

Esta sección se describe los diferentes *scripts* y librerías que se seleccionaron para la conformación e implementación de la herramienta Dash WebM Converter. Dentro de estas se encuentran: FFmpeg, sample_muxer y Webm_dash_manifest, las cuales fueron escogidas teniendo en cuenta que son de código libre y permiten la codificación y adecuación de los contenidos multimedia WebM, el cual es un formato flexible para el despliegue y consumo de contenidos multimedia en la web. Las herramientas seleccionadas funcionan en segundo plano y son ejecutadas desde la aplicación Dash WebM Converter, la cual fue desarrollada en el lenguaje Python. La secuencia de operación de estas herramientas y/o librerías es presentada en la figura 3.4, en donde la ejecución de la primera (FFmpeg) codifica y genera los flujos de audio y video a partir de un archivo base WebM, la ejecución de la segunda (sample_muxer) agrega índices de búsqueda a cada fotograma clave del archivo WebM y finalmente la tercera (webm_dash_manifest) se encarga de la creación del archivo descriptor MPD de los contenidos codificados y adaptados por las dos anteriores herramientas. A continuación se describe de forma detallada la manera en la que cada una de las herramientas mencionadas fue utilizada. De igual manera en el anexo G de la presente monografía se detalla el proceso manual para la generación del contenido multimedia de acuerdo al estándar DASH, proceso que es automatizado por la herramienta DASH WebM Converter.

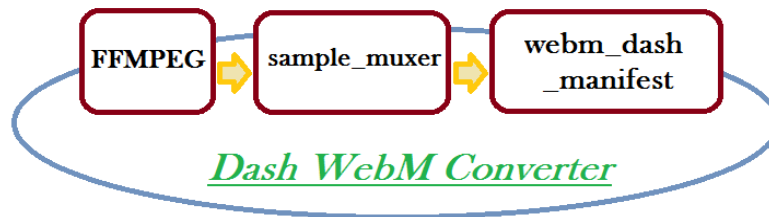


Figura 3. 4. Scripts usados en la herramienta de codificación automática. Fuente: propia.

3.6.1 FFmpeg

Es una herramienta libre que se ejecuta mediante línea de comandos ya sea en windows o linux [56]. Su forma manejo es descrita en el anexo I de la presente monografía. Este software posee múltiples funciones de codificación y adecuación de contenidos multimedia, algunas de las cuales fueron usadas en el presente trabajo con el fin de generar flujos de video y audio a diferentes tasas de bits o velocidades a partir de un archivo WebM base o de origen.

Dentro del proceso de generación de contenido multimedia DASH WebM, FFmpeg hace la función de pre-codificación, de tal modo que los flujos multimedia generados son adaptados por el *script* sample_muxer, descrito a continuación. Para que la herramienta FFmpeg pueda funcionar de manera correcta, se requiere el uso de librerías apropiadas para la codificación de video y audio como son: libvpx y libvorbis respectivamente.

3.6.2 Sample_muxer

Es un *script* escrito en lenguaje C++, perteneciente a la biblioteca LibWebm y que funciona mediante línea de comandos. Este *script* es utilizado para agregar índices de búsqueda a cada fotograma clave de un archivo WebM, con el fin de facilitar el canjeo entre cada *stream*¹³ de video al momento de reproducir el contenido multimedia adaptativo [57].

El tiempo de trabajo de este *script* es directamente proporcional al tamaño del *stream* de video o audio sobre el cual es aplicado. Su parámetro de entrada es el archivo WebM previamente adaptado por la herramienta FFMPEG y su salida es otro archivo WebM completamente adaptado de acuerdo al estándar DASH.

¹³ Un stream es cada uno de los segmentos que forman parte del flujo multimedia transmitido por la red.

3.6.3 Webm_dash_manifest

Este es un *script* escrito en lenguaje C++ que viene incluido dentro del proyecto libre WebM-tools perteneciente al proyecto WebM de Google Inc. [57]. La ejecución de este *script* mediante línea de comandos, permite la generación del archivo descriptor MPD del contenido multimedia WebM DASH, teniendo como parámetros de entrada cada uno de los flujos de video y audio que ya han sido adaptados al estándar DASH. Cada flujo es identificado dentro del archivo MPD por un id diferente, además los *stream* de video y audio son descritos en dos agrupaciones diferentes.

3.7 Proceso de codificación de streaming adaptativo

Una vez ilustrados los diferentes componentes de la herramienta automática de codificación Dash WebM Converter, en esta sección se describen las diferentes fases del proceso de codificación de los contenidos multimedia adaptativos WebM DASH, realizados por la herramienta propuesta (ver figura 3.5).

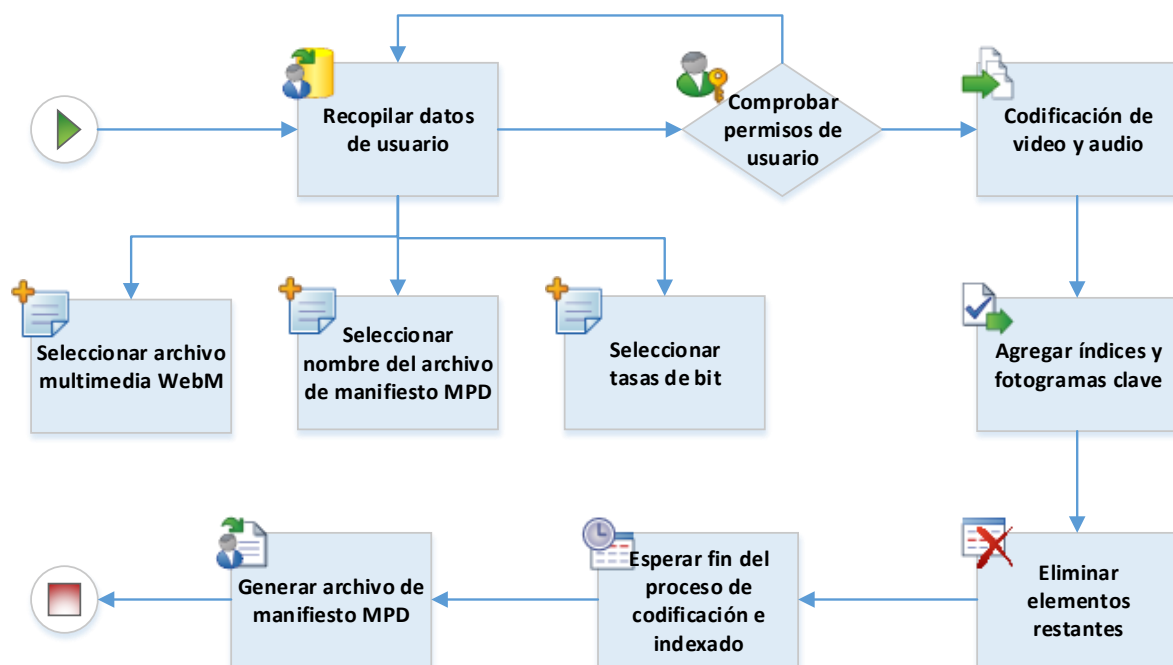


Figura 3. 5. Proceso de codificación de streaming adaptativo. Fuente: propia.

Los archivos WebM para *streaming* adaptativo deben contar con flujos de audio y video multiplexados, con un índice de búsqueda asociado a cada fotograma clave del flujo de video, de tal forma que se permita el cambio entre diversos flujos de *streaming* sin discontinuidad, a medida que varía el ancho de banda. A partir de lo anterior, una de las funciones de la herramienta de codificación es adaptar los

contenidos WebM para ser transmitidos, ya que por defecto cada uno de estos archivos corresponde a un *stream*, sin fragmentación y sin puntos de sincronización.

La primera fase del proceso de codificación consiste en recibir, desde la interfaz gráfica de la herramienta, tres datos básicos para este proceso, como son: la ruta del archivo multimedia WebM, las tasas de video a codificar, de acuerdo a la calidad del contenido multimedia y el nombre del documento de manifiesto MPD (ver la figura 3.6).

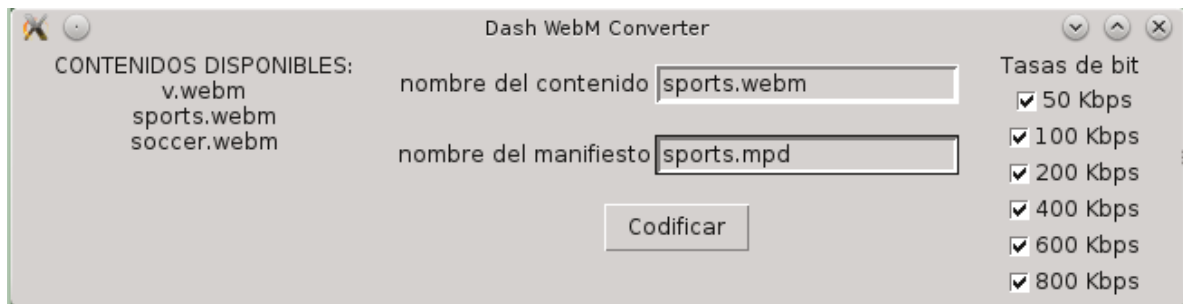


Figura 3. 6. Interfaz Gráfica de la herramienta Dash WebM Converter. Fuente: propia.

Una vez brindados y comprobados los permisos del sistema operativo, inicia la fase de codificación, en la cual la herramienta Dash WebM Converter genera en paralelo tantos flujos de video WebM, como tasas de bits sean seleccionadas en la primera fase. De igual forma, un solo flujo de audio WebM por todos los flujos de videos codificados es generado, considerando que el flujo de audio no tiene mayor incidencia en el comportamiento del contenido multimedia con respecto al ancho de banda. Para realizar el procedimiento anterior, la herramienta Dash WebM Converter hace invocaciones en segundo plano y en paralelo a la herramienta FFmpeg, la cual es ejecutada como un proceso (modo silencioso) del sistema operativo Linux.

Es importante mencionar que dentro de los parámetros usados para la invocación de FFmpeg, debe tenerse en cuenta el uso de la librería libvpx (códec VP8) para la codificación de video y de la librería libvorbis para la codificación de audio. Así mismo, debe indicarse a FFMPEG el intervalo de fotogramas máximo y mínimo entre cada fotograma clave. Dichos valores son utilizados para alinear todos los puntos de sincronización en las secuencias de video a canjear, (ver la figura 3.7).

```

Stream #0:0 -> #0:0 (vp8 -> libvpx)
Press [q] to stop, [?] for help
size= 3024kB time=00:04:24.16 bitrate= 93.8kbits/s    bitrate= 51.3kbits/s
video:0kB audio:2941kB subtitle:0 data:0 global headers:4kB muxing overhead 2.686046%
>>>>>> ./libwebm/samplemuxer -i _sports/sports_audio.webm -o _sports/sports_audio_sm.web
m -output_cues 1 -cues_on_audio_track 1 -max_cluster_duration 5 -audio_track_number 2
>>>>>> rm -r _sports/sports_audio.webm
>>>>>>FIN>>>> archivo multimedia sports.webm >>> de audio codificado a 128 kbps
frame= 7916 fps= 16 q=0.0 Lsize= 1882kB time=00:04:24.13 bitrate= 58.4kbits/s
video:1827kB audio:0kB subtitle:0 data:0 global headers:0kB muxing overhead 2.984974%
>>>>>> ./libwebm/samplemuxer -i _sports/sports_50.webm -o _sports/sports_50k_sm.webm
>>>>>> rm -r _sports/sports_50.webm88kB time=00:03:12.45 bitrate= 203.8kbits/s
>>>>>>FIN>>>> archivo multimedia sports.webm >>> codificado a 50kbps
frame= 6933 fps= 13 q=0.0 size= 2972kB time=00:03:51.33 bitrate= 105.3kbits/s

```

Figura 3. 7. Fase de codificación de los contenidos WebM. Fuente: propia.

Después de que son generados los flujos de video WebM con diferentes tasas de bit y considerando que la herramienta FFmpeg no permite la alineación de los parámetros “Cluste” y “Cues” de los contenidos multimedia, la siguiente fase del proceso consiste en alinear los puntos de sincronización de las secuencias de vídeo y agregar los índices de búsqueda para cada fotograma clave dentro de los mismos flujos de video. Para realizar lo anterior, la herramienta Dash WebM Converter ejecuta en segundo plano el *script* `sample_muxer` de la librería `libwebm`. Los contenidos resultantes tras la ejecución del anterior proceso (nombre terminado en “_sm”) son presentados en la figura 3.8. Por su parte, el resto de contenidos innecesarios de las fases anteriores son eliminados por la herramienta Dash WebM Converter.

Finalmente, la última fase consiste en la generación del archivo de manifiesto MPD, el cual contiene información de los flujos de video y audio disponibles para canjear al momento de ser solicitados por el cliente web de *streaming* DASH. Para generar el archivo MPD, la herramienta Dash WebM Converter ejecuta en segundo plano el *script* `webm_dash_manifest` de la librería `webm-tools`, enviándole como parámetro el listado de archivos asociados a los flujos de video y audio generados a partir de las tasas de bits asignadas en la fase 1.

El documento de manifiesto MPD generado, comprende características de los flujos de video y audio tales como: URL de los flujos, tipo de contenedor, codecs de audio y video, tasa de bits, resolución de pantalla del video, tasa de muestreo del audio, entre otros. La figura 3.8 muestra los archivos de *streaming* adaptativo generados, en conjunto con su archivo descriptor MPD.

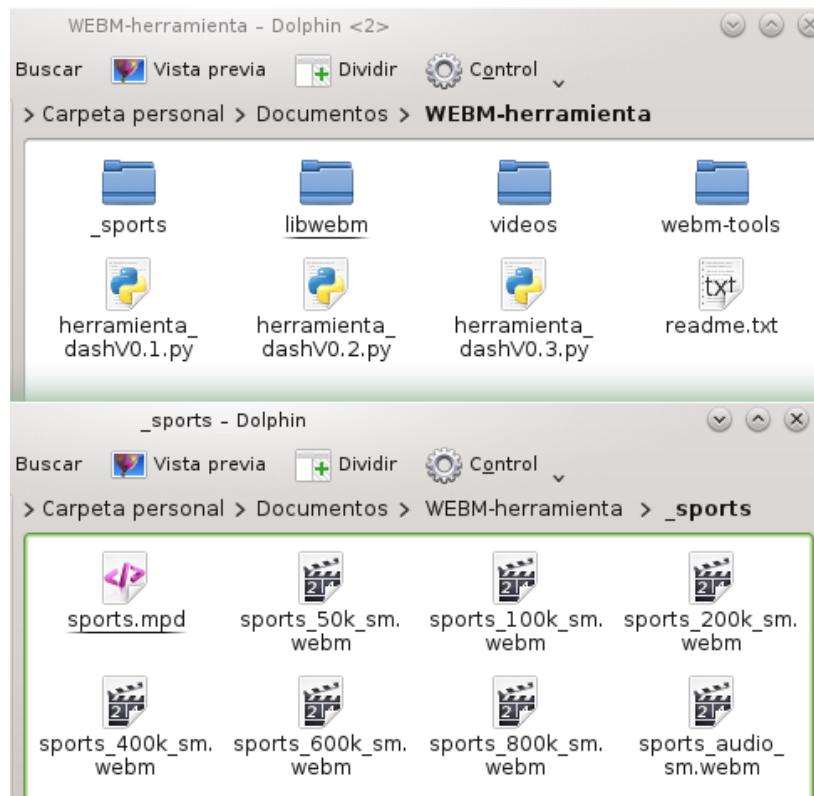


Figura 3. 8. Recursos de la herramienta de codificación automática. Fuente: propia.

3.8 Módulos herramienta de codificación

Esta sección presenta el diagrama modular de la herramienta Dash WebM Converter, la cual fue desarrollada en el lenguaje Python. Tal como fue mencionado en la sección anterior, la función de la herramienta Dash WebM Converter es ejecutar de manera secuencial las tareas de codificación, segmentación y generación del archivo de manifiesto MPD para contenidos multimedia adaptativos DASH WebM. Para realizar los procesos anteriores la herramienta desarrollada, hace uso en segundo plano y respectivamente de las librerías y/o herramientas libres: FFmpeg, sample_muxer y webm_dash_manifiest.

En el diagrama de la figura 3.9 se distinguen los siguientes los siguientes bloques funcionales principales: interfaz gráfico de usuario, lanzador de hilos, ejecutor de comandos, lector de archivos multimedia.

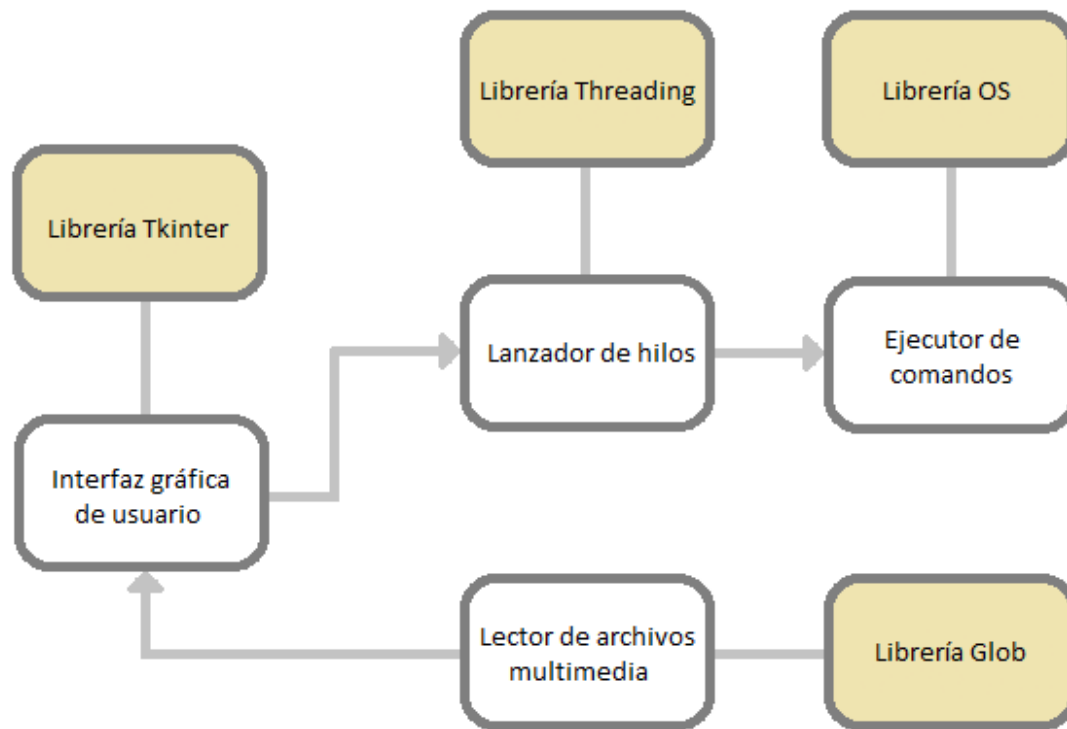


Figura 3. 9. Diagrama de bloques herramienta de codificación automática. Fuente: propia.

El módulo de interfaz gráfica de usuario está conformado por la librería Tkinter de Python, la cual permite desplegar diferentes componentes gráficos como: campos de texto, etiquetas y botones, a través de los cuales es solicitado el archivo multimedia WebM base a codificar, el nombre del documento de manifiesto MPD y las tasas de bits a codificar. De manera simultánea en que es lanzada la herramienta Dash WebM Converter, puede visualizarse en la interfaz gráfica el contenido multimedia disponible para codificar, esta tarea es realizada por el modulo lector de archivos multimedia, a través de la librería Glob (ver la figura 3.9).

Por su parte el módulo Lanzador de Hilos es encargado de crear tantas instancias de hilos o procesos simultáneos, como tasas de bits seleccionadas desde la interfaz gráfica, más un hilo adicional para el contenido de audio. Para lo anterior, la herramienta Dash WebM Converter hace uso de la librería Threading de Python. Cada hilo de video hace una invocación en segundo plano de la herramienta FFmpeg, enviando como parámetro una tasa de codificación diferente, mientras que en el caso del hilo de audio, una tasa de bits estándar es elegida. Al terminar la tarea de codificación por parte de cada hilo lanzado, la herramienta Dash WebM Converter ejecuta el *script* `sample_muxer` de la librería `libwebm`, con el fin de agregar los índices de búsqueda para cada fotograma clave de cada flujo de video.

Finalmente, el módulo ejecutor de comandos, es el encargado de ejecutar comandos del sistema operativo Linux a través de la librería OS de python. Este

módulo interactúa de manera directa con la consola del sistema operativo y permite lanzar en segundo plano y durante diferentes momentos del proceso de codificación, las herramientas y/o librerías: FFMPEG, sample_muxer y webm_dash_manifest.

3.9 Archivo descriptor MPD

En esta sección se describe la estructura básica del documento de manifiesto MPD generado por la herramienta de codificación Dash WebM Converter en su fase final (ver figuras 3.10 y 3.11). A modo de ejemplo se considera uno de los documentos MPD generados por la herramienta planteada en este capítulo.

Tal como fue mencionado anteriormente, este documento descriptor es generado por la herramienta Dash WebM Converter a través de la ejecución en segundo plano del *script* webm_dash_manifest perteneciente a la librería webm-tools. El documento MPD está basado en el lenguaje de marcado extensible XML y contiene las siguientes configuraciones dentro de la etiqueta principal <MPD>:

- El parámetro “Perfil de presentación” está fijado por defecto en el perfil: “webm-on-demand”, indicando así el modo de transmisión del contenido multimedia (ver figura 3.9).
- La etiqueta “Periodo” incluye el punto de inicio y la duración total del contenido multimedia (ver la figura 3.9).

```

9  <profiles="urn:webm:dash:profile:webm-on-demand:2012">
10 <Period id="0" start="PT0S" duration="PT264.16S" >
11   <AdaptationSet id="0" mimeType="video/webm" codecs="vp8" width="320" height="240"
12     subsegmentAlignment="true" subsegmentStartsWithSAP="1" bitstreamSwitching="true">
13     <Representation id="0" bandwidth="74285">
14       <BaseURL>sports_50k_sm.webm</BaseURL>
15       <SegmentBase indexRange="1925436-1926323">
16         <Initialization range="0-248" />
17       </SegmentBase>
18     </Representation>
19     <Representation id="1" bandwidth="119023">
20       <BaseURL>sports_100k_sm.webm</BaseURL>
21       <SegmentBase indexRange="3435010-3435898">
22         <Initialization range="0-248" />
23       </SegmentBase>
24     </Representation>

```

Figura 3. 10. Documento de descripción MPD (cabecera). Fuente: propia.

- La etiqueta <AdaptationSet> con atributo id=0, describe la información de los flujos de video codificados a diferentes tasas de bits, especificando el formato de empaquetamiento, el códec de video y la resolución del contenido multimedia (ver figura 3.11). Adicionalmente esta etiqueta incluye los atributos: segmentAlignmentFlag = true y bitstreamSwitchingFlag = true, los cuales al

estar fijados en “true”, indican que es posible el intercambio de streams de video durante la reproducción del contenido multimedia, de acuerdo al ancho de banda disponible.

- La etiqueta <AdaptationSet> con atributo id=1, describe el único flujo de audio asociado los diferentes flujos de video codificados, indicando el formato de empaquetamiento, el códec de audio y la tasa de muestreo utilizados (ver figura 3.10).
- La información más específica de cada uno de los flujos de video y audio codificados está contenida dentro de la etiqueta <Representation> con un id para cada *stream* individual (ver figuras 3.10 y 3.11). En esta etiqueta hace parte del nodo de nivel superior <AdaptationSet>, y describe el ancho de banda asociado a cada flujo de audio o video y la URL de cada *stream* dentro del servidor de difusión.

```

50 <AdaptationSet id="1" mimeType="audio/webm" codecs="vorbis" audioSamplingRate="44100"
51 subsegmentStartsWithSAP="1">
52 <Representation id="6" bandwidth="119772">
53 <BaseURL>sports_audio_sm.webm</BaseURL>
54 <SegmentBase indexRange="3096600-3097489">
55 <Initialization range="0-4520" />
56 </SegmentBase>
57 </Representation>
58 </AdaptationSet>

```

Figura 3. 11. Documento de descripción MPD (pie). Fuente: propia.

3.10 Servicio VoD DASH

A partir del escenario de *streaming* adaptativo propuesto en la sección 3.4 del presente capítulo, se desarrolló un servicio de video bajo demanda apoyado en el estándar DASH, cuyo diagrama de flujo de trabajo se presenta en la sección 3.9. Dentro de este diagrama se pueden identificar varias etapas como son: codificación de contenidos, publicación de contenidos, difusión y recepción de contenidos (ver figura 3.11). El proceso es iniciado a partir de la solicitud del administrador del servicio, el cual accede a las funcionalidades de la herramienta de codificación desde un navegador web. La herramienta DASH WebM Converter a su vez, recibe los archivos WebM desde el cliente o bien puede utilizar aquellos que ya están alojados en el servidor. Una vez confirmado el proceso de codificación, se procede automáticamente y de manera secuencial con los pasos 2 y 3, los cuales están asociados a las tareas de codificación y publicación.

Una vez iniciado el proceso de codificación, la herramienta adapta el contenido WebM seleccionado al estándar DASH. Después de adaptado el contenido, este es alojado dentro de los recursos del servidor web y añadido o publicado en el catálogo de contenidos del servicio de VoD. De esta manera el contenido multimedia está listo para ser consumido desde el cliente web, para lo cual se hace la estimación

del ancho de banda local y se consulta en el archivo MPD asociado al contenido, el segmento adecuado para dicho ancho de banda. Finalmente se accede y se reproduce el segmento del contenido multimedia consultado en el archivo MPD.

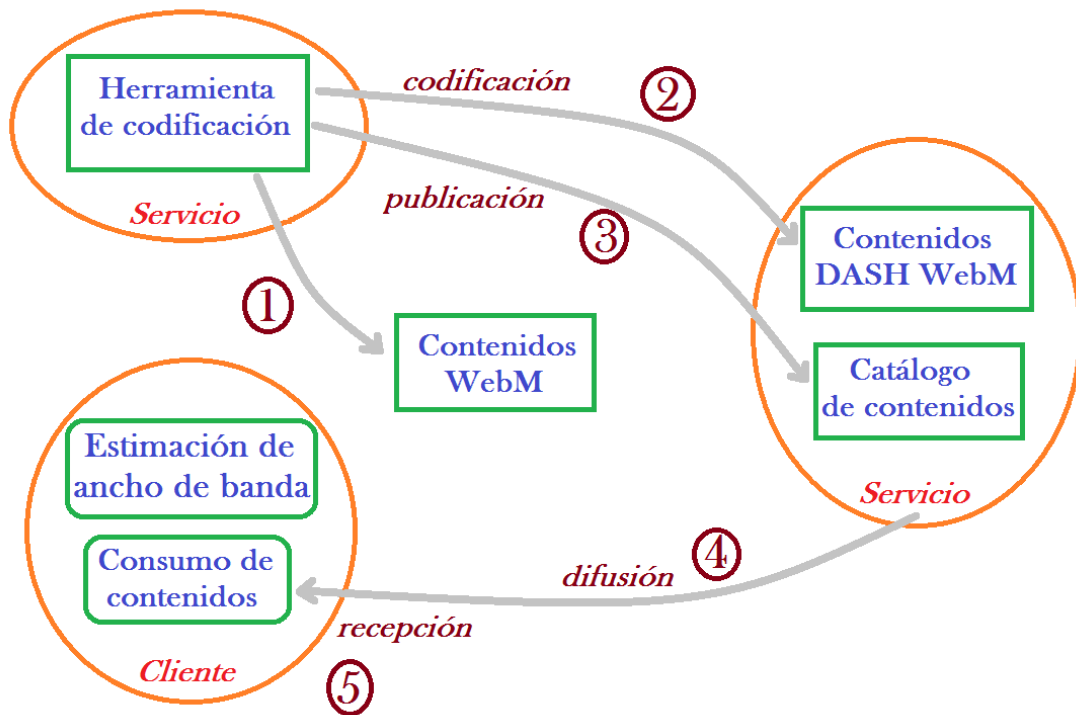


Figura 3. 12. Flujo de trabajo del servicio VoD. Fuente: propia.

Así, a partir de las diferentes tareas realizadas presentadas en la figura 3.12, en la figura 3.13 se describen de forma detallada las 4 diferentes etapas que conforman el escenario de *streaming* adaptativo DASH, las cuales son: Codificación, Agregación, Difusión y Consumo. En la etapa de Codificación, los contenidos multimedia de entrada son codificados y adecuados al estándar DASH. En la fase de Agregación, los contenidos multimedia son alojados en el servidor web y publicados en el catálogo de contenidos multimedia. Así mismo en esta fase son agregados los servicios interactivos asociados al servicio de VoD. En la fase de distribución, los contenidos multimedia DASH codificados, son distribuidos a través de la red usando el protocolo HTTP. Finalmente, en la fase de Consumo o de Cliente, el ancho de banda local es estimado y de acuerdo a dicho ancho de banda se solicita el segmento adecuado y alojado en el servidor web. Así mismo en esta fase los servicios interactivos son consumidos por el cliente web.

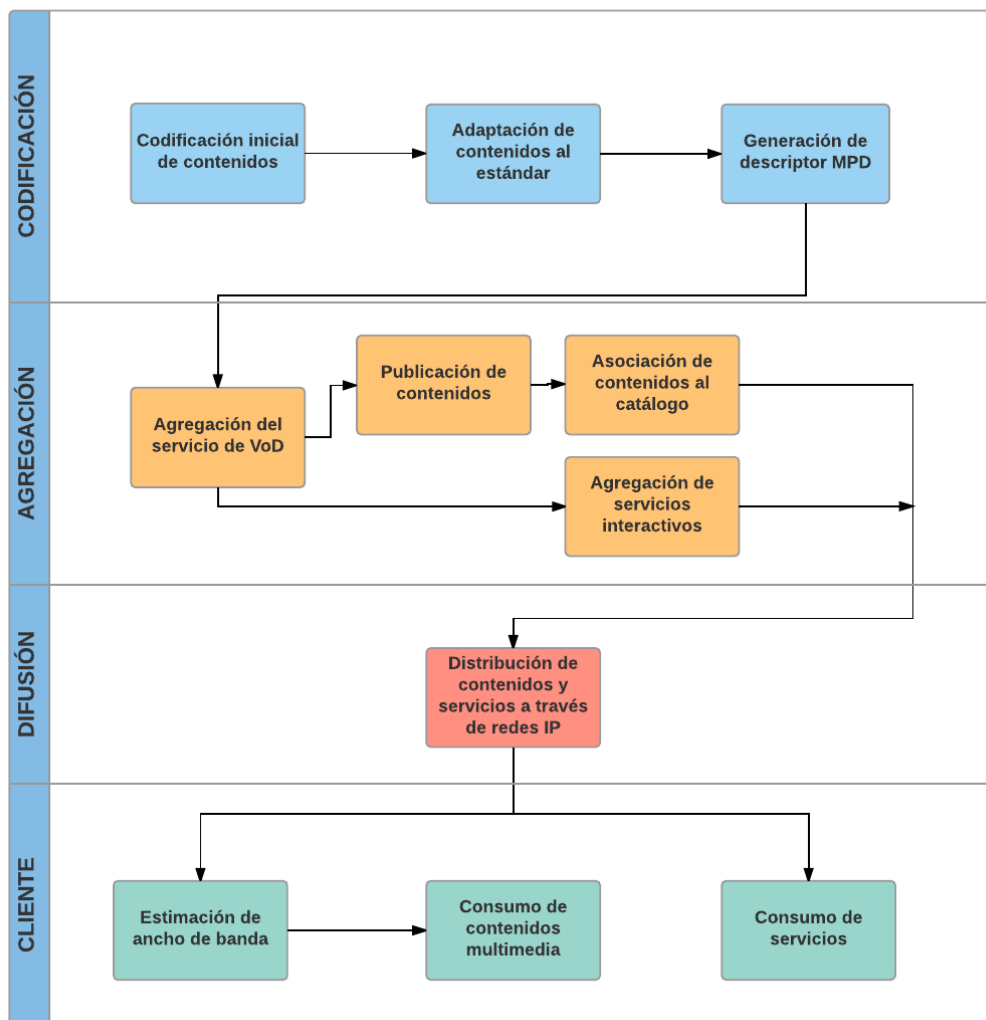
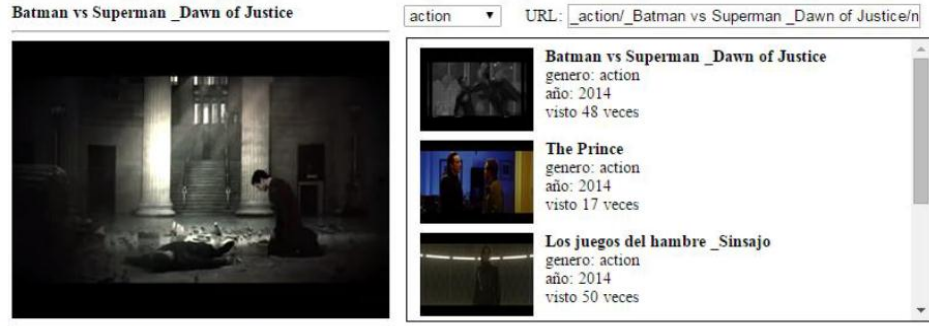


Figura 3. 13. Flujo de trabajo del servicio VoD. Fuente: propia.

De acuerdo a las fases descritas en la figura 3.13, en la figura 3.14 se presenta un servicio de VoD soportado en el escenario de *streaming* adaptativo DASH desarrollado y presentado en este capítulo. Este servicio fue construido haciendo uso de contenidos multimedia asociados a trailers de películas, los cuales fueron previamente codificados y adecuados al formato de *streaming* adaptativo DASH. El servicio de VoD está constituido por un componente de reproducción ubicado a la izquierda, el cual fue implementado usando HTML5. Así mismo, en la parte inferior del componente de reproducción se cuenta con una aplicación que posibilita la valoración del contenido multimedia y su clasificación dentro del listado de trailers. A la derecha del componente de reproducción se ubica el catálogo de trailers de películas, el cual presenta el nombre de cada contenido, el género, el año y el número de veces que cada contenido ha sido reproducido. Finalmente en la parte inferior se observa un componente gráfico que ilustra las variaciones en el ancho de banda a lo largo del tiempo y a medida que los contenidos multimedia son reproducidos. El funcionamiento del servicio VoD DASH se detalla en el anexo H de la presente monografía.

ir_demo2/

Batman vs Superman _Dawn of Justice action URL:



puntuación: ★★★★★ *votar!*





371	
188	
97	
52	

Figura 3. 14. Servicio VoD DASH. Fuente: propia.

Capítulo 4

4. Dataset de contenidos de video

4.1 Introducción

Los sistemas de recomendaciones basados en contexto han sido difundidos con éxito en entornos de distribución de contenidos multimedia musicales. En dichos escenarios, un listado de sugerencias es presentado al usuario a partir del análisis y procesamiento de variables del contexto. Los contenidos multimedia recomendados en estos entornos, por lo general han sido previamente clasificados en emociones, teniendo en cuenta un conjunto de propiedades o características musicales. Lo anterior considerando que los contenidos musicales tienen la capacidad de incidir en el estado de ánimo de una persona una persona [58]. Esto se debe a que el contenido multimedia musical es transmitido a dos órganos sensitivos como son la vista y el oído, los cuales reciben de manera continua estímulos del medio externo que mantienen alerta al usuario ante cualquier situación.

La incidencia de la música en las emociones de una persona, tiene su explicación en la influencia de algunas características propias del contenido multimedia musical, como son: arousal, valence, ritmo, tempo, tono, modo, acústica entre otras. Estas características son importantes, puesto que permiten clasificar el contenido musical en un determinado estado de ánimo. La música por lo general está asociada con diferentes patrones de señales acústicas por ejemplo, mientras que el arousal se relaciona con el tempo (rápido/lento), la intensidad del tono (alta/baja), pitch (alto/bajo), loudness (alto/bajo), y el timbre (brillante/suave), la valencia está relacionado con el modo (mayor/menor) y armonía (consonante/disonante) [59].

De esta manera, para el diseño, experimentación e implementación de un sistema de recomendaciones de contenidos musicales basado en contexto, es necesario disponer de un dataset de contenidos multimedia de video que permita relacionar las emociones inferidas a partir de variables del contexto, con los contenidos

musicales. Un dataset es entendido como una representación de datos residentes en memoria, que proporcionan un modelo de programación relacional, coherente e independiente del origen de datos que contiene [60]. En este capítulo se presenta el proceso de conformación de un dataset de contenidos musicales de video, los cuales han sido clasificados de acuerdo al modelo emocional de arousal-valence.

El resto del capítulo está organizado de la siguiente forma: en la sección 2.2 se presenta el procedimiento usado para la generación del dataset de contenidos musicales de video. En la sección 4.3 se describen un conjunto de características musicales que son consideradas para la clasificación del contenido multimedia musical por emociones. En la sección 4.4 se presentan un conjunto de modelos de clasificación emocional de la música, los cuales fueron explorados en el presente trabajo. En la sección 4.5 se describen un conjunto de soluciones comerciales y/o tecnologías que se examinaron de cara a la conformación del dataset de contenidos de video musical. Finalmente en la sección 4.6 se describe de manera detallada el procedimiento usado para la conformación del dataset de contenidos multimedia, lo cual incluye la adaptación de un modelo de clasificación emocional y un diagrama de flujo del algoritmo usado para la conformación del dataset de contenidos musicales de video. En la sección 4.6 se describe la adaptación del modelo de clasificación emocional usado para la generación del dataset de contenidos musicales de video y el diagrama de flujo con la secuencia de pasos seguidos para la construcción del mismo. Finalmente en la sección 4.7 se presenta el prototipo de un servicio de VoD basado en emociones, desarrollado con el fin de verificar el funcionamiento del dataset de contenidos multimedia de videos musicales construido.

4.2 Procedimiento de generación del dataset

Para la generación del dataset de contenidos musicales de video, se definieron 5 fases a saber: estudio de características musicales y modelos de emociones, exploración de soluciones comerciales basadas en emociones, propuesta de un modelo basado en emociones, diseño y construcción del dataset de contenidos multimedia musicales, y evaluación del dataset de contenidos multimedia musicales de video (ver figura 4.1).

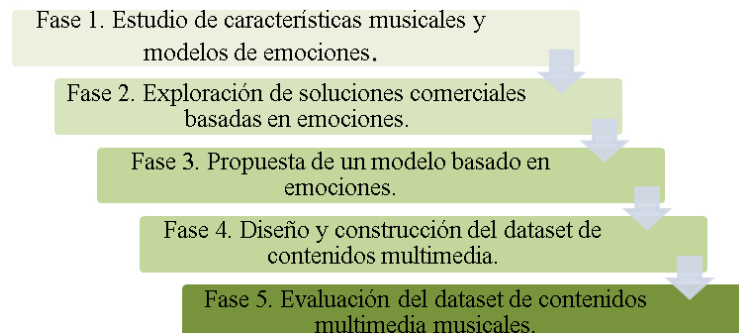


Figura 4. 1. Procedimiento de conformación del dataset. Fuente:propia.

En la primera fase se realizó el estudio de las principales características musicales y los principales modelos de emociones, con el fin de identificar claramente la relación entre los contenidos multimedia y las emociones. En la segunda fase se hizo una exploración por las herramientas, librerías y tecnologías para la clasificación de contenidos multimedia de acuerdo a las emociones, con el propósito de identificar posibles opciones a nivel de desarrollo para la construcción del dataset. En la fase tres se presenta la propuesta adaptada del modelo de clasificación de emociones, obtenida a partir de los modelos presentados en la fase uno. El modelo de clasificación propuesto fue utilizado para conformar el dataset a partir de las tecnologías identificadas en la fase dos. Por su parte, En la fase cuatro se realizó la construcción del dataset de contenidos multimedia, considerando el modelo de clasificación propuesto en la fase tres y las herramientas seleccionadas en la fase dos. Finalmente en la fase cinco, el dataset de contenidos multimedia es evaluado a través de la implementación de un servicio de VoD video bajo demanda, basado en emociones.

La fase uno de la metodología es abordada en las secciones de características musicales y modelos de emociones (secciones 4.3 y 4.4). La fase dos es considerada en la sección de soluciones comerciales (sección 4.5). La fase tres de la metodología se aborda en la sección de propuesta de modelo clasificación de emociones (sección 4.6.1). La fase cuatro es incluida en la sección de procedimiento de generación del dataset (sección 4.6.3). Finalmente, la fase cinco es abordada en la sección de evaluación del dataset de contenidos multimedia (sección 4.7).

