

Universidad de la República
Facultad de Ingeniería

Instituto de computación

Informe de Proyecto de Grado 2012

**Sistema de Recomendaciones Grupales
para Películas**

Integrantes

Guillermo Fernández
Fernando Olivera
Waldemar López

Tutores

Dr. Pablo Rodríguez Bocca
Ing. Bruno Rienzi

Resumen

Cada vez es mayor la oferta de contenido multimedia a través de Internet, existiendo una gran cantidad de contenido disponible desde diversos proveedores. Por este motivo surge la necesidad de ordenarlos a través de recomendaciones que permitan al consumidor elegir el contenido que desea ver. En el proyecto abordamos principalmente este problema: qué contenido quiere ver el consumidor en forma individual o como parte de un grupo. Por otro lado se ha masificado el uso de los dispositivos con acceso a Internet. Dentro de la gama de opciones tenemos, por ejemplo: PCs, teléfonos móviles, tabletas, televisores inteligentes, etc. Esto plantea el desafío de encontrar el aporte que pueden hacer estos dispositivos en un sistema de recomendación grupal.

Durante el proyecto estudiamos los diferentes algoritmos de recomendaciones individuales y grupales existentes, obteniendo la taxonomía de los mismos. Además, investigamos los proveedores de contenidos más populares, en particular, los proveedores de video a demanda. Clasificamos los diferentes dispositivos existentes y nos introdujimos en el concepto de primera y segunda pantalla.

En base a esto construimos dos componentes: un servicio de recomendaciones genérico y un sistema de recomendaciones de películas. El servicio de recomendaciones genérico, instanciado con una carga inicial de datos obtenidos a partir de un proyecto de la Universidad de Minnesota: MovieLens. Es un proyecto de investigación académico donde desarrollaron un sitio Web de recomendaciones de películas y tienen disponible los datos de las calificaciones de los usuarios. En nuestro servicio de recomendaciones utilizamos el algoritmo de recomendación individual Slope One, obteniendo los resultados de rendimientos esperados para los datos cargados. El otro componente es un caso de estudio un sistema de recomendaciones grupales de películas, donde los datos de las películas fueron sincronizados desde Internet Movie DataBase (IMDb). El principal objetivo del sistema es recomendar películas para un grupo de personas, en particular tiene una vista para dispositivos móviles sobre HTML5 que recomienda películas dentro de una cartelera cinematográfica para un grupo de personas. Además, el sistema integra diferentes tecnologías dentro de una arquitectura orientada a servicios.

Palabras claves: sistemas de recomendación, recomendación individual, recomendación grupal, filtrado colaborativo, Slope One, multidispositivo, integración de sistemas, reproducción de contenido multimedia.

Índice General

1 INTRODUCCIÓN.....	1
1.1 MOTIVACIÓN.....	1
1.2 OBJETIVOS.....	1
1.3 RESULTADOS Y CONTRIBUCIONES.....	2
1.4 ORGANIZACIÓN DEL DOCUMENTO.....	2
2 ESTADO DEL ARTE.....	5
2.1 RECOMENDACIONES INDIVIDUALES.....	5
2.1.1 Notación.....	6
2.1.2 Filtro Colaborativo.....	7
2.1.3 Sistema de Recomendación Basado en Contenido.....	11
2.1.4 Otros Algoritmos.....	13
2.1.5 Estudios comparativos.....	13
2.1.6 Posibles deficiencias.....	15
2.1.7 Resumen de Técnicas de recomendación.....	16
2.2 RECOMENDACIONES GRUPALES.....	17
2.2.1 La televisión como sistema adaptativo al medio.....	18
2.2.2 Estrategias de recomendaciones grupales.....	18
2.2.3 Consideraciones sobre privacidad.....	20
2.3 SISTEMAS DE RECOMENDACIÓN DE REFERENCIA.....	21
2.4 PROVEEDORES DE VIDEO BAJO DEMANDA.....	22
2.4.1 Internet video archive.....	22
2.4.2 Netflix.....	23
2.4.3 Crackle.....	23
2.4.4 Redbox.....	23
2.5 MULTIDISPOSITIVO.....	24
2.5.1 La Televisión en todos lados.....	24
2.5.2 La primera pantalla / pantalla principal.....	25
2.5.3 Segunda pantalla.....	26
2.5.4 Pautas de diseño para interfaces de usuario.....	26
3 SOLUCIÓN PROPUESTA.....	29
3.1 ANÁLISIS DE REQUERIMIENTOS.....	31
3.2 ARQUITECTURA DEL SISTEMA.....	32
3.2.1 Componentes principales.....	33
3.2.2 Vista de despliegue.....	37
3.3 ALGORITMO DE RECOMENDACIÓN.....	39
3.3.1 Algoritmo Grupal.....	39
3.3.2 Algoritmo Individual.....	40
3.4 CASOS DE USO.....	41
3.4.1 CU6 – Recomendar Película Usuario.....	43
3.4.2 CU9 – Aceptar Invitaciones.....	47
3.5 IMPLEMENTACIÓN.....	51
3.5.1 Carga y sincronización de datos.....	51
3.5.2 Proveedores de contenido.....	57
3.5.3 Problemas Técnicos.....	58

4 PRUEBAS REALIZADAS.....	61
4.1 PRUEBAS DEL SISTEMA.....	61
4.1.1 Pruebas funcionales.....	61
4.1.2 Pruebas dispositivos móviles.....	61
4.2 PRUEBAS DE ALGORITMOS DE RECOMENDACIONES.....	62
4.2.1 Recursos.....	63
4.2.2 Resultados.....	63
4.2.3 Pruebas con voluntario.....	64
4.2.4 Conclusiones.....	65
5 CONCLUSIONES Y TRABAJOS FUTUROS.....	67
5.1 CONCLUSIONES.....	67
5.2 TRABAJOS FUTUROS.....	69
6 GLOSARIO.....	73
BIBLIOGRAFÍA	75
ANEXO A ESTRATEGIAS DE RECOMENDACIONES GRUPALES.....	83
A.1 ESTRATEGIA CALIFICACIÓN PLURAL.....	83
A.2 ESTRATEGIA UTILITARIA.....	83
A.2.1 Aditiva (o promedio).....	83
A.2.2 Multiplicativa.....	84
A.3 ESTRATEGIA RECuento BORDA.....	85
A.4 ESTRATEGIA REGLA DE COPELAND.....	85
A.5 ESTRATEGIA CALIFICACIÓN POR APROBACIÓN.....	86
A.6 ESTRATEGIA DEL MENOS FELIZ.....	87
A.7 ESTRATEGIA DEL MÁS FELIZ.....	87
A.8 ESTRATEGIA MEDIA SIN INFELICES.....	88
A.9 ESTRATEGIA CON JUSTICIA.....	88
A.10 ESTRATEGIA DEL INDIVIDUO MÁS RESPETADO O DICTADURA.....	89
ANEXO B SISTEMAS DE RECOMENDACIÓN DE REFERENCIA.....	91
B.1 IMDB (INTERNET MOVIE DATABASE).....	91
B.1.1 Funcionalidades.....	91
B.1.2 Análisis técnico	92
B.1.3 Análisis Crítico.....	95
B.2 ROTTEN TOMATOES Y FLIXTER.....	95
B.2.1 Funcionalidades.....	96
B.2.2 Análisis técnico.....	97
B.2.3 Análisis Crítico.....	99
B.3 MOVIELENS.....	99
B.3.1 Funcionalidades.....	100
B.3.2 Análisis Crítico.....	101
B.4 HULU.....	101
B.4.1 Funcionalidades.....	102
B.4.2 Análisis técnico.....	104
B.4.3 Análisis Crítico.....	107
B.5 NANOCROWD.....	107
B.5.1 Funcionalidades.....	108
B.5.2 Análisis técnico.....	109
B.5.3 Análisis Crítico.....	109

B.6 JINNI.....	110
B.6.1 Funcionalidades.....	110
B.6.2 Análisis técnico.....	112
B.6.3 Análisis Crítico.....	113
B.7 METACRÍTIC.....	114
B.7.1 Funcionalidades.....	114
B.7.2 Análisis técnico.....	115
B.7.3 Análisis Crítico.....	116
ANEXO C TABLAS DE DISPOSITIVOS.....	119
C.1 LA TELEVISIÓN EN TODOS LADOS.....	119
C.2 LA PRIMERA PANTALLA / PANTALLA PRINCIPAL.....	119
C.2.1 Dispositivos para la primera pantalla.....	119
C.3 LA SEGUNDA PANTALLA.....	131
C.3.1 Dispositivos móviles.....	131
ANEXO D PAUTAS DE DISEÑO PARA INTERFACES DE USUARIO.....	147
D.1 PAUTAS GENERALES.....	147
D.1.1 Percepción visual.....	147
D.1.2 Teoría del Color.....	148
D.1.3 Patrones de visualización.....	150
D.1.4 Aspectos culturales.....	150
D.2 PAUTAS DE DISEÑO PARA LA PRIMERA PANTALLA.....	151
ANEXO E MOVIERECOMMENDERAPP – DOCUMENTACIÓN DE API.....	159
E.1 GENERALIDADES.....	159
E.1.1 Diseño RESTFul.....	159
E.1.2 Formato para las peticiones.....	159
E.1.3 Formato de la respuesta.....	159
E.1.4 Tipo de datos para la respuesta.....	160
E.1.5 Formato de fechas.....	160
E.1.6 Autenticación y Autorización.....	160
E.2 OPERACIONES.....	161
E.2.1 Crear un evento.....	161
E.2.2 Obtener el detalle de un evento.....	163
E.2.3 Invitar a un usuario a un evento.....	166
E.2.4 Obtener el detalle de una invitación a un evento.....	167
E.2.5 Obtener el detalle de una invitación a un grupo.....	169
E.2.6 Aceptar unirse a un evento.....	171
E.2.7 Rechazar unirse a un evento.....	173
E.2.8 Obtener la lista de eventos en la que un usuario es miembro.....	174
E.2.9 Obtener la lista de eventos con invitación pendiente para un usuario.....	176
E.2.10 Obtener la lista de eventos en que un usuario es propietario.....	178
E.2.11 Calificar una película.....	180
E.2.12 Obtener la lista de amigos de un usuario.....	181
E.2.13 Indicar gusto/disgusto sobre una entrada de cartelera para el contexto de un evento.....	183
E.2.14 Preferencia de gusto de un usuario sobre una entrada de cartelera en un evento.....	185
E.2.15 Obtener la cartelera cinematográfica.....	187
ANEXO F CASOS DE USO.....	195
F.1 CU1 - REGISTRO DE USUARIO.....	195
F.2 CU2 - LOGIN.....	196
F.3 CU3 - CONSULTAR DATOS DE PELÍCULAS.....	197

F.4 CU4 - REPRODUCIR PELÍCULA.....	198
F.5 CU5 - CALIFICAR PELÍCULA.....	201
F.6 CU6 - RECOMENDAR PELÍCULA USUARIO.....	204
F.7 CU7 - RECOMENDAR PELÍCULA GRUPO.....	211
F.8 CU8 - CREAR EVENTO.....	213
F.9 CU9 - ACEPTAR INVITACIONES.....	213
F.10 CU10 - RECOMENDACIÓN EVENTO.....	215
F.11 CU11 - SINCRONIZAR PROVEEDOR.....	216
F.12 CU12 - SINCRONIZAR CATÁLOGO.....	217
ANEXO G CASOS DE PRUEBA.....	221
G.1 PRUEBAS FUNCIONALES.....	221
<i>G.1.1 Escenarios.....</i>	<i>221</i>
<i>G.1.2 Casos de prueba.....</i>	<i>224</i>
ANEXO H GUÍA DE USO DEL SITIO MOVIERECOMMENDER.....	229
H.1 PASOS A SEGUIR.....	229
<i>H.1.1 Cómo buscar una película para calificarla?.....</i>	<i>229</i>
<i>H.1.2 Como pedir una predicción para cierta película?.....</i>	<i>233</i>
<i>H.1.3 Cómo pedir una recomendación sobre películas que me interesan?.....</i>	<i>235</i>

Lista de ilustraciones

Ilustración 1: Abstracción gráfica de SR.....	5
Ilustración 2: Ejemplo Estrategia Utilitaria.....	19
Ilustración 3: Vista general de la solución.....	30
Ilustración 4: Captura de aplicación móvil (Recomendaciones para un evento).....	31
Ilustración 5: Captura de aplicación Web (Reproducción de película).....	32
Ilustración 6: Arquitectura - Componentes principales.....	33
Ilustración 7: Servidor de recomendación de películas.....	34
Ilustración 8: Diagrama de despliegue.....	38
Ilustración 9: Diagrama de Casos de uso.....	42
Ilustración 10: Diagrama de Secuencia - Recomendar Película Usuario (1 de 2).....	45
Ilustración 11: Diagrama de Secuencia - Recomendar Película Usuario (2 de 2).....	46
Ilustración 12: Diagrama de Secuencia - Aceptar invitación (1 de 3).....	48
Ilustración 13: Diagrama de Secuencia - Aceptar Invitación (2 de 3).....	49
Ilustración 14: Diagrama de Secuencia - Aceptar Invitación (3 de 3).....	50
Ilustración 15: Modelo inicial para el volcado de los ficheros de MovieLens.....	53
Ilustración 16: Modelo normalizado.....	55
Ilustración 17: Ejemplo: Diferencias de promedios de calificaciones en Slope One.....	57
Ilustración 18: Ejemplo Calificación Plural.....	83
Ilustración 19: Ejemplo Estrategia Utilitaria Aditiva.....	84
Ilustración 20: Ejemplo Estrategia Utilitaria Multiplicativa.....	85
Ilustración 21: Ejemplo Estrategia Recuento Borda.....	85
Ilustración 22: Ejemplo Estrategia Regla de Copeland.....	86
Ilustración 23: Ejemplo de Estrategia Calificación por aprobación.....	87
Ilustración 24: Ejemplo Estrategia del menos feliz.....	87
Ilustración 25: Ejemplo Estrategia del más feliz.....	88
Ilustración 26: Ejemplo Estrategia media sin infelices.....	88
Ilustración 27: Ejemplo Estrategia del individuo más respetado.....	89
Illustration 28: Smart TV.....	119
Ilustración 29: Tabletas.....	132
Illustration 30: Smartphones.....	135
Ilustración 31: Ejemplo de respuesta al obtener detalles de un Evento.....	164
Ilustración 32: Ejemplo de respuesta al invitar un usuario a un Evento.....	168
Ilustración 33: Ejemplo de respuesta de pedir detalles de la invitación a un grupo.....	170
Ilustración 34: Ejemplo de consulta de lista de membrecías del usuario.....	175
Ilustración 35: Ejemplo de respuesta de listado de invitaciones pendientes.....	177
Ilustración 36: Ejemplo de respuesta del listado de eventos organizados por el usuario.....	179
Ilustración 37: Ejemplo de resupesta de listado de amigos.....	183
Illustration 38: Ejemplo de respuesta de cartelera (2013).....	192
Ilustración 39: Diagrama de secuencia - CU Reproducir Película.....	200
Ilustración 40: Diagrama de secuencia - CU Calificar Película (1 de 2).....	202
Ilustración 41: Diagrama de secuencia - CU Calificar Película (2 de 2).....	203
Ilustración 42: Diagrama de secuencia - CU Recomendar Película Usuario (1 de 5).....	206
Ilustración 43: Diagrama de secuencia - CU Recomendar Película Usuario (2 de 5).....	207
Ilustración 44: Diagrama de secuencia - CU Recomendar Película Usuario (3 de 5).....	208
Ilustración 45: Diagrama de secuencia - CU Recomendar Película Usuario (4 de 5).....	209
Ilustración 46: Diagrama de secuencia - CU Recomendar Película Usuario (5 de 5).....	210
Ilustración 47: Home - entrar en Búsqueda Avanzada.....	229

Ilustración 48: Página de Búsqueda Avanzada.....	230
Ilustración 49: Resultado - página 1/2 - búsqueda de Predator.....	231
Ilustración 50: Detalles de película Predator.....	232
Ilustración 51: Calificación de película Predator.....	233
Ilustración 52: Home - entrar en Predicción Individual.....	234
Ilustración 53: Predicción Individual - Selección película "Predator 2".....	234
Ilustración 54: Predicción Individual - Predicción película "Predator 2".....	235
Ilustración 55: Home - entrar en Recomendación Individual.....	236
Ilustración 56: Recomendación Individual - Filtro catálogo.....	237
Ilustración 57: Recomendación Individual - Resultado de Filtro catálogo.....	238
Ilustración 58: Recomendación Individual - Resultado de recomendación.....	238

Lista de tablas

Tabla 1: Resultados estudio juego de datos Movielens-100k.....	14
Tabla 2: Resultados estudio con juego de datos Libimseti.cz.....	14
Tabla 3: Técnicas de Recomendación.....	17
Tabla 4: Resultados de ejemplo de estrategia Aditiva.....	19
Tabla 5: Tabla comparativa – Sistemas de Recomendación de Referencia.....	22
Tabla 6: Ejemplo para predicción grupal.....	40
Tabla 7: Pruebas dispositivos.....	62
Tabla 8: Densidad de usuarios por calificaciones del juego de datos del sistema.....	63
Tabla 9: Implementación Segunda variante – Resultados.....	64
Tabla 10: MAE de pruebas con voluntarios.....	65
Tabla 11: Resultados de ejemplo de estrategia Aditiva.....	84
Tabla 12: Respuesta de IMDbApi de información de película “El Orfanato”.....	94
Tabla 13: Respuesta Api Rotten Tomatoes de información de película “Machete”.....	98
Tabla 14: Respuesta Api información de película Hulu.....	105
Table 15: Tablet as - Características de Hardware.....	134
Table 16: Tablet as - Características de Software.....	135
Tabla 17: Smartphones - Características de Hardware.....	140
Tabla 18: Smartphones - Características de Software.....	143

1 Introducción

1.1 Motivación

En los últimos años ha crecido el consumo de contenido multimedia a través de Internet, existiendo una gran oferta de contenido disponible lo cual, si bien es positivo ya que marca una democratización del contenido, hace cada vez más difícil para el consumidor elegir el contenido más ajustado a sus preferencias. Por este motivo, nos parece interesante ordenarlo de alguna manera que le permita al consumidor elegir el contenido que desea. Para lograrlo queremos estudiar los algoritmos de recomendaciones individuales. Además nos interesa estudiar los algoritmos de recomendaciones grupales, los cuales tienen el objetivo de recomendar el contenido para un grupo de individuos teniendo en cuenta las preferencias de cada uno de los individuos del grupo.

Por otro lado vemos que se ha masificado el uso de dispositivos con acceso a Internet. Esto permite tener acceso a la información de forma rápida y en cualquier lugar. Nos planteamos el desafío de encontrar qué aporte pueden hacer estos dispositivos en las recomendaciones grupales.

Por último, vemos cómo la oferta de contenido multimedia proviene de diferentes fuentes. Esto plantea el problema de integrar las fuentes a nuestro sistema. En particular nos interesa analizar el estado actual de los proveedores de video bajo demanda y las posibilidades que brindan para poder integrarlos desde sistemas externos.

1.2 Objetivos

El proyecto tiene como objetivo principal recomendar contenido multimedia, proporcionar acceso directo al contenido que el consumidor quiere en forma individual o como parte de un grupo. Para esto identificamos los siguientes puntos:

- Estudiar los diferentes algoritmos de recomendaciones individuales y grupales existentes, obteniendo la taxonomía de los mismos.
- Construir una arquitectura que permita hacer recomendaciones de manera genérica, sin tener conocimiento del contenido a recomendar.
- Investigar la oferta de dispositivos con acceso a Internet, indagando sus capacidades técnicas para formar parte de un sistema de recomendaciones grupales.
- Desarrollar un caso de estudio que cubra los aspectos mencionados. Primero un servicio de recomendaciones genérico, que recomiende cualquier tipo de contenido. Luego un

prototipo de un sistema de recomendaciones individuales y grupales de películas, con una vista para dispositivos móviles, que recomiende películas dentro de una cartelera cinematográfica para un grupo de individuos.

Finalmente, como objetivo secundario, queremos integrar diferentes proveedores de video bajo demanda en nuestro prototipo.

1.3 Resultados y contribuciones

En este proyecto logramos cubrir con los objetivos planteados, resolviendo el principal problema de recomendaciones. Desarrollamos un prototipo que recomienda películas en forma individual o para un grupo. Este prototipo es una aplicación para dispositivos móviles con una vista Web, desarrollada bajo el estándar HTML5. A continuación mostramos los resultados obtenidos de mayor relevancia:

- Realizamos la carga inicial de datos tanto desde MovieLens como desde Internet Movie Database (IMDb), logrando integrar los datos de ambas fuentes.
- Sobre una base de calificaciones de películas obtenida desde MovieLens probamos algunas variantes del algoritmo de recomendación individual Slope One. Reafirmamos los resultados obtenidos por otros experimentos estudiados en cuanto al rendimiento del algoritmo y planteamos algunos problemas de performance en su implementación.
- Enfrentamos problemas de eficiencia de este algoritmo para el volumen de datos manejado. Implementamos una variante del algoritmo Slope One reduciendo la cantidad de calificaciones de los usuarios. El criterio utilizado se basa en considerar las películas con mayor cantidad de calificaciones otorgadas. Además, planteamos alternativas que se podrían implementar con otros recursos: procesos concurrentes, cache de predicciones o partición de la matriz de diferencias en clases de equivalencias para los ítems.
- Desarrollamos una aplicación Web de recomendaciones grupales para dispositivos móviles, portable a diferentes dispositivos dentro del estándar de HTML5.
- Validamos el prototipo desarrollados tanto funcionalmente como el rendimiento de los algoritmos de recomendación utilizados y la portabilidad del sistema sobre diferentes dispositivos móviles.

1.4 Organización del documento

Este documento se divide en cinco capítulos que presentan el trabajo realizado en el proyecto. A continuación brindamos una breve descripción de cada capítulo:

- Capítulo 1 – Introducción al problema abordado en el proyecto y la motivación para realizarlo. Se indican los principales objetivos, el alcance, los resultados y las contribuciones que se hacen en el proyecto.
- Capítulo 2 – Estado del arte: Se presentan los conceptos relacionados con los algoritmos de recomendación individual y grupal obteniendo una taxonomía de los mismos y se presentan evaluaciones y comparaciones de estos algoritmos. Se estudian los principales sistemas de recomendación de referencia, en particular en qué forma realizan recomendaciones y se estudian distintos proveedores de video bajo demanda. Finalmente se estudian las características de los principales dispositivos con acceso a Internet disponibles en el mercado, definiendo algunas propiedades que nos permita compararlos.
- Capítulo 3 – Solución Propuesta: Se define el prototipo desarrollado, describiendo los componentes más relevantes, mostrando y fundamentando las decisiones tomadas sobre la arquitectura construida. Se explican los algoritmos de recomendaciones implementados y se especifica técnicamente la solución.
- Capítulo 4 – Pruebas realizadas: Se muestra el diseño de los casos de prueba de relevancia para el sistema de recomendación de películas y el resultado de su ejecución. Se presentan los resultados de las pruebas para los dispositivos móviles probados y los resultados obtenidos en la prueba del algoritmo de recomendación utilizado.
- Capítulo 5 – Conclusiones y trabajos futuros: Se exponen las conclusiones que han surgido del proyecto y se presentan ideas de trabajos futuros.

Adicionalmente se entregarán los siguientes anexos

- Anexo A: Estrategias de Recomendaciones Grupales
- Anexo B: Sistemas de Recomendación de Referencia
- Anexo C: Tablas de Dispositivos
- Anexo D: Pautas de diseño para interfaz de usuario
- Anexo E: MovieRecommenderApp – documentación de API
- Anexo F: Casos de Uso
- Anexo G: Casos de prueba
- Anexo H: Guía de uso del sitio MovieRecommender

2 Estado del arte

2.1 Recomendaciones individuales

Nuestro trabajo tiene como principal objetivo ordenar de forma sistematizada el contenido multimedia de forma que los individuos obtengan el contenido que desean. Los sistemas de recomendación individual son una parte central para cubrir este objetivo. Vamos a estudiar las diferentes técnicas de recomendaciones individuales, sus características, qué requieren para poder desarrollarlas y los desafíos intrínsecos a cada una.

A fin de presentar los conceptos básicos relacionados a los Sistemas de Recomendación (SR) nos referiremos a la introducción de SR que realizan Vozalis y Margaritis en su artículo *Analysis of Recommender Systems' Algorithms* [1], indicando explícitamente cualquier otra referencia sobre la que se haga mención. Este explica que los SR fueron introducidos como una técnica inteligente para lidiar con la sobrecarga de información y/o productos. Luego describe los conceptos principales que manejaremos en este documento.

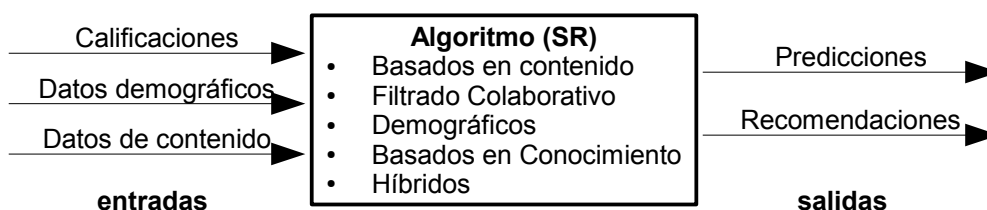


Ilustración 1: Abstracción gráfica de SR

Gráficamente, podemos abstraer los SR como se muestra en la Ilustración 1.

En cualquier SR aparecen dos entidades básicas que son: el *usuario* (al cual a veces se refieren como *cliente*) y el *ítem* o *producto*. En este documento utilizaremos el término *ítem* y el término *usuario*. El *ítem* es el objeto unidad del contenido que manejan los SR, el cual puede ser de cualquier tipo como ser películas, música, libros, chistes, restaurantes, etc. [2]. El *usuario* es una persona que utiliza el SR para proveerse de recomendaciones del sistema sobre los ítems.

La **entrada** de un SR depende del tipo de algoritmo de filtrado que éste emplee. Generalmente la entrada pertenece a alguna de las siguientes categorías:

1. *Calificaciones (ratings, también llamados votos)*, que expresan la opinión de los usuarios respecto a los ítems. Normalmente son provistas por los usuarios y siguen una escala numérica específica (ejemplo, 1-malo a 5-excelente). Un esquema común de calificación es el *esquema de calificación binario* el cual solo permite que las calificaciones tomen el valor 0 o 1. Las calificaciones también pueden ser obtenidas implícitamente a partir del historial de compras del usuario, logs web, accesos a hipervínculos, hábitos de navegación web u otro tipo de patrones de acceso a la información.

2. *Datos demográficos*, refieren a información como ser la edad, el género y la educación de los usuarios. Este tipo de información es usualmente difícil de obtener. Normalmente es recopilada explícitamente de parte del usuario.
3. *Datos de contenido*, se basa en el análisis textual de documentos relacionados a los ítems que el usuario ha calificado. Las características extraídas por este análisis son utilizadas como entradas en los algoritmos de filtrado a fin de inferir el perfil del usuario.

El **objetivo** de un SR es generar sugerencias sobre nuevos ítems o predecir la utilidad de un ítem específico para determinado usuario (también denominado usuario activo). En ambos casos el proceso se basa en la entrada provista, la cual está relacionada con las preferencias del usuario en cuestión.

La **salida** de un SR puede ser una *Predicción* o una *Recomendación*.

- Una *Predicción* es expresada como un valor numérico, p_{ai} , que representa la calificación anticipada del usuario activo u_a sobre el ítem i . Este valor deberá estar contenido en la misma escala numérica que la entrada referente a las calificaciones inicialmente provistas por el usuario activo. Esta forma de salida de SR también es conocida como *Puntuación Individual (Individual Scoring)*.
- Una *Recomendación* es expresada como una lista de n ítems, que el usuario activo espera que sean de su preferencia. El enfoque usual en este caso requiere que la lista incluya solo aquellos ítems que el usuario activo no haya comprado, visto o calificado. Esta forma de salida de SR es también conocida como *Top-n Recomendaciones* o *Ranking de Puntuación (Ranked Scoring)*.

En esta sección definimos la notación utilizada durante todo el proyecto, estudiamos las diferentes técnicas de recomendaciones individuales dentro de las categorías de Filtro Colaborativo, Basadas en contenido, Demográficas, Basadas en Conocimiento e Híbridas. Además describimos los problemas que frecuentemente presentan estas técnicas y finalmente las resumimos y comparamos a partir de ciertas propiedades estudiadas.

2.1.1 Notación

Para describir los sistemas de recomendaciones adoptamos la notación sugerida por Candillier en su trabajo *State-of-the-Art Recommender Systems* [2].

Sea $U = \{u_1, \dots, u_N\}$ un conjunto de N usuarios, $I = \{i_1, \dots, i_M\}$ un conjunto de M ítems, $S_u \subseteq I$ representa el conjunto de elementos que el usuario u ha calificado. Aquí es importante notar que S_u puede ser vacío, lo que denotaría que el usuario u no ha revelado preferencia alguna sobre los ítems. Las opiniones de un usuario, cuya cantidad corresponde a la cardinalidad del conjunto S_u , generalmente son indicadas en la forma de puntos de calificación. Estos puntos de calificación los representaremos con la expresión r_{ui} , con $u \in U$ e $i \in I$, que denota la calificación que otorgó un usuario u a un ítem i . Todas estas calificaciones confeccionan una matriz usuarios-ítems de dimensión $N \times M$ a la que denotaremos R . También se distinguirá al

usuario $u_a \in U$ como *usuario activo* (al que denotaremos simplemente como usuario a) y se define $NR \subseteq I$ como el subconjunto de ítems para los cuales el usuario activo no ha indicado su opinión aún, y como consecuencia, sobre el cual el SR deberá generar las sugerencias pertinentes. También denotaremos como $\bar{r}_u$ al valor promedio de calificaciones otorgadas por el usuario u , y como $\bar{r}_i$ al valor promedio de calificaciones recibidas por el ítem i .

De esta manera el objetivo de los distintos SR será predecir la calificación p_{ai} del usuario a sobre el ítem i .

2.1.2 Filtro Colaborativo

Según Candillier en su trabajo *State-of-the-Art Recommender Systems* [2], los SR de filtros colaborativos (SRFC) fueron los pioneros en el área. En este caso el sistema toma como entrada las calificaciones que otorga el usuario a cada ítem o grupo de estos. En estos sistemas se maneja la noción de *vecindad de usuarios*, la cual se define según similitud en los perfiles de usuarios comparándolos en base a las calificaciones sobre ítems que comparten entre sí. Del mismo modo, los ítems pueden ser comparados con base en la calificación común que les otorgan los usuarios creando de esta manera la noción de *vecindad para los ítems*. En consecuencia, la calificación de un ítem para un determinado usuario se puede predecir basándose en las calificaciones otorgadas por la vecindad de usuarios y la de ítems. En otras palabras, se trata de comparar perfiles de usuarios con gustos similares.

Candillier [2] explica que podemos distinguir tres enfoques principales dentro de los algoritmos de SRFC: *basado en usuarios*, *basado en ítems* y *basado en el modelo*. Los algoritmos *basados en usuarios* y *basados en ítems* son clasificados como *algoritmos basados en memoria* (memory-based), o sea que operan sobre toda la información recopilada de los usuarios al momento de realizar las predicciones. Los algoritmos de recomendación *basada en el modelo* (model-based) utilizan la información recopilada para confeccionar un modelo que luego es utilizado para realizar las predicciones [3].

El objetivo de los enfoques de tipo colaborativo es entonces predecir la calificación p_{ai} del usuario activo a sobre el ítem i , es decir que ya ha calificado algún ítem (S_a no es un conjunto vacío).

2.1.2.1 Enfoque basado en el usuario

En este enfoque se construye un modelo de vecindad entre usuarios utilizando una función de similitud entre usuarios. Los usuarios que tienen calificaciones similares, obtienen un mayor valor similitud. Por el contrario, si hay marcadas diferencias en las calificaciones tienen un menor valor de similitud. Sea $similitud(a, u)$ la similitud entre el usuario activo a y un usuario u (que definimos con detalle en la sección 2.1.2.4 Medidas de Similitud), sea K la cantidad de usuarios considerada para estudiar su similitud, entonces se denota T_a al conjunto de vecinos del usuario a con los cuales se maximiza su similitud. A partir de esto, vemos una manera de predecir la calificación que le otorgaría el usuario a sobre un ítem i :

$$p_{ai} = \frac{\sum_{\{u \in T_a / i \in S_u\}} \text{similitud}(a, u) \times r_{ui}}{\sum_{\{u \in T_a / i \in S_u\}} \text{similitud}(a, u)}$$

La predicción de la calificación del usuario sobre determinado ítem, se basa en las calificaciones otorgadas a dicho ítem por parte de los usuarios vecinos más cercanos. A su vez, estas calificaciones se ponderan según el valor de similitud [2]. De aquí se desprende que el tiempo de complejidad de construir el modelo de vecinos es $O(N^2 \times M \times K)$. De esta manera, el orden de predecir la calificación es $O(K)$ una vez que se cuenta con el modelo de vecinos.

2.1.2.2 Enfoque basado en el ítem

De forma análoga al enfoque basado en el usuario, se define el enfoque basado en el ítem, utilizando una función de similitud entre ítems. Los ítems que tienen calificaciones similares, obtienen mayor valor de similitud. Por el contrario, si hay marcadas diferencias en las calificaciones, tienen un menor valor de similitud. Siendo $\text{similitud}(i, j)$ la medida de similitud entre los ítems i y j , K la cantidad ítems vecinos a tomar en cuenta y T_i el conjunto de vecinos del ítem i , vemos una manera de predecir las calificaciones de un usuario es mediante la suma ponderada como se muestra a continuación:

$$p_{ai} = \frac{\sum_{\{j \in S_a \cap T_i\}} \text{similitud}(i, j) \times r_{aj}}{\sum_{\{j \in S_a \cap T_i\}} |\text{similitud}(i, j)|}$$

La predicción de la calificación que otorgaría un usuario sobre un ítem se basa en las calificaciones que otorgó el usuario a los ítems vecinos más cercanos al ítem. Estas calificaciones a su vez se ponderan según el valor de similitud [2]. Aquí el tiempo de complejidad de construir el modelo de vecinos es $O(M^2 \times N \times K)$. De esta manera, el orden de predecir la calificación es $O(K)$.

2.1.2.3 Enfoque basado en el modelo

La idea general de este enfoque es generar un modelo probabilístico de información fuera de línea, para luego realizar la predicciones lo más rápido posible. Las Redes Bayesianas y el Modelo de Clúster son dos implementaciones de este enfoque. En el trabajo *Empirical Analysis of Predictive Algorithms for Collaborative Filtering* [3] se puede encontrar una descripción de estas implementaciones y un estudio que los compara. Según Candiller, en su trabajo *State-of-the-Art Recommender Systems* [2], el Modelo de Clúster tiene una complejidad para construir el modelo de orden $O(N \times M \times K \times L)$, donde L es el número de ejecuciones que requiere el algoritmo y tiene un orden de $O(1)$ para realizar la predicción.

2.1.2.4 Medidas de Similitud

Como se puede apreciar en los enfoques basado en el usuario y basado en el ítem, la similitud definida entre usuarios o entre ítems es crucial en lo que respecta a estos algoritmos [2]. Una medida de similitud se presenta como una métrica de relevancia entre dos vectores. Cuando los valores de dichos vectores son asociados con un modelo de usuario, se habla de medidas de

similitud basada en el usuario. Cuando los valores son relacionados con un modelo de ítem, se denominan medidas de similitud basada en el ítem [4]. En lo que respecta al cómputo, la diferencia fundamental entre los SRFC basados en usuario y los SRFC basados en ítem es cómo se toman los vectores en la matriz usuario-ítem de calificaciones. En el primero, la similitud es computada sobre las filas de esta matriz, mientras que en el segundo, el cómputo se realiza sobre las columnas de la matriz [5]. Si bien existen diversas medidas de similitud aquí nos concentramos en explicar aquellas que se utilizaron para los estudios comparativos que hemos tomado de referencia, los cuales mencionamos en la Sección 2.1.5: Correlación de Pearson, Vector Coseno de Similitud y Vector Coseno de Similitud Ajustado. Otras medidas de similitud en las cuales no vamos a profundizar son: Diferencia Cuadrática, Índice de Jaccard [2], Correlación de Spearman [6], Distancia Euclidiana.

Coeficiente de Correlación de Pearson

El Coeficiente de Correlación de Pearson mide la correlación lineal entre dos vectores de calificaciones, como se muestra en la siguiente fórmula [4].

$$similitud(u_1, u_2) = \frac{\sum_{i \in S_{u_1} \cap S_{u_2}} (r_{u_1 i} - \bar{r}_{u_1}) \times (r_{u_2 i} - \bar{r}_{u_2})}{\sqrt{\sum_{i \in S_{u_1} \cap S_{u_2}} (r_{u_1 i} - \bar{r}_{u_1})^2 \times \sum_{i \in S_{u_1} \cap S_{u_2}} (r_{u_2 i} - \bar{r}_{u_2})^2}}$$

La correlación de Pearson, según se explica en el libro *Mahout in Action* [6], es un número entre -1 y 1 que mide la tendencia de dos series de números. Es decir que mide la tendencia de los números a moverse juntos proporcionalmente. Cuando la tendencia es alta la correlación es cercana a 1; cuando aparenta haber poca relación esta es cercana a 0; y cuando la relación aparenta ser opuesta, la correlación se aproxima a -1. Candillier en su trabajo *State-of-the-Art Recommender Systems* [2] explica que si bien este método es de los más utilizados tradicionalmente, sufre de un inconveniente debido a que computa solo calificaciones sobre ítems que ambos usuarios hayan calificado. Por ejemplo, si dos usuarios sólo tienen una película en común y calificaron con la misma puntuación, se vuelven complemente similares, lo cual no es correcto de asumir.

Vector Coseno de Similitud

Karypis en su trabajo *Evaluation of Item-Based Top-N Recommendation Algorithm* [7], explica que el vector coseno de similitud es una manera de computar la similitud entre dos ítems, tratando cada ítem como un vector en el espacio de usuarios. Entonces se puede utilizar la medida del coseno entre estos dos vectores como la medida de similitud. Formalmente, la medida de similitud entre dos ítems i y j es definida por el coseno de vectores de dimensión N correspondientes a la columna i -ésima y j -ésima de la matriz que forma la relación r (matriz usuario-ítem de calificaciones). El coseno entre estos vectores es dado por la siguiente fórmula.

$$similitud(i, j) = \cos(\vec{i}, \vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\|_2 \|\vec{j}\|_2}$$

Con un razonamiento análogo Gong [4] presenta la siguiente fórmula que aplica el método de Vector Coseno para medir la similitud entre dos usuarios u_1 y u_2 .

$$similitud(u_1, u_2) = \frac{\sum_{i \in S_{u_1} \cap S_{u_2}} r_{u_1 i} \times r_{u_2 i}}{\sqrt{\sum_{i \in S_{u_1} \cap S_{u_2}} r_{u_1 i}^2 \times \sum_{i \in S_{u_1} \cap S_{u_2}} r_{u_2 i}^2}}$$

Vector Coseno Ajustado

El cálculo de similitud entre ítems utilizando el método de Vector Coseno no tiene en cuenta las diferentes escalas de calificación con las que se manejan los diferentes usuarios, lo que puede considerarse un inconveniente importante de esta medida. El Vector Coseno de Similitud Ajustado [5], compensa dicho inconveniente restando el valor promedio de calificaciones que otorga cada usuario, con el fin de darle el peso que merece el valor de su calificación sobre el ítem en cuestión. Formalmente, la siguiente fórmula presenta el cálculo que realiza esta medida.

$$similitud(i, j) = \frac{\sum_{u \in U} (r_{u i} - \bar{r}_u) \times (r_{u j} - \bar{r}_u)}{\sqrt{\sum_{u \in U} (r_{u i} - \bar{r}_u)^2} \times \sqrt{\sum_{u \in U} (r_{u j} - \bar{r}_u)^2}}$$

2.1.2.5 Slope One

Slope One es una familia de algoritmos de SRFC basado en memoria introducida en el año 2005 por Daniel Lemire y Anna Maclachlan en su trabajo *Slope One Predictors for Online Rating-Based Collaborative Filtering* [8]. A diferencia de los algoritmos de SRFC vistos anteriormente, Slope One no maneja el concepto de similitud. Según sus autores, el algoritmo funciona bajo el principio de *diferencial de popularidad* entre ítems para los usuarios. Este principio se puede definir como *diferencias promedio de las calificaciones* que indica cuanto más gustó un ítem que otro en promedio.

El objetivo de estos algoritmos es resolver algunos de los problemas que presentaban el resto de los algoritmos de recomendaciones individuales. En particular se enfoca en estos cinco objetivos:

- Sencillo de implementar y mantener. Fácil de interpretar.
- Actualización al instante: si se agregan nuevas calificaciones, cambian todas las predicciones al instante.
- Eficientes en tiempo de consulta. Las consultas pueden resolverse rápidamente, el costo es el espacio de almacenamiento que requieren.
- Los usuarios con pocas recomendaciones, recibirán buenas recomendaciones.
- Precisión razonable con simpleza y posibilidad de escalar.

El esquema Slope One toma en cuenta la información generada por otros usuarios que han calificado el mismo ítem y por otros ítems que han sido calificados por el mismo usuario [8].

Formalización

El nombre Slope One (Pendiente Uno) parte del hecho que en este tipo de algoritmos de recomendación se asume que hay cierta relación lineal entre el valor de calificación de un ítem y otro, lo que hace válido estimar la preferencia P_i para cierto ítem i basándose en las preferencias P_j que se tiene de otro ítem j mediante una función lineal $P_j = P_i \cdot const + b$. Además Slope One agrega la simplificación de asumir que la constante $const$ toma valor 1: pendiente uno. Así simplemente se resume en encontrar las diferencias promedio de preferencias $b = P_i - P_j$ para cada par de ítems. Con estas diferencias podemos definir la matriz de desviaciones promedios entre ítems:

$$dev_{ij} = \frac{\sum_{u \in S_{ij}} (r_{ui} - r_{uj})}{|S_{ij}|}$$

Donde S_{ij} denota el conjunto de usuarios que calificaron ambos ítems i y j , $|S_{ij}|$ es el número de usuarios del conjunto. Con esta matriz podemos definir la predicción del usuario u sobre el ítem i de la siguiente manera:

$$p_{ui} = \frac{\sum_{j \in [S_u - \{i\}]} (r_{uj} + dev_{ij})}{|R_j|}$$

Donde $R_j = \{i : i \in S_u, i \neq j\}$ representan todos los ítems calificados por el usuario u excepto el ítem i .

Variante Slope One con peso:

Es una variante del algoritmo Slope One que toma en cuenta la cantidad de calificaciones de los ítems. Para esto definimos la cantidad de usuarios que calificaron los ítems i y j $c_{ij} = |S_{ij}|$. Con esto, se re-define la predicción como:

$$p_{ui} = \frac{\sum_{j \in [S_u - \{i\}]} (r_{uj} + dev_{ij}) * c_{ij}}{\sum_{j \in [S_u - \{i\}]} c_{ij}}$$

De esta manera, los ítems con más calificaciones tendrán más peso en la predicción.

2.1.3 Sistema de Recomendación Basado en Contenido

Los SR basados en contenido (SRBC) son sistemas clasificadores derivados de investigaciones realizadas sobre máquinas de aprendizaje [9]. La recomendación de un ítem a un usuario se realiza en base a la descripción del ítem y en un perfil de los gustos del usuario [10]. Se intenta detectar las características de un ítem calificado positivamente por el usuario y recomendar posteriormente aquellos ítems que posean dichas características. El foco se sitúa en la similitud entre atributos de los ítems y no entre opiniones de usuarios como sucede en los SRFC. Así es que este tipo de sistemas concentran su atención en cómo realizar la descripción de los ítems, cómo confeccionar el perfil de gustos o preferencias de los usuarios y cómo realizar la comparación de los ítems contra el perfil para determinar la mejor recomendación. De manera que la intervención en aspectos del dominio del sistema es más elevada que en el caso de SRFC

[2]. En algunos casos se necesita una persona que determine cuáles son las características de un ítem. Por ejemplo, para el contenido multimedia no hay una forma automática de analizarlo. En otros casos el sistema podría analizar texto, reconociendo y/o buscando palabras claves [11], como por ejemplo ítems de tipo artículos, papers, etc. La descripción de un ítem puede ser por ejemplo: el género de una película; la ubicación de un restaurante; u otra característica dependiendo siempre del tipo de ítem del que se trate (para el caso de características de video se recomienda ver la sección *The Movie Features* en el trabajo titulado *A Content-based Movie Recommendation System Empowered by Collaborative Filtering* [12]). El perfil del usuario puede consistir de varios tipos de información, donde los más comúnmente utilizados son los siguientes según señala Pazzani en su trabajo *Content-Based Recommendation Systems (User Profiles)* [10]:

- Un modelo de preferencias de los tipos de ítems (descripciones de) que le interesan al usuario, también llamado *modelo de usuario* o *query*. Aquí la preferencia indica una relación entre un usuario dado y la información que maneja el sistema, la cual deberá ser capaz de ser procesada por el sistema e incluir información útil para realizar las recomendaciones [2]. Este modelo suele representarse mediante funciones que para cualquier ítem calculan qué probabilidad hay de que le interese al usuario.
- El historial de las interacciones del usuario con el SRBC. Estas podrían ser: registro de información de los ítems que el usuario ha visto junto con la interacción que realizó con el sistema (ej., el usuario compra un ítem y le otorga cierta calificación), también registro del historial de búsquedas del usuario dentro del sistema, lo cual es una manera implícita de recopilar preferencias del usuario.

De la sección *Learning user profiles* del trabajo *Learning and Revising User Profiles: The Identification of Interesting Web Sites* [13], se desprende que la información necesaria para confeccionar el perfil de usuario puede ser obtenida implícitamente a partir de las preferencias de ítems que tiene el usuario. Basándose en la descripción de los ítems que el usuario haya calificado anteriormente, o explícitamente a partir de cuestionarios sobre sus preferencias (personalización de usuario). Los perfiles suelen ser representados como vectores de peso sobre las descripciones de los ítems donde cada posición del vector indica una característica y su valor el peso o importancia que tiene dicha característica para ese usuario. Esta representación no es arbitraria sino que se utiliza a fin de facilitar la construcción de algoritmos de aprendizaje.

Los algoritmos de aprendizaje de clasificaciones son una parte importante de los SRBC, puesto que se utilizan para la confección de funciones que modelen los intereses de cada usuario. Dado un nuevo ítem y un modelo del usuario, la función generada por el algoritmo predice si el usuario tendrá interés o no por dicho ítem. Muchos de estos algoritmos generan dicha función de forma que calcule la probabilidad de que al usuario le guste determinado ítem. Esta probabilidad puede usarse luego para dar orden a la lista de recomendaciones. Como alternativa, un algoritmo puede hacer que la función retorne directamente un valor numérico que podría ser el grado de desinterés del usuario [1].

2.1.4 Otros Algoritmos

2.1.4.1 Demográficos

Krulwich en su trabajo *LIFESTYLE FINDER: Intelligent User Profiling Using Large-Scale Demographic Data* [14], explica que la tarea principal de los algoritmos presentados anteriormente se centra en generalizar perfiles de usuario a partir de los intereses y características. Krulwich presenta el método demográfico como una combinación de las ventajas de los SRFC y SRBC. Explica que éste puede ser altamente efectivo, requiere pocas cantidades de información y que los perfiles generados son de amplio alcance. Sin embargo, también remarca el hecho de que estas ventajas traen como consecuencia una ligera disminución en la precisión de las predicciones. En este método, los datos de entrada son utilizados para clasificar los usuarios en términos de su información demográfica. Estas características son utilizadas para confeccionar *clústeres demográficos* con los cuales se clasificarán a los usuarios.

2.1.4.2 Basados en Conocimiento

Los SR basados en Conocimiento (SRCO), utilizan el conocimiento acerca del usuario y los ítems para generar una recomendación, razonando acerca de cuáles son los ítems que se encuentran entre los requerimientos del usuario [9]. Estos sistemas utilizan la interacción con el usuario para la exploración del dominio y concluyen en la generación de una recomendación dirigida (en general) explícitamente por el usuario. Es decir que generan una recomendación partiendo del conocimiento que da el usuario sobre sus necesidades, del conocimiento de los ítems a recomendar, buscando los que mejor se adapten a las necesidades de éste. A modo de referencia, en el artículo *Hybrid Recommender Systems: Survey and Experiments* [15] se presenta un ejemplo de estos sistemas llamado *Entrée* que trata sobre un SRCO de restaurantes.

2.1.4.3 Híbridos

Los SR Híbridos intentan combinar las mejores características de cada uno de los SR que presentamos anteriormente con el fin de mejorar las recomendaciones generadas minimizando a su vez los problemas producidos en cada uno de ellos.

2.1.5 Estudios comparativos

En primer lugar vamos a presentar el estudio realizado por Patrick Marschik en su Tesis del año 2010 [16], en el cual se comparan diferentes tipos de algoritmos de recomendación de Filtro Colaborativo. Veremos los datos que fueron objeto de estudio y cómo se midieron los resultados. Los algoritmos estudiados son:

- Basados en usuario, con el cálculo de similitud por Coeficiente de Correlación de Pearson.
- Basados en ítem, con el cálculo de similitud por Vector de coseno.
- Basados en ítem, con el cálculo de similitud por Vector de coseno ajustado.
- Slope One simple.

- Slope One con peso.

En este estudio se utilizó la métrica *Mean Absolute Error* (MAE) para evaluar y comparar los algoritmos de recomendación. La $T = \{(u, i) / \exists r_{ui}\}$ métrica MAE mide la desviación media entre la predicción de los usuarios sobre los ítems y el valor calificado realmente. Se define formalmente de la siguiente manera:

$$MAE = \frac{\sum_{(u,i) \in T} |r_{ui} - p_{ui}|}{|T|}$$

Donde r_{ui} y p_{ui} ya fueron definidos en la sección 2.1.1 de este capítulo y definimos $|T|$ como la cantidad de elementos del conjunto $T = \{(u, i) / \exists r_{ui}\}$. El conjunto de datos utilizados se tomó de MovieLens-100k [17], con una calificación de hasta 5 puntos con 0.5 puntos de granularidad. Cada usuario ha calificado al menos 20 películas. Este conjunto de datos se separó en dos: 80% de datos de entrenamiento, sobre los cuales se realizan los cálculos de los algoritmos y 20% de datos de prueba, sobre los cuales se realizan las predicciones. En la tabla 1 se presentan los resultados obtenidos.

Algoritmo	MAE
Basado en usuarios – Coeficiente de Correlación de Pearson	0,751
Basado en ítems – Vector Coseno	0,837
Basado en ítems – Vector Coseno Ajustado	0,804
Slope One	0,779

Tabla 1: Resultados estudio juego de datos MovieLens-100k

Otro experimento estudiado está documentado en el libro *Mahout in Action* [6], donde se calcula el MAE de distintos algoritmos de SRFC. El juego de datos utilizado es tomado del sitio checo Libimseti [18], donde los usuarios califican los perfiles de otros usuarios del 1 al 10 con granularidad de 1. Los datos están publicados en *Occam's Lab - Ratings Data Recopilation* [19], contando con 17.359.346 calificaciones de 135.359 usuarios. A continuación se muestran los algoritmos estudiados y los resultados obtenidos.

Algoritmo	MAE
Basado en usuarios – Coeficiente de Correlación de Pearson	1,19
Basado en ítems – Coeficiente de Correlación de Pearson	2,32
Slope One	1,41

Tabla 2: Resultados estudio con juego de datos Libimseti.cz

Según estos dos estudios, si nos enfocamos en el resultado de predicción, el mejor algoritmo de recomendación es el Basado en el usuario con el cálculo de similitud por Coeficiente de Correlación de Pearson. Luego le sigue el algoritmo Slope One y finalmente los algoritmos Basados en el ítem tienen el peor rendimiento de los tres.

2.1.6 Posibles deficiencias

A continuación definimos los problemas más conocidos de los SR.

2.1.6.1 Ramp-Up

El problema de Ramp-Up [20] se produce cuando en un sistema recién puesto en marcha, no existe información suficiente para generar buenas recomendaciones. Es decir, el sistema debe conocer información sobre las decisiones de usuarios para garantizar la calidad de las recomendaciones futuras. Los sistemas de SRFC, SRBC y Demográficos padecen este problema ya que reciben como entrada el historial de calificaciones de los usuarios sobre los ítems.

2.1.6.2 Free-rider

Este se produce cuando el usuario hace uso (en general implícitamente) de las recomendaciones realizadas por sus pares, pero no genera las suyas propias. De esta forma, si cada usuario que accede al sistema realiza esta acción, el mismo se estancará y no se potenciará el perfeccionamiento de las sugerencias [21].

2.1.6.3 Problema de calificación previa (arranque en frío)

Supongamos que un SRFC esté funcionando con los datos suficientes como para recomendar los ítems ingresados. Cuando ingresan nuevos ítems al sistema estos nunca serán recomendados debido a que ningún usuario los ha calificado. Este problema es conocido con varios nombres: Problema de calificación previa (Early Rate); Primer calificación (First Rater); o Arranque en Frío (Cold Start Problem) [1]. Sarwar en su artículo *Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System* [22] presenta como solución incorporar *Filterbots*. Estos son procesos automáticos que se encargarán de calificar nuevos ítems tan pronto son ingresados al sistema. Lo bueno de estos *Filterbots*, es que el sistema puede tratarlos como simples usuarios que generan muchas calificaciones, pero no exigen predicción/recomendación alguna.

2.1.6.4 Usuarios inusuales (oveja gris)

También conocido como problema de la *Oveja Gris*, se refiere a individuos con gustos inusuales. Es decir que sus perfiles no encajan ni tampoco desencajan consistentemente con ningún grupo de usuarios. Por tal motivo, estos usuarios difícilmente se vean beneficiados por el SR, puesto que rara vez recibirán recomendaciones acertadas [1].

2.1.6.5 Vulnerabilidad a los ataques

Vulnerabilidad a los ataques es influenciar de manera negativa las recomendaciones que genera el sistema. Massa en su artículo *Using Trust in Recommender Systems: an Experimental Analysis* [11] explica esta vulnerabilidad con el siguiente ejemplo. Supongamos que un usuario malicioso desea que el SR le recomiende a cierto usuario u el ítem i . Este puede crear un nuevo usuario falso, calificar de igual manera todos los ítems que ha calificado el usuario u y también otorgar una alta calificación al ítem i . De esta forma, cuando el SR busque usuarios similares al usuario u , encontrará que tiene una gran similitud con el usuario falso. Al ver que éste calificó muy bien al ítem i , se volverá altamente probable que finalmente recomiende dicho ítem al usuario u . El

usuario malicioso también podría crear una gran cantidad de usuarios falsos repitiendo el proceso anterior. Este es un claro problema que pueden sufrir los SRFC, sobre todo si no cuentan con un mecanismo y/o configuración de privacidad que restrinja qué usuarios pueden ver las calificaciones otorgados por otros usuarios.

2.1.6.6 Escasez de datos

Según G. Adomavicius y A. Tuzhilin en su artículo *Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions* [23], en cualquier SR, el número de calificaciones con las que se cuenta es usualmente muy pequeño comparado al número de calificaciones que se necesita para predecir. Es por lo tanto importante poder realizar predicciones efectivas a partir de pocos datos de ejemplo. También, el éxito de los SRFC depende en su capacidad de manejar enorme cantidad de usuarios. Por ejemplo, en un SRFC de películas podría suceder que muchas películas calificadas por unos pocos usuarios sean recomendadas rara vez, aún pese a que les hayan otorgado calificaciones altas.

2.1.6.7 Sobre-especialización

Los SRBC pueden sufrir del problema de sobre-especialización, también conocido como problema de novedades. Lo que sucede es que los ítems recomendados terminan siendo muy similares o idénticos a los que el usuario ha calificado positivamente, en otras palabras el usuario pasa a estar restringido a una cierta tipología de ítems. Esto ocasiona que, al momento de generar las recomendaciones, aquellos ítems que difieren en algún aspecto de los ya preferidos son rechazados y nunca serán incluidos en la lista de sugerencias para el usuario en cuestión privando a éste de ítems que capaz son sensiblemente diferentes o pertenecen a un nuevo/novedoso tipo de ítems que podría interesarle al usuario [2].

2.1.7 Resumen de Técnicas de recomendación

Bruke en su artículo *Hybrid Recommender Systems: Survey and Experiments* [15], resume las técnicas de recomendación, obviando las híbridas que son combinaciones de estas, según las siguientes características:

1. Antecedentes del usuario. Es información con la que cuenta el sistema antes de empezar con el proceso de recomendación.
2. Datos de entrada. Esta es la información que el usuario hace llegar al sistema a fin de poder generar las recomendaciones.
3. Y un algoritmo que combina los datos de antecedentes y de entrada para formular las recomendaciones.

En base a los puntos anteriores, Bruke formula una tabla que resume las técnicas de recomendación. Nosotros nos basamos en ésta para armar la tabla 3, donde mostramos un resumen de las técnicas mencionadas en este informe. En esta tabla, I es un conjunto de ítems sobre el cual se realizaran las recomendaciones, U es un conjunto de usuarios cuyas preferencias

ya se conocen, u es el usuario para el cual se necesita generar las recomendaciones e i es un ítem para el cual se desea saber si le interesará al usuario u según sus preferencias.

Técnica de recomendación	Base	Datos de entrada	Procesamiento
Colaborativo	Calificaciones de U sobre ítems en I .	Calificaciones de usuario u de ítems en I .	Identificar usuarios en U similares a u , y extrapolar sus calificaciones sobre ítem i .
Basado en contenido	Características de los ítems en I .	Calificaciones de usuario u de ítems en I .	Generar un clasificador que encaje el comportamiento de calificaciones del usuario u y lo aplique en el ítem i .
Demográficos	Información demográfica sobre U y sus calificaciones sobre los ítems en I .	Información demográfica sobre el usuario u .	Identificar usuarios con similar demografía que el usuario u , y extrapolar sus calificaciones otorgadas al ítem i .
Basados en Conocimiento	Características de los ítems en I . Conocimiento sobre cómo esos ítems cumplen con los requerimientos de un usuario.	Una descripción de los requerimientos o gustos del usuario u .	Deducir qué el ítem i cumple con los requerimientos del usuario u .

Tabla 3: Técnicas de Recomendación

2.2 Recomendaciones grupales

El gran despliegue de múltiples dispositivos conectados a Internet permite tener acceso a la información de forma rápida y en cualquier lugar. Esto favorece que los individuos formen grupos virtuales intercambiando información en forma fluida y oportuna. Según Judith Masthoff, en su trabajo *Group Modeling: Selecting a Sequence of Television items to Suit a Group of Viewers* [24], se han desarrollado pocos sistemas que tengan en cuenta a un conjunto de individuos para generar recomendaciones. Uno que se aproxima es PolyLens (extensión del sitio MovieLens) [25], es un sistemas de recomendación de películas para grupos de usuarios. El estudio de uso de PolyLens ha ayudado a entender las consideraciones que se deben tener en cuenta a la hora de desarrollar un sistema para grupos. En este contexto nos parece interesante investigar cómo funcionan los sistemas de recomendaciones para grupos de individuos. Un sistema de recomendación grupal considera a los individuos con algún criterio que maximice la felicidad (también llamada satisfacción) del grupo. El concepto de felicidad del grupo puede variar, en ocasiones la felicidad del grupo corresponde al promedio de preferencias de los individuos, en otras ocasiones corresponde con la preferencia del menos feliz, etc. Existen diferentes estrategias y algoritmos para calcular el grado de felicidad según la definición que usemos. Todas estas estrategias utilizan como insumo fundamental la predicción de gusto individual estudiada en la sección 2.1. En las siguientes secciones definiremos lo que es un sistema adaptativo, resumiremos las estrategias de recomendación grupal y estudiaremos las consideraciones a tener en cuenta cuando trabajamos con grupos de individuos.

2.2.1 *La televisión como sistema adaptativo al medio*

Un sistema adaptativo es aquel que puede ajustarse así mismo automáticamente ante los cambios del ambiente en el que se encuentra [26]. En estos sistemas el concepto clásico de usuario del sistema cambia y se habla de habitante del ambiente [27]. Por ejemplo, la televisión como sistema que se adapta, donde los habitantes del ambiente son los televidentes, el grupo de individuos que está mirando la televisión. Uno de los primeros sistemas adaptativos puros fue MusicFX [27]. Éste sistema funcionaba en un gimnasio, cuyo principal objetivo era la reproducción de estaciones radiales. Para elegir la estación, tenía en cuenta las preferencias de géneros musicales de quienes se encontraban en ese momento desarrollando su sesión de ejercicio. La experiencia de MusicFX es aplicable a cualquier entorno compartido en el que las personas se reúnen por un período de tiempo prolongado. En particular es aplicable cuando un conjunto de individuos mira televisión, ya que a diferencia del uso de la PC, ver televisión es una actividad familiar o una actividad social [28] [29]. Livingstone & Bovill en su proyecto de investigación en el Reino Unido denominado *Young people, new media* [30], establece que la televisión es el medio que más a menudo se comparte en familia. Encabeza la lista de actividades que los padres comparten con sus hijos y que más de las dos terceras partes de los niños miran sus programas favoritos con alguien más, generalmente alguien de la familia.

Para complacer a todos los individuos o a su gran mayoría, la televisión debe adaptarse al medio. Debe saber quiénes son los televidentes en cada momento, cuáles son sus preferencias individuales y saber contestar la pregunta fundamental: ¿qué contenido es el mejor para sugerir al grupo actual de televidentes? Para conocer los habitantes del ambiente en todo momento, el sistema debe reconocer los eventos de entrada y salida del ambiente por parte de los individuos. Se han desarrollado varias soluciones a este problema, variando en complejidad y usabilidad, algunos ejemplos son: las tarjetas infrarrojas que detectan a los individuos automáticamente [31] y mecanismos de reconocimiento de voz o de imagen como Blink [32] en sistemas Windows.

2.2.2 *Estrategias de recomendaciones grupales*

El objetivo de esta sección es mostrar algunas estrategias utilizadas para generar recomendaciones para un grupo dado, las cuales están definidos en el trabajo de Judith Masthoff en *Group Modeling: Selecting a Sequence of Television items to Suit a Group of Viewers* [24]. Los ejemplos se basan en ese trabajo y plantean un escenario con tres individuos (John, Adam y Mary) y diez elementos entre los cuales elegir (A, B, C, D, E, F, G, H, I, J). A su vez, cada ejemplo corresponde a una estrategia diferente y el resultado siempre corresponde con la lista ordenada desde el elemento que mejor satisface las condiciones de la estrategia, hasta el que menos la satisface. Vamos a describir las estrategias utilitarias y la estrategia del *menos feliz*, en el Anexo A: Estrategias de Recomendaciones Grupales se pueden encontrar el resto de las estrategias para recomendaciones grupales.

2.2.2.1 Estrategia utilitaria

Se utilizan valores utilitarios para medir la felicidad de cada individuo en cada alternativa. El valor utilitario generalmente se corresponde con la calificación del individuo en dicha alternativa. Se destacan dos versiones de esta estrategia, la aditiva y la multiplicativa.

- **Aditiva (o promedio)**

Para cada alternativa se realiza la suma de los valores utilitarios de cada individuo. Luego, las alternativas son ordenadas según el resultado de la suma. Debemos notar que si se realiza el promedio de los valores utilitarios de cada alternativa, el orden de la lista sigue siendo el mismo, por ello a esta estrategia también se la conoce como la “estrategia media”. Supongamos la siguiente tabla con los valores utilitarios de cada individuo en cada alternativa:

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	21	18	13	22	26	26	17	23	20	22

Group List: (E, F) H (D, J) A I B G C

Ilustración 2: Ejemplo Estrategia Utilitaria

Fuente: Group Modeling: Selecting a Sequence of Television items to Suit a Group of Viewers [24]

En este ejemplo las alternativas ganadoras son E y F con una suma de 26 y un valor utilitario medio de 8,67, seguidas de H con una suma de 23 y un valor utilitario medio de 7,67. Aquí el sistema que ofrece la recomendación deberá contar con políticas para resolver si presentar ambas u optar por una alternativa ganadora. La suma y valor utilitario medio para éstas y el resto de las alternativas se resume en la tabla 4:

Alternativa	Suma	Valor medio	Orden
A	21	7,00	4
B	18	6,00	6
C	13	4,33	8
D	22	7,33	3
E	26	8,67	1
F	26	8,67	1
G	17	5,67	7
H	23	7,67	2
I	20	6,67	5
J	22	7,33	3

Tabla 4: Resultados de ejemplo de estrategia Aditiva

- **Multiplicativa**

Para cada alternativa se realiza la multiplicación de los valores utilitarios de cada individuo. Luego las alternativas son ordenadas según el resultado de la multiplicación. Si bien esta versión

es muy similar a la aditiva la lista resultado puede ser muy diferente. Tomando como ejemplo la tabla que se muestra en la ilustración 2, la opción ganadora es F con una multiplicación de 648 y ya no hay empate a diferencia de la versión aditiva.

2.2.2.2 Estrategia del menos feliz

Fue la estrategia elegida para el diseño del sistema PolyLens que ya mencionamos antes en este capítulo. Para cada opción se considera la menor calificación otorgada por cualquier individuo. Luego se ordenan los resultados de manera descendente y la opción que encabeza la lista es la ganadora. La idea de esta estrategia es que la felicidad del grupo es dictada por la felicidad del individuo menos feliz. Si bien para grupos pequeños puede ser una buena estrategia, para grupos grandes tiene el inconveniente de que aunque a una enorme mayoría le encante una determinada película, podría pasar que la misma nunca se elija como ganadora porque hay un solo individuo que la detesta [37][29].

2.2.3 Consideraciones sobre privacidad

Al momento de trabajar con grupos existen algunas consideraciones de privacidad a tener en cuenta. La privacidad es un problema importante a la hora de generar recomendaciones para grupos, principalmente si en pro de la transparencia, el sistema indica que determinado ítem se muestra porque a un integrante particular del grupo le interesa [33]. La utilización de datos personales puede ser revelada o no a los usuarios, diferentes implementaciones hacen esta información disponible a los usuarios, la ocultan o dejan el control a cada usuario particular [25]. Para mejorar la privacidad, se podría agregar una medida de privacidad entre 0 y 1 a cada elemento para que el usuario configure junto con la valoración. Así una medida de privacidad de 0, indicaría que la valoración solo se podría utilizar para generar recomendaciones individuales. Una medida de privacidad de 1 indicaría que al usuario no le interesa en absoluto si otros usuarios conocen la valoración que ha otorgado al elemento [33]. De esta manera, contar con una medida de privacidad puede ayudar, pero podría no ser suficiente para asegurar la privacidad. Puede pasar que un usuario quiera cambiar de opinión sobre un determinado elemento dependiendo en el contexto y grupo en el que se encuentre. Para dar solución a este problema se puede optar por mantener diferentes valoraciones del individuo sobre los ítems dependiendo del contexto en el que se encuentre. Para ello se podría soportar múltiples personalidades de un individuo o que un individuo pueda presentarse ante el sistema usando diferentes usuarios. Una desventaja de cualquiera de estas soluciones es que el individuo necesita estar cambiando explícitamente su personalidad o el usuario ante los cambios en la conformación del grupo, lo cual podría ocasionar inconvenientes de privacidad al revelar al resto del grupo las diferentes personalidades o usuarios que tiene un individuo [33].

Por último, según Judith Masthoff la protección de la privacidad también es importante a la hora de elegir una estrategia de recomendación grupal [33]. Masthoff concluye en su artículo *In Pursuit of Satisfaction and the Prevention of Embarrassment: Affective State in Group Recommender Systems* que se debe evitar usar la estrategia del menos feliz y califica como “buena” la estrategia utilitaria multiplicativa, que a priori parecía ser la que mejor privacidad

ofrecía. También se sugiere una opción para disimular las preferencias de los usuarios: la utilización de un usuario virtual con gustos propios. De esta manera, el resultado de una recomendación nunca se podría atribuir a un individuo en particular ya que siempre existe la posibilidad de que haya sido influenciada por el usuario virtual. También se podría valorar como positiva la integración de un usuario virtual para agregar un poco de casuística a las recomendaciones [33].

2.3 Sistemas de Recomendación de Referencia

Luego de haber estudiado los sistemas de recomendaciones, vamos a estudiar casos prácticos de recomendaciones de películas. Estas aplicaciones pueden tener algunas características del prototipo que queremos construir. El objetivo es hacer un análisis crítico de cada uno, encontrar las fortalezas y debilidades dentro de las siguientes características:

- Funcionalidades sociales.
- Integración con redes sociales.
- Recomendaciones.
- Proveedores de Video soportados.
- Interacción con proveedores de video bajo demanda.
- Protocolos y Estándares.
- Dispositivos soportados.

Primero exploramos diferentes sistemas de recomendaciones de películas, dentro los cuales elegimos siete para profundizar: Internet Movie Database, Rotten Tomatoes y Flixter, MovieLens, Hulu, Nanocrowd, Jinni, Metacritic. En algunos casos los elegimos por su popularidad, por la cantidad de usuarios, por el tráfico que generan, y en otros por alguna particularidad técnica que nos pareció interesante, donde encontramos una buena documentación y datos abiertos. En el Anexo B: Sistemas de Recomendación de Referencia, se puede profundizar en cada sistema de referencia.

Nota: Para obtener datos estimados del tráfico generado por los sitios analizados, se utilizaron las herramientas Alexa [34] y Compete [35].

Luego de haber explorado estos sistemas de recomendaciones de películas, encontramos una posible categorización en cuanto a los algoritmos de recomendación que utilizan: Filtrado Colaborativo, Basado en el Contenido y un Híbrido entre estos dos. En algunos casos encontramos conceptos de los sistemas basados en contenido como ser una fuerte presencia de los especialistas del cine, analizando las películas y desagregando su contenido en variables que se toman en cuenta a la hora de realizar las recomendaciones. En otros casos, vemos que usan técnicas de filtro colaborativo en los cuales entendemos que tiene más importancia la cantidad de usuarios que aportan datos a este modelo.

En cuanto a la información que ofrecen los sitios como servicios, vemos que no están estandarizados los metadatos de las películas. Si estuviera estandarizado se podrían cruzar los datos de diferentes fuentes con cierta fidelidad. Si bien se pueden obtener los datos e intentar igualarlos de alguna manera, se presentan algunas dificultades para hacerlo.

En la Tabla 5, resumimos cada sistema, nombramos el algoritmo de recomendación utilizado, indicamos si ofrece su catálogo de forma pública, listamos los proveedores de películas externos asociados e indicamos si se integra con redes sociales.

Sistema	Algoritmo de recomendación	Ofrece catálogo	Proveedores video externos	Integración con Redes sociales
IMDb	Híbrido	Si (archivos de texto plano vía ftp)	Amazon Instant Video, Internet Archive	Facebook, Twitter
Rotten Tomatoes y Flixter	Basado en contenido	Si (API Rest)	NetFlix, Ultravioleta	Facebook, Myspace, Bebo, Orkut
MovieLens	Filtrado colaborativo	No	No	No
Hulu	Por género	Si (API Rest)	No	Facebook, Twitter
Nanocrowd	Basado en contenido	No	Hulu, Crackle, Amazon Instant Video, Netflix, Internet Archive	No
Jinni	Basado en contenido	Si (API Rest)	NetFlix, Amazon, Hulu, LoveFilm, Blockbuster	Facebook, Twitter
Metacritic	No ofrece	No	IVA (Internet Video Archive), Youtube	Facebook, Twitter

Tabla 5: Tabla comparativa – Sistemas de Recomendación de Referencia

2.4 Proveedores de video bajo demanda

Luego de haber estudiado algunos casos prácticos de sistemas de recomendación, vemos que el contenido multimedia se brinda desde diferentes fuentes. Motivados por investigar el problema que plantea la integración de dichas fuentes, hemos analizado el estado actual de los proveedores de contenido multimedia. Nos centramos en lo que refiere a video bajo demanda y las posibilidades que brindan para poder integrarlos desde otros sistemas. Dentro de la vasta gama de proveedores en este punto presentamos el análisis realizado de aquellos más populares, profundizando en los más ricos en cuanto a posibilidades de integración. Mencionaremos aquellos aspectos que consideremos relevantes y/o que hayan tenido influencia a la hora de optar por incorporar o no los distintos proveedores a nuestra investigación.

2.4.1 Internet video archive

Internet Video Archive (IVA) [36] es un agregador de avances de películas y TV, avances de juegos y videos musicales. Este administra vínculos y fuentes de más de quinientos mil videos exclusivos de más de mil proveedores de contenido. Entre los socios de IVA se encuentran: Rotten Tomatoes, Barnes & Noble, Fandango, Flixster, Sony, Metacafe, Daily Motion, Hulu, etc. El catálogo de videos de películas cubre cinematografías que van desde 1930 hasta el día de hoy

donde se incluyen entrevistas, detrás de escena, vistas previas de DVDs, avances de películas en distintos lenguajes. A su vez los videos se encuentran organizados y categorizados, asociados con precisa meta-información que hace posible a los consumidores encontrar lo que buscan a través de sus servicios. En dicha meta-información se definen títulos, cast, directores, resúmenes de la trama, calificaciones MPAA¹, duración, fechas de publicación en distintos países, etc. La información del catálogo se encuentra sincronizada con las bases de datos de Rovi, AMG, IMDB, Baseline, Flixster, Hulu, Metacritic, Crackle, Netflix, Itunes, Amazon ASIN entre otros. En lo que respecta a la computación en la nube de sus servicios de datos, su sistema utiliza Microsoft Windows Azure². Ofrece sus servicios y la información que administra implementando el estándar OData API³, la cual provee gran variedad de opciones para integrarla fácilmente en distintas plataformas.

2.4.2 Netflix

Netflix [37] es un proveedor de videos bajo demanda. Cuenta con más de 23 millones de usuarios en Estados Unidos y Canadá y ha abarcado una importante cuota del mercado latinoamericano. Este ofrece sus servicios en Estados Unidos, Latinoamérica, Irlanda, Gran Bretaña y Canadá. Este ofrece una API de datos a la que se puede acceder autenticándose mediante el protocolo OAuth si se cuenta con una clave de desarrollador. Con esta API puede buscarse dentro de su catálogo información sobre películas, series, programas de TV, etc., que incluye desde el título, director, reparto, calificación popular de sus usuarios, calificación MPAA, premios y nominaciones, hasta detalles referentes a los formatos y la calidad de reproducción disponibles [38]. Ofrece también productos para integrarse en plataformas de video juegos PS3, PS Vita, Wii, XBOX, televisores LG life Good, Samsung Smart TV, dispositivos móviles como Android, iPad, iPhone y WP7, y para reproductores de transmisión de datos WD Western Digital.

2.4.3 Crackle

Este sitio provee un listado de películas que rota cada cierto tiempo, las cuales pueden verse gratuitamente desde su sitio Web [39], aunque también puede encontrarse a través de otros proveedores de servicios de video. Un ejemplo es Google TV donde puede encontrarse en Google Spotlight TV sección TV & Movies⁴. Por esto mismo, es de esperar que el contenido no es de lo más actualizado, las películas más recientes que pueden encontrarse tienen fecha de publicación de hace cinco años anterior a la fecha actual.

2.4.4 Redbox

Redbox [40] es un servicio de alquiler de películas y video juegos muy popular en Estados Unidos. Este servicio permite a los usuarios buscar películas incluso a partir del historial de gustos del usuario, ver información, agregarlas a su carro de compras para finalmente pagar su alquiler directamente en los kioscos/redboxes. En cuanto al aspecto técnico, este servicio ofrece

¹ Motion Picture Association of America www.mpa.org

² Microsoft Windows Azure www.windowsazure.com

³ OData API www.odata.org

⁴ Google TV Spotlight <http://www.google.com/tv/spotlight-gallery.html>

una API RESTful [41] para navegar y obtener información de sus productos. Las respuestas están disponibles en formato XML, JSON y JSONP. Esta API, si bien necesita una clave de desarrollador para poder consumirla, sólo permite obtener información de los productos. Para concretar reservas y realizar pagos deberá realizarse directamente en el sitio de RedBox.

2.5 Multidispositivo

En la actualidad existe una gran variedad de dispositivos conectados a Internet. Hasta hace algunos años la pantalla preferida para la visualización de videos era casi exclusivamente la televisión tradicional. Sin embargo, hoy en día podemos observar un crecimiento del uso de tabletas, consolas, *set top boxes*, teléfonos y televisores inteligentes, computadoras portátiles, entre otros. Estos dispositivos permiten la reproducción de contenidos multimedia con las más variadas características en tamaño, peso, hardware y software [42]. El tamaño de la pantalla, su resolución, la forma de ingresar los datos, las capacidades del sistema operativo, los formatos de audio y video soportados y el navegador de Internet son algunas de las principales características a tener en cuenta a la hora de desarrollar aplicaciones para estos dispositivos. Entender estas características es importante para la construcción del prototipo para nuestro trabajo. Por ello, en las siguientes secciones estudiaremos los conceptos de televisión en todos lados, primera y segunda pantalla. Luego veremos algunas pautas para el desarrollo de aplicaciones para los dispositivos analizados.

2.5.1 La Televisión en todos lados

Hace algunos años se podía ver al telespectador como pasivo, sentado frente al televisor en el living de su casa observando lo que ha sido programado por su cable operador. Hoy en día los individuos tienen la posibilidad de ver los contenidos en cualquier momento y en cualquier lado. Ver el noticiero en el teléfono inteligente mientras se viaja en algún medio de transporte, o ver una película en la cama con la tableta utilizando algún servicio de video como Netflix, Hulu, Amazon o Youtube son algunos de los ejemplos que podemos citar [42]. A los consumidores de contenidos no les importa cómo es que los servicios multimedia de entretenimientos son entregados ni a través de qué tipo de redes lo hacen. Tampoco les interesa si la tecnología utilizada es analógica o digital, QAM o IP, solo les interesa el tipo de servicio que es ofrecido, no los tipos de forma de entrega del servicio o los tipos de redes utilizadas.

Según el artículo de *Alan Breznic* para *Cisco* [43], algunos estudios indican que los consumidores disfrutan de los servicios DVR basados en multi-habitación o red. En este artículo se mencionan diferentes investigaciones que muestran cómo a los consumidores les gusta sentir la libertad de mover los contenidos transparentemente dentro del hogar, y particularmente mover videos HD entre los diferentes tipos de pantallas. Los consumidores quieren ser capaces de ver contenidos en vivo, a demanda, grabados, online, todos en una misma pantalla no separados en diferentes dispositivos. Es por esta razón que los consumidores de contenidos, los individuos que hacen uso de los nuevos dispositivos, están comenzando a ver estos dispositivos tan solo como pantallas [44].

En este contexto, los cable operadores están cambiando sus estrategias para atraer y conservar a sus clientes. Es aquí donde nace el modelo de “televisión en todos lados”. Ejemplos de productos y servicios con estas características son: HBO GO [45], Moviecity Play [46], ESPN Play [47], entre otros. El modelo de televisión en todos lados o *TV Everywhere* por su nombre en inglés, se basa en los siguientes principios: beneficio solo para suscriptores de pago; más y mejor contenido disponible para su distribución, accesible desde cualquier dispositivo conectado y en cualquier momento; contenido en vivo y bajo demanda; algunos programas y series pueden visualizarse en vivo y otros pueden accederse en cualquier momento a demanda [48].

2.5.2 La primera pantalla / pantalla principal

El concepto de primera pantalla se refiere al dispositivo principal utilizado para la visualización de contenidos. Históricamente y hasta ahora la televisión ha sido el dispositivo principal preferido, aunque dando lugar cada vez más a otros dispositivos como los notebooks, tablets y smartphones que si bien se siguen considerando como segunda pantalla están ganando terreno en la batalla por la pantalla principal [42]. De esta manera, la televisión y los dispositivos conectados que utilizan la pantalla de la TV para la reproducción de contenidos, son los que consideramos “la primera pantalla” y son de vital importancia en la era de la televisión social. Sin ellos no tendría sentido la segunda pantalla y las aplicaciones complementarias como comentarios, chats y la interacción con otros televidentes[49].

2.5.2.1 Dispositivos para la primera pantalla: Televisores conectados y Televisores inteligentes

De la misma forma que los teléfonos inteligentes compiten con los teléfonos tradicionales, incluyendo nuevas características como conexión a Internet, navegadores, correo, redes sociales, entre otras; los televisores conectados y los televisores inteligentes compiten con los televisores tradicionales. Es delgada la diferencia entre lo que son los televisores conectados y los que son televisores inteligentes. Un televisor inteligente, es un dispositivo que además de tener acceso a Internet (y a redes hogareñas) para realizar *streaming* de contenidos, tiene características adicionales, entre ellas y según *Intel* [50], podemos destacar:

- Búsqueda de contenido en línea y personal, así también como de programación, todos desde la misma interfaz de la TV.
- Descarga de aplicaciones adicionales (típicamente desde una tienda de aplicaciones).
- Conexión a redes sociales mientras se ven películas o programas favoritos.
- Control de la TV mediante un único control remoto o comandos de voz.
- Acceso a una infinidad de posibilidades de entretenimientos [60][61].

En el contexto de este trabajo, las características que pretendemos analizar son indistintas sin importar la categorización exacta del televisor ya sea como dispositivo conectado o inteligente. En el Anexo C: Tablas de Dispositivos, esta la lista de los Televisores estudiados y sus características, *Set top boxes* y consolas de videojuego, al igual que los reproductores de *blu-ray*,

son dispositivos que se conectan directamente a la TV y que ejecutan un *middleware* capaz de proporcionar características similares a un televisor conectado o inteligente [51].

2.5.3 Segunda pantalla

Segunda pantalla es un término que se refiere a un dispositivo electrónico adicional (por ejemplo, tabletas, teléfonos inteligentes) que permite a una audiencia de televisión interactuar con el contenido que están consumiendo, ya sea en programas de TV, películas, música, o juegos de video. Datos adicionales se muestran en la segunda pantalla sincronizada con el contenido que se está viendo en el televisor [52]. Las aplicaciones para la segunda pantalla no son diseñadas para mostrar contenidos tradicionales - como un programa o película - sino que se diseñan para agregar una “capa social” a la experiencia de visualización de contenidos [53]. Los dispositivos móviles tales como los teléfonos inteligentes y tabletas son actualmente la opción más usada como segunda pantalla. En el Anexo C: Tablas de Dispositivos, esta la lista de los teléfonos inteligentes y tabletas con sus características.

2.5.3.1 Tabletass

Una tableta es un tipo de computadora portátil, de mayor tamaño que un teléfono inteligente o una PDA, integrado en una pantalla táctil (sencilla o multitáctil). Mientras que las primeras tabletas requerían del uso de una pluma stylus, las tabletas modernas son operadas primariamente con los dedos y el uso de la pluma es opcional. No requieren de teclado físico ni ratón. Estos últimos se ven reemplazados por un teclado virtual y, en determinados modelos, por una *mini-trackball* integrada en uno de los bordes de la pantalla. Una tableta funciona como una computadora, solo que más orientada a la multimedia, lectura de contenidos y a la navegación Web que a usos profesionales. Entre sus principales utilidades se destacan: lectura de libros electrónicos, consulta y edición de documentos ofimáticos, navegación Web, llamadas telefónicas, GPS, reproducción de música, visualización de vídeos, cámara fotográfica y de video HD, videoconferencia, juegos, entre otros. Todas las tabletas vienen con un navegador Web incorporado y una variedad de aplicaciones instaladas. Las aplicaciones adicionales se pueden conseguir habitualmente desde la tienda de aplicaciones en línea del vendedor [55].

2.5.3.2 Teléfono inteligente

Un teléfono inteligente (*smartphone*) es un teléfono móvil con mayores características de cómputo y conectividad que un teléfono estándar. Los primeros teléfonos inteligentes combinaban principalmente las funciones de un asistente digital personal (PDA) y un teléfono móvil o teléfono con cámara. Los modelos de hoy también combinan las funciones de reproductores multimedia portátiles, cámaras digitales compactas de bajo coste, cámaras de vídeo de bolsillo, y unidades de navegación GPS. Los teléfonos inteligentes modernos suelen incluir pantallas táctiles de alta resolución, navegadores Web que pueden acceder y mostrar correctamente páginas Web estándar y tienen alta velocidad de acceso a datos a través de Wi-Fi de banda ancha móvil [56].

2.5.4 *Pautas de diseño para interfaces de usuario*

En el Anexo D: Pautas de diseño para interfaz de usuario, analizamos los principales aspectos que debemos considerar durante la etapa de desarrollo de las diferentes interfaces de usuario para cada pantalla. Comenzamos resumiendo las pautas generales de diseño, para luego centrarnos en las pautas específicas para la televisión como primera pantalla.

3 Solución Propuesta

Luego de haber estudiado los diferentes conceptos que tocaban de alguna manera nuestro proyecto, pasamos a describir la solución propuesta. El principal objetivo que abarcamos es recomendar películas para un grupo de individuos. Esto quiere decir que se conforma un grupo de individuos con el objetivo de ver una película y el sistema le recomienda la más adecuada para ese grupo. El grupo puede estar físicamente en el mismo lugar, visualizando la misma pantalla o puede estar unido virtualmente.

Nuestra solución presenta dos grandes componentes: un servicio de recomendaciones genérico y un sistema de recomendación de películas que utilice este servicio. El servicio de recomendación genérico tiene como principal cometido implementar los algoritmos de recomendaciones individuales y grupales. Este se abstrae del objeto particular a recomendar, recomendando ítems de forma genérica.

El sistema de recomendación de películas es un prototipo para dispositivos móviles. Este prototipo permite a los usuarios formar grupos y obtener recomendaciones a partir de una cartelera cinematográfica, facilitando la decisión de qué película elegir. Este sistema ofrece a los usuarios información sobre películas. Como parte de la solución resolvemos el problema de la carga inicial y sincronización de datos. Además, construimos otra vista Web del sistema que tiene las funcionalidades básicas de recomendación y facilita la reproducción de películas desde proveedores externos. En la ilustración 3 se muestra una vista general de la solución propuesta.

3. Solución Propuesta

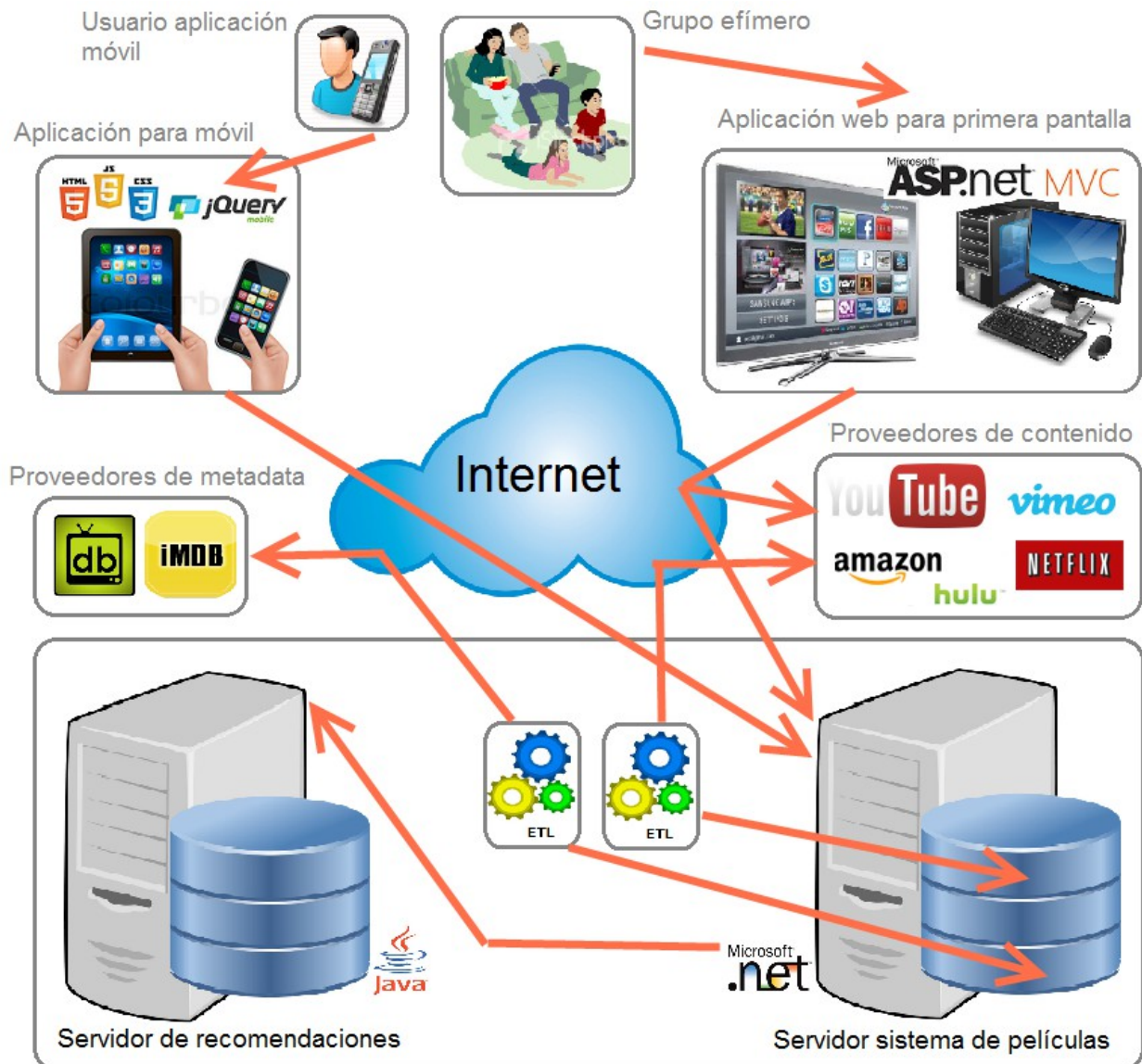


Ilustración 3: Vista general de la solución

En la siguiente sección “Análisis de Requerimientos“, vamos a analizar los requerimientos incluidos en el prototipo y los requerimientos descartados. En la sección “Arquitectura del sistema“ especificamos la arquitectura construida. Luego, en la sección “Algoritmo de recomendación“ hablamos sobre la elección del algoritmo de recomendación individual y grupal. Explicamos las pruebas realizadas y las decisiones tomadas para llegar a los algoritmos de recomendación finalmente implementados. En la sección “Casos de Uso“ mencionamos los casos de uso del prototipo y profundizaremos en los más críticos. Finalmente en la sección “Implementación“ vamos a describir algunos puntos importantes de la implementación del prototipo.

3.1 Análisis de Requerimientos

A partir de lo estudiado en el capítulo 2, nos parece interesante enfocar nuestra solución en recomendaciones grupales. El principal motivo es entender el aporte que se puede realizar a los usuarios que forman grupos desde un sistema de recomendaciones grupales. Esto ha sido menos atacado por la comunidad académica y por los servicios comerciales.

Inicialmente pensamos en construir un prototipo para TV inteligente. En la sección 2.5 sobre Multidispositivos, vimos que técnicamente esta construcción era un trabajo difícil en el marco del proyecto por no contar con la infraestructura necesaria para realizar pruebas sobre una plataforma de TV inteligente. Por este motivo descartamos esa opción para el proyecto. Luego pensamos en los dispositivos móviles como aporte a la conformación de grupos y el flujo de información necesaria para generar recomendaciones.

Entonces, pensamos en un prototipo para dispositivos móviles donde se puedan crear eventos. El objetivo de cada evento es facilitar al grupo la elección de una película para ir a ver al cine. En el evento se ingresa una fecha, se obtiene la cartelera cinematográfica disponible y se invita a participar a otros usuarios. Los usuarios invitados aceptan o rechazan la invitación conformando el grupo que va a elegir una película. Cada miembro del grupo puede ver la información de las películas de la cartelera cinematográfica. El sistema predice la calificación del grupo para cada película y las ordena desde la más recomendable hasta la menos recomendable para el grupo. Luego, cada usuario elige la película que le gustaría ver y esta información le llega al resto de los integrantes del grupo. Con estos elementos el grupo finalmente decide que película va a ver al cine. Cuando termina, cada usuario puede calificar la película desde su dispositivo móvil.

Además, construimos otra vista Web del prototipo que tiene las funcionalidades básicas de recomendaciones y películas: recomendar una película a un individuo, recomendar una película a un grupo, predecir la calificación de un individuo a una película, predecir la calificación de un grupo a una película, consulta de datos de películas, reproducir y calificar películas. Para solucionar el problema de escasez de datos para

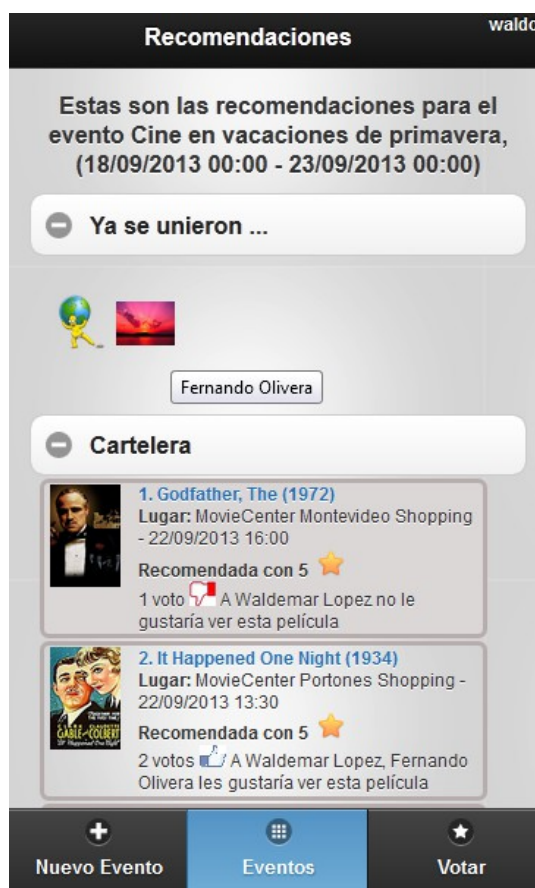


Ilustración 4: Captura de aplicación móvil (Recomendaciones para un evento)

nuevos usuarios, cuando se registra un usuario el sistema sugiere que califique diez películas. Debemos decir que dejamos fuera del alcance del proyecto algunas funcionalidades accesorias al prototipo como: gestión de grupos, reproducción de avances de la película desde el dispositivo móvil, sincronización automática de contenidos y gestión de la cartelera cinematográfica. En las ilustraciones 4 y 5 se pueden ver algunas capturas de pantalla que ejemplifican estas funcionalidades.

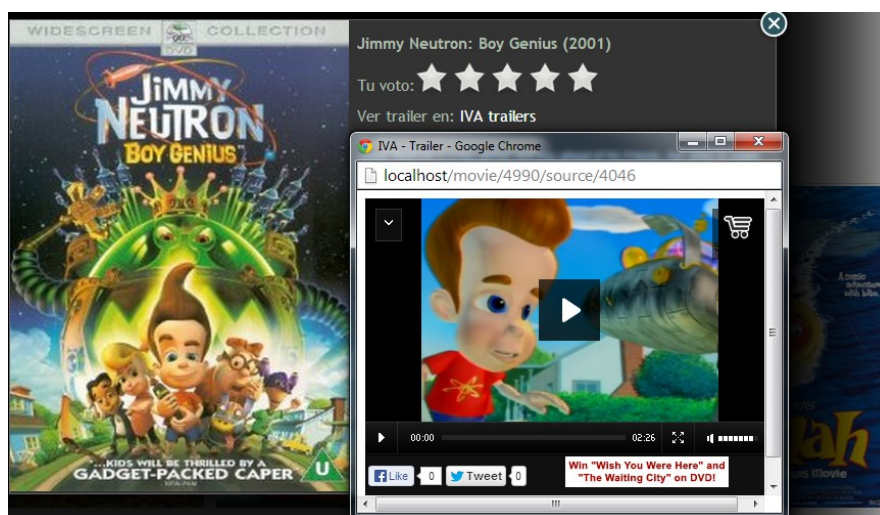


Ilustración 5: Captura de aplicación Web (Reproducción de película)

3.2 Arquitectura del sistema

La arquitectura a nivel macro combina un estilo cliente-servidor, donde los clientes pueden ser: un cliente Web liviano o un cliente móvil HTML5. Se combinan otros estilos arquitectónicos aplicados a nivel de servidor que describimos a continuación.:

- Arquitectura orientada a servicios (SOA) que exponen los principales servicios y procesos de negocio.
- Modelo-vista-controlador (MVC) para dar soporte a la aplicación Web.
- Capa de servicios Web diseñada bajo los principios de una arquitectura RESTful aplicada a una API liviana para dispositivos móviles sobre HTTP.

En el Anexo E: MovieRecommenderApp – documentación de API definimos los principios de una arquitectura RESTful. Además, dada la complejidad de cómputo, demanda de recursos y criticidad en performance para las operaciones del sistema que tienen que ver con las recomendaciones y predicciones, se ha optado por un enfoque distribuido para desacoplar estos servicios genéricos del resto de los servicios del sistema, permitiendo escalar la solución agregando más componentes que ejecuten estos servicios críticos de recomendación. Por la misma razón, tratando de minimizar los cuellos de botella y pensando en una arquitectura escalable, se ha agregado un servidor de imágenes especializado, quitando la responsabilidad al servidor principal que atiende a la aplicación Web o móvil de servir imágenes y crear conexiones HTTP para recursos costosos y solicitados con mucha frecuencia.

Pensando que este sistema pueda a su vez ser parte de algún otro sistema mayor, con sus propios usuarios ya registrados, hemos desacoplado el módulo de autenticación. De esta forma la fuente

de datos con los usuarios y sus parámetros de autenticación (credenciales de acceso, tipo de autenticación, etc.) se encuentre en un sistema externo como puede ser un servidor centralizado de autenticación, o un servidor de directorio, entre otros. De esta forma resulta sencillo cambiar este módulo por otro si se requiere otro mecanismo de autenticación, aumentando así la flexibilidad, evolución y mantenimiento del sistema.

3.2.1 Componentes principales

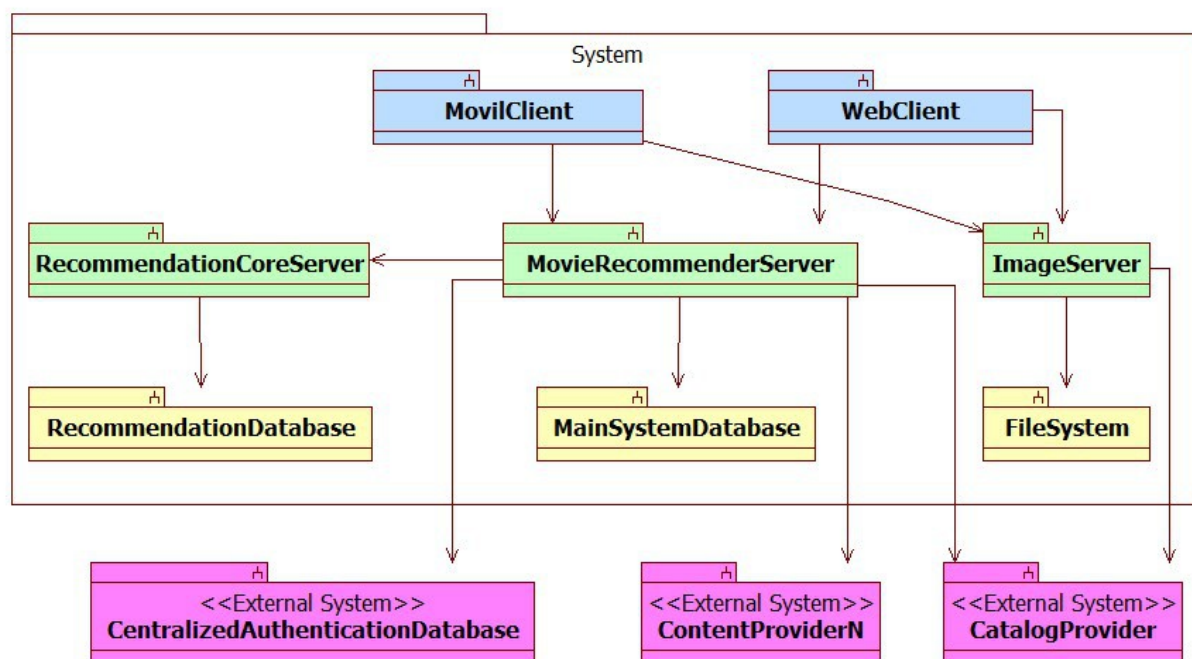


Ilustración 6: Arquitectura - Componentes principales

En esta sección vamos a describir los principales componentes lógicos de la solución presentada. En la ilustración 6, se muestra el diagrama de estos componentes y luego se describen brevemente. Para los componentes más complejos, con un criterio *top-down*, vamos a profundizar en su descripción.

3.2.1.1 Servicio Núcleo de Recomendación (RecommendationCoreServer)

Este componente es el encargado de resolver las recomendaciones grupales e individuales. Como estudiamos en el capítulo 2, los algoritmos de recomendación individual que utilizan la técnica de Filtro Colaborativo tienen como entrada las calificaciones de los usuarios por ítem. Por lo tanto, para este tipo de algoritmo no es necesario saber la naturaleza de los ítems que están recomendando ya que esta es la diferencia con los algoritmos basados en el contenido. Si pensamos en desarrollar un servicio de recomendación que utilice algoritmos de Filtro Colaborativo, podemos ver a los ítems de forma abstracta y no necesitamos conocer su naturaleza. Estos ítems podrían ser Películas, Libros, Video Juegos, Música, etc. Bajo esta premisa, pensamos que lo mejor es encapsular las recomendaciones en este módulo que brinda

las funcionalidades de recomendación como servicios. Estos servicios son capaces de realizar recomendaciones individuales para usuarios sobre ítems y recomendaciones para un grupo de usuarios. Para lograrlo, requieren poder mantener usuarios, ítems y las calificaciones de los usuarios sobre los ítems. Además este servicio debe permitir crear grupos y agregar usuarios a estos grupos.