4.3 Características musicales

Existen diferentes características musicales que permiten asociar un contenido multimedia, con el estado de ánimo que presenta una persona en determinado momento. Así mismo, dichas características pueden ser usadas para inferir la emoción que puede percibir una persona al escuchar ese contenido musical. Así,

estas características musicales son fundamentales puesto que ayudan a categorizar el contenido musical de acuerdo a un determinado estado de ánimo. Dentro de estas propiedades musicales se destacan el arousal (energía) y la valence (valencia), ya que son consideradas en la mayoría de modelos de clasificación musical.

El arousal determina el estado de excitación o relajación de una persona y está relacionado directamente con el tempo (rápido/lento), la intensidad del tono (alta/baja), *pitch* (alto/bajo), *loudness* (alto/bajo), y el timbre (brillante/suave). En lo que respecta a la valencia, esta determina la positividad o negatividad musical de un contenido multimedia y está relacionada con el modo (mayor/menor) y armonía (consonante/disonante) [59]. En este apartado se describen un conjunto de características o propiedades musicales, que pueden ser usadas para determinar el estado de ánimo de una persona. Dentro de estas variables se encuentran: valence, arousal, ritmo, tempo, speechiness, liveness, acousticness, danceability y modo.

4.3.1 Arousal (Energía)

Esta propiedad musical, representa una medida de percepción de la intensidad y la actividad a lo largo de una pista musical. Típicamente las pistas rápidas que cuentan con sonidos fuertes y ruido como por ejemplo el rock pesado, tienen asociado un valor de energía alto, mientras que una pista de música clásica como: Air de Beethoven, tendría asociado un valor de energía bajo. Otras características que contribuyen en este atributo son el rango dinámico, la percepción del volumen, el timbre y la entropía general [61].

4.3.2 Valence (Valencia)

Esta característica describe la positividad musical transmitida por una pista de audio. Las pistas con alta valencia están asociadas con emociones positivas tales como: estar feliz, alegre eufórico, entre otros. Por otra parte, las pistas de con baja valencia están asociadas a emociones negativas o estados de ánimo tales como: tristeza, depresión, enojo [61].

4.3.3 Ritmo

El ritmo es el patrón de pulsos o notas de fuerza variable, que se describe a menudo en términos del tempo, métrica o fraseo (ver figura 1). Una canción con un ritmo rápido a menudo se percibe como un arousal alto, además un ritmo fluido se

asocia generalmente con una valencia positiva, mientras que un ritmo firme está asociado con una valencia negativa [59].

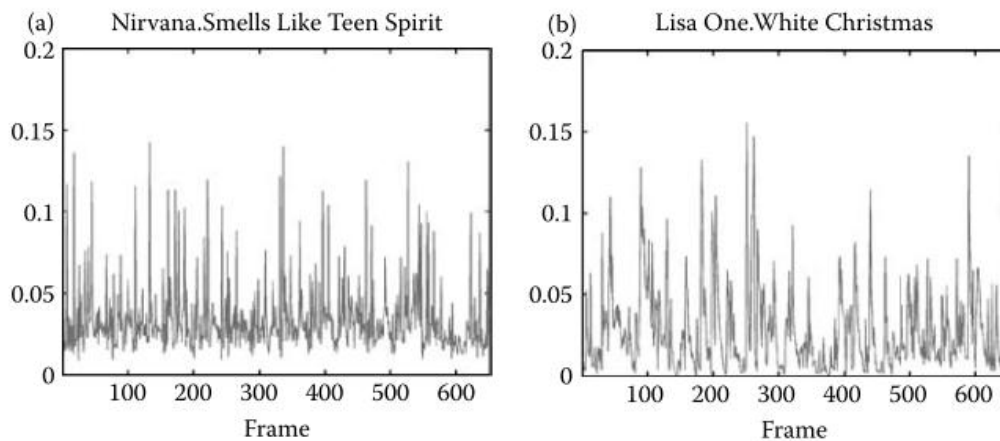


Figura 4. 2. Ejemplo patencia de audio. Fuente: tomado de [59].

En la figura 4.2, se puede observar un ejemplo del espectro de dos canciones, donde en (a) se aprecia que la variación es mucho más rápida que en (b). De acuerdo a lo anterior y teniendo en cuenta la definición dada de ritmo, se puede inferir que la primera tendría un mayor valor de arousal.

4.3.4 Tempo

Esta propiedad sirve en una pieza musical para transmitir emociones, de tal modo que la música rápida según estudios es percibida o relacionada con emociones activas (felicidad), mientras que la música lenta tiende a percibirse como una emoción pasiva (tristeza). El tempo varía usualmente entre 20ppm y 240ppm, aunque puede tomar valores menores o mayores a estos [61].

4.3.5 Speechiness

Esta es una característica que permite detectar la presencia de las palabras habladas en una pista. Los valores que son superiores a 0.66 describen pistas que probablemente están hechas totalmente de palabras. Por su parte, valores comprendidos entre 0.33 y 0.66 describen pistas que pueden contener tanto música como interpretación de palabras. Finalmente, los valores por debajo de 0.33 probablemente puedan ser solo música [61].

4.3.6 Liveness

Esta propiedad musical permite detectar la presencia o no de audiencia en la grabación. Así, valores cercanos a 1.0 sugieren que la probabilidad de que la pista sea en vivo es alta. Por su parte, valores entre 0.6 y 0.8 describen pistas que pueden o no estar en vivo o contener audiencia simulada, por lo general está simulación es colocada al inicio o al final de la pista musical. Por último, los valores por debajo de 0.6 son grabaciones hechas en un estudio [61].

4.3.7 Acousticness

Permite medir la probabilidad de que una grabación se haya creado únicamente con elementos tales como la voz e instrumentos acústicos, en lugar de utilizar elementos electrónicos. Las pistas con valores bajos suelen incluir guitarras eléctricas, distorsión, sintetizadores, entre otros. Así mismo, valores cercanos a 1 indican que la canción presenta esta característica [61].

4.3.8 Danceability

Esta característica permite conocer que contenido es el más adecuado para el baile. Las pistas con valores cercanos a 1, son los que mayormente reflejan esta característica en todo su desarrollo. Según [61], algunos elementos que permiten caracterizar esta propiedad son el tempo, la estabilidad del ritmo, la regularidad que presente la pista, entre otros.

4.3.9 Modo

Es la propiedad que indica la modalidad (mayor o menor) de una pista, el tipo de escala de la que deriva su contenido melódico [61].

4.4 Emoción

En el campo de la emoción, ha sido amplia la variedad de definiciones propuestas, aunque ninguna de estas ha sido absolutamente aceptada por todos los investigadores, una definición común es la que se plantea a continuación. “La emoción es un conjunto complejo de interacciones entre los factores subjetivos y objetivos, mediadas por sistemas neurales-hormonales, que pueden dar lugar a

experiencias afectivas, tales como sentimientos de excitación o *arousal*, placer/displacer y que conducen a un comportamiento que es a menudo, pero no siempre, expresivo, dirigidos hacia una meta, y adaptativos” [62].

4.4.1 Estructura de las emociones

Las emociones pueden ser reducidas a un núcleo afectivo específico placer o displacer. Así mismo, otros estudios sugieren el uso de modelos en dos dimensiones, tales como el modelo de *arousal-valence*, donde el *arousal* puede ser positivo o negativo y la valencia puede ser positiva o negativa pudiendo caracterizar cualquier emoción por sus coordenadas en un espacio de dos dimensiones (ver figura 4.3). Como ejemplo, la felicidad usualmente tiene valencia positiva y *arousal* moderadamente positivo, en cambio la tristeza tiene un *arousal* moderadamente negativo y valencia negativa [63].

Figura 4.3(a): Modelo de dos dimensiones

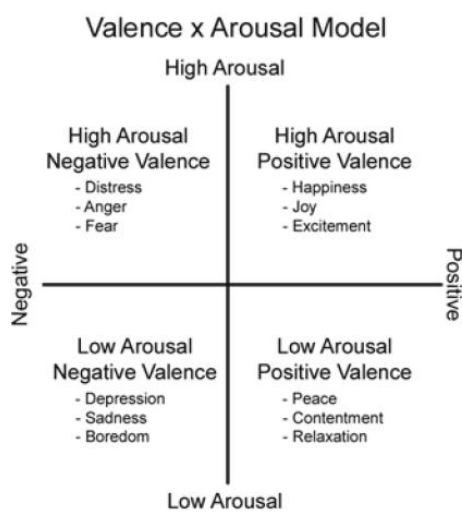


Figura 4.3(b): Modelo mixto de valence

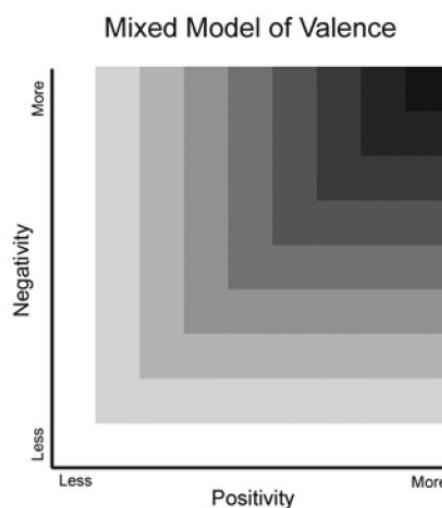


Figura 4. 3. Modelos de emociones. Fuente: tomado de [63].

En la figura 4.3(a) se puede apreciar el diagrama del modelo afectivo de dos dimensiones (*arousal-valence*), donde se muestran ejemplos de algunas emociones ubicadas en el plano cartesiano según su nivel de positividad (valencia) y *arousal*. Así mismo, en la figura 4.3(b) se aprecia el modelo de valencia mixta, en el cual las partes más oscuras representan una gran mezcla de emociones, producto de que se comparten activación tanto positiva como negativa variando en grados. A continuación, se describen algunos de los modelos de emociones más difundidos, dentro de los que se encuentran: modelo circunflejo de Russell, modelo de Hevner, modelo de Plutchick.

4.4.2 Modelo circunflejo

El modelo circunflejo o de Russell (ver figura 4.4), es tal vez uno de los modelos más investigados. Este modelo está formado por una estructura circular de dos dimensiones (valencia/activación), que parte el espacio cartesiano en cuatro cuadrantes, en el cual las emociones son trazadas basándose en su nivel de actividad (activo/pasivo) y su valencia (positiva/negativa). El modelo circunflejo del afecto, muestra que los estados afectivos surgen de interpretaciones cognitivas de sensaciones nerviosas centrales, que son el producto de dos sistemas neurofisiológicos independientes, dichos sistemas uno está relacionado con la valencia (placer/desagrado) y el otro al *arousal* (estado de alerta). Cada emoción puede ser vista como una combinación lineal de estas dos dimensiones o diversos grados de valencia [64].

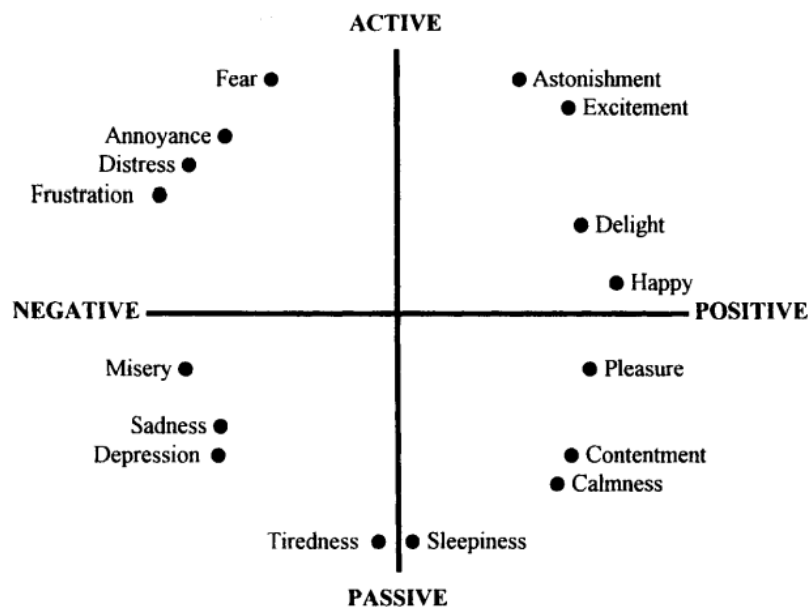


Figura 4. 4. Modelo circunflejo de las emociones adaptado por Russell. Fuente: tomado de [64].

Este modelo tiene una estructura clara con implicaciones en la forma en que las emociones son experimentadas en términos de la evaluación cognitiva, es decir por la manera en la que estas son clasificadas según los ejes activo-pasivo y positivo-negativo del modelo. Lo anterior a partir del análisis de las variables musicales de *arousal* y *valence*.

4.4.3 Modelo de Hevner

Este modelo se desarrolló con base a una experimentación en donde se manipularon características musicales como la melodía, armonía, modo, ritmo y tempo, a piezas musicales. El experimento consistió en darle a escuchar a un grupo

de personas, la versión original y una versión modificada, en la cual se alteró una sola característica. Después de escuchar las dos versiones, se les preguntaba a las personas acerca de la emoción que describía mejor la pieza escuchada, y se les pedían señalar dicha emoción en el modelo. Lo anterior permitió realimentar el modelo inicial con diferentes emociones, tal como se muestra en la figura 4.5 [65].

Según el estudio realizado se concluyó que utilizando una determinada característica musical e identificando el estado de ánimo de una persona, es posible modificar su estado emocional de forma gradual, siendo el tempo y el modo las variables que más impacto generaron en el oyente.

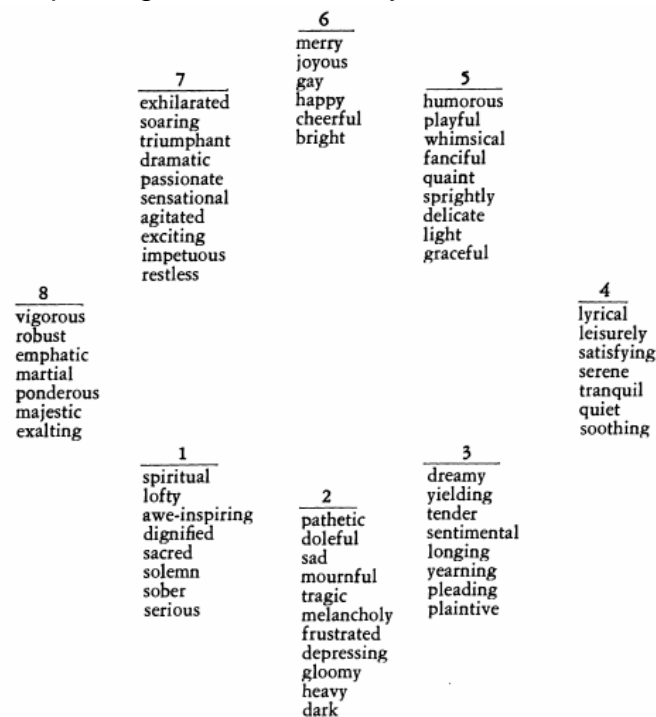


Figura 4. 5. Modelo de las emociones de Hevner. Fuente: tomado de [65].

4.4.4 Modelo tridimensional

Este modelo agrega la variable de intensidad emocional, a las dimensiones de la valencia y actividad (equivalente al *arousal* para este modelo). Dicha variable se refiere a que tan fuerte o débil puede ser una emoción. Esta no se debe confundir con actividad, ya que esta significa la desviación del estado fisiológico normal de una persona, siendo las disminuciones en este estado, representaciones de baja actividad y los aumentos una alta actividad. Así, la variable de intensidad está relacionada con el grado en que la experiencia emocional produce un cambio de un estado neutral a uno no emocional.

4.4.5 Modelo de plutchick

Es un modelo tridimensional en el cual se postulan 8 emociones primarias (tristeza, sorpresa, miedo, ira, entre otras) de las cuales se derivan las demás. Este modelo propone que las emociones varían en polaridad, intensidad y grado de similitud, de tal modo que cada una de ellas se intensifica a medida que se mueve desde el exterior hacia el centro de la circunferencia. A modo de ejemplo, una emoción como el aburrimiento, puede ser intensificada en odio si no se controla. Además, cada sector del círculo tiene su emoción opuesta correspondiente, lo contrario de la alegría y al mismo tiempo opuesto a su posición en la figura se encuentra la tristeza (ver figura 4.6). Las emociones con ningún color representan un estado de ánimo que es una mezcla de dos estados emocionales primarios, por ejemplo la anticipación y la alegría se combinan para ser el optimismo [66].

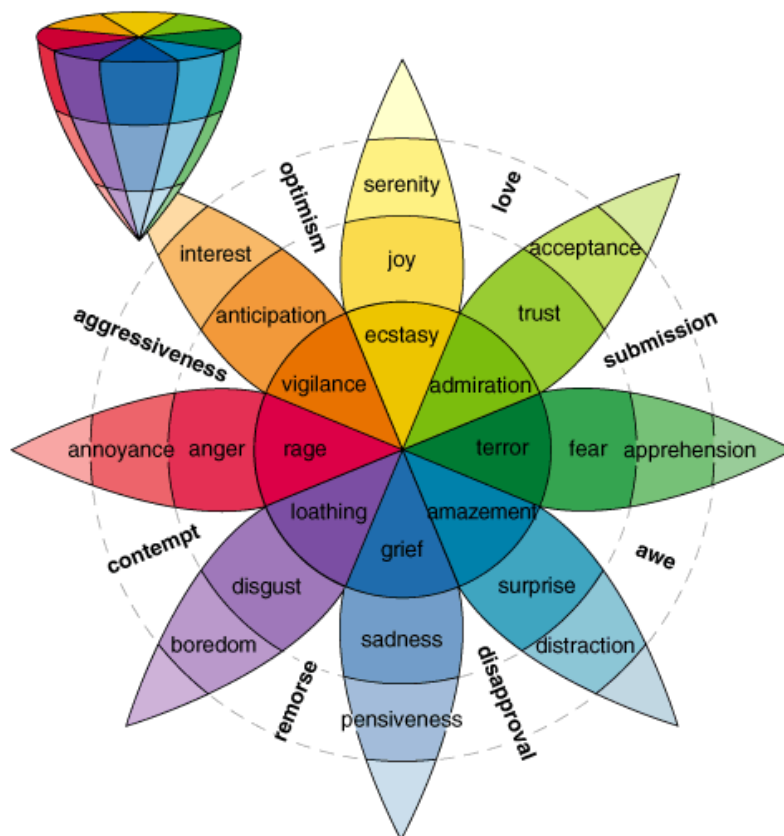


Figura 4. 6. Modelo tridimensional de plutchick. Fuente: tomado de [66].

En este trabajo, se decidió escoger el modelo circunflejo de dos dimensiones planteado por Russell, debido a su sencillez, amplio estudio y aceptación por parte de diferentes expertos del área. Así la propuesta del modelo de clasificación usada en este trabajo parte de este modelo y es presentada de manera detallada en la sección 4.6.1.

4.5 Soluciones comerciales

A continuación, se presentan un conjunto de soluciones comerciales y/o librerías de desarrollo que vinculan las emociones a partir del análisis de las características del contenido musical. Los datos proporcionados por estas soluciones pueden ser usados como en el presente trabajo para la generación de recomendaciones de contenidos multimedia. Dentro de las soluciones exploradas se encuentran: Musicoverly, Gracenote y EchoNest.

4.5.1 Musicoverly

La API de Musicoverly¹⁴ proporciona datos para generar recomendaciones musicales y listas de reproducción de todo tipo: desde un estado de ánimo, un artista, una pista, un género / estilo, un tema, un período / año (ver figura 4.7). La respuesta, una lista de pistas / artistas, se puede filtrar y personalizar de acuerdo a varios factores como son: la popularidad, país oyente, el tipo de similitud.

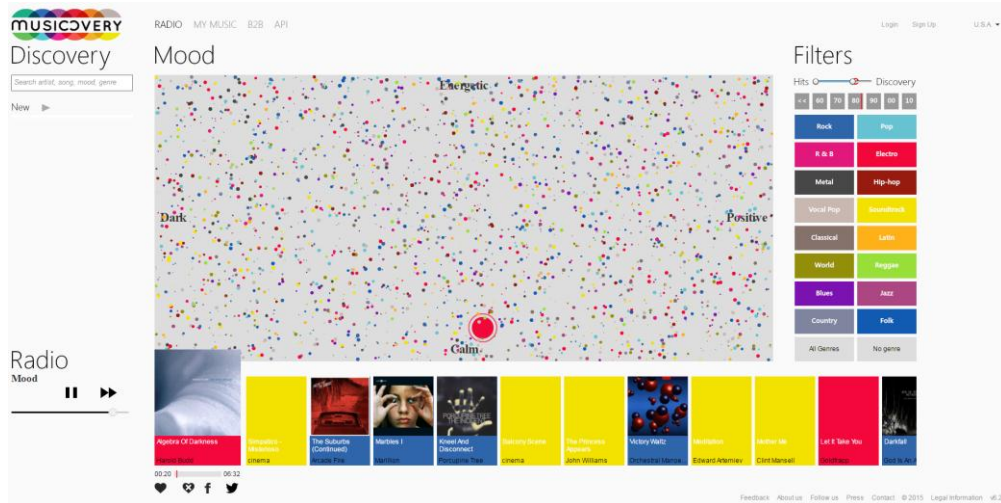


Figura 4. 7. Interfaz de usuario Musicoverly. Fuente: tomado de [67].

Es posible acceder libremente a la API Musicoverly, con la restricción de solo poder realizar 200 consultas en total. De esta manera, es necesario solicitar una clave de API, permitiendo extender a 5000 las consultas. Para consultar sin limitación Musicoverly API, necesita una clave de API Premium. La API tiene soporte para realizar los desarrollos en lenguaje PHP y el formato de respuesta es de tipo JSON (*JavaScript Object Notation*) o XML (*eXtensible Markup Language*) [67].

¹⁴ Proporciona datos para la recomendación de música y generación de listas de reproducción. <http://musicoverly.com/>.

El servicio provisto por musicoverly puede ser accedido desde la web y cuenta con una interfaz gráfica que presenta un espacio cartesiano similar al del modelo arousal-valence. En esta interfaz basta con presionar cualquier de los puntos del espacio cartesiano para reproducir el contenido musical. Así mismo, este servicio permite categorizar y realizar búsquedas por diferentes géneros (ver figura 4.7).

4.5.2 Gracenote

Esta API ofrece un conjunto de metadatos de música a través del protocolo de internet HTTP (*Hypertext Transfer Protocol*), permitiendo categorizar los contenidos por género, región de origen y estado de ánimo (ver figura 4.8). Además, permite la posibilidad de realizar búsquedas de portadas de disco, imágenes, biografías, artículos y otros contenidos relacionados. Para hacer uso de esta API es necesario obtener un identificador de cliente y una clave de API, mediante las cuales es posible realizar un conjunto limitado de peticiones al servicio de gracenote. Este API puede usarse para el desarrollo en entornos móviles y aplicaciones web, permitiendo el desarrollo en lenguajes de programación como: Python, PHP, Ruby, Java. Finalmente, formato de respuesta que maneja esta API es XML o JSON, lo cual lo hace extensible para su uso en diferentes entornos tecnológicos [68].

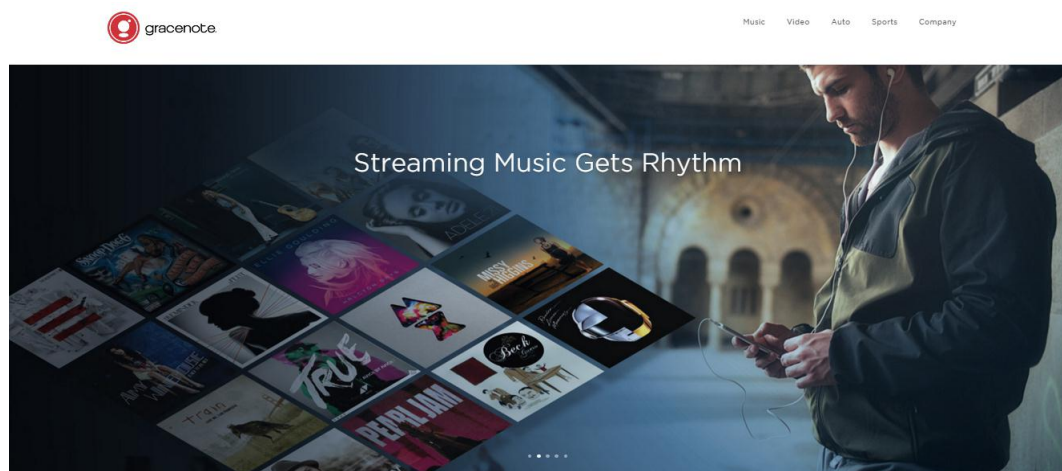


Figura 4. 8. Página web oficial de gracenote. Fuente: tomado de [68].

4.5.3 EchoNest

Es una herramienta desarrollada en los laboratorios del MIT por Tristán Jehan y Brian Whitman de la cual es propietario *Spotify*. Esta API permite obtener características del contenido tales como: la valencia, la energía, tempo, duración,

popularidad, entre otros (ver figura 4.9). Para hacer uso de la API de EchoNest¹⁵, es necesario obtener una llave (clave) de la API, la cual da acceso o permite hacer 20 peticiones por minuto de manera ilimitada.

Esta herramienta además proporciona los mensajes de respuestas en formato JSON o XML. Entre los lenguajes a los que da soporte se encuentran: Python, Java, Ruby, PHP, Objective-C/iOS, C++, Javascript, para los cuales existen librerías que permiten implementar las funcionalidades básicas que provee EchoNest. Algunas de las librerías oficiales son: pyechonest la cual es la biblioteca para Python, Jen para clientes Java, Enios para Objective-C. Así mismo, dentro de las librerías no oficiales se encuentran: PHP echonest, Ruby echonest, node echonest para Javascript, entre otros [61].

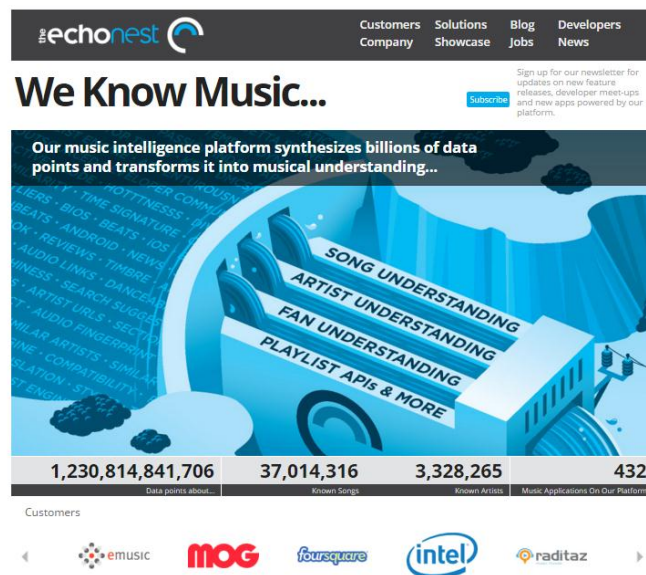


Figura 4. 9. Página web oficial de EchoNest. Fuente: tomado de [61].

4.6 Conformación dataset de video

Para la conformación del dataset de videos musicales, se eligió trabajar con la API de EchoNest, puesto que a pesar de estar limitado a 20 peticiones por minuto, estas consultas pueden realizarse de manera ilimitada a diferencia de las librerías de gracernote y musicoverly. Además de lo anterior, Echonest permite soporte para el desarrollo con lenguajes como Python y Java, los cuales fueron usados para la construcción de distintos prototipos software realizados en el presente trabajo. Para el desarrollo con Python se hizo uso de la librería pyechonest, mientras que en el caso de Java se utilizó la librería Jen.

¹⁵ EchoNest es una plataforma inteligente de música, encargada del análisis de datos musicales. Página web oficial de EchoNest: <http://the.echonest.com>.

4.6.1 Modelo de emociones

El dataset de contenidos de video musical fue conformado teniendo en cuenta las 5 emociones consideradas por la API de EchoNest, y tomando como base el modelo de emociones de dos dimensiones (*arousal-valence*). Los estados de ánimo considerados en la representación del modelo son: excitado, feliz, relajado, triste y enojado. Además, se eligió trabajar sobre el modelo de *arousal-valence*, debido a su simplicidad en el manejo de dos coordenadas (arousal, valence), amplia difusión y facilidad de adaptación a nuevos diseños. Así, el modelo planteado es una adaptación del modelo *arousal-valence*, el cual también es considerado en otros e investigaciones como las presentadas en [59] [63] [69] [70]. Dicho modelo se puede apreciar en la figura 4.10.

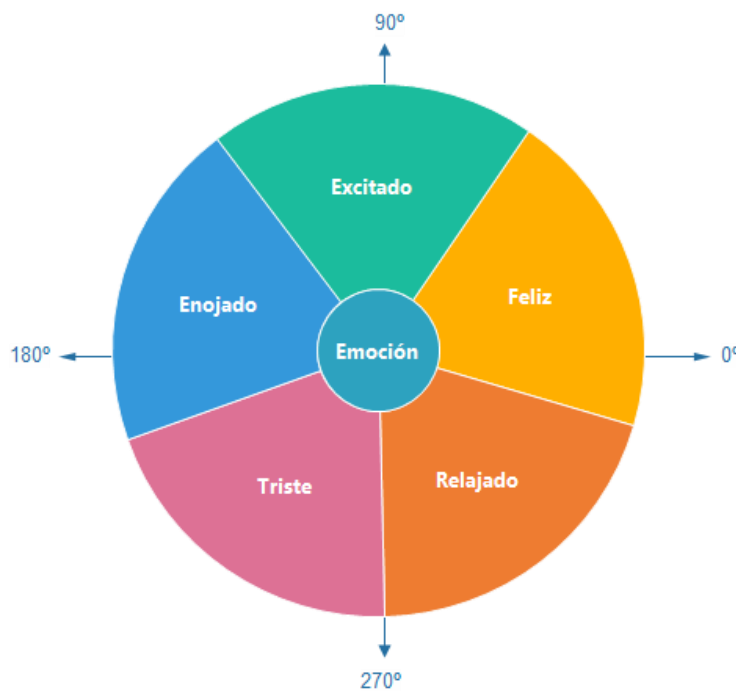


Figura 4. 10. Modelo de *arousal-valence* adaptado a 5 emociones. Fuente: propia.

Cada una de las emociones descritas en la figura 2.9 tiene una amplitud de 72° y un rango específico de acción en el plano de las emociones. La distribución de las emociones y el rango asociado a cada una son presentadas en la tabla 4.1.

Tabla 4. 1. Tabla de estados de emoción, según el rango de ángulos. Fuente: propia.

RANGO DE ANGULOS	EMOCION ASOCIADA
$<54^\circ$ y $\geq 342^\circ$	Feliz
$\geq 54^\circ$ y $<126^\circ$	Excitado
$\geq 126^\circ$ y $<198^\circ$	Enojado
$\geq 198^\circ$ y $<270^\circ$	Triste
$\geq 270^\circ$ y $<342^\circ$	Relajado

4.6.2 Reconocimiento de emociones del contenido

Para verificar el funcionamiento del modelo de clasificación de la sección 4.6.1 y explorar la API de Echonest, se desarrolló una herramienta en lenguaje Java para el reconocimiento del estado de ánimo asociado a un contenido multimedia musical. En la aplicación desarrollada se pueden observar propiedades musicales tales como: la energía (energy), la valencia (valence) , el tempo, el ángulo formado por la valencia y la energía en el plano cartesiano, la emoción asociada a esos valores de energía y valencia, entre otros parámetros los cuales se pueden apreciar en la figura 4.11.

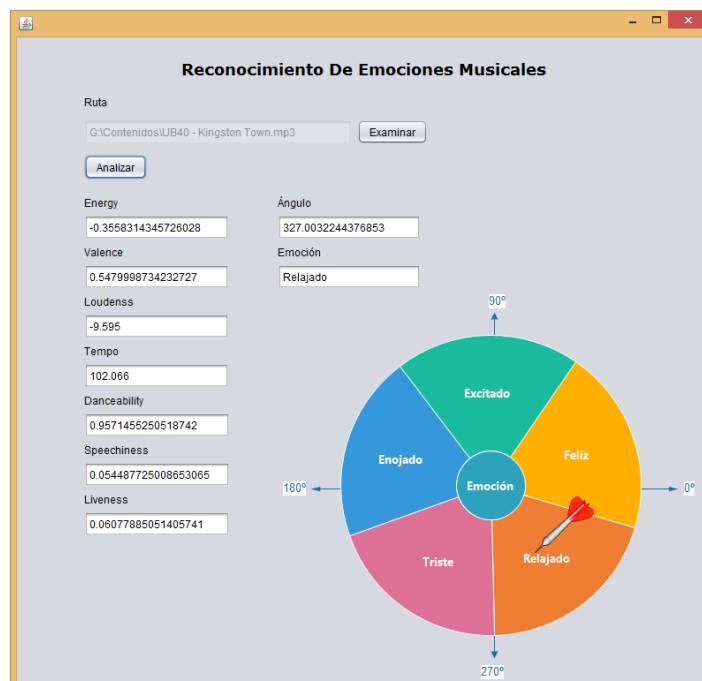


Figura 4. 11. Reconocimiento de emociones musicales. Fuente:propia.

A modo de ejemplo, en la figura 4.11 se observa como usando la librería Jen de Java (provista por Echonest), se analizaron las características musicales para la canción del grupo pop-reggae UB40 llamada “Kingston Town”. Los parámetros que permiten conocer a que emoción está asociada la canción son el arousal y el valence. En cuanto al arousal este tiene un valor de -0.3558, lo que permite ubicarlo en la parte inferior o negativa del eje vertical o eje del arousal. En lo que respecta a la valence, esta tiene un valor de 0.547, lo cual la ubica en la mitad positiva del eje horizontal o eje de la valencia. De esta manera, a partir de los anteriores valores, es posible calcular trigonométricamente el ángulo formado entre el valor de la propiedad arousal (eje y) y la propiedad valence (eje x). A partir del ángulo obtenido con la anterior operación, es posible determinar la emoción asociada al contenido multimedia, teniendo en cuenta la información de la tabla 4.1. El anterior proceso puede evidenciarse en el diagrama de bloques de la herramienta de clasificación Java de clasificación (ver figura 4.12).

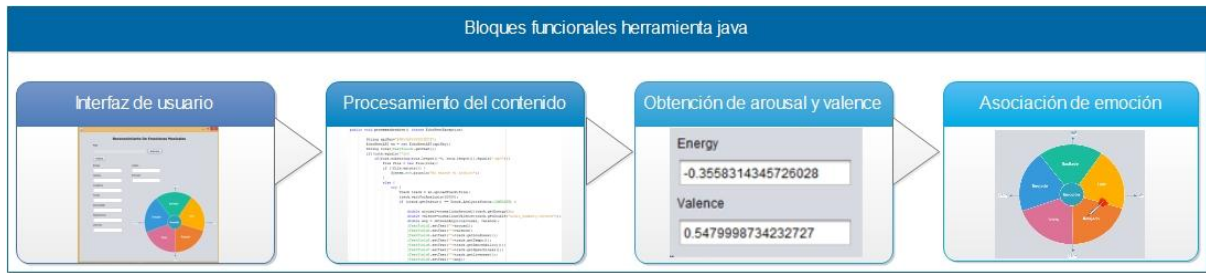


Figura 4. 12. Bloque funcionales herramienta Java. Fuente: propia.

4.6.3 Diagrama de flujo del dataset de contenidos

En la figura 4.13 se presentan cada una de las diferentes etapas del proceso de conformación del dataset de contenidos de video, ilustradas mediante un diagrama de flujo. Así mismo dentro de estas etapas se describen las diferentes herramientas y librerías usadas en el proceso de generación del dataset. Estas etapas son descritas de manera más específica en el anexo A de la presente monografía.

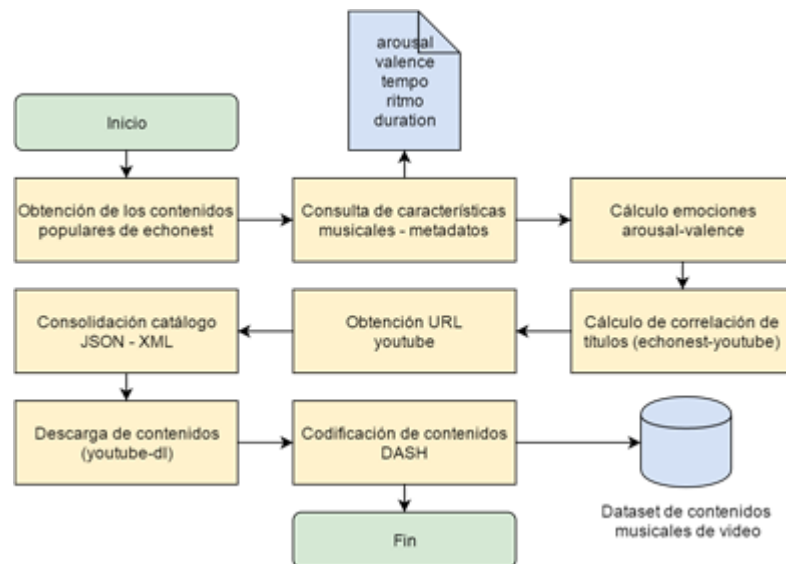


Figura 4. 13. Diagrama de flujo del dataset de contenidos multimedia. Fuente: propia.

Para la generación del dataset de contenidos multimedia de video, se hizo uso de la librería pyechonest de Python, la cual requiere contar con una llave de acceso para el desarrollo con la API, que se obtiene al registrarse en la página web oficial de Echonest. Además de lo anterior, se utilizaron varias librerías adicionales de Python que fueron necesarias para la conformación del dataset, tales como: math, youtube_dl, urllib, cookielib, sys, time, entre otros (ver figura 4.14).

```

#***** Librerías y Llaves *****#
import math,sys,youtube_dl,time,cookielib,urllib,urllib2,re,logging
from difflib import SequenceMatcher
from tinydb import TinyDB, where
from pyechonest import config
from pyechonest import artist
from pyechonest import track
from pyechonest import song

config.ECHO_NEST_API_KEY="EPX09QF26IUQIBUCP"
#*****#

```

Figura 4. 14. Librerías utilizadas y llave de EchoNest. Fuente: propia.

A continuación, se describen cada uno de los pasos del proceso de conformación o generación del dataset de contenidos musicales de video.

En el paso “1” a través de la API de Echonest, se procede a generar un listado con las canciones más populares del catálogo de contenidos de Echonest. Para lo anterior, se genera un *script* encargado de realizar 18 peticiones cada 60 segundos, dadas las restricciones de acceso provistas por la API. En la figura 4.15, se puede apreciar el código utilizado para realizar las búsquedas de las canciones más populares.

```

def generar_script(int):
    while True:
        res = song.search(sort='song_hottnesss-desc', buckets=['audio_summary'], results=100, start=int)

        tam=len(res)
        tiempos=[18,36,54,72,90]
        lista_repetidos=list()

        for i in range(tam):
            if(i in tiempos):
                time.sleep(60)

            cancion=res[i]
            cur=cancion.song_currency

```

Figura 4. 15. Búsqueda de canciones más populares EchoNest. Fuente: propia.

En el paso “2” se obtiene con ayuda de la API de Echonest un conjunto de características asociadas al contenido multimedia musical, dentro de las que se encuentran: title, artist, energy, valence, tempo, entre otras. En la figura 4.16 se visualizan cada una de las características mencionadas, las cuales son pieza clave para para la asociación de las emociones a cada video musical. Además de lo anterior, en este paso se realiza la discriminación de las canciones que puedan resultar repetidas en el proceso de búsqueda.

```

if(cur not in lista_repetidos):
    title=cancion.title
    artist=cancion.artist_name
    energy=cancion.audio_summary['energy']
    valence=cancion.audio_summary['valence']
    liveness=cancion.audio_summary['liveness']
    tempo=cancion.audio_summary['tempo']
    speechiness=cancion.audio_summary['speechiness']
    acousticness=cancion.audio_summary['acousticness']
    instrumentalness=cancion.audio_summary['instrumentalness']
    mode=cancion.audio_summary['mode']
    time_signature=cancion.audio_summary['time_signature']
    duration=cancion.audio_summary['duration']
    loudness=cancion.audio_summary['loudness']
    danceability=cancion.audio_summary['danceability']
    url=''
    visua=''

```

Figura 4. 16. Obtención de las características del contenido. Fuente: propia.

Una vez obtenidas las características de los contenidos multimedia musicales, se procede al paso “3” donde se hace la asociación de la emoción del contenido, haciendo uso de los parámetros de *arousal* y *valence*, los cuales permiten hallar el ángulo para su ubicación en el modelo de las 5 emociones. En la figura 4.17 se puede observar este proceso.

En el paso “3” se realiza el cálculo de las emociones asociadas al contenido multimedia, usando para ello los valores de los parámetros de arousal y valence, los cuales permiten obtener el ángulo trigonométrico sobre el modelo de 5 emociones presentado en la sección 6.1 (ver figura 4.17). Dado que los valores consultados a partir de la API de Echonest con respecto a las propiedades de arousal y valence están comprendidos entre 0 y 1, estos deben ser normalizados al rango -1 y 1 usando para ello las ecuaciones 1 y 2. De esta manera es posible ubicar el contenido musical de video sobre el modelo de 5 emociones de la sección 4.6.1.

$$arousal_normalizado = 2(arousal - 0.5) \quad (1)$$

$$valence_normalizado = 2(valence - 0.5) \quad (2)$$

Ecuación 4. 1. Fórmulas de normalización para arousal y valence.

Donde:

arousal: Es el valor de la propiedad arousal obtenido a partir de la API de Echonest.

valence: Corresponde al valor de la propiedad valence obtenida a partir de la API de Echonest.


```
e_n=2*(energy-0.5)
v_n=2*(valence-0.5)
ang=math.degrees(math.atan2(e_n,v_n))
if ang<0:
    ang=ang+360
else:
    ang=ang

# Estados de animo : 5 estados
if (ang>0 and ang<=54) or (ang>342):
    ani2='Happy'
elif ang>54 and ang<=126:
    ani2='Excited'
elif ang>126 and ang<=198:
    ani2='Angry'
elif ang>198 and ang<=270:
    ani2='Sad'
elif ang>270 and ang<=342:
    ani2='Relaxing'
```

Figura 4. 17. Obtención de la emoción del contenido. Fuente: propia.

En el paso 4 se comparan los contenidos multimedia obtenidos mediante la API de Echonest con los contenidos de video consultados a través de la API de youtube, con el fin de determinar el video asociado al contenido multimedia musical. De este modo, haciendo uso de la API de youtube se realiza la búsqueda dentro de la categoría música de cada uno de los títulos de los contenidos musicales consultados en Echonest. A partir de los resultados de la búsqueda, se filtran los videos más visitados de youtube y que presentan una correlación más alta en el texto de sus títulos, considerando para ello el método SequenceMatcher de la librería difflib de Python. Este método retorna un valor entre 0 y 1, el cual corresponde al porcentaje de correlación entre los títulos del contenido musical y el contenido de video (ver figura 4.18).

Así, en el paso 5 a partir de los videos con mayor correlación (superior a 0.6), se obtiene la URL del contenido de video, usando para tal propósito la librería youtube-dl, la cual permite consultar para cada contenido multimedia propiedades como: calidad, titulo, formato, categorías, fechas, entre otras. El hecho de considerar los contenidos más visitados de youtube, se justifica en que por lo general estos corresponden a los videos oficiales de las canciones, razón por la cual presentan una mejor calidad que el resto de los videos.

```

def obtener_url(artist,title):
    hdr = {'User-Agent': 'Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.11 (KHTML, like Gecko) Chrome/23.0.1271.64 Safari/537.11',
    consulta=artist+' - '+title+' Official Video'
    print consulta
    query_string = urllib.urlencode({'search_query' : consulta})
    html_content = urllib2.Request('https://www.youtube.com/results?' + query_string, headers=hdr)
    try:
        response = urllib2.urlopen(html_content)
        search_results = re.findall(r'href="\./watch?v=.{11}"', response.read())
    except urllib2.HTTPError, e:
        print e.fd.read()
    lista=List()
    visitas=List()
    datos = []
    url_elegida=''
    for i in range(0,len(search_results)):
        url='http://www.youtube.com/watch?v='+search_results[i]
        logging.getLogger().setLevel(logging.ERROR)
        if(url not in lista):
            try:
                video=obtener_info_video(url)
                time.sleep(2)
                titulo=video['title']
                visual=video['view_count']
                catego=video['categories']
                #print titulo,visual,catego
            except (EOFError, IOError, AttributeError, TypeError, ValueError, UnicodeError, UnicodeDecodeError, UnicodeEncodeError):
                proba=similar(consulta,titulo)
                print proba
                lista.append(url)
                if catego[0].encode('utf-8') == 'Music':
                    if proba >= 0.6:
                        visitas.append(visual)
                        datos.append({'visualizaciones':visual,'url':url})
                        print 'POSIBLE URL:',url
                        logging.getLogger().setLevel(logging.ERROR)
    maxvis=video_mvisitado(visitas)
    for i in range(0,len(datos)):
        if(maxvis == datos[i]['visualizaciones']):
            url_elegida=datos[i]['url']
    print "*****"
    print "URL ELEGIDA: ",url_elegida
    print "*****"
    datos=[url_elegida,maxvis]
    return datos

```

Figura 4. 18. Método para la obtención de url asociada a cada contenido. Fuente: propia.

En el paso 6 se consolidan como entradas de un documento JSON y/o XML la información de los contenidos musicales, lo cual incluye las propiedades musicales de los contenidos de video musical, la emoción obtenida a partir de las características musicales y la URL de youtube asociada a cada uno de los contenidos. En la figura 4.19 se presentan a modo de ejemplo varias de las entradas del catálogo JSON consolidado con los contenidos de video musicales. Cada entrada del documento JSON de la figura 4.19 tiene asociada un conjunto de características musicales que son representadas en diferentes colores, dentro de las que se destacan los parámetros de arousal y el valence, los cuales son usados para determinar la emoción a la que está asociada un contenido musical.

```

{"_default": {"1": {"liveness": 0.094833, "energy": 0.717726, "tempo": 129.882, "speechiness": 0.106457, "currency": 0.22104064240294416, "instrumentalness": 0.338528, "duration": 313.94844, "view_count": 822693136, "mood": "Excited", "artist": "Major Lazer", "url": "http://www.youtube.com/watch?v=YqeW9_5kURI", "title": "Lean On", "acousticness": 0.033117, "danceability": 0.862321, "time_signature": 4, "loudness": -5.691, "valence": 0.421374, "mode": 0}, "2": {"liveness": 0.064907, "energy": 0.480501, "tempo": 80.025, "speechiness": 0.081512, "currency": 0.16323143005491134, "instrumentalness": 1e-06, "duration": 229.52576, "view_count": 1187333053, "mood": "Angry", "artist": "Wiz Khalifa", "url": "http://www.youtube.com/watch?v=RgKAFK5dJsk", "title": "See You Again (feat. Charlie Puth)", "acousticness": 0.369444, "danceability": 0.689399, "time_signature": 4, "loudness": -7.503, "valence": 0.26631, "mode": 1}, "3": {"liveness": 0.515838, "energy": 0.873779, "tempo": 168.901, "speechiness": 0.191692, "currency": 0.0005813953488372093, "instrumentalness": 0.041976, "duration": 233.10512, "view_count": 782099, "mood": "Excited", "artist": "Rage Against The Machine", "url": "http://www.youtube.com/watch?v=3L4YrGaR8E4", "title": "Bulls on Parade", "acousticness": 0.0250803, "danceability": 0.398765, "time_signature": 4, "loudness": -10.218, "valence": 0.510731, "mode": 1}

```

Figura 4. 19. Muestra del archivo JSON generado. Fuente: propia.

El archivo JSON que contiene el catálogo de contenidos multimedia de video musical, fue construido por medio del gestor de bases de datos no-sql “tinydb”, el cual está diseñado para ser simple de usar, no requiere la instalación de un servidor y es compatible con las diferentes versiones del lenguaje Python. El proceso de construcción del documento JSON se ilustra en la figura 4.20.

```

##### Metodos TinyDB #####
def agregar_cancion(animo,title,artist,energy,valence,liveness,tempo,speechiness,acousticness,
instrumentalness,mode,time_signature,duration,Loudness,danceability,cur,url,visua):

    db = TinyDB('contents.json')
    lista=db.search((where('title')==title) & (where('artist')==artist))
    if(len(lista)==0):
        db.insert({'mood':animo,'title':title,'artist':artist,'energy':energy,'valence':valence,
'liveness':liveness,'tempo':tempo,'speechiness':speechiness,'acousticness':acousticness,
'instrumentalness':instrumentalness,'mode':mode,'time_signature':time_signature,
'duration':duration,'loudness':loudness,'danceability':danceability,'currency':cur,'url':url,'view_count':visua})
    print '*****'
    print 'CONTENIDO ALMACENADO'
    print '*****'

def actualizar_url_view(artista,titulo,url,visi):
    db = TinyDB('contents.json')
    db.update({'url':url},(where('artist')==artista) & (where('title')==titulo))
    db.update({'view_count':visi},(where('artist')==artista) & (where('title')==titulo))
    print '*****'
    print 'CONTENIDO ACTUALIZADO'
    print '*****'

def listar_todo():
    db = TinyDB('contents.json')
    lista=db.all()
    return lista
#####

```

Figura 4. 20. Método TinyDB y generación de archivo JSON. Fuente: propia.

Una vez obtenida la URL de los contenidos musicales de video en el paso 6, en el paso 7 se procede con la descarga de los contenidos multimedia en formato .mp4 haciendo uso de la librería youtube-dl (ver figura 4.21). De esta manera, se descargaron 200 videos para la conformación del *dataset*, el cual incluye en promedio 40 canciones por emoción de acuerdo al modelo propuesto en la sección 4.6.1.

```

def download_video(url,art,tit,ani):
    nombre=ani+'/'+'art+' - '+tit+'.mp4'
    ydl_opts = {
        'format': '22,18',
        'outtmpl': nombre,
        'forcethumbnail': True,
        'writethumbnail': True
    }
    with youtube_dl.YouTubeDL(ydl_opts) as ydl:
        ydl.download([url])
    time.sleep(2)
    return 'Descargado: 100%'
#####

```

Figura 4. 21. Descarga de contenidos multimedia. Fuente: propia.

En la tabla 4.2 se presenta un resumen detallado con las diferentes características musicales para una entrada en particular del documento JSON/XML, las cuales son representadas en diferentes colores de acuerdo a la notación presentada en la figura 4.19. Dentro de estos campos se destacan las propiedades de *energy*

(*arousal*) y *valence*, las cuales fueron usadas para la determinar la emoción asociada a cada contenido multimedia musical.

Tabla 4. 2. Características musicales para cada video, archivo JSON. Fuente: propia.

CARACTERISTICA	VALOR	COLOR
Id	1	Yellow
Liveness	0.094833	Green
Energy	0.717726	Cyan
Tempo	129.882	Magenta
speechiness	0.106457	Blue
currency	0.22104064240294416	Red
instrumentalness	0.338526	Dark Blue
duration	313.94844	Teal
view_count	822693136	Brown
Mood	Excited	Purple
Artista	Major Lazer	Red
url	http://www.youtube.com/watch?v=YqeW9_5kURI	Olive
title	Lean On	Grey
acousticness	0.033117	Blue
danceability	0.862321	Red
time_signature	4	Purple
loudness	-5.691	Light Green
valence	0.421374	Cyan
mode	0	Dark Blue

4.7 Evaluación del dataset de contenidos

Con el fin de evaluar el dataset de contenidos multimedia de video, se diseñó e implemento un servicio de VoD basado en emociones, el cual presenta un listado de contenidos multimedia de video clasificados por emoción. En la figura 4.22 se presenta la arquitectura básica del servicio de VoD basado en emociones, la cual está formada por los siguientes módulos funcionales: proveedor de contenido multimedia, proveedor del servicio, proveedor de red y consumidor de contenido.



Figura 4. 22. Arquitectura para el despliegue del servicio de VoD. Fuente: propia.

El proveedor de contenido multimedia es el módulo encargado de alojar y proporcionar los contenidos multimedia de video al servicio de VoD. El proveedor

del servicio se encarga por su parte de controlar la lógica del servicio de VoD y permitir la difusión del contenido multimedia de video a través de la red haciendo uso del protocolo HTTP. Este módulo está formado por el servidor de contenidos multimedia Apache, el gestor de bases de datos MySQL (almacena la información de los usuarios y del dataset) y tecnologías del lado del servidor como PHP. El módulo proveedor de red se encarga de brindar soporte de red para garantizar la distribución del contenido multimedia de video a través de la red y hasta el usuario. Finalmente, el módulo cliente tiene por función consumir el contenido multimedia de video, para lo cual hace uso de tecnologías del lado del cliente como: HTML5, Javascript y el framework bootstrap para el diseño, lógica de la interfaz y la reproducción del contenido. Para el consumo del servicio se requiere un navegador con soporte para JavaScript y HTML5, ejemplo: Google Chrome, Firefox, Opera, entre otros.

En la figura 4.23 se presenta la interfaz web del servicio de VoD basado en emociones, la cual está formada por cuatro componentes principales: en “1” se encuentran un conjunto de botones que representan las cinco emociones de entrada, que permiten al usuario seleccionar manualmente el tipo de contenidos musicales que desea visualizar. Una vez el usuario escoge la emoción de entrada, se le presentan un conjunto de contenidos que han sido previamente clasificados en el dataset de videos musicales (“2”), según el modelo de clasificación de emociones presentado en la sección 4.6.1. A partir del catálogo presentado en “2”, el usuario puede escoger el contenido a visualizar, mediante el componente de reproducción (“3”). Finalmente, asociado al componente de reproducción existe un panel de control de reproducción que permite adelantar, atrasar, controlar el volumen y la resolución del video.



Figura 4. 23. Interfaz de usuario del servicio de VoD. Fuente: propia.

Capítulo 5

5. Sistema de recomendaciones contextual

5.1 Introducción

Los sistemas de recomendaciones son herramientas encargadas de guiar a un usuario en la escogencia de un contenido a partir de diversas opciones [18] [19]. Dentro de las aproximaciones clásicas de los sistemas de recomendaciones se encuentran los basados en contenidos y los basados en filtros colaborativos, los cuales recomiendan un conjunto de contenidos a partir de la información del perfil del usuario y la información de usuarios vecinos. A pesar de que estas aproximaciones han sido ampliamente difundidas, tienen como dificultad el problema de arranque en frío, así como la necesidad de que las recomendaciones generadas se aproximen más a las necesidades reales de un usuario. Es por lo anterior que surgen los sistemas de recomendaciones basados en contexto, los cuales a partir de variables de contexto infieren información útil, cuando el sistema se encuentra en estados preliminares. En este capítulo se presenta el diseño e implementación de un sistema de recomendaciones basado en contexto para contenidos multimedia, el cual considera como variables del contexto, un conjunto de variables de tipo fisiológico o biométrico. El objetivo del sistema de recomendaciones es mejorar el problema del arranque en frío y permitir que las recomendaciones sean más adecuadas a las necesidades del usuario.

El resto del capítulo está organizado de la siguiente forma: en la sección 5.2 se presenta la estructura general de un sistema de captura y análisis de medidas fisiológicas. En la sección 5.3 se presentan las diferentes variables de contexto consideradas por el sistema de recomendaciones basado en contexto. En la sección 5.4 se presentan los diferentes equipos de medidas fisiológicas explorados para la obtención de las variables de contexto. En la sección 5.5 se presenta el diseño del módulo hardware-software para la captura de variables de contexto de

usuario, considerando los equipos explorados en la sección 5.4. En las secciones 5.6 y 5.7 se presentan los conceptos y definiciones de los sistemas de recomendaciones y sus aproximaciones clásicas. En la sección 5.8 se presentan los conceptos y definiciones de los sistemas de recomendaciones basados en contexto. Finalmente, en la sección 5.9 se presenta el diseño e implementación del sistema de recomendaciones basado en contexto, lo cual incluye el diseño de su sistema de recomendaciones clásico asociado.

5.2 Sistema de medidas fisiológicas

El sistema nervioso autónomo (SNA) se encarga de controlar los sistemas viscerales del organismo y su función global consiste en mantener la situación de homeostasis del organismo y efectuar las respuestas de adaptación ante cambios en el medioambiente externo e interno [71]. El SNA está integrado por el sistema simpático, parasimpático y entérico. Esta actividad autonómica es vista como un componente importante de la respuesta a la emociones [72], las cuales pueden ser analizadas haciendo uso de las medidas fisiológicas.

El objetivo de la fisiología humana es brindar una explicación acerca de las características y mecanismos específicos del cuerpo humano que hacen que sea un ser vivo. El hecho de mantenerse vivo es el resultado de sistemas de control complejos, que regulan actividades como la respiración, la circulación, los sentidos, el metabolismo, la temperatura, entre otros [73]. Dichas actividades se controlan directamente por el SNA y están asociadas a variables o señales fisiológicas, que pueden ser monitorizadas con ayuda de distintos tipos de sensores.

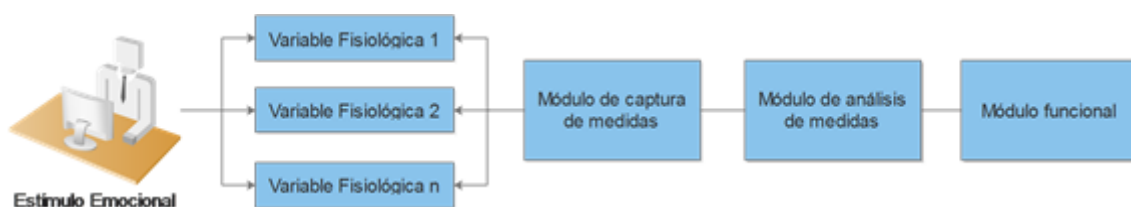


Figura 5. 1. Sistema de medidas fisiológicas. Fuente: propia.

En la figura 5.1 se pueden observar los módulos que conforman un sistema de medidas fisiológica de manera general. En primera instancia se capturan las medidas fisiológicas haciendo uso de plataformas hardware. A partir de la captura de estas variables, se procede al análisis dependiendo de lo que se quiera inferir de cada una de las variables fisiológicas. Una vez realizado el análisis anterior, el siguiente módulo corresponde a la funcionalidad específica para la cual está pensado el sistema de medidas fisiológicas. En el caso del presente trabajo de doctorado este último módulo tiene por función inferir las emociones de un usuario,

para a partir de ellas generar un conjunto de recomendaciones de contenidos multimedia.

5.3 Variables del contexto de usuario

Tal como se menciona en el capítulo uno, el contexto puede ser entendido como cualquier información que puede ser utilizada para caracterizar el estado de una entidad. Una entidad puede ser una persona, un lugar o un objeto relevante para la interacción entre el usuario y la aplicación. En el caso del servicio de video bajo demanda se distinguen cuatro dimensiones del contexto a saber cómo se puede observar en la figura 5.2.



Figura 5. 2. Variables del contexto. Fuente: propia.

En lo que respecta al sistema de recomendaciones basado en contexto que se presenta en este capítulo, se aborda el contexto desde la dimensión de usuario a través del uso de variables fisiológicas como: **frecuencia cardíaca, voz y respuesta galvánica de la piel**, las cuales son utilizadas para la inferencia de emociones contribuyendo a atacar el problema de arranque en frío. A continuación se presenta un estudio detallado de cada una de estas variables fisiológicas antes mencionadas.

5.3.1 Variabilidad de la frecuencia cardíaca

Uno de los métodos no invasivos más utilizados en el análisis y valoración de la actividad cardíaca, es la frecuencia cardíaca (FC). Los latidos en el corazón se producen con una frecuencia variable, es decir que el tiempo entre dos latidos va variando entre uno y otro [74]. Entonces la variación de la frecuencia cardíaca (VFC) es la variación de la frecuencia del latido cardíaco durante un intervalo de tiempo. Una manera habitual de medir esta variabilidad es por medio de un electrocardiograma, en el cual se detecta cada una de las ondas R tal como se aprecia en la figura 5.3.

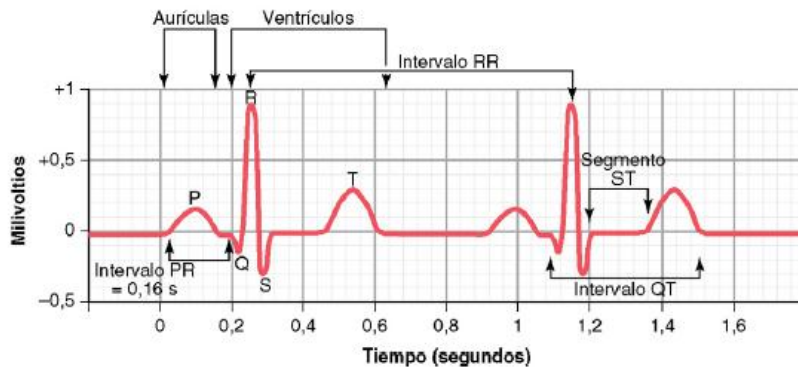


Figura 5. 3. Variabilidad de la frecuencia cardíaca. Fuente: tomado de [51].

Al momento de detectar cada onda R se calcula el tiempo entre las distintas ondas R consecutivas o intervalos RR. Este intervalo RR se encarga de medir el periodo cardíaco y a la serie o conjunto de intervalos RR es lo que se conoce como VFC.

Hay distintas métricas usadas para el análisis para el análisis de la VFC, las hay en el dominio del tiempo, la frecuencia y a través de métodos geométricos. Para el desarrollo del presente trabajo se ha optado por el uso de los métodos en el dominio del tiempo y métodos geométricos para el cálculo del índice de estrés mental, el cual puede ser relacionado con el nivel de arousal del modelo de emociones presentado en el capítulo 4. A continuación se describen cada una de estas métricas en conjunto con sus ecuaciones asociadas.

5.3.1.1 Métodos en el dominio del tiempo

Las medidas en el dominio del tiempo son más sencillas de realizar y calcular que las medidas del dominio de la frecuencia. Dichas medidas cuantifican la variación comprendida en el intervalo entre latidos (IBI), utilizando para ello medidas estadísticas. Algunas de estas y con las cuales se ha trabajado en la presente investigación son: RMSSD, pNN50, la mRR y la SDNN [75].

5.3.1.1.1 RMSSD

Es también conocido como el valor efectivo o valor RMS de los intervalos RR y se refiere a la raíz cuadrada del valor medio de la suma de las diferencias al cuadrado de un grupo de intervalos RR sucesivos. Las unidades de esta métrica están dadas en milisegundos (ms) y representa a nivel fisiológico la actividad de la cadena parasimpática en la regulación de la actividad cardíaca vagal [76] (ver ecuación 5.1).

$$RMSSD = \sqrt{\text{media} \left[\left[\frac{RR_{i+1} - RR_i}{i} \right]^2 \right]}$$

Ecuación 5. 1. Fórmula para calcular RMSSD.

Donde:

RR_i: Corresponde al valor obtenido para el intervalo RR número i.

RR_{i+1}: Corresponde al valor obtenido para el intervalo RR número i+1.

5.3.1.1.2 pNN50

Esta medida representa el porcentaje de intervalos RR consecutivos que discrepan en más de 50 ms entre sí. A nivel fisiológico esta métrica es interpretada como el índice de nivel de prevalencia de regulación del sistema parasimpático sobre el simpático (valor relativo) [76], (ver ecuación 5.2).

$$pNN50 = \frac{\left[\left[\frac{RR_{i+1} - RR_i}{i} \right] * 100 \right]}{N - 1}$$

Ecuación 5. 2. Fórmula para calcular pNN50

Donde:

RR_i: Corresponde al valor obtenido para el intervalo RR número i.

RR_{i+1}: Corresponde al valor obtenido para el intervalo RR número i+1.

N: Corresponde al número de intervalos RR usados para el cálculo.

5.3.1.1.3 mRR

Esta métrica hace referencia a la media de los intervalos RR, es decir el promedio del conjunto de intervalos RR capturados [76] (ver ecuación 5.3).

$$mRR = \frac{(\sum_{i=1}^N RR_i)}{N}$$

Ecuación 5. 3. Fórmula para calcular mRR.

Donde:

RR_i: Corresponde al valor obtenido para el intervalo RR número i.

N: Corresponde al número de intervalos RR usados para el cálculo.

5.3.1.1.4 SDNN

Esta métrica representa la desviación estándar de los intervalos RR, es decir cuantifica la variabilidad global o nivel de dispersión del conjunto de intervalos RR [76] (ver ecuación 5.4).

$$SDNN = \sqrt{\frac{\sum_{i=1}^N (RR_i - mRR)^2}{N - 1}}$$

Ecuación 5. 4. Fórmula para calcular SDNN.

Donde:

RR_i: Corresponde al valor obtenido para el intervalo RR número i.

mRR: Es el valor de la métrica mRR definido en el apartado 5.3.2.13.

N: Corresponde al número de intervalos RR usados para el cálculo.

5.3.1.2 Medida del nivel de estrés mental

Al igual que ocurre con las emociones, no hay un común acuerdo para definir el estrés mental, sin embargo la definición que se ha considerado en este trabajo es: “El estrés puede ser interpretado como una amenaza a la integridad psicológica de un individuo que da lugar a respuestas fisiológicas y/o de comportamiento”, [77]. Para el análisis del estrés mental es posible utilizar las medidas estadísticas de RMSDD, pNN50 y la ecuación del índice de estrés propuesto por Bayevsky. A continuación, se describe como estas medidas son usadas para estimar el nivel de estrés de un individuo.

5.3.1.2.1 RMSSD y pNN50

Se ha encontrado según la literatura una relación entre las medidas estadísticas en función del tiempo como lo son la RMSDD y pNN50, con el estrés mental percibido por una persona. Como se vio en la sección 5.3.2.1, la primera corresponde al valor medio de la suma de las diferencias al cuadrado de todos los intervalos RR sucesivos y la última al porcentaje de los intervalos RR consecutivos que discrepan en más de 50 ms entre sí.

Una forma de averiguar cuándo el usuario presenta un nivel de estrés alto, es observando cuándo las métricas de RMSSD y pNN50 presentan un valor bajo, mientras que, si el usuario disminuye su estrés mental o se encuentra en un estado de relajación, dichas métricas tienden a presentar valores altos (ver tabla 5.1).

Tabla 5. 1. Medidas estadísticas y nivel de estrés. Fuente: propia.

MEDIDA ESTADÍSTICA		NIVEL DE ESTRÉS
RMSSD	↓	↑
RMSSD	↑	↓
pNN50	↓	↑
pNN50	↑	↓

5.3.2.2.2 Índice de estrés de Bayevsky

Este algoritmo surge a lo largo del trabajo desarrollado por la unión soviética en el análisis de la variable HRV (heart rate variability) en el área de la medicina espacial. El Índice de estrés (SI) o índice del sistema de regulación de tensión es entendido como el nivel de tensión de los sistemas de regulación (nivel de prevalencia de la actividad de regulación central por encima de los mecanismos autonómicos). El cálculo del SI se basa en métodos geométricos haciendo uso de la curva de distribución cardio-histograma dibujada a partir del estudio de las variaciones del pulso o ritmo cardíaco, la cual identifica características como: Mo (moda), Amo (Amplitud de la moda), M*DMn (rango de variación o varianza) [76] (ver ecuación 5.5).

$$SI = \frac{AMo}{2Mo * (M * DMn)}$$

Ecuación 5. 5. Formula Índice de estrés mental.

Dónde:

Mo: Moda. Es entendido como el nivel presumible en el que se encuentra trabajando el sistema cardiovascular. En términos matemáticos, hace referencia al valor de RR que más presenta el usuario en el conjunto de medidas capturadas.

AMo: Amplitud de la moda. En términos fisiológicos es el índice nominal de actividad de la cadena de regulación simpática. En términos matemáticos se refiere al porcentaje o porción de los intervalos que corresponden con el valor de la moda de la muestra tomada.

M*DMn: Rango de varianza. Es la diferencia entre los valores máximos y mínimos de los cardio-intervalos RR. Este valor es también conocido como varianza.

En la tabla 5.2, se pueden apreciar los tres posibles rangos que el índice de estrés puede presentar, los cuales están asociada a tres estados emocionales: estresado, relajado y normal. Así mismo esta tabla presenta la relación entre el rango de estrés y el nivel de arousal de un contenido multimedia, lo cual es tenido en cuenta para la inferencia de emociones.

Tabla 5. 2. Rangos SI asociados a cada estado. Fuente: propia.

RANGO DE SI	ESTADO ASOCIADO	NIVEL DE AROUSAL
>150	estresado	>0.5
≥40 y ≤150	normal	≥0.14 y ≤0.5
<40	relajado	<0.5

A partir de lo anterior, se puede concluir que el índice de estrés mental permite asociarse a 3 estados emocional como se observa en la tabla 3.2. Estos determinan los rangos del nivel de arousal, los cuales son fundamentales en el sistema de recomendaciones, ya que por medio de estos niveles junto con la valence se identifican los contenidos que responden al estado de ánimo asociado como se verá posteriormente en este capítulo.

5.3.2 Medida de la respuesta galvánica de la piel

Se sabe que en la piel los humanos cuentan con una serie de glándulas sudoríparas, las cuales se encargan principalmente de la regulación térmica del cuerpo. Esta es controlada por el sistema nervioso autónomo a través de distintas hormonas, y fibras nerviosas (ver figura 5.4) [73].

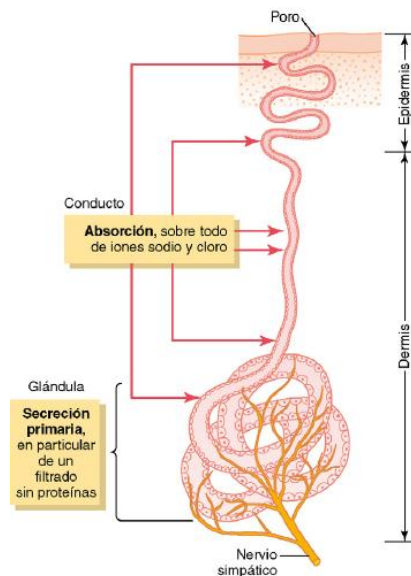


Figura 5. 4. Glándula sudorípara. Fuente: tomado de [51].

Según los estudios presentados en [78], cuando un individuo está bajo estrés mental, la actividad de las glándulas sudoríparas comienza a aumentar permitiendo la secreción de sudor, logrando de esta manera que incremente la conductividad de la piel. Las zonas que cuentan con más densidad de glándulas sudoríparas son las manos y los pies. En la figura 5.5 se puede observar un ejemplo del aumento en la conductividad de la piel.

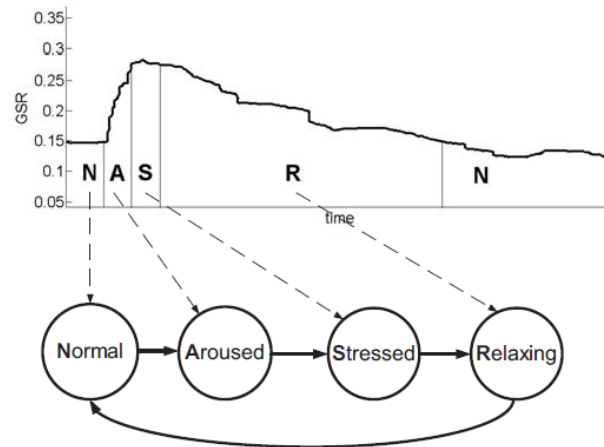


Figura 5. 5. Gráfico señal GSR. Fuente: tomado de [57].

Así, en la figura 5.5 se observa como en una primera instancia se obtiene la señal base o normal de un individuo, y a medida que se somete a un estrés mental comienza la activación de las glándulas sudoríparas, ocasionando que aumente la activación o arousal hasta llegar a un pico de estrés. Una vez el individuo entra en una etapa de relajación, se observa como la conductividad comienza a disminuir paulatinamente hasta llegar a la línea base o normal del usuario.

Como conclusión cabe resaltar que la medida de la conductancia trabaja como soporte a la VFC. Así, el GSR se encarga entonces de determinar si el individuo comienza aumentar su nivel de estrés o por el contrario comienza a relajarse, lo cual en el caso particular de un sistema de recomendaciones de contenidos emocionales con características de arousal y valence, permitirá cambiar el orden del listado de contenidos, teniendo en cuenta el nivel de arousal.

5.3.2.1 Regresión Lineal y pendiente

Para el presente proyecto, la medida del GSR se implementó como indicador de cuando comienza aumentar o disminuir el nivel de estrés en el usuario. Ya que las medidas que se toman pueden variar dentro de determinados rangos dependiendo si la persona está estresada o se mantiene relajada, se hizo uso del método de regresión lineal, el cual realiza una aproximación para determinar si una función es creciente o decreciente al obtener el valor de su pendiente (ver figura 5.6). Si la función es negativa significa que es decreciente en el intervalo que se analiza, mientras que si es positiva, la función es creciente en el intervalo analizado. Para el cálculo de la regresión lineal de los datos capturados con el sensor GSR se hizo uso de la función "linalg.lstsq" de la librería numpy de Python (ver figura 5.7), la cual después de establecer las variables dependiente e independiente, retorna la pendiente de la recta y el corte en el eje y para su posterior análisis.

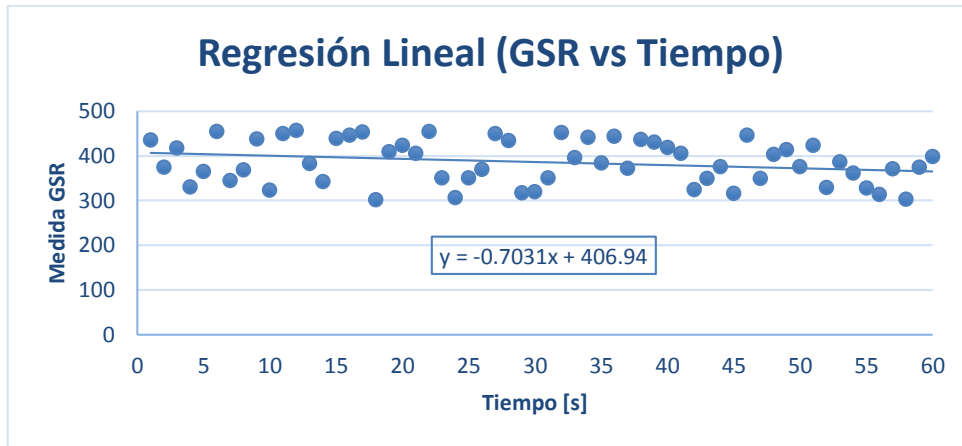


Figura 5. 6. Gráfico regresión lineal a conjunto de datos. Fuente: propia.

```
def get_valor_estimulo(self, lista_gsr):
    n = len(lista_gsr)
    lista_x = list(range(1,n+1))
    x = numpy.array(lista_x)
    y = numpy.array(lista_gsr)
    A = numpy.vstack([x, numpy.ones(len(x))]).T
    p, c = numpy.linalg.lstsq(A, y)[0]
    print "pendiente->",p, " -- corte->",c
    return p
```

Figura 5. 7. Método para cálculo de regresión lineal. Fuente: propia.

5.3.3 Procesamiento de la voz

La voz es uno de los principales modos de comunicación usado entre las personas, razón por la cual a nivel de la literatura se han venido proponiendo sistemas informáticos capaces de simular y reconocer la voz de manera electrónica. En cuanto al estudio de las emociones a partir de la voz, se destacan ciertos parámetros que pueden ser usados para el análisis emocional, tales como: el pitch, la duración, la calidad de la voz [79] [80].

5.3.3.1 Pitch

El pitch es definido como la frecuencia fundamental (F0) a la que vibran las cuerdas bucales. Esta frecuencia se considera como una de las principales portadoras de la información sobre las emociones. En términos generales el pitch asociado a la voz puede ser un indicador del grado de exaltación del locutor. De este modo la curva de tono es discontinua para emociones consideradas negativas (miedo, enfado) y suave para las emociones positivas (alegría).

5.3.3.2 Duración

La duración es una característica que está asociada a la velocidad del habla y a la situación de los acentos y pausas. Dentro de los parámetros de la duración que pueden usarse para el análisis emocional se destacan: la velocidad de locución y el número de pausas con su respectiva duración. Con respecto al primer parámetro, un locutor en estado de excitación tiende a acortar la duración de las sílabas, por lo que la velocidad de locución (sílabas/segundo o palabras/minuto) se incrementará. En lo que se refiere al segundo parámetro, un locutor en estado de exaltación tenderá a hablar con menos pausas y pausas más cortas, mientras que un locutor deprimido hablará más lentamente, con pausas más largas.

5.3.3.3 Calidad de la voz

La calidad de la voz es una característica que puede estar asociada a parámetros como: la intensidad, las irregularidades de la voz, el cociente de energía entre alta y baja frecuencia, la respiración/laringerización. La intensidad está relacionada con la percepción del volumen, lo cual se ve reflejado en la amplitud de la forma de onda. Las irregularidades de la voz durante el enfado se ven representadas en las fluctuaciones de un pulso glotal a otro, mientras que en emociones como la pena, la desaparición de la voz puede ser considerada como un indicador. Con respecto al cociente entre energía de alta y baja frecuencia, es importante mencionar que una gran cantidad de energía en las frecuencias altas está asociada a emociones como el enfado, mientras que una baja concentración de energía en las frecuencias altas se relaciona con la depresión o la calma. En lo referente a la respiración/laringerización, la primera describe la generación de ruido respiratorio, de tal modo que la componente fundamental es más fuerte y las frecuencias altas son remplazadas por ruido respiratorio. Por su parte, la segunda característica se caracteriza por la vibración aperiódica de las cuerdas vocales (pulso glotal estrecho y pitch bajo) lo que se ve reflejado en una voz chirriante.

Dentro del presente trabajo de doctorado se hizo uso de la herramienta libre OpenEar¹⁶, la cual permite extraer y analizar los componentes auditivos de la voz, teniendo en cuenta entre otros parámetros el pitch, la duración y la calidad de la voz [80]. Mediante el análisis de estos parámetros OpenEar permite la obtención de características emocionales como son el arousal y la valence. Mediante estos dos parámetros es posible la clasificación de la emoción según el modelo de Russell. Así, dentro del servicio de VoD propuesto en este trabajo, el software OpenEar permitió determinar la emoción de entrada con la que el usuario empieza a hacer uso de la aplicación. A partir de esta emoción inicial, el sistema genera un listado inicial de recomendaciones musicales asociados al arousal y valence de entrada,

¹⁶ Herramienta libre para el análisis y obtención de emociones a partir de la voz desarrollada en la Universidad Técnica de Múnich. La herramienta está disponible en: <https://sourceforge.net/projects/openart/>

el cual va cambiando según el estudio de la variación del arousal evidenciado en el comportamiento de las variables fisiológicas VFC y GSR.

5.4 Equipos para medidas fisiológicas

En la actualidad con el auge de internet de las cosas, han aparecido un gran número de dispositivos que permiten la captura de las variables fisiológicas, los cuales se han venido utilizando en su mayor parte en el ámbito deportivo, con el fin de obtener datos en tiempo real sobre el rendimiento físico de los atletas. A estos equipos de medición se les conoce comúnmente como wearables y son dispositivos que pueden usarse en el cuerpo o como accesorio en la ropa, con el propósito de obtener información fisiológica de interés de un individuo. Estos dispositivos se caracterizan por ser multitarea y tener la capacidad de interactuar con otros dispositivos y conectarse a internet. Algunos de los dispositivos utilizados para la medición de variables fisiológicas como la frecuencia cardíaca y la respuesta galvánica de la piel se presentan a continuación.

5.4.1 Pulse sensor cardíaco-arduino

El pulse sensor¹⁷ para frecuencia cardíaca plug-and-play fue diseñado para ser compatible con la tarjetas de desarrollo Arduino. Este sensor puede ser utilizado por estudiantes, artistas, deportistas, fabricantes y desarrolladores de aplicaciones, con el fin de incorporar fácilmente los datos de frecuencia cardíaca en tiempo real sobre sus proyectos (ver figura 5.8).

Figura 5.8 (a)



Figura 5.8 (b)

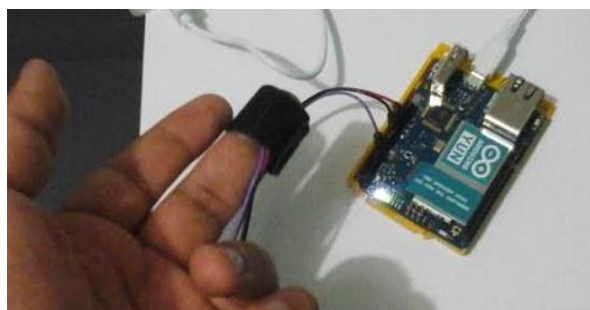


Figura 5. 8. Dispositivo sensor de pulso cardíaco-arduino. Fuente:propia.