3.2.1.2 Servidor de recomendación de películas (MovieRecommenderServer)

Este componente resuelve principalmente la lógica de negocio del sistema de recomendaciones de películas. Brinda sus funcionalidades como servicios y permite la autenticación interactuando con el componente externo *CentralizedAuthenticationModule*. Para resolver las recomendaciones se comunica con el componente *RecommendationCoreServer*. Sincroniza el catálogo de películas y los proveedores de contenido para la reproducción con los sistemas externos: *CatalogProvider* y *ContentProvider* respectivamente. En la ilustración 7 detallamos este componente y luego explicamos sus módulos.

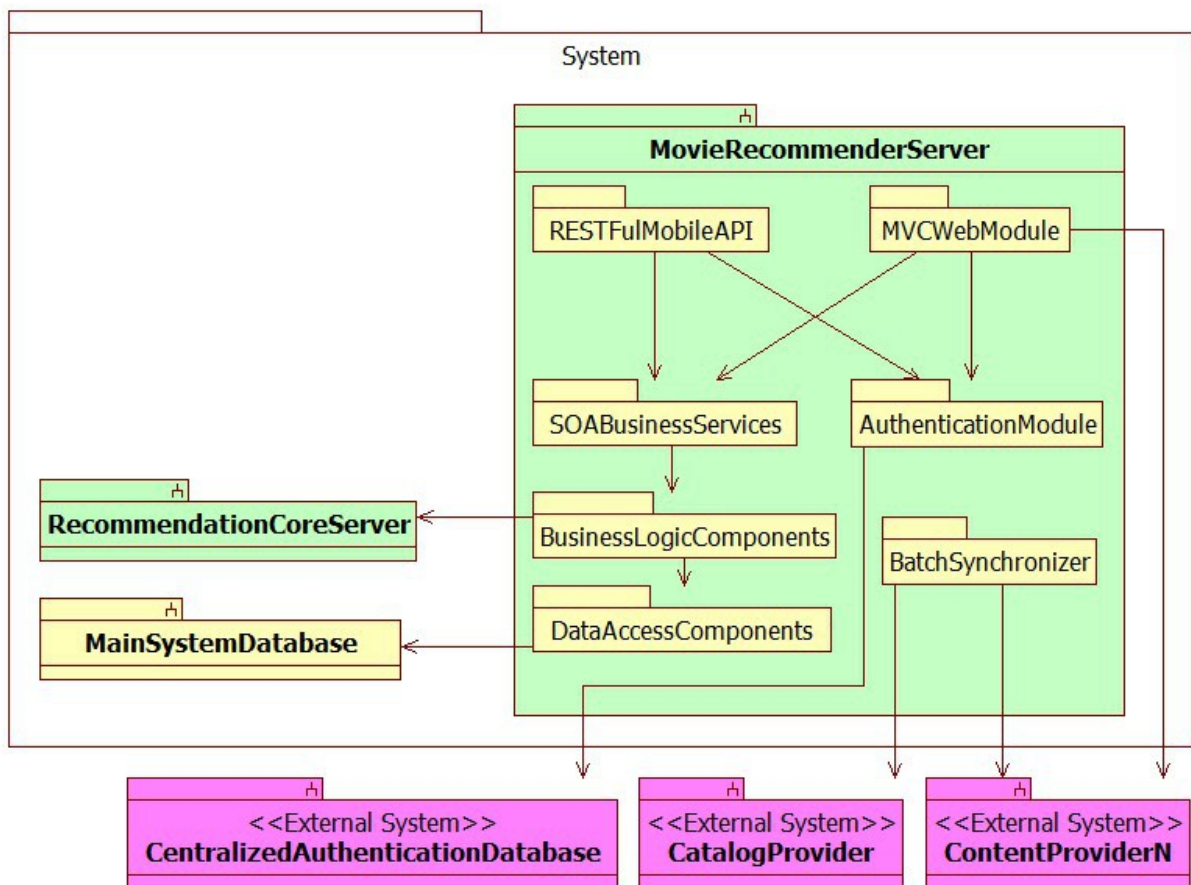


Ilustración 7: Servidor de recomendación de películas

Servicio de negocio (SOABusinessServices): Servicio que provee acceso a las funcionalidades, procesos y flujos de negocio para el dominio particular de las recomendaciones y predicciones de películas, gestión de eventos, usuarios, grupos, etc. Tiene las bondades de los principios de SOA

como modularidad, flexibilidad, escalabilidad, robustez y facilidad de integración. Se comunica con el componente que tiene la lógica del negocio: *BusinessLogicComponents*.

Módulo Web (MVCWebModule): Módulo que ejecuta en el servidor Web. Contiene las páginas y recursos requeridos para dar soporte a la aplicación Web, desde la cual los usuarios pueden solicitar recomendaciones o predicciones genéricas. Este módulo sigue un estilo arquitectónico de la categoría Vista-Modelo-Controlador, donde los controladores son el punto de acceso de los clientes a través de solicitudes Web desde cualquier navegador estándar. Las vistas definen la lógica de la interfaz para un modelo específico. Cada controlador puede decidir la vista a presentar dependiendo de diferentes variables. Por ejemplo, el navegador utilizado por el cliente, podríamos tener una vista para móvil, otra vista para un explorador Web específico y otra vista diferente para otro tipo de navegador. Se comunica con la capa de servicios *SOABusinessServices* para tener acceso a las funcionalidades del sistema y con el módulo *AuthenticationModule* para lograr la autenticación.

API REST (RESTFulMobileAPI): Conjunto de operaciones que definen la interfaz para aplicaciones para dispositivos móviles. Define una capa de servicios Web diseñados bajo los principios de una arquitectura RESTFul, aplicada a una API liviana para dispositivos móviles sobre HTTP. Algunos de estos principios son correspondencia uno a uno entre operaciones CRUD (*create, read, update, delete*) y métodos HTTP. Los detalles técnicos de esta API se pueden encontrar en el Anexo E: MovieRecommenderApp – documentación de API. Se comunica con la capa de servicios *SOABusinessServices* para tener acceso a las funcionalidades del sistema.

Módulo de autenticación (AuthenticationModule): Encapsula la lógica de acceso a las fuentes de datos requeridas para validar a los usuarios y autentica a los usuarios en el sistema. Aumenta la flexibilidad, evolución y el mantenimiento del sistema al ser fácilmente intercambiable por otro módulo con esta responsabilidad si cambian los requerimientos de autenticación. Se comunica con el sistema externo *CentralizedAuthenticationDatabase* que brinda un servicio de autenticación centralizado.

Capa Lógica (BusinessLogicComponents): Implementa la lógica de negocios con todas las funcionalidades principales del dominio de la aplicación, las cuales describimos brevemente:

- Catálogo: Sus tareas principales son la de brindar una interfaz de consulta sobre el catálogo sincronizado y uniformizado, además de operaciones para registrar la actividad de los usuarios frente a los elementos del catálogo de películas (valorar, comentar, visitar, agregar una ubicación del contenido enlazada a un proveedor de contenidos, etc.)
- Usuarios y Grupos: Sus tareas principales tienen que ver con la gestión de usuarios y grupos (permanentes y efímeros). Sus operaciones siguen las buenas prácticas en cuanto a la gestión de la privacidad y gestión de grupos.
- Eventos: Sus tareas principales son la gestión de los eventos: crear, invitar, aceptar/rechazar invitación.

- Recomendaciones: Sus tareas principales son la de generar recomendaciones y predicciones tanto a nivel individual como grupal. Resuelve la comunicación con el módulo de recomendación (*RecommendationCoreServer*).
- Proveedores de contenido: Su principal tarea es la reproducción de contenidos desde los diferentes proveedores de contenidos (video bajo demanda) soportados.

Sincronizador (Batch Synchronizer): Resuelve la sincronización de la información del sistema de catalogo y fuentes de video con los proveedores externos. Para esto también resuelve y se encarga de llevar a cabo la comunicación con los proveedores representados en la ilustración 7 como los sistemas *CatalogProvider* y *ContentProvider*.

3.2.1.3 ImageServer

Servidor de imágenes especializado, se comunica con el componente externo *CatalogProvider* y realiza un cache de imágenes. Intenta mejorar el rendimiento, por un lado liberando la cantidad de hilos disponibles de la aplicación principal para responder a solicitudes de páginas y otros recursos críticos del sistema. Por otro lado mejorando los tiempos de respuesta globales al no competir con otras peticiones del navegador a un mismo dominio, las cuales están topeadas en 6 conexiones simultáneas por dominio (aproximadamente) en todos los navegadores modernos.

3.2.1.4 Capa de presentación

MobileClient: Cliente móvil que accede a la API RESTFul (*RESTFulMobileAPI* del componente *MovieRecommenderServer*) para acceder a las funcionalidades del sistema a través de peticiones HTTP y trabajando con los objetos devueltos por la API. Utiliza *frameworks* propios para dispositivos móviles intentando maximizar la usabilidad y experiencia del usuario. Además, la utilización de HTML5 mejora la portabilidad, el mantenimiento y tiempos de desarrollo, basta desarrollar una única aplicación HTML5 que podrá ser ejecutada en cualquier dispositivo móvil moderno sin necesidad de tener que programar (en un lenguaje diferente, potencialmente desconocido) un cliente nativo por cada tipo distinto de dispositivo. Se comunica con el módulo *MVCWebModule* del componente *MovieRecommenderServer* y con el servidor de imágenes *ImageServer*.

WebClient: Cliente Web que provee a los usuarios funcionalidades de recomendación tanto individual como grupal y permite la reproducción de películas. Se comunica con el módulo *MVCWebModule* del componente *MovieRecommenderServer* y con el servidor de imágenes *ImageServer*.

3.2.1.5 CentralizedAuthenticationDatabase

Componente externo que brinda un servicio de autenticación centralizado, validando a los usuarios y autentica a los usuarios en el sistema.

3.2.1.6 CatalogProvider

Componentes externos que publican sus datos sobre el catálogo de películas que tienen disponibles.

3.2.1.7 ContentProvider

Componentes externos que permiten la reproducción de las películas que tienen disponibles.

3.2.2 Vista de despliegue

A continuación presentamos una posible vista de despliegue de los distintos componentes de software de la arquitectura propuesta. Cabe aclarar que no es la única posibilidad de despliegue para estos componentes, pues existen muchas configuraciones posibles con más o menos nodos (servidores) interconectados. Esta es una de ellas, la cual trata de maximizar el rendimiento del sistema global, considerando los recursos y capacidad de hardware con los que se cuenta.

La configuración de servidores se puede ver dividida en: servidores Web, servidores de aplicación y servidores de base de datos. Los servidores Web alojan los componentes de servidor para dar soporte a la aplicación Web o a los clientes móviles a través de la API REST. En el servidor Web principal también se ubica el componente responsable de la autenticación de usuarios, restringiendo el acceso a usuarios no autorizados directamente sin necesidad de intervención del servidor de aplicaciones principal. También existe un servidor Web especializado para servir contenidos estáticos, en particular imágenes.

Los dispositivos clientes, ya sea un PC o un móvil interactúan con el sistema solicitando y enviando datos a través de Internet hacia los servidores Web indirectamente a través de un *router/firewall*, el cual brinda un punto de acceso al sistema sin exponer directamente los servidores hacia Internet, asegurando un nivel mínimo de seguridad contra intrusos e intentos de acceso no autorizados.

En cuanto a los servidores de aplicación existe un servidor principal para el dominio específico del problema de recomendación de películas y un servidor dedicado específico para ejecutar los algoritmos de recomendación de forma genérica. El servidor de aplicación principal ejecuta los componentes de acceso a datos, lógica de negocio para películas y a su vez expone los servicios de negocio a través de una interfaz de servicio para las capas superiores en una arquitectura orientada a servicios (SOA).

El servidor específico que ejecuta las recomendaciones en esta vista de despliegue es único, pero perfectamente se podría contar con varios de estos y un balanceador de carga por delante para escalar la solución mejorando los tiempos de respuesta para las recomendaciones. Para el caso de las recomendaciones grupales, la posibilidad de aumentar estos servidores de recomendación permitiría partir el grupo de usuarios en tantos subconjuntos como servidores tengamos disponibles, y solicitar a cada servidor las predicciones de gustos para un subconjunto específico de manera concurrente. Luego bastaría procesar los resultados finales en el servidor de aplicaciones principal.

En cuanto a los servidores de base de datos observamos que se requieren al menos tres, uno para persistir los datos específicos en el dominio de las recomendaciones de películas, eventos, grupos, etc., otro para las recomendaciones de ítems de forma genérica y por último un servidor centralizado de autenticación con los usuarios válidos del sistema y sus datos de acceso.

3. Solución Propuesta

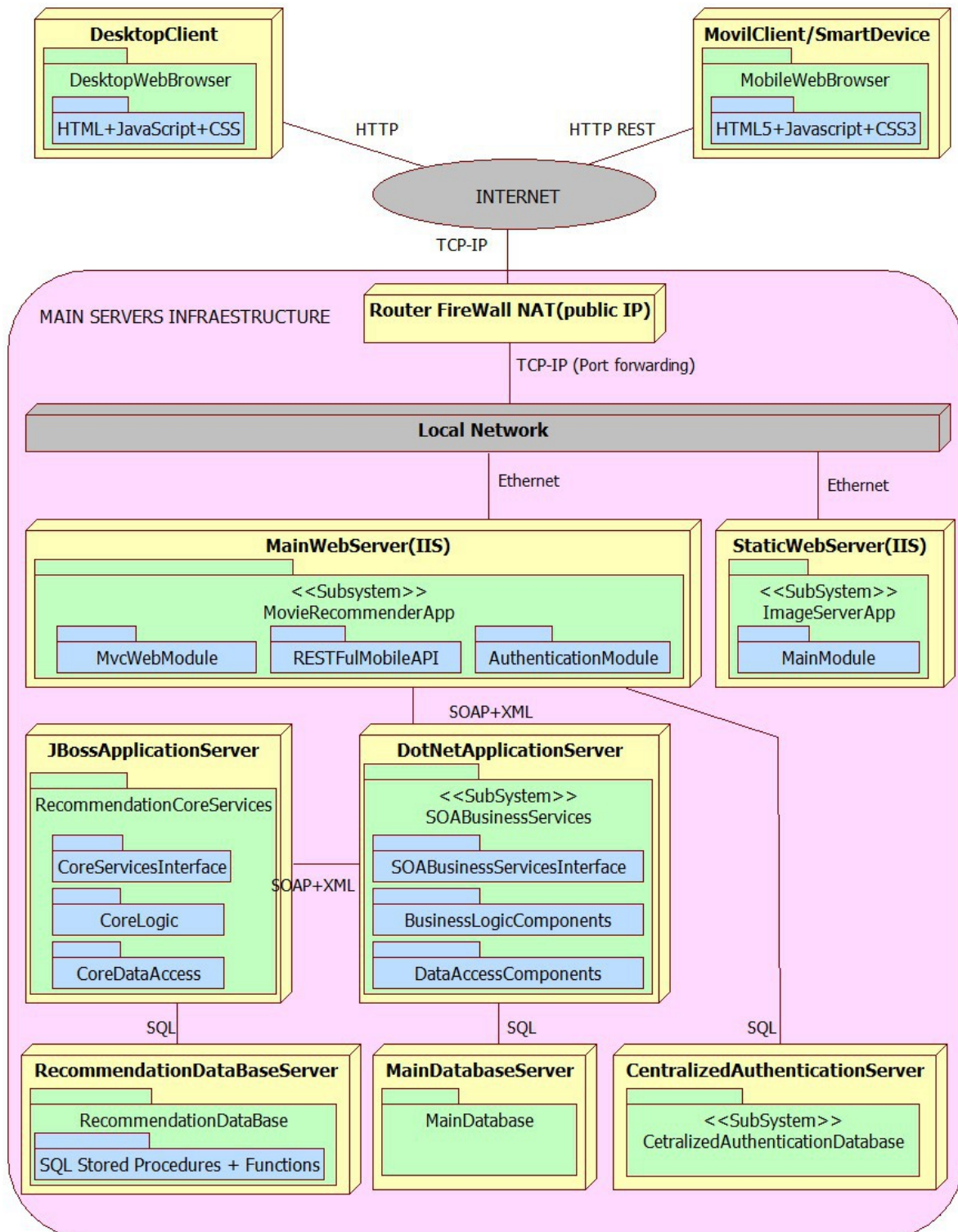


Ilustración 8: Diagrama de despliegue

El único servidor que requiere alojar un componente de software desarrollado durante nuestra solución, es el servidor de base de datos para las recomendaciones de ítems de forma genérica. En esta base datos de recomendaciones necesitamos incluir un conjunto de procedimientos

almacenados y funciones, con el objetivo de maximizar el rendimiento de acceso a datos durante el proceso de generación de recomendaciones y predicciones.

3.3 Algoritmo de recomendación

La solución construida está pensada para utilizar cualquier algoritmo de recomendación grupal y cualquier algoritmo de Filtro Colaborativo para recomendación individual. Recordamos que los algoritmos de recomendación individual que utilizan la técnica de Filtro Colaborativo tienen como variable fundamental la calificación del usuario sobre un ítem. Por lo tanto, se podría implementar cualquiera de estos algoritmos de Filtro Colaborativo a partir de las calificaciones de los usuarios sobre los ítems. En particular para el prototipo, nos planteamos como objetivo principal poder obtener una recomendación grupal de forma rápida, con un resultado de recomendación aceptable para los recursos que tenemos disponibles. Para la recomendación grupal nos basamos en la predicción de la calificación de cada individuo sobre el catálogo de ítems, esto quiere decir que la recomendación grupal se basa en el algoritmo individual de Filtro Colaborativo. Para obtener la predicción individual, elegimos un algoritmo rápido, sencillo de implementar con un rendimiento aceptable. Primero vamos a especificar el algoritmo de recomendación grupal y luego el algoritmo de recomendación individual.

3.3.1 Algoritmo Grupal

Para el algoritmo grupal elegimos la estrategia utilitaria multiplicativa porque cuida la privacidad del grupo según lo estudiado en la sección 2.2.3. Utilizamos como valor utilitario, la predicción de un individuo u sobre un ítem i : p_{ui} , en la sección 3.3.2 vamos a explicar como se calcula. De esta manera definimos la función de predicción de felicidad (o satisfacción) del grupo G sobre un tem i como:

$$felicidad_{Gi} = \prod_{u \in G} (p_{ui})$$

Si calculamos $felicidad_{Gi}$ del grupo sobre un conjunto de ítems, podemos establecer un orden de preferencias del grupo sobre ese conjunto de ítems. Si ordenamos las preferencias en forma descendente, obtenemos la recomendación de uno o más ítems para el grupo G . De esta manera tenemos resuelta la recomendación de ítems para un grupo.

Podemos observar que el mínimo valor de $felicidad_{Gi}$ es 1 y que su máximo valor es 5^n , siendo n el número de integrantes del grupo G .

Nuestro sistema expresa la predicción individual p_{ui} en un rango de valores de 1 a 5. Buscamos que la predicción grupal se base en la preferencia del grupo pero a su vez respete éste rango de valores.

Definimos P_{Gi} como la predicción de calificación (entre 1 y 5) de un grupo G sobre un ítem i .

Este valor se obtiene ajustando el valor de $felicidad_{Gi}$. Para poder expresar P_{Gi} en el rango de valores entre 1 y 5 evaluamos dos posibles soluciones que presentamos a continuación, si bien optamos por una sola para su implementación. Para describirlas utilizamos el ejemplo de

calificación que se muestra en la tabla 6, donde los usuarios u_1, u_2, u_3 han otorgado calificaciones sobre los ítems i_1, i_2 .

Usuario/Item	i_1	i_2
u_1	4	5
u_2	4	5
u_3	4	5

Tabla 6: Ejemplo para predicción grupal

Solución propuesta 1: Fácilmente podemos ver que para los tres usuarios se predicen muy buenos valores para ambos ítems, para i_2 logrando un puntaje perfecto y para i_1 un poco por debajo aunque para los 3 usuarios se predice que i_1 le gustará mucho. Al aplicar la fórmula $felicidad_{Gi}$, obtenemos los valores de 64 ($4 \times 4 \times 4$) y 125 ($5 \times 5 \times 5$) sobre los ítems i_1 y i_2 respectivamente. Al momento de tener que utilizar la escala de 1 a 5, quizá lo más intuitivo es decir que 125 corresponde con la mayor felicidad posible (un 100%) y que 64 es apenas mayor que la mitad de este máximo (la mitad exacta es 62.5) y representa el 51.2% de felicidad, por lo tanto calculamos $p_{Gi1}=2.56$ y $p_{Gi2}=5$, generando la impresión de que al grupo no le interesaría el ítem i_1 , lo cual no es correcto.

Solución propuesta 2: En la solución anterior, vemos que por la variación exponencial de la escala, no se respetan las proporciones de los individuos. Entonces podemos intentar calcular el concepto de promedio de una suma a nivel de producto, lo que se conoce como media geométrica. La media geométrica es relevante cuando varias cantidades son multiplicadas para producir un total. En matemática y estadística, la media geométrica de una cantidad arbitraria de números (por decir n números) es la raíz n -ésima del producto de todos los números. Con este cálculo estamos considerando todos los valores de la distribución y es menos sensible que la media aritmética a los valores extremos. En esta propuesta la ecuación de predicción grupal toma la forma

$$p_{Gi} = \sqrt[n]{felicidad_{Gi}}, \text{ siendo } n \text{ la cantidad de usuario que integran el grupo } G$$

Para el ejemplo de la tabla 6, las predicciones grupales toman los valores $p_{Gi1} = \sqrt[3]{64} = 4$ y $p_{Gi2} = \sqrt[3]{125} = 5$. A nuestro ver esta solución representa mejor la preferencia del grupo, por lo cual es la que optamos por implementar.

3.3.2 Algoritmo Individual

Ahora elegimos la forma de calcular la predicción de la calificación de un individuo u sobre un ítem i : P_{ui} . Como vimos en la sección 2.1.5, el algoritmo Slope One cumple con los objetivos buscados según el estudio realizado por Patrick Marschik [16]: tiene un rendimiento aceptable,

resuelve la predicción de forma rápida y es sencillo de implementar. Si observamos la fórmula del algoritmo Slope One, se define la predicción del usuario u sobre el ítem i como:

$$p_{ui} = \frac{\sum_{j \in [S(u) - \{i\}]} (r_{uj} + dev_{ij})}{|S(u) - \{i\}|}$$

Donde $S(u)$ son los ítems que calificó el usuario u , r_{uj} es la calificación del usuario u sobre la película j y dev_{ij} es la desviación de promedios entre los ítems i y j . De este cálculo podemos desprender que la cantidad de ítems calificados por el usuario incide en el tiempo de ejecución del algoritmo. Si definimos N como el tamaño del conjunto $S(u)$, la fórmula de p_{ui} es de orden $O(N)$ suponiendo que la matriz de desviaciones dev ya está calculada. Si utilizamos un N menor, la cantidad de operaciones será menor. Por otro lado, el algoritmo Slope One con peso, estudiado en la sección 2.1.2.5, pondera las películas con mayor cantidad de calificaciones otorgadas por los usuarios, en el supuesto de que estas pueden contribuir a obtener un mejor resultado. Siguiendo este criterio, reducimos el conjunto $S(u) - \{i\}$, re-definido como $S'(u) - \{i\}$, donde $S'(u)$ es el subconjunto de ítems de $S(u)$ que tienen la mayor cantidad de calificaciones. En la sección 4.2 analizamos el rendimiento del algoritmo y el tiempo de predicción tomando como variable el tamaño de $S'(u)$. De esta manera, definimos la predicción p_{ui} como:

$$p_{ui} = \frac{\sum_{j \in [S'(u) - \{i\}]} (r_{uj} + dev_{ij})}{|S'(u) - \{i\}|}$$

3.4 Casos de Uso

En esta sección presentamos una visión general (ilustración 9) de los casos de uso del sistema. En las siguientes secciones presentamos dos de los casos de uso más críticos para la arquitectura, en el Anexo F: Casos de Uso se especifican con detalle todos los casos de uso. A continuación definimos los actores involucrados en los distintos casos de uso que son tanto usuarios del sistema como sistemas externos invocados desde el Sistema de Recomendaciones de Películas.

- **Usuario Web:** Usuario Principal que tiene como objetivo que el Sistema le recomiende a él o a un grupo que integra una película para luego decidir reproducirla.
- **Usuario Móvil:** Usuario que opera desde un dispositivo móvil y tiene como principal objetivo obtener la recomendación para un grupo de personas que pretende ir al cine. También elige qué película le gustaría ver de la cartelera cinematográfica.
- **Administrador:** Se encarga de la sincronización del Sistema con los proveedores de contenido y los proveedores de catálogo.
- **Servicio Recomendación:** Es un Servicio encargado de los algoritmos de recomendaciones grupales e individuales.

- Proveedor Catálogo: Provee información acerca de las películas. El Sistema puede obtener esta información e incorporarla.

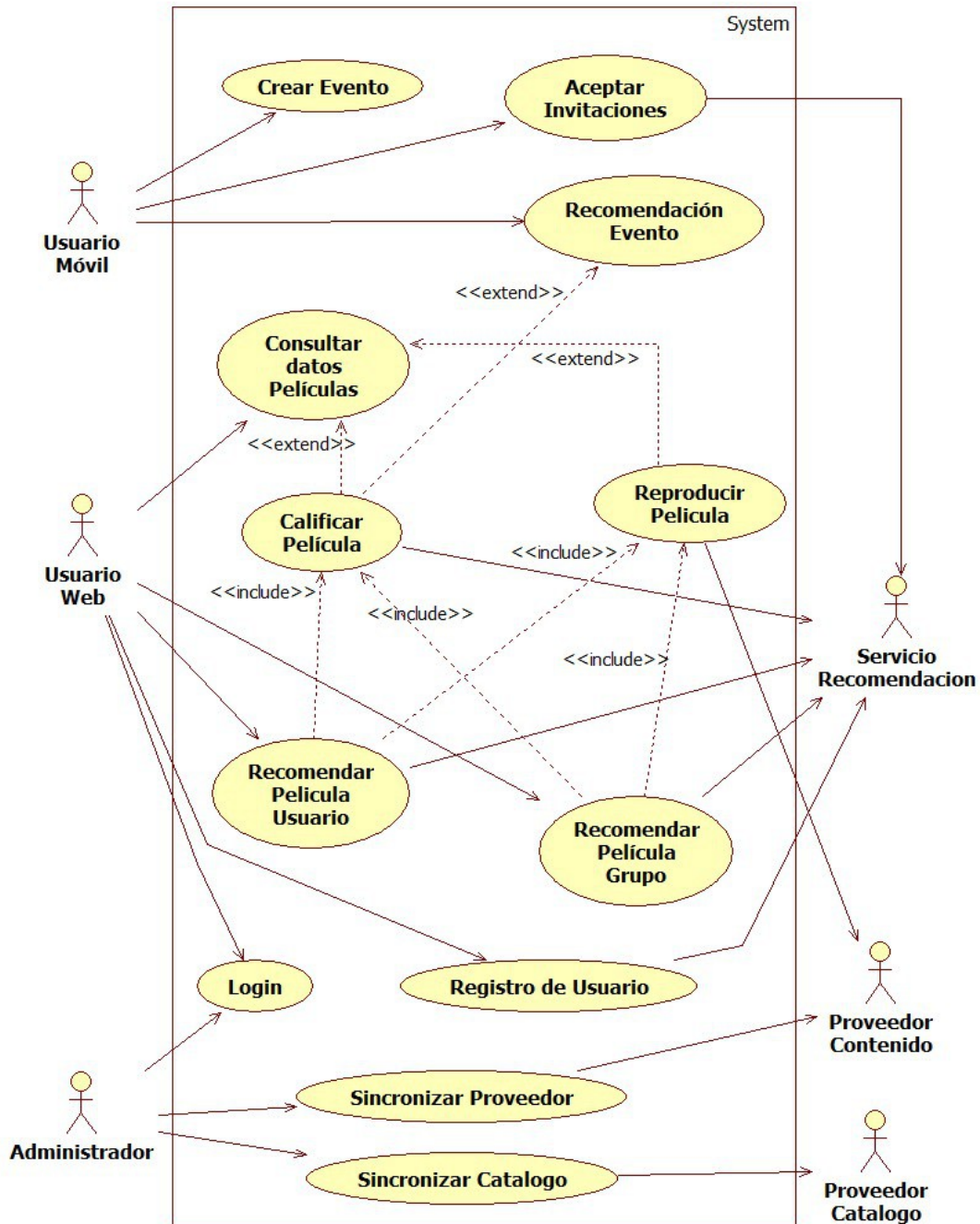


Ilustración 9: Diagrama de Casos de uso

- Proveedor Contenido: Sitio externo que provee la reproducción de películas.

3.4.1 CU6 – Recomendar Película Usuario

El usuario quiere que el sistema le recomiende una película y poder reproducirla. Para mejorar la recomendación, el usuario puede ingresar una serie de datos para filtrar el conjunto de películas a recomendar.

Pre-Condiciones: El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Recomendación de Película para Usuario		
	2 – El sistema presenta los siguientes campos de búsqueda: Título, Reparto, Director, Escritor, País Origen, Idioma, Género. Además permite modificar la cantidad de películas recomendadas, por defecto en 10 películas.	
3 – El usuario ingresa los datos y solicita la recomendación.		
	4 – El sistema solicita al Servicio de Recomendación que ejecute el algoritmo de recomendación individual que está utilizando para el usuario.	
	.	5 – El Servicio de Recomendación ejecuta el algoritmo individual y retorna una lista con la cantidad de ítems recomendables.
	6 – El sistema muestra la lista con la cantidad películas recomendadas de forma ordenada, desde la más recomendable hasta la menos recomendable para el usuario	
7 – El usuario elige una película		
	8 – El sistema muestra los datos de la película.	
9 – Incluir Caso de Uso Reproducción de película		
10 – Incluir Caso de Uso Calificar película		

Flujos Alternativos:

3A – Cantidad recomendada

3A.1 El usuario ingresa los datos, cambia la cantidad de películas recomendadas y solicita la recomendación.

4A – No se encuentran para recomendar

4A.1 El sistema no encuentra ninguna película para recomendar.

4A.2 Vuelve al punto 1.

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones: El sistema le recomendó una lista de películas (por defecto 10) al usuario de las cuales el usuario eligió una para reproducir y luego la calificó.

Diagrama de secuencia: En las ilustraciones 10 y 11 se presenta parte del diagrama de secuencia de la operación más relevante del caso de uso.

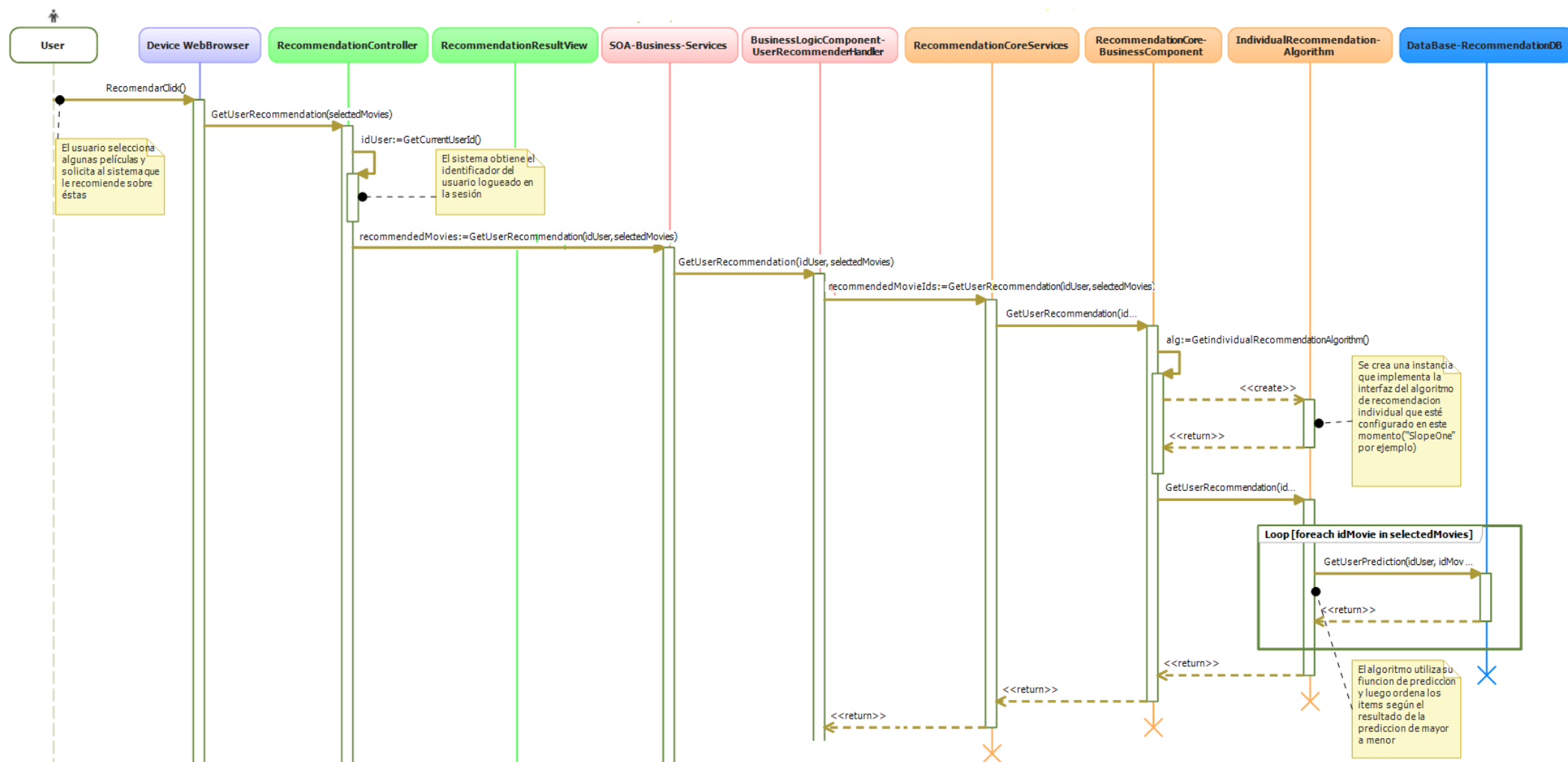


Ilustración 10: Diagrama de Secuencia - Recomendar Película Usuario (1 de 2)

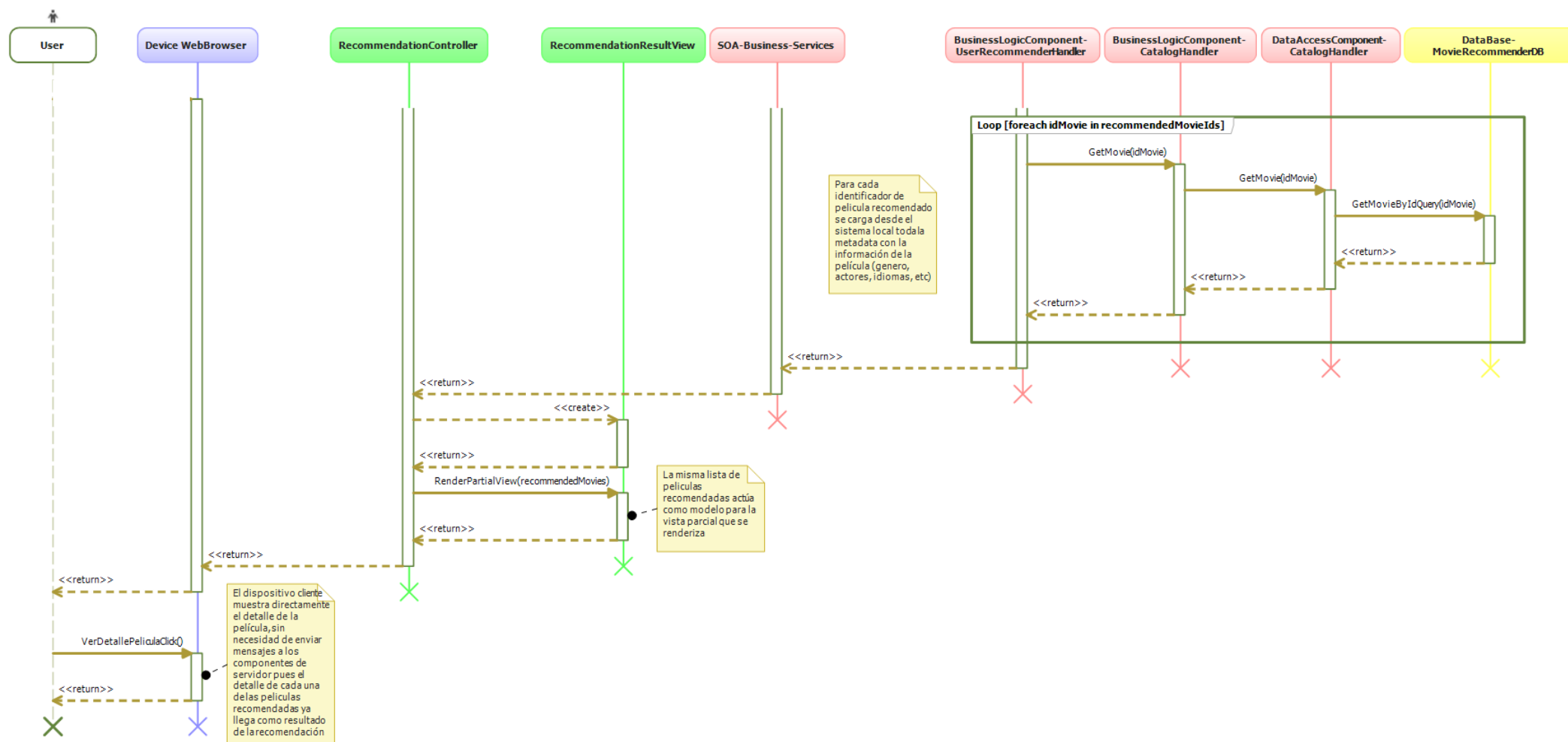


Ilustración 11: Diagrama de Secuencia - Recomendar Película Usuario (2 de 2)

3.4.2 CU9 – Aceptar Invitaciones

Un usuario quiere ver si fue invitado en algún evento para ir al cine y aceptar o rechazar dicha invitación.

Pre-Condiciones: El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Invitaciones.		
	2 – El sistema muestra la lista de invitaciones a las cuales fue invitado el usuario.	
3 – El usuario elige una invitación y la acepta.		
	4 – El sistema agrega al usuario como miembro del grupo del evento. El sistema solicita al servicio de recomendación que recomiende 10 películas para el grupo a partir de la cartelera de películas incluidas entre la fecha de inicio del evento y la fecha de fin.	
		5 – El Servicio de Recomendación ejecuta el algoritmo grupal y retorna una lista de 10 ítems recomendables.
	6 – El sistema actualiza la recomendación para el grupo	

Flujos alternativos:

2A – Sin invitaciones.

2A.1 El sistema indica que no hay invitaciones pendientes para el usuario

2A.2 Finaliza el caso de uso.

3A – Rechazar invitación.

3A.1 El usuario elige rechazar invitación

3A.2 El sistema registra que el usuario rechazó la invitación.

3A.3 Finaliza el caso de uso.

Post-Condiciones: El sistema agregó al usuario como miembro del evento y recalculó la recomendación grupal.

Diagrama de secuencia: En las ilustraciones 12 y 13 se presenta parte del diagrama de secuencia de la operación más relevante del caso de uso.

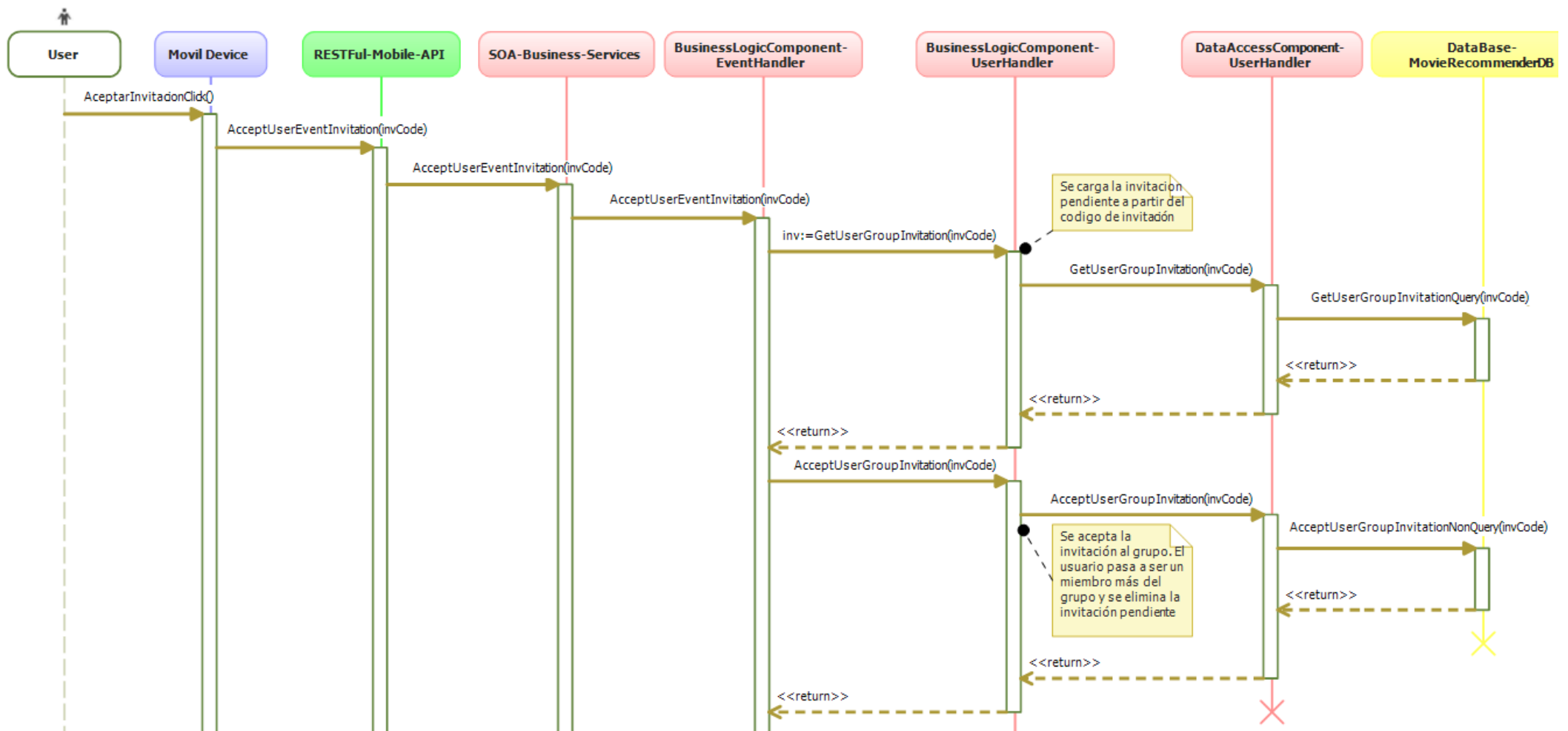


Ilustración 12: Diagrama de Secuencia - Aceptar invitación (1 de 3)

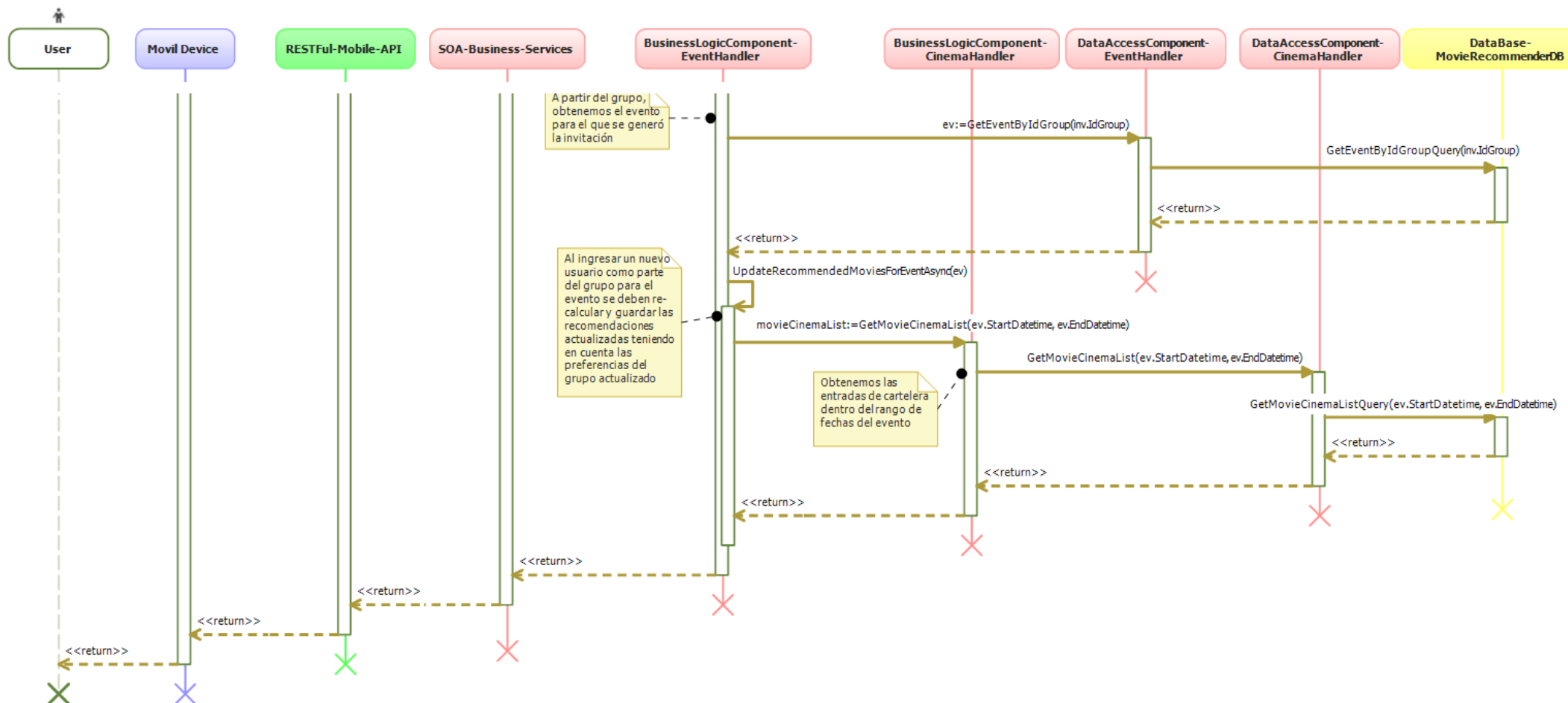


Ilustración 13: Diagrama de Secuencia - Aceptar Invitación (2 de 3)

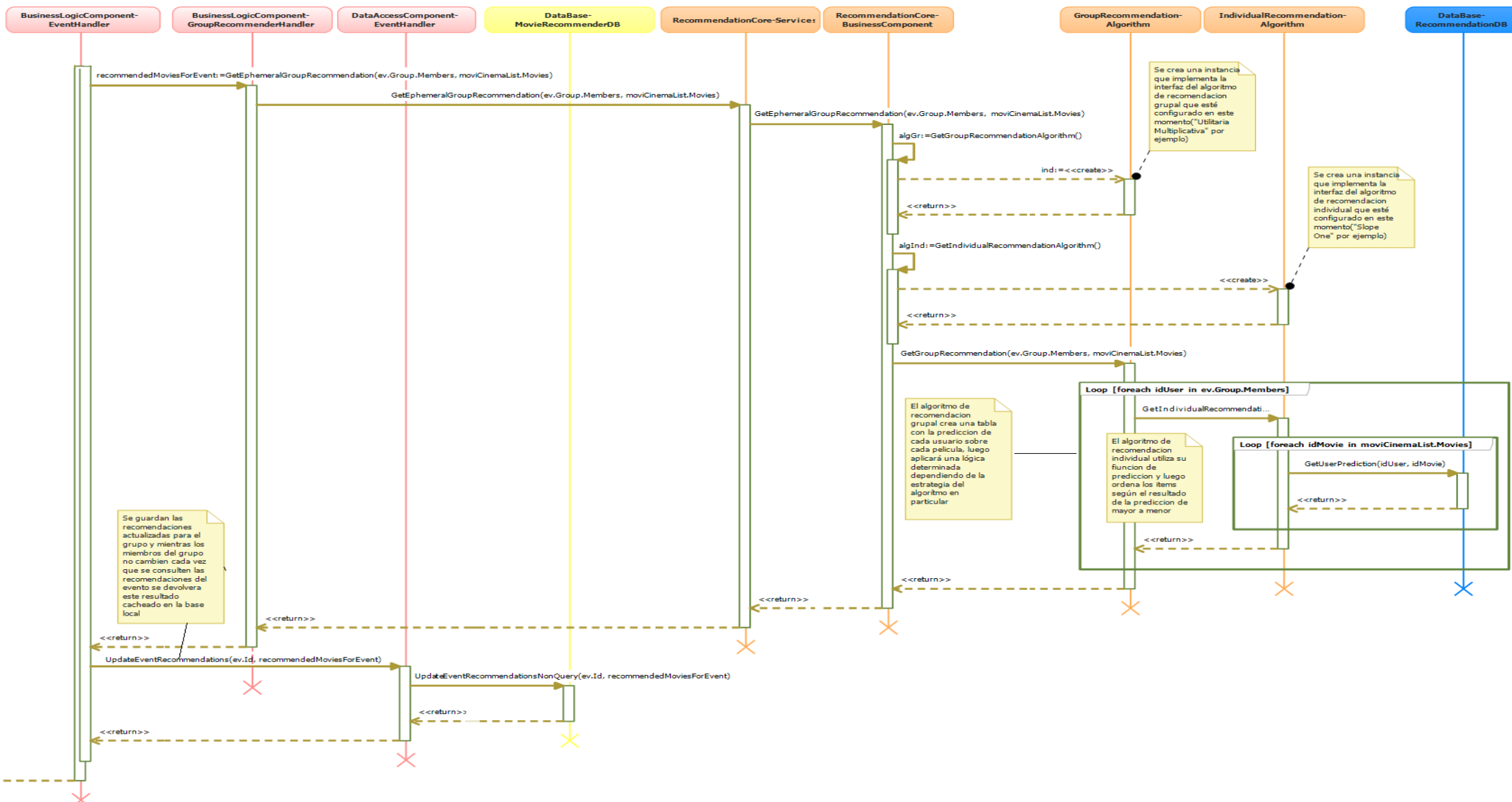


Ilustración 14: Diagrama de Secuencia - Aceptar Invitación (3 de 3)

3.5 Implementación

3.5.1 Carga y sincronización de datos

Cuando desarrollamos un sistema de recomendaciones y queremos ponerlo en práctica, inicialmente queremos generar recomendaciones de valor. Para lograrlo se presentan tres grandes problemas. En esta sección presentamos las soluciones que desarrollamos para estos problemas.

El primer problema que atacamos es el problema denominado *Ramp-Up*, este problema plantea cómo lograr que el sistema sea capaz de generar recomendaciones aceptables tempranamente. Como vimos al estudiar el algoritmo elegido para generar la recomendación individual, *Slope One*, se basa en encontrar usuarios que han calificado con el mismo criterio las películas ya vistas por el usuario a recomendar para luego ver la diferencia de gusto que han tenido contra la película a predecir. De aquí se desprende la importancia de contar con una base de datos que cuente con información pre-cargada de usuarios, un catálogo amplio y cómo los usuarios han calificado a los diferentes ítems de dicho catálogo.

El segundo problema relativo al catálogo y su metadata tiene que ver con encontrar mecanismos que nos permitan complementar y enriquecer el detalle de la metadata ofrecida para cada película, desde diferentes proveedores de catálogos, una vez importados los títulos y géneros desde un catálogo base.

Por último, el tercer problema a resolver consiste en dejar disponible en el sistema todas las estructuras de datos auxiliares y elementos requeridos para el algoritmo de recomendación.

3.5.1.1 Carga inicial desde MovieLens

Dada la necesidad de disponer de información pre-cargada de usuarios y sus calificaciones sobre un catálogo de películas, luego de haber estudiado el sitio MovieLens descrito en la sección 2.3, encontramos interesante el trabajo de investigación realizado por GroupLens. En este trabajo han recolectado datos de calificaciones de usuarios a películas para el dominio público, los usuarios han sido cambiados a usuarios anónimos por políticas de privacidad.

Consideramos que una carga inicial incorporando esta información a nuestro sistema resuelve parte importante del problema del *Ramp-Up*. Con esta carga no importa que nuestro sistema cuente con muy pocos usuarios reales y pocas calificaciones realizadas, podremos encontrar en estos usuarios “fantasmas” y en sus calificaciones, la base para construir el algoritmo de recomendación individual elegido: *Slope One*.

Para realizar esta carga inicial consideramos el juego de datos conocido como *MovieLens 10M* [17], que es el mayor de entre los juegos de datos disponibles de MovieLens y consta de

10.000.054 calificaciones aplicadas a 10.681 películas por 71.567 usuarios. Utilizar el juego de datos de mayor volumen nos brindará más opciones de recomendación y mejores valores de predicción de acuerdo a la definición del algoritmo Slope One. La información se descarga en tres archivos de texto plano: movies.dat, ratings.dat y tags.dat.

Movies.dat consiste del catálogo de películas en sí mismo, indicándose cada una en una línea con el formato **MovieID::Title::Genres** donde MovieID es el identificador de la película en MovieLens, Title corresponde con el título de la película y Genres consiste de la lista de géneros a los que aplica la película expresada como una lista separada por *pipes*. Los posibles géneros identificados son: Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, Western.

Ratings.dat es el archivo que contiene en una escala de 5 con incrementos de a 0.5 las calificaciones que los usuarios han realizado sobre los elementos del catálogo. Cada calificación se representa en una línea con el formato **UserID::MovieID::Rating::Timestamp**, donde UserID es un identificador único para reconocer al usuario en MovieLens, MovieID es el identificador de la película, Rating el valor con la calificación que ha otorgado el usuario a la película y Timestamp la fecha y hora en que ocurrió la calificación expresada en formato de fecha Unix.

Tags.dat consiste en los textos con los que los usuarios han etiquetado a las diferentes películas, pero no ha sido considerado para el volcado inicial de datos a nuestra aplicación.

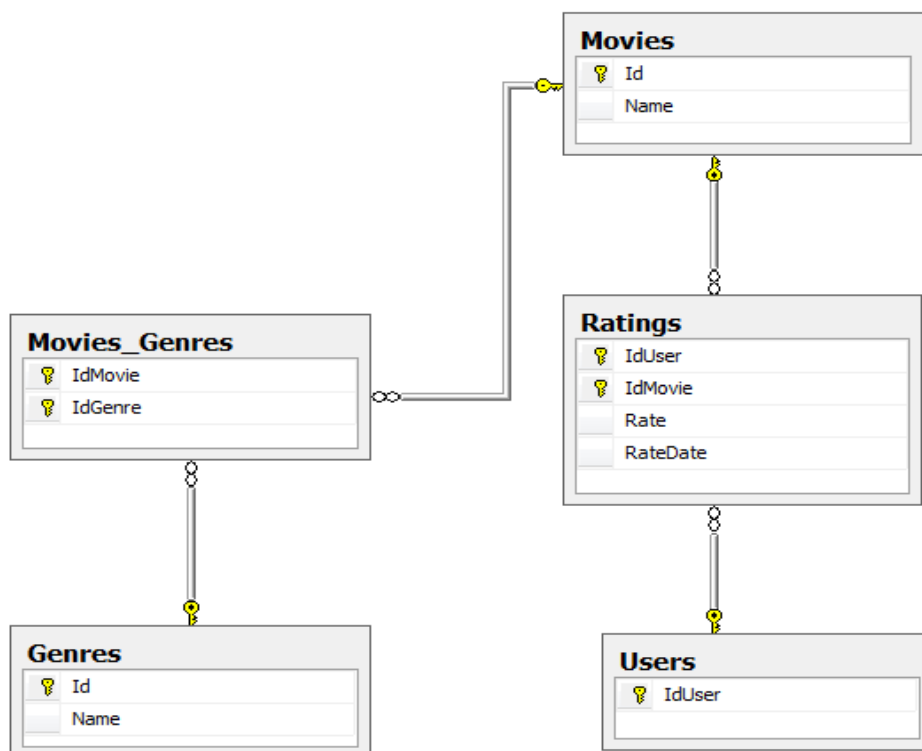


Ilustración 15: Modelo inicial para el volcado de los ficheros de MovieLens

Una vez descargados estos archivos y entendido su formato, desarrollamos un aplicativo utilitario capaz de cargarlos enteramente a memoria. Procesamos según la especificación y volcamos la información en una base de datos con restricciones de integridad, claves primarias y foráneas como se muestra en la ilustración 15. El orden de carga es: películas y géneros en una primera instancia (archivo “Movies.dat”) y luego usuarios y sus calificaciones (archivo “Ratings.dat”) en una segunda instancia.

3.5.1.2 Carga inicial y sincronización desde IMDb

El problema del catálogo y su metadata queda resuelto parcialmente con la carga inicial desde MovieLens, ya que podemos importar el catálogo de películas utilizado durante el trabajo de investigación de GrupoLens [77]. Observamos que esta información no es suficiente para lo que pretendemos mostrar a los usuarios en nuestro sistema. Es por ello que necesitamos contar con una carga inicial de toda la metadata asociada a estas películas desde fuentes externas. Dada esta necesidad y luego de estudiar en la sección 2.3 el sitio IMDb, entendemos que brinda interfaces para lograr complementar la información cargada con: sinopsis, actores, directores, escritores, etc.

Para realizar esta carga inicial se planifican las siguientes etapas:

- a) Carga de identificadores desde IMDb
- b) Carga de metadata desde IMDb
- c) Normalización de datos
- d) Carga de imágenes (posters) desde IMDb

3.5.1.2.1 Carga de identificadores desde IMDb

En esta primera etapa se busca identificar cada película de las importadas desde MovieLens entre las películas del catálogo de IMDb. Esta identificación es la que permitirá en las siguientes etapas realizar la carga de la metadata que necesitamos. Para lograr este objetivo, desarrollamos un aplicativo utilitario que busca películas ingresadas al sistema que no cuenten aún con un identificador sincronizado desde IMDb. Para ello, realiza una búsqueda por título sobre los contenidos de IMDb. Luego, si se encuentra una única coincidencia, guarda el identificador encontrado.

Lo expresado anteriormente resultaría sencillo si los títulos de las películas estuvieran expresados exactamente igual en ambos sistemas, lo cual como es de esperar en la realidad no sucede. Es por ello que se identifican dos transformaciones obligatorias a realizar a los títulos de MovieLens previo a la búsqueda en IMDb:

- a) Quitar el año de realización como parte del título de la película
- b) Identificar la ocurrencia de , A | , An | , The | , La | , Le |, Les al final del título, reubicarlos al comienzo y quitar la coma inicial.

En caso de que la búsqueda recupere más de un resultado, el identificador no es sincronizado automáticamente y lo que hacemos es guardar en una estructura temporal los “identificadores candidatos” encontrados. Este conflicto se resuelve con intervención humana. Por último, es importante notar que aunque este proceso se define para una carga inicial, también aplica para sincronizar identificadores de nuevas películas que se agreguen al sistema luego del primer volcado inicial.

3.5.1.2.2 Carga de metadata desde IMDb

Con el identificador de IMDb cargado en la etapa anterior, ahora somos capaces de obtener el detalle de cada película a partir de este identificador. Para lograr esto, extendemos las funcionalidades del aplicativo utilitario que usamos anteriormente para cargar los identificadores desde IMDb, dándole capacidad de cargar información detallada de cada película. Por cada película del sistema que no cuenta con una última fecha de actualización, el aplicativo utiliza la herramienta IMDbPY ⁵ para obtener y guardar el detalle de la metadata obtenida desde IMDb.

Una vez obtenida la metadata de las películas, el aplicativo utilitario modifica la última fecha de actualización de cada película, indicando de esta forma que estas películas ya han tenido su carga inicial de metadata. Si bien la carga de metadata se ha desarrollado para realizar un volcado inicial, se podría ejecutar el mismo proceso de forma periódica, una vez al día por ejemplo, pudiendo sincronizar la información de las nuevas películas que se hayan ingresado al sistema.

3.5.1.2.3 Normalización de datos

Una vez finalizada la carga de metadata desde IMDb, lo que logramos es obtener para cada película un conjunto de propiedades. Cada propiedad de la forma clave y valor, donde la clave se corresponde con algún atributo de la metadata de la película (título, géneros, actores, directores, idiomas, etc.) y el valor con una cadena de texto que da valor al atributo. En particular observamos que los atributos multivaluados se representan como cadenas de texto separados por comas.

⁵ <http://imdbpy.sourceforge.net/>

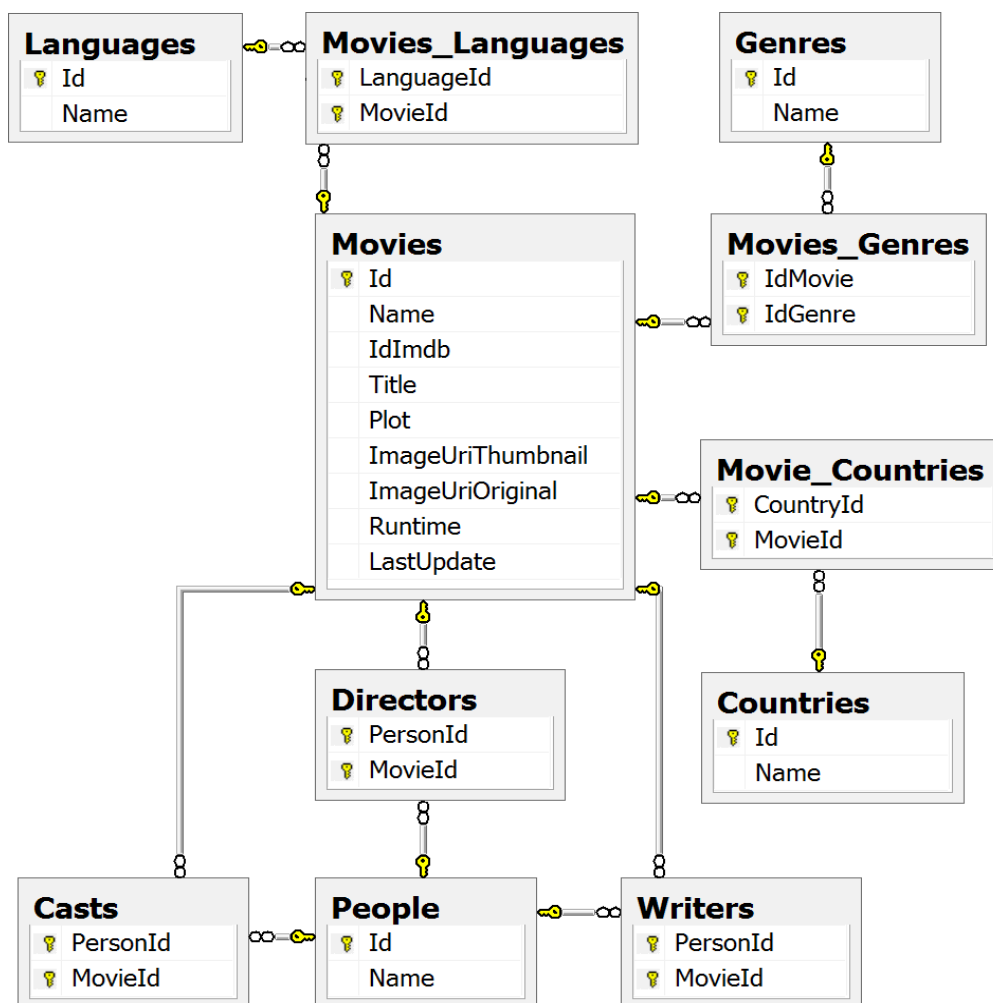


Ilustración 16: Modelo normalizado

Estos datos no se encuentran normalizados. Planteamos la normalización de la metadata de una película creando tablas específicas para nuevas entidades y nuevas tablas de relación entre estas y la tabla de películas. Esta normalización se puede apreciar en el modelo de la ilustración 16. Como muestra el modelo normalizado, se han creado nuevas tablas para las entidades *Genres*, *Countries*, *Languages* y *People* y además tablas de relación *Casts*, *Directors*, *Writers* (estas 3 primeras relacionando a las personas con las películas según el rol con que han participado en cada una), *Movies_Genres*, *Movies_Languages* y *Movies_Countries* (estas 3 últimas para indicar los géneros, idiomas y países respectivamente asociados a cada película).

Entre las ventajas del nuevo modelo destacamos que evita tener información repetida (el texto de los nombres de las personas que participaban en más de una película por ejemplo), elimina la necesidad de columnas vacías o con texto vacío, optimiza y mejora un eventual proceso de actualización o corrección de datos. Otra ventaja adicional es la simplificación y mejora de rendimiento de las consultas (evitando los cortes por columnas de tipo texto que no son clave primaria), características que aprovechamos para maximizar los tiempos de respuesta en algunas características del prototipo a construir como son los filtros de búsqueda avanzada de películas.

3.5.1.2.4 Carga de imágenes (posters) asociados a las películas

Analizando el sitio de IMDb podemos observar que se puede obtener el detalle de cada película accediendo a la página <http://akas.imdb.com/title/IMDBID>, donde IMDBID es el identificador de la película en IMDb. Dentro de este detalle se encuentra la URI del poster de la película. Además podemos especificar los valores de alto y ancho que queremos para la imagen. De esta forma, para cada película cargamos la URI del poster con el tamaño original y tamaño “miniatura”. Este proceso se ejecuta como una nueva opción que se agrega al aplicativo utilitario. Una vez ejecutado este proceso logramos cargar las URIs para los posters de todas las películas, con excepción de aquellas que o bien no pudimos reconocer su identificador asociado en IMDb, o bien porque la película no tiene un poster asociado en IMDb.

Ejecutando las pruebas sobre un ambiente de desarrollo esta solución no tiene inconvenientes, es decir no hay ningún problema en referenciar directamente desde nuestro sitio a las imágenes de IMDb. Sin embargo al pasar a un ambiente más real, realizando un despliegue en Internet usando un dominio público, observamos que las imágenes de las películas no se cargan en ningún caso. Observamos también que todas las peticiones HTTP a imágenes desde el sitio de IMDb responden con un código de acceso denegado, aunque si accedemos a esta dirección URL desde el navegador la imagen se muestra correctamente. Para solucionar este inconveniente planteamos la construcción de un servidor local de contenido estático, en particular para servir imágenes. Este servidor manipula el cabezal HTTP de la petición de imágenes, logrando solucionar el problema de la carga desde IMDb. Además, el servidor cumple la función de cache para las imágenes que ya fueron cargadas desde IMDb. Las imágenes se cargan a demanda desde IMDb quedando disponibles en el servidor en las futuras peticiones.

3.5.1.3 Carga inicial de datos para el algoritmo Slope One

Como mencionamos en el capítulo anterior, el algoritmo Slope One utiliza como insumo una matriz $A_{N \times N}$ donde cada entrada dev_{ij} representa la desviación de promedios de calificaciones entre dos películas y se calcula según la siguiente fórmula:

$$dev_{ij} = \frac{\sum_{u \in S_{ij}} (r_{ui} - r_{uj})}{|S_{ij}|},$$

donde S_{ij} denota el conjunto de usuarios que calificaron ambos ítems i y j , y $|S_{ij}|$ es el número de usuarios del conjunto.

El valor de dev_{ij} podemos entenderlo como la diferencia entre los promedios de gustos para los ítems i y j , por lo tanto, dev_{ji} será el valor opuesto a dev_{ij} : $dev_{ij} = -dev_{ji}$, entonces la matriz es asimétrica. Además, si comparamos un ítem consigo mismo (si i es igual a j), la diferencia de promedios de gustos es cero. Estas dos observaciones son importantes porque permitirá cargar solo la parte superior de la matriz, pues el resto de los valores son calculables. En la ilustración 17 vemos un ejemplo de la matriz de diferencias de promedios.

Para realizar el proceso de carga, construimos un aplicativo que itera sobre todas las películas, Por cada película, computa el valor de diferencia de promedios con el resto de las películas con

	Item 101	Item 102	Item 103	Item 104	Item 105	Item 106	Item 107
Item 101		-0.833	0.875	0.25	0.75	-0.5	2.5
Item 102			0.333	0.25	0.5	1.0	-
Item 103				0.167	1.5	1.5	-
Item 104					0.0	-0.25	1.0
Item 105						0.5	0.5
Item 106							-
Item 107							

Ilustración 17: Ejemplo: Diferencias de promedios de calificaciones en Slope One

Fuente: Mahout In Action – The algorithm [6]

un identificador mayor (el identificador mayor nos asegura solo considerar la parte superior de la matriz). El resultado se guarda en la celda a_{ij} de la matriz.

3.5.2 Proveedores de contenido

En esta sección explicamos el trabajo realizado para incorporar en nuestro sistema el uso de proveedores de contenido (video bajo demanda). Si bien en algunos proveedores se cuenta con diversidad de opciones como ser obtener videos de series, películas completas y/o clips, etc., por tratarse de un prototipo nos restringimos a recuperar información de avances de películas. Antes de abordar con más detalle cada proveedor incorporado en nuestra solución, debemos hablar del que resuelve la interacción con los proveedores. Este componente, denominado Administrador de Proveedores de Contenidos, permite incorporar de manera sencilla nuevos proveedores de contenidos, resuelve la sincronización y almacena los identificadores y/o direcciones al contenido de éstos vinculándolos con el ítem de nuestro sistema. Luego, el sistema cuenta con una vista de reproducción por cada proveedor soportado que considera sus particularidades de reproducción. Para esto, con cada proveedor incorporado, se debe implementar funcionalidades definidas que utiliza el sistema para lograr la reproducción.

3.5.2.1 IVA

Entre sus variados servicios, IVA [36] ofrece funcionalidades para obtener información sobre su contenido. Este servicio, denominado IVA OData API 1.0, permite acceder a todo el contenido de IVA. En nuestro caso nos interesa acceder a sus recursos avances de películas. Como su nombre lo indica, este servicio implementa OData, que es un protocolo estandarizado para definir y consumir APIs de datos. Este incorpora una arquitectura REST. Este servicio permite realizar consultas sobre el modelo de datos que ofrece IVA⁶. Se requiere una clave de desarrollador, versión gratuita en este caso. En el modelo de datos de IVA se encuentra el identificador IMDb. Con este identificador encontramos la película en nuestro sistema y le asociamos la URL de reproducción que nos brinda IVA.

⁶ Diagrama de Datos de IVA [http://www.internetvideoarchive.com/IVA/images/video-stills/entitydesignerdiagramexpanded-\(1\).jpg](http://www.internetvideoarchive.com/IVA/images/video-stills/entitydesignerdiagramexpanded-(1).jpg)

3.5.2.2 IMDb

IMDb cuenta con una base propia de avances de películas, detrás de escena, imágenes de su contenido multimedia, etc. A partir de la carga inicial de información de nuestro sistema, obtenemos las URLs desde donde se pueden reproducir los avances de películas desde IMDb. Agregamos estas direcciones a la lista de fuentes de reproducción de contenidos disponible para las películas en nuestro sistema. IMDb no facilita documentación sobre el uso de su reproductor de video. En el análisis del sitio que se realiza en la sección 2.3, observamos que las URLs de reproducción cumplen cierto formato en el que reciben como parámetro el identificador de una película. También observamos que no todas las películas tienen avances disponibles, en cuyo caso el intento de acceso a esta dirección devuelve una página HTML con el texto “*not have a valid trailer*”. Considerando estas observaciones, la solución propuesta consiste en un utilitario que acceda a la página de reproducción para cada película y procese la respuesta recibida. En caso de obtener una respuesta correcta que no contenga el texto “*not have a valid trailer*”, significa que existe un avance de película disponible para la película. En este caso guardamos una nueva fuente de contenido para la película.

3.5.2.3 Proveedores descartados

Como mencionamos en la sección 2.4, algunos proveedores popularmente conocidos, que en un principio se presentaban con una buena opción para utilizar en nuestro caso de estudio, han sido descartados. A modo informativo, en este punto mencionaremos el estado actual de los mismos en lo que concierne a este estudio. Crackle, si bien este sitio era uno de los grandes candidatos, el servicio API, desarrollada en su momento por Grouper Team, se encuentra cerrado desde 2008 y aún no ha vuelto a estar activo [57]. Redbox únicamente opera en Estados Unidos, por lo que hemos decidido no considerarlo. Por último, Netflix anunció que a principios del 2013 deja de emitir nuevas claves de desarrollador para sus API públicas [58], por lo cual no hemos podido incorporarlo a nuestra solución.

3.5.3 Problemas Técnicos

En el desarrollo del proyecto nos encontramos con algunos problemas técnicos. Vamos a describir los más relevantes y mencionaremos la solución que desarrollamos para cada uno:

1) **Carga de imágenes:** Como ya mencionamos, identificamos un problema para cargar imágenes desde IMDb para un despliegue en Internet usando un dominio público. En este caso, la respuesta al pedido HTTP es de acceso denegado. De ésta observación y puesto que en un ambiente de desarrollo usando un dominio local las imágenes se cargan correctamente, inferimos que el problema es con dominios públicos de Internet. En particular, con el encabezado `http-referer`, ya que si accedemos directamente sin un referer (por ejemplo, accediendo a la URL desde el navegador directamente), entonces el acceso es correcto.

Como solución a este problema, desarrollamos un servidor de imágenes que recibe las peticiones a través de una dirección que tiene el identificador de película como parámetro. Dado este identificador, podrá saber cuál es la URI del poster, con la URI podrá obtener dicha imagen directamente desde IMDb armando una petición HTTP de forma programática con las

características descriptas. Una vez obtenida la imagen desde IMDb, el servidor de imágenes la guarda tanto en el sistema de archivo como en memoria. De esta forma, logramos construir un servidor de imágenes con dos niveles de cache: un primer nivel en memoria y un segundo nivel en disco.

2) **Tamaño de base de datos:** Al momento de confeccionar el ambiente de desarrollo para nuestro sistema, optamos por utilizar SQL Server Express R2 2008 como base de datos. Elegimos esta opción por ser la última versión gratuita ofrecida por Microsoft y por su sencilla integración con el framework Microsoft Visual Studio, sobre el cual desarrollamos los módulos MovieRecommenderApp y SOABusinessServices. La versión de base de datos tiene limitado el tamaño, pudiendo alcanzar como máximo 4 Gigabytes. Este hecho sumado al tamaño de juego de datos utilizado trajo aparejadas dificultades técnicas y condicionantes durante la exploración de alternativas para mejorar los tiempos de respuesta del sistema. El principal impedimento surge a la hora de definir índices en la base de datos para mejorar los tiempos de acceso. El tamaño actual de la base de datos en combinación con el límite mencionado no permite que puedan definirse más índices útiles e incluso tampoco que puedan recrearse los ya existen.

4 Pruebas realizadas

4.1 Pruebas del sistema

Dentro de las pruebas que se realizaron al sistema, sólo vamos a mencionar la metodología utilizada para las pruebas funcionales y para las pruebas de dispositivos móviles, que son las que nos parecen más interesantes.

4.1.1 *Pruebas funcionales*

Para realizar las pruebas funcionales utilizamos el método de inferir casos de prueba a partir de los casos de uso, cubriendo todos los flujos al menos una vez. En el Anexo G: Casos de prueba, se encuentra el diseño de los casos de prueba. Además se especifica el ciclo de vida funcional de la entidad Evento, ya que la misma es la más relevante para nuestro prototipo. A continuación se describe el método elegido.

- Modelo de la realidad: Casos de uso
- Estrategia de selección: Los escenarios/condición de los casos de uso
- Criterio de cubrimiento: Se deben recorrer todos los flujos de los casos de uso en algún escenario. Además se cubren todos los escenarios optimizando la cantidad de casos de prueba. Luego, dentro de las variables operacionales identificadas en los escenarios / condición, definimos su dominio y clases de equivalencia. En la generación de casos de prueba, debemos cubrir todas las clases de equivalencia definidas.

Probamos 8 casos de uso, incluyendo todos los casos de uso que tienen interacción con el usuario final se cubrieron 30 escenarios, generando 33 casos de prueba. Además se ejecutó el ciclo funcional de la entidad Evento. Luego de realizar iteraciones en el desarrollo, todos los casos de prueba funcionan correctamente.

4.1.2 *Pruebas dispositivos móviles*

Probamos el comportamiento funcional de nuestro sistema en los diferentes dispositivos móviles, en cada dispositivo probamos los navegadores más populares. Para realizar estas pruebas definimos una variable de compatibilidad especificando tres categorías o posibles resultados de las pruebas. La prueba consiste en ejecutar las funcionalidades del sistema en el navegador del dispositivo móvil y verificar a qué categoría pertenece. A continuación definimos las categorías de compatibilidad y luego se muestran los resultados de las pruebas en la tabla 7

- **A:** Se pueden ejecutar las funcionalidades sin dificultad.
- **B:** Se pueden ejecutar las funcionalidades del sistema de alguna manera alternativa.
- **C:** No se pueden ejecutar las funcionalidades del sistema.

Modelo	S.O	Navegador	Categoría de compatibilidad
Emulador Windows Phone 7.5	Windows Phone 7.5	Internet Explorer 9	A
Tableta Kolke pc7134	Android 4.0.4	Chrome 24.0.1312	C
Tableta Kolke pc7134	Android 4.0.4	Firefox 20.0.1	A
Tableta Kolke pc7134	Android 4.0.4	Android Browser 4.0.4	B
Teléfono Inteligente Sony Ericsson x8	Android 2.3.7	Firefox 22.0	A
Teléfono Inteligente Sony Ericsson x8	Android 2.3.7	Android Browser 2.3.7	B
Teléfono Inteligente Sony Ericsson x8	Android 2.3.7	Opera Mobile 11.5	A

Tabla 7: Pruebas dispositivos

En ejecución de estas pruebas podemos verificar que la mayoría de los navegadores probados en los diferentes dispositivos están en la categoría A o B. En algunos casos encontramos que el navegador no permite ejecutar las funcionalidades del sistema, categoría C, como es el caso de Chrome versión 24. Una observación importante es que en todos los dispositivos encontramos algún navegador para el cual la prueba resultó como categoría A, logrando que el sistema funcione en todos los dispositivos probados. La otra observación es que a pesar de estar dentro del estándar HTML5, todavía no se cumplen sus especificaciones en todos los navegadores. Esto es esperable ya que HTML5 está en una etapa de borrador, donde si bien hay especificaciones están denominadas estables, existen otras especificaciones que aún están cambiando [59].

4.2 Pruebas de algoritmos de recomendaciones

Como vimos en la sección 3.3, el algoritmo de recomendación grupal utiliza al algoritmo individual. Nos planteamos como objetivo poder obtener una recomendación grupal de forma rápida con un resultado de recomendación que sea aceptable. Para lograrlo, necesitamos obtener la predicción individual con un rendimiento también aceptable. Para comprobar el cumplimiento de estos objetivos, definimos dos variables bajo estudio del algoritmo:

- Tiempo en resolver la recomendación (expresado en el formato hh:mi:ss)
- Rendimiento del algoritmo de predicción individual, utilizando la métrica MAE definidos en la sección 2.1.5, calculado para todos los usuarios y sus calificaciones.

Para estudiar estas variables ejecutamos el algoritmo para que recomiende 10 ítems para un grupo tipo de 7 usuarios. Estudiamos el resultado de estas variables para el algoritmo Slope One y para la adaptación que implementamos de este algoritmo. Además, observamos el tiempo de resolver la recomendación variando la cantidad de ítems recomendables. Vamos a ejecutar estas pruebas sobre el juego de datos *MovieLens 10M* [17]: 69.882 usuarios, 10.681 ítems, 10.000.058 calificaciones. A continuación se muestra la distribución de calificaciones por usuarios:

Cantidad de calificaciones	Cantidad de usuarios	Porcentaje
Entre 20 y 40	21055	30,14
Entre 41 y 80	17485	25,03
Entre 81 y 100	4836	6,92
Entre 101 y 120	4192	6,00
Entre 121 y 160	5325	7,62
Entre 161 y 200	3420	4,89
Entre 201 y 500	6695	9,58
Entre 501 y 750	2031	2,91
Entre 751 y 1000	791	1,13
Más de 1000	4038	5,78

Tabla 8: Densidad de usuarios por calificaciones del juego de datos del sistema

4.2.1 Recursos

Para ejecutar estas pruebas contamos con un Procesador Intel Core I7 1.6 Ghz; 4 Gb de memoria RAM; Sistema Operativo Windows 7 Home Premium y base de datos SQL Server Express 2008 R2.

4.2.2 Resultados

4.2.2.1 Slope One

Paso cero, consiste en ejecutar el algoritmo Slope One sobre todo el juego de datos. Para medir el tiempo tomamos una muestra proporcional de los usuarios respetando los porcentajes de distribución de calificaciones, el MAE se calculó sobre todo el juego de datos, a continuación se presentan los datos obtenidos.

- MAE: 0,69169
- Tiempo: 00:05:05

De los resultados obtenidos, podemos saber que el algoritmo refleja un error que es el esperado según los estudios mencionados en 2.1.5, incluso un menor error, lo que tiene sentido por usar un mayor volumen en el juego de datos estudiado. Cuando observamos el tiempo de realizar una recomendación para un grupo, vemos que no es un tiempo aceptable para un sistema interactivo.

4.2.2.2 Slope One adaptado

Como vimos en la sección 2.1.5, implementamos una adaptación al algoritmo Slope One con el objetivo de mejorar el tiempo de respuesta de recomendación. Para analizar los resultados de esta implementación definimos N , como variable de entrada, refleja el tamaño de $S'(u)$ (son los ítems N que calificó el usuario u y que a su vez tienen más calificaciones), analizamos los resultados variando N .

N	20	40	80	100
Tiempo	00:01:52	00:05:07	00:06:55	00:09:35
MAE	0,68568	0,68722	0,68984	0,69210
RMSE	0,89533	0,89615	0,89714	0,89799

Tabla 9: Implementación Segunda variante – Resultados

Los resultados reflejan que reduciendo la cantidad de calificaciones utilizadas en el algoritmo se logra mejorar el rendimiento. Como contrapartida, reducir la cantidad de calificaciones utilizadas aumenta la probabilidad de caer en el problema de Escasez de Datos, definido en la sección 2.1.6.6. En cuanto al tiempo, también confirmamos cómo disminuye el tiempo en función con el N tomado.

4.2.2.3 Cantidad de ítems recomendables

Como vimos en la prueba anterior, se puede obtener un mejor resultado de recomendación acotando la cantidad de calificaciones utilizadas en el algoritmo de predicción individual. De acuerdo a las pruebas realizadas, a partir de ahora fijamos este tope en $N = 20$. Entonces el orden de calcular la predicción de un usuario sobre un ítem es de 20 operaciones. Como se definió el algoritmo de recomendación grupal, si tenemos M ítems en el catálogo, el cálculo de recomendación para un grupo de U usuarios es de orden $O(M) \times O(U) \times 20$. Si limitamos M , obtendremos un menor tiempo de recomendación. Como contrapartida, las posibles recomendaciones estarán acotadas. A continuación mostramos el tiempo de recomendación tomando como variable M la cantidad de ítems para recomendar.

- Tiempo de recomendación con $M = 100$: 00:00:04
- Tiempo de recomendación con $M = 250$: 00:00:07
- Tiempo de recomendación con $M = 500$: 00:00:12
- Tiempo de recomendación con $M = 1000$: 00:00:23

4.2.3 Pruebas con voluntario

Para evaluar el algoritmo de recomendación implementado optamos por invitar a un grupo de personas voluntarias. Publicamos en Internet nuestro sistema de recomendación de películas y entregamos a cada voluntario un manual de utilización del sitio (ver Anexo H: Guía de uso del sitio MovieRecommender) y una planilla para completar. Una vez registrado como usuario, se solicitaba a la persona que califique veinte películas de su elección. Para lo cual, le pedimos que la mitad sean películas que sean de su agrado y las otras que no sean de su agrado. De esta manera el usuario genera la información de entrada que utiliza el algoritmo para sus posteriores recomendaciones. Luego, el voluntario completa la planilla ingresando títulos de películas que haya encontrado en el sistema junto con la calificación que le otorgaría. Esta prueba la ejecutaron cuatro voluntarios, con los cuales obtenemos los datos presentados en la tabla 7 para esta medición utilizamos la métrica MAE.

MAE
0,68106

Tabla 10: MAE de pruebas con voluntarios

Comparando con las mediciones realizadas en las pruebas anteriores, vemos que los valores obtenidos son muy similares y se encuentran dentro de rangos aceptables según los estudios mencionados. Si bien los voluntarios perciben que las predicciones son aceptables, no podemos tomar en cuenta estos datos por ser juego de datos bastante pequeño, que no permite sacar conclusiones. El objetivo de estas pruebas es presentar una idea de cómo podríamos probar el algoritmo de recomendación individual, con más recursos se podría constituir una muestra más representativa.

4.2.4 Conclusiones

Como vemos en las mediciones tomadas para la variable M (cantidad de ítems recomendables), con los recursos disponibles, podemos ofrecer recomendaciones en los tiempos establecidos. Dependiendo de las necesidades y características que tenga el cliente que utilice el servicio de recomendación, ajustará la variable M más adecuada a sus necesidades. Tengamos presente que cuanto mayor sea M , más películas es posible recomendar pero el tiempo de esta recomendación también crecerá. Por ejemplo, si es un sistema que está recomendando una película para ver en el momento, es más importante tener un buen tiempo de recomendación que disponer de todo el catálogo. Por el contrario, si es un sistema que envía las recomendaciones por correo electrónico semanalmente, no importa cuánto demora el proceso en calcular las recomendaciones, podría realizar los cálculos en un proceso *batch*. Con respecto al algoritmo Slope One, podemos concluir que se cumplen los siguientes objetivos que plantean los autores: sencilla implementación, el rendimiento del algoritmo es el esperado (MAE 0.68568), ingresar un nuevo ítem es instantáneo y que los usuarios con pocas recomendaciones reciben buenas recomendaciones. No podemos probar la eficiencia en cuanto al tiempo de cálculo del algoritmo, ya que no implementamos otros algoritmos para poder comparar sobre la misma infraestructura tecnológica. Dada la limitación de recursos para este proyecto, podemos pensar algunas líneas para probar pensando en futuros trabajos:

- Procesos concurrentes: Se puede particionar el cálculo por ítems, ejecutando un proceso para un conjunto de ítems, dividiendo el tiempo de recomendación en varios procesos.
- Cache de predicciones, se podría pensar en tener pre-calculada la predicción de los usuarios. Esto implicaría contar con la capacidad de almacenar una gran cantidad de datos.
- Particionar la matriz de diferencias de promedio agrupando ítems equivalentes, en el caso de las películas puede ser por género, por año de realizada, etc. De esta forma cuando se quiere predecir la calificación de un ítem al usuario, se ejecuta el algoritmo sobre la matriz que pertenece el ítem, la cual es de menor tamaño que la original. Manejar un volumen menor, es más eficiente para el manejador de base de datos.

5 Conclusiones y trabajos futuros

5.1 Conclusiones

El trabajo realizado en este proyecto se enfocó en el estudio de los sistemas de recomendaciones, tanto individuales como grupales, profundizando en el algoritmo de recomendación individual Slope One dentro de la técnica de filtrado colaborativo. También se estudió el aporte de los diferentes dispositivos para usuarios finales en este tipo de sistemas, logrando un prototipo donde se pudieran integrar.

Los algoritmos de recomendación individual se clasificaron dentro de las categorías: filtro colaborativo, basados en contenido, demográficos y basados en conocimiento. Se encontraron algunos problemas de performance a la hora de implementar el algoritmo Slope One para grandes volúmenes de datos. Se implementó una variante de este algoritmo reduciendo la cantidad de calificaciones de los usuarios, considerando solo las películas con mayor cantidad de calificaciones otorgadas. Se plantearon otras posibles mejoras de performance como podrían ser el uso de procesos concurrentes, cache de predicciones o partición de la matriz de diferencias en clases de equivalencias para los ítems.

Además se estudiaron algunos experimentos que comparan el rendimiento de algunos algoritmos de Filtro Colaborativo. Para esta comparación se especificaron las métricas utilizadas para medir su rendimiento. En cuanto a los algoritmos de recomendación grupal, se estudiaron las principales estrategias, profundizando en las variantes de estrategia utilitaria y estrategia de menos feliz. También se estudiaron las consideraciones para mantener la privacidad del grupo y cómo inciden las estrategias elegidas en la privacidad.

Como principal resultado del trabajo, se obtuvo el diseño e implementación de un prototipo para un sistema de recomendaciones grupal de películas y un servicio de recomendación genérico. El servicio de recomendación genérico es capaz de recomendar ítems independientemente de su naturaleza. Si bien este servicio fue diseñado para trabajar con distintos algoritmos de recomendación de filtro colaborativo, se implementó el algoritmo Slope One para recomendaciones individuales dadas las propiedades que definen sus autores. Para las recomendaciones grupales se siguió la estrategia utilitaria multiplicativa por ser la estrategia con más cuidado de la privacidad de los individuos. El prototipo de recomendaciones de películas consume el servicio de recomendación genérico. Este prototipo tiene una vista Web Móvil para recomendaciones grupales. De esta forma, cada individuo obtiene la recomendación para un grupo de amigos desde su dispositivo móvil. Desde esta vista móvil se pueden elegir películas dentro de la cartelera cinematográfica. Esta información está visible para todo el grupo de

amigos. Este prototipo fue implementado bajo el estándar HTML5 con el objetivo que funcionara en diferentes dispositivos móviles. También se desarrolló una vista Web que permitió probar las principales funcionalidades de recomendaciones e incorporar la reproducción de avances de películas desde diferentes fuentes proveedoras de video.

Para resolver el problema de *Ramp-Up* y disponer de los insumos para probar el algoritmo Slope One, importamos datos desde MovieLens: usuarios, calificaciones y películas que están disponible para el uso público. A su vez cruzamos estos datos con la información de las películas provista por IMDB, obteniendo una carga inicial de datos que hizo posible probar el prototipo implementado y los algoritmos de recomendación utilizados.

Sobre estos datos cargados, pudimos probar el rendimiento del algoritmo individual Slope One e implementar algunas variantes y compararlas. Otra variable bajo estudio fue el rendimiento del algoritmo implementado con los recursos que teníamos disponibles. Finalmente implementamos una variante del algoritmo Slope One, obteniendo el rendimiento esperado según los experimentos estudiados. Además generamos una tabla con los tiempos de respuesta del algoritmo, tomando como variable la cantidad de ítems que se consideran como entrada para la recomendación. En cuanto a los objetivos que plantean los autores del algoritmo Slope One encontramos que fue sencilla su implementación, que el rendimiento del algoritmo es el esperado, ingresar un nuevo ítem es instantáneo y que los usuarios con pocas recomendaciones, reciben buenas recomendaciones. No pudimos comparar su eficiencia en cuanto al tiempo de cálculo para la predicción, ya que no implementamos otros algoritmos sobre la misma infraestructura tecnológica que utilizamos para el algoritmo Slope One.

Además, hicimos el ejercicio de realizar calificaciones por cuatro usuarios externos. Con estas calificaciones calculamos el rendimiento del algoritmo y obtuvimos la percepción de los usuarios sobre el rendimiento de algoritmo. Decimos que esto fue un ejercicio porque los datos obtenidos no pueden ser objeto de estudio por su volumen, pero sí es un ensayo de cómo lo podríamos hacer si tuviéramos los recursos necesarios.

Encontramos algunas dificultades para probar el rendimiento de los algoritmos por contar con una base de datos acotada y con recursos limitados en cuanto a las capacidades de hardware en los equipos que teníamos disponible. Para probar el prototipo para dispositivos móviles utilizamos los emuladores que estaban disponibles por los proveedores, lo cual no es una prueba real. Además probamos el prototipo en algunos dispositivos reales pero no pudimos obtener la variedad deseada. En estas pruebas, notamos que la aplicación funciona como se espera en algunos casos; en otros tuvimos que realizar algunos ajustes y en otros no logramos compatibilidad. Con las pruebas realizadas vemos que la opción de usar HTML5 como interfaz

para dispositivos móviles es factible pero que aún no está consolidada en el universo de dispositivos móviles.

En resumen, se realizó un caso de estudio de recomendaciones grupales de películas, el cual utiliza la estrategia multiplicativa para recomendaciones grupales y una variante del algoritmo Slope One para recomendaciones individuales, obteniendo algunos datos del rendimiento del algoritmo Slope One que confirman los experimentos estudiados. Dentro del dominio de películas que manejamos en el sistema, logramos cargar los datos de forma masiva e integramos la reproducción desde distintas fuentes externas. Exploramos el camino de una vista para dispositivos móviles, nos enfrentamos a algunos desafíos que presenta la variedad y la compatibilidad entre los navegadores disponibles para dispositivos de diferentes características. Podemos concluir que de manera global se han alcanzado los objetivos que nos planteamos en este proyecto.

Finalmente, luego de terminar con el proyecto logramos construir una opinión crítica sobre los sistemas de recomendaciones, tanto individual como grupal. Los sistemas de recomendaciones individuales están maduros, encontramos abundantes experimentos y estudios sobre distintos algoritmos con resultados que son constatables. Por el contrario, en los sistemas de recomendaciones grupales no encontramos demasiados casos prácticos en la actualidad. Pensamos que la masificación de las redes sociales pueda ser un disparador para que este campo sea más explorado en el futuro cercano.

5.2 Trabajos futuros

En el desarrollo de este proyecto se identificaron algunos trabajos que se podrían realizar en el futuro, además quedaron planteados problemas con posibles soluciones. A continuación se describen los que consideramos más importantes.

Algoritmos de recomendación: En este proyecto nos enfocamos en el algoritmo de recomendación individual Slope One, probando algunas variantes. Este algoritmo fue categorizado dentro los SRFC, obteniendo los resultados presentados. Los algoritmos de SRFC tienen como propiedad que reciben como entrada la calificación de los usuarios sobre los ítems. Sobre esta base se podría trabajar en implementar distintos algoritmos de SRFC como basados en el usuario o basados en el ítem con los datos que ya están cargados. Luego de implementarlos, se podrían comparar estos algoritmos en las variables de rendimiento (expresada en MAE) y performance. Por otro lado, cuando estudiamos los algoritmos de recomendación individual se describieron algunos problemas que presentan, en particular los problemas de los algoritmos de SRFC: Ramp-Up, Ovejas Gris, Escasez de datos. En nuestro proyecto atacamos el problema de Ramp-up y Escasez de datos, cargando las calificaciones y la matriz de diferencias con los datos de MovieLens. Pensamos que sería interesante atacar el problema de Oveja Gris, donde se

pueden identificar y agrupar los usuarios de comportamiento diferente para aplicarles el algoritmo de recomendación individual.

También se puede profundizar en el estudio del algoritmo Slope One. Por ejemplo, se podría probar una variante donde se utilice la historia del error de predicción del usuario como un término en la fórmula de predicción. Otra variante a recorrer puede ser agrupar los usuarios según su similitud en las calificaciones y aplicar el algoritmo Slope One por grupo de usuarios. También se pueden agrupar los ítems por características y aplicar el algoritmo sobre cada grupo de ítems.

En cuanto a las recomendaciones grupales sería interesante profundizar en las técnicas para mejorar la privacidad del grupo. Una mejora puede ser soportar diferentes perfiles de usuarios para un mismo individuo. Esto le permitiría al individuo cambiar de opinión sobre un determinado elemento dependiendo en el contexto y grupo en el que se encuentre. También se podrían pensar algunas estrategias para evaluar los algoritmos de recomendación grupal, por ejemplo, analizar si el grupo fue al cine a ver la película que le recomendó el sistema.

Mejorar la performace: Con los recursos disponibles en este proyecto llegamos al rendimiento presentado. Creemos que el algoritmo Slope One se puede dividir en procesos paralelos. Se puede ejecutar un proceso por cada ítem a comparar. Para hacer esta prueba se necesita tener recursos de hardware que lo permitan, ya sea a través de varios equipos, donde se requiere una granja de servidores de base de datos sincronizados, o un servidor con capacidad de varios procesadores y memoria suficiente. También se podrían usar procesadores vectoriales como GPU. Con estos recursos se podría probar cuánto puede mejorar el rendimiento del algoritmo.

Finalmente, en el estado del arte estudiamos los conceptos de primera y segunda pantalla. Si bien nuestra solución cuenta con un componente para dispositivos móviles, no aplicamos estos conceptos por el alcance determinado. Podríamos pensar en incluir estos conceptos en la solución obtenida, utilizando diferentes clientes Webs contruidos bajo el estándar HTML5 para dispositivos de segunda pantalla. Consideramos interesante probar si en la actualidad es posible construir una solución así, bajo el estándar HTML5.

6 Glosario

En los siguientes puntos se definen conceptos y términos que son de interés remarcar en el contexto de este proyecto.

SR: Sistema de Recomendaciones.

SRFC: Sistema de Recomendaciones Filtrado Colaborativos. Son sistemas de recomendación que realizan la recomendación para un usuario basándose en las calificaciones otorgadas por otros usuarios de intereses similares.

SRBC: Sistema de Recomendaciones Basados en Contenido. Son sistemas que realizan la recomendación a un usuario según su perfil de preferencias que generan a partir de las características de los ítems que el usuario ha calificado.

SRCO: Sistema de Recomendaciones Basados en Conocimiento (Knowlegde). Son sistemas que utilizan el conocimiento acerca del usuario y los ítems para generar una recomendación, infiriendo acerca de cuáles son los ítems que se encuentran entre los requerimientos del usuario.

Dispositivo Móvil: Computadora pequeña con portabilidad que tiene capacidad de cómputo. Ejemplos de dispositivo móvil son: Teléfonos Inteligentes y Tablet.

HTML5: Versión 5 del estandar HTML en etapa de borrador que incorpora características como: Cache Local (WebStorage) y soporte para navegar fuera de línea, API de Geolocalización, Canvas para dibujar, reproducción de video y microdata.

JSON: JavaScript Object Notation: Formato ligero utilizado para el intercambio de datos, permite representar objetos de JavaScript utilizando una notación liviana.

MAE: Error Absoulto Medio (Mean Absolute Error). Medida utilizada para calcular el promedio de error absoluto entre un valor real y la predicción de este valor.

MVC: Modelo Vista Controlador (Model View Controller): Patrón de diseño de arquitectura de software que se separa las responsabilidades de los componentes en tres: Presentación (Vista); Lógica de negocio (Controlador) y Datos (Modelo).

SOA: Arquitectura Orientada a Servicios (Service Oriented Architecture). Concepto de arquitectura de software que permite exponer como servicios las funcionalidades del negocio.

Bibliografía

- [1] E. Vozalis y K. G. Margaritis, «Analysis of Recommender Systems' Algorithms», *Parallel and Distributed Processing Laboratory, Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece*, 2003.
- [2] L. Candiller, J. Kris, F. Fessant, y F. Meyer, «State-of-the-Art Recommender Systems», en *Collaborative and Social Information Retrieval and Access: Techniques for Improved User Modeling*, Orange Labs Lannion, France, 2009.
- [3] J. S. Breese, D. Heckerman, y C. Kadie, «Empirical Analysis of Predictive Algorithms for Collaborative Filtering», *Microsoft Research, Redmond, WA 98052-6399*, 1998.
- [4] S. Gong, «A Collaborative Filtering Recommendation Algorithm, Based on User Clustering and Item», *Zhejiang Business Technology Institute, China*, vol. 5, n.º 7, jul. 2010.
- [5] B. M. Sarwar, G. Karypis, J. A. Konstan, y J. Riedl, «Item-Based Collaborative Filtering Recommendation Algorithms», *Department of Computer Science and Engineering, University of Minnesota*, may 2001.
- [6] S. Owen, R. Anil, T. Dunning, y E. Friedman, *Mahout In Action*. Manning Publications Company, 2012.
- [7] G. Karypis, «Evaluation of Item-Based Top-N Recommendation Algorithm», *Department of Computer Science & Engineering at the University of Minnesota*, 2001.
- [8] D. Lemire y A. Maclachlan, «Slope One Predictors for Online Rating-Based Collaborative Filtering», *SDM'05*, feb. 2005.
- [9] R. Bruke, «Knowledge-based recommender systems», *Encyclopedia of Library and Information Systems*, 2000.
- [10] M. Pazzani y D. Billsus, «Content-Based Recommendation Systems», *Rutgers University, ASBIII, 3 Rutgers Plaza New Brunswick, NJ 08901*, 2007.
- [11] P. Massa y B. Bhattacharjee, «Using Trust in Recommender Systems: an Experimental Analysis», *International Graduate School in Information and Communication Technologies University of Trento, 38050 Povo (TN), Italy*, 2004.
- [12] H. Karaman, «A Content-based Movie Recommendation System Empowered by Collaborative Missing Data Prediction», *A thesis submitted to the graduate School of Natural and Applied Sciences of Middle East Technical University*, 2010.
- [13] M. Pazzani y D. Billsus, «Learning and Revising User Profiles: The Identification of Interesting Web Sites», *Department of Information and Computer Science, University of California, Irvine, Irvine, CA 92697*, 1997.
- [14] B. Krulwich, «LIFESTYLE FINDER: Intelligent User Profiling Using Large-Scale Demographic Data», *AI Magazine AAAI*, vol. 18, n.º 2, 1997.
- [15] R. Bruke, «Hybrid Recommender Systems: Survey and Experiments», *California State University, Fullerton Department of Information Systems and Decision Sciences*, 2002.
- [16] P. Marschik y A. Rauber, «Evaluation of Collaborative Filtering Algorithms», *Vienna University of Technology, Vienna, Austria*, 2010.
- [17] «MovieLens 10M/100k Data Set README», *MovieLens 10M/100k Data Set README*, abr-2012. [En línea]. Disponible en: <http://www.grouplens.org/system/files/ml-10m-README.html>.
- [18] «Libimseti.cz - charlar, ligar, relación seria», *Libimseti.cz*. [En línea]. Disponible en: <http://libimseti.cz/>. [Accedido: 10-jun-2012].