En la figura 5.9a) se observa el sensor de frecuencia cardíaca y sus componentes adicionales para realizar la medición. Estos componentes de

¹⁷ Pulse sensor arduino usado para la medida de la VFC, página web fabricante: <http://pulsesensor.com/>.

medición pueden ser utilizados en los dedos o en el lóbulo de la oreja. En la figura 5.9b) se observa el sensor conectado al dedo corazón del usuario, sobre el cual se está haciendo la medición de la frecuencia cardíaca. El sensor está conectado a la tarjeta Arduino en donde son obtenidas las medidas de las señales de FC y VFC.

5.4.2 Cinturon Zephyr HxM BT

Zephyr¹⁸ es una banda o cinturón para el pecho, el cual cuenta con un sensor de frecuencia cardíaca (ver figura 5.9). Este dispositivo además de medir el ritmo cardíaco por medio de electrodos presentes en el cinturón, toma muestras de los intervalos RR, velocidad a la que se mueve una persona y distancia que recorre. Las medidas capturadas por el cinturón pueden ser transmitidas usando el protocolo bluetooth de bajo consumo (bluetooth LE).

Figura 5.9 (a)



Figura 5.9 (b)



Figura 5. 9. Dispositivo sensor zephyr. Fuente: propia.

El sensor se comunica vía bluetooth con el dispositivo móvil y cuenta con aplicaciones para Android, Windows phone y PC. Esta herramienta cuenta un paquete SDK con recursos para desarrolladores. Así mismo, el dispositivo posee una batería que se carga vía USB. En la figura 5.9a) se presenta el dispositivo con componentes y en la figura 5.9b) se muestra un ejemplo de cómo debe ir conectado correctamente el dispositivo en el usuario.

5.4.3 Cinturón Polar

El sensor de polar¹⁹ es compatible con dispositivos con tecnología bluetooth LE (ver figura 5.10). Para el acceso a la información del sensor se requiere una aplicación separada que permita establecer la conexión y obtener los datos periódicamente. Así mismo, para acceder a los datos del cinturón mediante un

¹⁸ Cinturón zephyr utilizado para la medida de la VFC, página web fabricante: <http://www.zephyranalyzer.com/>.

¹⁹ Sensor polar para medida de la VFC, página web fabricante: http://www.polar.com/en/products/accessories/H7_heart_rate_sensor.

Smartphone, se recomienda el uso de la aplicación APP Polar Beat, la cual sincroniza los datos tomados con la cuenta polar personal trainer del usuario.

Figura 5.10 (a)



Figura 5.10 (b)



Figura 5. 10. Dispositivo sensor polar. Fuente: propia.

Este sensor cuenta con una banda para el pecho con dos electrodos para la captura de las medidas y una pila como alimentación para el dispositivo. Polar ofrece a los desarrolladores documentación para google fit, Apple healthkit, y para su app propietaria accesslink, para lo cual es necesario realizar el registro con Polar para adquirir la cuenta de desarrollador. En la figura 5.10a) se observa el dispositivo y componentes; y en la figura 5.10b) se puede ver su modo de uso.

5.4.4 Galvanic Skin Response (GSR) sensor

El grove sensor GSR para Arduino es un sensor que permite la medición de la conductancia eléctrica de la piel. Este sensor trabaja con un voltaje de entrada de 5V/3.3V, la sensibilidad se puede ajustar mediante un potenciómetro. El dispositivo presenta dos bandas elásticas para los dedos, las cuales tienen dos electrodos que permiten la captura de las medidas (ver figura 5.11).

Figura 5.11 (a)



Figura 5.11 (b)



Figura 5. 11. Dispositivo sensor GSR. Fuente: propia.

En la figura 5.11a) se observa el dispositivo sensor GSR y sus componentes, tarjeta de conexión y cables, además de los electrodos. En la figura 5.11b) se presenta un ejemplo de conexión del sensor GSR en los dedos índice y corazón de la mano.

5.4.5 Arduino Yún

Es una placa electrónica de hardware libre útil para el desarrollo de proyectos de internet de las cosas, la cual combina el potencial de Linux con la facilidad y flexibilidad de Arduino. Cuenta con interfaz Ethernet y wifi, puerto USB, ranura para tarjeta micro-SD, 20 pines digitales entrada/salida de los cuales 7 se pueden usar como salidas PWM y 12 como entradas analógicas (ver figura 5.12). La tarjeta Arduino Yún²⁰ se distingue de otras placas de la familia, en que se puede comunicar con la distribución de Linux a bordo, la cual ofrece un ordenador en red de gran alcance que permite el soporte de comandos tipo cURL y la escritura de *scripts* propios de Shell y Python. Una de las ventajas principales de la tarjeta Arduino Yún es el soporte de servicios web, lo cual permite la implementación de servicios para el acceso remoto a la información capturada de los sensores. La configuración básica de este dispositivo es presentada en detalle en el anexo D de la presente monografía.



Figura 5. 12. Tarjeta hardware Arduino Yún. Fuente: propia.

Así, para la captura de información del contexto, en el presente proyecto de doctorado se eligió trabajar con la plataforma de hardware libre y de internet de las cosas Arduino Yún. Lo anterior teniendo en cuenta las ventajas en cuanto al

²⁰ Placa electrónica de hardware libre Arduino Yún, página web fabricante: <https://www.arduino.cc/en/Main/ArduinoBoardYun>.

despliegue de servicios web y su facilidad para el acceso a información de múltiples sensores, dentro de los cuales se destacan los de tipo biométrico o fisiológico.

5.5 Diseño mecanismo hardware-software

El mecanismo hardware-software diseñado para la captura de variables de contexto es presentado en la figura 5.13. Primero se obtiene la emoción de entrada mediante el módulo de análisis de la voz, el cual usa las propiedades de arousal y valence presentes en la muestra de audio capturada a partir de la voz. Para extraer las propiedades de arousal y valence, se hizo uso de la herramienta de software libre OpenEar, la cual posibilita además la clasificación de dichas propiedades en el modelo emocional de Rusell. A partir de la emoción de entrada se genera un listado de recomendaciones inicial que tiene en cuenta los contenidos asociados a dicha emoción.

Una vez obtenido un arousal de referencia (asociado a la emoción de entrada), este valor es retroalimentado de manera constante a través del análisis de las variables fisiológicas de VFC y GSR. Las variables VFC y GSR son obtenidas por el módulo de captura mediante los sensores: pulse sensor y GSR sensor. A partir de estas medidas, en el módulo de análisis de VFC se procede al cálculo del índice de estrés y de las medidas estadísticas en el dominio del tiempo (RMSSD, pNN50, mRR, SDNN), a través de las cuales es posible obtener y retroalimentar el arousal de referencia (calculado mediante el análisis de la voz), teniendo en cuenta los rangos de relación presentados en la tabla 5.2. Por otra parte en el módulo de análisis del GSR se observa si los valores de conductividad capturados muestran la tendencia del usuario al estrés o a la relajación, para lo cual se hace uso del método de regresión lineal.

De este modo, el módulo de lógica recibe una emoción de entrada del módulo de análisis de la voz, según la cual genera un listado inicial de recomendaciones. Este listado inicial es filtrado de acuerdo al arousal obtenido de manera constante por los módulos de análisis de VFC y GSR. Lo anterior quiere decir que el listado de recomendaciones inicial cambia dependiendo si el usuario se encuentra en un estado de estrés o por el contrario comienza a presentar un estado de relajación, lo que se ve reflejado en el aumento o disminución del arousal o la excitación del usuario.

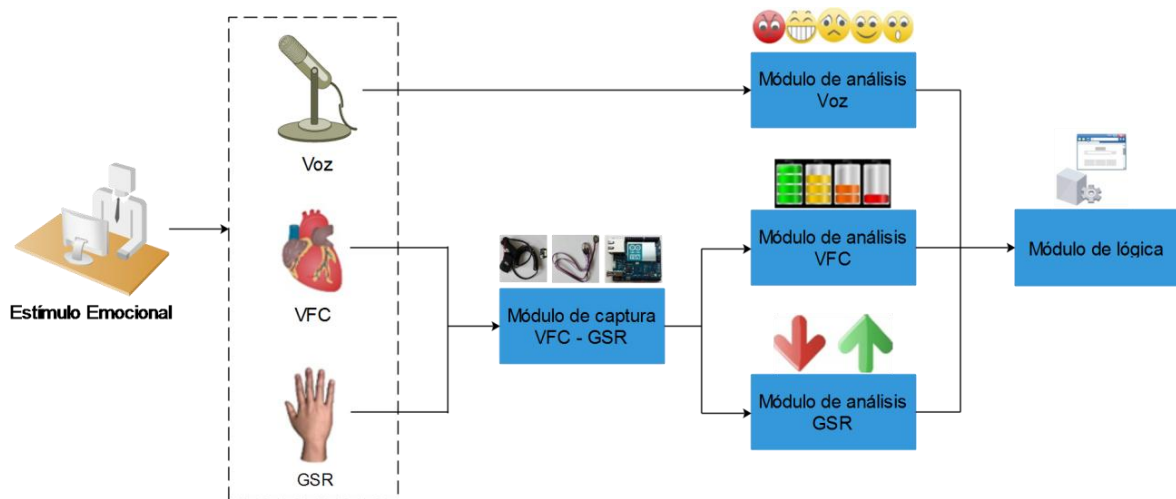


Figura 5. 13. Módulos funcionales para las variables fisiológicas. Fuente: propia.

En la figura 5.14 se puede visualizar un ejemplo de captura de las distintas métricas para el análisis de la VFC, dentro de las que se encuentran: ritmo cardíaco, intervalos RR, promedio del ritmo cardíaco, RMSSD, pNN50, mRR, nivel de estrés y estado de ánimo asociado. En el proceso de captura de las variables se hizo uso del framework web flask de python. Este microframework permite crear aplicaciones web de forma rápida y con muy pocas líneas de código. Flask incluye un servidor web de desarrollo y permite el despliegue de servicios tipo RESTful [81].

Ritmo	81
RR	765
Prom Ritmo	80
rMSSD	12.882091284495269
pNN50	0
mRR	754
mRRi	0.25862068965517243
Nivel Estres	182.98087557080885
Desc Nivel Estres	estresado

Figura 5. 14. Medidas de análisis para la VFC. Fuente: propia.

A modo de ejemplo de la relación entre el índice de estrés y la VFC, en la figura 5.15 se presenta un gráfico con la captura de los intervalos RR tomados durante un minuto para los tres niveles de estrés calculados con el índice de estrés. En este gráfico se puede apreciar que para el caso de las muestras de los intervalos RR normal y relajado, la fluctuación de las curvas es más notoria que para el caso de los intervalos RR estresado. Lo anterior permite concluir que entre menos sea la variación entre los diferentes rangos de los valores RR, el nivel de estrés tiende a ser más alto.

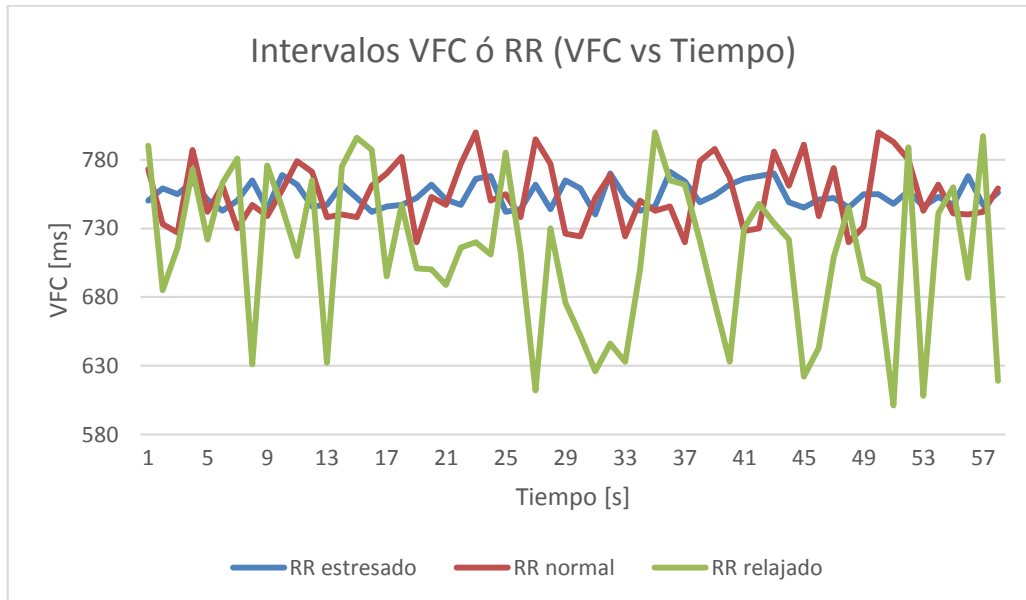


Figura 5. 15. Gráfico estados de ánimo asociados a un índice de estrés (SI). Fuente: propia.

Como conclusiones del diseño del mecanismo hardware-software para la captura de variables del contexto de tipo fisiológico, es importante mencionar que se escogió la plataforma Arduino Yún, por su flexibilidad para el soporte de múltiples sensores, así como sus ventajas para el despliegue de servicios web. Así para la captura de variables fisiológicas se hizo uso de los sensores: pulse sensor (VFC) y GSR, los cuales permitieron el cálculo del índice de estrés y el nivel de arousal. Gracias al arousal inferido es posible retroalimentar el nivel de arousal de referencia obtenido mediante el análisis de la voz, lo cual se ve representado en la actualización constante del listado de recomendaciones.

5.6 Concepto de S.R.

El aumento del tráfico en las redes y la generación de contenidos que surgen cada vez en mayor cantidad, ha llevado al desarrollo de sistemas que permitan el ágil acceso a los contenidos, siendo una de las soluciones la implementación de S.R. para los distintos servicios ofrecidos en la web. Estos se encargan de identificar las preferencias de los usuarios y con base en ellas realizar sugerencias o proporcionar ítems personalizados en la selección a partir de múltiples opciones (ver figura 5.17) [82].

Los ítems por lo general son desplegados en forma de listas las cuales se encuentran ordenadas de acuerdo a un ranking. Estos juegan un papel importante en sitios reconocidos en la web como son: YouTube, Netflix, Yahoo, Amazon, IMDb, entre otros, los cuales al implementarlo mejoran la experiencia en la interacción del usuario con el servicio. Existen en los S.R. distintos tipos de clasificaciones, estos

se dividen principalmente en colaborativos y basados en contenido, aunque también existen divisiones como los basados en conocimiento y los híbridos que mezclan el tipo colaborativo y basados en contenido [18].

A pesar de los beneficios que brindan los S.R. existen problemática que no permiten explotar todo su potencial como son la escasez de datos para efectuar recomendaciones, inconvenientes para la sugerencia de contenidos nuevos y el problema que compete a este trabajo el cual es el de arranque en frío, que surge cuando el sistema está en estados preliminares, es decir, cuando el usuario no ha interactuado con el sistema, por lo cual no hay contenido relevante para ser recomendado al usuario, estos problemas son muy comunes en el enfoque clásico de los S.R. En la figura 5.16 se muestra la arquitectura de un S.R. que describe el funcionamiento básico de este, en donde la información tomada de los módulos de gestión de los ítems y usuarios son tomadas por los S.R. para enriquecer o realimentar el perfil del usuario e miras a la recomendación de contenido relevante a sus gustos.

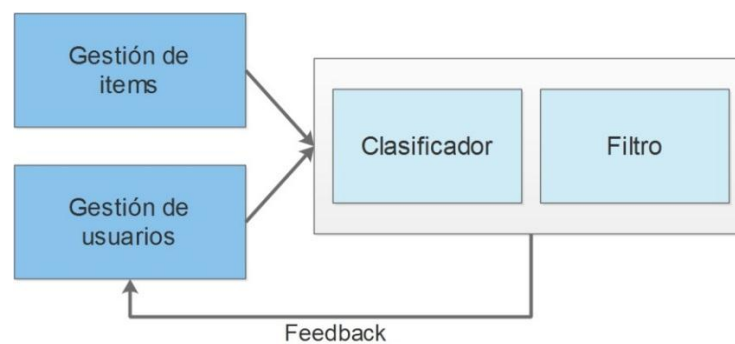


Figura 5. 16. Arquitectura básica del S.R. Fuente: propia.

5.7 Enfoque clásico de los S.R.

A continuación, se describirán los enfoques clásicos de los S.R. como son: el enfoque basado en contenido y los colaborativos, métodos muy utilizados y que fueron base para el origen de nuevos métodos.

5.7.1 Enfoque basado en contenido

Los sistemas que implementan este tipo de recomendaciones basadas en contenido, analizan una serie de descripciones de los elementos o contenidos que el usuario ya ha valorado con anterioridad y de acuerdo a esto construyen un perfil de sus intereses basado en las características del contenido, ver figura 5.17. Este perfil representa de forma estructurada las preferencias de los usuarios, entre mejor

se describan los gustos, más eficiente son las sugerencias que proporciona el sistema. Los S.R. basados en contenido al no tener en cuenta las calificaciones dadas por los demás usuarios para la recomendación de contenido, presentan un problema para sugerir nuevos ítems al consumidor. Además, cuando aún no se han valorado suficientes contenidos este tipo de sistemas presentan el problema de arranque en frío.

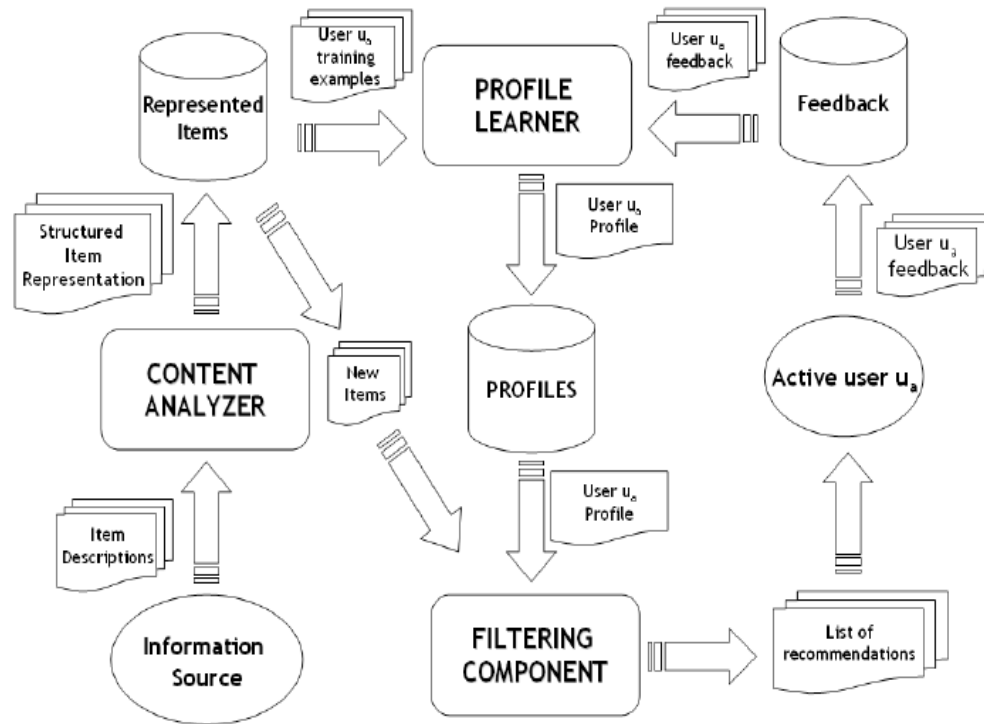


Figura 5. 17. Arquitectura de los S.R. basados en contenido. Fuente: tomado de [13].

Estos sistemas realizan el proceso de recomendación empleando tres pasos, los cuales están a cargo de un componente determinado y pueden ser observados en la figura 5.17. Estos componentes son: analizador de contenido, perfil del usuario y componente de filtrado.

El analizador de contenido se encarga de representar el contenido de los ítems o sugerencias, la información de cada ítem, son analizadas por técnicas de extracción de características, estas son utilizadas por el componente de perfil del usuario el cual se encarga de representar las preferencias del usuario con base en las interacciones que el usuario ha tenido ya sean positivas o negativas con el sistema, de esta manera se puede conocer las preferencias del usuario con relación al contenido que se le proporciona. El componente de filtrado se encarga de realizar la recomendación de ítems teniendo como base el perfil del usuario y para ello proporciona una lista de sugerencias al consumidor. En la figura 5.18 se presenta el esquema funcional empleado por los S.R. basado en contenidos.

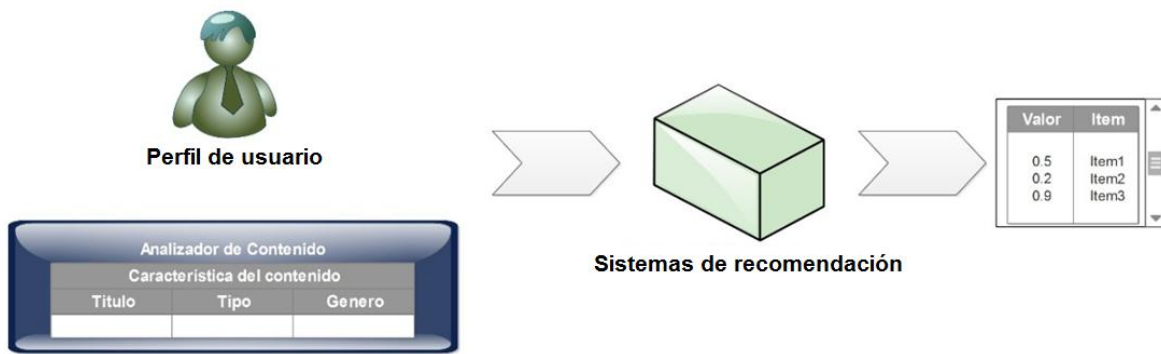


Figura 5. 18. Diagrama de secuencia de un S.R. basado en contenido. Fuente: propia.

Cabe resaltar que uno de los métodos más usados en entornos de contenido multimedia para los S.R. basados en contenido es el de la clasificación y predicción de contenidos multimedia, para la cual se suelen usar aproximaciones de tipo bayesiano, las cuales hacen uso del teorema de bayes con el objetivo de predecir los contenidos multimedia que pueden ser relevantes al usuario mediante el uso de ponderados probabilísticos.

5.7.1.1 Clasificador Bayesiano

Es un marco probabilístico para resolver problemas de clasificación, basándose en la definición de probabilidad condicional y el teorema de bayes. Los clasificadores bayesianos consideran cada atributo y la etiqueta de clase como variables aleatorias (continuas o discretas). Teniendo en cuenta un registro con N atributos (A_1, A_2, \dots, A_n), el objetivo es predecir la clase C_k encontrando el valor de C_k que maximiza la probabilidad a posteriori de la clase, dados los datos $P(C_k/A_1, A_2, \dots, A_n)$ [83].

El teorema de bayes dice, que siendo A y B dos sucesos aleatorios cuyas probabilidades se denotan por $p(A)$ y $p(B)$, siendo $p(B) > 0$ y conociendo las probabilidades a priori de los sucesos A y B , además de conocer la probabilidad condicionada del suceso B dado el suceso A , es decir $p(B|A)$. De acuerdo a lo anterior es posible establecer de esta manera la probabilidad a posteriori del suceso A que verifica el suceso B , es decir $p(A|B)$, pudiéndose calcular a partir de la ecuación 5.6.

$$p(A|B) = \frac{p(A)p(B|A)}{p(B)}$$

Ecuación 5. 6. Fórmula de probabilidad condicionada de A dado B .

Donde:

P(A): Probabilidad de que ocurra el suceso A .

P(B): Probabilidad de que ocurra el suceso B .

P(B|A): Probabilidad del suceso B, dado que ocurre el suceso A.

El clasificador de Naive Bayes busca encontrar la hipótesis más probable. La ecuación 5.7 modela la fórmula de Naive Bayes.

$$V_{nb} = \arg \max_{v_j \in V} P(v_j) \prod P(a_i | v_j)$$

Ecuación 5. 7. Fórmula de Naive Bayes.

Donde:

P(v_j): Probabilidad de ocurrencia de la categoría v_j.

P(a_i|v_j): Probabilidad de ocurrencia del atributo a_i dado que sucede la categoría v_j.

En La ecuación 5.7 V_{nj} es la probabilidad de que conocidas la n características que describen al ejemplo, estas pertenezcan a la categoría v_j. Esta es cada una de las categorías V dentro de las que se puede clasificar el ejemplo. La probabilidad de P(a_i|v_j) está definida de forma general como:

$$P(a_i | v_j) = \frac{nc}{n}$$

Ecuación 5. 8. Probabilidad condicionada, Naive Bayes.

Donde:

nc: número de veces que ocurre la categoría a_i en la categoría v_j.

n: número de casos totales de la categoría v_j.

En caso de que nc sea igual a cero se usa la estimación de Laplace, definida en la ecuación 5.9.

$$P(a_i | v_j) = \frac{nc + 1}{n + k}$$

Ecuación 5. 9. Fórmula para la estimación de Laplace.

Donde:

nc: número de veces que ocurre la categoría a_i en la categoría v_j.

n: número de casos totales de la categoría v_j.

k: es el número de valores distintos de la categoría a_i.

Un ejemplo del uso del teorema de bayes se puede observar en la tabla 5.3, para este caso el clasificador de Naive Bayes puede ser usado para predecir la posible categoría de un caso x a partir de un conjunto de casos ocurridos, en donde se considera el cálculo de una recomendación para una tienda virtual de películas en

las que los usuarios han valorado los contenidos multimedia con calificaciones de 1 a 3.

Tabla 5. 3. Ejemplo del teorema de Naive Bayes. Fuente: propia.

MUESTRA	GÉNERO	AÑO	CALIFICACIÓN
1	Acción	2015	1
2	Drama	2014	2
3	Comedia	2013	2
4	Acción	2016	1
5	Comedia	2013	3
6	Drama	2015	1
7	Acción	2015	2

Se tienen 3 categorías V con los siguientes elementos los cuales corresponden a la Calificación = {1, 2, 3}. Las películas cuentan con 2 características $a = \{\text{Genero, Año}\}$, el ejemplo consiste en hallar la calificación de una película, que presenta como características $a = \{\text{comedia, 2014}\}$.

Para obtener la predicción de la posible calificación que tendría la película, es necesario establecer el valor de cada una de las siguientes expresiones o probabilidad condicionales, ver ecuación 5.7; de las cuales la que presente la probabilidad más alta será la calificación que obtendrá la película.

- $P(1) * P(\text{Comedia}|1) * P(2014|1)$
- $P(2) * P(\text{Comedia}|2) * P(2014|2)$
- $P(3) * P(\text{Comedia}|3) * P(2014|3)$

Para ilustrar como se obtienen cada una de las probabilidades se hará paso a paso para la calificación cuyo valor es uno, ver ecuación 4.6.

- $P(1) = 3/7$ (la probabilidad de la calificación 1 del experimento, existen 3 calificaciones que son 1 del total de 7 que existen).
- $P(\text{Comedia}|1) = (nc+1)/(n+k) = (0+1)/(2+3) = (1/5)$ (dado que no existen casos en lo que el género sea comedia y la valoración sea 1 se hace uso de la ecuación 4)
- $P(2014|1) = (nc+1)/(n+k) = (0+1)/(1+4) = 1/5$ ($K=4$ porque hay 4 años en el experimento y $n=1$ debido a que solo hay una característica con valor 2014)
- $P(1) * P(\text{Comedia}|1) * P(2014|1) = (3/7) * (1/5) * (1/5) = 3/175$
- $P(2) * P(\text{Comedia}|2) * P(2014|2) = (3/7) * (1/2) * 1 = 3/14$
- $P(3) * P(\text{Comedia}|3) * P(2014|3) = (1/7) * (1/2) * (1/5) = 1/70$

Al obtener todos los cálculos se puede observar, que como la probabilidad más alta la presenta la expresión $P(2) * P(\text{Comedia}|2) * P(2014|2)$ con $3/14$, esto quiere decir

que es probable que la calificación de un película con genero comedia y año 2014 sea de 2. El método descrito anteriormente se utilizó en el presente trabajo para el diseño e implementación del S.R. clásico.

5.7.2 S.R. colaborativos

El filtrado colaborativo está ampliamente implementado en los S.R. Este realiza las recomendaciones teniendo como base a otros usuarios que presentan gustos similares, la similitud en los gustos es obtenida de acuerdo al historial de calificaciones del conjunto de usuarios encontrado. Los S.R. colaborativos buscan solucionar el problema de la recomendación de nuevos contenidos presente en el anterior enfoque, pero aún no es una solución final a este problema si se tiene escasa información por parte de los demás usuario, además de que presenta inconveniente del arranque en frío ya que el sistema en un inicio no cuenta con información importante para efectuar una sugerencia, además de no conocer con precisión los gustos del usuario. En la figura 5.19 se puede observar el gráfico resume de su funcionamiento.

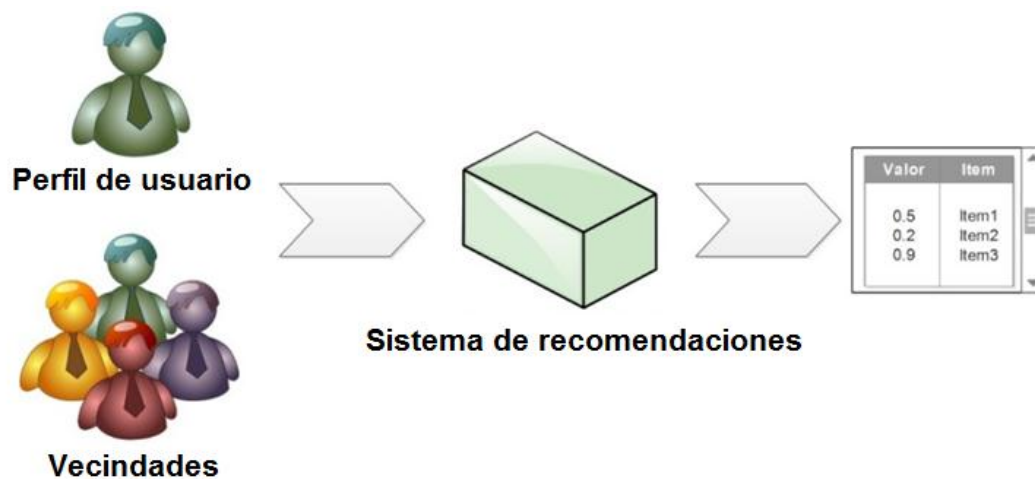


Figura 5. 19. Diagrama de secuencia de un S.R. basado en filtros colaborativos. Fuente: propia.

La mayoría de sistemas colaborativos emplean la técnica basada en vecindades, la cual consiste en agrupar un número de usuarios con un determinado grado de similitud, la predicción al usuario activo se realiza con el cálculo de una media ponderada de las calificaciones de los usuarios seleccionados.

Previendo los anteriores problemas de los enfoques clásicos, como alternativa de solución a la escasez de datos y el arranque en frío, se usan los S.R. basados en contexto, los cuales buscan hacer frente a dichos problemas mediante variables contexto de usuario, en la sección 4.4 el lector podrá observar esta temática.

5.8 S.R. basados en contexto

Antes de explicar lo que son los S.R. basados en contexto, es necesario conocer que se quiere decir con contexto, el cual de forma general se define como: “El conjunto de elementos que tienen influencia alguna sobre el objeto” [82]. En la literatura los sistemas conscientes del contexto, se definieron inicialmente como la información relacionada con la ubicación del usuario, los objetos y personas que se encuentra a su alrededor. Cada vez se añaden muchas más variable del contexto como el tiempo, el clima y las variables fisiológicas, que enriquecen más a los sistemas conscientes del contexto.

En los S.R. basados en contexto se tiene en cuenta parámetros que son y pueden ser conocidos por los sistemas, además se buscan variables que puedan tener un impacto en la selección y clasificación de los resultados de las recomendaciones, cualquier tipo de información sobre las condiciones en las que se encuentra un usuario, consideradas relevantes para la interacción con el sistema, permitiendo realizar sugerencia personalizadas de acuerdo a su contexto físico, personal, social y de red, ver figura 5.20. Las actuales aproximaciones, en cuanto al uso de información del contexto con técnicas de recomendación tradicionales, se enmarcan en el ámbito de los contenidos multimedia musicales en escenarios fijos o móviles. En ambos casos, se obtienen con ayuda de sensores, un conjunto de variables del contexto tales como: temperatura, ruido ambiente, ubicación, luminosidad, hora, situación geográfica, entre otras. A través de estas variables, se determina el estado de ánimo o la actividad diaria que está ejecutando el usuario en un determinado periodo del día, usando para ello algoritmos de inferencia y clasificación, destacándose los métodos de lógica difusa. Según el estado de ánimo o la actividad diaria inferida, se recomienda un contenido determinado, que ha sido previamente clasificado. Dada la necesidad de variables del contexto desde diferentes fuentes, los entornos de movilidad se adecuan de mejor forma al despliegue e implementación de los S.R. híbridos, basados en contexto [27] [29].

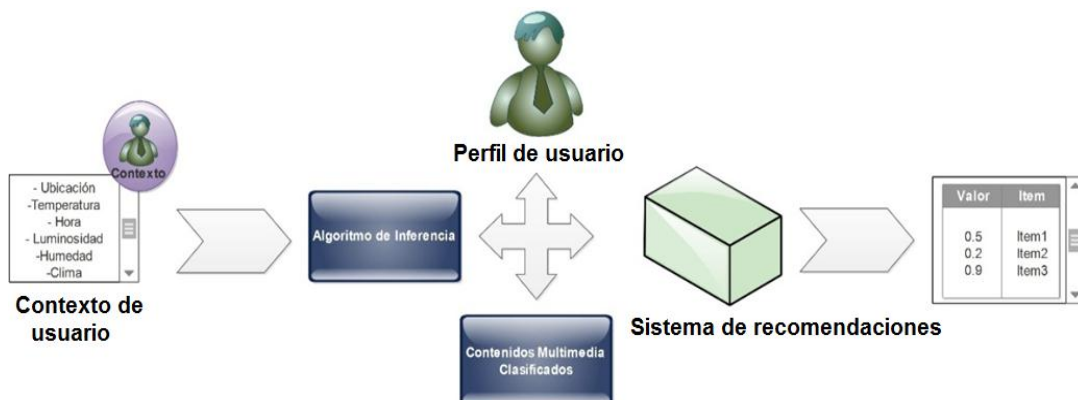


Figura 5. 20. Diagrama de secuencia de un S.R. basado en contexto. Fuente: propia.

Con base en la definición dada, en el presente trabajo se han escogido como variables del contexto de usuario las medidas fisiológicas como son: la voz, la VFC, y GSR, las cuales permiten la generación del algoritmo de inferencia de emociones y el empleo de un clasificador de contenidos que emplea 5 estados emocionales en su funcionamiento, lo cual es presentado en el numeral 5.10.

5.9 Diseño del S.R. basado en contexto

Como propuesta de un sistema de recomendaciones basado en emociones, se plantea en primer lugar el comportamiento funcional del mismo, dentro del cual se considera como variables de entrada al S.R. las medidas fisiológicas de la voz, VFC y GSR, las cuales proveen la información necesaria para la inferencia de emociones. Dicho comportamiento puede ser observado en la figura 5.21.

En “1” se envía una muestra de audio al módulo de inferencia, el cual con ayuda de la herramienta libre OpenEar permite la obtención de las propiedades de arousal y valence en “2”. Una vez obtenidas estas propiedades, se envían a la interfaz de acceso del servicio en conjunto con el perfil de usuario (“3” y “4”), la cual envía estos datos contextuales al módulo de lógica en “5”. A partir de los datos de contexto y el perfil de usuario, en “6” el módulo de lógica se encarga de solicitar el catálogo de contenidos multimedia al sistema de recomendaciones. El sistema de recomendaciones obtiene a su vez a partir del dataset la información musical de los contenidos multimedia (“7” y “8”). A partir de la información de los contenidos multimedia, del perfil y el contexto de usuario, el sistema de recomendaciones se encarga de clasificar y filtrar los contenidos que son relevantes para el usuario, generando un listado inicial de recomendaciones (“9”). Para realizar el proceso anterior, el sistema de recomendaciones hace uso de un clasificador de tipo booleano, el cual se encarga de predecir el orden de los contenidos por emoción haciendo uso del arousal y la valence asociados a la emoción de entrada, además de tener en cuenta la posible valoración que el usuario haya dado al contenido multimedia en pasadas interacciones. El listado inicial de recomendaciones es enviado al módulo de lógica en “10” y presentado en la pantalla del dispositivo de acceso en “11”.

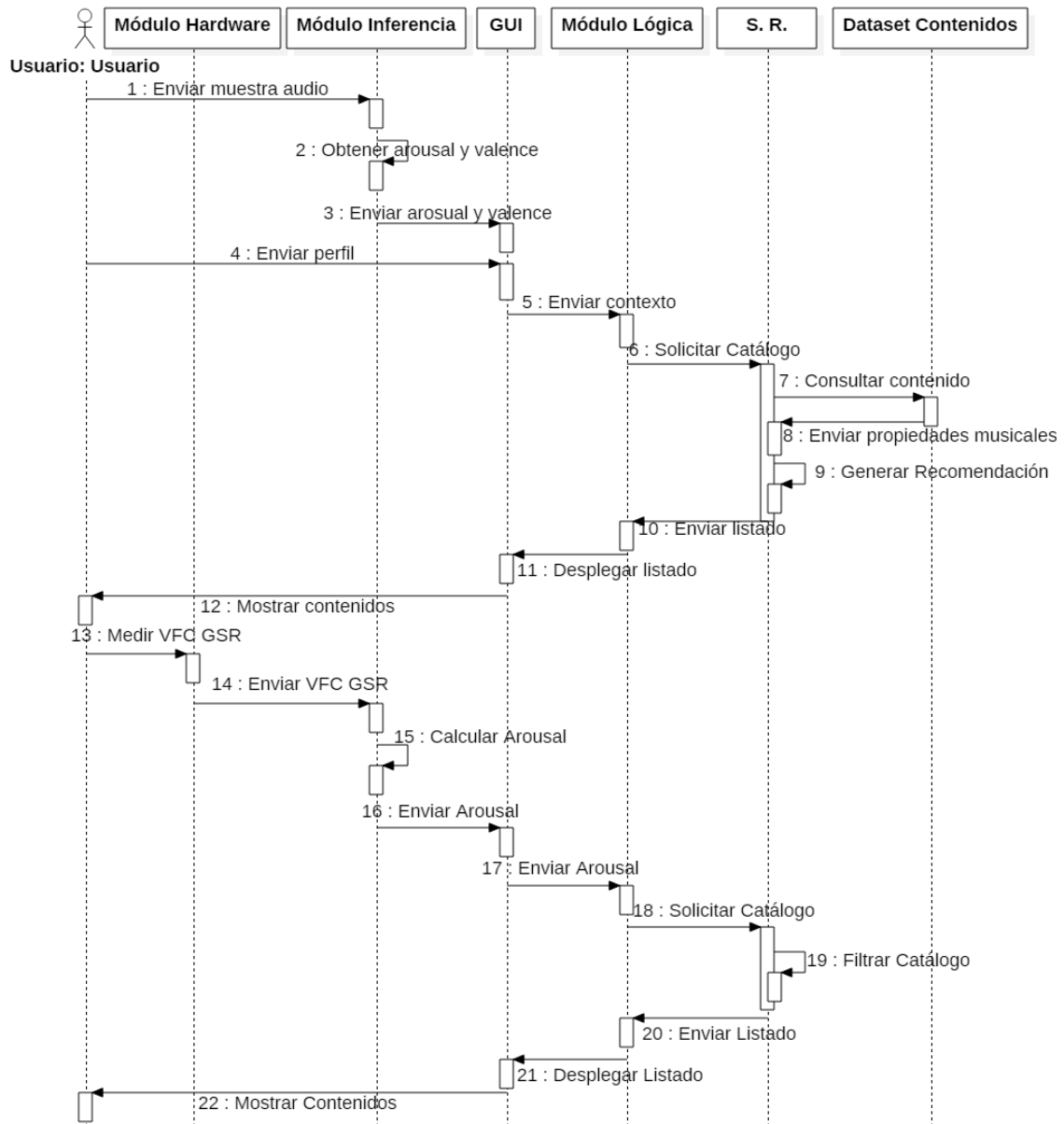


Figura 5. 21. Estructura funcional el S.R. basado en contexto. Fuente: propia.