- [19] V. Petricek, «Occam's Lab - Ratings Data Recopilation», *Occam's Lab*. [En línea]. Disponible en: <http://www.occamlab.com/petricek/data/libimseticomplete.zip>. [Accedido: 22-may-2012].
- [20] R. Bruke, «Integrating Knowledge-based and Collaborative-filtering Recommender Systems.», *AAAI Technical Report WS-99-01*, 1999.
- [21] P. Resnick y H. R. Varian, «Guest Editors' Introduction: Recommender Systems.», *IEEE*, vol. 22, n.º 3, pp. 18-21, may 2007.
- [22] B. M. Sarwar, J. A. Konstan, y J. Herlocker, «Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System», *GroupLens Research Project, Dept. of Computer Science and Engineering University of Minnesota Minneapolis, MN 55455*, 1998.
- [23] G. Adomavicius y A. Tuzhilin, «Towards the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions», *IEEE Transactions on Knowledge and Data Engineering*, 2005.
- [24] J. Masthoff, «Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers», *Kluwer Academic Publishers*, pp. 37-85, oct. 2002.
- [25] M. O'Connor, D. Cosley, J. A. Konstan, y J. Riedl, «PolyLens: a recommender system for groups of users», en *Proceedings of the seventh conference on European Conference on Computer Supported Cooperative Work*, Norwell, MA, USA, 2001, pp. 199-218.
- [26] B. H. C. Cheng, R. de Lemos, H. Giese, P. Inverardi, y J. Magee, «Software Engineering for Self-Adaptive Systems», *Springer-Verlag Berlin Heidelberg*, 2009.
- [27] J. F. McCarthy y T. D. Anagnost, «MUSICFX: An Arbiter of Group Preferences for Computer Supported Collaborative Workouts», 1998.
- [28] A. E. Patrick Barwise, *Television and Its Audience*. SAGE Publications, 1998.
- [29] N. Kasari, «TV audience segments based on viewing behaviour», presentado en Worldwide Broadcast Audience Research Symposium, Toronto, Canadá, 1998.
- [30] M. B. Sonia Livingstone, «Young people, new media», 1999.
- [31] Neil W. Van Dyke Henry Lieberman y Adrian S. Vivacqua, «Let's browse: A collaborative web browsing agent», presentado en IUI '99 Proceedings of the 4th international conference on Intelligent user interfaces, Los ángeles, 1998.
- [32] «Luxand,Blink - Face recognition Software». [En línea]. Disponible en: <http://www.luxand.com/blink>. [Accedido: 21-may-2012].
- [33] Judith Masthoff y Albert Gatt, «In Pursuit of Satisfaction and the Prevention of Embarrassment: Affective State in Group Recommender Systems», *University of Aberdeen*, 2006.
- [34] «Alexa the Web Information Company», *Alexa the Web Information Company*, 03-ene-2012. [En línea]. Disponible en: <http://www.alexa.com>. [Accedido: 22-abr-2012].
- [35] «Compete», *Compete Site Analytics*. [En línea]. Disponible en: <http://www.compete.com>. [Accedido: 22-abr-2012].
- [36] «Internet Video Archive, The world's largest distributor of online movie trailers». [En línea]. Disponible en: <http://www.internetvideoarchive.com/>. [Accedido: 10-may-2012].
- [37] «Netflix - Watch TV Shows Online, Watch Movies Online», *Netflix*. [En línea]. Disponible en: <http://www.netflix.com>. [Accedido: 23-abr-2012].
- [38] C. Kaiser, «HTML5 and Video Streaming», *The Netflix Tech Blog*, 22-dic-2010. .

- [39] «Crackle - Ver Películas Online Gratis, Series de Televisión y Originales 24-7», *Crackle.com*. [En línea]. Disponible en: <http://www.crackle.com.uy>. [Accedido: 23-abr-2012].
- [40] «Redbox», *Redbox.com*. [En línea]. Disponible en: <http://www.redbox.com>. [Accedido: 12-jun-2012].
- [41] S. Tyagi, «RESTful Web Services», *ORACLE*, ago-2006. [En línea]. Disponible en: <http://www.oracle.com/technetwork/articles/javase/index-137171.html>. [Accedido: 12-jun-2012].
- [42] G. Faber, «Shifting Power: How Google TV entered the living room», University of Amsterdam, Amsterdam, 2011.
- [43] A. Breznic, «MultiScreen Gateways: Paving the way for IP video», p. 14, nov-2011.
- [44] B. D. Jhonson, «4 Predictions for Connected Devices in 2011», *Mashable*, de Diciembre de-2010. [En línea]. Disponible en: <http://mashable.com/2010/12/23/predictions-connected-devices/>. [Accedido: 18-jul-2013].
- [45] «HBO GO», *HBO GO. Es HBO. En cualquier lugar*. [En línea]. Disponible en: <http://www.hbogola.com/#whatis/>. [Accedido: 18-jul-2013].
- [46] «Moviestory Play - Que es Moviestory». [En línea]. Disponible en: <http://www.moviestoryplay.com/que-es/>. [Accedido: 18-jul-2013].
- [47] «ESPN Play», *ESPN Play*. [En línea]. Disponible en: <http://espni.go.com/player/welcome/latam/index.htm>. [Accedido: 18-jul-2013].
- [48] «Everything You Need to Know About TV Everywhere», *GIGAOM*, 23-jun-2009. [En línea]. Disponible en: <http://gigaom.com/2009/06/23/what-you-need-to-know-about-tv-everywhere/>. [Accedido: 18-jul-2013].
- [49] «First screen first!», *CNET*, 02-mar-2012. [En línea]. Disponible en: http://reviews.cnet.com/8301-19727_7-57389421-10170017/first-screen-first/. [Accedido: 18-jul-2013].
- [50] A. Grundner, «What Is and Isn't a Smart TV», *eHomeUpgrade*, 14-jun-2011. [En línea]. Disponible en: <http://www.ehomeupgrade.com/2011/01/14/what-is-and-isnt-a-smart-tv/>. [Accedido: 18-jul-2013].
- [51] «Smart TV», *Smart TV - Wikipedia, the free encyclopedia*. [En línea]. Disponible en: http://en.wikipedia.org/wiki/Smart_TV. [Accedido: 02-jul-2012].
- [52] «Second screen», *Second screen - Wikipedia, the free encyclopedia*. [En línea]. Disponible en: http://en.wikipedia.org/wiki/Second_screen. [Accedido: 19-jul-2013].
- [53] C. Warren, «How Connected Devices and Streaming Video Are Changing the Way We Consume Content», *Mashable*, de Agosto de-2011. [En línea]. Disponible en: <http://mashable.com/2011/08/18/connected-devices-faq/>. [Accedido: 17-jul-2013].
- [54] «Sony Developer Sites», *Android Developer Program*. [En línea]. Disponible en: <http://android.developer.sony.com/>. [Accedido: 09-jul-2012].
- [55] «Tablet computer», *tablet computer Definition from PC Magazine Encyclopedia*. [En línea]. Disponible en: <http://www.pcmag.com/encyclopedia/term/52520/tablet-computer>. [Accedido: 19-jul-2013].
- [56] «Smartphone», *Smartphone - Wikipedia, the free encyclopedia*. [En línea]. Disponible en: <http://en.wikipedia.org/wiki/Smartphone>. [Accedido: 19-jul-2013].
- [57] «Grouper Video API no longer available notification», *ProgrammableWeb*. [En línea]. Disponible en: <http://www.programmableweb.com/api/grouper-video>. [Accedido: 10-may-2012].

- [58] D. Jacobson, «Changes to the Public API Program», *Netflix API Developer Blog*, 08-mar-2013. .
- [59] «HTML5», *HTML 5.1 Nightly*. [En línea]. Disponible en: <http://www.w3.org/html/wg/drafts/html/master/>.
- [60] «Alexa - IMDb.com site information and ranking», *Alexa the Web Information Company*. [En línea]. Disponible en: <http://www.alexac.com/siteinfo/imdb.com>. [Accedido: 22-abr-2012].
- [61] «Compete - IMDb.com site information and ranking», *Compete Site Analytics*. [En línea]. Disponible en: <https://siteanalytics.compete.com/imdb.com/>. [Accedido: 22-abr-2012].
- [62] «Internet Movie Database». [En línea]. Disponible en: <http://www.imdb.com>. [Accedido: 22-abr-2012].
- [63] «Internet Archive», *Internet Archive: Digital Library of Free Books, Movies, Music & Wayback Machine*. [En línea]. Disponible en: <http://archive.org>. [Accedido: 22-abr-2012].
- [64] «Amazon.com: Amazon Instant Video», *Amazon Instant Video*. [En línea]. Disponible en: <http://www.amazon.com/Instant-Video>. [Accedido: 22-abr-2012].
- [65] «IMDB FTP», *IMDB FTP*. [En línea]. Disponible en: <ftp://ftp.fu-berlin.de/pub/misc/movies/database/>. [Accedido: 22-abr-2012].
- [66] B. Fritz, «The IMDb API v2.0». [En línea]. Disponible en: <http://www.imdbapi.com>. [Accedido: 22-abr-2012].
- [67] «IMDB API», *IMDB API, IMDB web service*. [En línea]. Disponible en: <http://www.deanclatworthy.com/imdb/>. [Accedido: 22-abr-2012].
- [68] S. Korokithakis, «IMDB API», *IMDB API*. [En línea]. Disponible en: <http://imdbapi.poromenos.org>. [Accedido: 22-abr-2012].
- [69] J. Greenstein, «Flixster Recommendations!!», *Flixster.com*, 2007. [En línea]. Disponible en: <http://community.flixster.com/forums/topic/266961230>. [Accedido: 22-abr-2012].
- [70] «Alexa - rottentomatoes.com site information and ranking», *Alexa the Web Information Company*. [En línea]. Disponible en: <http://www.alexac.com/siteinfo/rottentomatoes.com>. [Accedido: 22-abr-2012].
- [71] «Compete - rottentomatoes.com site information and ranking», *Compete Site Analytics*. [En línea]. Disponible en: <https://siteanalytics.compete.com/rottentomatoes.com/>. [Accedido: 22-abr-2012].
- [72] Alan Henry, «Five Best Movie Recommendation Services | Lifehacker Australia», 13-feb-2012. [En línea]. Disponible en: <http://www.lifehacker.com.au/2012/02/five-best-movie-recommendation-services/>. [Accedido: 22-abr-2012].
- [73] «Rotten Tomatoes», *Movies | Movie Trailers | Reviews - Rotten Tomatoes*. [En línea]. Disponible en: <http://www.rottentomatoes.com>. [Accedido: 22-abr-2012].
- [74] «UltraViolet Your Movies In The Cloud», *UltraViolet*. [En línea]. Disponible en: <http://www.uvu.com/>. [Accedido: 22-abr-2012].
- [75] «Rotten Tomatoes API», *Rotten Tomatoes Developers*. [En línea]. Disponible en: <http://developer.rottentomatoes.com>. [Accedido: 22-abr-2012].
- [76] «Universidad de Minnesota», *Movielens - movie recommendations*. [En línea]. Disponible en: <http://movielens.umn.edu/login>. [Accedido: 22-abr-2012].
- [77] Universidad de Minnesota, «GroupLens Research, University of Minnesota». [En línea]. Disponible en: <http://www.grouplens.org/>. [Accedido: 22-abr-2012].

- [78] «Hulu», *Watch TV. Watch Movies. | Online | Free | Hulu*. [En línea]. Disponible en: <http://www.hulu.com>. [Accedido: 22-abr-2012].
- [79] J. Kilar CEO, «2011, 2012 and beyond», *Hulu Blog*, 12-ene-2012. .
- [80] «Alexa - Hulu.com site information and ranking», *Alexa the Web Information Company*. [En línea]. Disponible en: <http://www.alexa.com/siteinfo/hulu.com>. [Accedido: 22-abr-2012].
- [81] «Compete - Hulu.com site information and ranking», *Compete Site Analytics*. [En línea]. Disponible en: <https://siteanalytics.compete.com/hulu.com/>. [Accedido: 22-abr-2012].
- [82] «oEmbed», *oEmbed*. [En línea]. Disponible en: <http://www.oembed.com/>. [Accedido: 22-abr-2012].
- [83] «Nanocrowd - The movie search engine», *Nanocrowd*. [En línea]. Disponible en: <http://www.nanocrowd.com/>. [Accedido: 23-abr-2012].
- [84] B. Deitz, «Netflix Tech Tips: Finding the Right Movie for That Right Moment», *Switched.com*, 14-ene-2011. .
- [85] Juan Diego Polo, «Cómo descubrir si te va a gustar una película», *www.whatsnew*. [En línea]. Disponible en: <http://www.whatsnew.com/2010/02/25/como-descubrir-si-te-va-a-gustar-una-pelicula/>. [Accedido: 23-abr-2012].
- [86] J. D. Polo, «Nanocrowd – Inteligente sistema de recomendaciones de películas», *www.what's new*, 06-abr-2009. .
- [87] «Nanocrowd Movie Information Search Engine», *appappeal*, 06-abr-2009. [En línea]. Disponible en: <http://www.appappeal.com/app/nanocrowd/>. [Accedido: 23-abr-2012].
- [88] «Find movies, TV shows matching your taste & watch online», *Jinni*. [En línea]. Disponible en: <http://www.jinni.com/>. [Accedido: 23-abr-2012].
- [89] Don Reisinger, «Top 10 movie recommendation engines | Webware», *CNET*, 29-mar-2009. [En línea]. Disponible en: http://news.cnet.com/8301-17939_109-10200031-2.html. [Accedido: 23-abr-2012].
- [90] Robin Wauters, «Jinni Is Building A Smart "Taste Engine" For Google TV (Screenshots)», *TechCrunch*, 21-may-2010. [En línea]. Disponible en: <http://techcrunch.com/2010/05/21/jinni-google-tv/>. [Accedido: 23-abr-2012].
- [91] Todd Spangler, «Jinni's Movie-Mood Engine Impresses At CableLabs Winter Conference - 2010-02-09 02:13:28», *Multichannel News*, 08-feb-2010. [En línea]. Disponible en: http://www.multichannel.com/article/448356-Jinni_s_Movie_Mood_Engine_Impresses_At_CableLabs_Winter_Conference.php. [Accedido: 23-abr-2012].
- [92] Richard MacManus, «Jinni: Wants to be Pandora for Movies», *Readwriteweb*, 22-ene-2009. [En línea]. Disponible en: http://www.readwriteweb.com/archives/jinni_pandora_for_movies.php. [Accedido: 23-abr-2012].
- [93] «LoveFilm an Amazon Company», *LoveFilm.com*. [En línea]. Disponible en: <http://www.lovefilm.com/>. [Accedido: 01-may-2012].
- [94] «Blockbuster: Movies | New Releases | DVD Tentat | Movie Trailers | Movie Reviews», *Blockbuster.com*. [En línea]. Disponible en: <http://www.blockbuster.com/>. [Accedido: 01-may-2012].
- [95] «Movie Reviews, TV Reviews, Game Reviews, and Music Reviews», *Metacritic*. [En línea]. Disponible en: <http://www.metacritic.com/>. [Accedido: 23-abr-2012].

- [96] «YouTube, Broadcast Yourself». [En línea]. Disponible en: <http://www.youtube.com/>. [Accedido: 10-may-2012].
- [97] «Open Data Protocol (OData)». [En línea]. Disponible en: <http://www.odata.org>. [Accedido: 11-may-2012].
- [98] «MCEBrowser», *Anthony Park :: Software*. [En línea]. Disponible en: <http://www.anpark.com/Software.aspx>. [Accedido: 14-jul-2012].
- [99] «Introducing Xbox SmartGlass», *Xbox SmartGlass | Companion Application - Xbox.com*, 04-jun-2012. [En línea]. Disponible en: <http://www.xbox.com/en-US/smartglass>. [Accedido: 14-jul-2012].
- [100] «Xbox 360 Registered Developers Program», *Xbox 360 Registered Developers Program - Xbox.com*. [En línea]. Disponible en: <http://www.xbox.com/en-US/developers/xbox360/registereddevelopers>. [Accedido: 14-jul-2012].
- [101] «Developing apps for Windows Phone or Xbox?», *Windows Phone Dev Center*. [En línea]. Disponible en: <http://create.msdn.com/en-us/home/membership>. [Accedido: 14-jul-2012].
- [102] «To become an Authorized Developer for Nintendo game platforms», *Nintendo Software Development Support Group - Apply*. [En línea]. Disponible en: <https://www.warioworld.com/apply/devapplication.asp>. [Accedido: 17-jul-2012].
- [103] «Roku - The Eclipse BrightScript Plugin Guide - SDK Documentation». .
- [104] «Boxee Wiki - Applications», *Main Page - DevWiki*. [En línea]. Disponible en: <http://developer.boxee.tv/Applications>. [Accedido: 30-jul-2012].
- [105] «Symbian SDKs», *Symbian SDKs*. [En línea]. Disponible en: http://developer.nokia.com/info/sw.nokia.com/id/ec866fab-4b76-49f6-b5a5-af0631419e9c/S60_All_in_One_SDKs.html. [Accedido: 27-jun-2012].
- [106] «NOKIA Developer - QT», *Nokia Developer - Qt*. [En línea]. Disponible en: <http://developer.nokia.com/Develop/Qt/>. [Accedido: 27-jun-2012].
- [107] «Qt SDK», *Qt SDK - Nokia Developer Wiki*. [En línea]. Disponible en: http://www.developer.nokia.com/Community/Wiki/Qt_SDK. [Accedido: 27-jun-2012].
- [108] I. Serrano Regol, «Directrices para el Diseño de Interfaces de Televisión Digital Interactiva», feb. 2012.
- [109] «Percepción». [En línea]. Disponible en: <http://psicopsi.com/PERCEPCION-1>. [Accedido: 19-jul-2013].
- [110] «LAS LEYES DE LA GESTALT», *MindMatic*. [En línea]. Disponible en: <http://www.mindmatic.com.ar/gestalt.pdf>. [Accedido: 19-jul-2013].
- [111] A. Crofford, «The Psychology of Color – Must See for Web Designers [Infographic]», 28-ene-2011. [En línea]. Disponible en: <http://www.testking.com/techking/infographics/the-psychology-of-color-must-see-for-web-designers-infographic/>. [Accedido: 19-jul-2013].
- [112] C. Rees, «BBC Enhanced TV Formats», British Broadcasting Corporation, Londres, 2004.
- [113] A. Marcus, «Cross-Cultural User-Interface Design», Aaron Marcus and Associates, Inc, 1144 65th Street, Suite F, Emeryville.
- [114] «Google TV Design Patterns - The 10ft Environment», 14-ago-2012. [En línea]. Disponible en: https://developers.google.com/tv/android/docs/gtv_android_patterns#Environment.
- [115] «Google TV Design Patterns - TV Displays», 14-ago-2012. [En línea]. Disponible en: https://developers.google.com/tv/android/docs/gtv_android_patterns#Display.

- [116] «Google TV Design Patterns - Sound and the UI», 14-ago-2012. [En línea].
Disponible en:
https://developers.google.com/tv/android/docs/gtv_android_patterns#Sound.
- [117] «Google TV Design Patterns - Navigation Design», 14-ago-2012. [En línea].
Disponible en:
https://developers.google.com/tv/android/docs/gtv_android_patterns#Navigation.
- [118] «Google TV Design Patterns - User Interface Design», 14-ago-2012. [En línea].
Disponible en:
https://developers.google.com/tv/android/docs/gtv_android_patterns#UIDesign.
- [119] «Google TV Design Patterns - Additional Tips», 14-ago-2012. [En línea].
Disponible en:
https://developers.google.com/tv/android/docs/gtv_android_patterns#Tips.

Anexo A Estrategias de Recomendaciones Grupales

A.1 Estrategia Calificación plural

Cada individuo califica por su alternativa preferida, luego la alternativa con más cantidad de calificaciones es la ganadora. En caso de necesitarse elegir un conjunto de alternativas, el método se puede repetir primero calificando para elegir a la primera alternativa, luego calificando para elegir a la segunda, etc.

Supongamos la tabla de preferencias de alternativas para cada posición que se aprecia en la ilustración 18.

	1	2	3	4	6	7	8	10
John	A,E,I	E,I	I	I	H,J	J	G	C
Adam	B,D,F,H	B,D,F,H	B,D,F,H	B,D,H	B,H	B	B	C
Mary	A	E	F	D,I	H,J	J	B,G	C
Group	A	E	F	D,I	H	J	B,G	C

Group List: AEF(I,D)HJ(B,G)C

Ilustración 18: Ejemplo Calificación Plural

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

En este ejemplo para el primer lugar hay 3 calificaciones para la alternativa A, 1 calificación para las alternativas B, D, E, F, H, I y no hay calificaciones para las alternativas C, G y J, por lo tanto la alternativa ganadora para el primer lugar es la alternativa A. Repitiendo este razonamiento para el resto de los lugares obtenemos que E es la preferida para el segundo lugar con 3 calificaciones, F para el tercero con 2 calificaciones, I y D empatan en el cuarto lugar con 3 calificaciones (por lo tanto se podrá elegir I para el cuarto lugar y D para el quinto o al revés) H para el sexto con 4 calificaciones, J para el séptimo con 3 calificaciones, B y G empatan en el octavo (por lo tanto se podrá elegir B para el octavo y G para el noveno o al revés) y C para el décimo y último lugar.

A.2 Estrategia utilitaria

Valores utilitarios son utilizados para medir la felicidad de cada individuo en cada alternativa. Se destacan dos versiones de esta estrategia, la aditiva y la multiplicativa.

A.2.1 Aditiva (o promedio)

Para cada alternativa se realiza la suma de los valores utilitarios de cada individuo. El valor utilitario de un individuo en una alternativa generalmente se corresponde con la calificación del individuo en dicha alternativa. Luego las alternativas son ordenadas según el resultado de la suma. Debemos notar que si se realiza el promedio de los valores utilitarios de cada alternativa el orden de la lista sigue siendo el mismo, por ello a esta estrategia también se la conoce como la “estrategia media”.

Supongamos la tabla de la ilustración 19 con los valores utilitarios de cada individuo en cada alternativa.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	21	18	13	22	26	26	17	23	20	22

Group List: (E, F) H (D, J) A I B G C

Ilustración 19: Ejemplo Estrategia Utilitaria Aditiva

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

En este ejemplo las alternativas ganadoras son E y F con una suma de 26 y un valor utilitario medio de 8,67, seguidas de H con una suma de 23 y un valor utilitario medio de 7,67. La suma y valor utilitario medio para estas y el resto de las alternativas se resume en la tabla 11.

Alternativa	Suma	Valor medio	Orden
A	21	7,00	4
B	18	6,00	6
C	13	4,33	8
D	22	7,33	3
E	26	8,67	1
F	26	8,67	1
G	17	5,67	7
H	23	7,67	2
I	20	6,67	5
J	22	7,33	3

Tabla 11: Resultados de ejemplo de estrategia Aditiva

A.2.2 Multiplicativa

Para cada alternativa se realiza la multiplicación de los valores utilitarios de cada individuo. Luego las alternativas son ordenadas según el resultado de la multiplicación. Si bien esta versión es muy similar a la aditiva la lista resultado puede ser muy diferente.

Supongamos la tabla de la ilustración 20 con los valores utilitarios de cada individuo en cada alternativa:

En este ejemplo la alternativa ganadora es F con una multiplicación de 648 y ya no hay empate a diferencia de la versión aditiva.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	100	180	48	378	630	648	180	432	210	384

Group List: F E H J D I (B, G) A C

Ilustración 20: Ejemplo Estrategia Utilitaria Multiplicativa

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.3 Estrategia Recuento borda

Cada individuo ordena la lista según su preferencia. Al elemento en el último lugar de la lista se le otorga 0 punto al anterior 1 y así sucesivamente. Luego para cada alternativa se suman los puntos otorgados por cada individuo y la alternativa con más puntos es la ganadora. En caso de que varias alternativas compartan un mismo lugar en la preferencia de un individuo el puntaje obtenido por cada una será el mismo (para ese individuo) y se calcula distribuyendo los puntos, así si las alternativas $a_1, a_2, \dots, a_n$ comparten la posición “p”, podemos formalizar el cálculo de los puntos que se le otorgan (para ese individuo) infiriendo la siguiente fórmula:

$$\sum_{\{0 \leq i < n\}} (p+i)/n$$

Considerando la tabla con los valores utilitarios del punto anterior tendríamos los puntajes que se aprecian en la ilustración 21.

	A	B	C	D	E	F	G	H	I	J
John	8	1	0	2.5	8	6	2.5	4.5	8	4.5
Adam	0	7.5	4.5	7.5	3	7.5	2	7.5	1	4.5
Mary	9	1.5	0	5.5	8	7	1.5	3.5	5.5	3.5
Group	17	10	4.5	15.5	19	20.5	6	15.5	14.5	12.5

Group List: F E A(H, D) I J B G C

Ilustración 21: Ejemplo Estrategia Recuento Borda

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.4 Estrategia Regla de Copeland

Es una forma de calificación por mayoría. El grupo compara todas las alternativas entre sí y califica, dándole 1 punto a la alternativa con más calificaciones, -1 punto a la alternativa con menos calificaciones y 0 en caso de empate. Podemos formalizar este concepto diciendo que para cada dupla de alternativas (a, b) se otorga un puntaje calculado de esta forma:

$$puntaje(a, b) = \begin{cases} -1 & \text{si existen más integrantes del grupo que prefieren } b \text{ antes que } a \\ 0 & \text{si la cantidad de integrantes del grupo que prefiere } a \text{ es igual} \\ & \text{a la cantidad que prefiere } b \\ +1 & \text{si existen más integrantes del grupo que prefieren } a \text{ antes que } b \end{cases}$$

	A	B	C	D	E	F	G	H	I	J
A	0	–	–	–	0	–	–	–	0	–
B	+	0	–	+	+	+	0	+	+	+
C	+	+	0	+	+	+	+	+	+	+
D	+	–	–	0	+	+	–	0	0	–
E	0	–	–	–	0	–	–	–	–	–
F	+	–	–	–	+	0	–	–	–	–
G	+	0	–	+	+	+	0	+	+	+
H	+	–	–	0	+	+	–	0	+	–
I	0	–	–	0	+	+	–	–	0	–
J	+	–	–	+	+	+	–	+	+	0
Index	+7	–6	–9	+1	+8	+5	–6	0	+3	–3

Group List: E A F I D H J (B, G) C

Ilustración 22: Ejemplo Estrategia Regla de Copeland

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

Luego, por cada alternativa se suman los puntajes obtenidos y la alternativa con mayor puntaje es la ganadora. Considerando la tabla con los valores utilitarios del punto anterior tendríamos los siguientes puntajes que se presentan en la ilustración 22.

Es interesante observar que si $A_{n \times n}$ es la matriz de preferencias para n alternativas, entonces siempre se va a cumplir que $A(i,j) = -IA(j,i)$.

A.5 Estrategia Calificación por aprobación

Cada individuo otorga un puntaje a cada una de las alternativas. Luego se considera un valor mínimo de aceptación y a cada puntaje se lo transforma en un 0 o en un 1 dependiendo si el mismo es menor o igual al mínimo o lo supera respectivamente. Podemos formalizar este concepto de la siguiente manera: sea m un valor por encima del cual se considera que una alternativa es aceptada para un individuo. Luego, para cada individuo i y cada alternativas x se otorga un puntaje calculado de esta forma:

$$puntaje(a, x, m) = \begin{cases} 0 & \text{si el individuo } i \text{ NO califica a la alternativa } x \text{ por encima de } m \\ +1 & \text{si el individuo } i \text{ califica a la alternativa } x \text{ por encima de } m \end{cases}$$

Por último, por cada alternativa se suman los puntajes calculados para cada individuo y la alternativa con mayor puntaje es la ganadora.

Considerando un valor de $m=5$ y la tabla con los valores utilitarios del punto anterior tendríamos los puntajes que se aprecian en la ilustración 23.

	A	B	C	D	E	F	G	H	I	J
John	1			1	1	1	1	1	1	1
Adam		1	1	1	1	1	1	1		1
Mary	1			1	1	1		1	1	1
Group	2	1	1	3	3	3	2	3	2	3

Group List: (D, E, F, H, J) (G, A, I) (B, C)

Ilustración 23: Ejemplo de Estrategia Calificación por aprobación

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.6 Estrategia del menos feliz

Fue la estrategia elegida para el diseño del sistema PolyLens que ya mencionamos antes en este capítulo. Para cada alternativa se considera la menor calificación otorgada por cualquier individuo, luego se ordenan los resultados de manera descendente y la alternativa que encabeza la lista es la ganadora. La idea de esta estrategia es que la felicidad del grupo es dictada por la felicidad del individuo menos feliz. Si bien para grupos pequeños puede ser una buena alternativa, para grupos grandes tiene el inconveniente de que aunque a una enorme mayoría le encante una determinada alternativa podría pasar que la misma nunca se elija como ganadora porque hay un solo individuo que la detesta.[25]

Considerando la tabla con los valores utilitarios del punto anterior tendríamos los puntajes que se presentan en la ilustración 24.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	1	4	2	6	7	8	5	6	3	6

Group List: F, E, (H, J, D), G, B, I, C, A

Ilustración 24: Ejemplo Estrategia del menos feliz

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.7 Estrategia del más feliz

Para cada alternativa se considera la mejor calificación a diferencia de la estrategia del menos feliz que considera la menor.

Considerando la tabla con los valores utilitarios del punto anterior tendríamos los puntajes que se presentan en la ilustración 25.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	10	9	8	9	10	9	6	9	10	8

Group List: (A, E, I), (B, D, F, H), (C, J), G

Ilustración 25: Ejemplo Estrategia del más feliz

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.8 Estrategia media sin infelices

Fue la estrategia elegida (con algunas variantes) para el diseño del sistema MusicFX que ya mencionamos antes en este capítulo. Primero se define un valor m que indica que para que una alternativa pueda participar en la elección todos los individuos deben haber calificado dicha alternativa con un valor mayor que m . Con esto nos aseguramos que cualquier alternativa candidata (y por ende la que resulte ganadora) tenga la aprobación mínima de todos los individuos, de allí su nombre que indica que nadie esta infeliz con los candidatos. Luego que se obtienen las alternativas candidatas se procede de la misma manera que la estrategia utilitaria en su versión aditiva cuando se considera el promedio, o sea se calcula el promedio de las calificaciones de cada individuo y se ordena de manera descendente de acuerdo a este promedio, siendo la alternativa que encabeza la lista la alternativa ganadora.

Considerando $m=3$ y la tabla con los valores utilitarios del punto anterior tendríamos los puntajes que se aprecian en la ilustración 26.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	–	18	–	22	26	26	17	23	–	22

Group List: (E, F), H, (D, J), B, G (threshold 4); (E, F), H, (D, J), I, B (threshold 3)

Ilustración 26: Ejemplo Estrategia media sin infelices

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

A.9 Estrategia con justicia

Esta estrategia trata de evitar el problema que una alternativa nunca pueda ser elegida como vimos que podía llegar a suceder en el caso de la estrategia del menos feliz. La idea es ir otorgando turnos donde en cada turno solo la preferencia de un individuo es considerada. Por ejemplo, podríamos empezar por Jhon que prefiere A, E o I, supongamos que elige E. Luego es el turno de Adam que elige F y luego le llega el turno a Mary que elige A. En este punto ya todos tuvieron su turno, pero como aún restan alternativas para ordenar se prosigue de la misma manera ahora empezando por Mary.

De lo dicho anteriormente se desprende que si bien esta estrategia puede ser una buena alternativa para grupos pequeños, puede que no funcione tan bien en grupos grandes ya que en este caso podría pasar que un usuario pierda el interés luego de mucho tiempo esperando su turno.

A.10 Estrategia del individuo más respetado o dictadura

Se consideran solo las calificaciones otorgadas por el individuo más respetado sin tener en cuenta las calificaciones ni opiniones del resto. Un escenario así podría ocurrir cuando un invitado mira televisión con los miembros habituales de una familia o cuando es el cumpleaños de algún individuo del grupo.

Supongamos entonces que es el cumpleaños de Adam y que el grupo decide que se va a ver lo que quiere Adam, entonces la tabla resultado sería la que se aprecia en la ilustración 27.

	A	B	C	D	E	F	G	H	I	J
John	10	4	3	6	10	9	6	8	10	8
Adam	1	9	8	9	7	9	6	9	3	8
Mary	10	5	2	7	9	8	5	6	7	6
Group	1	9	8	9	7	9	6	9	3	8

Group List: F H D B J C E G I A

Ilustración 27: Ejemplo Estrategia del individuo más respetado

Fuente: Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers [24]

Anexo B Sistemas de Recomendación de Referencia

B.1 IMDB (Internet Movie Database)

El principal motivo por el cual elegimos este sitio fue su popularidad, en este momento es visitado por el 2.1 % de los usuarios globales de Internet en los últimos 3 meses, además se estima que ocupa el puesto 39 del tráfico global de Internet, según Alexa [60] y Compete [61]. Como una consecuencia de dicha popularidad, encontramos reiteradamente en las críticas de las películas referencias a los indicadores de votación de los usuarios de IMDB [62]. Estos se toman como un dato de peso en el contexto de la crítica.

IMDB comenzó como un proyecto de un grupo internacional de aficionados al cine, esencialmente algo realizado por fanáticos para otros fanáticos del cine. Se trata de una enorme colección de información de películas, series y programas de TV. Trata de catalogar todos los detalles pertinentes acerca de cada contenido, desde quienes participan, quien lo realizó, lugares de filmación, etc. Permite agregar contenidos a listas de reproducción, se pueden encontrar comentarios, opiniones, valoraciones de otros usuarios y sitios de fanáticos en la Web para cada contenido. Muestra contenidos relacionados a partir de los gustos de los usuarios y realiza sugerencias personalizadas. Si bien no tiene actualmente una API oficial, tiene disponible la descarga de su base de datos actualizada periódicamente y existen varios sitios que trabajan sobre esta base para ofrecer APIs públicas vía Web.

B.1.1 Funcionalidades

Características sociales

- Lista de amigos.
- Mensajes privados entre amigos.
- Mostrar a la comunidad lo que estoy viendo.
- Colaboración entre “contribuyentes” para ampliar la información de los contenidos y críticas.
- Comentarios en los contenidos y posibilidad de respuestas creándose hilos de discusión.
- Últimos contenidos visualizados por el usuario (histórico de visualización).
- Valoración de Videos (de 1 a 10 estrellas).
- “A quién le gustó este video también le gustó:”.

Integración con Redes sociales

Permite publicar de valoraciones sobre los contenidos en:

- Facebook.
- Twitter.

Recomendaciones

Primero, se consideran todas las películas y programas que el usuario haya valorado o añadido a su lista de visualización. Luego, se comparan con las calificaciones realizadas por otros usuarios. Finalmente, se pueden encontrar películas y programas de televisión a partir de las personas con gustos similares. Para cada recomendación, se puede ver una lista de las películas o programas de televisión en que se basa la recomendación, y cada uno de estos o bien han tenido una calificación alta por parte del usuario, o bien han sido agregados a su lista de visualización. Internamente cada vez que el usuario valora de manera positiva un contenido o agrega un contenido a su lista de visualización IMDb, considera que es un contenido de interés. El usuario siempre tiene la posibilidad a través del botón “no” de indicar que la recomendación no es apropiada, también puede enviar un correo a engage@imdb.com indicando las características de su preferencia. Además, se brinda la posibilidad al usuario de mejorar sus recomendaciones personalizadas invitándolo a valorar diferentes colecciones de contenidos para que el sistema aprenda de sus gustos.

Video

Permite la reproducción de avances y cortos en la propia página. Ofrece enlaces a Internet Archive [63] para films y películas de archivo. Ofrece enlaces a Amazon Instant Video para comprar o rentar contenidos completos de pago. También muestra una galería de imágenes asociada al contenido a modo de presentación. Lista de Videos relacionados. Ofrece una guía de programación para las señales más conocidas en la sección “TV Listings”.

Proveedores de Video soportados

- Amazon Instant Video [64].
- IMDb (avances).
- Internet Archive [63].

B.1.2 Análisis técnico

Servicios a otras aplicaciones y API

Actualmente las interfaces oficiales con sistemas externos se logran a través de la descarga local de la base de datos que es actualizada periódicamente. El formato de intercambio para la base de datos se basa en archivos de texto plano [65].

Hoy día no cuenta con una API pública oficial, sin embargo existen varios sitios que han construido APIs no oficiales que trabajan sobre una copia de la base de datos de IMDb que es sincronizada con bastante frecuencia.

Algunos de estos sitios son:

- IMDb API [66]
- IMDb Web Service [67].
- IMDb API, Stavros Korokithakis [68].

A modo de ejemplo consideraremos IMDb API:

La función de búsqueda se puede acceder vía REST sobre HTTP GET. Por ejemplo, para buscar "El orfanato" (notar que buscamos por el título en español y no por su título original) enviamos el siguiente request HTTP:

GET /?t=El+Orfanato&plot=full&tomatoes=true HTTP/1.1

Host: www.imdbapi.com

Y la respuesta que obtenemos es en formato json con los siguientes datos:

Title	The Orphanage
Year	2007
Rated	R
Released	11 Jan 2008
Genre	Drama, Mystery, Thriller
Director	Juan Antonio Bayona
Writer	Sergio G. Sánchez
Actors	Belén Rueda, Fernando Cayo, Roger Príncipe, Mabel Rivera
Plot	Laura, a former orphan, raises her adopted son Simón together with her husband Carlos in an old house and former orphanage where she was raised. While at the orphanage Simón tells Laura that he has five invisible friends which she believes are a product of his active imagination. Laura decides to reopen the orphanage to cater for disabled children and throws a party. During the party Simón tries to persuade Laura to go and take a look at his friends cabin but she's too busy. Later on she sees a mysterious masked boy and realizes that Simón has also disappeared. Laura feels the presence of other people in the house and months later Laura invites a team of parapsychologists to try to unravel the mystery
Poster	http://ia.media-imdb.com/images/M/MV5BMtc3MjE0NzQzMV5BMl5BanBnXkFtZTYwM-zI0ODc4._V1_SX320.jpg
Runtime	1 hr 45 mins
Rating	7.6
Votes	56133
ID	tt0464141
tomatoMeter	85
tomatoImage	certified
tomatoRating	7.3
tomatoReviews	164
tomatoFresh	139
tomatoRotten	25
Response	True

Tabla 12: Respuesta de IMDbApi de información de película "El Orfanato"

Podemos ver que la consulta recuperó el título "The Orphanage" (título original) aunque no haya sido el título que habíamos solicitado, lo cual parece muy inteligente. Como en la petición pusimos "tomatoes=true" nos agrega información desde "rotten tomatoes".

Plataformas soportadas

Mobile Software Nativo:

- iPhone.
- IpodTouch.
- Android teléfonos móviles y tabletas.
- Windows Phone 7.
- BlackBerry.

Web Mobile

- Reproducción de avances con HTML5.

Web

- Reproducción de avances con HTML5 (para los navegadores que lo soportan).
- Reproducción de avances con JW Player - Flash (para los navegadores que no soportan el tag video de HTML5).

B.1.3 Análisis Crítico

Características fuertes

- Gran base de datos de contenidos (disponible para descarga, estructurada en archivos de texto plano).
- Muchas características sociales, destacándose la colaboración de los contribuyentes.
- Variedad de dispositivos soportados.
- Sitio Web con un diseño de interfaz de usuario agradable.
- AKA (*Also Known As*) que permite la búsqueda de contenidos por títulos que no son los originales.
- Reproducción de contenido en HTML5, agregando gran portabilidad.
- Recomendaciones personalizadas en base a los contenidos de interés del usuario y de las preferencias de otros usuarios con gustos similares.

Características débiles

- No tiene una API pública oficial para acceder al detalle de los contenidos.
- Bajo tiempo de respuesta de navegación entre páginas del sitio.
- Dentro de las recomendaciones personalizadas no permite filtrar por algunas variables como ser: género o estado de ánimo del espectador. Por este motivo las recomendaciones son una lista de películas con características heterogéneas.

B.2 Rotten Tomatoes y Flixter

Este sitio junto al sitio Flixster [69] fue comprado por Warner Bros. en mayo del año 2011, actualmente cuenta con más de 30 millones visitantes únicos por mes y se estima en el lugar 759 del ranking mundial de trafico según Alexa [70] y 849 según Compete [71]. El sitio *lifehacker* lo posiciona entre los 5 mejores recomendadores de películas [72]. Se puede observar un funcionamiento en complemento los sitios Rotten Tomatoes [73] y Flixter por lo que se analizaron como un conjunto.

Es un sitio social que ofrece a sus usuarios la posibilidad de calificar las películas, seguir aquellas películas que estén por lanzarse, estar informado de los últimos estrenos y avances e informarse sobre los actores de sus películas favoritas. También se puede seguir la cinematografía, permitiendo a sus usuarios formar parte de una comunidad en la que en todo momento pueden compartir sus gustos y/o novedades de películas con sus amigos, opinar sobre películas, avances como así también hacer nuevos amigos con gustos similares.

B.2.1 Funcionalidades

Características sociales

Foros de discusión para opinar con amigos y otros usuarios sobre películas u otros temas relacionados (sitio de la comunidad de Flixster).

Integración con Redes sociales

Integración con redes sociales Facebook, Myspace, Babo, Orkut de las que el usuario forme parte.

Información de películas

- Avances/clips.
- Cast de actores que participan en esta, pudiendo fácilmente acceder a información de los mismos.
- Fotos de la película, backstage, del reparto, etc.
- Comentarios de críticos reconocidos.
- Clasificación y comentarios realizados por los amigos.
- Listado de películas similares.
- Integración con sitios de contenido multimedia como Ultravioleta [74] y NetFlix [37]. Con este el usuario puede acceder/comprar fácilmente a las películas de su interés.

Recomendaciones

Al autenticarse se puede ver una lista de películas recomendadas para el usuario en la página principal en la sección *Recommended for You*, donde se llegan a recomendar 16 películas. Según un post publicados por moderadores de la comunidad de Flixster [69], las recomendaciones se basan principalmente en dos aspectos: películas que les han gustado a los amigos del usuario y las películas similares a las que el usuario a calificado mejor. Para confeccionar este último

conjunto de películas similares se utilizan las sugerencias que el usuario ha realizado a sus amigos y cómo las ha clasificado.

Las opciones de clasificación sobre una película que puede realizar un usuario son las siguientes que son excluyentes entre sí.

- *Not interested*. El usuario muestra total desinterés en ver la película.
- *Want to see it*. Muestra interés en verla, de esta forma también se agrega a una lista de películas del usuario de manera que pueda seguir las mismas y acceder rápidamente a estas en su perfil.
- Calificación. Puede calificar la películas entre los valores 0,5 a 10,0 en múltiplos de 0,5. Esta calificación se otorga “pintando” 5 estrellas donde se puede optar por pintar estrellas enteras y/o media estrella.

Actualmente el usuario también puede optar por importar las calificaciones que haya realizado en Netflix. Una vez que se selecciona una película, se puede ver los comentarios más recientes realizados por los usuarios junto con su valor de calificación. Aparte de esta una sección con gran variedad de comentarios de críticos de empresas principalmente de Estados Unidos como son New York Magazine, Detroit News, Denver Post, ComingSoon.net, Movies.com, etc.

Proveedores de Video soportados

- NetFlix [37].
- Ultravioleta.

B.2.2 Análisis técnico

Servicios a otras aplicaciones (API)

Cuenta con aplicaciones en las redes sociales Facebook, Myspace, Bebo y Orkut, a través de las cuales se puede publicar películas, encontrar amigos como así también exportar sus películas favoritas a estas redes sociales.

Ofrece un API pública para acceder a la información sobre su catálogo de películas, se puede acceder vía REST y retorna la información en formato json, fue liberada en Febrero del 2011. Para utilizar esta API con mínimos requerimientos de uso, solamente se precisa estar registrado como desarrollador. Esto sí es obligatorio puesto que se necesita una `api_key` para consumir la misma [75].

A continuación se muestra ejemplos de uso.

Esta petición retorna los dos principales recursos que provee la API que son lists y movies

<http://api.rottentomatoes.com/api/public/v1.0.json?apikey=gnhuuuqu77gnazascbaz6hva>

- links
 - lists: <http://api.rottentomatoes.com/api/public/v1.0/lists.json>
 - movies: <http://api.rottentomatoes.com/api/public/v1.0/movies.json>

Por ejemplo, podemos armar una búsqueda de la película Machete, cuyo resultado se muestra a continuación, ingresando en la busqué el texto “Machete 2010”.

http://api.rottentomatoes.com/api/public/v1.0/movies.json?apikey=gnhuuuqu77gnazascbaz6hva&q=Machete+2010&page_limit=1

En este caso nos retorna 5 resultados posibles, a continuación mostramos la información de una de las películas encontradas.

id	770672158
title	Machete
year	2010
mpaa_rating	R
runtime	105
critics_consensus	Machete is messy, violent, shallow, and tasteless -- and that's precisely the point of one of the summer's most cartoonishly enjoyable films
release_dates	theater: 2010-09-03 dvd: 2011-01-04
critics_rating	Certified Fresh
critics_score	73
audience_rating	Upright
audience_score	64
synopsis	Robert Rodriguez brings a new breed of antihero to the big screen in this hyper-violent adaptation of the "fake" trailer originally featured in the 2007 double-feature exploitation throwback Grindhouse. Machete (Danny Trejo) is a former Mexican Federale who plays by his own rules, and doesn't answer to anybody...
thumbnail	http://content8.flixster.com/movie/11/15/24/11152450_mob.jpg
Profile	http://content8.flixster.com/movie/11/15/24/11152450_pro.jpg
detailed	http://content8.flixster.com/movie/11/15/24/11152450_det.jpg
original	http://content8.flixster.com/movie/11/15/24/11152450_ori.jpg
abridged_cast	Danny Trejo, id:358124392, characters: Machete Robert De Niro, id:162655521, characters: Sen. McLaughlin Jessica Alba, id:162652782, characters: Sartana Steven Seagal, id:162671905, characters: Torrez
alternate_ids	imdb: 0985694

Tabla 13: Respuesta Api Rotten Tomatoes de información de película “Machete”

En el sitio para desarrolladores de Rotten Tomatoes [75] se puede ver ejemplos y la definición del resto de los métodos ofrecidos.

Plataformas soportadas

Mobile Software Nativo:

- iPhone, iPod touch, y iPad. Requiere iOS 4.0 o superior.
- Android.
- Windows Phone 7.
- BlackBerry.

Web

- Reproducción de avances con Flash.

Web Mobile

- Reproducción de avances con tag video de HTML5.

B.2.3 Análisis Crítico

Características fuertes

- Simple e intuitiva para el usuario.
- Abundante información sobre las películas desde diversas fuentes especializadas.
- Reproducción de contenido en HTML5, agregando gran portabilidad.
- Ofrece la información de su catálogo de películas con un servicio REST público.

Características débiles

- Dentro de las recomendaciones personalizadas no permite filtrar por algunas variables como ser: género o estado de ánimo del espectador. Por este motivo las recomendaciones son una lista de películas con características heterogéneas.

B.3 Movielens

Movielens [76] no fue elegido por su popular como los anteriores. Nuestro interés se da porque este sitio es un proyecto de investigación académico del equipo GrupoLens [77], Departamento de Ciencias de Computación, Universidad de Minnesota. Una de las áreas de interés de este grupo, son los Algoritmos de recomendación, en particular este sitio se basa en filtrado

colaborativo. En ese marco, desarrollan este proyecto como caso de estudio, donde necesitan de los datos que pueden ofrecer los espectadores de películas. Cuenta con un fuerte respaldo de documentación de dominio público, fundamentando las decisiones de diseño y algoritmia en cuanto a las recomendaciones propuestas. Es aquí donde podemos encontrar un trabajo de investigación sobre recomendaciones en general y en particular recomendaciones de contenido multimedia. Estos son los motivos por los cuales queremos profundizar en su estudio.

Utiliza filtrado colaborativo para hacer recomendaciones de películas y vídeos que el usuario puede disfrutar, y para ayudarlo a evitar lo que no es de su gusto.

Con base en las calificaciones de las películas que ha visto, Movielens genera predicciones personalizadas para películas que no se han visto todavía. Además, es un vehículo de investigación única para que decenas de estudiantes universitarios de diversos aspectos de las tecnologías de personalización y de filtrado.

B.3.1 Funcionalidades

Características sociales

- Se puede preguntar a la comunidad.
- Los usuarios y sus datos son públicos.

Integración con Redes sociales

No tiene.

Recomendaciones

En este documento vamos a ver una versión simplificada ya que nos vamos a explicar en el Estado del Arte sobre Recomendaciones. Se toman todas las calificaciones y usuarios, se ponen bajo un modelo que hace coincidir a usuarios con opiniones similares sobre películas, el llamado "punto-punto" del modelo. Cada miembro del sistema tiene una "vecindad" de otros usuarios afines. Las valoraciones de estos vecinos se utilizan para realizar recomendaciones personalizadas para el usuario destino [64][63]. Unos de los datos de entrada para este modelo son los tags que el usuario puede ingresar a la película, estos tags son libres, el usuario puede ingresar cualquier valor. Luego con estos tags se construye los tags de la comunidad para una película de forma ordenada. Dentro de las recomendaciones personalizadas, permite filtrarlas por genero, tags, época, además agrega el dato de predicción de calificación del usuario. Con esto se logra una mayor precisión en la recomendación.

Video

No tiene reproducción de video o asociación con algún proveedor de video.

B.3.2 *Análisis Crítico*

Características fuertes

- El algoritmo de recomendación es público y esta fundamentado.
- La posibilidad de ingresar variables de entrada a las recomendaciones personalizadas permite obtener un mejor resultado.

Características débiles

- Comparativamente tiene pocos usuarios, en particular, para este proyecto es importante contar con la valoración de las películas que hacen los usuarios de forma masiva, para hacer más efectivo el modelo de recomendación.
- No tiene reproducción de contenido propia, si bien tiene conexión con NetFlix, la mayoría de las películas no se encuentra esa asociación, cuando se quiere navegar a NetFlix, sale un error que no encuentra la película.
- No tiene información extra de la películas, es decir la información que brinda es reducida a: Título, Reparto, Director, Año, Genero y Sinopsis.

B.4 Hulu

El sitio Hulu [78] tiene varios aspectos interesantes: en primer lugar es uno de los más populares, con 1.5 millones de usuarios subscriptos (pagan por el servicio plus) [79], con la aclaración de que solo se puede subscribir desde Estados Unidos o Japón. Genera una gran cantidad de tráfico a nivel mundial, posicionándose en el lugar 216 según las estimaciones de Alexa sobre el tráfico mundial [80] y en la posición 100 según Compete [81]. Es proveedor contenido multimedia, esta asociado con cadenas reconocidas como FOX, NBC, MTV, ABC entre otras. Y finalmente cuenta con versiones para gran variedad de dispositivos, algo que nos interesa estudiar como se logra.

Cuenta con una versión Web gratuito que ofrece: *streaming* de programas de televisión y películas de mas de 400 proveedores entre los que se reconocen ABC, BET, Comedy Central FOX, NBC, MTV, VH-1 y muchas otras redes y estudios. Actualmente sólo se ofrecen a los usuarios en los Estados Unidos y Japón.

Tiene una amplia cantidad de opciones de video clips/avances para todo tipo de películas, documentales y series de tv.

La versión paga es Hulu Plus y tiene las siguientes diferencias con la versión gratuita, básicamente en 3 puntos

- Dispositivos de acceso: Hulu es una opción gratuita para que los usuarios vean programas de TV en sus computadoras, así el acceso esta condicionado a ser mediante

computadoras únicamente. La opción Hulu Plus habilita a que los usuarios puedan ver el contenido de Hulu en calidad HD (720p) a través de TV, Reproductores de *Blu-ray*, consolas de juegos, *set-top boxes*, iPhones y demás dispositivos conectados a Internet claro. Cabe destacar que por cada cuenta de usuario, el *streaming* de video solo esta disponible para un dispositivo a la vez.

- Contenido: Hulu ofrecer para la mayoría de sus programas de TV los últimos episodios de estos mientras que Hulu Plus ofrece la temporada completa. Si bien se puede probar Hulu Plus durante 7 días, al estar su contenido restringido a Estados Unidos y Japón no hay manera para nosotros de poder apreciarlo.
- Calidad de video: Hulu ofrece los videos en calidad estándar mientras que Hulu Plus permite ver en HD siempre que se cuente con esta opción.

Es de destacarse su sección de desarrollo denominada Labs, donde Hulu publica información, demos y características de las nuevas funcionalidades y proyectos que se encuentra desarrollando. En lo que nos compete, es de interés resaltar el proyecto Social, que busca mejorar y aprovechar funcionalidades que Facebook ofrece para que el usuario pueda compartir sus experiencias en Hulu con sus amigos.

B.4.1 Funcionalidades

Queue. La *Queue* es una lista de reproducción que se encuentra accesible en todas las páginas del sitio. El usuario en todo momento para todo video, avances, clip, tiene la opción de agregarlo a su *Queue* para reproducirlo luego. También a esta se agregarán automáticamente los episodios y/o clips de series de los que el usuario sea seguidor y/o avances de películas que haya agregado a su lista de favoritos.

Playlist. El sitio permite que el usuario gestione sus propias listas de reproducción a parte de la *Queue*. Estas pueden compartirse/enviarse a amigos como sugerencia de reproducción.

Seguimiento de programas de TV. Al agregar una serie a la lista de favoritos del usuario, le da la opción de realizar un seguimiento cada nuevo capítulo.

Características sociales

Hilos de discusión. Para todo clip, avances, serie y película el usuario tiene la opción de crear hilos de discusión relacionados con la misma. Estos hilos tienen carácter únicamente público si bien se pueden crear cuantos se desee por lo cual también es de notar la gran cantidad de hilos que hay. Por esto también y como aparente medida para facilitar la moderación de los hilos, todo

usuario encontrará en cada post opciones para clasificarlo como spam, ofensivo o spoiler (“Something you would've preferred to find out yourself”, revelar lo que sucederá)

Amigos

De estos se puede conocer sus actividades recientes, su *Queue*, sus favoritos. Según este lo decida se puede tener acceso a algunas de sus listas de reproducción.

Integración con Redes sociales

Permite autenticarse a través de red social Facebook. Si bien no permite conectarse a otras redes sociales o microblogs, el usuario puede compartir links al contenido que desee en Twitter, vía email y en obviamente en Facebook. Luego no ofrece más beneficios de conectarse como usuario de Facebook al momento si bien .

Recomendaciones

Independientemente de si el usuario se encuentra viendo una serie de TV, una película o simplemente clips, al final de la página y como última sección se encuentran las recomendaciones. Esta siempre recomienda del tipo de video que se encuentre viendo, si se encuentra viendo un avances de película, le recomendará avances sobre la misma película y/o películas similares, o en el caso de una serie de TV le recomendará los siguientes capítulos de esta y/o los anteriores más próximos de la misma temporada. También la sección de Lista actúa de lista de sugerencias de videos relacionados al que se encuentra viendo actualmente. En todo caso no se aprecia que las recomendaciones se reparen en lo que el usuario califica y suele ver en el sitio. Esto capaz es debido a la gran cantidad de contenido que llevan a sugerirle al usuario variedad de lo mismo. Si bien a la fecha de Abril de 2012 se encuentra en marcha un proyecto Hulu para mejorar el sistema de recomendaciones [20], más precisamente están actualizando su algoritmo de recomendación de forma que este tome como de mayor importancia las calificaciones que hace el usuario sobre su programación favorita, y que no se base tanto en el tipo de contenido que ve.

Video

Al finalizar un video en el reproductor se muestran botones para compartir este en Twitter, Facebook o vía mail, como también da opción de copiar código HTML para embeberlo en un sitio externo. Luego también muestra en forma de *thumbnails* videos relacionados con este para reproducirlos directamente. Respecto a los videos relacionados el reproductor cuenta con opción de “Reproducir continuamente” de forma que al finalizar un video se realiza una cuenta regresiva de 5 segundos y se continuar reproduciendo un de estos videos.

Otras

Encuestas de publicidad. Cada cierto tiempo le pregunta al usuario si desea participar en una encuesta con fin de mejorar la publicidad ofrecida. En esta encuesta hace preguntas como: quien es dueño de la computadora de la que accede, con quien la comparte, genero del usuario, realiza preguntas detalles de las propagandas sobre tiendas al por menor (Ej. Walmart, Macy's) que se muestran asiduamente. Luego al final de estas permite que el usuario elija una organización (pro-salud, pro-apoyo a niños huérfanos, etc) para que esta prevalezca sobre las que las organizaciones que se publicitan en el sitio.

Proveedores de Video soportados

Hulu actúa como distribuidor de contenido. La gran variedad y cantidad de contenido que se encuentra en Hulu se debe a su gran cantidad de *partners* de contenido.

Sitios fuentes de Videos

Dentro de la sección *Partners* de su sitio, se encuentra la lista de *partners* de contenido de Hulu que supera los 400 integrantes.

B.4.2 Análisis técnico

Hulu permite que el usuario califique los videos mediante un rate de 1 a 5 estrellas. Cuando se accede a la página de uno de sus videos, debajo se encuentra lo que denominan Lista de clips, en la cual se encuentra marcado el video que se esta reproduciendo junto a una lista de videos relacionados con el mismo en donde el usuario puede calificarlos y/o agregarlos a su *Queue*. El usuario también tiene opción de definir/marcar tags para clasificar el video. Debajo del reproductor se puede ubicar una sección denominada Tags en la que el usuario a su criterio puede seleccionar las tags que ya se encuentran asignadas al video por la mayoría de los usuarios como así también definir nuevas tags. Luego a través de estas se puede buscar contenido similar según el criterio de los usuarios.

Interacción con fuentes de video

Para reproducir los videos utiliza su propio reproductor flash. Este lo embebe en la página y le ingresa parámetros para identificar el contenido a reproducir que el sitio mismo provee.

Servicios (API)

Hulu ofrece información de los videos y demás contenido del sitio a través de servicios REST en formato json. Se puede obtener información una vez que se identifica la URL del contenido. Dentro de la información que brinda sobre el video, agrega el protocolo oEmbed [82] como servicio a otros sitios que quieran usar a Hulu como proveedor de contenido, embebiendo la reproducción desde Hulu.

A continuación mostramos un ejemplo para obtener información de un video relacionado con la película The Avengers, remarcamos en negrita el titulo del video, la ruta del video si se pretendiera embeber en un reproductor y el código HTML para embeberlo en un sitio externo.

Peticion: `http://www.hulu.com/api/oembed.json?url=http://www.hulu.com/watch/287904/the-avengers.`

author_name	Hulu
type	Video
thumbnail_width	145
title	The Avengers (Movie Trailers)
thumbnail_height	80
thumbnail_url	http://thumbnails.hulu.com/156/40050156/40050156_145x80_generated.jpg
Width	512
Duration	125.12
embed_url	http://www.hulu.com/embed/uoDUHggeJ_xiDeO-6syXUg
provider_name	Hulu
Version	1.0
air_date	Fri May 04 00:00:00 UTC 2012
height	296
provider_url	http://www.hulu.com/
html	<object width="512" height="296"><param name="movie" value="http://www.hulu.com/embed/uoDUHggeJ_xiDeO-6syXUg"></param><param name="flashvars" value="ap=1"></param><embed src="http://www.hulu.com/embed/uoDUHggeJ_xiDeO-6syXUg" type="application/x-shockwave-flash" width="512" height="296" flashvars="ap=1"></embed></object>
cache_age	3600

Tabla 14: Respuesta Api información de película Hulu

El código HTML para embeber su reproductor en sitios externos también se ofrece al explorar videos dentro del sitio. Cuando se reproduce cualquier video, entre las opciones para compartirlo, se puede copiar y personalizar el código HTML necesario para poder embeber el reproductor y vincularlo al video en cuestión. Funcionalidad no menor en este “generador” de HTML es que permite elegir qué captura de la película sea el *thumbnail*.

Plataformas soportadas

Lista de dispositivos para los que se ofrece Hulu Plus

Algunos de estos son:

Mobile Software Nativo:

- Phone (Phone 4, 3GS, iOS 4).
- iPad.
- iPod touch (Generacion 3 y 1 iOS 4).
- Android.
- Amazon Kindle Fire.
- Nook Tablet.
- Nintendo 3DS.

Televisores y reproductores de Blu-Ray:

- Samsung (Solo TV Internet compatible y reproductore Blu-ray).
- VIZIO (Solo TV Internet compatible y reproductore Blu-ray).
- Sony ('09+ BRAVIA TVs, '09+ BRAVIA Blu-ray reproductores, Netbox y Dash).
- LG (Solo TV Internet compatible).
- Panasonic (VIERA, solo TV Internet compatible).
- Sharp (solo TV Internet compatible).
- Toshiba Blu-rays (Solo Internet compatible).
- Dinex Blu-rays.

Reproductores Media

- Roku Streaming Player.
- Tivo Premiere.
- WD TV Media Players.
- RCA Media Players.
- Netgear Media Players.

Consolas de juegos

- Xbox 360.

- Sony PlayStation 3.
- Wii.

Web

Reproducción con Flash.

B.4.3 Análisis Crítico

Características fuertes

- La gran variedad de contenido y la rapidez con la que se agrega los últimos lanzamientos y novedades es claramente un punto fuerte junto con la cantidad de dispositivos de acceso a la que ofrece el servicio en su versión paga Hulu Plus son claramente sus puntos más fuertes.
- La posibilidad de reproducir directamente es un punto a favor a la hora de recomendar ya que se puede saber si el usuario realmente vio la película y como accedió a ella, por recomendación o por una búsqueda.
- Disponibles en diferentes pantallas, gran portabilidad sobre diferentes plataformas.

Características débiles

- Actualmente no presenta gran interacción con sitios sociales, si bien tiene en marcha proyectos que se enfocan en este aspecto como es Social.
- No cuenta con un sistema de recomendaciones personalizadas para cada usuario.

B.5 Nanocrowd

Nanocrowd [83] nos resulta un sitio interesante por algunas variantes en el sistema de recomendación, en particular, como determina si una película le gustó al usuario o no. Está incluido en el top five de sitios de recomendaciones para NetFlix según *switched* [84]

Nanocrowd es un sitio Web colaborativo donde los usuarios dan su opinión a través de comentarios sobre las películas que vieron, así otros pueden saber si la película será de su interés o no. El análisis de los comentarios es la materia prima que abastece a las principales características y funcionalidades de Nanocrowd. El análisis de las frases y palabras en los comentarios es tan potente que hace que el sistema no requiera de la puntuación de los usuarios para saber si les ha gustado o no una película. De esta manera, se logra medir el “crowd rating” de cada película, el cual actúa de manera similar al concepto de valoración promedio de contenidos de otros sitios. También a partir del análisis de los comentarios, surge uno de los conceptos más importantes y novedosos que utiliza el sitio para la generación de recomendaciones, los “nanogenre”. Un “nanogenre” no es más que la lista de tres palabras que el sistema infiere son las palabras que mejor describen a la película [85][86][87].

Entre las principales funcionalidades ofrecidas a los usuarios se encuentran la recomendación de películas a partir de alguna otra película que el usuario indique, y a partir de ésta, las listas de las películas serán las películas que mas le gustará al usuario (y menos). Otras características son la “nube de etiquetas” con las palabras asociadas a una película y la posibilidad de agregar películas a la lista “Mis películas” donde es posible indicar el grado de interés en la película, una valoración (de 1 a 5) y alguna nota recordatoria.

Finalmente, ya sea luego de una búsqueda o sugerencia es posible reproducir la película o contenido directamente desde alguno de los proveedores de video a demanda soportados, algunos gratuitos como Hulu o Crackle y otros de pago como Amazon.

B.5.1 Funcionalidades

Características sociales

No tiene.

Integración con Redes sociales

Si bien utiliza las redes sociales para promocionar el sitio, no brinda características propias integradas a las redes sociales.

Recomendaciones

- Recomendaciones personalizadas logradas a través de un wizard de tres pasos en pocos minutos haciendo uso de los nanogenres.
- Lista de películas que más le gustarían al usuario a partir de una película seleccionada por el usuario.
- Lista de películas que menos le gustarían al usuario a partir de una película seleccionada por el usuario.
- Predicción de interés del usuario sobre una película seleccionada por éste a partir de un quizá.

Proveedores de Video soportados

- Hulu [78].
- Crackle [39].

- Amazon Instant Video [64].
- Netflix [37].
- Internet Archive [63].

B.5.2 Análisis técnico

Calificación del video

El sistema logra decidir si a un usuario le ha gustado o no un video a partir de un análisis gramatical de los comentarios que el usuario ha realizado sobre el contenido. De esta manera el sistema calcula la valoración promedio que debería tener el video.

Plataformas soportadas

Mobile Software Nativo:

- iPad.

Web

B.5.3 Análisis Crítico

Características fuertes

- Cuando realiza recomendaciones toma en cuenta variables de entrada que ayudan a acotar los resultados para obtener lo que el usuario está buscando.

Características débiles

- Pobre en características sociales, solo cuenta con los comentarios. Tampoco se integra con otras redes sociales.
- No cuenta con un API pública para obtener información sobre su catálogo de películas.
- No ofrece versiones para distintas pantallas, ni cuenta con una versión Web para móviles.

B.6 Jinni

En el análisis de sitios de recomendación de películas, CNET pone a Jinni [88] en primer lugar [89], también es elegido por Google como el primer partner de Google TV para formar una alianza estratégica [90] y es destacado en “CableLabs Winter Conference” [91]. En particular notamos que esta presente más fuertemente que en otros sitios el concepto de ingeniería semántica para buscar películas, nos interesa estudiar de qué forma se caracterizan las películas y que variables se utilizan.

Este sitio esta enfocado principalmente a la recomendación de videos a través del perfil de usuario y una caracterización de cada película. Esta caracterización la llaman “personalidad la película” o el “genoma de la película”. El genoma de la película es creado por profesionales del cine [92] y se ayuda con procesamiento de lenguaje natural y aprendizaje automático. Además tiene en cuenta el estado de ánimo del usuario, el cual es solicitado a la hora de recomendar películas.

¿Cómo funciona?

En el interior, el genoma se puede dividir en dos: la experiencia – el estado de ánimo y el tono de su contenido – y la historia – elementos de la trama (One Man Army, la batalla de los sexos), estructuras (no lineal, la historia-dentro-de-historia), las banderas (violencia, desnudos) y mucho más. El genoma también incluye muchos aspectos externos, como la entrega de premios.

B.6.1 Funcionalidades

Características sociales

No tiene el concepto de “amigo” de manera que se pueda elegir, sí está el concepto de “vecino” que significa alguien con un perfil de película parecido. Podes ver su opinión, seguramente se utiliza su votación para recomendarte.

Integración con Redes sociales

Solo puede conectarse con Facebook y compartir con Twitter y Facebook.

Recomendaciones

Primero se crea un perfil de usuario con la siguiente información, con los valores gusta, sin opinión, o no gusta:

Género: acción, aventura, animación, comedia, crimen, documental, drama, erótico, familia, fantasía, extranjero, histórico, horror, independiente, música, musical, misterio, parodia, período, romance, ciencia ficción, deporte, novela de suspense, guerra occidental.

Categorías: Amistad, infidelidad, Leyendas y Mitos, Padres e Hijos, Policía de Investigaciones, política, Carrera contra el tiempo, Auto, Descubrimiento, Relaciones sexuales, habilidad sobrenatural, Descubre la verdad, juventud

Jinni asigna grupos de genes a cada usuario, procedente de los aspectos del contenido que te gusta. Estos genes, junto con algunas otras medidas, se ajustan continuamente a medida que

aprendemos a partir de sus puntuaciones, críticas y otras acciones en el sitio Web. De esta manera se compone una versión simplificada, un boceto del genoma de películas predilectas para el usuario. No hay dos personas que tienen exactamente el mismo boceto.

Por otro lado, las películas son etiquetadas manualmente por un equipo de profesionales del cine. Cada título cuenta con medio centenar de genes, entre miles de posibilidades. Luego, utilizando aprendizaje automático y procesamiento del lenguaje natural, se calculan de forma automática los índices de Jinni para los nuevos títulos del sistema, utilizando el análisis de opiniones de usuarios y los metadatos. También incorpora múltiples perspectivas (de comentarios) en lugar de una mera opinión de una persona. Todo el mundo que vota en los genes de la película, esto sirve como insumo para que el equipo de Jinni pueda revisar y mejorar los genes asignados a la película.

Las recomendaciones se hacen mediante la comparación del genoma de películas que se compone para el usuario y los genes de todos los títulos en nuestro catálogo, pensando en las preferencias del usuario y de algunos otros filtros. De manera simplificada, si el usuario tiene los genes de triángulos amorosos, vamos a recomendar títulos con esos genes dominantes. El usuario recibe recomendaciones de Jinni, teniendo en cuenta las películas que siguen sus vecinos (la gente identificada por tener sabor similar).

El servicio incluye de búsqueda semántica, un enfoque de sentido basado en la interpretación de las consultas mediante la identificación de conceptos dentro del contenido, en lugar de palabras clave. La búsqueda es por el humor, la trama y otros parámetros, e incluye opciones para explorar y refinar con términos adicionales.

Ejemplo simplificado del genoma de la película Malcolm X:

El estado de ánimo de Malcolm X se invita a la reflexión, emocional y cautivadora. La trama se centra en torno al liderazgo, con un aumento de la parte superior, y el racismo. Es un drama, el período la película histórica. Estilísticamente, Malcolm X es épica. En el enfoque, es seria y realista. El ritmo es lento. Malcolm X se encuentra en Harlem. Ocurre en la década de 1950 y en la década de 1960. Se extrae de una biografía, originalmente una historia real, y asada en un libro. Malcolm X es conocido por ser un ganador del premio de crítica y público. Tenga en cuenta que incluye malas palabras y el contenido violento.

Video

Redirige a otros sitios proveedores de video.

Muestra la caracterización del video.

Otras

Tiene todas las personas catalogadas directores, actores, puede mostrar todos los trabajos de una persona (director, actor, etc.).

Proveedores de Video soportados

- NetFlix.
- Amazon.
- Hulu.
- LoveFilm [93].
- Blockbuster [94].

B.6.2 Análisis técnico

Interacción con fuentes de video:

Integración con NetFlix, permite excluir las películas que el usuario vio en Netflix, ver recomendaciones de Netflix y agregar a lista.

También se puede ver la información asociada a la película en otros sitios de recomendación como por ejemplo: IMDb y Rotten Tomatoes.

Protocolos:

SOAP / XML

Servicios (API):

Tiene un api como para poder incorporar SOAP API como para incorporar en distintos dispositivos pero no parece publica.

Plataformas soportadas

PS3.

Wii.

Xbox .

Ipad.

Android.

Windows Mobile.

Apple TV.

Google TV.

Web.

- Reproducción de avances con Flash.

B.6.3 Análisis Crítico

Características fuertes

- Puede unir las base de datos de las películas con otros recomendadores como IMDb y Rotten Tomatoes.
- En el sistema de recomendaciones existe una mixtura entre el juicio de expertos y una automatización y aprendizaje automático del perfil del usuario.
- Se integra con distintos proveedores de videos.
- Cubre las 3 pantallas de reproducción e incluye versiones para dispositivos variados.

Características débiles

Solo permite saber si el usuario vio la película en NetFlix, si lo hizo en otro proveedor, no lo toma en cuenta.

B.7 Metacritic

En Metracritic [95] encontramos recomendaciones para diverso tipo de contenido, esto nos interesa ya que hasta ahora hemos estudiado en general, películas. Además encontramos este sitio en la lista de los 5 mejores servicios de recomendaciones de películas según [72].

Metacritic es un sitio Web que recopila “reviews” de críticos y publicaciones reconocidas para albunes musicales, juegos, películas, programas de TV y DVDs. A partir de estas críticas el sitio genera calificaciones mediante el concepto propietario de metascoring (ver sección calificación del video para mayor detalle) el cual resume en un color y un único valor numérico (porcentaje) la calificación del contenido basándose en las críticas realizadas [85].

Provee además información de cada contenido y un extracto de cada crítica con un enlace a la crítica completa y al crítico o publicación de la que se obtuvo. En particular para las películas y programas de TV la información es recuperada desde IMDb, que a su vez muestra los metascoring y críticas desde metacritic (colaboran mutuamente).

B.7.1 Funcionalidades

Características sociales

- Página de perfil del usuario que muestra los comentarios y valoraciones realizadas por el usuario.
- Escribir críticas sobre los contenidos y avances.
- Unirse a conversaciones (comentarios) en la sección de reportes.
- *Portlet* para mostrar los 5 contenidos más vistos en todo momento.
- *Feeds* a través de RSS.

Integración con Redes sociales

Permite compartir enlaces a críticas de contenidos en las redes sociales:

- Facebook.
- Twitter.

Recomendaciones

- Si bien existe un portlet de “productos recomendados”, estas recomendaciones son genéricas y no son personalizadas para cada usuario.

Video

- Permite la reproducción de avances embebidos desde IVA (Internet Video Archive) [36].
- Permite la reproducción de avances embebidos desde Youtube [96].
- Permite ver las críticas de cada video.
- Muestra el metascoring de cada video (valoración promedio ponderada).
- Permite que los usuarios valoren los videos.

Proveedores de Video soportados

- IVA (Internet Video Archive).
- Youtube.

B.7.2 Análisis técnico

Calificación del video

Para cada contenido se obtiene un puntaje numérico ponderado de cada crítica, la ponderación de la crítica se basa en la reputación y calidad del crítico o publicadora que la realiza. Luego el total es promediado, obteniendo como resultado el “metascore” del contenido (concepto propietario de metacritic). A este metascore, que es un porcentaje, se le asigna un color que puede ser verde (favorable), amarillo (críticas variadas) o rojo (desfavorable) dependiendo de que tan alto sea el valor obtenido. El metascore resume en un color y un único valor numérico la calificación del contenido basándose en las críticas realizadas. Los usuarios también pueden valorar los contenidos pero solo a modo informativo para otros usuarios ya que estas valoraciones no son tenidas en cuenta para calcular el metascore de los contenidos.

Interacción con fuentes de video

- Incrustado de reproductor para visualización de avances desde IVA (Internet video archive). IVA es el mayor distribuidor en línea de avances de películas, juegos, TV y videos musicales
- Incrustado de reproductor para visualización de avances desde Youtube.

Protocolos

- Utiliza OData [97] para obtener información de los avances desde IVA (Internet video archive). A través de OData obtiene el ID del avances que luego es utilizado para embeber el reproductor.

Servicios a otras aplicaciones y API

Metacritic no cuenta con una API pública, a pesar de esto, muchos sitios realizan Web-scraping para mostrar metascores y críticas desde Metacritic.

Plataformas soportadas

Solo Web.

B.7.3 Análisis Crítico

Características fuertes

- Integración con sitios externos reconocidos como IMDb y IVA (Internet video archive).
- Soporta otros contenidos multimedia además de películas y TV como ser juegos y música.

Características débiles

- No tiene una API pública oficial para acceder a las críticas o a los metascores de los contenidos.
- No provee soporte para dispositivos móviles.
- No realiza recomendaciones personalizadas teniendo en cuenta gustos o preferencias del usuario.

Anexo C Tablas de Dispositivos

A continuación se especifican las características de los diferentes dispositivos estudiados según sus fabricantes.

C.1 La Televisión en todos lados

Atrás queda la época del telespectador pasivo sentado frente al televisor en el living de su casa observando lo que ha sido programado por su cable operador. Hoy día los individuos quieren ver los contenidos en cualquier momento y en cualquier lado. Ver el noticiero en el smartphone mientras se viaja en algún medio de transporte público, o ver una película en la cama con la tablet utilizando algún servicio de video como Netflix, Hulu, Amazon o Youtube son algunos de los ejemplos que podemos citar.[42]

C.2 La primera pantalla / pantalla principal

C.2.1 Dispositivos para la primera pantalla

Televisores conectados y Televisores inteligentes

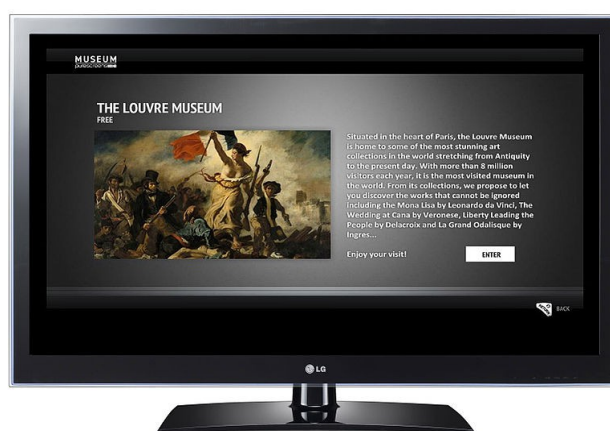


Illustration 28: Smart TV

Samsung (series 8000)

- Medidas (46'', 55'', 65'')
 - Alto: 663 mm (46''), 772 mm (55''), 912 mm (65'')
 - Ancho: 1090 mm (46''), 1283 mm (55''), 1519 mm (65'')
 - Espesor: 23 mm (46'', 55''), 30,5 mm (65'')
 - Diagonal: 1166 mm (46''), 1387 mm (55''), 1638 mm (65'')
- Peso: 17 kg (46''), 22 kg (55''), 36,1 kg (65'')
 - Resolución: 1920 x 1080 (46'', 55'', 65'') (full HD)

- Conectividad: Wi-Fi, Wi-Fi Direct, Wi-Fi Hotspot, DLNA y Ethernet
- Control
 - Mando a distancia
 - Mando a distancia virtual: “Samsung Qwerty Remote” para smartphones y tablets
 - Teclado inalámbrico
 - Control por gestos
 - Control por voz
 - Control de acceso a aplicaciones por reconocimiento facial
- Soporte Picture in Picture (PiP), permitiendo ver TV al mismo tiempo que se muestra una aplicación
- Distribución de aplicaciones: “Samsung Apps”
- Sistema operativo: versión “nativa” de Google Android, basada en HoneyComb (Samsung está trabajando en su S.O “Bada” como posible reemplazo)
- Navegador: Maple browser con soporte para Flash
- Formatos de video soportados: H.264, HD DivX MPEG1/MPEG2: DAT, MPG, MPEG, TS, TRP, VOB, MKV, DivX, AVI, MPEG4: MP4, MKV, DivX, AVI H.264: TS, TRP, MP4, MKV, DivX, AVI
- Desarrollo de aplicaciones:
 - Kit de desarrollo: Samsung TV SDK
 - Tipo de aplicaciones: Web basadas en Javascript o Flash con acceso a la API del SDK
 - Emulador: SI
 - UI: “10 foot user interface”, Pantalla completa, área específica o siempre al frente.

Sony Internet TV con Google TV, primera generación (series NSX-GT1)

- Medidas (24”, 32”, 40”, 46”)
 - Alto: 353 mm (24”), 481 mm (32”), 586 mm (40”), 661 mm (46”)
 - Ancho: 571 mm (24”), 766 mm (32”), 954 mm (40”), 1087 mm (46”)
 - Espesor: 71 mm (24”), 56 mm (32”), 53 mm (40”), 54 mm (46”)
 - Diagonal: 597 mm (24”), 800 mm (32”), 1016 mm (40”), 1168 mm (46”)
 - Peso: 8,7 kg (24”), 11,8 kg (32”), 17,1 kg (40”), 21 kg (46”)
 - Resolución: 1920 x 1080 (full HD)

- Conectividad: Wireless LAN : 802.11.a/b/g/n, DLNA y Ethernet
- Procesador: 1.6 GHz Intel® Atom
- Almacenamiento: 8 GB
- Control
 - Mando a distancia
 - Teclado Qwerty
 - Mouse óptico
 - Mando a distancia virtual con “Google TV Remote App”
 - Búsqueda por voz
- PiP: SI
- Distribución de aplicaciones:”Android Market”
- Sistema operativo / Middleware: Google TV (basado en Android)
- Navegador: Chrome con soporte para Flash
- Formatos de video soportados: H.264/MPEG-4 AVC
- Desarrollo de aplicaciones:
 - Kit de desarrollo: Android SDK, Google TV Add-On, Android Development Toolkit (ADT), Eclipse
 - Tipo de aplicaciones: Nativas para Android TV
 - Emulador: SI
 - UI: “10 foot user interface”

Sony Bravia Internet TV (series EX)

- Medidas (32”, 40”)
 - Alto: 482 mm (32”), 586 mm (40”)
 - Ancho: 755 mm (32”), 943 mm (40”)
 - Espesor: 42 mm (32”, 40”),
 - Diagonal: 800 mm (32”), 1016 mm (40”)
- Peso: 7,7 kg (32”), 11,2 kg (40”)
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi, Wi-Fi Direct, DLNA y Ethernet

- Control
 - Mando a distancia
 - Mando a distancia virtual: “Media Remote” para dispositivos iPhone / Android
- PiP: SI
- Distribución de aplicaciones: “Yahoo Connected TV Store”
- Sistema operativo / Middleware: Yahoo Connected TV Platform
- Navegador: Navegador Web nativo muy básico (no soporta reproducción de video embebido)
- Formatos de video soportados: MPEG-1, MPEG-2 PS, MPEG-2 TS, AVC (MPEG-4 part 10), MPEG-2 TS, AVC (MPEG-4 part 10), MP4, MPEG-4 Visual (MPEG-4 part 2), MP4, WMV (9/VC1), Asf
- Desarrollo de aplicaciones:
 - Kit de desarrollo: “Yahoo Connected TV App Development Kit” (incluye: Ubuntu 10.04 virtual machine + App Development Kit (ADK) Debian software package)
 - Tipo de aplicaciones: BRAVIA Internet Widgets powered by Yahoo Widgets (Web basadas en Javascript, CSS, XML)
 - Emulador: SÍ (incluido en el ADK)
 - UI: “10 foot user interface”

Panasonic Smart Viera (series WT50)

- Medidas (47”, 55”)
 - Alto: 630 mm (47”), 726 mm (55”)
 - Ancho: 1066 mm (47”), 1236 mm (55”)
 - Espesor: 27 mm (47”, 55”)
 - Diagonal: 1190 mm (47”), 1390 mm (55”)
 - Peso: 13 kg (47”), 17 kg (55”)
 - Resolución: 1920 x 1080 (full HD)
 - Conectividad: Wi-Fi, Bluetooth, DLNA y Ethernet
 - Control
 - Mando a distancia

- Mando a distancia virtual: “Viera Remote App” para iPhone, iPad, smartphones y tablets Android
- PiP: SI
- Distribución de aplicaciones: “Viera Connect Market”
- Sistema operativo / Middleware: “Viera Connect Platform”
- Navegador: Navegador Web Nativo con soporte de HTML5
- Formatos de video soportados: AVCHD 3D/Progressive, SD-VIDEO/MotionJPEG (Lumix)/DivXPlusHD/WMV/MKV/AVI/MP4/MOV, MP3/AAC/WMA Pro/FLAC, JPEG/MPO
- Desarrollo de aplicaciones:
 - Kit de desarrollo: No tiene. Cualquier editor de texto/Javascript puede ser utilizado.
 - Tipo de aplicaciones: Javascript, ejecutadas de forma nativa por el motor de Javascript de la plataforma (no navegador)
 - Emulador: No tiene. Existen especificaciones para montar un entorno de pruebas usando un servidor apache local el cual puede guardar líneas de logs lanzadas por las aplicaciones que se están probando mediante la instrucción *console.log(“algun_texto_de_debug”)*

Consolas de videojuegos

Xbox 360

- Fabricante: Microsoft
- Medidas
 - Alto: 183 mm
 - Ancho: 305 mm
 - Espesor: 292 mm
- Peso: 4,9 kg
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi 802.11a/b/g y Ethernet
- Procesador: 3.2 GHz (PowerPC) Tri-Core IBM Xenon
- Almacenamiento: Disco duro extraíble de 20, 60, 120, 250 o 320 GB
- RAM: 512 Mb
- Control

- Mando inalámbrico
- Teclado
- Control por gestos (a través de Kinect)
- Control por voz (a través de Kinect)
- Mando a distancia virtual: para smartphones con el S.O Windows Phone
- Distribución de aplicaciones: "Xbox Live Marketplace"
- Sistema operativo / Middleware: Sistema embebido diseñado especialmente para Xbox 360
- Navegador:
 - Actualmente no existe un navegador nativo, aunque es posible usar la navegación de alguna PC conectada a la red con Windows Media Center y que haya sido extendido mediante add-ons, por ejemplo con WCEBrowser [98].
 - Microsoft ha anunciado en la E3 2012 que para la segunda mitad del 2012 estará disponible una versión específica de Internet Explorer Xbox 360, con características de control mediante gestos por Kinect, voz y mediante Xbox Smartglass [99].
- Formatos de video soportados: WMV, H.264, MPEG-4 Parte 2 Advanced Simple Profile (ASP).
- Desarrollo de aplicaciones:
 - Kit de desarrollo: Xbox Development Kit (XDK), XNA Game Studio, Microsoft Visual Studio, SmartGlass SDK, Nine Engine, OpenXDK.
 - Tipo de aplicaciones: Juegos, Aplicaciones de consola
 - Emulador: No
 - Para depurar aplicaciones nativas es necesario adquirir un kit de hardware adicional para soportar depuración desde Visual Studio. Además se debe ser miembro del "Xbox 360 Registered Developers Program"[100]
 - En caso del desarrollo de juegos con XNA el registro se debe hacer en el "App hub" [101] que habilita entre otras cosas la depuración en las consolas Xbox 360 usando "XNA Game Studio Connect"

Playstation 3

- Fabricante: Sony
- Medidas
 - Alto: 99 mm

- Ancho: 325 mm
- Espesor: 274 mm
- Peso: 5 kg
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi IEEE 802.11 b/g, Bluetooth 2.0 (EDR), Ethernet
- Procesador: Cell Broadband Engine 3,2 GHz con 1PPE y 7 SPEs
- Almacenamiento: Disco duro SATA 2,5" de 20 GB, 40 GB, 60 GB, 80 GB, 120 GB, 160 GB, 250 GB, 320 GB
- RAM: 256 MB
- Control
 - Mando inalámbrico con sensor de movimiento integrado (Playstation Move)
 - Teclado
- Distribución de aplicaciones: "Playstation Store"
- Sistema operativo / Middleware: Sistema Operativo específico diseñado para Playstation 3. En versiones previas a la actualización 3.21 del firmware era también posible la instalación de Linux pero Sony ha retirado (por seguridad) a partir de 1 de Abril del 2010 la posibilidad de instalar otros sistemas operativos.
- Navegador: NetFront Browser (NX basado en webkit)
- Formatos de video soportados: MPEG-1, MPEG-2, H.264/MPEG-4 AVC, MPEG-4, DivX, XviD, y VC-1 (WMV)
- Desarrollo de aplicaciones:
 - Kit de desarrollo: PlayStation SDK. Sólo disponible para desarrolladores aprobados, actualmente de difícil acceso
 - Tipo de aplicaciones: Juegos, aplicaciones de consola
 - Emulador: No
 - Para depurar aplicaciones nativas es necesario adquirir un kit de hardware adicional para soportar depuración. Además se debe ser un desarrollador aprobado por Sony.

Wii

- Fabricante: Nintendo
- Medidas

- Alto: 44 mm
- Ancho: 157 mm
- Espesor: 215 mm
- Peso: 1,13 kg
- Resolución: 480p (PAL/NTSC), 480i (PAL/NTSC) y 576i (PAL/SECAM), pantalla panorámica anamórfica 4:3 y 16:9
- Conectividad: Wi-Fi IEEE 802.11b/g, Bluetooth y Compatible con adaptador de USB 2.0 a Ethernet LAN
- Procesador: PowerPC Broadway 729 MHz (fabricado por IBM)
- Almacenamiento: 512 Mb, ampliable a 32 GB con tarjetas SD y SDHC
- RAM: 88 MB (24 MB 1T-SRAM interna, 64 MB GDDR3 SDRAM externa)
- Control
 - Mando inalámbrico
- Distribución de aplicaciones: “Wii Shop Channel”
- Sistema operativo / Middleware: Sistema operativo propietario de Nintendo
- Navegador: “Canal Internet”, navegador desarrollado por Opera para Nintendo (soporte para Javascript y Flash)
- Formatos de video soportados:
- De forma nativa reproduce “Motion JPEG standard” en contenedores .mov y .avi desde los dispositivos de almacenamiento masivo.
- De forma nativa reproduce “Flash video” (.flv versiones 8 o anteriores) para videos de internet desde el explorador web incorporado.
- Instalando WiiMC (Wii Media Centre) es posible extender los videos soportados
 - Extensiones: 3gp, asf, avi, bik, divx, dpg, dv, evo, f4v, flc, fli, flv, iso, m1v, m2t, m2ts, m2v, m4v, mkv, mov, mp4, mp4v, mpe, mpeg, mpg, mqv, nsx, nuv, ogm, qt, raw, rmvb, swf, trp, ts, vdr, vfw, viv, vob, vro, wmv, y4m (resolución máxima permitida: 1280x720).
 - Codecs: MPEG-4 (típicamente XVID/DIVX .avi's) con resoluciones hasta 656x352, MPEG-2 PS (típicamente DVD rips y grabaciones TV-DVR .mpg's) con resoluciones hasta 720x480, MPEG-2 TS (típicamente grabaciones HDTV .ts's) con resoluciones hasta 1280x720
- Desarrollo de aplicaciones:

- Solo para desarrolladores y empresas autorizados por Nintendo. Se debe primero cumplir con los requisitos específicos. Las empresas ya deben haber desarrollado otros juegos en el pasado, y en el caso de desarrolladores individuales, deben tener experiencia en la industria del juego. Nintendo elige sólo las personas y empresas que creen que están calificados. Para comenzar el proceso de autorización se debe completar un formulario de solicitud [102]
- En caso que la empresa o desarrollador sean aprobados, la empresa recibirá por escrito las especificaciones de programación para las plataformas de Nintendo, y la posibilidad de comprar las herramientas de desarrollo de software para uso exclusivo en la ubicación física de la compañía.
- Los desarrolladores autorizados tendrán acceso a la página web de soporte de Nintendo para el Desarrollo de Software y recibir actualizaciones técnicas. Para tener acceso a la información de Nintendo será necesario firmar un acuerdo de confidencialidad.
- Kit de desarrollo de Wii: diseñado para compilar código, testear juegos, y garantizar una programación dentro de los límites de la consola. Los equipos van desde \$ 2.000 a \$ 10.000.
- Para la publicación se puede intentar contactar con alguna empresa con derechos de publicación como Ubisoft o Electronic Arts. En caso de no tener ese contacto, se puede presentar el juego en WiiWare (canal de descarga de juegos para la consola), en caso de ser aprobado por Nintendo se agregará para el próximo lanzamiento.

Set top boxes

Apple TV (tercera generación)

- Fabricante: Apple
- Medidas
 - Alto: 98 mm
 - Ancho: 98 mm
 - Espesor: 23 mm
- Peso: 270 gramos
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi 802.11a/b/g/n y Ethernet
- Procesador: Chip A5 de Apple con un núcleo 1 GHZ (diseñado por Apple, fabricado por Samsung)
- Almacenamiento: Oficialmente no tiene almacenamiento interno, sin embargo terceras partes han determinado que dispone de 8 GB de memoria flash para cache de contenidos.

- RAM: 512 MB
- Control
 - Mando a distancia Apple Remote
 - Mando a distancia virtual: App “Apple Remote” para iPhone, iPad, iPod
- Distribución de aplicaciones: N/A
- Sistema operativo / Middleware: iOS
- Navegador: No.
- Formatos de video soportados: H.264, MP4 (en formatos de archivos .m4v, .mp4, y .mov) y M-JPEG (1.280 por 720 píxeles, 30 fotogramas por segundo, en formato de archivo .avi)
- Desarrollo de aplicaciones: Actualmente no es posible desarrollar aplicaciones para Apple TV.
 - Kit de desarrollo: N/A
 - Tipo de aplicaciones: N/A
 - Emulador: N/A

Roku 2 XS

- Fabricante: Roku
- Medidas
 - Alto: 84 mm
 - Ancho: 84 mm
 - Espesor: 23 mm
- Peso: 85 gramos
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi 802.11 b/g/n, Ethernet y Bluetooth v3.0 (solo para “Roku Game Remote”)
- Procesador: Broadcom BCM2835 700MHz ARM11
- Almacenamiento: 256 MB memoria flash
- RAM: 256 MB
- Control
 - Mando a distancia (con sensor de movimiento para juegos)

- Distribución de aplicaciones: Roku Channel Store
- Sistema operativo / Middleware: Sistema Basado en Linux
- Navegador: No.
- Formatos de video soportados: H.264 (en formatos de archivo .mp4 y .mkv)
- Desarrollo de aplicaciones:
 - Kit de desarrollo: Roku SDK con Roku-BrightScript como lenguaje de programación (y NDK para juegos). También existe un plugin para Eclipse “The Eclipse BrightScript Plugin” que permite desarrollar y debuggear usando Eclipse como IDE con características como control y resalte de sintaxis.
 - Tipo de aplicaciones: Canales VOD gratuitos, por suscripción, canales de música, radio por Internet, intercambio de fotos, aplicaciones sociales, así como juegos casuales. Adicionalmente, Roku ha desarrollado una nueva API para juegos en 2D como parte del SDK 3.0. Los juegos que hacen uso de la GPU (procesador gráfico) y OpenGL en el Roku 2 necesitan ser desarrolladas usando el Kit de Desarrollo Nativo de Roku (NDK), permitiendo a los desarrolladores de juegos portar sus juegos existentes en lugar de desarrollarlos en Brightscript.
 - Emulador: No
 - Si bien no se dispone de un emulador, es posible debuggear conectando el IDE con algún dispositivo Roku en la red [103]
 - Otros:
 - Los canales Roku están escritos en un lenguaje específico llamado Roku-BrightScript que es un lenguaje de programación similar a VisualBasic y es aprendido rápidamente por los programadores experimentados.
 - La comunicación con los servicios y servidores se realiza a través de HTTP utilizando estándares basados en XML, como las tecnologías (M) RSS, APIs REST y JSON. Para el vídeo, se recomienda H.264 con audio AAC-LC envuelto en un contenedor MP4. Roku también es compatible con el códec de vídeo VC-1, y los codecs de audio WMA y MP3.
 - El registro como desarrollador es gratuito.
 - Todos los canales desarrollados requieren pasar el proceso de aceptación por parte de Roku.

Boxee Box

- Fabricante: D-Link

- Medidas
 - Alto: 114 mm
 - Ancho: 114 mm
 - Espesor: 114 mm
- Peso: 998 gramos
- Resolución: 1920 x 1080 (full HD)
- Conectividad: Wi-Fi 802.11 b/g/n y Ethernet
- Procesador: 1.2GHz Intel CE4100 Atom
- Almacenamiento: 1 GB memoria flash
- RAM: 1 GB
- Control
 - Mando a distancia doble cara (cara trasera incorpora teclado QWERTY)
 - Mando a distancia virtual: App para iPhone y Android. Con búsqueda por control por voz.
- Distribución de aplicaciones: Boxee Applications Directory
- Sistema operativo / Middleware: Sistema Basado en Linux
- Navegador: Navegador web nativo basado en Webkit con soporte para flash y HTML5
- Formatos de video soportados: Adobe Flash 10.1, FLV/On2 VP6 (FLV/FV4/M4V), H.264 AVC (TS/AVI/MKV/MOV/M2TS/MP4), VC-1 (TS/AVI/MKV/WMV), MPEG-1 (DAT/MPG/MPEG), MPEG-2 (MPG/MPEG/VOB/TS/TP/ISO/IFO), MPEG-4 (MP4/AVI/MOV), DivX 3/4/5/6 (AVI/MKV), Xvid (AVI/MKV), WMV9 (WMV/ASF/DVR-MS)
- Desarrollo de aplicaciones:
 - Cuenta con varios Kit de desarrollo.
 - En caso de desarrollar una aplicación nativa se utiliza Python como lenguaje de programación y un conjunto de APIs provistas por Boxee (Python API⁷, GUI/XML API⁸, Remote Control Interface API⁹). Se debe además, respetar una estructura de carpetas y archivos con formato XML bien definidos (anatomía de la aplicación) [103]. Como IDE es posible utilizar Notepad++, NetBeans, Eclipse, entre otros.

⁷ Python API http://developer.boxee.tv/Python_API

⁸ GUI/XML API http://developer.boxee.tv/Boxee_GUI_API

⁹ Remote Control Interface API http://developer.boxee.tv/Remote_Control_Interface

- En caso de desarrollar una aplicación web para el navegador de Boxee se puede utilizar cualquier IDE para desarrollo web y las APIs específicas de JavaScript e interacción con el navegador provistas por Boxee (Browser/JavaScript API¹⁰)
- Tipo de aplicaciones: Nativas o Web (haciendo uso del navegador nativo de Boxee)
- Emulador: No
- Si bien no se dispone de un emulador, es posible debuggear conectando un dispositivo USB con la aplicación deployada siguiendo la estructura de carpetas definida y bajo una carpeta raíz que se debe llamar “BoxeeApps”. De esta forma y especificando en el archivo “descriptor.xml” que la aplicación se debe ejecutar en modo debug se puede analizar el log de errores que va arrojando el sistema accediendo al dispositivo bajo su característica de servidor web [104]
- El registro como desarrollador es gratuito.
- Todas las aplicaciones desarrolladas requieren pasar el proceso de aceptación QA por parte de Boxee. Boxee revisará las solicitudes presentadas y una vez aprobada, será añadida al directorio de aplicaciones Boxee y una respuesta por correo electrónico es enviada al desarrollador.

C.3 La segunda pantalla

C.3.1 Dispositivos móviles

Tabletas

From Computer Desktop Encyclopedia
© 2012 The Computer Language Co. Inc.