A medida que el usuario interactúa con el catálogo de contenidos multimedia, se realiza un seguimiento sobre sus variables fisiológicas. De este modo en “13” mediante el módulo hardware se obtienen las variables de VFC y GSR. Mediante estas variables, el módulo de inferencia estima en “15” el valor de arousal de la emoción de entrada, calculando para ello el estrés mental del usuario (SI). El nuevo arousal calculado es enviado en “16” a la interfaz de usuario y en “17” al módulo de lógica, el cual solicita el nuevo catálogo de contenidos multimedia al sistema de recomendaciones en “18”. En “19” el sistema de recomendaciones a partir del nuevo arousal se encarga de filtrar el listado de contenidos previamente generado, obteniendo así un nuevo catálogo de contenidos. Finalmente este nuevo catálogo de contenidos es desplegado en la pantalla del dispositivo de acceso en “20” y “21”.

Un resumen explicativo, acerca del diseño del S.R. desarrollado se puede ver en la figura 5.22.



Figura 5. 22. Diseño del S.R. basado en contexto. Fuente: propia.

5.9.1 Diseño del clasificador bayesiano

El clasificador diseñado en este trabajo forma parte del S.R. clásico. Este es del tipo booleano como se explicó en la sección 4.3.1.1 y está basado en el algoritmo de Naive Bayes. El clasificador tiene por objetivo ordenar el contenido relevante al estado emocional y con base en esto inferir los posibles gustos musicales del usuario al interactuar con el sistema. En la tabla 5.4 se presenta un ejemplo de un clasificador bayesiano propuesto, en donde se observa 5 columnas a saber: título del video, nivel de arousal, tiempo, valence y calificación.

Tabla 5. 4. Ejemplo de clasificador bayesiano con valores booleanos. Fuente: propia.

VIDEO	A	T	V	C
<i>Bulls on Parade</i>	1	1	1	0
<i>Bombtrack</i>	1	1	1	1
<i>Freedom</i>	0	0	1	1
<i>Settle For Nothing</i>	1	0	0	0
<i>Ellens Gesang III (¡AVE MARIA!)</i>	0	1	0	?

Donde las letras significan A=*arousal*, T=*tiempo*, V=*valence* y C=*calificación*.

Para obtener las características del contenido multimedia, se consultan los parámetros musicales al dataset de video, conformado en el capítulo 2 del presente documento. Dado que los valores de las variables utilizadas como son el arousal,

el tempo y la valence, cuentan con diferentes rangos, se procede a normalizar los parámetros de arousal y valence entre -1 y 1, mientras que en el caso del tempo la normalización se realiza entre 0 y 1 (ver ecuaciones 5.10 (a), (b) y (c)).

$$arousal_{Normalizado} = (arousal - 0.5) * 2 \quad (a)$$

$$valence_{Normalizado} = (valence - 0.5) * 2 \quad (b)$$

$$tempo_{Normalizado} = \frac{tempo - 20}{240 - 20} \quad (c)$$

Ecuación 5. 10. Fórmulas para normalizar arousal, valence y tempo.

Donde:

arousal: es el valor de la propiedad arousal obtenida con la API de Echonest.

valence: es el valor de la propiedad valence obtenida con la API de echonest.

tempo: es el valor de la propiedad tempo obtenida con la API de echonest.

Una vez normalizados los valores, se establece el rango para cada parámetro musical, en el cual el contenido multimedia puede tomar el valor de 0 o 1. Así, si los valores de arousal y la valence son mayores que 0, al video se le asigna en estos parámetros un “1” o de lo contrario un “0”. En el caso del tempo, si el valor normalizado es mayor a 0.5, al video se le asigna en este parámetro un “1” o de lo contrario un “0”, (ver Tabla 5.3). Estos valores conforman los atributos de entrada de un clasificador booleano de Naive Bayes, que fue implementado en Java usando la API de la herramienta de análisis de datos weka²¹. Un ejemplo de uso básico de weka es presentado en el anexo E de la presente monografía.

Un ejemplo para la obtención de los valores booleanos se muestra continuación en la tabla 5.5 y tabla 5.6, donde se tiene para una pista los siguientes valores de arousal, valence y tempo como se aprecia a continuación.

Tabla 5. 5. Ejemplo de normalización con un contenido origina. Fuente: propia.

NOMBRE	AROUSAL	TEMPO	VALENCE
<i>Freedom</i>	0,491285	72,064	0,558701

- El *arousal* normalizado es $(0.491285-0.5)*2 = -0,01743$
- En el tempo tenemos que $(72.064-20)/220 = 0,236654545$
- En la *valence* se plantea $(0.558701-0.5)*2 = 0,117402$

Al tener en cuenta los rangos establecidos para determinar el valor booleano de cada característica, se tiene que el arousal al presentar un valor de -0.01743 y como este tiene signo negativo, el valor booleano que se le asigna es 0, para el tempo se tiene el valor de 0.2366 como este es menor a 0.5, se le asigna a la variable de tempo el valor de 0, para el caso de la valence como el valor normalizado es de

²¹ Plataforma software para minería de datos. Página web oficial de weka: <http://www.cs.waikato.ac.nz/~ml/weka/>.

0.1174 y como es mayor a 0, el valor booleano que se le asigna es de 1. Lo anterior puede ser visto a continuación:

Tabla 5. 6. Ejemplo de normalización con un contenido normalizado. Fuente: propia.

NOMBRE	AROUSAL	TEMPO	VALENCE
Freedom	0	0	1

Luego del cálculo de la lista de valores booleanos, el clasificador predice la posible valoración que el usuario daría el contenido (columna C de la Tabla 5.4) teniendo como base calificaciones anteriormente realizadas, generando un listado de videos ordenados según la hipótesis más probable. En la figura 5.23, se presenta en resumen el diagrama de clasificador bayesiano, que tiene en cuenta el contexto de usuario, donde el S.R. clásico hace parte del S.R. contextual.

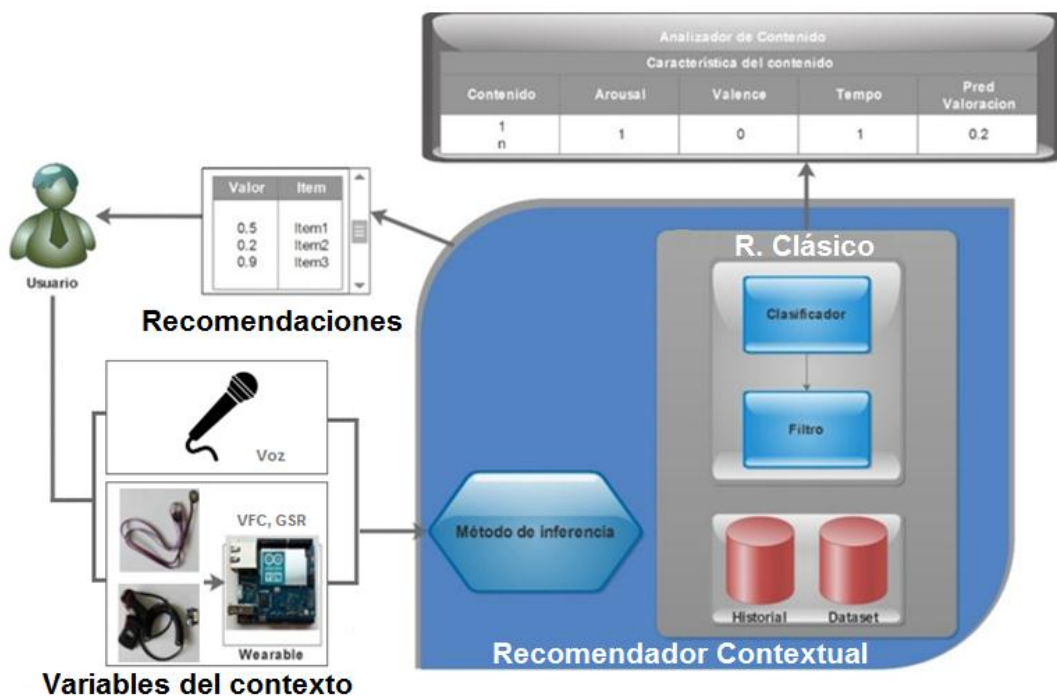


Figura 5. 23. Diagrama final del clasificador bayesiano. Fuente: propia.

5.9.2 Servicio de VoD basado en emociones

La arquitectura del servicio de VoD propuesto en el presente trabajo comprende cómo se ha venido enfatizando a lo largo del documento, los siguientes bloques funcionales: modulo hardware de Arduino, el servidor de contexto, servidor de lógica, servidor de procesamiento, el servidor de contenidos y la base de datos (ver figura 5.24).

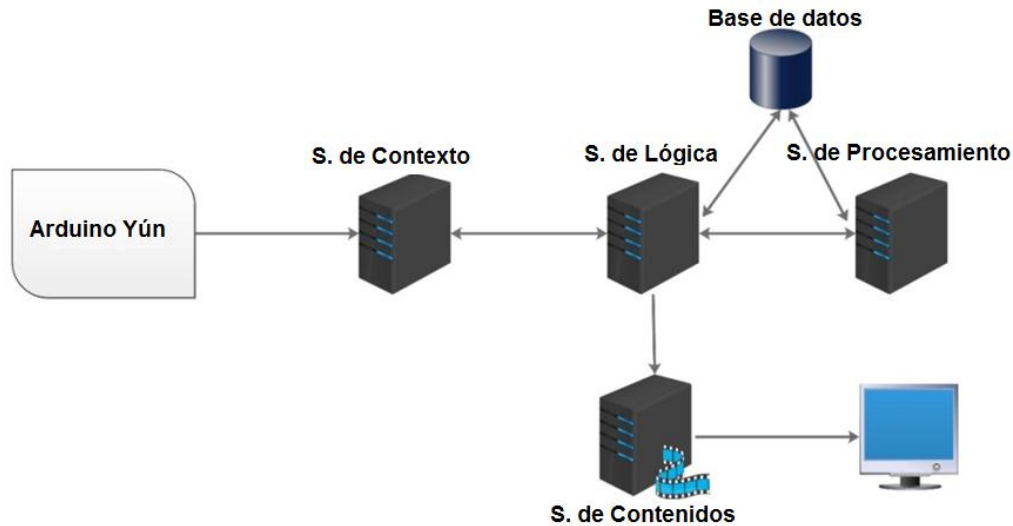


Figura 5. 24. Arquitectura general del servicio de VoD. Fuente: propia.

El módulo Arduino proporciona un servicio web RESTful, este se encarga de recibir las medidas capturadas con ayuda de los sensores, estas son enviadas al servidor de contexto desarrollado con Flask de Python, donde se realiza el cálculo del SI, arousal y valence para el establecimiento de la emoción (ver anexo C). Esta información es consultada por el servidor de lógica el cual recibe dicha información haciendo uso de JSON, en este módulo se efectúa la validación del usuario y se realiza la solicitud del catálogo de contenidos al servidor de procesamiento. La base de datos a la que se realizan las consultas contiene las tablas de perfil de usuario y lista de contenidos, en donde se registraran las calificaciones que el cliente actualice de cada video. Por otra parte, en el servidor de procesamiento se genera la lista de recomendaciones haciendo uso de las predicciones realizadas por weka al usuario, de acuerdo al perfil de este y teniendo en cuenta la emoción de entrada. Una vez generada la lista de recomendación es enviada empleando el formato json al servidor de lógica el cual solicita los videos vía http al servidor de contenidos para desplegarlos en la interfaz de usuario del servicio de VoD.

El servicio de video bajo demanda del presente proyecto fue generado como medio de evaluación del sistema de recomendaciones desarrollado, además los S.R. vienen incluidos en las prestaciones de los servicios de VoD. En el presente numeral se pretende mostrar al lector el diseño de la interfaz de usuario del servicio de VoD. Como primera interfaz se tiene la de validación del usuario y la segunda permitirá la visualización de los videos. En la figura 5.25 se puede observar la primera.



Figura 5. 25. Interfaz validación de usuario del servicio de VoD. Fuente: propia.

En “1” el usuario como acción inicial se valida en el sistema, para ello como primera medida son obtenidas las medidas fisiológicas a través del módulo hardware-software implementado en la sección 5.6. En “2”, al presionar el botón “Obtener valores” se comienza la captura de la voz y posteriormente de la variable VFC, las cuales permiten obtener una emoción de entrada (arousal, valence) y realizar un seguimiento emocional al usuario durante su interacción con el servicio. En “3” el usuario visualiza gráficamente la medida de la frecuencia cardíaca en tiempo real. Una vez obtenidos los valores, el usuario puede iniciar sesión, si no se encuentra registrado tendrá la opción de hacerlo. Al realizar dicha acción de validación, como segundo paso se le muestra la interfaz presentada en la figura 5.26.



Figura 5. 26. Interfaz principal de usuario del servicio de VoD. Fuente: propia

En “1” el usuario visualiza los videos para ello cuenta con el panel de control multimedia para el mismo y la opción de valorar el contenido, en esta el usuario debe presionar en el botón actualizar y automáticamente queda registrada la calificación o valoración dada a cada video. En “2” el usuario ve el monitoreo continuo de las variables fisiológicas de VFC, GSR, además se muestra el nivel de estrés y la emoción de usuario que se presente en el momento, también al igual

que con la primera interfaz, el gráfico de HR vs tiempo se puede visualizar. En “3” es mostrado al consumidor la lista de contenidos recomendados, en donde la emoción detectada por el sistema es la pieza clave para el despliegue de los contenidos relevantes al usuario y la medida GSR se encarga de reordenarlos de acuerdo al estímulo o aumento del arousal en la persona.

Capítulo 6

6. Evaluación y Pruebas

6.1 Introducción

En el presente capítulo se describen las diferentes pruebas realizadas a los principales módulos de la arquitectura basada en contexto, destacándose principalmente las pruebas realizadas sobre el escenario de *streaming* adaptativo DASH y sobre el componente de sistema de recomendaciones basado en contexto. Adicionalmente se realizaron otras pruebas sobre los diferentes componentes de la arquitectura, como es el caso de las realizadas sobre servidor de contexto, el servidor de procesamiento y del módulo hardware-software de captura de variables de contexto.

En cuanto a la evaluación del escenario de *streaming* adaptativo DASH, se realizaron pruebas de consumo de memoria RAM y consumo de CPU sobre la herramienta de codificación DASH WebM converter, la cual forma parte del entorno de *streaming* adaptativo propuesto. Así mismo, se evaluó el comportamiento del escenario de *streaming* adaptativo ante diferentes emulaciones de ancho de banda, con el objetivo de verificar la funcionalidad del entorno de *streaming* adaptativo DASH. En lo que respecta a la evaluación realizada sobre el sistema de recomendaciones propuesto, se hicieron pruebas de precisión sobre el sistema de recomendaciones clásico que forma parte del núcleo del sistema de recomendaciones basado en contexto. Así mismo, con el objetivo de evaluar el listado de recomendaciones generadas por el recomendador basado en contexto, se hizo uso de las métricas de: distancia Euclidiana, distancia Manhattan, similitud del coseno, correlación de Pearson y coeficiente OLS, con el fin de obtener la similitud del listado de contenidos generado con respecto al estado de ánimo de entrada al sistema.

En lo referente a los otros componentes de la arquitectura basada en contexto, se realizaron pruebas de carga y de estrés sobre el módulo hardware-software de captura de variables del contexto, el servidor de contexto y el servidor de procesamiento. A partir de estas pruebas, se buscó comprobar la pertinencia de las tecnologías usadas para la implementación de estos componentes software del servicio de VoD. El resto del capítulo está organizado de la siguiente forma: en la sección 6.1 se presentan las pruebas realizadas sobre el entorno de *streaming* adaptativo DASH, lo cual incluye las pruebas de consumo de memoria sobre la herramienta Dash WebM Converter y las pruebas de consistencia de ancho de banda sobre el entorno de *streaming* adaptativo. En la sección 6.2 se describen las pruebas realizadas sobre el sistema de recomendaciones basado en contexto, lo cual incluye la evaluación del recomendador clásico y la comparación de este con el recomendador basado en contexto. En la sección 6.3 se presentan las pruebas de carga y estrés sobre el servidor de procesamiento, el servidor de contexto y el módulo hardware software de captura de variables del contexto.

6.2 Evaluación del entorno de streaming

En esta sección se presentan las pruebas realizadas sobre el entorno de *streaming* adaptativo DASH, de manera específica las pruebas de consumo de RAM y CPU sobre la herramienta DASH WebM Converter la cual cumple funciones de codificación dentro del escenario. Así mismo, se presenta la evaluación del comportamiento del escenario de *streaming* adaptativo, ante diferentes emulaciones de ancho de banda, con el propósito de verificar la funcionalidad del entorno DASH.

6.2.1 Prueba de consumo de memoria herramienta DASH Webm Converter

Dentro de esta sección, se presentan las pruebas de consumo de memoria RAM y porcentaje de uso de CPU, realizadas tanto a la herramienta de codificación propuesta, como a la herramienta FFMPEG usada en segundo plano por la herramienta DASH WebM Converter. Para la captura de los datos presentados en esta sección, se desarrolló una herramienta de monitoreo en el lenguaje Python, la cual permitió capturar cada 5 segundos, los valores de memoria y porcentaje de cpu de los procesos del sistema operativo asociados a la herramienta de codificación y la herramienta FFMPEG. La obtención de estos valores los realizó la herramienta de monitoreo, mediante el comando “ps aux” de Linux, el cual brinda un reporte de la cantidad de memoria RAM y el porcentaje de CPU utilizados por cada uno de los procesos activos del sistema operativo. Así mismo, se usó el

comando `grep` y el lenguaje de programación `awk` (pertenecientes al sistema operativo Linux), con el fin de filtrar la información de los procesos.

En las figuras 6.1 y 6.2 se presentan los resultados de las pruebas de porcentaje de uso de CPU y consumo de memoria RAM, realizadas sobre la herramienta FFMPEG (usada en segundo plano por DASH WebM Converter), durante la codificación de un contenido multimedia WebM a 6 tasas de bits diferentes (50,100, 200,400,600 y 800 kbps). Dado que la herramienta de codificación lanza tantos hilos de FFMPEG como tasas de bits a codificar, más un hilo independiente para el audio, en las figuras 6.1 y 6.2 se presentan el comportamiento de los 7 hilos usados en una prueba de funcionamiento de la herramienta de codificación.

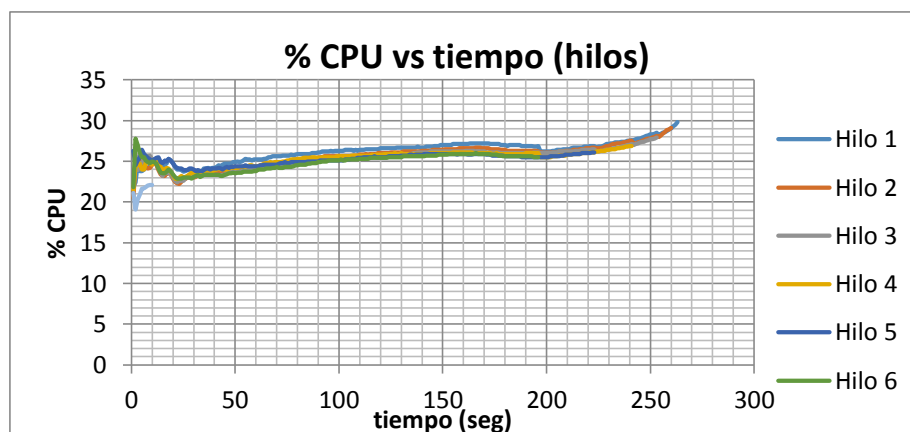


Figura 6. 1. Porcentaje de CPU vs tiempo (FFMPEG). Fuente: propia.

De acuerdo a la Figura 6.1, el hilo de audio (hilo 7) es el primero en terminar su ejecución con un tiempo cercano a los 10 segundos y es el que menos porcentaje de CPU ocupa, con valores comprendidos entre 19% y 22%. Así mismo, cabe resaltar que los hilos destinados a la codificación de video emplean un porcentaje de CPU similar durante todo el proceso, el cual oscila entre 22% y 30%.

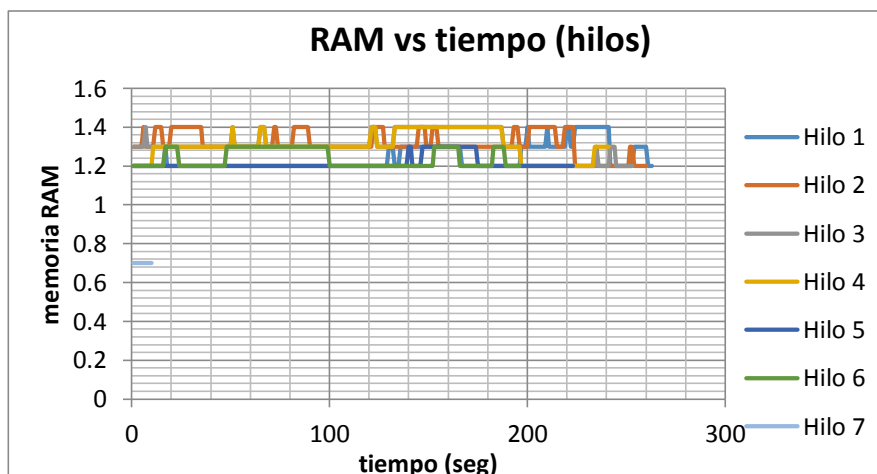


Figura 6. 2. Memoria RAM vs tiempo (FFMPEG). Fuente: propia.

Según la Figura 6.2, el hilo de audio es el primero que termina su ejecución y es el que menos cantidad de RAM consume, con un valor de 0.7 MB. Entre tanto los 6 hilos de video tienen un consumo similar de memoria que oscila entre 1.2 MB y 1.4 MB durante el proceso de codificación. Teniendo en cuenta lo anterior, en promedio el consumo de cada hilo de video en la herramienta FFMPEG es de 1,2 MB.

En las Figuras 6.3 y 6.4 se muestran los resultados de las pruebas de porcentaje de uso de CPU y consumo de memoria RAM, realizadas sobre la herramienta DASH WebM Converter, durante la codificación de un contenido multimedia WebM a 6 tasas de bits diferentes. De acuerdo a la Figura 6.3, el porcentaje de uso de CPU de la herramienta de codificación presenta un comportamiento creciente entre los 0 y 100 segundos, alcanzando un valor de 20 %. Por su parte entre los 100 y 200 segundos, el porcentaje de uso de CPU se mantiene alrededor del 21 %. Finalmente, después de los 200 segundos, el porcentaje de CPU tiene un comportamiento creciente alcanzando un valor máximo de 26%.

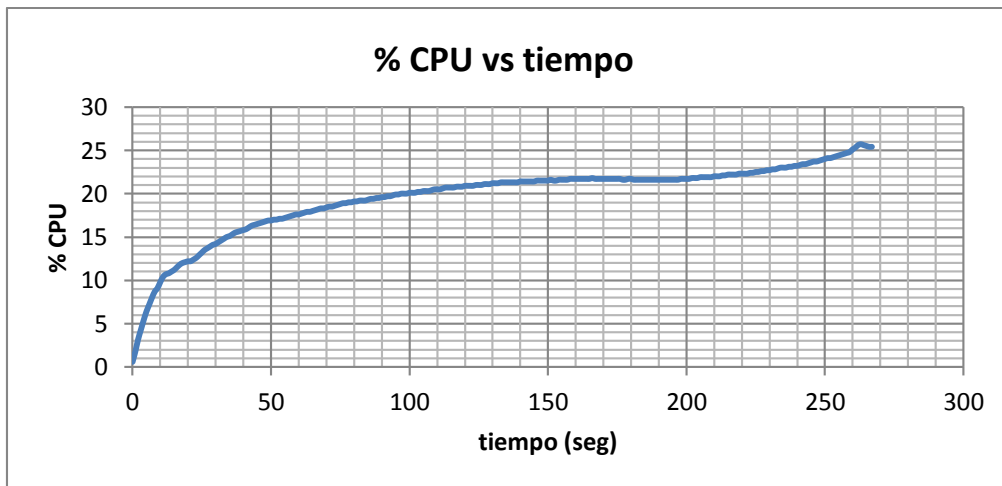


Figura 6. 3. Porcentaje de CPU vs tiempo DASH WebM Converter. Fuente: propia.

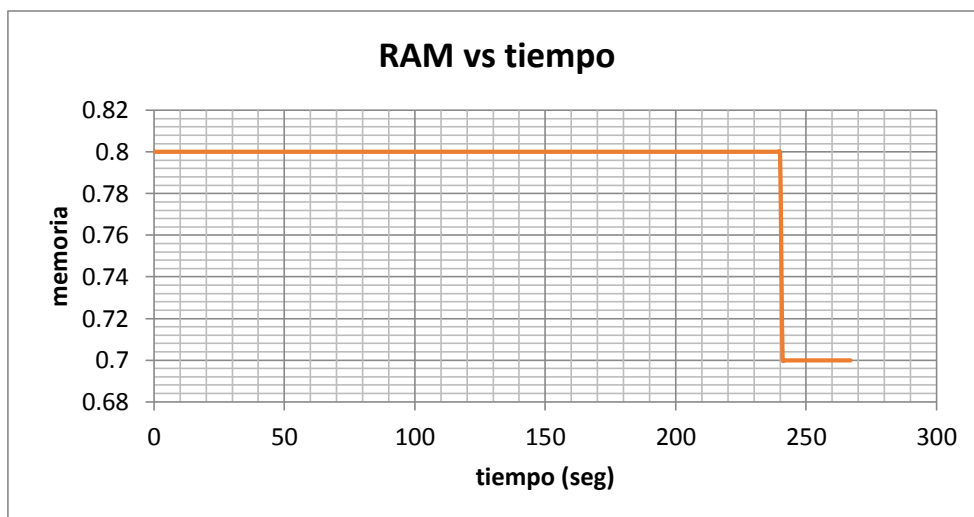


Figura 6. 4. RAM vs tiempo DASH WebM Converter. Fuente: propia.

De acuerdo a la Figura 6.4, el consumo de memoria de la herramienta de codificación se mantiene casi constante hasta los 240 segundos, con un valor de 0.8 MB. Después de los 240 segundos, el valor de consumo de memoria disminuye a 0.7 MB hasta el final del proceso de codificación.

De acuerdo a las mediciones de consumo de memoria RAM realizadas sobre la herramienta FFmpeg, para un contenido multimedia con 6 tasas de bits a codificar, la cantidad de memoria RAM asignada a cada hilo de video es en promedio de 0.2 MB. Lo anterior permite inferir que el hecho de usar procesamiento multihilo no representa un incremento considerable en la cantidad de memoria RAM, con respecto a las ventajas obtenidas de optimización de tiempo.

6.2.2 Pruebas de ancho de banda sobre el escenario de streaming

Para la realización de este tipo de prueba se utilizó la herramienta DummyNet²², la cual consta de un emulador de red en vivo. Entre las funciones principales de esta herramienta están: emulación de ancho de banda, pérdida de paquetes, retardo, entre otras funcionalidades. Esta herramienta de emulación es multiplataforma y puede operar sobre sistemas operativos como: OSX, Linux, Windows y FreeBSD.

La prueba de emulación de ancho de banda fueron realizadas sobre el sistema operativo Windows 7, para lo cual fue necesario utilizar la arquitectura de 32 bits puesto que se requiere el uso de un controlador sin firma digital para el adaptador de red. Durante la instalación, es posible encontrar algunos inconvenientes debido a permisos especiales en el sistema operativo, por lo cual es conveniente verificar que los *scripts* necesarios se encuentren en el directorio del sistema C:/Windows/System32/: "cygwin1.dll" e "ipfw". Para facilitar la prueba de consumo de ancho de banda se desarrolló el *script* test_dummysnet.bat (ver figura 6.5), el cual debe ejecutarse con permisos de administrador. El *script* primero se encarga de limpiar cualquier configuración de red hecha por la herramienta dummysnet, después configura los parámetros para ajustar el ancho de banda de la red, a continuación el *script* se detiene mientras se realiza la prueba. Una vez terminada la prueba se presiona cualquier tecla y el *script* vuelve a deshacer la configuración de dummysnet.

²² Documentación disponible en: <http://info.iet.unipi.it/~luigi/dummysnet/>

```

test_dummysnet.bat
1 @echo on
2 @set CYGWIN=nodosfilewarning
3 @ipfw -q flush
4 @ipfw -q pipe flush
5 @echo #####
6 @echo ## Setting up ##
7 @echo #####
8 ipfw pipe 3 config bw 200kbit/s mask all
9 ipfw add pipe 3 ip from any to any
10 ipfw pipe show
11
12 @echo ""
13 @echo "Network Tweaking Done, Start Testing."
14 @echo ""
15 @echo "Press Enter when testing is done, to restore original Network settings."
16 pause
17
18 @echo #####
19 @echo ## Cleaning up ##
20 @echo #####
21 @ipfw -q flush
22 @ipfw -q pipe flush
23 pause

```

Figura 6. 5. Script para emulación de ancho de banda. Fuente: propia.

La prueba de consumo de ancho de banda se realizó sobre una versión del servicio de VoD DASH disponible en: http://dash-unicauca.org/user_demo2 (ver la figura 6.6). Es importante mencionar, que para ajustar el ancho de banda de la red, debe digitarse el valor del parámetro bw en la línea 8 del *script* test_dummysnet (ver figura 6.5).

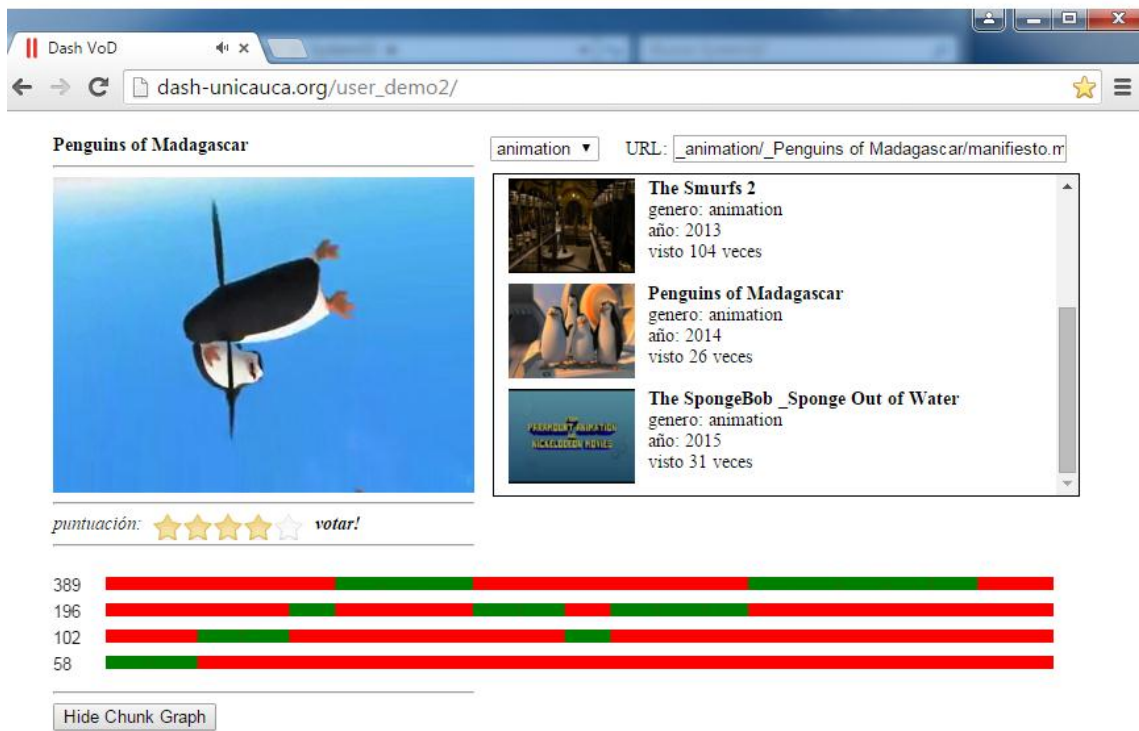


Figura 6. 6. Servicio de VoD en la nube. Fuente: propia.

Los valores de ancho de banda utilizados para la realización de la prueba fueron: 20, 100, 200, 400, 600, 800 y 1500 kbit/s, teniendo en cuenta las diferentes tasas de codificación de los contenidos multimedia utilizados. Es importante mencionar que el valor real del ancho de banda de la red, ofrecida por el operador es de 1 Mbit/s. Los datos arrojados por la realización de la prueba son presentados en la figura 6.7. En la parte izquierda de la gráfica se muestra el monitor de recursos de Windows, donde se indica el ancho de banda delimitado por dummynet, y en la parte derecha aparece el sistema de medición de ancho de banda, desarrollado sobre el cliente web del entorno de *streaming* adaptativo DASH.

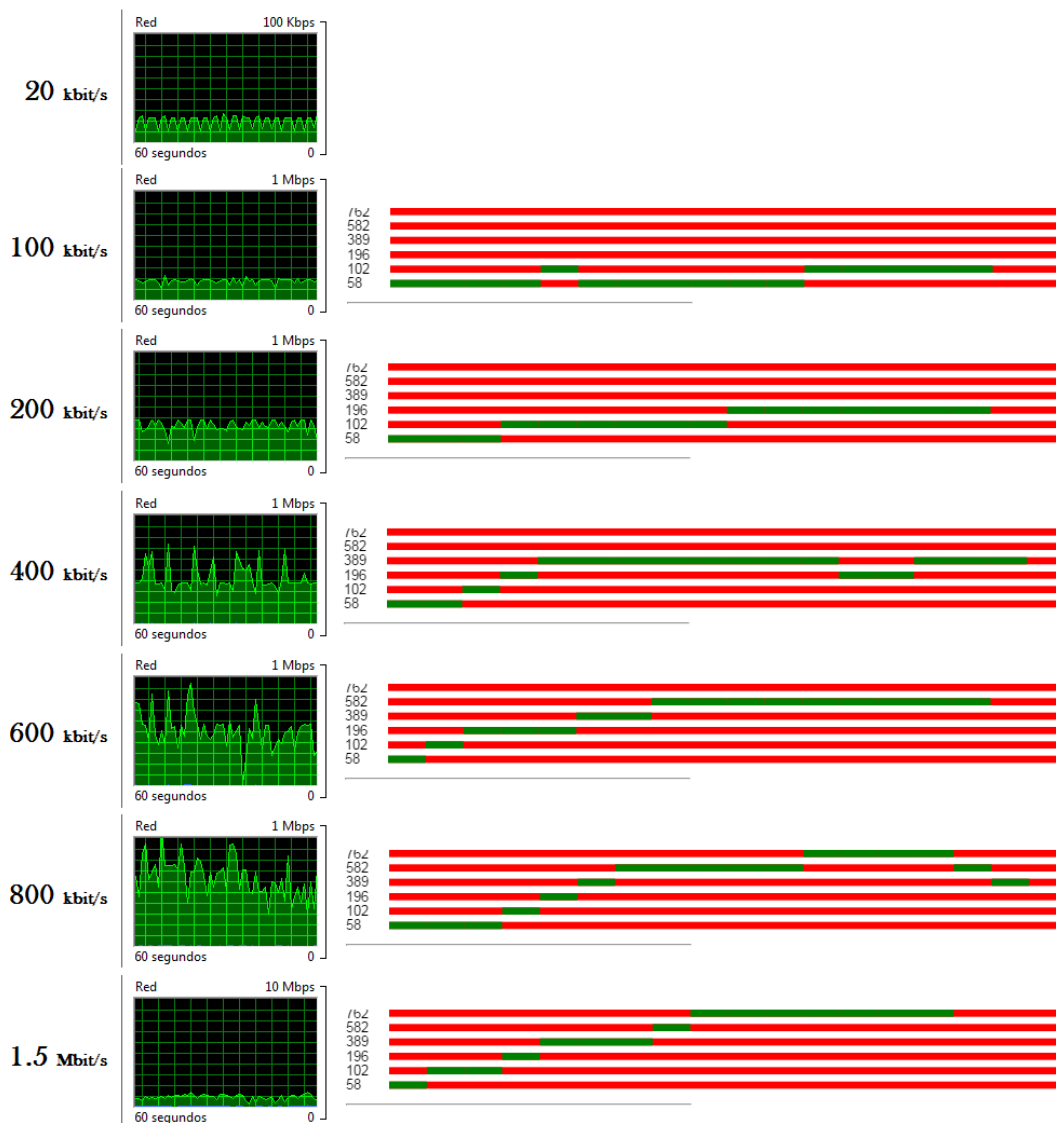


Figura 6. 7. Emulación de ancho de banda. Fuente:propia.

Como puede observarse en la figura 6.7, la herramienta dummynet cumple satisfactoriamente su función de emular el ancho de banda de la red, según el valor ingresado en el *script*. Así mismo, según los resultados encontrados tras la realización de la prueba, se puede concluir que el servicio de VoD sigue los cambios

emulados en el ancho de banda, de tal manera que el entorno de *streaming* adaptativo cumple de manera satisfactoria los requisitos especificados por el estándar DASH.

6.3 Evaluación del S.R contextual

En esta sección se presentan las pruebas realizadas sobre el sistema de recomendaciones basado en contexto, lo cual incluye la evaluación de precisión sobre el sistema de recomendaciones clásico o recomendador bayesiano que forma parte del núcleo funcional del sistema de recomendaciones basado en contexto. Así mismo, se incluye la evaluación de las recomendaciones generadas por el sistema de recomendaciones basado en contexto, comparando la similitud de las mismas con la emoción de entrada del usuario en cuestión. Esta comparación fue realizada a partir de las características musicales de arousal y valence.

6.3.1 Evaluación S.R. clásico

Las pruebas de evaluación del S.R. clásico, se basaron en el análisis y descripción de los porcentajes de predicción presentados por el recomendador bayesiano implementado en el servidor de procesamiento, haciendo uso de la API de weka para Java, el servidor web Spark y la base de datos MySQL. Así, a partir de lo anterior, se realizaron pruebas al S.R. clásico con usuarios registrados en el servicio de VoD, que han y no valorado el contenido multimedia del servicio. En esta sección se presenta en primera instancia el reporte del análisis de las predicciones generadas por el S.R. clásico, el cual es proporcionado por la herramienta de minería de datos weka. Así mismo se presenta la matriz de confusión que modela el desempeño del algoritmo de clasificación usado en el S.R. clásico.

6.3.1.1 Resumen reporte de predicciones

En la figura 6.8, se presenta el reporte del análisis de las predicciones que provee el clasificador Naive Bayes al S.R. clásico, a través de la herramienta de minería de datos weka. Los resultados obtenidos a partir del clasificador bayesiano ayudan a filtrar y clasificar de una manera más adecuada las recomendaciones que el S.R. clásico provee a usuarios, a través del servicio de VoD. El clasificador bayesiano hace uso del gestor base de datos MySQL para consultar la base de datos generada a partir del dataset de contenidos presentado en el capítulo 3, la

cual contiene los datos de los usuarios y de los 200 contenidos asociados a las 5 emociones mencionadas en la sección 2.6.1.

```

*** RESUMEN PREDICCIONES ***
Correctly Classified Instances      118          67.8161 %
Incorrectly Classified Instances    56          32.1839 %
Kappa statistic                     0
Mean absolute error                 0.4366
Root mean squared error             0.4673
Relative absolute error             99.8608 %
Root relative squared error         100.0187 %
Total Number of Instances          174
Ignored Class Unknown Instances    26
Número de predicciones: 200
*****

```

Figura 6. 8. Resumen de predicciones del clasificador bayesiano. Fuente: propia.

De acuerdo a la figura 6.8, el clasificador realizó correctamente 118 clasificaciones de contenidos con un porcentaje de predicción de 67.81%, 56 clasificaciones incorrectas con un porcentaje de 32.19%, para tener un total de 174 contenidos o instancias consideradas para la clasificación, que han sido valoradas por los usuarios. Además de 26 instancias que no han sido valoradas por usuarios, para tener un total de 200 instancias a predecir.

Así mismo, se observa que el error medio absoluto (MAE), el cual es una medida de la precisión del listado de sugerencias, presenta para el clasificador bayesiano un porcentaje de 43.66% de incertidumbre en las recomendaciones, lo que quiere decir que 4 de cada 10 recomendaciones pudieron ser erróneas. De la misma manera, el error cuadrático medio (RMSE), el cual es un indicador de error medio de las predicciones numéricas que indica el grado de desviación de las predicciones, para el caso del clasificador bayesiano tuvo un valor de 46.73 %. Por su parte, el error relativo absoluto (RAE), el cual es un indicador relativo de la precisión simple que mide la diferencia [0,100%] entre el valor absoluto y el valor verdadero de cada recomendación, arroja para el clasificador un valor adecuado de 99.86 % en las recomendaciones (mientras mayor sea este valor mayor grado de similitud existe en las predicciones de las recomendaciones). En lo que respecta a la raíz cuadrada del error relativo (RRSE), el cual es un indicador promedio de los valores de predicción reales, para el caso del clasificador bayesiano es un 100,01% acertado en la predicción de cada una de las recomendaciones que el S.R. provee.