Ilustración 29: Tabletas

¹⁰ Browser/JavaScript API http://developer.boxee.tv/JavaScript_API

A continuación presentamos un resumen de algunas de las principales tabletas en la actualidad:

- iPad tercera generación
- Samsung Galaxy Tab 10.1
- Sony Tablet S 3G
- Blackberry Playbook
- Amazon Kindle Fire
- Asus Eee Pad Transformer
- MS Surface

Características de Hardware

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
iPad tercera generación	Cortex-A9 Dual-core a 1GHz chipset Apple A5X	16, 32, 64 GB	1 GB	- 802.11a/b/g/n Wi-Fi - Bluetooth 4.0 + AD2P	Alto: 241,2 mm Ancho: 185.7 mm Espesor: 9.4 mm pantalla: 246.4 mm (diagonal)	652 gr	Multi-Touch Resolución: 2048 x 1536 px
Samsung Galaxy Tab 2 10.1	Dual-core 1GHz Cortex-A9	16, 32 GB	1GB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 900 / 1900 / 2100 - HSDPA, 21 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 a/b/g/n, Wi-Fi Direct, dual-band, Wi-Fi hotspot - Bluetooth 3.0 + AD2P	Alto: 175.3 mm Ancho: 256,7 mm Espesor: 9.7 mm pantalla: 256.5 mm (diagonal)	581 gr	Multi-Touch Resolución: 1280 x 800 px
Sony Tablet S 3G	Dual-core 1 GHz Cortex-A9 chipset Nvidia Tegra 2 T20	16, 32 GB	1GB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 900 / 2100	Alto: 174.3 mm Ancho: 241,2 mm Espesor: 10.1- mm - 20.6 mm	598 gr	Multi-Touch Resolución: 1280 x 800 px

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
				- HSDPA, 14.4 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 b/g/n, DLNA - Bluetooth 2.1 + AD2P	pantalla: 238 mm (diagonal)		
Blackberry Play-book	Dual-core 1 GHz Cortex-A9 chipset TI OMAP 4430	16, 32, 64 GB	1GB	- Wi-Fi 802.11 a/b/g/n - Bluetooth v2.1 + EDR	Alto: 130 mm Ancho: 194 mm Espesor: 10 mm pantalla: 177.8 mm (diagonal)	425 gr	Multi-Touch Resolución: 1024 x 600 px
Amazon Kindle Fire	Dual-core 1 GHz Cortex-A9 chipset TI OMAP 4430	8 GB	512 MB	- Wi-Fi 802.11 b/g/n	Alto: 190 mm Ancho: 120 mm Espesor: 11.4 mm pantalla: 177.8 mm (diagonal)	413 gr	Multi-Touch Resolución: 1024 x 600 px
Asus Eee Pad Transformer	Dual-core 1 GHz Cortex-A9 chipset Nvidia Tegra 2 T20	16, 32 GB	1GB	- Wi-Fi 802.11 b/g/n - Bluetooth 2.1 + EDR	Alto: 177 mm Ancho: 271 mm Espesor: 13 mm pantalla: 256.54 mm (diagonal)	680 gr	Multi-Touch Resolución: 1280 x 800 px
MS Surface Pro	Ivy Bridge Core i5 x86 dual core processor	64, 128 GB	-	- Wi-Fi 802.11 a/b/g/n - Bluetooth 4.0	Alto: 177 mm Ancho: 271 mm Espesor: 13.5 mm pantalla: 269.24 mm (diagonal)	903 gr	Multi-Touch Resolución: 1920 x 1080 px

Table 15: Tabletas - Características de Hardware

Características de Software

Modelo	SO	Desarrollo de Aplicaciones	Navegador	Soporte Audio y Video
iPad tercera generación	iOS 5.1 (multitarea)	- Desarrollo de aplicaciones usando	- Safari Mobile (motor de renderizado Webkit)	- H.264 con audio AAC-LC en formatos .m4v, .mp4,

Modelo	SO	Desarrollo de Aplicaciones	Navegador	Soporte Audio y Video
		Objective-C como lenguaje de programación. - IDE: XCode	- Soporte: HTML, HTML5	y .mov - MPEG-4 640 x 480 pixels, con audio AAC-LC en .m4v, .mp4, y mov - Motion JPEG (M-JPEG) 1280 x 720 pixels, con audio en ulaw, PCM stereo en .avi
Samsung Galaxy Tab 2 10.1	Android OS, v4.0.3 Ice Cream Sandwich (multitarea)	Desarrollo para Android (IV)	Navegadores para Android **** Soporte (en navegador predeterminado): HTML, HTML5, Flash	- Formatos de video soportados: MPEG4, H.263, H.264, DivX, WMV, 3GP, Flash Video, MKV - Formatos de audio soportados: MP3, AAC, FLAC, WMA, WAV, AMR, OGG, MIDI
Sony Tablet S 3G	Android OS, v3.2 Honeycomb, actualizable a v4.x (multitarea)	Desarrollo para Android (IV)	Navegadores para Android **** Soporte (en navegador predeterminado): HTML, XHTML, Flash	- Formatos de video soportados: MPEG4, H.263, H.264, WMV - Formatos de audio soportados: MP3, AAC, FLAC, WMA, WAV, OGG, MIDI
Blackberry Playbook	BlackBerry Tablet OS, actualizable a v2.0 (multitarea)	Desarrollo para Blackberry	- Navegador por defecto - Soporte (en navegador predeterminado): HTML, HTML5, Flash, JavaScript	- Formatos de video soportados: MPEG4, H.264, WMV - Formatos de audio soportados: MP3, AAC, WMA
Amazon Kindle Fire	Android OS, v2.3 (multitarea)	Desarrollo para Android (IV)	Amazon Silk (con motor de renderizado basado en WebKit, diseñado para hacer uso de la estructura Amazon Web Services (AWS)) Soporte: HTML	- Formatos de video soportados: MP4/VP8 - Formatos de audio soportados: MP3/WAV/AAC/OGG/MIDI
Asus Eee Pad Transformer	Android OS, v3.2 Honeycomb, actualizable a v4.x (multitarea)	Desarrollo para Android (IV)	Navegadores para Android **** Soporte (en navegador predeterminado): HTML, HTML5, XHTML, Flash	- Formatos de video soportados: H.263, H.264, MPEG 4 - Formatos de audio soportados: MP3
MS Surface Pro	Windows 8 Pro	Desarrollo para Windows 8 (III)	- Internet Explorer Mobile 10 - Soporte: HTML, HTML5, Flash	- Formatos de video soportados: - - Formatos de audio soportados: -

Table 16: Tabletas - Características de Software

Smartphones



Illustration 30: Smartphones

A continuación presentamos un resumen de algunos de los principales dispositivos con característica de smartphone en la actualidad (2010 o posterior).

- **Principales modelos basados en iOS de Apple**
 - **iPhone 4** de Apple, con **iOS4.0** como sistema inicial (2010)
 - **iPhone 4S** de Apple con **iOS5.0** como sistema inicial (2011)
- Nota:** Ambos modelos soportan hasta iOS6
- Principales modelos basados en Windows Phone de Microsoft
 - **HTC 7 Mozart** de HTC Corporation, con Windows Phone 7 OS7.0 como sistema inicial (2011)
 - **HTC 7 Pro** de HTC Corporation, con Windows Phone 7 OS7.0 como sistema inicial (2011)
 - **Nokia Lumia 710** de Nokia con Windows Phone 7.5 OS7.1 como sistema inicial (2011)
 - **Nokia Lumia 900** de Nokia con Windows Phone 7.5 OS7.1 como sistema inicial (2012)
- Principales modelos basados en Android de Google
 - **Samsung Nexus S** de Samsung Electronic con Android 2.3 como sistema inicial (2010)
 - **Galaxy Nexus** de Samsung Electronic con Android 4.0 como sistema inicial (2011)
 - **Samsung Galaxy S III** de Samsung Electronic con Android 4.0.4 como sistema inicial (2012)
- Principales modelos basados en Symbian
 - **Nokia N8** de Nokia, con Symbian^3 como sistema inicial (2010)

- **Nokia N9** de Nokia, con MeeGo OS, v1.2 Harmattan como sistema inicial (2011)
- Principales modelos basados en BlackBerry
 - **BlackBerry Torch 9800** de RIM (*Research In Motion*) con BlackBerry OS6 (2010)
 - **BlackBerry Bold 9790** de RIM (Research In Motion) con BlackBerry OS7 (2011)

Características de Hardware

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
iPhone 4	ARM Cortex A8 a 1.0 GHz	8, 16, 32 GB	512 MB	UMTS/HSDPA/HSUPA (850, 900, 1900, 2100 MHz) - GSM/EDGE (850, 900, 1800, 1900 MHz) - CDMA model: CDMA EV-DO Rev. A (800, 1900 MHz) - 802.11b/g/n Wi-Fi (802.11n 2.4GHz only) - Bluetooth 2.1 + EDR wireless technology	Alto: 115,2 mm Ancho: 58.6 mm Espesor: 9.3 mm pantalla: 89.6 (diagonal)	137 gr	Multi-Touch Resolución: 960 x 640 px
iPhone 4S	800 MHz dual-core Apple A5	16, 32, 64 GB	512 MB	- World phone - UMTS/HSDPA/HSUPA (850, 900, 1900, 2100 MHz) - GSM/EDGE (850, 900, 1800, 1900 MHz) - CDMA EV-DO Rev. A (800, 1900 MHz) - 802.11b/g/n Wi-Fi (802.11n 2.4GHz only) - Bluetooth 4.0 wireless technology	idem. iPhone 4	140 gr	idem. iPhone 4
HTC 7 Mozart	1 GHz Qualcomm QSD8250 Scorpion	8 GB	576 MB	- HSDPA/WCDMA: 900/2100 MHz - GSM 850/900/1800/1900 MHz - HSUPA: 2Mbps - GPRS: 114 kbps baja- da	Alto: 119 mm Ancho: 60.2 mm Espesor: 11.9 mm pantalla: 94 mm (diagonal)	130 gr	Multi-Touch Resolución: 480 x 800 px

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
				- EDGE: 560 kbps baja - Wi-Fi®: IEEE 802.11 b/g/n - Bluetooth® 2.1 con A2DP	nal)		
HTC 7 Pro	idem. HTC 7 Mozart	idem. HTC 7 Mozart	idem. HTC 7 Mozart	idem. HTC 7 Mozart	Alto: 117 mm Ancho: 59 mm Espesor: 15.5 mm pantalla: 91.4 mm (diagonal)	185 gr	idem. HTC 7 Mozart
Nokia Lumia 710	1.4 GHz Qualcomm MSM8255 Scorpion	8 GB	512 MB	- GSM 850/900/1800/1900 - WCDMA 900/1900/2100 - HSPA Cat6 5.76 Mbps - WLAN IEEE 802.11 b/g/n - GPRS Class B - HSDPA Cat10 14.4 Mbps - Dual Transfer Mode (MSC 32) - EDGE Class B - Bluetooth® 2.1 con A2DP, EDR	Alto: 119 mm Ancho: 62.4 mm Espesor: 12.5 mm pantalla: 94 mm (diagonal)	126 gr	Multi-Touch Resolución: 480 x 800 px
Nokia Lumia 900	1.4 GHz Qualcomm APQ8055	16 GB	512 MB	- GSM 850/900/1800/1900 - WCDMA 900/1900/2100 - HSPA Cat6 5.76 Mbps - WLAN IEEE 802.11 b/g/n - GPRS Class B - WCDMA	Alto: 127.8 mm Ancho: 68.5 mm Espesor: 11.5 mm pantalla: 94 mm (diagonal)	160 gr	Multi-Touch Resolución: 480 x 800 px

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
				- Dual Transfer Mode (MSC 11) - EDGE Class B - HSPA+ Cat14 21 Mbps - Diversity antenna array to maximize RF performance - HSPA+ Dual Carrier 42 Mbps - Bluetooth® 2.1 con A2DP, EDR			
Samsung Nexus S	1 GHz ARM Cortex-A8 Samsung Hummingbird S5PC110	16 GB	512 MB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 900 / 1700 / 2100 - HSDPA, 7.2 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 b/g/n, DLNA, Wi-Fi hotspot - Near Field Communication (NFC) - Bluetooth® 2.1 con A2DP	Alto: 123.9 mm Ancho: 63 mm Espesor: 10.9 mm pantalla: 101.6 mm (diagonal)	129 gr	Multi-Touch Resolución: 480 x 800 px
Galaxy Nexus	1.2 GHz dual-core Texas Instruments OMAP4460 ARM Cortex-A9	16 o 32 GB	1 GB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 900 / 1700 / 1900 / 2100 - HSDPA, 21 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 a/b/g/n, dual-band, DLNA, Wi-Fi hotspot - Near Field Communication (NFC) - Bluetooth® 3.0 con A2DP	Alto: 135.5 mm Ancho: 67.9 mm Espesor: 8.9 mm pantalla: 118.1 mm (diagonal)	135 gr	Multi-Touch Resolución: 720 x 1280 px
Samsung Galaxy S III	1.4 GHz quad-core Exynos	16, 32, 64 GB + hasta 64 GB micro SD	1 GB	- GSM 850/900/1800/1900 - HSDPA 850/900/1900 /2100 - LTE (según región)	Alto: 136.6 mm Ancho: 70.6 mm Espesor: 8.6 mm	133 gr	Multi-Touch Resolución: 720 x 1280 px

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
				-HSDPA, 21 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 a/b/g/n, DLNA, Wi-Fi Direct, Wi-Fi hotspot - Near Field Communi- cation (NFC) - Bluetooth® 4.0 con A2DP, EDR	mm pantalla: 121.9 mm (diagonal)		
Nokia N8	680 MHz ARM11 pro- cessor (ARM v6 architectu- re)	16 GB	256 MB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 900 / 1700 / 1900 / 2100 - HSDPA, 10.2 Mbps; HSUPA, 2.0 Mbps - Wi-Fi 802.11 b/g/n, UPnP technology - Bluetooth® 3.0 con A2DP	Alto: 113.5 mm Ancho: 59 mm Espesor: 12.9 mm pantalla: 88.9 mm (diagonal)	135 gr	Multi-Touch Resolución: 360 x 640 px
Nokia N9	1 GHz ARM Cortex-A8 (Texas Ins- truments OMAP3630)	16 o 64 GB	1 GB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 900 / 1700 / 1900 / 2100 - HSDPA, 14.4 Mbps; HSUPA, 5.7 Mbps - Wi-Fi 802.11 a/b/g/n, Wi-Fi hotspot - Bluetooth® 2.1 con A2DP, EDR	Alto: 116.45 mm Ancho: 61.2 mm Espesor: 12.1 mm pantalla: 99.1 mm (diagonal)	135 gr	Multi-Touch Resolución: 480 x 854 px
BlackBe- rry Torch 9800	624 MHz Marvell Ta- vor PXA940 (ARM Cortex A8) com- pliant	4 GB	512 MB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 1900 / 2100 / 800 - HSDPA, 14.4 Mbps; HSUPA, 5.7 Mbps - Wi-Fi 802.11 b/g/n, UMA (carrier-depen- dent) - Bluetooth® 2.1 con A2DP	Alto: 111 mm Ancho: 62 mm Espesor: 14.6 mm pantalla: 80.77 mm (diagonal)	161 gr	Multi-Touch Resolución: 360 x 480 px
BlackBe- rry Bold 9790	1 1 GHz Mar- vel Tavor MG	8 GB	768 MB	- GSM 850 / 900 / 1800 / 1900 - HSDPA 850 / 1900 /	Alto: 110 mm Ancho: 60	107 gr	Multi-Touch

Modelo	CPU	Storage	RAM	Red y Conectividad	Medidas	Peso	Pantalla
				2100 / 800 - HSDPA 900 / 1700 / 2100 - HSDPA, 7.2 Mbps; HSUPA, 5.76 Mbps - Wi-Fi 802.11 a/b/g/n, UMA (carrier-dependent) - Near Field Communication (NFC) - Bluetooth® 2.1 con A2DP, EDR	mm Espesor: 11.4 mm pantalla: 62.23 mm (diagonal)		Resolución: 360 x 480 px

Tabla 17: Smartphones - Características de Hardware

Características de Software

Modelo	SO	Desarrollo de Aplicaciones	Navegador	Soporte Audio y Video
iPhone 4	iOS4.0 (multitarea)	- Desarrollo de aplicaciones usando Objective-C como lenguaje de programación. - IDE: XCode	- Safari Mobile (motor de renderizado Webkit) - Soporte: HTML, HTML5	- H.264 con audio AAC-LC en formatos .m4v, .mp4, y .mov - MPEG-4 640 x 480 pixels, con audio AAC-LC en .m4v, .mp4, y mov - Motion JPEG (M-JPEG) 1280 x 720 pixels, con audio en ulaw, PCM stereo en .avi
iPhone 4S	iOS5.0 (multitarea)	idem. iPhone 4	idem. iPhone 4	idem. iPhone 4
HTC 7 Mozart	WP7 (multitarea (II))	Desarrollo para Windows Phone (I)	Internet Explorer Mobile 7 (motor de renderizado basado en un híbrido entre las versiones desktop de IE7 - IE8)	- Formatos de video soportados: 3gp, .3g2, .mp4, .m4v, .mbr, .wmv (Windows Media Video 9 and VC-1) - Formatos de audio soportados: .m4a, .m4b, .mp3, .wma (Windows Media Audio 9)
HTC 7 Pro	WP7 (multitarea (II))	Desarrollo para Windows Phone (I)	idem. HTC 7 Mozart	idem. HTC 7 Mozart

Modelo	SO	Desarrollo de Aplicaciones	Navegador	Soporte Audio y Video
Nokia Lumia 710	WP7.5 OS7.1 (multitarea (II))	Desarrollo para Windows Phone (I)	Internet Explorer Mobile 9 (motor de renderizado basado en la versión desktop de IE9)	- Formatos de video soportados: WMV 9, H.264/AVC, MPEG-4, 3GPP formats (H.263), AVI, ASF - Formatos de audio soportados: AAC, AAC+, MP3, MP4, WAV, WMA
Nokia Lumia 900	WP7.5 OS7.1 (multitarea (II))	Desarrollo para Windows Phone (I)	idem. Nokia Lumia 710	- Formatos de video soportados: WMV 9, H.264/AVC, MPEG-4, 3GPP formats (H.263), AVI, VC-1, ASF - Formatos de audio soportados: AAC, AAC+, MP3, MP4, WAV, WMA
Samsung Nexus S	Android 2.3 Gingerbread (multitarea)	Desarrollo para Android (IV)	Navegadores para Android (V) Soporte (en navegador predeterminado): HTML, XHTML, Flash	- Formatos de video soportados: MP4 / DivX / WMV / H.264 / H.263 - Formatos de audio soportados: MP3 / WAV / eAAC+ / AC3 / FLAC
Galaxy Nexus	Android 4.0 ICE CREAM SANDWICH (multitarea)	Desarrollo para Android (IV)	Navegadores para Android (V) Soporte (en navegador predeterminado): HTML, HTML5, Flash	- Formatos de video soportados: MP4, H.264, H.263 - Formatos de audio soportados: MP3 / WAV / eAAC+ / AC3
Samsung Galaxy S III	Android IceCream Sandwich 4,0,4 (multitarea) + Touchwiz 4,0	Desarrollo para Android (IV)	Navegadores para Android (V) Soporte (en navegador predeterminado): HTML, HTML5, Flash	- Formatos de video soportados: MP4 / DivX / XviD / WMV / H.264 / H.263 - Formatos de audio soportados: MP3 / WAV / eAAC+ / AC3 / FLAC / OGG
Nokia N8	Symbian^3	Symbian^3 SDK / QT SDK *****	Nokia Browser 7.2 (motor de renderizado WebKit 525), soporte para: HTML 4.1, XHTML, MP, WML, CSS, Flash Lite 4.0	- Formatos de video soportados: DivX/XviD/MP4/H.264/H.263/WMV player - Formatos de audio soportados: MP3 / WMA / WAV /

Modelo	SO	Desarrollo de Aplicaciones	Navegador	Soporte Audio y Video
				eAAC
Nokia N9	MeeGo OS, v1.2 Harmattan	QT SDK *****	Browser for MeeGo (motor de renderizado WebKit2), soporte para: HTML5, CSS3	- Formatos de video soportados: WMV 9 / H.264/AVC / MPEG-4 / XVID / Matroska / 3GPP formats (H.263) / AVI / VC-1 / Flash Video / ASF - Formatos de audio soportados: FLAC / Ogg Vorbis / AAC / AAC+ / AMR-WB / HE AAC V1 / HE AAC V2 / MP3 / M4A / WAV / WMA
BlackBerry Torch 9800	BlackBerry OS 6.0	Desarrollo para BlackBerry	Torch browser (inicialmente llamado "Iris Browser" de propiedad de Torch Mobile, empresa que luego adquirió BlackBerry, con motor de renderizado basado en WebKit)	- Formatos de video soportados: MPEG4, H.263, H.264, WMV3 - Formatos de audio soportados: MP3, AMR-NB, AAC-LC, AAC+, eAAC+, WMA, WMV, FLAC, Ogg Vorbis
BlackBerry Bold 9790	BlackBerry OS 7.0	Desarrollo para BlackBerry	Bold Web Browser (basado en Webkit)	- Formatos de video soportados: DivX / XviD / MP4 / H.264 / H.263 / WMV - Formatos de audio soportados: MP3 / WMA / WAV / eAAC+ / FLAC

Tabla 18: Smartphones - Características de Software

(I) Desarrollo para Windows Phone

Se utiliza el Windows Phone SDK con Silverlight out-of-browser, C# o visual basic .net como lenguaje de programación y XAML como lenguaje de marcado para la UI. Los IDEs disponibles son el Microsoft Visual Studio (programación) y Microsoft Expression Blend (Diseño gráfico).

El SDK incluye el Windows Phone Emulador permitiendo el desarrollo y pruebas en un ambiente emulado sin necesidad del dispositivo físico real.

Además se tiene la posibilidad de agregar complementos como el "Silverlight for Windows Phone Toolkit" el cual ofrece controles adicionales para el desarrollo de aplicaciones para Windows Phone.

(II) Multitarea en Windows Phone

Si bien se lo puede considerar un sistema multitarea, esta característica esta limitada. WP7.5 sigue ejecutando solamente una aplicación en primer plano a la vez, pero agrega la posibilidad de crear servicios que ejecutan en segundo plano.

(III) Desarrollo para Windows 8

Se cuenta con varias herramientas que facilitan el desarrollo, entre ellas:

- Microsoft Visual Studio 2012 Express RC
- Windows 8 SDK
- Blend for Visual Studio
- Project templates

Se puede optar por escribir aplicaciones al estilo Windows Metro, usar tecnologías web (HTML5, CSS3, JavaScript), usar código nativo (C++), código manejado (C#, Visual Basic); aplicaciones de escritorio que usan el modelo de programación nativo (Win32/COM); o aplicaciones de escritorio que usan el framework .NET

(IV) Desarrollo para android

Existen tres modalidades de desarrollo para Android:

- **Android SDK:** Es la forma de desarrollo más utilizada. Incluye debugger, librerías, emulador, documentación, códigos de ejemplo y tutoriales. Se utiliza Java como lenguaje de programación. El IDE oficial soportado es Eclipse para el cual existe el plugin “Android Development Tools” (ADT).
- **Desarrollo nativo:** En ocasiones es utilizado para desarrollar alguna tarea crítica en performance. El “native development kit” de Android (NDK) es un conjunto de herramientas que permiten embeber componentes escritos en C y C++ que hacen uso de código nativo (ARM o x86) en las aplicaciones Android.
- **App Inventor:** Se comprende de un IDE basado en web orientado a programadores que recién se inician, incluso para personas que no saben nada de programación. El desarrollo se realiza de manera visual, componiendo controles en las diferentes pantallas y agregandoles lógica para su comportamiento también de manera visual en bloques preconfigurados para las estructuras de datos y de control mayormente utilizadas en cualquier lenguaje de programación.

(V) Navegadores para Android

El sistema operativo Android posee un navegador por defecto de serie, pero existen varias opciones de navegadores que compiten por ser el navegador preferido en este sistema operativo. Los principales son: Boat, Opera, Dolphin HD, Firefox, Chrome, Miren, Skyfire.

(VI) Symbian^3 SDK y QT SDK

Symbian^3 SDK permite el desarrollo de aplicaciones usando Symbian C++, Qt, Standard C/C++, Java ME y Web Runtime para dispositivos basados en la plataforma Symbian. Incluye documentation, emulator etc. excluyendo IDEs [105].

Qt SDK es un framework (no es un runtime) que reduce el esfuerzo de codificación a través del uso de intuitivas APIs sin comprometer la performance de las aplicaciones, además elimina la dependencia con los SDKs específicos para cada dispositivo. Permite crear interfaces de usuario de manera sencilla y rápida de forma declarativa usando Qt Quick y Qt Quick Components. El SDK está disponible para Windows, Linux y Mac. [106].

Anexo D Pautas de diseño para interfaces de usuario

En las siguientes secciones analizaremos los principales aspectos que debemos considerar durante la etapa de desarrollo para las interfaces de usuario de primera pantalla. Comenzaremos resumiendo las pautas generales de diseño para luego centraremos en las pautas específicas para la televisión como primera pantalla.

D.1 Pautas generales

Según *Ivan Serrano Regol*, en su trabajo de investigación *Directrices para el diseño de interfaces de televisión digital interactiva* [108] presentado en el *WebcatBCN* de Febrero de 2012, la percepción visual, la teoría del color, los patrones de visualización y los aspectos culturales son algunos de los aspectos generales que se deben tener en cuenta a la hora de diseñar cualquier tipo de interfaz de usuario, a continuación desarrollamos cada uno de estos aspectos.

D.1.1 Percepción visual

La percepción es la elaboración, interpretación, análisis e integración de los estímulos captados por nuestros órganos de los sentidos. Es un proceso cognitivo básico complejo superior por el que somos capaces de elaborar, interpretar, analizar e integrar una sensación que captan nuestros sentidos. La percepción trata de simplificar estímulos, principalmente los más complejos que nos proporciona el ambiente, a los que estamos constantemente expuestos porque si no tardaríamos muchísimo en tratar de discriminar todos los estímulos a los que estamos expuestos [109]. Las leyes de la Gestalt [110] son un conjunto de principios descriptivos sobre el proceso de la percepción visual, coherente con la premisa gestalista de que toda nuestra experiencia perceptiva está estructurada y posee una tendencia a unir y a relacionar en grupos los estímulos percibidos. Estos principios se estructuran básicamente en dos leyes: “Ley de la figura-fondo” y “Ley de la buena forma” o de la ‘Pregnancia’” (llamada también Ley del agrupamiento).

Ley de figura-fondo[110]

Según la Ley de la figura-fondo, el proceso perceptivo remite a un mecanismo básico según el cual tendemos a focalizar nuestra atención sobre un objeto o determinado grupo de objetos (figura) destacándolos del resto de los objetos que los envuelven (fondo).

Ley de la buena forma

- **Ley de semejanza[108]**

En igualdad de las demás condiciones, tendemos a percibir como parte de una misma estructura u objeto los elementos semejantes. Por lo tanto se recomienda agrupar juntos los elementos que visualmente son similares.

- **Ley de proximidad[108]**

En igualdad de las demás condiciones, tendemos a percibir como juntos (o como un mismo objeto) los elementos más próximos en el espacio o en el tiempo. Por lo tanto se recomienda agrupar los elementos que pertenecen a un mismo grupo muy cerca.

- **Ley de simetría o regularidad[108]**

En igualdad de circunstancias, tendemos a percibir como unidad aquellos elementos que presentan el mayor grado de simplicidad, simetría, regularidad y estabilidad. Se recomienda la organización simétrica ya que las asimétricas causa que los elementos se pierdan en la pantalla.

- **Ley de continuidad o buena continuidad[108]**

En igualdad de circunstancias, tendemos a percibir como parte de una misma figura los estímulos que guardan entre sí una continuidad de forma. La percepción busca equilibrio y por lo tanto debemos alcanzar que las formas sean estables y completas.

- **Ley de la simplicidad[108]**

En igualdad de condiciones tendemos a percibir como unidad los elementos que presentan el mayor grado de simplicidad, simetría y regularidad. Por lo tanto se recomienda diseñar estructuras que sean lo más simples posibles.

- **Ley de la experiencia[108]**

La experiencia previa del sujeto receptor interviene junto con los aspectos mencionados anteriormente en la construcción de las formas percibidas.

- **Ley de cierre[110]**

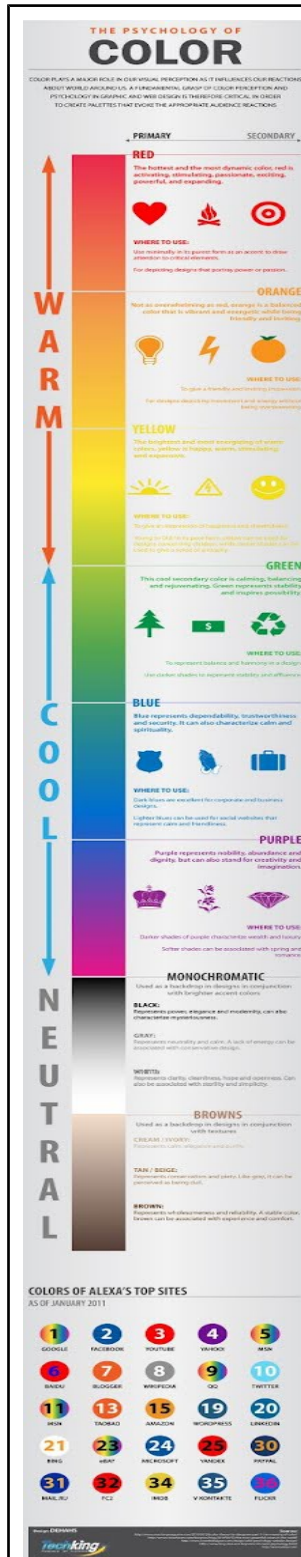
Nuestra mente añade los elementos faltantes para completar una figura. Perceptivamente, tendemos a completar y llenar aquellas partes y vacíos que se nos presentan incompletas, de tal manera que nos permite la captación del objeto en su totalidad.

- **Ley de la comunidad (también llamado principio de dirección común)[110]**

Muchos elementos que parecen construir un patrón o un flujo en la misma dirección son percibidos como un único elemento.

D.1.2 Teoría del Color

Los colores juegan un papel importante en la estética y funcionalidad, además de jugar un papel en la orientación, estructura y clarificar diferencias entre los elementos visuales, también pueden facilitar el acceso a la información [108].



testking.com, “La psicología del color” [111]

Rojo (primario):

El más caliente y el color más dinámico, el rojo es activo, estimulante, apasionado, poderoso emocionalmente, y expansión. Se recomienda usarlo para llamar la atención sobre elementos críticos.

Naranja (secundario):

No es tan abrumador como el rojo, el naranja es un color balanceado, vibrante, energético, agradable y acogedor. Se recomienda usarlo para brindar sensación de agrado y en diseños que representen movimiento y energía.

Amarillo (primario):

El más brillante y energizante entre los colores cálidos. El amarillo es felicidad, calidez, estimulante y expansivo. Se recomienda usarlo para brindar sensación de felicidad y alegría o para diseño para niños, mientras que los tonos más oscuros pueden ser usados para mostrar una sensación de antigüedad.

Verde (secundario):

Color secundario que es calma, balance y rejuvenecimiento. Representa estabilidad e inspira posibilidad. Se recomienda usarlo para representar balance y armonía. Usar las tonalidades más oscuras para representar estabilidad y riqueza.

Azul (primario):

Representa seriedad, confiabilidad y seguridad. También puede caracterizar a la calma y a la espiritualidad. Las tonalidades oscuras son ideales para diseños corporativos y de negocios. Las más claras pueden ser utilizadas para sitios web sociales que representan calma y amabilidad.

Púrpura (secundario):

Representa nobleza, abundancia y dignidad, pero también puede significar creatividad e imaginación. Las tonalidades oscuras de púrpura representan riqueza y lujo, mientras que las más claras pueden asociarse a la primavera y al romance.

	Colores monocromáticos Se utilizan como fondo en los diseños en combinación con los colores de acento más brillantes. Negro: Representa el poder, la elegancia y la modernidad, también se puede caracterizar misterio. Gris: Representa la neutralidad y la calma. La falta de energía puede estar asociada con un diseño conservador. Blanco: Representa la claridad, la limpieza, la esperanza y la apertura. También puede estar asociado con la esterilidad y la simplicidad. Marrones Se utilizan como fondo en los diseños en combinación con texturas. Crema / Marfil: Representa tranquilidad, elegancia, y la pureza. Tan / Beige: Representa el conservadurismo y la piedad. Al igual que el gris, que puede ser percibida como aburrida. Marrón: Representa la salubridad y fiabilidad. Un color estable. Puede estar asociado con la experiencia y comodidad.
--	---

D.1.3 Patrones de visualización

La investigación indica que los espectadores exploran las pantallas de televisión de la parte superior izquierda hasta la inferior del lado derecho. Debido a los hábitos de lectura, este comportamiento es más fuerte en las interfaces de iTV con mucho texto que en la televisión convencional [112].

D.1.4 Aspectos culturales

La cultura de una región o país puede hacer percibir un mismo objeto de maneras diferentes en puntos geográficos diferentes [113]. Para realizar una interfaz multicultural se debe tener en cuenta todo lo relacionado con características concretas de cada país o región (lenguaje, sistemas

de medida, iconografía). Todos estos factores contribuyen a que una misma interfaz sea entendida de la misma manera para todo el mundo, dejando de lado posibles confusiones que puedan surgir y conducir al abandono de la interfaz por parte del usuario.

D.2 Pautas de diseño para la primera pantalla.

En particular analizaremos las interfaces de usuario para la televisión como primera pantalla. Dado que el referente actual de medio interactivo es la web, es probable que las interfaces desarrolladas para la iTV se asemejen a las de Internet. Según *Ivan Serrano Regol*, en su trabajo de investigación *Directrices para el diseño de interfaces de televisión digital interactiva* [108] presentado en el *WebcatBCN* de Febrero de 2012, los puntos más relevantes en el diseño de interfaces Web que pueden ayudar al diseño de interfaces de iTV son:

- Composición visualmente agradable.
- Organizar los elementos en pantalla.
- La navegabilidad, diseñada para ofrecer caminos claros desde el origen al destino, sin terminología compleja y con la misión de orientar al usuario.
- Uso correcto de la tipografía y su legibilidad en pantalla.
- Ha de ser rápido en todas sus vertientes, no sólo técnicamente, sino desde el punto de vista de la interacción.
- Debemos crear una jerarquía visual clara que se repita en cada página, dividiendo las diferentes zonas de la misma de forma definida y constante.
- Debemos diseñar teniendo en cuenta que existen diferentes tipos de resoluciones y configuraciones de pantalla
- La consistencia, tanto a nivel de contenidos como del formato visual así como la disposición de elementos en las páginas, ofreciendo entornos homogéneos que ayuden a potenciar una comunicación efectiva del mensaje y, a su vez, ayuden al usuario a formarse un modelo mental del sitio.
- Para obtener un sistema de navegación eficiente, hay que controlar el número de clics que debe hacer el usuario antes de llegar a la información. Cuantos más niveles de profundidad más compleja será nuestra estructura y, por lo tanto, más difícil de entender por parte del usuario.
- A pesar de que organicemos adecuadamente la información, es fundamental ofrecer al usuario un sistema alternativo al de la navegación para encontrar la información, es decir, necesitamos ofrecer un motor de búsqueda que sea eficaz y mantenga una excelente indexación.

En cuanto a la interfaz de una aplicación de iTV, la misma es el aspecto visual del diseño en la pantalla del Televisor. La Interfaz es la primera y última parte de la aplicación que el observador detecta. Un buen diseño gráfico ayuda a hacer un servicio emocionalmente atractivo y

estéticamente agradable, ayudando a cómo una aplicación se debe utilizar [108]. Las interfaces de usuario para TV exitosas son simples tanto en concepto como en diseño [114]. Comúnmente, al entorno de visualización para la televisión se lo conoce como ambiente de 10 pies y a la pantalla de televisión como la interfaz de usuario de 10 pies. Al crear aplicaciones para un entorno de 10 pies, vale la pena recordar los siguientes conceptos básicos [114]:

- El ambiente de 10 pies es tradicionalmente para consumir contenidos.
 - El entorno de 10 pies es un entorno de diversión y no en un ambiente de trabajo.
 - El ambiente de 10 pies suele ser un entorno social, no un entorno de usuario único.
- La experiencia visual de una interfaz de usuario de 10 pies es una mezcla de computadora y televisor.
 - Las pantallas de televisión tienen características similares tanto a una computadora como a una TV.
 - La resolución de la pantalla es similar al computador, pero se ve afectada por las características de la TV.
 - El color es diferente en las pantallas de televisión.
- Los televisores tienen sonido de alta calidad.
 - Los televisores suelen estar conectados a buenos sistema de sonido.
 - A diferencia del computador, los usuarios aceptan el sonido de un televisor de buena manera y no se molestan por ello.
- Una interfaz de usuario de 10 pies requiere de herramientas de navegación sencillas y visibles.
 - Diseñar la navegación de forma tal de avanzar de izquierda a derecha y de arriba a abajo.
 - Limitar la necesidad de un ratón.
 - Brindar retroalimentación visual.
- Una interfaz de 10 pies funciona mejor si es diseñada con patrones basados en zonas de contenido.
 - Las zonas son independientes de los tipos de dispositivos.
 - La disposición de las zonas son particulares a cada tipo de dispositivos.
 - Muchas interfaces diferentes puede basarse en el concepto de zona.

Algunas directrices sugeridas por Ismael Serrano para interfaces de TV [108]:

- Simplicidad en el diseño. Un diseño de interfaz iTV debe ser fácil de entender, independientemente de la experiencia, el conocimiento, o el nivel de concentración actual de los usuarios.
- Los realces no deben competir con el contenido principal. El propósito de estos elementos no es dominar el proceso de interacción, sino presentar un entorno temático y visual que el usuario puede entender con el fin de integrar la información adicional.
- Las interfaces deben tener un tema consistente. Mantener la continuidad visual es importante para mantener a los usuarios involucrados, así como hacerles saber lo que puede esperar de la aplicación.
- Crear una estructura de la pantalla efectiva. Una aplicación para iTV debe mantener los elementos textuales lo mínimo posible. Además, el texto que se divide en trozos es más fácil de leer desde la distancia.
- Seguir el modelo sobre los estándares existentes en el diseño para televisión. Hay que considerar la experiencia que se tiene en diseñar para televisión y colocar atención a los colores y tipos de letra apropiados para la TV.
- Menús simples e intuitivos. Los menús deben ser simples con mensajes cortos e intuitivos.

Otras directrices recomendadas en la documentación de Google TV:

- Utilizar un teléfono móvil como modelo para la interfaz de usuario [115]. El tamaño de un televisor moderno es engañoso. A pesar de que los televisores modernos son generalmente más grandes que 40" en diagonal, los espectadores se sientan proporcionalmente más lejos de ellos que de un monitor de computadora, la percepción del espectador es que la televisión es más pequeña que un monitor. Para simular esta experiencia, utilizar un teléfono móvil como modelo a la hora de diseñar la interfaz de usuario.
- Para evitar una apariencia desordenada en la pantalla, dejar más espacio en blanco entre los elementos de la página [115]. Para ello, usar una combinación de márgenes mayores y más relleno. Este también es un buen consejo para interfaces táctiles.
- Los televisores están siempre en modo horizontal [115]. En un televisor, el espacio disponible va de izquierda a derecha y no de arriba a abajo. Poner los controles de navegación del lado derecho o izquierdo de la pantalla y ahorrar espacio vertical para el contenido
- Las pantallas de televisión tienen más contraste y niveles de saturación que los monitores de computadora[108], por ello cuando se trabaja con colores sólidos se recomienda:

- Usar con moderación el color blanco puro (# FFFFFFFF). Los blancos puros provocar vibración o imagen fantasma. En su lugar utilizar # F1F1F1 (hexagonal) o 240/240/240 (RGB).
- Evitar los blancos, rojos y naranjas, debido a que causan una distorsión particularmente mala.
- Tener en cuenta los diferentes modos de presentación que los televisores pueden tener, incluyendo Estándar, Vivo, Cine/Teatro, Juegos, etc. Asegurarse de probar la aplicación en todos estos modos.
- Evitar el uso de grandes gradientes, ya que pueden dar lugar a bandas.
- En lo posible, probar la aplicación en pantallas de baja calidad con una pobre gama de colores y ajustes.
- Evitar las fuentes ligeras o las fuentes que tienen trazos muy estrechos o muy amplios. Utilizar fuentes sencillas como sans-serif y uso de *anti-aliasing* para aumentar la legibilidad [115]. Algunas estrategias para mejorar la lectura del texto son:
 - Limitar cada párrafo a no más de 90 palabras.
 - Romper el texto en trozos pequeños que los usuarios pueden procesar.
 - Mantener las longitudes de línea de alrededor de 5 a 7 palabras por línea. No menos de 3 o más de 12.
 - Los textos claros sobre fondo oscuro son más fácil de leer, en comparación con texto oscuro sobre fondo claro.
 - Dejar más espacio vertical entre líneas para el texto en pantalla de lo que sería para el texto impreso.
- El sonido no es perjudicial en la televisión, como puede serlo en la computadora, por lo tanto se recomienda usarlos en las interfaces para la TV [116]. Tener en cuenta las siguientes pautas:
 - Usar sonidos que sean apropiados para un entorno de sala de estar.
 - Por defecto a un volumen bajo.
 - Suponer que los usuarios estarán viendo la televisión o escuchando música mientras utilizan la aplicación. Brindar una forma fácil de silenciar la aplicación (mute). Evitar que las interacciones dependan por completo de las señales de audio, y utilizar las señales con moderación.
- Los usuarios pueden encontrar dificultosa la navegación de las aplicaciones [117]. Estas son algunas buenas prácticas:

- En caso de contar con un dispositivo “D-pad” utilizarlo en vez de navegar con el ratón. Los usuarios estarán probablemente más familiarizados con este tipo de navegación al estar acostumbrados a usar el control remoto.
- Si se decide navegar usando el ratón, utilizar superficies extra grandes de selección para el diseño de la interfaz de usuario. Hacer los botones seleccionables grandes, para que los usuarios puedan posicionar fácilmente el puntero sobre ellos.
- Evitar que la navegación con el ratón sea compleja o tenga que ser precisa. No utilizar funcionalidades del tipo “arrastrar y soltar” o menús desplegables. Estos son muy difíciles de controlar en el contexto de la televisión.
- Proveer comentarios interfaz de usuario muy obvio para la navegación. Expandir una superficie de selección cuando recibe el foco, o resaltarlo. Utilice las transiciones y los movimientos que proporcionan señales visuales para proceder o invertir la acción. Por ejemplo, si el usuario selecciona una acción que provoca una nueva página para deslizarse por la derecha, que intuitivamente puede entender que al presionar D-Pad Izquierda va a volver a la página anterior.
- Brindar una retroalimentación obvia de la navegación. Por ejemplo si con la flecha derecha pasamos a la página siguiente, el usuario intuitivamente va a entender que con la flecha izquierda volverá a la página previa.
- Proveer un esquema donde el usuario pueda aprender rápidamente a predecir cómo moverse en la aplicación.
- La navegación en las interfaces de TV pueden resultar nueva a muchos de los usuarios, por ello, se debe tratar de brindar mecanismos de ayuda y asistencia a la navegación. Algunos aspectos a considerar son:
 - En cuanto a las flechas del dispositivo D-pad:
 - ¿Se mueven entre páginas?
 - ¿Pueden abrir menús de contexto o navegación?
 - En cuanto al botón “atrás”:
 - ¿Se mueve a una página previa?
 - ¿Puede deshacer una acción?
 - ¿Cierra una ventana *pop-up*?
 - Teclas “Media”:
 - ¿Que hacen las teclas de reproducir y pausa?
 - ¿Que hacen las teclas de retroceder y avanzar?
 - Otras teclas:

- ¿Hay teclas para cancelar o para cerrar un *pop-up*?, por ejemplo, ¿la tecla ESC cierra una ventana de diálogo de tipo *pop-up*?

Posicionar el cursor para seleccionar puede ser inexacto [117], por ello para ayudar al usuario a hacer selecciones se sugiere:

- Resaltar el control de la interfaz seleccionado cuando el puntero este por encima
- Hacer que los controles de selección sean grandes y proporcionar el padding adicional suficiente alrededor de las etiquetas de texto.
- Evitar el uso de clics del ratón para cerrar los cuadros de diálogo. En las aplicaciones de computadora a menudo los diálogos se cierran cuando el usuario hace clic en otro lugar, lo que no es obvio en el contexto de la TV, a una distancia de 10 pies. En su lugar, proporcionar un control explícito para cerrar el cuadro de diálogo, y asegurarse de que el D-pad de navegación puede acceder a él.
- No obligar al usuario a seleccionar el control primario para activarlo. Activar el control primario o el por defecto (dándole el foco) y resaltarlo para indicar esto.
- Si una parte de la ventana o cuadro de diálogo requiere el desplazamiento, que tenga el foco, o permitir que se desplace sin tener el foco. Requerir que el usuario haga clic en un elemento antes de desplazarse es molesto.
- Si parte de una ventana o diálogo requiere usar el *scroll*, es recomendable darle el foco o permitir que se pueda usar el *scroll* sin necesidad de tener el foco. Obligar a que el usuario tenga que hacer clic en algún elemento antes de usar el *scroll* es obstrusivo.
- Las transiciones en la aplicación pueden enseñar o confundir al usuario. Hay que asegurarse de que las transiciones tengan un significado en lugar de simplemente entretener [103]. Las siguientes son algunas consideraciones:
 - Evitar recargar toda la página.
 - Para cualquier acción que afecte sólo a un fragmento, mantener la transición por completo dentro de ese fragmento.
 - Hacer que los mensajes de progreso/avance tengan significado y sean específicos. Las imágenes animadas deben indicar cuánto se ha avanzado, y cuánto trabajo queda por hacer.
 - No mostrar una animación de "carga" para cada elemento de una grilla. Hacer esto para cada elemento resulta en una interfaz desordenada que los usuarios encuentran difícil de ver.
- Mantener el diseño y comportamiento consistente [118]

- A una distancia de 10 pies los elementos que no son visibles son olvidados, por ello es recomendable mantener todas las acciones y opciones importantes visibles [118], incluyendo búsquedas, configuraciones, botón “atrás”, etc.
- Identificar las partes vitales de la interfaz antes de empezar a trabajar [105].
 - Agrupar el contenido, controles, y las interacciones por prioridad.
 - Desechar todos los elementos no esenciales.
- Usar un modo visible de navegación o una jerarquía de información. [119]
 - Guiar a los usuarios hacia un modelo "mental" que funcione para toda la interfaz.
 - Confiar en los hábitos que los usuarios han adquirido del uso de los navegadores o aplicaciones.
- Hacer que la acción principal sea alcanzable en un solo clic. [119]
 - Destacar el resto de las acciones en pantalla. No esconder las características clave de un menú.
 - Utilizar sólo algunas funciones clave.
 - Mostrar siempre una manera fácil para que los usuarios regresen a su ubicación anterior. No confiar en el botón Atrás.
- Anticiparse y seleccionar la próxima acción del usuario siempre que sea posible. [119]
 - Por ejemplo, mover el cursor a la siguiente caja de texto, o seleccionar el primer ítem de una lista.
- Evitar utilizar iconos abstractos. [119]
 - Utilizar etiquetas claras y concisas y ponerlas a prueba con los usuarios.
- Limitar el scroll vertical.[119]
 - Pensar en cómo se escalará el contenido cuando aumenta de tamaño. ¿Qué pasa si se convierte en una lista diez veces más grande?
 - En caso de tener que hacer scroll, asegurarse de que una parte de la sección más baja está siempre visible en pantalla.

Anexo E MovieRecommenderApp – documentación de API

E.1 Generalidades

E.1.1 Diseño RESTful

REST define un conjunto de principios arquitectónicos por los cuales se puede diseñar servicios web que se centran en los recursos del sistema, incluyendo cómo los estados para estos recursos son direccionados y transferidos sobre HTTP.

Una implementación concreta de un servicio Web REST sigue cuatro principios básicos de diseño:

- Utiliza los métodos HTTP explícitamente (GET, POST, PUT, DELETE, etc.).
- No mantiene estado.
- Expone las URIs con una estructura similar a la de directorios en un sistema de archivos.
- Transfiere XML, JavaScript Object Notation (JSON), o ambos.

REST establece una correspondencia uno-a-uno entre crear, leer, actualizar y eliminar (CRUD) y los métodos HTTP de acuerdo con este mapeo:

- Para crear un recurso en el servidor, utilizar POST.
- Para recuperar un recurso, utilizar GET.
- Para cambiar el estado de un recurso o para actualizar, utilizar PUT.
- Para quitar o eliminar un recurso, utilizar DELETE.

E.1.2 Formato para las peticiones

El formato de la petición está definido por el header HTTP “Content-Type”, el cual es obligatorio para los métodos HTTP que incluyen cuerpo en el mensaje. En otro caso este dato no es requerido. El formato específico soportado por la API es:

Content-Type: application/x-www-form-urlencoded

E.1.3 Formato de la respuesta

El formato de la respuesta depende del tipo de datos aceptado por el cliente. Para conocer el tipo de datos aceptado el sistema consulta el header “Accept” de la petición HTTP.

Los valores posibles aceptados por la API son:

- application/**json**; charset=utf-8
- application/**xml**; charset=utf-8

E.1.4 Tipo de datos para la respuesta

La respuesta (sin importar el formato) siempre representa un objeto con las siguientes 3 propiedades:

- **Value** => Respuestas que devuelven un valor, se guarda en esta propiedad.
- **Status** => Indica si se pudo procesar o no la solicitud. Valores posibles: "ok", "error".
- **Message** => Si Status es "error" entonces aquí se guarda un mensaje descriptivo del error.

E.1.5 Formato de fechas

Las fechas siguen el formato “aaaaMMddhhmmss”, donde “aaaa” son 4 dígitos para el año, “MM” dos dígitos para el mes, “dd” dos dígitos para el día, “hh” dos dígitos para las horas (sistema de 24 horas), “mm” dos dígitos para los minutos y “ss” dos dígitos para los segundos. Para números menores a 10 igual se deben usar los dos dígitos, por ejemplo, indicar 08 en vez de simplemente 8.

Ejemplo: Para representar el momento del día correspondiente a la 1PM con 5 minutos y 15 segundos del día 15 de diciembre del año 2012 se debe indicar así: “20121215130515”

E.1.6 Autenticación y Autorización

La gran mayoría de las operaciones de la API requieren de haber iniciado sesión previamente para poder ejecutarlas.

Como HTTP es un protocolo sin estado, se sigue el esquema utilizado usualmente para mantener el estado de la sesión del usuario mediante cookies encriptadas de sesión y autorización.

Por simplicidad, en esta documentación, en todos los ejemplos que se muestran se obvia el header “Cookie” de la petición HTTP que ejecuta la invocación a la operación de la API, pero se

debe recordar que el mismo es requerido para poder ejecutar todas las operaciones que tienen como pre-requisito haber iniciado sesión previamente.

Un ejemplo de uso del header “Cookie” para mantener la sesión y datos de autorización del usuario es:

```
Cookie: _ASP.NET_SessionId=wj33xqn4wsocjybl5ufrmeok;  
.ASPXAUTH=07905CC9EA4C1BE91C8C4BC8EA058F4D1A612FAEDFB2E04CAC607BBD  
372C1B19A1DEA8A4D5BC0E3ED89CD8F2D5B52A223261D7AD5E2DB8AE834AF3B2261  
B14B997738F195917173BE9D45DDA7A3FD0EE091F3FA42A0E3A830DCCEE7926D80741  
465B764D21DB8DB66DDF6F402076622E456466DB8F95CB16C73A2C75AA78D4E3
```

Es importante mencionar que todas aquellas operaciones que solicitan un identificador de usuario para el que se desea obtener información o realizar un cambio de estado en el sistema (relacionado al usuario), tienen autorización para utilizarla con otro identificador de usuario del que se ha iniciado sesión solo aquellos usuarios con rol administrador, es decir que un usuario ordinario solo podrá ejecutar las operaciones de la API en su nombre y nunca en nombre de otro usuario.

Por ejemplo, si un usuario ordinario (supongamos que tiene identificador 123) ejecuta la operación de la API para obtener la preferencia de gusto de un usuario sobre una entrada de cartelera, a través en la URI “/api/users/71568/event-cinemalistingsentry/4/200/likes” recibirá en la respuesta un objeto JSON de la forma {"Value":403,"Status":"error","Message":"El usuario no esta autorizado"}

E.2 Operaciones

E.2.1 Crear un evento

Descripción: Crea un evento con un usuario como propietario y miembro inicial.

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: Se creó un nuevo evento en el sistema con los datos especificados. Además se registró al usuario que ha ejecutado la operación como propietario del evento y se lo asignó automáticamente como miembro en el grupo de usuarios del evento.

Uri: /api/events/

Método: POST

Parámetros de entrada:

- **name:** Nombre del evento
- **description:** Descripción del evento
- **type:** Tipo de evento (valores posibles “CINEMA” o “CUSTOM”)
- **utcStartDatetime:** Fecha y hora de inicio del evento en UTC
- **utcEndDatetime:** Fecha y hora de fin del evento en UTC
- **idOwner:** Identificador del usuario propietario para el evento

Valor de retorno:

Devuelve el identificador del evento creado

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST http://localhost:2605/api/events/ HTTP/1.1

Host: localhost:2605

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Content-Length: 137

name=ev1&description=Primer+evento+desde+Fiddler2&type=CINEMA&UtcStartDatetime=20121208200000&UtcEndDatetime=20121208235959&idOwner=40000

Ejemplo de respuesta:

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Thu, 06 Dec 2012 03:55:17 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private
Content-Type: application/json; charset=utf-8
Content-Length: 38
Connection: Close

```
{"Value":5,"Status":"ok","Message":""}
```

E.2.2 Obtener el detalle de un evento

Descripción: Devuelve los datos de un evento: nombre, descripción, tipo, rango de fechas en que se desarrolla el evento, usuarios que se han unido al evento, invitaciones pendientes, las películas candidatas dentro de ese rango de fechas, las recomendadas hasta ahora, etc.