Es importante mencionar que de las métricas evaluadas, la MAE y RMSE son las más relevantes en cuanto a la validez del clasificador bayesiano y por ende del recomendador clásico, puesto que indican el grado de precisión y desviación de las recomendaciones. Así, en este caso, luego de analizar los valores de estas dos métricas para el clasificador diseñado, se concluye que este presenta un conjunto adecuado de recomendaciones que permiten configurar un aceptable sistema de

recomendaciones clásico, el cual pretende ser mejorado haciendo uso de variables del contexto de usuario.

6.3.1.2 Matriz de confusión

La matriz de confusión es una herramienta que fue usada para evaluar las predicciones del clasificador bayesiano, además de proveer información acerca de las clasificaciones actuales y/o pronósticos generados por el sistema de clasificación. En la figura 6.9, se presenta la matriz de confusión obtenida mediante weka, la cual describe el desempeño del clasificador bayesiano utilizado por el S.R. clásico.

```

*** MATRIZ DE CONFUSIÓN ***
0.0|56.0|
0.0|118.0|
*****

```

Figura 6. 9. Matriz de confusión que modela el clasificador bayesiano. Fuente: propia.

Para mayor comprensión de los datos arrojados por la matriz de confusión del clasificador bayesiano, se presenta en la tabla 6.1 un ejemplo sintetizado con los datos obtenidos para este clasificador. Para ello se identifican con las letras A, B, C y D, cada uno de los datos estimados en la figura 6.9.

Tabla 6. 1. Ejemplo matriz de confusión. Fuente: propia.

		PREDICCIÓN	
		Negativa	Positiva
INSTANCIA	Negativa	A = 0	B = 56
	Positiva	C = 0	D = 118

El análisis de la tabla 6.1 puede ser entendido mejor de la siguiente forma:

- A es el número de predicciones correctas, para una instancia negativa, en este ejemplo 0.
- B es el número de predicciones incorrectas, para una instancia positiva, en este ejemplo 56.
- C es el número de predicciones incorrectas, para una instancia negativa, en este ejemplo 0.
- D es el número de predicciones correctas, para una instancia positiva, en este ejemplo 118.

Así, a partir del ejemplo anterior, se puede concluir que el clasificador bayesiano presenta un total de 118 predicciones correctas, lo cual corresponde a un porcentaje de 59% de certeza en la recomendación de contenidos. Por su parte, el clasificador presenta un porcentaje de incertidumbre de 28% en las

recomendaciones. De esta manera, se puede concluir que este clasificador presenta un adecuado número de predicciones positivas por sobre las erróneas, lo que contribuye a la configuración de un aceptable sistema de recomendaciones, que busco ser mejorado haciendo uso de variables del contexto de usuario.

6.3.2 Evaluación S.R. contexto

Para evaluar el S.R. basado en contexto se realizaron pruebas haciendo uso de métodos de comparación lineal de similitud con respecto a la emoción de entrada de usuario, confrontando las recomendaciones generadas por el S.R. clásico y el S.R. basado en contexto. Cabe recordar que la obtención de la emoción de entrada de usuario se hizo a partir de los parámetros musicales de arousal y valence, los cuales fueron usados para confrontar los listados de sugerencias de los dos recomendadores. Así, en primera instancia se presentan los métodos de comparación lineal que se consideraron en la comparación de similitud y posteriormente los resultados obtenidos mediante el uso de estos métodos.

6.3.2.1 Métodos de comparación lineal

Uno de los retos en los S.R. basados en contexto es la forma de evaluación de los mismos, por lo cual en el presente trabajo y dada la relación existente entre las características de los contenidos multimedia musicales y las emociones, se propuso el uso de métodos lineales de similitud para comparar el listado de recomendaciones generadas por el S.R. clásico y el S.R. basado en contexto, tomando en cuenta los parámetros arousal y valence de la emoción de entrada y de cada uno de los contenidos musicales sugeridos por los dos sistemas de recomendaciones. A continuación se presentan los métodos de comparación considerados, dentro de los que se encuentran: distancia Euclidiana, distancia Manhattan, similitud del Coseno, coeficiente de correlación de Pearson y coeficiente OLS (Ordinary least squares) [84] [85].

6.3.2.1.1 Distancia Euclidiana

Es la métrica de similitud más básica, cuyo propósito es medir la distancia entre dos puntos en el espacio, lo cual corresponde a la longitud de una línea recta trazada entre los dos puntos [84]. La ecuación 6.1, muestra la formula usada para la obtención de la distancia euclidiana, la cual se define como la sumatoria de las diferencias o distancias entre dos puntos. Así, mientras más tienda la distancia euclidiana a 0, mayor va a ser la similitud entre los puntos. De manera contraria, mientras más tienda la distancia euclidiana a 1, mayor será la diferencia entre los puntos en el espacio.

$$D_{uv} = \sqrt{\sum_{i=1}^n |a_{ui} - a_{vi}|^2}$$

Ecuación 6. 1. Distancia Euclídiana.

Donde:

a_{ui} : valor del punto a_i sobre la componente u .

a_{vi} : valor del punto a_i sobre la componente v .

6.3.2.1.2 Distancia Manhattan

Esta métrica se encarga de medir la suma de las diferencias absolutas en cada uno de sus componentes para 2 vectores en el espacio [84]. Esta medida es aún más precisa que la distancia euclídiana para vectores de 2 o más dimensiones. A continuación, se presenta la ecuación 6.2 que permite el cálculo matemático de la métrica, la cual es definida como la sumatoria de las diferencias absolutas de 2 vectores en el espacio. Tal como se aprecia en la ecuación 6.2., al igual que la distancia euclídiana, mientras más tienda el valor de esta métrica a 0, mayor será la similitud entre los vectores en el espacio.

$$D_{uv} = \sum_{i=1}^n |a_{ui} - a_{vi}|$$

Ecuación 6. 2. Distancia Manhattan.

Donde:

a_{ui} : valor del vector a_i sobre la componente u .

a_{vi} : valor del vector a_i sobre la componente v .

6.3.2.1.3 Similitud del Coseno

Es la medida de similitud del producto escalar de dos vectores, la cual se encarga de medir el ángulo espacial entre los vectores y evaluar la dispersión de los mismos. Aunque no es considerada una métrica, debido a que no cumple la desigualdad triangular, es muy útil en los S.R. para medir la similitud de dos vectores [84]. La ecuación 6.3, representa la fórmula matemática usada para el cálculo de la similitud del coseno de los vectores u y v .

$$\cos(u, v) = \frac{a_u \cdot a_v}{\|a_u\| \|a_v\|} = \frac{\sum_{i=1}^n a_{ui} \cdot a_{vi}}{\sqrt{\sum_{i=0}^n a_{ui}^2} \sqrt{\sum_{i=0}^n a_{vi}^2}}$$

Ecuación 6. 3. Similitud del Coseno.

Donde:

a_{ui} : valor del vector a_i en su componente u .

a_{vi} : valor del vector a_i en su componente v .

Cabe mencionar que, aunque la medida de similitud oscila en la escala de $[-1,1]$, se puede utilizar para parametrizar su escala de $[0,1]$ la siguiente ecuación 6.4:

$$1 - \frac{\cos^{-1}(\cos(u, v))}{\pi}$$

Ecuación 6. 4. Formula parametrización, similitud del coseno.

Donde:

$\cos(u,v)$: valor de la similitud del coseno.

En conclusión, esta medida indica que dos vectores son similares si el valor de la similitud del coseno tiende más a 1, cuya interpretación se puede mencionar que los vectores son simétricos mientras mayor sea su valor de 0 a 1.

6.3.2.1.4 Coeficiente de correlación de Pearson

Mide el grado de covariación entre distintas variables relacionadas linealmente, a diferencia de la covarianza este método es independiente de la escala de medida [84]. Este coeficiente mide la similitud en una escala de $[-1,1]$, donde 1 es similar y -1 disimiles. La ecuación 6.5, representa el coeficiente de correlación de Pearson de las variables u y v . Este método se define como el cociente entre la covarianza y las desviaciones estándar de cada una de las variables.

$$\text{CorrP}(u, v) = \frac{\sum_{i=1}^n (a_{ui} - \bar{a}_u) \cdot (a_{vi} - \bar{a}_v)}{\sqrt{\sum_{i=1}^n (a_{ui} - \bar{a}_u)^2} \sqrt{\sum_{i=1}^n (a_{vi} - \bar{a}_v)^2}}$$

Ecuación 6. 5. Coeficiente de Correlación de Pearson.

Donde:

a_{ui} : valor del vector a_i en su componente u .

a_{vi} : valor del vector a_i en su componente v .

Si se desea establecer su escala entre $[0,1]$, también se puede hacer uso de la ecuación 6.4. La interpretación del coeficiente es similar a la de la similitud del coseno; mientras mayor sea el valor del método más similares son las variables.

6.3.2.1.5 Coeficiente mínimos cuadrados ordinarios (Ordinary Least Squares, OLS)

Es un método no paramétrico de medición de variables o modelado lineal, el cual se encarga de indicar si dos vectores son asociados entre sí. Este método se basa

en el ajuste de una línea recta a una muestra de datos, minimizando la suma de los cuadrados de las desviaciones [85]. A continuación, se presenta la ecuación 6.6 que modela el coeficiente OLS. Este método se define como el cociente entre el producto de las medias y el cuadrado de una de las variables o vector.

$$OLSCoef(u, v) = \frac{\sum_{i=0}^n a_{ui} \cdot a_{vi}}{\sum_{i=0}^n a_{ui}^2}$$

Ecuación 6. 6. Coeficiente OLS.

Donde:

a_{ui} : valor del vector a_i en su componente u .

a_{vi} : valor del vector a_i en su componente v .

Para evaluar el S.R. que considera el contexto propuesto en el presente trabajo de grado, se optó por realizar pruebas utilizando los métodos de comparación mencionados en la sección 6.2.2.1, con base en las predicciones de contenidos multimedia que provee el S.R. basado en contexto en comparación al S.R. clásico.

6.3.2.2 Aplicación de las métricas de similitud

La evaluación del S.R. basado en contexto, se realizó mediante el cálculo de la similitud entre los valores de arousal y valence de la emoción de entrada y el listado de sugerencias generado por el recomendador clásico y el recomendador basado en contexto. Los cálculos de la similitud mencionados, fueron obtenidos haciendo uso de cada una de las métricas de comparación descritas en la sección 6.2.2.1. A partir de los resultados del cálculo de estas métricas se obtuvo la media aritmética y la desviación estándar de las similitudes obtenidas para los dos sistemas de recomendaciones. Lo anterior con el fin de determinar el S.R. que brinda los resultados más precisos o similares con respecto a la emoción de entrada del usuario y de manera más específica con respecto a las características musicales de arousal y valence asociados a dicha emoción, ver figura 6.10.

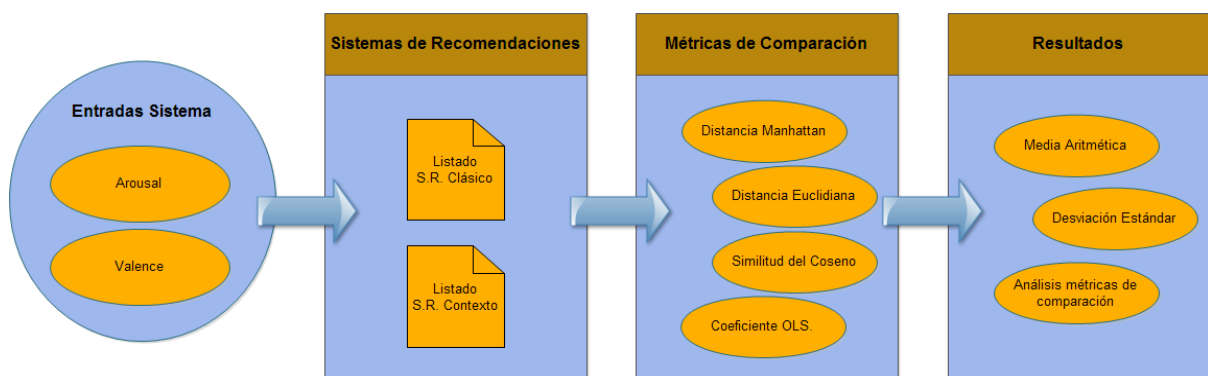


Figura 6. 10. Diagrama de evaluación S.R. Fuente: propia.

La emoción de entrada de usuario, es determinada a partir del arousal y valence obtenidos mediante la entrada de voz del usuario. Además de los cálculos de similitud realizados en esta sección, se presenta un análisis de las métricas de comparación lineal con base en las recomendaciones de contenidos de video que proporcionaban los S.R. al usuario, con el fin de evaluar y describir la métrica que mejor resultados aporta según la emoción de entrada de usuario. A continuación, se presentan los resultados de las pruebas de comparación realizadas con las recomendaciones de cada S.R., teniendo en cuenta las 5 posibles emociones de entrada (feliz, excitado, enojado, triste y relajado).

6.3.2.2.1 Evaluación según la media aritmética

La media aritmética es una medida de tendencia central que representa el valor promedio de un conjunto de datos [86]. Esta medida se utilizó para obtener las medias aritméticas de los resultados analizados a través de las pruebas realizadas sobre ambos S.R. con las métricas de comparación, mencionadas en la sección 6.2.2.1. Para el desarrollo de estas pruebas, se consideraron algunas abreviaturas conceptuales en cada uno de las representaciones de los datos, las cuales se presentan en la tabla 6.2.

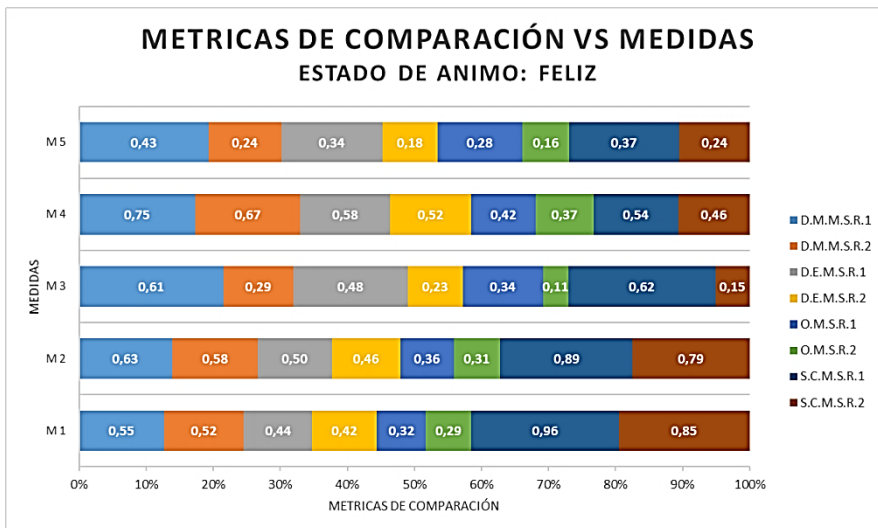
Tabla 6. 2. Abreviaturas de media aritmética para métricas de comparación. Fuente: propia.

MEDIDA	ABREVIATURA
Media de la distancia Manhattan en S.R. Clásico	D.M.M.S.R.1
Media de la distancia Manhattan en S.R. Contexto	D.M.M.S.R.2
Media de la distancia Euclidiana en S.R. Clásico	D.E.M.S.R.1
Media de la distancia Euclidiana en S.R. Contexto	D.E.M.S.R.2
Media del coeficiente OLS. en S.R. Clásico	O.M.S.R.1
Media del coeficiente OLS. en S.R. Contexto	O.M.S.R.2
Media de la similitud del Coseno en S.R. Clásico	S.C.M.S.R.1
Media de la similitud del Coseno en S.R. Contexto	S.C.M.S.R.2

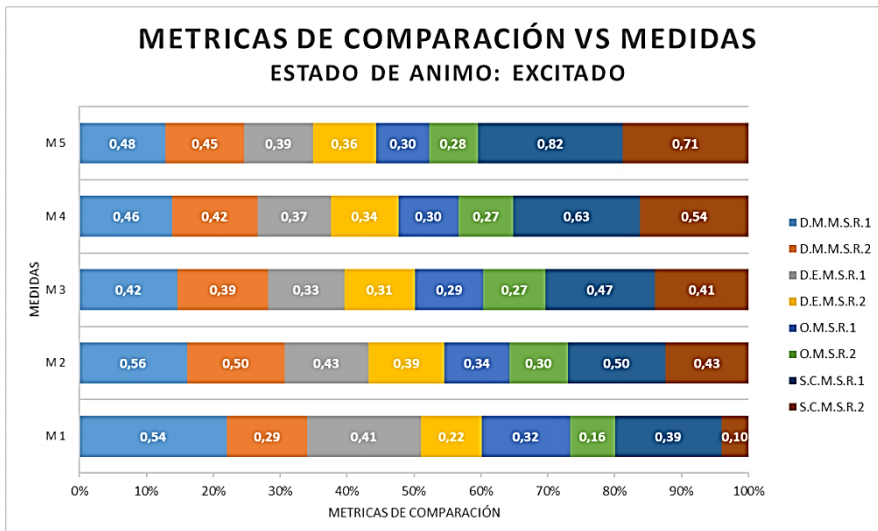
La evaluación de los S.R. se llevó a cabo con usuarios registrados en el servicio de VoD, que habían valorado y no habían valorado contenidos aún. Para ello se realizaron cinco pruebas secuenciales (M1, M2, M3, M4 y M5) para cada uno de los posibles estados de ánimo seleccionados en el presente trabajo (feliz, excitado, enojado, triste y relajado), de manera que se buscó abordar el plano entero del modelo de emociones escogido.

En la figura 6.11 se presentan las pruebas según la media aritmética para cada una de las métricas de comparación, realizadas con cada estado de ánimo sobre los dos recomendadores: clásico (S.R.1) y contexto (S.R.2). En las gráficas de la figura 6.11, se presentan las métricas de dos en dos (S.R.1 y S.R.2), donde las 2 primeras son las métricas según la media aritmética para la distancia Manhattan (S.R.1 en

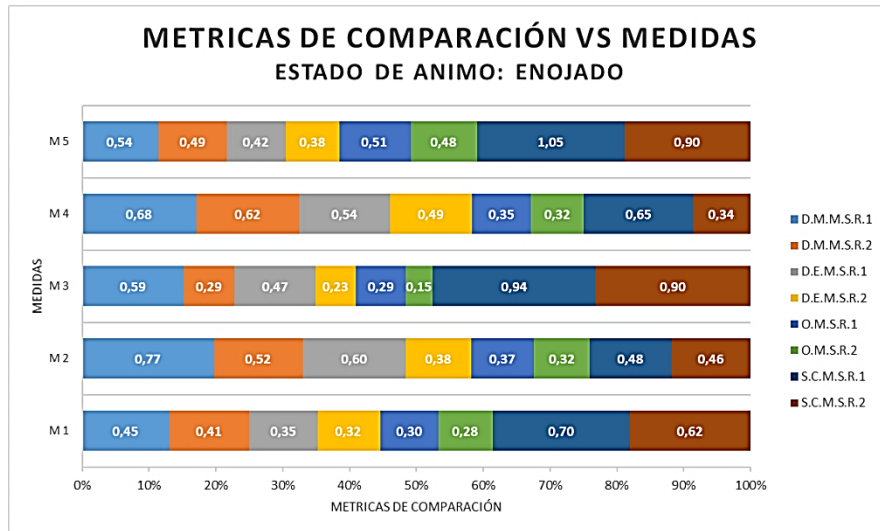
color azul claro y S.R.2 en color naranja), las siguientes 2 de la distancia Euclidiana (S.R.1 en color gris y S.R.2 en color amarillo), seguidamente las 2 del coeficiente OLS (S.R.1 en color azul y S.R.2 en color verde) y finalmente las 2 métricas de la similitud del coseno (S.R.1 en color azul oscuro y S.R.2 en color café). Además de lo anterior, se hizo un ajuste con los resultados de la métrica similitud del coseno, de tal modo que permitiera mayor comprensión en los datos, para lo cual se hizo la parametrización de los datos usando la ecuación 6.4 y este producto se multiplicó por un factor de 10 con cada resultado de la métrica. Así mismo, cabe mencionar que aunque se hizo un ajuste con los datos de similitud del Coseno, esta métrica no brinda información relevante en comparación con las otras métricas estudiadas.



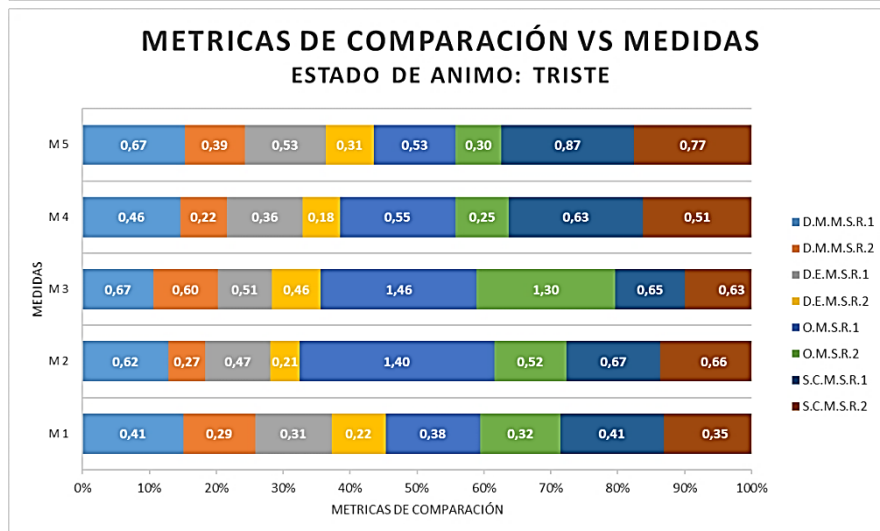
(a)



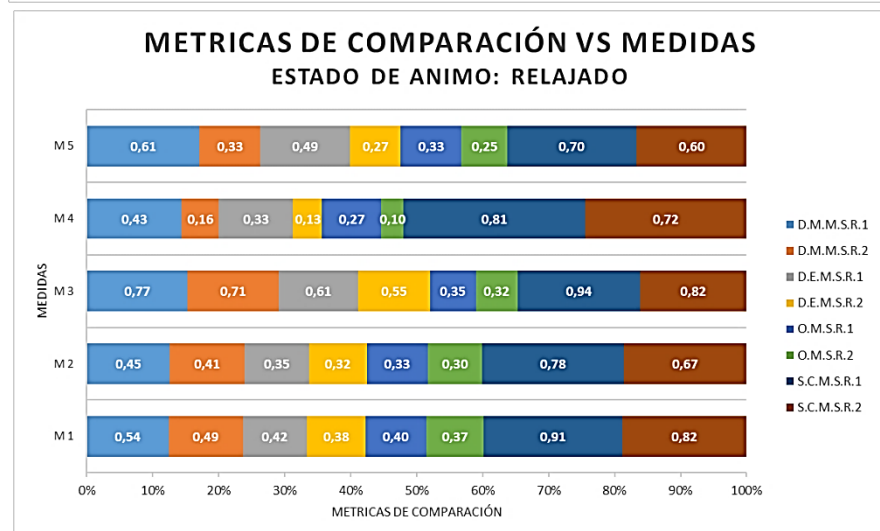
(b)



(c)



(d)



(e)

Figura 6. 11. Métricas de comparación vs medidas (media aritmética). Fuente: propia.

En la figura 6.11a) se muestra la media de cada una de las métricas de comparación contra cada una de las 5 muestras secuenciales capturadas para el estado de ánimo “feliz”, entre el S.R. clásico y el S.R. basado en contexto. En esta gráfica se

aprecia como en las medidas “M3” y “M5”, se encuentra diferencias de 0,32 y 0,19 entre D.M.M.S.R.1 y D.M.M.S.R.2; 0,25 y 0,16 entre D.E.M.S.R.1 y D.E.M.S.R.2; 0,23 y 0,12 entre O.M.S.R.1 y O.M.S.R.2; 0,47 y 0,13 entre S.C.M.S.R.1 y S.C.M.S.R.2; lo cual permite evidenciar que las recomendaciones del S.R.2 son más cercanas en cuanto a similitud al estado de ánimo de entrada de usuario. Las demás métricas son similares en las otras medidas, pero con menores rangos de diferencia que varían entre 0,03 y 0,08 para D.M.M.; 0,02 y 0,06 para D.E.M.; 0,03 y 0,05 para O.M.; 0,09 y 0,11 para S.C.M.

En la figura 6.11b) se muestra la media de cada una de las métricas de comparación contra cada una de las 5 muestras secuenciales capturadas para el estado de ánimo “excitado”, entre el S.R. clásico y el S.R. basado en contexto. En esta gráfica se aprecia como en la medida “M1”, se evidencian diferencias de 0,25 entre D.M.M.S.R.1 y D.M.M.S.R.2; 0,19 entre D.E.M.S.R.1 y D.E.M.S.R.2; 0,16 entre O.M.S.R.1 y O.M.S.R.2; 0,29 entre S.C.M.S.R.1 y S.C.M.S.R.2; de tal manera que se observando como levemente el S.R.2 presenta recomendaciones más adecuadas (en términos de similitud) que el S.R.1, a partir del estado de ánimo de entrada. Además, se distingue como a medida que se realizaron las medidas para el estado de ánimo excitado, los resultados fueron similares al anterior estado.

En la figura 6.11c) se muestran los resultados para el estado de ánimo de entrada “enojado”. Los datos presentados en esta gráfica son algo diferentes con respecto a resultados anteriores, ya que para algunas medidas, las métricas presentan mejores resultados que en otras. Por ejemplo, los resultados de las medidas “M2” y “M3” presentan gran diferencia en sus métricas de comparación, a su vez que “M1”, “M4” y “M5” tienen un menor grado de similitud. Los resultados de la medida “M3” presentan una diferencia de 0,30 entre D.M.M.S.R.1 y D.M.M.S.R.2, además una diferencia de 0,24 entre D.E.M.S.R.1 y D.E.M.S.R.2, y una diferencia de 0,14 entre O.M.S.R.1 y O.M.S.R.2 favoreciendo al S.R.2. Lo anterior difiere en el caso de los datos de la medida “M4”, donde la diferencia entre D.M.M.S.R.1 y D.M.M.S.R.2 es 0,06, entre D.E.M.S.R.1 y D.E.M.S.R.2 es 0,05 y 0,03 entre O.M.S.R.1 y O.M.S.R.2, siendo así las diferencias mucho menores. Lo anterior permite evidenciar como el S.R.2 presenta mejores recomendaciones que el S.R.1, de acuerdo al estado de ánimo de entrada de usuario. En lo que respecta a las diferencias representativas en las medidas, encuentra explicación en el hecho de que se llevó forzosamente al usuario a un estado de ánimo enojado por un periodo de tiempo, a través de diversos factores que pudieran producir estrés como son: el ruido, oscuridad, entre otros.

En la figura 6.11d) se presentan los resultados para el estado de ánimo de entrada “triste”. En esta gráfica se observan los mejores resultados con respecto a resultados anteriores, donde las diferencias en las métricas de comparación rondan los rangos de 0,35 entre D.M.M.S.R.1 y D.M.M.S.R.2; 0,26 entre D.E.M.S.R.1 y D.E.M.S.R.2; 0,88 entre O.M.S.R.1 y O.M.S.R.2; y 0,01 entre S.C.M.S.R.1 y

S.C.M.S.R.2, evidenciando así que las recomendaciones del S.R.2 son más adecuadas (en cuanto a similitud) que el S.R.1 de acuerdo al estado de ánimo de entrada de usuario.

En la figura 6.11e) se presentan los datos obtenidos para el estado de ánimo de entrada “relajado”, ilustrándose los resultados de las métricas contra cada una de las 5 muestras secuenciales capturadas para el estado de ánimo “relajado”. En esta gráfica se aprecia como para cada medida existe una tendencia similar a la de las gráficas anteriores, permitiendo concluir que el S.R.2 presenta para el caso del estado de ánimo relajado un conjunto de recomendaciones más adecuadas (en cuanto a similitud) que las del S.R.1.

Así, a modo de conclusión de las pruebas realizadas en esta sección, se observa como los resultados analizados para ambos S.R.: clásico (S.R.1) y basado en contexto (S.R.2), evidencian que las recomendaciones de S.R.2 son más precisas que el S.R.1 (en cuanto a similitud), con respecto a las diferentes emociones de entrada del usuario. De igual manera, es importante precisar que para las pruebas realizadas con los estados de ánimo: feliz, relajado y triste, se presentan mejores resultados de similitud en las recomendaciones. En lo que se refiere al estado de ánimo excitado, se presentan también resultados adecuados, pero con un grado menor de similitud. Finalmente, en lo que respecta al estado de ánimo enojado, es el único que presenta menores diferencias en sus resultados para ambos recomendadores, pero con resultados positivos de similitud a favor del S.R.2.

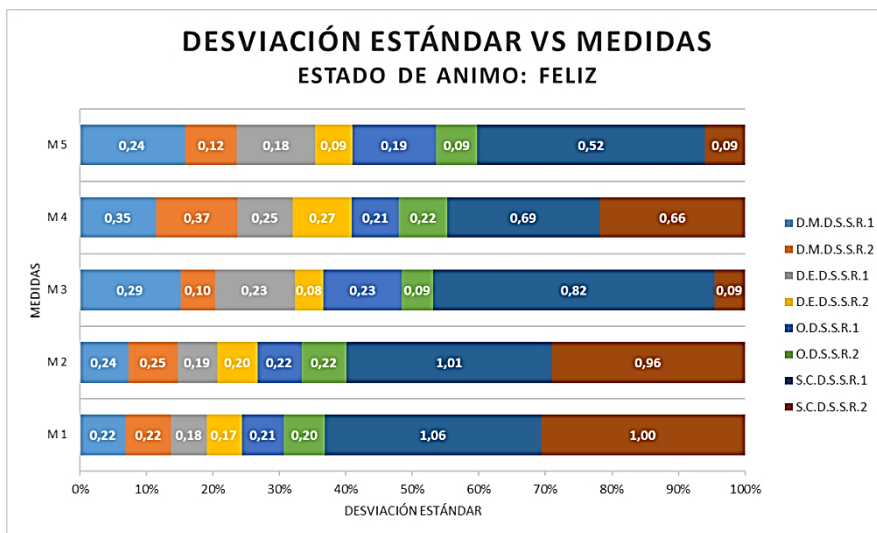
6.3.2.2.2 Evaluación según desviación estándar

La desviación estándar es un índice numérico que indica la dispersión de un conjunto de datos, de modo que entre menor sea este valor, menor es grado de dispersión y mayor similitud presentan los resultados [86]. Esta medida se utilizó para obtener la desviación estándar de los resultados analizados a través de las pruebas realizadas en ambos recomendadores con las métricas de comparación, mencionadas en la sección 6.2.2.1. En la tabla 6.3 se muestran un conjunto de abreviaturas que se definieron con el fin de permitir un mejor manejo y análisis de los resultados de las gráficas.

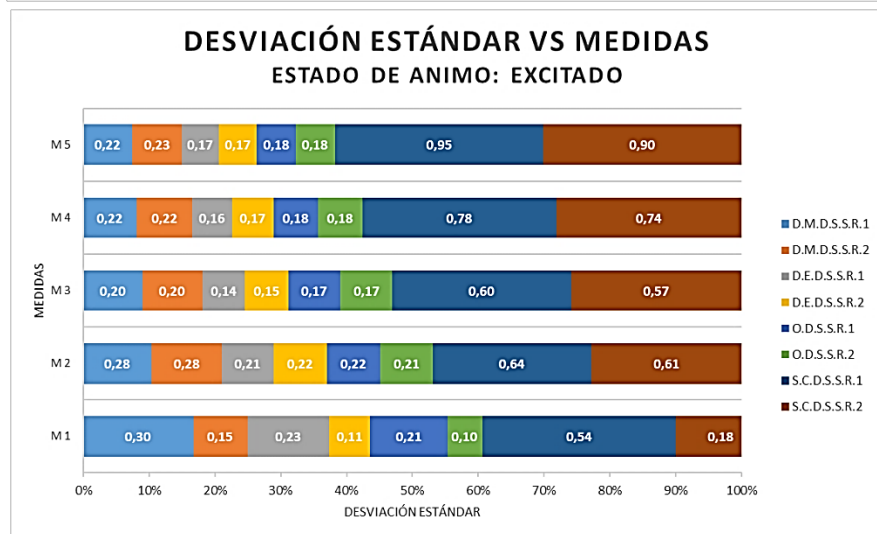
Tabla 6. 3. Abreviaturas de desviación estándar para métricas de comparación. Fuente: propia.

MEDIDA	ABREVIATURA
<i>Desviación estándar de la distancia Manhattan en S.R Clásico</i>	<i>D.M.D.S.S.R.1</i>
<i>Desviación estándar de la distancia Manhattan en S.R Contexto</i>	<i>D.M.D.S.S.R.2</i>
<i>Desviación estándar de la distancia Euclidiana en S.R Clásico</i>	<i>D.E.D.S.S.R.1</i>
<i>Desviación estándar de la distancia Euclidiana en S.R Contexto</i>	<i>D.E.D.S.S.R.2</i>
<i>Desviación estándar del coeficiente OLS. en S.R Clásico</i>	<i>O.D.S.S.R.1</i>
<i>Desviación estándar del coeficiente OLS. en S.R Contexto</i>	<i>O.D.S.S.R.2</i>
<i>Desviación estándar de la similitud del Coseno en S.R Clásico</i>	<i>S.C.D.S.S.R.1</i>
<i>Desviación estándar de la similitud del Coseno en S.R Contexto</i>	<i>S.C.D.S.S.R.2</i>

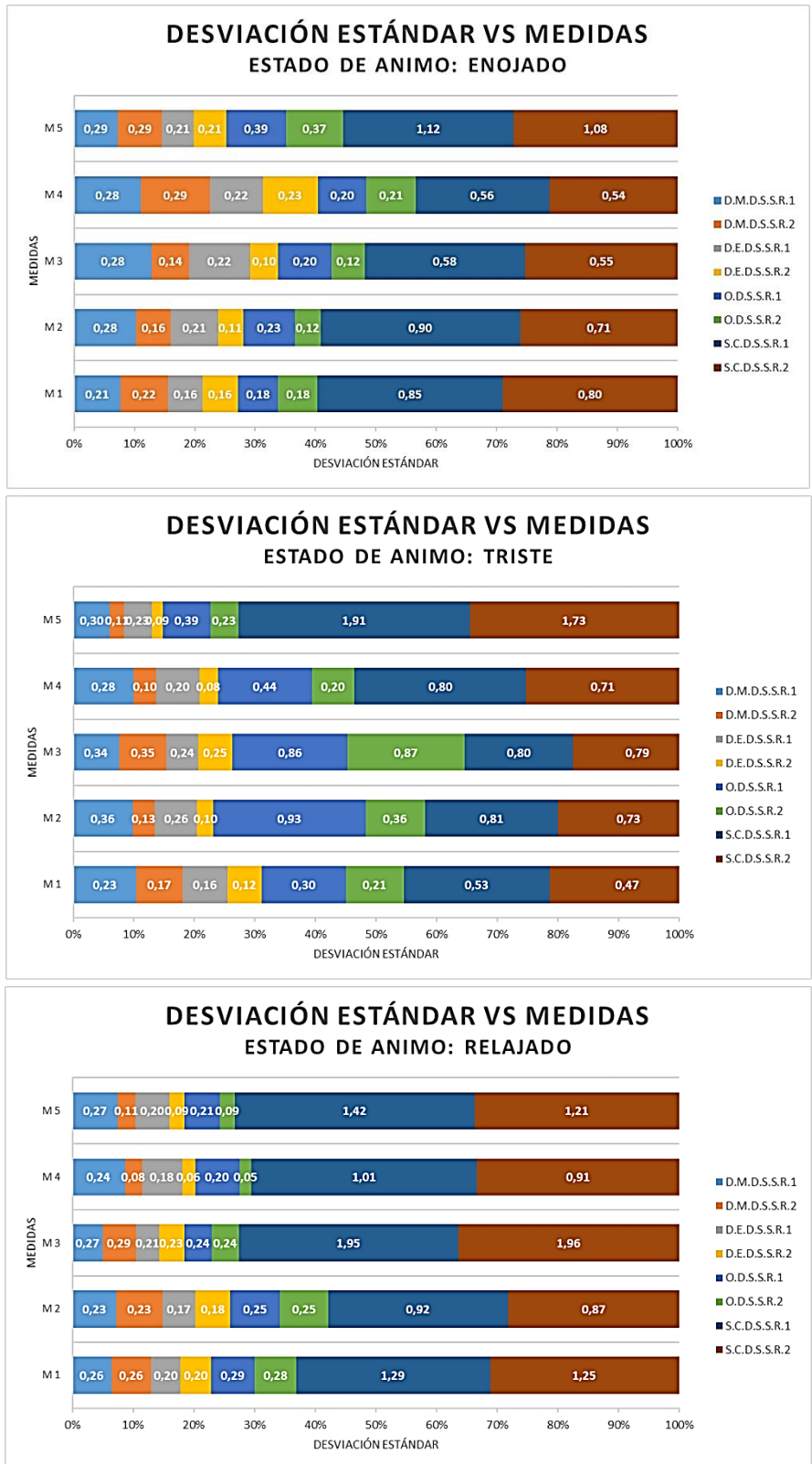
El análisis según desviación estándar con cada uno de los resultados, se llevó a cabo por medio de cinco pruebas secuenciales (M1, M2, M3, M4, M5) considerando los cinco posibles emociones de entrada (feliz, excitado, enojado, triste y relajado). En la figura 6.12, se presentan los resultados de desviación estándar realizados a través de las pruebas a cada S.R., teniendo en cuenta el estado de ánimo de entrada de usuario. En las gráficas de la figura 6.12, se presentan las métricas de dos en dos (S.R.1 y S.R.2), donde las 2 primeras son las métricas según la desviación estándar para la distancia Manhattan (S.R.1 en color azul claro y S.R.2 en color naranja), las siguientes 2 de la distancia Euclidiana (S.R.1 en color gris y S.R.2 en color amarillo), seguidamente las 2 del coeficiente OLS (S.R.1 en color azul y S.R.2 en color verde) y finalmente las 2 métricas de la similitud del coseno (S.R.1 en azul oscuro y S.R.2 en café).



(a)



(b)



(c)

(d)

(e)

Figura 6. 12. Desviación estándar vs medidas. Fuente: propia.

En la figura 6.12a) se muestran los resultados según desviación estándar para el estado de ánimo de entrada “feliz”. En esta gráfica se observa como para las medidas “M3” y “M5”, el S.R.2 es más adecuado y presenta menor dispersión en

los resultados que el S.R.1, teniendo como rangos de diferencia en las métricas de comparación valores entre 0,14 y 0,18 para “M3”, y valores entre 0,09 y 0,12 para “M5”. Por su parte, en el caso de “M2” y “M4”, los resultados muestran al S.R.2 levemente desfavorable con respecto al S.R.1, presentando diferencias en las métricas de comparación entre 0,01 y 0,02 a favor S.R.1. Así mismo, en la medida “M1” los resultados son equivalentes a los anteriores, con rangos de diferencia entre 0,01 y 0,05, mostrando levemente que las recomendaciones del S.R.2 son más adecuadas que las del S.R.1.

En la figura 6.12b) se presentan los resultados para el estado de ánimo de entrada “excitado”. En esta gráfica se puede apreciar como para la medida “M1”, los resultados presentan al S.R.2 más adecuado en las recomendaciones que el S.R.1. De igual forma, en las demás medidas los valores presentan un comportamiento similar, favoreciendo levemente los resultados de las métricas al S.R.2, teniendo en cuenta el estado de ánimo de entrada.

En la figura 6.12c) se presentan los resultados para el estado de ánimo de entrada “enojado”. Tal como se aprecia en esta figura, los resultados de las medidas “M1” y “M5” son similares, con unas pequeñas diferencias comprendidas entre los rangos de 0,01 y 0,05. Por su parte, en las medidas “M2” y “M4” el S.R.2 presenta resultados más adecuados en las recomendaciones que el S.R.1, de acuerdo al estado de ánimo de entrada de usuario.

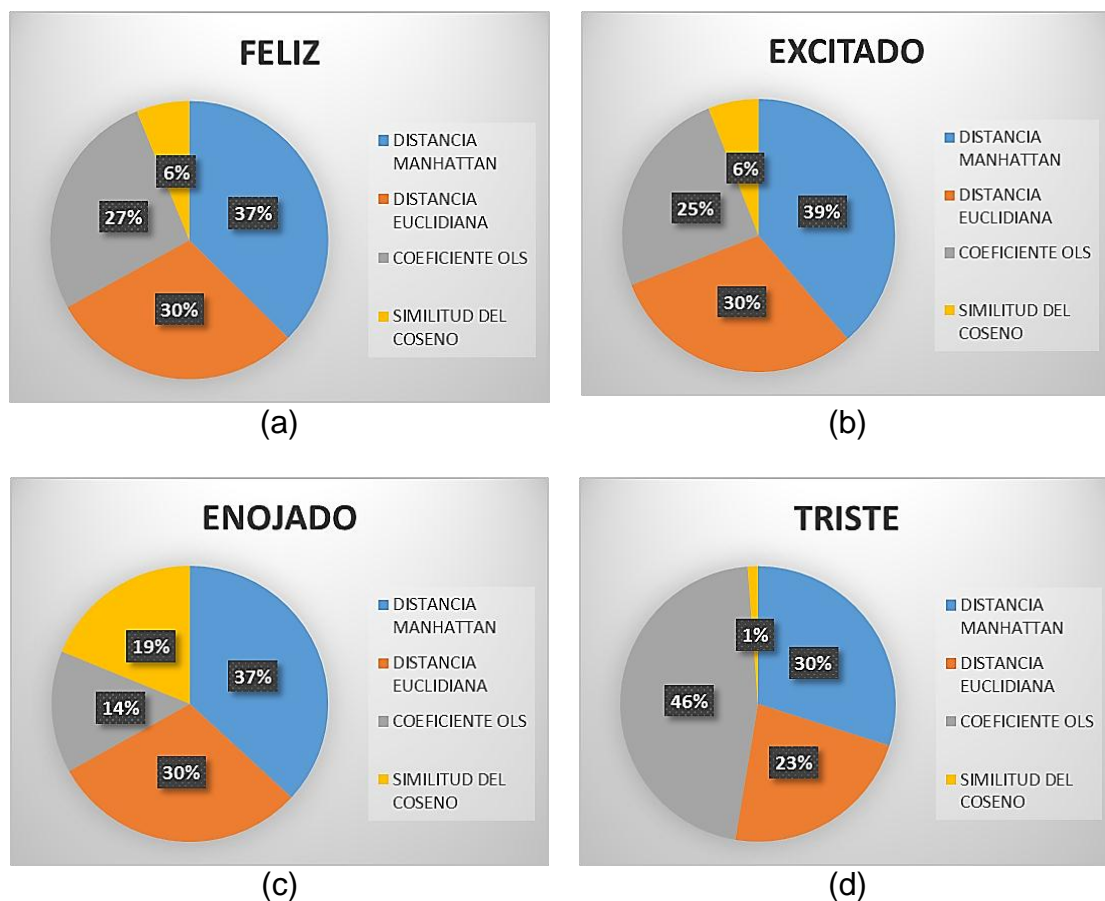
En la figura 6.12d) se presentan los resultados para el estado de ánimo de entrada “triste”. En esta gráfica se observa como el S.R.2 es más adecuado y con menor grado de dispersión en los resultados que el S.R.1, teniendo rangos de diferencia entre 0,06 y 0,23 para D.M.D.S; 0,04 y 0,16 para D.E.D.S; 0,09 y 0,57 para O.D.S; y 0,01 y 0,18 para S.C.D.S, de acuerdo al estado de ánimo de entrada.

En la figura 6.12e) se muestra los resultados para el estado de ánimo de entrada “relajado”. Los resultados presentados en esta gráfica son similares a los anteriores con rangos de diferencia entre los 0,01 y 0,16 en sus métricas de comparación, presentando levemente al S.R.2 con recomendaciones más adecuadas y con menor grado de dispersión que el S.R.1, de acuerdo al estado de ánimo de entrada.

Después de un análisis sobre las desviaciones estándar para cada una de las métricas de comparación, se puede observar que el S.R.2 presenta recomendaciones de contenidos más precisas y con un menor grado de dispersión que el S.R.1. Así, el sistema de recomendaciones basada en contexto propuesto, puede ser una alternativa de solución al problema de arranque en frío y escasez de datos presente en los S.R. clásicos.

6.2.2.3 Análisis métricas de comparación

El análisis de las métricas de comparación se llevó a cabo con la diferencia entre los resultados obtenidos en las pruebas realizadas al S.R. clásico y al S.R. basado en contexto. Lo anterior con el fin de distinguir la métrica de comparación más adecuada para la evaluación del S.R. basado en contexto, teniendo en cuenta los diferentes estados de ánimo de entrada. En la figura 6.13, se muestra la representación de los resultados de comparación, con cada una de las siguientes métricas: distancia Manhattan, distancia Euclidiana, coeficiente OLS y similitud del Coseno; métricas que fueron utilizadas en el análisis y pruebas de las secciones anteriores. En estos gráficos, se utilizó el color azul para representación de la distancia Manhattan, naranja para la distancia Euclidiana, gris para el coeficiente OLS y amarillo para la similitud del Coseno. Los datos presentados en el análisis de las métricas de comparación, son los porcentajes de diferencia entre el recomendador clásico y recomendador basado en contexto.



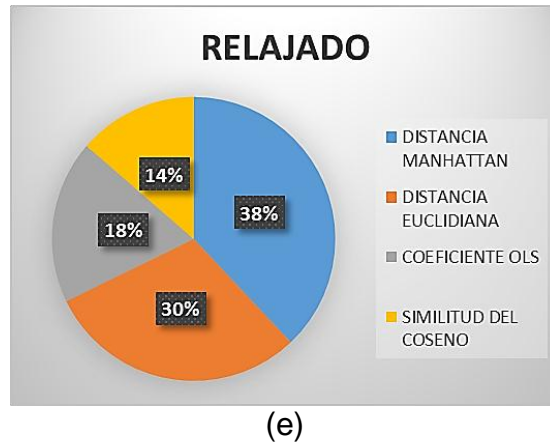


Figura 6. 13. Análisis métricas de comparación. Fuente: propia.

En la figura 6.13a) se muestra el porcentaje de similitud para cada una de las métricas de comparación sobre el estado de ánimo “feliz”. En esta gráfica la distancia Manhattan presenta el mayor valor de similitud, con un 37% frente a un 30% de la distancia Euclidiana, 27% del coeficiente OLS y 6% de la similitud del coseno. De manera similar, en la figura 6.13b) se presentan los porcentajes para el estado “excitado”, con un 39% la distancia Manhattan frente a las demás métricas. En la figura 6.13c) se muestran los porcentajes para el estado “enojado”, con un 37% la distancia Manhattan frente a las demás métricas que mantienen un menor grado de similitud. En la figura 6.13d) se observan los porcentajes para el estado de ánimo “triste”, donde el coeficiente OLS es la métrica de mayor porcentaje con un 46%, frente a un 30% de la distancia Manhattan, lo anterior considerando que el coeficiente OLS es un método no paramétrico, cuyos mayores valores se presentan para el estado “triste”. En la figura 6.14e) se presentan los porcentajes del estado de ánimo “relajado”, mostrando que la distancia Manhattan conserva mayor porcentaje de similitud en comparación a las demás (resultado similar a los 3 primeros estados).

Tras analizar cada una de las gráficas anteriores, se puede observar que la distancia Manhattan es el método de comparación lineal que cuenta con el porcentaje mayor de similitud para cada uno de los estados de ánimo de entrada evaluados. Así, esta métrica permite mejor comprensión y análisis de los resultados para el sistema de recomendaciones basado en contexto, por lo cual puede ser considerada para evaluar este tipo de sistemas.

6.4 Pruebas de carga y estrés

Para evaluar el servidor de procesamiento, el servidor de contexto y el módulo hardware-software Arduino Yún, se hizo uso de la herramienta apache

benchmarking²³, la cual permite realizar pruebas de estrés/carga sobre un servidor http, al proporcionar un número determinado de conexiones de manera secuencial/concurrente y entregar reporte de los tiempos de respuesta. Las características de los equipos utilizados en las pruebas son: computador (procesador i7 de 2.4-3.4 GHz, 16 GB Ram, 1 TB disco duro), Arduino Yún (procesador Atheros AR9331, Ram 64 MB, velocidad del reloj 16 MHz).

6.4.1 Servidor de Procesamiento

En la figura 6.14 se presentan las pruebas de carga realizadas sobre el servidor de procesamiento, el cual se encarga de generar el listado de recomendaciones, de acuerdo al perfil de usuario y haciendo uso de la herramienta weka. En esta prueba se realizaron conexiones simultáneas (línea naranja) y secuenciales (línea azul) de 10 en 10 hasta alcanzar un total de 200.

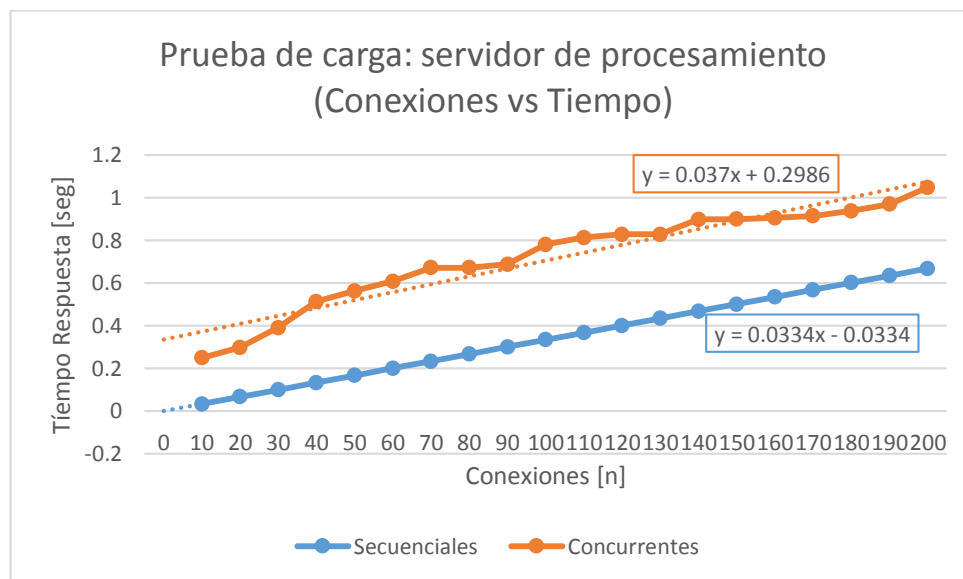


Figura 6. 14. Pruebas de carga, servidor de procesamiento. Fuente: propia.

De acuerdo a la figura 6.14, las conexiones simultáneas o concurrentes presentan los picos más altos de tiempo en comparación con las realizadas de forma secuencial, alcanzando un tiempo máximo de 1.04 segundos sobre las 200 solicitudes simultáneas. Así mismo, según los resultados obtenidos en las pruebas concurrentes, se observa en promedio un incremento en 0.0125 segundos por cada conexión adicional. En el caso de las conexiones secuenciales, se obtiene un valor total de 0.66 segundos para las 200 conexiones. De igual manera, según los resultados obtenidos en las pruebas secuenciales, se observa en promedio un incremento de 0.003 segundos por conexión adicional.

²³ Herramienta para el análisis del rendimiento de servidores, apache benchmarking página web: <https://httpd.apache.org/docs/2.4/programs/ab.html>.

Finalmente, estas pruebas permiten corroborar la relación directamente proporcional entre el número de conexiones y el tiempo de respuesta para las conexiones simultáneas y secuenciales, considerando que a mayor número de conexiones es mayor la capacidad de procesamiento que se requiere para resolver dichas peticiones.

Para establecer el valor máximo de conexiones que puede soportar el servidor de procesamiento, se realizaron las pruebas de estrés que se presentan en la figura 6.15. Así, se hicieron incrementos de 1000 en 1000 en las conexiones secuenciales y simultáneas, obteniendo un tope máximo de 16000 conexiones para las solicitudes concurrentes, con un tiempo total de 180.1 segundos. De igual forma, en el caso de las solicitudes secuenciales, se alcanzó un tope de 20000 conexiones con un tiempo total de 121.8 segundos. Estos resultados, permiten concluir que el servidor de procesamiento basado en el framework web spark de java en conjunto con la herramienta weka, es adecuado en el uso de proyectos de mediana complejidad.

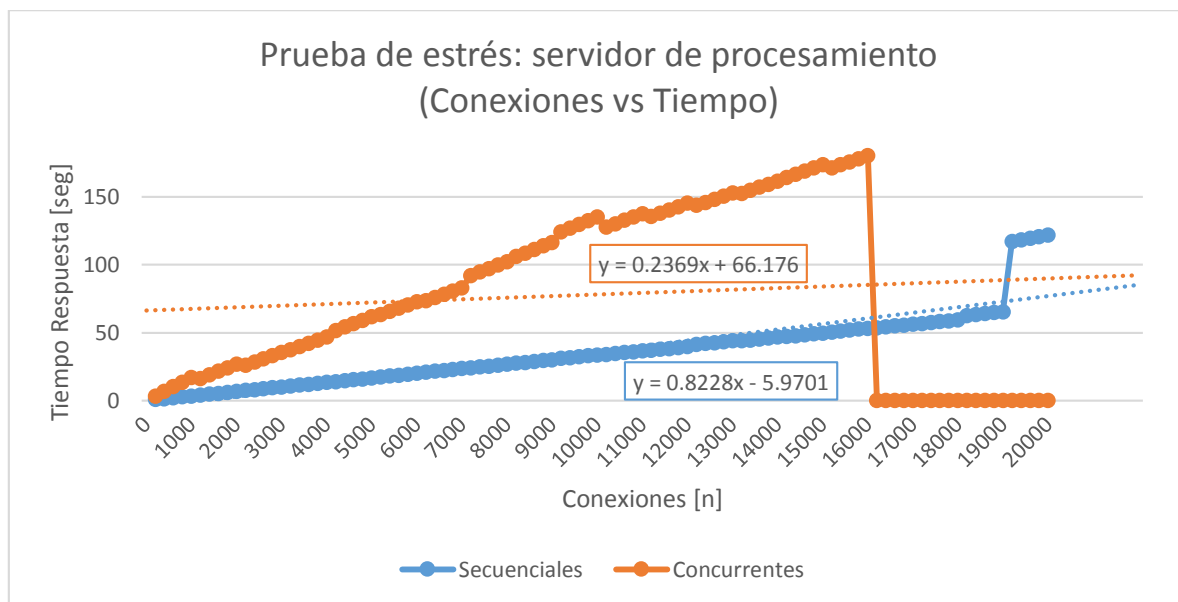


Figura 6. 15. Pruebas de estrés, servidor de procesamiento. Fuente: propia.

6.4.2 Servidor de contexto

En la figura 6.16 se presentan las pruebas de carga realizadas sobre el servidor de contexto, el cual es el encargado del procesamiento de los datos relacionados con las variables fisiológicas, que son utilizados por el S.R. basado en contexto para determinar el estado de ánimo de entrada de usuario. En esta prueba se realizaron conexiones simultáneas (línea naranja) y secuenciales (línea azul) de 10 en 10 hasta alcanzar un total de 200 peticiones.

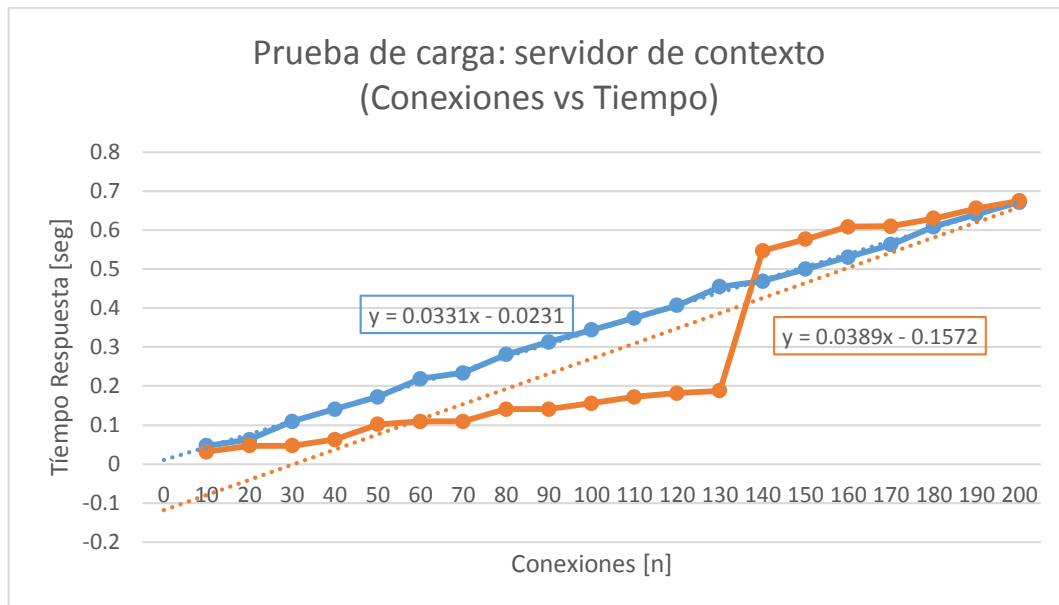


Figura 6. 16. Pruebas de carga, servidor de contexto. Fuente: propia.

De acuerdo a la figura 6.16, el tiempo empleado por el servidor de contexto para procesar 200 conexiones secuenciales, oscila entre 0.050 y 0.675 segundos, presentando en promedio un incremento en 0.003 segundos por cada conexión adicional. De igual manera, el tiempo empleado por el servidor de contexto para procesar 200 peticiones simultáneas oscila entre 0.030 y 0.675 segundos, presentando en promedio un incremento de 0.001 segundos por cada conexión hasta las 130 conexiones, y un incremento promedio cercano a 0.004 segundos por cada conexión adicional después de las 130 conexiones. Estas pruebas permiten corroborar la relación directamente proporcional entre el número de conexiones y el tiempo de respuesta para las conexiones simultáneas y secuenciales, considerando que a mayor número de conexiones es mayor la capacidad de procesamiento que se requiere para resolver dichas peticiones.

Las pruebas de estrés al servidor de contexto se llevaron a cabo con 20000 conexiones secuenciales y simultáneas, hasta lograr que el sistema colapsara. En la figura 6.17 se presenta la representación de los datos con respecto a las pruebas de estrés al servidor de contexto. Para establecer el valor máximo de conexiones que puede soportar el servidor de contexto, se realizaron las pruebas de estrés con conexiones secuenciales (línea azul) y simultáneas (línea naranja), las cuales son presentadas en la figura 6.17. Así, se hicieron incrementos de 400 en 400 en las conexiones secuenciales y simultáneas hasta alcanzar un total de 20000 conexiones. Según los resultados obtenidos, el tiempo empleado por el servidor de contexto para responder a 20000 conexiones secuenciales oscila entre 0 y 36 segundos, presentando mayor consumo de recursos entre las 10000 y 20000 conexiones secuenciales, sin colapsar el sistema para el máximo número de colecciones realizadas. En el caso de las conexiones simultáneas, el tiempo empleado por el servidor de contexto para responder a 20000 conexiones oscila

entre 7 y 62 segundos, siendo las 18000 conexiones el punto crítico donde el servidor de contexto empieza a colapsar.

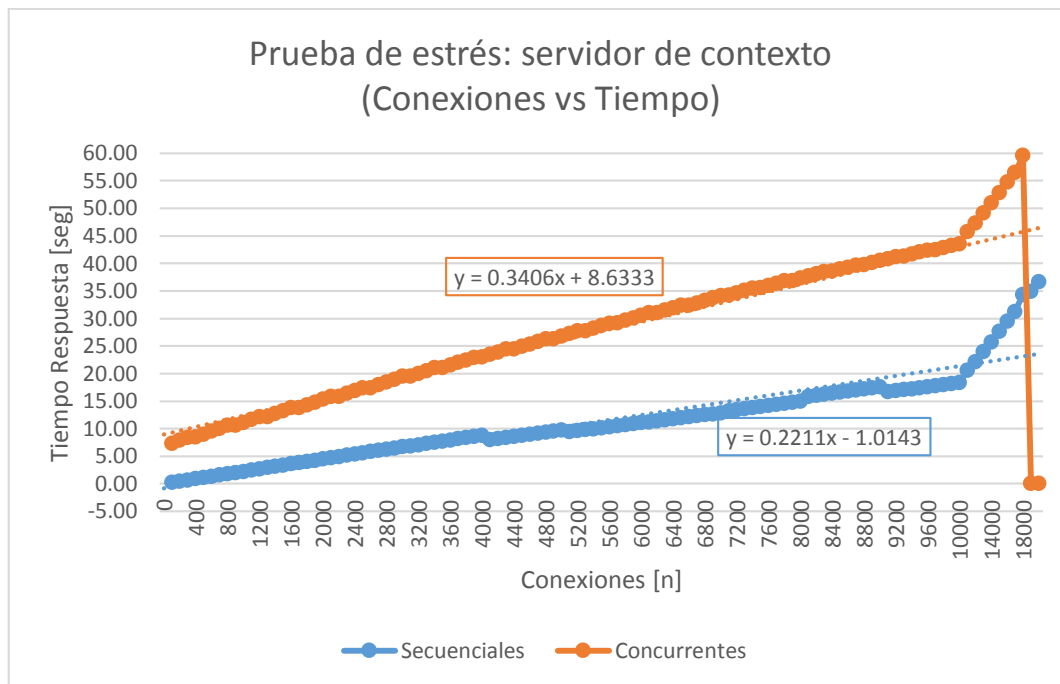


Figura 6. 17. Pruebas de estrés, servidor de contexto. Fuente: propia.

A modo de conclusión, el servidor de contexto presenta buena capacidad de procesamiento de acuerdo a las pruebas de carga y estrés que se realizaron, soportando una capacidad de alrededor de 20000 conexiones secuenciales y cerca de 18000 conexiones simultaneas. De igual manera, cabe mencionar que el tiempo de respuesta para las primeras 200 conexiones simultaneas y secuenciales no excede un segundo. Así, el servidor de contexto, soportado en el framework Flask de Python presenta un rendimiento adecuado y está en la capacidad de integrarse con los demás componentes del S.R: basado en contexto y el servicio de VoD.

6.4.3 Módulo hardware-software (Arduino Yún)

Para el módulo de Arduino Yún las pruebas de carga sufrieron un leve cambio, puesto que en este caso se efectuaron las conexiones sin concurrencia, debido a la baja capacidad de procesamiento con la que cuenta este dispositivo. De este modo, sobre la tarjeta Arduino-Yun se realizaron con ayuda de herramienta apache benchmark conexiones de 10 en 10 hasta un total de 200, evidenciando plenamente una gran variación en los tiempos de respuesta, de tal modo que se presentan picos de tiempo para 10 y 180 conexiones, con un tiempo de 1.407 segundos en cada caso (ver figura 6.18).

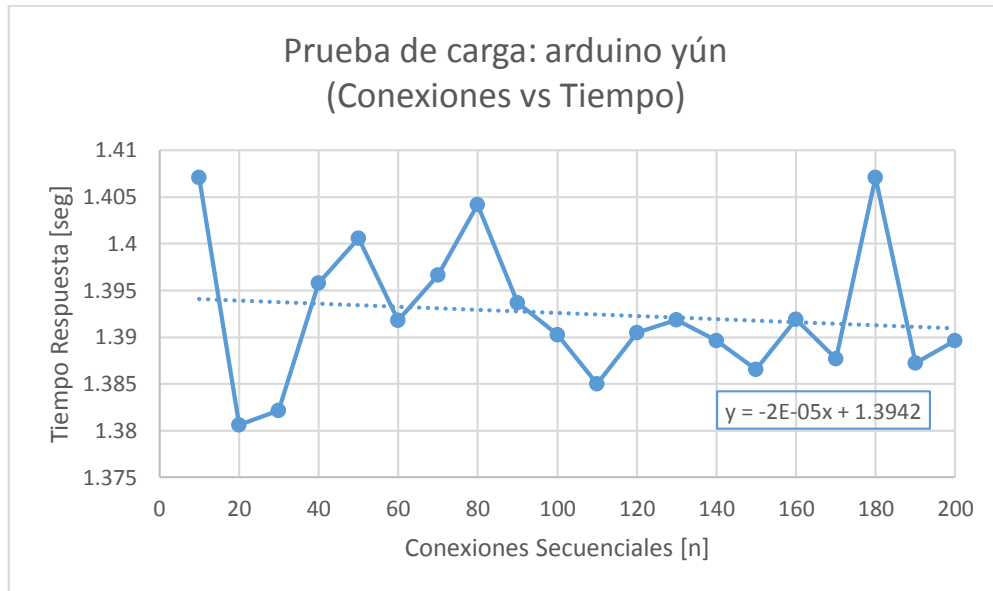


Figura 6. 18. Pruebas de carga, Arduino Yún. Fuente: propia.

Con el fin de probar los límites del dispositivo Arduino en cuanto a rendimiento en el procesamiento de las solicitudes, se llevaron a cabo pruebas de estrés, para lo cual se realizaron 200 conexiones con distintos valores de concurrencia (10,20,30,40,50,60), obteniendo que para 60 conexiones concurrentes el tiempo de respuesta es de 99.4 segundos (ver figura 6.19). Así mismo, de acuerdo a la línea de tendencia de la gráfica, se presenta en promedio un incremento de 1.6 segundos por cada conexión simultánea adicional. A pesar de que los tiempos son altos para múltiples conexiones, es importante considerar que el módulo Arduino-Yun pretende ser usado como unidad personal para la captura de variables de contexto.

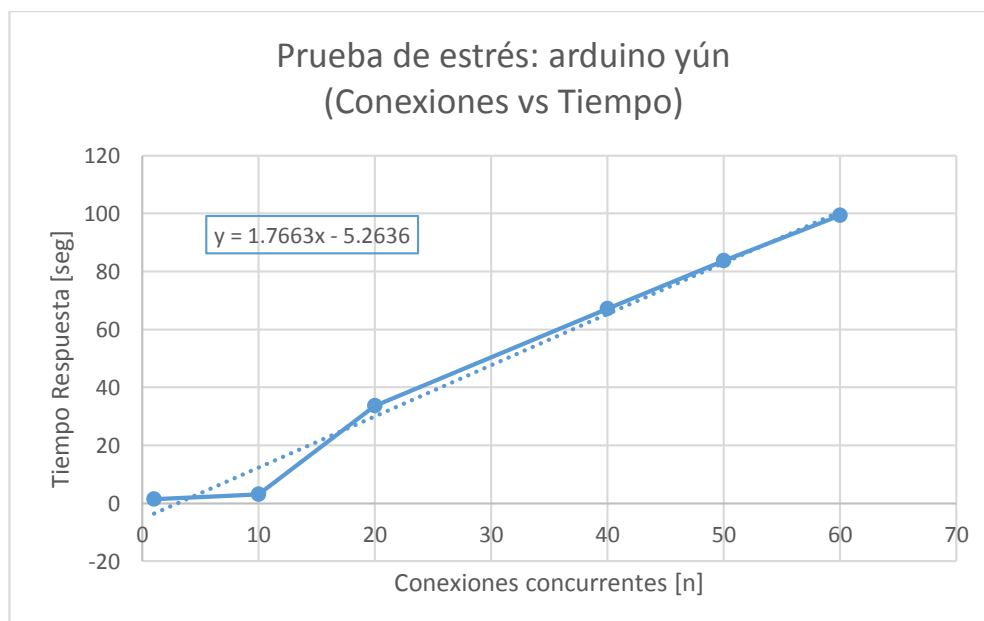


Figura 6. 19. Pruebas de estrés, Arduino Yún. Fuente: propia.

Como conclusiones finales, las pruebas de carga y estrés permiten indicar que el framework Spark de Java puede ser considerado una alternativa adecuada para la implementación de servicios de telecomunicaciones, puesto que es una alternativa al uso de tecnologías como Node.js. Así mismo, a partir de las pruebas realizadas sobre el servidor de contexto soportado en el framework Flask de Python, se concluye que es una alternativa viable para la implementación de servidores intermedios, dada la rapidez en el desarrollo y despliegue de servicios. En lo que respecta al módulo hardware software, si bien los tiempos para múltiples conexiones no son adecuados, el hecho de que esta tarjeta permita la compatibilidad de múltiples sensores y este destinada para la captura personal de variables de contexto, la convierten en una alternativa adecuada para el uso en proyectos relacionados con sistemas sensibles al contexto.

Capítulo 7

7. Conclusiones y Trabajos Futuros

En este capítulo se presentan las conclusiones del trabajo de doctorado, considerado los objetivos y aportes propuestos. Así mismo se plantean los trabajos futuros derivados de la presente investigación, con el fin de aportar líneas futuras de trabajo con respecto a las arquitecturas de telecomunicaciones y sistemas de recomendaciones basados en contexto. Finalmente, se presentan el listado de publicaciones en eventos y revistas obtenidas producto del desarrollo y ejecución de esta investigación, así como los registros software realizados a partir de las implementaciones hechas.

7.1 Conclusiones

Las conclusiones del presente trabajo han sido organizadas de acuerdo a los principales aportes investigativos obtenidos. De este modo, a continuación se presentan las conclusiones organizadas considerando las siguientes categorías: arquitectura basada en contexto, entorno de *streaming* adaptativo, servicio de VoD, sistema de recomendaciones.

7.1.1 Arquitectura basada en contexto

- La arquitectura de alto nivel propuesta en este trabajo, permite enriquecer el servicio de VoD, al abordar diferentes dimensiones de la definición del contexto para servicios de telecomunicaciones, como son: usuario, red y dispositivo, contribuyendo así al mejoramiento de los problemas de acceso ágil y consumo adecuado del contenido multimedia. En la dimensión de usuario la arquitectura

considera un conjunto de variables fisiológicas relevantes, mientras que en la dimensión de red y dispositivo la arquitectura está soportada en un escenario de *streaming* adaptativo DASH.

- La arquitectura de alto nivel propuesta en esta investigación, busca aprovechar en la dimensión de usuario la tendencia actual de internet de las cosas, por obtener información de diferentes fuentes conectadas a internet, como es el caso de los sensores integrados en dispositivos *wereables*. A través del uso de estos dispositivos en el presente trabajo, se buscó obtener información relevante del usuario, permitiendo la recomendación ágil y precisa de contenidos multimedia según las necesidades del usuario.
- De acuerdo a la definición general del contexto y en el caso particular de los servicios de telecomunicaciones, las variables de contexto en las diferentes dimensiones de la arquitectura pueden ser consideradas infinitas, por lo cual conviene seleccionar las variables que brinden información relevante con respecto a las necesidades del usuario, con el fin de recomendar o brindar los contenidos multimedia más adecuados. En el presente trabajo se seleccionaron la variación del ritmo cardíaco, la conductividad de la piel y la voz, considerando la conexión existente entre estas y los modelos de emociones a través de los contenidos multimedia musicales.
- Para el soporte de servicios de transmisión de contenidos multimedia y de servicios de telecomunicaciones en general, en la mayoría de los casos se requiere contar con una infraestructura tecnológica de elevado costo. En el presente trabajo de doctorado, se propuso una arquitectura cuya infraestructura básica cuenta con componentes hardware de mediano costo, acompañado de herramientas software libre que permiten emular componentes reales de la cadena de valor del servicio de IPTV, a través de los cuales se facilita el despliegue servicios asociados al servicio de VoD (sistemas de recomendaciones, servicios interactivos, etc.).
- La arquitectura propuesta para el servicio de VoD aprovecha las ventajas de los contenidos multimedia musicales y su relación con el modelo de emociones, con el objetivo de relacionar el perfil emocional del usuario con un conjunto de contenidos musicales previamente clasificados, a través de un sistema de recomendaciones basado en contexto. Así, a partir de las características musicales de arousal y valence, es posible clasificar un contenido multimedia con las emociones, las cuales son inferidas haciendo uso de variables de contexto de usuario.
- Según los resultados obtenidos en la fase de evaluación de los componentes principales de la arquitectura basada en contexto, es posible concluir que esta contribuye a los retos de agilizar el acceso y mejorar el consumo del contenido multimedia del servicio de VoD. En cuanto al primer reto, la arquitectura plantea la vinculación de un sistema de recomendaciones basado en contexto, que considera variables del contexto de usuario con el fin de generar recomendaciones adecuadas a la necesidad del usuario. En lo que respecta al

segundo reto, la arquitectura está soportada a nivel de la distribución de contenidos en el estándar de *streaming* adaptativo DASH.

- Las pruebas de carga realizadas al módulo de captura de variables de contexto Arduino-Yún demostraron que este componente no es adecuado para soportar una gran cantidad de peticiones concurrentes, debido a sus limitaciones de procesamiento. A pesar de lo anterior, es importante considerar y aclarar que este módulo está pensado dentro de la arquitectura para la captura personal de variables de contexto, función en la cual se obtuvieron tiempos de respuesta adecuados.

7.1.2 Entorno de streaming adaptativo

- El entorno de *streaming* adaptativo desarrollado en el presente trabajo de doctorado, hace uso de herramientas libres dentro de las fases de codificación, difusión y recepción de contenidos multimedia DASH. Así mismo, considerando que este entorno demostró responder de manera adecuada a las pruebas de ancho de banda realizadas, puede ser considerado como una alternativa viable para la implementación de servicios de distribución de contenidos multimedia soportados en *streaming* adaptativo, como es el caso del servicio de video bajo demanda o servicios de transmisión de video en vivo.
- El entorno de *streaming* adaptativo DASH fue evaluado mediante pruebas de transmisión de contenidos multimedia DASH a diferentes rangos de ancho de banda, los cuales fueron emulados con ayuda de la herramienta provista por el proyecto libre dummynet. Estas pruebas demostraron que el entorno de *streaming* adaptativo DASH siguió adecuadamente los diferentes cambios en el ancho de banda emulados, a través de la conmutación de tasa de bits en los contenidos multimedia transmitidos en formato WebM. Lo anterior, permite concluir que este entorno de *streaming* es apto para la construcción y despliegue de servicios de transmisión de contenidos multimedia en vivo o bajo demanda.
- El entorno de *streaming* adaptativo propuesto, fue conformado por las librerías y herramientas del mundo del software libre, las cuales según las pruebas realizadas en el presente trabajo demostraron ser adecuadas para llevar a cabo los procesos de codificación, adecuación, distribución y consumo de contenido multimedia WebM, considerando el estándar DASH. De esta manera, este escenario puede ser extendido o aplicado a otras aproximaciones de servicios, como es el caso de la transmisión de contenidos multimedia adaptativos en vivo.
- El uso del estándar de *streaming* adaptativo DASH, en servicios de transmisión de contenido multimedia en vivo o en diferido, permite mejorar considerablemente la calidad del servicio en entornos de movilidad, con relación al *streaming* tradicional soportado en los protocolos clásicos de transmisión de

contenido multimedia RTP y RTSP, garantizando así un menor tiempo de arranque, selección de la calidad de los contenidos multimedia en función de las capacidades de la red y del dispositivo, cambios automáticos y transparentes entre las calidades de los contenidos multimedia, supresión del uso del buffer en la recepción de los contenidos, entre otras ventajas.

- La herramienta de codificación automática propuesta perteneciente al entorno de *streaming* implementado, permite optimizar el proceso de codificación, segmentación y generación del archivo de manifiesto MPD para contenidos multimedia adaptativos WebM DASH. Para lo anterior, esta herramienta integra en segundo plano las funcionalidades provistas de manera separada por las librerías libewbm, webm-tools y la herramienta de codificación FFmpeg. Así mismo, la herramienta Dash WebM Converter optimiza el tiempo de codificación al lanzar de manera simultánea tantos hilos como tasas de bits estén asociadas a un contenido multimedia. De esta manera, la herramienta propuesta puede ser considerada como una opción adecuada para la construcción de entornos de *streaming* adaptativo, en lo que respecta a las fases de codificación y generación del contenido multimedia DASH.
- Mediante la herramienta de codificación automática DASH WebM Converter perteneciente al entorno de *streaming*, es posible el despliegue de un entorno de codificación y transmisión de *streaming* adaptativo WebM basado en el estándar DASH. Este entorno tiene como ventaja la reducción de la infraestructura para el montaje de servicios de transmisión de contenido multimedia, al no requerir el uso de un servidor de *streaming*, puesto que los contenidos y la lógica de navegación a través de ellos se encuentra centralizada en un servidor Web, evitando así problemas de sincronización. Lo anterior considerando que tanto la lógica del servicio de VoD, como el consumo de contenidos DASH operan sobre el protocolo HTTP.
- En cuanto a las pruebas de consumo de memoria realizadas sobre la herramienta FFmpeg, la cual fue usada en segundo plano por la herramienta de codificación DASH WebM Converter perteneciente al entorno de *streaming* adaptativo, es de mencionar que en estas mediciones se codificaron contenidos multimedia a 6 tasas de bits. De esta manera DASH WebM Converter lanzó para el proceso de codificación, 7 hilos o instancias simultáneas (6 hilos de video y un hilo de audio) de la herramienta FFmpeg. De acuerdo a los resultados obtenidos, la cantidad de memoria RAM asignada a cada hilo de video es en promedio de 0.2 MB, lo que permite concluir que el hecho de usar procesamiento multihilo no representa un incremento considerable en la cantidad de memoria RAM, con respecto a las ventajas obtenidas de optimización de tiempo.
- La librería javascript webm-dash-javascript, permite el despliegue adecuado del contenido multimedia DASH WebM desde el cliente web. Para lo anterior, esta librería estima el ancho de banda disponible en el lado del cliente, y de acuerdo a este valor obtiene el segmento de contenido multimedia apropiado para la

reproducción, de acuerdo a la información disponible en el archivo manifiesto MPD. De esta forma, `webm-dash-javascript` facilita la integración y reproducción de contenidos multimedia WebM en formato DASH a través del componente de video provisto por HTML5. Lo anterior permite concluir además, que esta librería puede considerarse en la conformación de un cliente web para la recepción de contenidos multimedia transmitidos en vivo.

7.1.3 Servicio de VoD

- El servicio de VoD propuesto en el presente trabajo se enmarca dentro de la definición de contexto para servicios de telecomunicaciones, en el sentido que vincula el contexto de red, el contexto del dispositivo y el contexto de usuario en torno a un servicio. Los contextos de red y de dispositivo fueron considerados mediante la inclusión del estándar de *streaming* adaptativo DASH en el consumo de contenidos multimedia a partir de las características de la red y del dispositivo de acceso. El contexto de usuario fue considerado mediante la vinculación de un conjunto de variables fisiológicas que sirvieron para obtener una emoción de entrada al servicio.
- Un servicio de VoD de contenidos multimedia adaptativos soportado en el estándar DASH, ofrece un enfoque diferente al tradicional al hacer uso del protocolo HTTP. Lo anterior considerando, que de esta manera el servicio es desplegado sobre la infraestructura existente de internet, permitiendo así la coexistencia e integración de servicios telco con servicios de la web.
- El servicio de VoD construido en el presente trabajo permite la distribución de contenido multimedia haciendo uso de la tecnología de *streaming* adaptativo DASH. Lo anterior tiene como ventaja que los contenidos multimedia codificados según este estándar, pueden ser alojados en un servidor web HTTP tradicional sin necesidad de requerir un servidor de *streaming* dedicado, permitiendo así integrar de manera más sencilla la lógica del servicio con los contenidos multimedia.
- Se planteó como aporte del presente trabajo de doctorado, un nuevo servicio de VoD basado en emociones, el cual a partir de una emoción de entrada de usuario, presenta un conjunto de contenidos de video musicales que han sido previamente clasificados y consignados en el dataset de contenidos multimedia. Este servicio permitió la evaluación del dataset de contenidos conformado, a la vez que posibilita su extensión a futuros proyectos que hagan uso de otras variables del contexto para la inferencia de emociones de usuario.
- El servicio de VoD basado en emociones propuesto, fue construido haciendo uso de herramientas y tecnologías de hardware y software libre, lo que posibilita que la arquitectura del servicio pueda ser considerada como base para

proyectos relacionados, cuyo propósito sea enriquecer el servicio de VoD a través de la vinculación de diferentes variables del contexto de usuario.

- El dataset de contenidos multimedia de video conformado y usado como soporte del servicio de VoD, constituye un aporte importante para el diseño e implementación de servicios de video basados en emociones, integrando las ventajas del dataset musical de Echonest y la información de video provista por el catálogo de contenidos de youtube. De igual manera, el modelo de clasificación de emociones en el cual se basa el dataset (sección 2.6.1), puede ser considerado para el desarrollo de servicios de contenidos multimedia basados en emociones, tales como: sistemas de recomendaciones, buscadores servicios publicitarios, entre otros.
- El uso de las API's provistas por Echonest para los lenguajes Python y Java permitió evidenciar que son una alternativa adecuada a tener en cuenta en la implementación de servicios, que consideren la relación de contenidos multimedia con modelos emocionales. Lo anterior considerando que estas librerías permiten acceder a un conjunto relevante de características musicales, sin restricciones a nivel del número de consultas realizadas.
- La herramienta Java para la clasificación de contenidos musicales desarrollada en el presente trabajo de doctorado, permitió verificar la correcta clasificación de los contenidos durante el proceso de conformación del dataset de video generado. Así mismo, esta herramienta posibilitó la evaluación de la pertinencia del modelo de cinco emociones propuesto a partir del modelo de Rusell. Esta herramienta puede ser de utilidad para la experimentación y clasificación de emociones en servicios que hagan uso de contenidos multimedia musicales.

7.1.3 Sistema de recomendaciones

- Teniendo en cuenta que los problemas de salto de canal y de gran tamaño en el catálogo de contenidos multimedia son transversales a las diferentes plataformas de televisión digital interactiva (Televisión Digital Terrestre, Televisión Móvil, WebTV, etc), los sistemas de recomendaciones basados en contexto y que tienen en cuenta el modelo emocional, son una opción válida para la recomendación de contenidos multimedia más adecuados a las necesidades emocionales de los usuarios, así como para mejorar la precisión de los sistemas de recomendaciones tradicionales desplegados en entornos de televisión.
- El presente trabajo incluyó a nivel del contexto de usuario, un conjunto de nuevas variables de tipo fisiológico, las cuales a diferencia de las tradicionales, permitieron obtener información más adecuada acerca de las percepciones y comportamientos emocionales de un usuario. Estas variables permitieron la inferencia de una emoción de entrada, la cual fue usada por el sistema de

recomendaciones basado en contexto para enriquecer las recomendaciones generadas por el recomendador bayesiano clásico. Así a partir de la información inferida de las variables del contexto, fue posible aportar a la solución del problema de arranque en frío, presente en los enfoques clásicos de sistemas de recomendaciones.

- La variable VFC se convierte en parte fundamental del sistema de recomendaciones basado en contexto, ya que permite la obtención del índice de estrés mental, el cual puede ser asociado directamente con la propiedad arousal de cada uno de los contenidos multimedia musicales. Así a través de esta variable es posible filtrar según la propiedad musical arousal, los contenidos multimedia sugeridos por el sistema de recomendaciones clásico.
- Otra de las variables fundamentales para el sistema de recomendaciones basado en contexto es la GSR, la cual puede asociarse directamente con el nivel de arousal de un usuario. Esta variable es usada por el S.R. basado en contexto para filtrar o reordenar el listado de recomendaciones obtenidas y realimentadas a partir de la variable VFC, de tal modo que el listado de contenidos es reordenados de acuerdo al aumento o disminución en el nivel de arousal.
- Características musicales como el arousal y la valence permitieron la clasificación de los contenidos multimedia musicales en el espacio cartesiano de las emociones, a través de la conformación de un dataset de contenidos multimedia emocionales. Lo anterior permitió agilizar el acceso a los contenidos del servicio de VoD, mediante el sistema de recomendaciones basado en contexto propuesto, el cual permitió relacionar emociones de entrada obtenidas a partir de variables de contexto, con los contenidos multimedia previamente clasificados y consolidados en el dataset generado.
- En este trabajo se presentó la implementación de un clasificador basado en el algoritmo de Naive Bayes que tiene en cuenta los parámetros musicales como medio para la predicción de contenidos multimedia. Así, a partir de los resultados obtenidos, se puede concluir que este clasificador es conveniente para el uso en la implementación de sistemas de recomendaciones que tengan en cuenta variables del contexto, además de las valoraciones hechas por los usuarios sobre los ítems o contenidos, en miras a que se deseen predecir la evaluación de nuevo contenido para los usuarios.
- En este trabajo se presentó como contribución la implementación de un nuevo clasificador para contenidos multimedia musicales basado en el algoritmo matemático de Naive Bayes. El aporte principal del clasificador es la inclusión dentro de sus atributos de parámetros musicales tales como: arousal, valence y tempo, los cuales fueron extraídos y discretizados a partir del dataset de contenidos multimedia musicales conformado. De acuerdo a las pruebas de precisión realizadas sobre el clasificador, se obtuvo que sus resultados son adecuados en cuanto a la predicción de contenidos multimedia emocionales, lo cual lo hace adecuado para el uso en la implementación de S.R. que consideren

la vinculación del modelo de emociones a partir de variables de contexto de usuario.

- El presente trabajo de doctorado planteó como aporte el desarrollo de un componente lógico de procesamiento soportado en la herramienta de minería de datos weka, mediante la cual fue posible implementar la lógica del sistema de recomendaciones clásico, soportada en el algoritmo de clasificación de Naive Bayes. De igual manera, a partir de este componente, se implementó la lógica para la generación de sugerencias por parte del recomendador basado en contexto. Este componente fue desarrollado haciendo uso del framework web de java spark, el cual según las pruebas de carga y estrés realizadas, demostró ser una herramienta adecuada para la implementación de sistemas inteligentes soportados en los algoritmos provistos por weka. Así mismo a partir de los resultados obtenidos, este componente puede ser usado también para la construcción de servicios de telecomunicaciones en la nube.
- En este trabajo de doctorado se presentó un sistema de recomendaciones clásico soportado en la propuesta de un nuevo clasificador bayesiano, el cual demostró ser adecuado y preciso para la recomendación de contenidos multimedia musicales, cuando se dispone de un dataset de contenidos multimedia musicales previamente clasificados. De esta manera, el S.R. basado en contexto permitió enriquecer las recomendaciones del S.R. clásico a través de un conjunto de variables del contexto, cuando el sistema presenta el problema de arranque en frío.
- La herramienta de minería de datos Weka posibilitó a través de su API, la implementación del algoritmo de clasificación de Naive Bayes, el cual fue usado para el desarrollo del sistema de recomendaciones clásico, encargado de la clasificación y predicción de contenidos multimedia a partir de los parámetros musicales de arousal y valence. De esta forma la herramienta weka a partir de su API puede ser considerada como adecuada para el desarrollo y adaptación de algoritmos de minería de datos en proyectos de telecomunicaciones.
- Según los resultados descritos en el capítulo 6 con respecto a las pruebas realizadas sobre el S.R. clásico y el clasificador bayesiano, estos arrojan valores adecuados de precisión en las recomendaciones y/o predicciones que el S.R. entrega al usuario. Lo anterior permite concluir que el diseño del clasificador a partir de características musicales de arousal y la valence, así como el uso de la API de la herramienta de minería de datos weka, son opciones validas a considerar para la construcción de servicios multimedia basados en emociones.
- De acuerdo a los resultados de las pruebas realizadas al S.R. basado en contexto con respecto al S.R. clásico, se puede evidenciar que las sugerencias generadas por el S.R. basado en contexto son más precisas con respecto a la emoción de entrada del usuario, es decir que este recomendador sigue de manera adecuada el estado emocional del usuario que el S.R. clásico. De esta manera, es posible concluir que el S.R. basado en contexto propuesto, es una

opción válida para mejorar la precisión y el problema de arranque en frío presente en los S.R. clásicos.

- El framework web Spark demostró ser un componente software adecuado para el desarrollo ágil de servicios de telecomunicaciones en el lenguaje Java, como es el caso de los sistemas de recomendaciones para contenidos multimedia. Lo anterior considerando que reúne características similares a las provistas por frameworks populares de otros lenguajes tales como node.js de javascript, adicionando las potencialidades y prestaciones del lenguaje Java. Así, el framework web Spark de Java facilita la construcción rápida de servicios en la nube, permitiendo el desarrollo de los mismos de manera similar a como se crean aplicaciones de escritorio y sin contar con una infraestructura adicional de servidor, logrando así una evolución en el desarrollo ágil de servicios en la nube a través del lenguaje java.
- Las pruebas de carga realizadas sobre el servidor de procesamiento implementado a través del framework web de Java spark y la API de weka, permitieron obtener un tiempo promedio de respuesta de 0.7 segundos en efectuar una carga de 200 solicitudes del listado de recomendaciones con diferentes valores de concurrencia. Así, a partir de estos resultados, es posible concluir que las tecnologías usadas para la construcción de este componente software pueden ser consideradas en el desarrollo de servicios de telecomunicaciones asociados a los sistemas de recomendaciones.
- La plataforma hardware de internet de las cosas Arduino Yún, permitió capturar de manera adecuada las señales de los sensores: pulse sensor y GSR, los cuales proporcionaron los datos fisiológicos necesarios para el cálculo del índice de estrés y la obtención de los parámetros musicales: arousal y valence, los cuales pueden relacionarse directamente con el modelo de emociones propuesto y empleado por el sistema de recomendaciones basado en contexto. Así, dentro de las principales ventajas de esta plataforma, se encuentra la compatibilidad con múltiples sensores de tipo fisiológico y el acceso a la información de dichos sensores de manera inalámbrica.

7.2 Trabajos futuros

Como trabajos futuros derivados del presente trabajo de doctorado se tienen:

- Uno de los trabajos futuros derivados del presente proyecto de doctorado está relacionado con la construcción de un entorno de *streaming* adaptativo DASH que permita el soporte de la transmisión de contenidos multimedia en vivo. En este sentido, la herramienta DASH WebM Converter desarrollada en este

trabajo se convierte en un punto de partida importante para realizar los procesos de codificación y adecuación del contenido multimedia DASH en tiempo real.

- Siguiendo con la temática anterior, otro de los trabajos futuros relacionado con la realimentación del entorno de *streaming* adaptativo, involucra mejorar el proceso de estimación del ancho de banda disponible, a través de la monitorización de la red en diferentes puntos de la cadena de distribución del contenido, de manera que se eviten posibles cuellos de botella producto de la estimación local del ancho de banda en el lado del cliente.
- Así mismo dentro de los trabajos futuros relacionados con el presente proyecto de doctorado, se propone mejorar el buscador de contenidos multimedia, mediante la vinculación de la información obtenida a partir de variables del contexto de usuario. Lo anterior teniendo en cuenta que el presente trabajo se enfocó en enriquecer la recomendación de contenidos, más no la búsqueda de los mismos.
- De igual modo, se pretende ampliar el funcionamiento de la herramienta de codificación automática Dash WebM Converter, de manera que se pueda extender la funcionalidad de codificación a contenidos multimedia adaptativos en formato MPEG4, puesto que en el presente trabajo se consideraron contenidos multimedia en formato WebM. Así mismo, es posible abordar la optimización de dicha herramienta mediante las configuraciones que ofrece FFMPEG, a través de la dedicación de núcleos del procesador para cada hilo de codificación.
- Otro de los trabajos futuros derivados del presente proyecto, está relacionado con la consideración y estudio de un conjunto de nuevas variables de tipo fisiológico como pueden ser: la respiración, las señales cerebrales, la temperatura corporal, entre otras. Lo anterior, con miras a mejorar la precisión y la relevancia de las sugerencias generadas por el sistema de recomendaciones basado en contexto propuesto en el presente trabajo.
- De igual modo, otro de los trabajos futuros a considerar está relacionado con extender el estudio emocional de los contenidos musicales a partir de las letras de las canciones, de tal modo que se pueda realimentar el dataset de contenidos multimedia musicales generado a partir de los parámetros musicales de arousal y valence.
- Así mismo, otro de los trabajos futuros asociado a la presente investigación, está relacionado con el estudio de otros tipos de contenido de video diferente al musical, de tal modo que considerando el audio, las imágenes y el texto contenidos en el mismo, sea posible predecir las emociones de dicho contenido.
- De otra parte, es posible enriquecer el sistema de recomendaciones basado en contexto desarrollado, mediante la vinculación del análisis emocional de los datos aportados por las redes sociales. Lo anterior se convierte en una oportunidad también, para enriquecer el servicio de video bajo demanda, de tal modo que las valoraciones de usuario estén centradas en emociones, tal como la red social Facebook lo ha hecho recientemente.

- En este trabajo se implementó un recomendador clásico bajo la aproximación basada en contenido, por lo que es posible mejorar la precisión de dicho sistema, mediante la inclusión de una aproximación basada en filtros colaborativos. De esta manera se espera que las sugerencias generadas por el recomendador basado en contexto puedan mejorar.
- Un reto a considerar a futuro como extensión del presente trabajo, es el análisis emocional de contenidos multimedia de mayor duración, puesto que según los experimentos realizados en el presente trabajo de doctorado, existen canciones que debido a su duración pueden presentar varias emociones.

7.3 Aportes investigativos

Como aportes investigativos del presente trabajo de doctorado se destacan:

- Un sistema de recomendaciones basado en contexto para contenidos multimedia musicales de video, el cual permite la recomendación de contenidos musicales de video a partir de variables de contexto de usuario como la voz, la VFC y el GSR. Este recomendador permite mejorar el problema de arranque en frío y entregar un conjunto de sugerencias más precisas que las aportadas por los sistemas de recomendaciones clásicos.
- Un sistema de recomendaciones clásico, soportado en un nuevo clasificador para contenidos multimedia musicales, el cual considera como atributos parámetros como el arousal y la valence de los contenidos multimedia musicales.
- Un dataset de contenidos multimedia musicales, el cual clasifica los contenidos de acuerdo al modelo de 5 emociones propuesto en el presente trabajo y teniendo en cuenta las propiedades musicales de arousal y valence.
- La adaptación de un modelo de emociones a partir del modelo de Rusel, el cual permite la clasificación de contenidos multimedia musicales en 5 emociones básicas. Estos contenidos multimedia fueron clasificados según el análisis de los parámetros de arousal y valence del contenido musical.
- Un método computacional para la inferencia de emociones a partir de variables de contexto como VFC y GSR, el cual permitió la asociación de emociones de entrada con los contenidos multimedia musicales clasificados y agrupados en el dataset generado.
- Un escenario de *streaming* adaptativo que permite el soporte de las tareas de codificación, segmentación, difusión y consumo de contenidos multimedia adaptativos DASH.
- Una herramienta para la codificación de contenidos multimedia DASH, la cual permite optimizar y automatizar en el lado del servidor los procesos de adecuación, segmentación y creación del archivo descriptor MPD.

- Un servicio de video bajo demanda para la transmisión mediante el estándar DASH de contenidos multimedia musicales, los cuales han sido previamente clasificados de acuerdo al modelo de arousal valence.
- Un módulo hardware-software para la captura y procesamiento de variables del contexto haciendo uso de tecnologías de hardware y software libre.

7.4 Publicaciones y registros software

7.4.1 Publicaciones en Congresos

- Artículo corto titulado: “Propuesta de un sistema de recomendaciones basado en contexto para contenidos multimedia”, presentado en el marco del 11 Congreso Colombiano de Computación (11CCC), desarrollado en Popayán-Colombia del 27 al 30 de Septiembre de 2016. (Premiado como mejor artículo corto del congreso).
- Artículo largo titulado: “Coding Multimedia Content using DASH Standard”, presentado en el marco del 11 Congreso Colombiano de Computación (11CCC), desarrollado en Popayán-Colombia del 27 al 30 de Septiembre de 2016.
- Artículo largo titulado: “Recomendador basado en el clasificador Naive-Bayes para comunidades académicas virtuales en TV Móvil”, presentado en el marco de las Jornadas Iberoamericanas sobre Aplicaciones y Usabilidad de la TV Digital Interactiva – JAUTI 2013, desarrollado en Cordoba-España entre el 9 al 16 de Septiembre de 2013.
- Artículo largo titulado: “Recommender Based on Folksonomy for Virtual Academic Communities in Mobile TV”, presentado en el marco del congreso Colcom 2012, desarrollado en Cali-Colombia del 16 al 18 de Mayo de 2012. (Artículo disponible en el explorador de la IEEE).
- Artículo largo titulado: “Escenario de Experimentación y Despliegue de servicios interactivos de TV Móvil”, presentado en el marco del I Congreso Internacional de Telecomunicaciones – CITTEL 2012, desarrollado en Bucaramanga-Colombia del 1 al 3 de Septiembre de 2011.
- Artículo largo titulado: “Esquema de servicios para Televisión Digital Interactiva, basados en el Protocolo REST-JSON”, presentado en el marco del Congreso Internacional de Telemática – CITA 2011, desarrollado en Gramado-Brazil del 16 al 18 de Mayo de 2011.
- Artículo largo titulado: “Modelado de tráfico para un servicio de Videochat”, presentado en el marco de las X Jornadas de Ingeniería Telemática – JITTEL 2011, desarrolladas en Santander-España del 16 al 30 de Septiembre de 2011.

- Artículo largo titulado: “Diseñando Interfaces Gráficas Usables de Aplicaciones en Entornos de TDi”, presentado en el V Congreso Latinoamericano de Interacción Humano Computador – CLIHC 2011, desarrollado en Porto de Galinhas – Brazil entre el 25 y el 28 de Octubre de 2011.
- Artículo largo titulado: “Recomendaciones para la Construcción de Juegos en Entornos de TDi Basados en el Estándar MHP”, presentado en el marco del Congreso de Tecnologías Emergentes en Telecomunicaciones y Telemática – TET 2010, desarrollado en Popayán-Colombia entre el 10 y el 12 de Junio de 2010.
- Artículo largo titulado: “Directrices para el Soporte de comunidades Académicas Virtuales en Entornos de TDi”, presentado en el marco del VI Congreso Internacional de Telecomunicaciones – CITTEL 2010, desarrollado en La Habana – Cuba entre el 29 de Noviembre y el 3 de Diciembre de 2010.
- Artículo corto titulado: “Architecture of reference for interactive Mobile TV services supported in personalization and semantic search”, presentado en el marco de la 5th Conference of the Euro-American Association on Telematics and Information Systems - EATIS 2010, desarrollado en Panamá entre el 22 y el 24 de Septiembre 2010.
- Artículo largo titulado: “Arquitectura para el Soporte de Comunidades Académicas Virtuales en Ambientes de TDi”, presentado en el marco del V Congreso Colombiano de Computación – 5CCC, desarrollado en Cartagena de Indias – Colombia entre el 13 y el 15 de Abril de 2010.

7.4.2 Publicaciones en Revistas

- Artículo titulado: “Sistema Hardware-Software para reconocimiento de emociones a partir de señales fisiológicas”, sometido a revisión en la revista “Ingeniería Biomédica” de la Escuela de Ingeniería de Antioquia -EIA- Universidad CES de Medellín – Colombia (indexada como categoría B de Colciencias). ISSN: 1909–9762.
- Artículo titulado: “Escenario para la transmisión de *streaming* adaptativo DASH”, aprobado para publicación en el Vol. 18-1 (2017) de la “Revista Colombiana de Computación” de la Universidad Autónoma de Bucaramanga-Colombia (indexada como categoría C de Colciencias). ISSN: 1657-2831.
- Artículo titulado: “Dataset de contenidos musicales de video, basado en emociones”, publicado en el Vol. 7-2 (2016) de la revista “Ingeniería, Ciencia, Tecnología e Innovación” de la Universidad San Buenaventura de Medellín-Colombia (indexada como categoría C de Colciencias). ISSN: 2027-5846.
- Artículo titulado: “Servicio de video bajo demanda basado en emociones”, publicado en en el Vol. 14-38 (2016) de la revista “Sistemas & Telemática” de la

Universidad Icesi de Cali-Colombia (indexada como categoría C de Colciencias). ISSN: 1692-5238.

- Artículo titulado: “Construcción y Evaluación de Servicios Interactivos en entornos de TVDi” en el Vol. 21-1 (2016) de la revista Ingeniería de la Universidad Distrital Francisco José de Caldas de Bogotá-Colombia (Indexada como categoría B de Colciencias). ISSN: 0121-750X.
- Artículo titulado: “Testing environment for video streaming support using open source tools” en el Vol. 34-2 (2016) de la revista “Ingeniería y Desarrollo” de la Universidad del Norte de Barranquilla-Colombia (Indexada como categoría A2 de Colciencias). ISSN: 0122-3461.
- Artículo titulado: “Sistema de recomendaciones de contenidos multimedia para comunidades académicas en entornos de TDi”, publicado en el Vol. 9-17 (2015) de la revista: “Entre Ciencia e Ingeniería” de la Universidad Católica de Pereira-Colombia (Indexada como categoría B de Colciencias). ISSN: 1909-8367.
- Artículo titulado: “Stress tests for videostreaming services based on RTSP protocol” en el Vol. 19-46 (2015) de la revista “Tecnura” de la Universidad Distrital Francisco José de Caldas de Bogotá-Colombia (Indexada como categoría A2 de Colciencias). ISSN: 0123-921X.
- Artículo titulado: “Directrices para el diseño de aplicaciones usables en entornos de televisión digital interactiva”, publicado en el Vol. 17-2 (2014) de la Revista “Ingeniería y Universidad” de la Universidad Javeriana de Bogotá-Colombia (Indexada como categoría A2 de Colciencias). ISSN: 0123-2126.
- Artículo titulado: “Arquitectura para el soporte de Comunidades Académicas Virtuales en ambientes de Televisión Digital Interactiva”, publicado en el Vol. 6-13 (2013) de la Revista “Formación Universitaria” del Centro de Información Tecnológica Tecnológica de la Serena – Chile (Homologada como categoría A1 de Colciencias). ISSN: 0718-5006.
- Artículo titulado: “Entorno de pruebas para servicios interactivos de Televisión Móvil, basada en el estándar DVB-H”, en el número 7 (2012) de la revista “Lámpsakos” de la Fundación Universitaria Luis Amigó de Medellín-Colombia (Indexada como categoría C de Colciencias). ISSN: 2145-4086.
- Artículo titulado: “Surveying Mobile Television”, en el Vol. 3-1 (2011) de la revista “Internacional Journal of Communication Networks and Information Security (IJNIS)” de la Universidad de Kohat – Pakistan. ISSN: 2076-0930.
- Artículo titulado: “Esquema de Servicios para Televisión Digital Interactiva, Basados en el protocolo REST-JSON”, en el Vol. 6-1 (2011) de la revista “Cadernos de Informática” de la Universidade Federal do Rio Grande do Sul de Porto Alegre-Brasil. ISSN: 1519-132X.
- Artículo titulado: “Escenario de Pruebas para el Soporte de Videostreaming Usando Herramientas Libres”, en el Vol. 7 (2014) de la Revista Investigación, Tecnología y Casa (I+T+C) de la Corporación Universitaria Unicofacauca de Popayán-Cauca. ISSN: 1909-5775.

7.4.3 Libros

- Libro titulado: “Arquitectura para el despliegue de servicios interactivos de TV Móvil”, el cual se encuentra en proceso de edición en la Editorial de la Universidad del Cauca.

7.4.4 Registros Software

Dentro de los registros software generados se encuentran (ver Anexo J):

- Registro del producto software titulado: “Herramienta web para la medición de estrés”, ante la Dirección Nacional de Derechos de Autor (DNDA). Libro 13, Tomo 52, Partida 136. 19 de Febrero de 2016.
- Registro del producto software titulado: “Herramienta para la clasificación de contenidos multimedia musicales”, ante la Dirección Nacional de Derechos de Autor (DNDA). Libro 13, Tomo 52, Partida 153. 9 de Marzo de 2016.
- Registro del producto software titulado: “Dataset de contenidos musicales de video, basado en emociones”, ante la Dirección Nacional de Derechos de Autor (DNDA). Libro 13, Tomo 54, Partida 129. 6 de Julio de 2016.
- Registro del producto software titulado: “Sistema de recomendaciones para el servicio de VoD, basado en la inferencia de emociones a partir de variables del contexto de usuario”, ante la Dirección Nacional de Derechos de Autor (DNDA). Libro 13, Tomo 54, Partida 128. 6 de Julio de 2016.

Referencias

- [1] Wilmar Campo, José Arciniegas, Roberto García, and David Melendi, "Análisis de Tráfico para un Servicio de Vídeo bajo Demanda sobre Redes HFC usando el Protocolo RTMP," *Información tecnológica*, vol. 21, no. 6, pp. 27-48, 2010.
- [2] Sandvine Intelligent Broadband Networks, "Global Internet Phenomena Report," *Sandvine*, vol. 2H 2013, pp. 5,6,9,10,23, 2013. [Online].
<https://www.sandvine.com/downloads/general/global-internet-phenomena/2013/2h-2013-global-internet-phenomena-report.pdf>
- [3] Christopher Mueller, Stefan Lederer, Christian Timmerer, and Hermann Hellwagner, "Dynamic adaptive streaming over HTTP/2.0," *Alpen-Adria-Universität Klagenfurt*, 2013.
- [4] M. Dabrowski, J. Gromada, and H. Moustafa, "Context-Awareness for IPTV Services Personalization," *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on*, pp. 37-44, 2012.
- [5] International Telecommunication Union ITU-T, "Supplement on IPTV service use cases - ITU-T Y-series Recommendations – Supplement 5," Mayo 2008.
- [6] Open IPTV Forum. (2013, Mayo) Specification Functional Architecture v2.2.
- [7] F. Boronat, M. García, and J. Lloret, *IPTV, la televisión por Internet*. Malaga-España: Vértice, 2008.
- [8] International Telecommunication Union ITU-T. (2008, Mayo) ITU-T Y-series Recommendations.
- [9] Jose Arrojo, *Configuración de la televisión interactiva: de las plataformas digitales a la TDT*. La Coruña - España: Gesbiblo, 2008.
- [10] Gilbert Held, *Understanding IPTV*.: CRC Press, 2006.
- [11] Stev Mack,.: John Wiley & Sons, 2002, p. 869.

- [12] David Costilla Mateos and Samuel Reaño Montoro, "Streaming de Audio/Video. Protocolo RTSP," *enginy@eps-Curs 2007/2008*, p. 15, 2008.
- [13] Christopher Müller and Christian Timmerer, "A VLC Media Player Plugin enabling Dynamic Adaptive Streaming over HTTP," *Alpen-Adria-Universität Klagenfurt - Multimedia Communication*, pp. 723-724, 2011.
- [14] Mayo de 2008. International Telecommunication Union ITU-T 5, "Supplement on IPTV service use cases," Mayo de 2008.
- [15] K. Pripuzic et al., "Building an IPTV VoD Recommender System: An Experience Report," *Telecommunications (ConTEL), 2013 12th International Conference on*, pp. 155-162, 2013.
- [16] Songbo Song, Hassnaa Moustafa, and Hossam Afifi, "IPTV Services Personalization Using Context-Awareness," *Informatica*, pp. 13-20, 2011.
- [17] Roberto Turrin and Paolo Cremonesi, "Recomender Systems for Interactive TV," in *EuroITV 2010*, Tampere, Finland, 2010.
- [18] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich, *Recommender Systems: An Introduction.*: Cambridge, 2010.
- [19] Ronald Yager, "Fuzzy logic methods in recommender systems," *Fuzzy Sets and Systems*, vol. 136, no. 2, pp. 133-149, 2003.
- [20] C. Porcel, A.G. López-Herrera, and E. Herrera, "A recommender system for research resources based on fuzzy linguistic modeling," *Expert Systems with Applications: An International Journal*, vol. 36, no. 3, pp. 5173-5183, 2009.
- [21] C. Porcel, J. M. Moreno, and E Herrera-Viedma, "A multi-disciplinar recommender system to advice research resources in University Digital Libraries," *Expert Systems with Applications: An International Journal*, vol. 36, no. 10, pp. 12520-12528 , 2009.
- [22] C. Porcel, A Tejada, M. Martínez, and Herrera E., "A hybrid recommender system for the selective dissemination of research resources in a Technology Transfer Office," *Information Sciences*, vol. 184, no. 1, pp. 1-19, 2012.
- [23] M. Zanker, Univ. Klagenfurt, D. Jannach, S. Gordea, and M. Jessenitschnig, "Comparing Recommendation Strategies in a Commercial Context ," *Intelligent Systems, IEEE*, vol. 22, no. 3, pp. 69 - 73 , 2007.
- [24] Prem Melville, Raymond J. Mooney, and Ramadass Nagarajan, "Content-Boosted Collaborative Filtering for Improved Recommendations," in *Proceedings of the Eighteenth National Conference on Artificial Intelligence(AAAI-2002)*, Edmonton, Canada, 2002.

- [25] Amit Thawani, Srividya Gopalan, and V Sridhar, "Context Aware Personalized Ad Insertion in an Interactive TV Environment," in *4th Workshop on Personalization*, 2004.
- [26] G. Chen and D. Kotz, "A survey of context-aware mobile computing research," 2000.
- [27] Han-Saem Park, Ji-Oh, Cho Yoo, and Sung-Bae, "A Context-Aware Music Recommendation System Using Fuzzy Bayesian Networks with Utility Theory," *Fuzzy Systems and Knowledge Discovery*, vol. 4223, pp. 970-979, 2006.
- [28] Xinxi Wang, David Rosenblum, and Ye Wang, "Context-aware mobile music recommendation for daily activities," in *MM '12 Proceedings of the 20th ACM international conference on Multimedia*, New York, NY, USA, 99-108.
- [29] Seungmin Rho, Byeong-jun Han, and Eenjun Hwang, "SVR-based music mood classification and context-based music recommendation," in *Proceedings of the 17th ACM International Conference on Multimedia*, Beijing, China, 2009.
- [30] Ja-Hwung Su, Hsin-Ho Yeh, P.S. Yu, and V.S. Tseng, "Music Recommendation Using Content and Context Information Mining," *Intelligent Systems, IEEE*, vol. 25, no. 1, pp. 16-26, 2010.
- [31] 3GPP TS 26.234. (2010) Transparent end-to-end Packet-switched Streaming Service (PSS). [Online]. <http://www.3gpp.org/DynaReport/26234.htm>
- [32] ISO/IEC 23009-1:2012, "Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats," *ISO/IEC 2012*, p. 3, Jan. 2014. [Online]. <http://standards.iso.org/ittf/licence.html>
- [33] Microsoft Corporation. (2009, Mar.) IIS Smooth Streaming Technical Overview. [Online]. <http://www.microsoft.com/en-us/download/details.aspx?id=17678>
- [34] W May and R Pantos. (2011, Sep.) HTTP Live Streaming draft-pantos-http-live-streaming-07. [Online]. <http://tools.ietf.org/html/draft-pantos-http-live-streaming-07>
- [35] Adobe. (2012) HTTP Dynamic Streaming. [Online]. <http://www.adobe.com/products/hds-dynamic-streaming.html>
- [36] CSI Analysing Converting technologies, "DASH it all," *CSI Analysing Converting technologies*, p. 39, May 2013. [Online]. <http://www.newtec.eu/frontend/files/userfiles/files/Newtec-in-News/2013-05-24-CSI-QandA-Interview-Talking-Ka-band-with-Newtec.pdf>
- [37] Beatriz Casín Núñez, "DASH: Un estándar MPEG para streaming sobre HTTP," *Facultat d'Informàtica de Barcelona - Universitat Politècnica de Catalunya*, 2013.

- [38] José Daniel Gambín Tomasi, "Desarrollo de un servicio de televisión interactiva HbbTV según el estándar ETSI TS 102 796 v1.1.1 (JUN 2010)," *UNIVERSIDAD POLITÉCNICA DE CARTAGENA - E.T.S. Ingeniería de Telecomunicación*, pp. 111-112, 2012.
- [39] F.S. da Silva, L.G.P. Alves, and G. Bressan, "PersonalTVware: A Proposal of Architecture to Support the Context-aware Personalized Recommendation of TV Programs," in *EuroITV 2009*, Leuven, Belgium, 2009.
- [40] Stuart Cunningham, Stephen Caulder, and Vic Grout, "Saturday Night or Fever? Context-Aware Music Playlists," in *2008, Proceedings of the 3rd International Audio Mostly conference on Sound in Motion*.
- [41] Hongguang Zhang and Shibao Zheng, "Personalized TV program recommendation based on TV-anytime metadata ," in *Consumer Electronics, 2005. (ISCE 2005). Proceedings of the Ninth International Symposium on* , St. Louis Missouri, USA, 2005.
- [42] D Asanov, "Algorithms and methods in recommender systems," in *Berlin Institute of Technology*, Berlin, Germany, 2011.
- [43] G. E. Chanchí, W. Campo, J. Amaya, and J. Arciniegas, "Esquema de Servicios para Televisión Digital Interactiva, basados en el protocolo REST-JSON," in *CITA 2011*, Gramado - Brasil, 2011.
- [44] Zhu Li and Imed Bouazizi, "Light Weight Content Fingerprinting for Video Playback Verification in MPEG DASH," *Samsung Research America*, 2013.
- [45] Christopher Muller et al., "Open Source Column: Dynamic Adaptive Streaming over HTTP Toolset," *ACM SIGMM Records*, Sep. 2013.
- [46] Christian Timmerer and Carsten Griwodz, "Dynamic Adaptive Streaming over HTTP: From Content Creation to Consumption," *MM'12*, Noviembre 2012.
- [47] Beatriz Nuñez, "DASH: Un estándar MPEG para streaming sobre HTTP," Barcelona, 2013.
- [48] I. M. Pao and Ming-Ting Sun, "Encoding stored video for streaming applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 11, no. 2, pp. 199-209, 2001.
- [49] R. Tortosa, J. M. Jimenez, J. R. Diaz, and J. Lloret, "Optimal codec selection algorithm for audio streaming," Valencia-España, 2014.
- [50] B. Rainer, S. Lederer, C. Müller, and C. Timmerer, "A Seamless Web Integration of Adaptive HTTP Streaming," in *20th European Signal Processing Conference (EUSIPCO 2012)*, Bucharest, 2012.

- [51] C. Muller et al., "Open source column: dynamic adaptive streaming over HTTP toolset," *ACM SIGMultimedia Records*, vol. 5, no. 1, pp. 11-13, 2013.
- [52] W. Hamidouche, G. Cocherel, J. Le Feuvre, M. Raulet, and O. Deforges, "4k real time video streaming with SHVC decoder and GPAC player," in *Multimedia and Expo Workshops (ICMEW), 2014 IEEE International Conference on*, Chengdu, 2014.
- [53] Nan Zheng and X. Liu, "Load balance optimization of a Red5 cluster in the mobile classroom project," in *Natural Computation (ICNC), 2013 Ninth International Conference on*, Shanghai-China, 2013.
- [54] N. Vun and M. Ansary, "Implementation of an embedded H.264 live video streaming system," in *IEEE International Symposium on Consumer Electronics (ISCE 2010)*, Braunschweig, 2010.
- [55] G. A. Hoffmann, K. S. Trivedi, and M. Malek, "A best practice guide to resources forecasting for the apache webserver," in *12th IEEE International Symposium Pacific Rim Dependable Computing (PRDC'06)*, Riverside-USA, 2006.
- [56] J. Bailey. (2011, Diciembre) Live Video Streaming from Android-Enabled Devices to Web Browsers. [Online]. <http://scholarcommons.usf.edu/etd/2995/>
- [57] WebM Proyect. Instructions to playback Adaptive WebM using DASH. [Online]. <http://wiki.webmproject.org/adaptive-streaming/instructions-to-playback-adaptive-webm-using-dash>
- [58] Peter Andersen and Laura Guerrero, *Handbook of communication and emotion*. California : Academic Press, 1997.
- [59] Yi Yang and Homer Chen, *Music Emotion Recognition*. Taiwan: CRC Press, 2011.
- [60] Jorge Alberto Rivera Guerra. (2011) redalyc. [Online]. <http://www.redalyc.org/articulo.oa?id=94419100008>
- [61] Tristan Jehan and Brian Whitman. (2005, Junio) The Echo nest. [Online]. <http://the.echonest.com/>
- [62] Paul Kleinginna and Anne Kleinginna, "A Categorized List of Emotion Definitions, with Suggestions for a Consensual Definition," in *Motivation and Emotion*. Georgia: Plenum Publishing Corporation, 1981, pp. 345-379.
- [63] Mari Jones and Richard Fay, *Music Perception*. Ohio: Springer, 2010.
- [64] James Russell, "A Circumplex Model of Affect," in *Personality and Social Psychology*. Vancouver : APA journals , 1980, pp. 1161-1178.

- [65] Kate Hevner, "Experimental Studies of the Elements of Expression in Music," in *The American Journal of Psychology*. Illinois: Jstor, 1936, pp. 246-268.
- [66] Robert Plutchik, "The Nature of Emotions," in *American Scientist*. Florida: American Scientist, 2001, pp. 344-350.
- [67] Vincent Castaignet and Vavrille Frederic. (2006) musicoverly. [Online]. <http://musicoverly.com/api/doc/documentation.php>
- [68] Ti Kan and Steve Scherf. (1998) Gracenote. [Online]. <https://developer.gracenote.com/web-api>
- [69] Jonathan Posner and James Russell, "The circumplex model of affect: An integrative approach to affective neuroscience, cognitive development, and psychopathology," in *Dev Psychopathol*. New York: NIH, 2005, pp. 715–734.
- [70] Owen Meyers, *A Mood-Based Music Classification and Exploration System*. Massachusetts: Massachusetts Institute of Technology, 2004.
- [71] X Navarro, "Fisiología del sistema nervioso autónomo," *Revista de neurología*, vol. XXXV, no. 6, pp. 553-562, 2002.
- [72] Sylvia D. Kreibig, "Autonomic nervous system activity in emotion: A review," in *Biological Psychology*. Geneva: ELSEVIER, 2010, pp. 394–421.
- [73] John E Hall, *Tratado de fisiología médica*. Barcelona : ELSEVIER, 2011.
- [74] Gil Roldas, "Archivos de Medicina del Deporte," in *Variabilidad de la Frecuencia Cardíaca*. Barcelona: AMD, 2008, pp. 41-127.
- [75] Sansanee Boonnithi and Phongsuphap, "Comparison of Heart Rate Variability Measures for Mental Stress Detection," in *Computing in cardiology*. Bangkok: IEEE, 2011, pp. 85-88.
- [76] R.M Bayevsky, G.G Ivanov, and L.V Chireykin, *HRV Analysis under the usage of different electrocardiography systems*. Moscow: These methodical recommendations are prepared according to the order of the Committee of Clinic Diagnostic Apparatus and the Committee of New Medical Techniques of Ministry of Health of Russia, 2002.
- [77] Cornelia Kappeler-Setz, *Multimodal Emotion and Stress Recognition*. Zurich: ETH ZURICH, 2012.
- [78] Feng-Tso Sun and Cynthia Kuo, "Activity-Aware Mental Stress Detection Using Physiological Sensors," in *Mobile Computing, Applications, and Services*. Santa Clara: Springer, 2012, pp. 211-230.

- [79] Juan Carmona, "Desarrollo de un segmentador fonético automático para habla expresiva basado en modelos ocultos de Markov," Departamento de Ingeniería Electrónica, Universidad Politécnica de Madrid, Madrid, Proyecto de fin de carrera 2012.
- [80] Florian Eyben, Martin Wollmer, and Bjorn Sculler, "openEar - Introducing the Munich Open-Source Emotion and Affect Recognition Toolkit," in *Affective Computing and Intelligent Interaction and Workshops*, Amsterdam, Netherlands, 2009, pp. 1-6.
- [81] Armin Ronacher. (2013, Junio) Flask Web Development, one drop at a time. [Online]. <http://flask.pocoo.org/>
- [82] Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul Kantor, *Recommender System Handbook*.: Springer, 2010.
- [83] "Learning naive Bayes classifiers from positive and unlabelled examples with uncertainty," in *International Journal of Systems Science*. Yangling: Taylor & Francis , 2012, pp. 1805-1825.
- [84] N. Haider and R. M. Rahman M. M. R. Siddiquee, "A fuzzy based recommendation system with collaborative filtering," in *8th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*, Dhaka, 2014.
- [85] Luiz Moutinho and Graeme D. Hutcheson, "Ordinary Least-Squares Regression," in *The SAGE Dictionary of Quantitative Management Research*.: SAGE Publications, 2011, pp. 224-228.
- [86] Ulrich Kohler and Frauke Kreuter, "Describing data using statistics," in *Data Analysis Using Stata, Third Edition*. Texas: Stata Press, 2012, pp. 173-186.
- [87] Jorn Bakker, Mykola Pechenizkiy, and Natalia Sidorova, "What's your current stress level? Detection of stress patterns from GSR sensor data," in *IEEE International Conference on Data Mining Workshops*, Eindhoven, 2011.