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A

Uri: /api/events/{idEvent}/users/{idUser}

Método: GET

Parámetros de entrada:

- **idEvent:** Identificador del evento solicitado
- **idUser:** Identificador del usuario para el que se solicita la información del evento

Valor de retorno:

Detalle del evento solicitado

Ejemplo de petición: (no se incluye el header cookie con los parámetros encriptados de autenticación, se solicita el evento con id=1):

GET http://localhost:2605/api/events/1 HTTP/1.1

Host: localhost:2605

Ejemplo de respuesta:

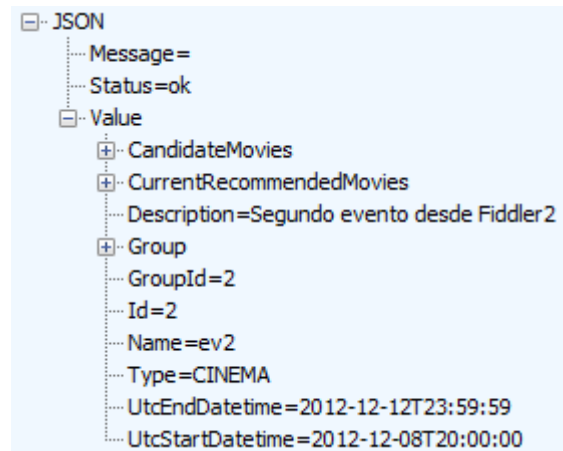


Ilustración 31: Ejemplo de respuesta al obtener detalles de un Evento

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Wed, 12 Dec 2012 03:50:00 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 3170

Connection: Close

```
{
  "Value": {
    "CandidateMovies": [
      {
        "Cinema": {
          "$id": "1",
          "Address": "Luis Alberto de Herrera 1290",
          "Airconditioning": true,
          "Id": 1,
          "Name": "MovieCenter Montevideo Shopping",
          "OfficialSite": "http://www.moviecenter.com.uy/",
          "Parking": true,
          "ScreenType": "grande",
          "Seats": null,
          "Telephone": "2900 3900",
          "ExtendedProperties": [
            {
              "Key": "UserLikes",
              "Value": true
            }
          ],
          "CinemaId": 1,
          "DateTime": "2012-12-10T00:00:00",
          "Id": 3,
          "Movie": {
            "$id": "2",
            "IdItem": 1,
            "Title": "Toy Story (1995)",
            "MovieIdItem": 1
          },
          "Cinema": {
            "$ref": "1"
          },
          "ExtendedProperties": [
            {
              "Key": "UserLikes",
              "Value": false
            }
          ],
          "CinemaId": 1,
          "DateTime": "2012-12-11T13:00:00",
          "Id": 4,
          "Movie": {
            "$id": "3",
            "IdItem": 2,
            "Title": "Jumanji (1995)",
            "MovieIdItem": 2
          },
          "Cinema": {
            "$id": "4",
            "Address": "Avenida Italia 5775",
            "Airconditioning": true,
            "Id": 2,
            "Name": "MovieCenter Portones Shopping",
            "OfficialSite": "http://www.moviecenter.com.uy/",
            "Parking": true,
            "ScreenType": "grande",
            "Seats": null,
            "Telephone": "2900 3900",
            "ExtendedProperties": [
              {
                "Key": "UserLikes",
                "Value": true
              }
            ],
            "CinemaId": 2,
            "DateTime": "2012-12-"
          }
        }
      }
    ]
  }
}
```

```

09T00:00:00","Id":5,"Movie":{"$ref":"2"},"MovieIdItem":1},{ "Cinema":{"$ref":"4
"ExtendedProperties":[{"Key":"UserLikes","Value":null}], "CinemaId":2,"DateTime":"2012-12-
09T13:00:00","Id":6,"Movie":{"$ref":"3"},"MovieIdItem":2},{ "Cinema":{"$ref":"4"},
"ExtendedProperties":[{"Key":"UserLikes","Value":true}], "CinemaId":2,"DateTime":"2012-12-
10T00:00:00","Id":7,"Movie":{"$id":"5","IdItem":3,"Title":"Grumpier Old Men
(1995)","MovieIdItem":3}], "CurrentRecommendedMovies":[{"Cinema":
{"$id":"6","Address":"Luis Alberto de Herrera
1290","Airconditioning":true,"Id":1,"Name":"MovieCenter Montevideo
Shopping","OficialSite":"http://www.moviecenter.com.uy/","Parking":true,"ScreenType":"grande
","Seats":null,"Telephone":"2900 3900"},
"ExtendedProperties":
[{"Key":"UserLikes","Value":null}], "CinemaId":1,"DateTime":"2012-12-
10T00:00:00","Id":3,"Movie":{"$id":"7","IdItem":1,"Title":"Toy Story
(1995)","MovieIdItem":1},{ "Cinema":{"$id":"8","Address":"Avenida Italia
5775","Airconditioning":true,"Id":2,"Name":"MovieCenter Portones
Shopping","OficialSite":"http://www.moviecenter.com.uy/","Parking":true,"ScreenType":"grande
","Seats":null,"Telephone":"2900 3900"},
"ExtendedProperties":
[{"Key":"UserLikes","Value":null}], "CinemaId":2,"DateTime":"2012-12-
09T00:00:00","Id":5,"Movie":{"$ref":"7"},"MovieIdItem":1},{ "Cinema":{"$ref":"8"},
"ExtendedProperties":[{"Key":"UserLikes","Value":false}], "CinemaId":2,"DateTime":"2012-12-
09T13:00:00","Id":6,"Movie":{"$id":"9","IdItem":2,"Title":"Jumanji
(1995)","MovieIdItem":2},{ "Cinema":{"$ref":"8"},
"ExtendedProperties":
[{"Key":"UserLikes","Value":true}], "CinemaId":2,"DateTime":"2012-12-
10T00:00:00","Id":7,"Movie":{"$id":"10","IdItem":3,"Title":"Grumpier Old Men
(1995)","MovieIdItem":3}], "Description":"Segundo evento desde Fiddler2",
"ExtendedProperties":[{"Key":"UserHasRated","Value":"true"}],
"Group":
{"UtcEndDate":"2012-12-08T23:59:59","Guid":"48200f90-b0ae-4242-b476-
a02fce0fa07d","Id":2,"Members":
[{"$id":"11","Email":"unknown@domain.com","FirstName":"Unknown","Id":40000,"JoinedGro
ups":[{"UtcEndDate":"2012-12-08T23:59:59","Guid":"48200f90-b0ae-4242-b476-
a02fce0fa07d","Id":2,"Members":[{"$ref":"11"}], "Owner":
{"$ref":"11"}, "OwnerId":40000,"PendingInvitations":
[]}], "LastName":"Unknown","Male":true,"Ocupation":null,"OwnedGroups":
[{"UtcEndDate":"2012-12-08T23:59:59","Guid":"48200f90-b0ae-4242-b476-
a02fce0fa07d","Id":2,"Members":[{"$ref":"11"}], "Owner":
{"$ref":"11"}, "OwnerId":40000,"PendingInvitations":[]}], "PendingInvitationReceived":
[], "PendingInvitationSent":[]], "Owner":{"$ref":"11"}, "OwnerId":40000,"PendingInvitations":
[]}, "GroupId":2,"Id":2,"Name":"ev2","Type":"CINEMA","UtcEndDatetime":"2012-12-
12T23:59:59","UtcStartDatetime":"2012-12-08T20:00:00","Status":"ok","Message":""}

```

E.2.3 Invitar a un usuario a un evento

Descripción: Envía una invitación para que un usuario se una a los miembros del grupo para un evento.

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: Se creó una invitación para el evento y usuario indicados. El usuario invitado podrá consultar los detalles de la invitación al ver los eventos con invitaciones pendientes de aceptar.

Uri: /api/events/invitation/create

Método: POST

Parámetros de entrada:

- **idUserFrom:** Identificador del usuario que envía la petición
- **emailTo:** email del usuario que va a recibir la invitación
- **idEvent:** identificador del evento al que se está realizando la invitación

Valor de retorno:

Devuelve el código de la invitación que se ha creado

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST http://localhost:2605/api/events/invitation/create HTTP/1.1

Host: localhost:2605

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Content-Length: 52

idUserFrom=40000&emailTo=waljoel@gmail.com&idEvent=1

Ejemplo de respuesta:

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Wed, 12 Dec 2012 04:21:12 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 75

Connection: Close

```
{"Value":"795f692c-ae64-4b42-a8dc-cead69c460e1","Status":"ok","Message":""}
```

E.2.4 Obtener el detalle de una invitación a un evento

Descripción: Obtiene el detalle de una invitación pendiente para que un usuario se una a los miembros para un evento

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario ha sido previamente invitado al evento

Post-Condiciones: N/A

Uri: /api/events/{idEvent}/users/{idUserTo}/invitation/

Método: GET

Parámetros de entrada

- **idEvent:** Identificador del evento para la invitación
- **idUserTo:** Identificador del usuario invitado

Valor de retorno:

Devuelve el detalle de la invitación

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET /api/events/11/users/40001/invitation/ HTTP/1.1

Host: localhost:2605

Ejemplo de respuesta:

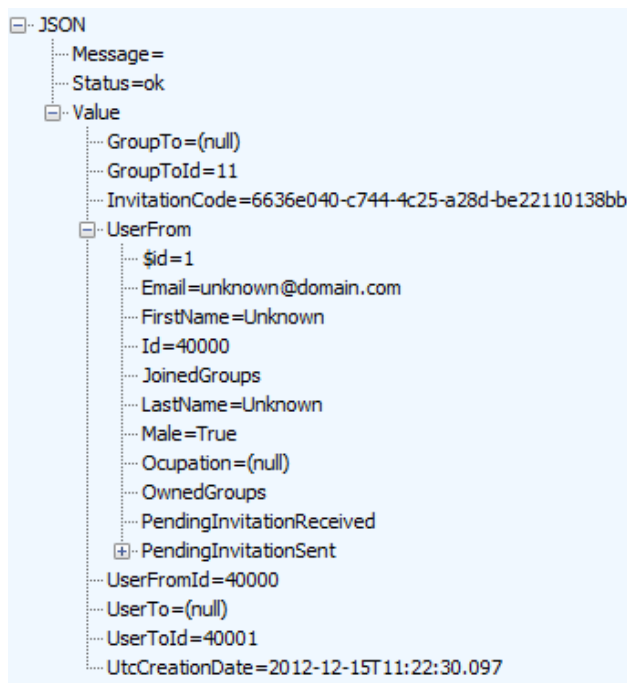


Ilustración 32: Ejemplo de respuesta al invitar un usuario a un Evento

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Tue, 18 Dec 2012 02:40:22 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 651

Connection: Close

```
{"Value":{"GroupTo":null,"GroupToId":11,"InvitationCode":"6636e040-c744-4c25-a28d-be22110138bb","UserFrom":{"$id":"1","Email":"unknown@domain.com","FirstName":"Unknown","Id":40000,"JoinedGroups":[],"LastName":"Unknown","Male":true,"Occupation":null,"OwnedGroups":[],"PendingInvitationReceived":[],"PendingInvitationSent":[{"GroupTo":null,"GroupToId":11,"InvitationCode":"6636e040-c744-4c25-a28d-be22110138bb","UserFrom":{"$ref":"1"},"UserFromId":40000,"UserTo":null,"UserToId":40001,"UtcCreationDate":"2012-12-15T11:22:30.097"}]},"UserFromId":40000,"UserTo":null,"UserToId":40001,"UtcCreationDate":"2012-12-15T11:22:30.097"},"Status":"ok","Message":""}
```

E.2.5 Obtener el detalle de una invitación a un grupo

Descripción: Obtiene el detalle de una invitación pendiente para que un usuario se una a un grupo. Para el caso de consultar el detalle de una invitación a un grupo efimero perteneciente a un evento es equivalente a utilizar la operación “Obtener el detalle de una invitación a un evento” para el evento al que pertenece el grupo.

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario ha sido previamente invitado al grupo

Post-Condiciones: N/A

Uri: api/groups/invitation/user/{idUser}/group/{idGroup}

Método: GET

Parámetros de entrada

- **idUser:** Identificador del usuario invitado

- **idGroup**: Identificador del grupo para el evento invitado

Valor de retorno:

Devuelve el detalle de la invitación

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET /api/groups/invitation/user/40001/group/11

Host: localhost:2605

Ejemplo de respuesta:

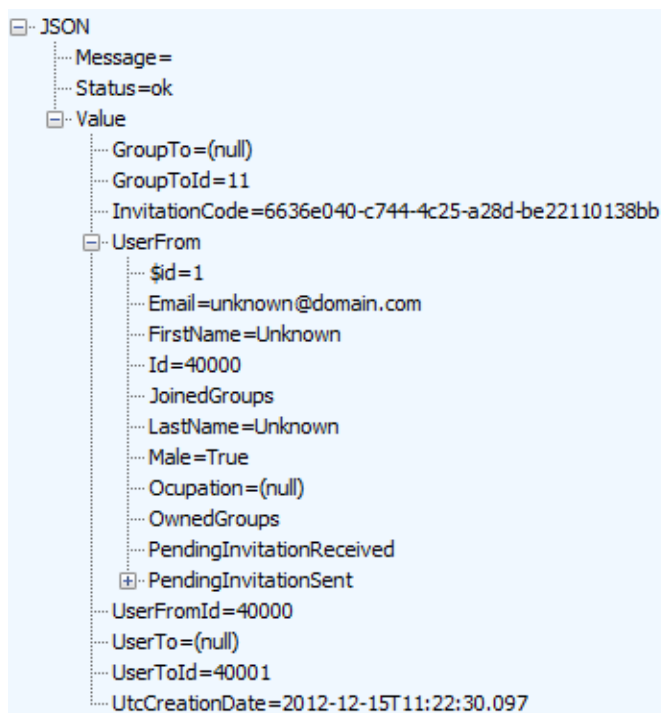


Ilustración 33: Ejemplo de respuesta de pedir detalles de la invitación a un grupo

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Tue, 18 Dec 2012 03:02:11 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 651

Connection: Close

```
{"Value":{"GroupTo":null,"GroupToId":11,"InvitationCode":"6636e040-c744-4c25-a28d-be22110138bb","UserFrom":{"$id":"1","Email":"unknown@domain.com","FirstName":"Unknown","Id":40000,"JoinedGroups":[],"LastName":"Unknown","Male":true,"Occupation":null,"OwnedGroups":[],"PendingInvitationReceived":[],"PendingInvitationSent":[{"GroupTo":null,"GroupToId":11,"InvitationCode":"6636e040-c744-4c25-a28d-be22110138bb","UserFrom":{"$ref":"1"},"UserFromId":40000,"UserTo":null,"UserToId":40001,"UtcCreationDate":"2012-12-15T11:22:30.097"}]},"UserFromId":40000,"UserTo":null,"UserToId":40001,"UtcCreationDate":"2012-12-15T11:22:30.097"},"Status":"ok","Message":""}
```

E.2.6 Aceptar unirse a un evento

Descripción: Acepta la invitación pendiente para que un usuario se una a los miembros del grupo para un evento

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario ha sido previamente invitado al evento

Post-Condiciones: Se agregó al usuario al conjunto de usuarios miembros para el evento y el sistema actualizó el conjunto de películas a recomendar para el evento teniendo en cuenta la nueva incorporación. Además, se eliminó la invitación para el usuario en el evento y la invitación ya no le aparecerá al usuario al ver los eventos con invitaciones pendientes de aceptar.

Uri: api/events/invitation/accept

Método: POST

Parámetros de entrada:

- **invitationCode:** Código de la invitación

Valor de retorno: N/A

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST http://localhost:2605/api/events/invitation/accept HTTP/1.1

Host: localhost:2605

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Content-Length: 51

invitationCode=795f692c-ae64-4b42-a8dc-cead69c460e1

Ejemplo de respuesta:

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Wed, 12 Dec 2012 04:36:40 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 39

Connection: Close

{"Value":"","Status":"ok","Message":""}

E.2.7 Rechazar unirse a un evento

Descripción: Rechaza la invitación pendiente para que un usuario se una a los miembros del grupo para un evento

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario ha sido previamente invitado al evento

Post-Condiciones: Se eliminó la invitación para el usuario en el evento. La invitación ya no le aparecerá al usuario al ver los eventos con invitaciones pendientes de aceptar.

Uri: api/events/invitation/reject

Método: POST

Parámetros de entrada:

- **invitationCode:** Código de la invitación

Valor de retorno: N/A

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST http://localhost:2605/api/events/invitation/reject HTTP/1.1

Host: localhost:2605

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Content-Length: 51

invitationCode=795f692c-ae64-4b42-a8dc-cead69c460e1

Ejemplo de respuesta:

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Wed, 12 Dec 2012 04:36:40 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 39

Connection: Close

```
{"Value":"","Status":"ok","Message":""}
```

E.2.8 Obtener la lista de eventos en la que un usuario es miembro

Descripción: Obtiene la lista de eventos en la que un usuario es miembro

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A

Uri: api/events/member/{idUser}

Método: GET

Parámetros de entrada:

- **idUser:** Identificador del usuario miembro

Valor de retorno: Devuelve a lista de eventos en la que un usuario es miembro

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET http://localhost:2605/api/events/member/10 HTTP/1.1

Host: localhost:2605

Ejemplo de respuesta:

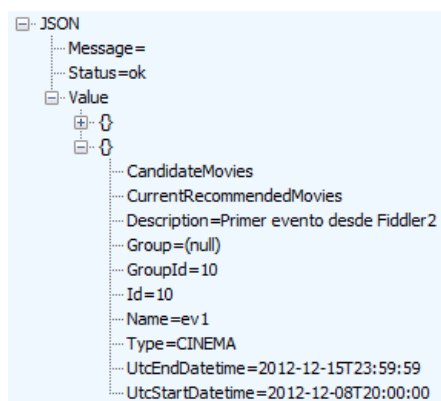


Ilustración 34: Ejemplo de consulta de lista de membrecías del usuario

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Tue, 18 Dec 2012 03:21:11 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 518

Connection: Close

```
{"Value": [{"CandidateMovies": [], "CurrentRecommendedMovies": [], "Description": "Primer evento desde Fiddler2", "Group": null, "GroupId": 9, "Id": 9, "Name": "ev1", "Type": "CINEMA", "UtcEndDatetime": "2012-12-15T23:59:59", "UtcStartDatetime": "2012-12-08T20:00:00"}, {"CandidateMovies": [], "CurrentRecommendedMovies": [], "Description": "Primer evento desde Fiddler2", "Group": null, "GroupId": 9, "Id": 9, "Name": "ev1", "Type": "CINEMA", "UtcEndDatetime": "2012-12-15T23:59:59", "UtcStartDatetime": "2012-12-08T20:00:00"}]}
```

```
Fiddler2", "Group": null, "GroupId": 10, "Id": 10, "Name": "ev1", "Type": "CINEMA", "UtcEndDatetim  
e": "2012-12-15T23:59:59", "UtcStartDatetime": "2012-12-  
08T20:00:00"}], "Status": "ok", "Message": ""}
```

E.2.9 Obtener la lista de eventos con invitación pendiente para un usuario

Descripción: Obtiene la lista de eventos que tienen una invitación pendiente de aceptación para un usuario

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A

Uri: api/events/pending/{idUser}

Método: GET

Parámetros de entrada:

- **idUser:** Identificador del usuario con invitaciones pendientes

Valor de retorno: Devuelve a lista de eventos a los que el usuario ha sido invitado pero la invitación aún no ha sido ni aceptada ni rechazada

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET http://localhost:2605/api/events/pending/10 HTTP/1.1

Host: localhost:2605

Ejemplo de respuesta:

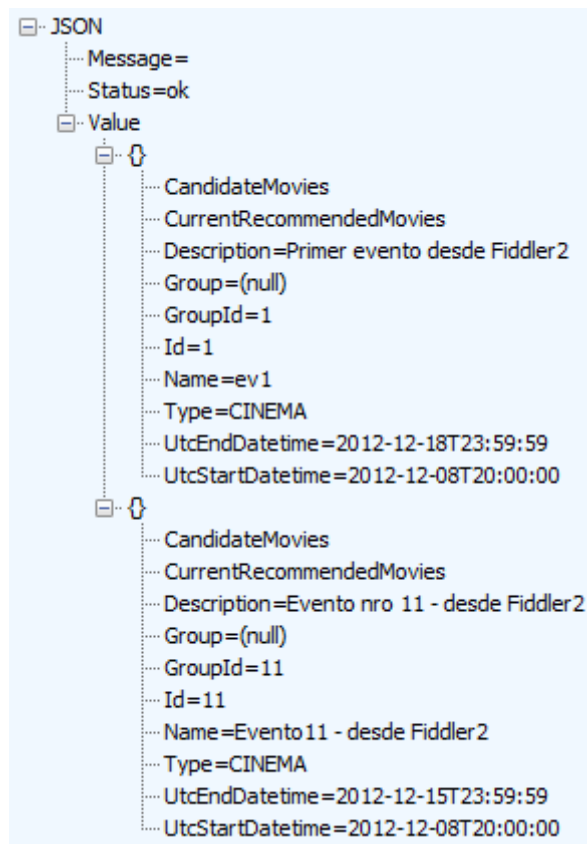


Ilustración 35: Ejemplo de respuesta de listado de invitaciones pendientes

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Tue, 18 Dec 2012 03:30:22 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 542

Connection: Close

```
{"Value": [{"CandidateMovies": [], "CurrentRecommendedMovies": [], "Description": "Primer evento desde Fiddler 2", "Group": null, "GroupId": 1, "Id": 1, "Name": "ev 1", "Type": "CINEMA", "UtcEndDatetime": "2012-12-18T23:59:59", "UtcStartDatetime": "2012-12-08T20:00:00"}, {"CandidateMovies": [], "CurrentRecommendedMovies": [], "Description": "Evento nro 11 - desde Fiddler 2", "Group": null, "GroupId": 11, "Id": 11, "Name": "Evento11 - desde Fiddler 2", "Type": "CINEMA", "UtcEndDatetime": "2012-12-15T23:59:59", "UtcStartDatetime": "2012-12-08T20:00:00"}]}
```

Fiddler2", "Group": null, "GroupId": 11, "Id": 11, "Name": "Evento11 - desde
Fiddler2", "Type": "CINEMA", "UtcEndDatetime": "2012-12-
15T23:59:59", "UtcStartDatetime": "2012-12-08T20:00:00"}], "Status": "ok", "Message": ""}

E.2.10 Obtener la lista de eventos en que un usuario es propietario

Descripción: Obtiene la lista de eventos que ha creado un usuario

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A

Uri: api/events/ownedby/{idUser}

Método: GET

Parámetros de entrada:

- **idUser:** Identificador del usuario propietario/creador del evento

Valor de retorno: Devuelve a lista de eventos que ha creado el usuario

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET http://localhost:2605/api/events/ownedby/40000 HTTP/1.1

Host: localhost:2605

Ejemplo de respuesta:

HTTP/1.1 200 OK

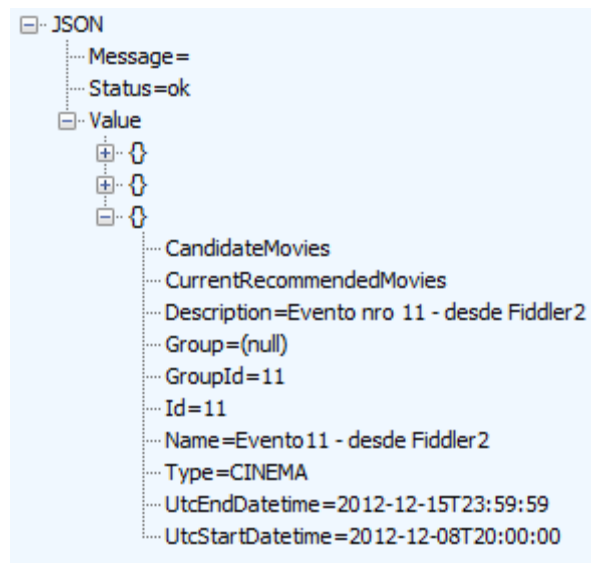


Ilustración 36: Ejemplo de respuesta del listado de eventos organizados por el usuario

Server: ASP.NET Development Server/10.0.0.0

Date: Tue, 18 Dec 2012 03:47:45 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 784

Connection: Close

```
{"Value":[{"CandidateMovies":[],"CurrentRecommendedMovies":[],"Description":"Segundo evento desde Fiddler2","Group":null,"GroupId":2,"Id":2,"Name":"ev2","Type":"CINEMA","UtcEndDatetime":"2012-12-18T23:59:59","UtcStartDatetime":"2012-12-08T20:00:00"}, {"CandidateMovies":[],"CurrentRecommendedMovies":[],"Description":"Primer evento desde Fiddler2","Group":null,"GroupId":10,"Id":10,"Name":"ev1","Type":"CINEMA","UtcEndDatetime":"2012-12-15T23:59:59","UtcStartDatetime":"2012-12-08T20:00:00"}, {"CandidateMovies":[],"CurrentRecommendedMovies":[],"Description":"Evento nro 11 - desde Fiddler2","Group":null,"GroupId":11,"Id":11,"Name":"Evento11 - desde Fiddler2","Type":"CINEMA","UtcEndDatetime":"2012-12-15T23:59:59","UtcStartDatetime":"2012-12-08T20:00:00"}], "Status":"ok", "Message":""}
```

E.2.11 Calificar una película

Descripción: Guarda la calificación que un usuario otorga a una película.

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A.

Uri: api/movie-user/rating

Método: POST

Parámetros de entrada:

- **idMovie:** Identificador de la película que se va a calificar
- **idUser:** Identificador del usuario que realiza la calificación
- **rating (float):** Valor de la calificación otorgada
- **idEvent:** identificador del evento de contexto en el que se realiza la calificación (opcional).

Valor de retorno: N/A

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST http://localhost:2605/api/movie-user/rating HTTP/1.1

Host: localhost:2605

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Content-Length: 37

idMovie=63617&idUser=40000&rating=3,5

Ejemplo de respuesta:

HTTP/1.1 200 OK

Server: ASP.NET Development Server/10.0.0.0

Date: Wed, 12 Dec 2012 04:36:40 GMT

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

Cache-Control: private

Content-Type: application/json; charset=utf-8

Content-Length: 39

Connection: Close

```
{"Value":"","Status":"ok","Message":""}
```

E.2.12 Obtener la lista de amigos de un usuario

Descripción: Devuelve la lista de amistades que tiene un usuario, indicando para cada uno la fecha desde cuando son amigos en el sistema.

Pre-Condiciones: El usuario ha iniciado sesión

Post-Condiciones: N/A

Uri: /api/users/{idUser}/friends/

Método: GET

Parámetros de entrada:

- **idUser:** identificador del usuario para el que se solicitan los amigos

Valor de retorno:

Lista de amistades del usuario

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET /api/users/71568/friends/ HTTP/1.1

Host: localhost

Ejemplo de respuesta:

```
{"Value":[{"$id":"1","Friend":
{"$id":"2","Email":"fernando.gaita@gmail.com","FirstName":"fernando","Id":71569,"
JoinedGroups":[],"LastName":"olivera","OwnedGroups":
[],"PendingInvitationReceived":[],"PendingInvitationSent":
[],"UserName":"fernando"},"FriendId":71569,"Since":"2013-02-
25T00:00:00","UserId":71568},{"$id":"3","Friend":
{"$id":"4","Email":"guille@gmail.com","FirstName":"guillermo","Id":71570,"JoinedGroups":
[],"LastName":"fernandez","OwnedGroups":[],"PendingInvitationReceived":
[],"PendingInvitationSent":[],"UserName":"guille"},"FriendId":71570,"Since":"2013-02-
25T00:00:00","UserId":71568},{"$id":"5","Friend":
{"$id":"6","Email":"nvl_5@hotmail.com","FirstName":"Natalia","Id":71571,"JoinedGroups":
[],"LastName":"Lucero","OwnedGroups":[],"PendingInvitationReceived":
[],"PendingInvitationSent":
[],"UserName":"nvl_5@hotmail.com"},"FriendId":71571,"Since":"2013-02-
25T00:00:00","UserId":71568}], "Status":"ok","Message":""}
```




Ilustración 37: Ejemplo de resupesto de listado de amigos

E.2.13 Indicar gusto/disgusto sobre una entrada de cartelera para el contexto de un evento

Descripción: Guarda la preferencia de gusto (“me gusta” / “no me gusta”) para una entrada de cartelera para el contexto de un evento, es decir que una misma entrada de cartelera puede gustarle al usuario para un evento pero no para otro, permitiendo al usuario decidir dependiendo del contexto de cada evento (como podría ser por ejemplo con que miembros se va a asistir, o el día u horario) si una entrada particular de cartelera le gusta o no.

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario pertenece a los miembros para el evento

Post-Condiciones: Se guardó la preferencia de gusto para el usuario sobre la entrada de cartelera en el contexto del evento.

Uri: /api/users/event-cinemalistsentry/likes

Método: POST

Parámetros de entrada:

- **idUser:** Identificador del usuario para el que se indica la preferencia
- **idEvent:** identificador del evento de contexto
- **idCinemaListingsEntry:** Identificador de la entrada de cartelera
- **likes (boolean):** Valor true/false que indica la preferencia sobre la entrada de cartelera para el evento

Valor de retorno: N/A

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

POST /api/users/event-cinemalistingsentry/likes HTTP/1.1

Content-Type: application/x-www-form-urlencoded; charset=utf-8

Host: localhost

Content-Length: 59

idUser=71568&idEvent=4&idCinemaListingsEntry=20&likes=false

Ejemplo de respuesta:

HTTP/1.1 200 OK

Cache-Control: private, s-maxage=0

Content-Length: 39

Content-Type: application/json; charset=utf-8

Server: Microsoft-IIS/7.5

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

X-Powered-By: ASP.NET

Date: Sat, 02 Mar 2013 18:54:09 GMT

```
{"Value":"","Status":"ok","Message":""}
```

E.2.14 Preferencia de gusto de un usuario sobre una entrada de cartelera en un evento

Descripción: Obtiene la preferencia de gusto (“me gusta” / “no me gusta”) para una entrada de cartelera para el contexto de un evento, es decir que si una entrada de cartelera es candidata en más de un evento, para un mismo usuario, esta operación podría devolver diferentes valores dependiendo del contexto del evento en que se consulte.

Pre-Condiciones:

- El usuario ha iniciado sesión
- El usuario pertenece a los miembros para el evento

Post-Condiciones: N/A.

Uri: `api/users/{idUser}/event-cinemalistingsentry/{idEvent}/{idCinemaListingsEntry}/likes`

Método: GET

Parámetros de entrada:

- **idUser:** Identificador del usuario para el que se solicita la preferencia
- **idEvent:** identificador del evento de contexto
- **idCinemaListingsEntry:** Identificador de la entrada de cartelera

Valor de retorno: Valor true/false/null que indica si el usuario le gustó, no le gustó o no ha indicado preferencia aún respectivamente sobre una entrada de cartelera en el contexto de un evento

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET /api/users/71568/event-cinemalistsentry/4/20/likes HTTP/1.1

Host: localhost

Ejemplo de respuesta (muestra que el usuario ha indicado que no le gusta la entrada de cartelera para el contexto del evento):

HTTP/1.1 200 OK

Cache-Control: private, s-maxage=0

Content-Length: 39

Content-Type: application/json; charset=utf-8

Server: Microsoft-IIS/7.5

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

X-Powered-By: ASP.NET

Date: Sat, 02 Mar 2013 18:54:09 GMT

```
{"Value":False,"Status":"ok","Message":""}
```

Ejemplo de respuesta (muestra que el usuario no ha indicado aún su preferencia por la entrada de cartelera para el contexto del evento):

HTTP/1.1 200 OK

Cache-Control: private, s-maxage=0

Content-Length: 39

Content-Type: application/json; charset=utf-8

Server: Microsoft-IIS/7.5

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

X-Powered-By: ASP.NET

Date: Sat, 02 Mar 2013 18:54:09 GMT

```
{"Value":null,"Status":"ok","Message":""}
```

E.2.15 Obtener la cartelera cinematográfica

Descripción: Obtiene el detalle de las entradas de cartelera que se encuentran dentro de un rango de fechas específico.

Pre-Condiciones:

- El usuario ha iniciado sesión

Post-Condiciones: N/A.

Uri: api/cinema/listings/{strUtcDateFrom}/{strUtcDateTo}/{idUser}

Método: GET

Parámetros de entrada:

- **strUtcDateFrom:** Fecha-hora (utc) a partir de la cual se deben considerar las entradas de cartelera.
- **strUtcDateTo:** Fecha-hora (utc) hasta la cual se deben considerar las entradas de cartelera.
- **idUser (opcional):** Identificador del usuario que solicita la información. En caso de especificarse se carga información de contexto, como por ejemplo las calificaciones que el usuario ha otorgado a las películas de la cartelera.

Valor de retorno: Lista de entradas de cartelera comprendidas para el rango de fechas

Ejemplo de petición (no se incluye el header cookie con los parámetros encriptados de autenticación):

GET /api/cinema/listings/20130201000000/20130301000000/71568 / HTTP/1.1

Host: localhost

Ejemplo de respuesta (entradas de cartelera de febrero del 2013)

HTTP/1.1 200 OK

Cache-Control: private, s-maxage=0

Content-Length: 4389

Content-Type: application/json; charset=utf-8

Server: Microsoft-IIS/7.5

X-AspNet-Version: 4.0.30319

X-AspNetMvc-Version: 4.0

X-Powered-By: ASP.NET

Date: Mon, 25 Mar 2013 06:05:49 GMT

```
{
  "Value": [
    {
      "Cinema": {
        "$id": "1",
        "Address": "Luis Alberto de Herrera 1290",
        "Airconditioning": true,
        "Id": 1,
        "Name": "MovieCenter Montevideo Shopping",
        "OfficialSite": "http://www.moviecenter.com.uy/",
        "Parking": true,
        "ScreenType": "grande",
```

```
"Seats": null,  
"Telephone": "2900 3900"  
},  
"CinemaId": 1,  
"DateTime": "20130226170000",  
"ExtendedProperties": [],  
"Id": 1,  
"Movie": {  
  "$id": "2",  
  "Cast": "Brian Benben (Roger Henderson), Mary Stuart Masterson (Penny Henderson), Ned  
Beatty (General Walt Whalen), George Burns (Milt Lackey), Scott Michael Campbell (Billy)",  
  "Country": "USA",  
  "Director": "Mel Smith",  
  "ExtendedProperties": [  
    {  
      "Key": "UserRating",  
      "Value": 5.0  
    }  
  ],  
  "Genres": "Comedy, Musical, Mystery",  
  "IdImdb": "tt0110939",  
  "IdItem": 513,  
  "ImageUriOriginal": "http://ia.media-  
imdb.com/images/M/MV5BNjU3MzI5ODIzNV5BMl5BanBnXkFtZTcwNDY4MzMzMQ@@._  
V1._SY317_CR4,0,214,317_.jpg",  
  "ImageUriThumbnail": "http://ia.media-  
imdb.com/images/M/MV5BNjU3MzI5ODIzNV5BMl5BanBnXkFtZTcwNDY4MzMzMQ@@._  
V1._SY105_CR4,0,71,105_.jpg",  
  "Language": "English",  
  "LastUpdate": "20130113181443",
```

"Plot": "In 1939, WBN, a fourth radio network, is about to take to America's airwaves. As if the confusion of the premiere night wasn't enough, Penny Henderson, the owner's secretary, must deal with an unhappy sponsor, an overbearing boss and a soon-to-be ex-husband who desperately wants her back. As the broadcast begins, a mysterious voice breaks the broadcast and suddenly members of the cast turn up dead. It's up to her husband Roger, to find out whodunit as the police chase him through the halls of WBN",

"Runtime": 108,

"Sources": [],

"Title": "Radioland Murders (1994)",

"Writer": "George Lucas, Willard Huyck, Gloria Katz, Jeff Reno, Ron Osborn"

},

"MovieItemId": 513

},

{

"Cinema": {

"\$ref": "1"

},

"CinemaId": 1,

"DateTime": "20130226190000",

"ExtendedProperties": [],

"Id": 3,

"Movie": {

"\$id": "3",

"Cast": "Marlon Brando (Don Vito Corleone), Al Pacino (Michael Corleone), James Caan (Sonny Corleone), Richard S. Castellano (Clemenza), Robert Duvall (Tom Hagen)",

"Country": "USA",

"Director": "Francis Ford Coppola",

"ExtendedProperties": [

{

"Key": "UserRating",

"Value": 0.0


```
}  
],  
  "Genres": "Crime, Drama",  
  "IdImdb": "tt0068646",  
  "IdItem": 858,  
  "ImageUriOriginal": "http://ia.media-  
imdb.com/images/M/MV5BMjEyMjcyNDI4MF5BMl5BanBnXkFtZTcwMDA5Mzg3OA@@._  
V1._SY317_.jpg",  
  "ImageUriThumbnail": "http://ia.media-  
imdb.com/images/M/MV5BMjEyMjcyNDI4MF5BMl5BanBnXkFtZTcwMDA5Mzg3OA@@._  
V1._SY105_.jpg",  
  "Language": "English, Italian, Latin",  
  "LastUpdate": "20130113182634",  
  "Plot": "The story begins as \"Don\" Vito Corleone, the head of a New York  
Mafia \"family\", oversees his daughter's wedding. His beloved son Michael has just come home  
from the war, but does not intend to become part of his father's business. Through Michael's life  
the nature of the family business becomes clear. The business of the family is just like the head of  
the family, kind and benevolent to those who give respect, but given to ruthless violence  
whenever anything stands against the good of the family. Don Vito lives his life in the way of the  
old country, but times are changing and some don't want to follow the old ways and look out for  
community and \"family\". An up and coming rival of the Corleone family wants to start selling  
drugs in New York, and needs the Don's influence to further his plan. The clash of the Don's  
fading old world values and the new ways will demand a terrible price, especially from Michael,  
all for the sake of the family",  
  "Runtime": 175,  
  "Sources": [],  
  "Title": "Godfather, The (1972)",  
  "Writer": "Mario Puzo, Francis Ford Coppola, Mario Puzo"  
},  
  "MovieIdItem": 858  
}  
],  
  "Status": "ok",  
  "Message": ""
```

}



Illustration 38: Ejemplo de respuesta de cartelera (2013)

Anexo F Casos de Uso

F.1 CU1 - Registro de Usuario

Se crea un usuario para utilizar el sistema.

Pre-Condiciones:

N/A

Flujo de eventos:

Acción de Actor	Respuesta del Sistema
1 – El usuario elige la opción Registro de Usuario	
	2 – El sistema presenta un formulario con los siguientes campos: Usuario, Contraseña, Confirmación Contraseña, Nombre Completo, correo electrónico e imagen del usuario.
3 – El usuario ingresa los datos	
	4 – El sistema da de alta al usuario y le informa que debe calificar 10 películas antes de que el sistema le pueda recomendar películas.

Flujos Alternativos:

4A – Usuario ya existe

4A.1 El sistema le indica al usuario que ya existe otro usuario con ese nombre de usuario, por lo tanto no es posible registrarlo.

4A.2 Vuelve al paso 3

4B – Contraseña sin confirmar

4B.1 El sistema le indica al usuario que la contraseña no coincide con la confirmación de la contraseña.

4B.2 Vuelve al paso 3

4C – Faltan datos

4B.1 El sistema le indica al usuario que debe completar todos los datos.

4B.2 Vuelve al paso 3

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones:

Queda ingresado el usuario en el sistema.

F.2 CU2 - Login

El usuario se autentica en el sistema.

Pre-Condiciones:

- El usuario no debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema
1 – El usuario elige la opción Login.	
	2 – El sistema solicita usuario y contraseña. Además ofrece la opción de recordar la autenticación para el navegador.
3 – El usuario ingresa usuario y contraseña	
	4 – El sistema verifica que el usuario y la contraseña sean correctos y el usuario queda autenticado.

Flujos Alternativos:

2A – Autenticación recordada

2A.1 El sistema verifica que el usuario tiene la autenticación recordada para el navegador. El usuario queda autenticado en el sistema

2A.2 Fin de caso de uso.

3A – Recuerda autenticación

3A.1 El usuario ingresa usuario y contraseña, además marca la opción de recordar.

3A.2 El sistema verifica que el usuario y la contraseña sean correctos, el usuario queda autenticado, se recuerda la autenticación para el navegador.

3A.3 Fin de caso de uso.

4A – Usuario o Contraseña Incorrectos

4A.1 El sistema no encuentra un usuario con esa contraseña y lo informa al usuario.

4A.2 Vuelve al paso 3.

G1. Cancelar

G1.1 El usuario cancela la consulta.

Post-Condiciones:

El usuario queda autenticado en el sistema.

F.3 CU3 - Consultar datos de Películas

El usuario busca la película que le interesa y consulta sus datos.

Pre-Condiciones:

El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema
1 – El usuario elige la opción Búsqueda de Películas	
	2 – El sistema presenta los siguientes campos de búsqueda: Título, Reparto, Director, Escritor, País Origen, Idioma, Género.
3 – El usuario ingresa los datos y realiza la búsqueda	
	4 – El sistema muestra la lista de películas que coinciden con los datos ingresados.
5 – El usuario elige la película y consulta sus datos	
	6 – El sistema muestra los datos de la película.

Punto Extensión:

Punto 6: Reproducir Película

Punto 6: Calificar Película

Flujos Alternativos:

4A – No se encuentran películas

4A.1 El sistema no encuentra ninguna película con coincide con los datos ingresados.

4A.2 Vuelve al punto 1.

G1. Cancelar

G1.1 El usuario cancela la consulta.

Post-Condiciones:

N/A

F.4 CU4 - Reproducir Película

El usuario quiere reproducir una película para lo cual, el sistema le brinda las opciones de proveedores de contenido que hay disponibles para esa película.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.
- Debe haber una película seleccionada.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Proveedor de Contenido
1 – El usuario elige la opción Reproducción de película.		
	2 – El sistema presenta la lista de proveedores de contenidos que hay disponibles para esta película	
3 – El usuario elige un proveedor e indica reproducir.		
	4 – El sistema solicita al proveedor que reproduzca el video.	
		5 – El Proveedor de contenidos reproduce la película

Flujos Alternativos:

G1. Cancelar

G1.1 El usuario cancela la reproducción.

Post-Condiciones:

El proveedor de contenidos elegido reproduce la película.

Diagrama de secuencia: En la ilustración 39 se presenta el diagrama de secuencia de la operación más relevante del caso de uso.

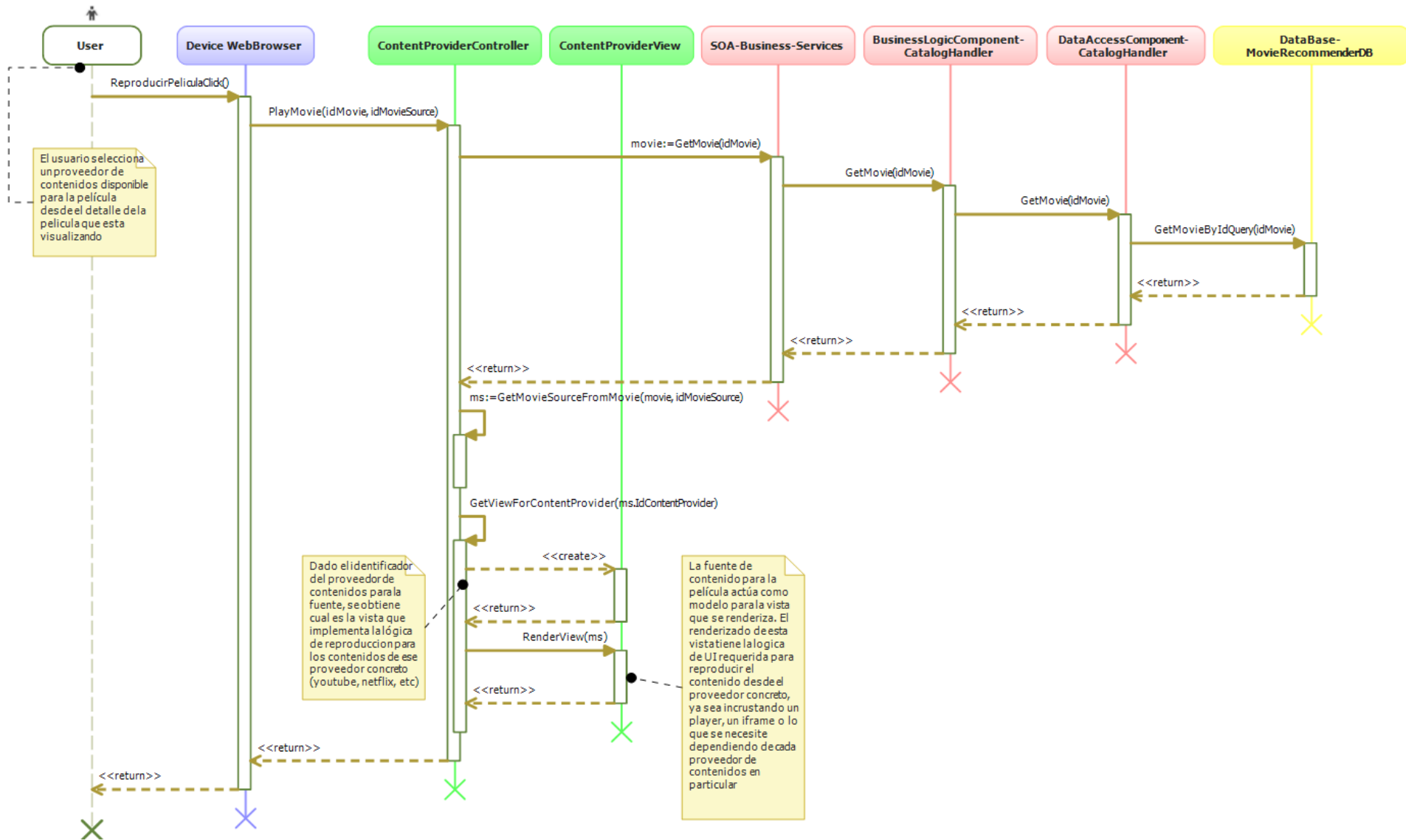


Ilustración 39: Diagrama de secuencia - CU Reproducir Película

F.5 CU5 - Calificar Película

El usuario califica una película dentro de la escala de 1 a 5.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.
- Debe haber una película seleccionada.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Calificar Película.		
	2 – El sistema muestra los posibles valores que puede calificar el usuario: 1 – 1,5 – 2 – 2,5 – 3 – 3,5 – 4 – 4,5 – 5.	
3 – El usuario ingresa un valor de calificación.		
	4 – El sistema solicita al Servicio de Recomendación que ingrese la calificación del usuario y la película seleccionada.	
		5 – El servicio de Recomendación ingresa la calificación del usuario para la película.

Post-Condiciones:

Queda registrado en el Servicio de Recomendación la calificación del usuario para la película.

Diagrama de secuencia: En las ilustraciones 40 y 41 se presenta el diagrama de secuencia de la operación más relevante del caso de uso.

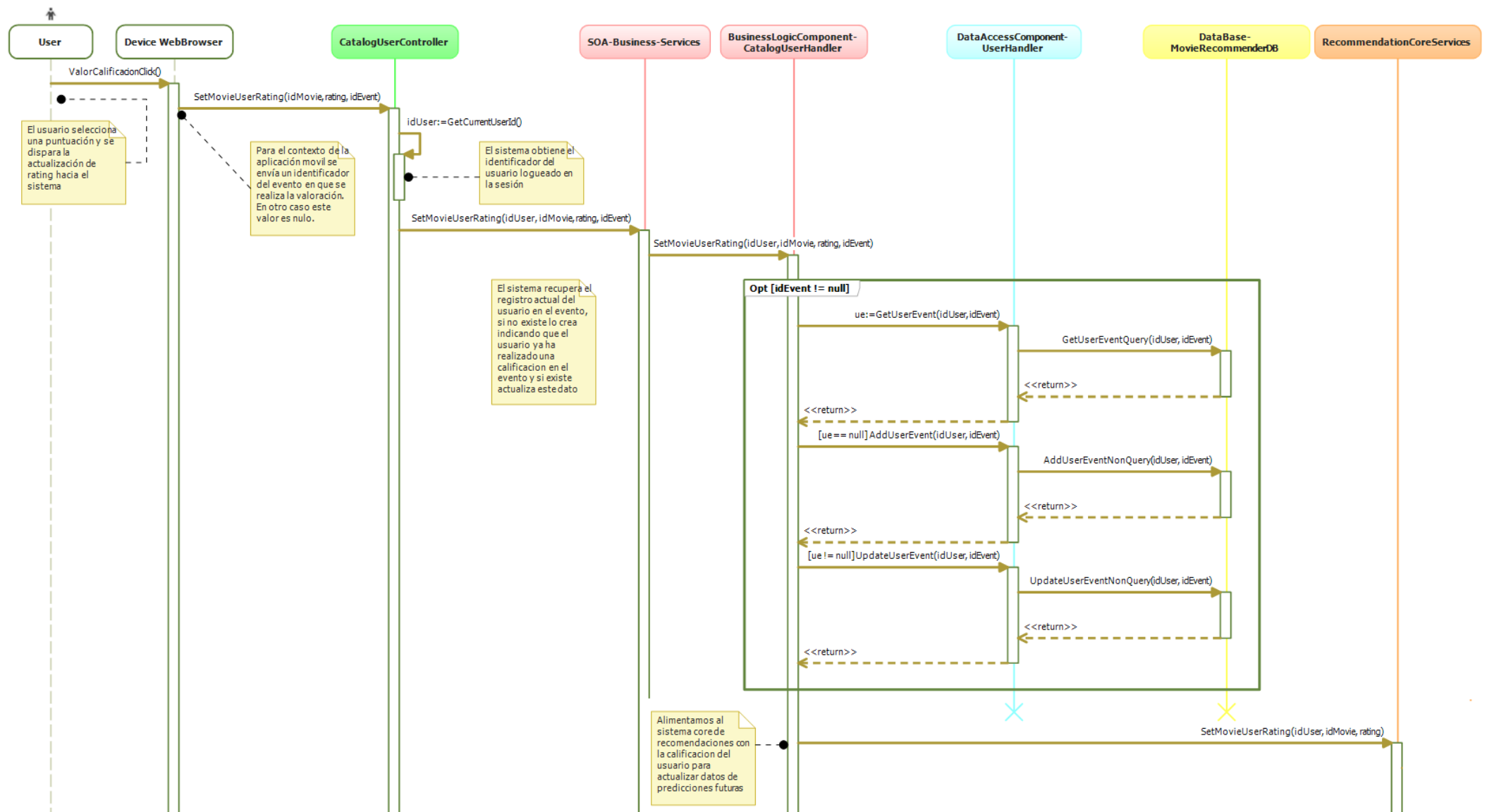


Ilustración 40: Diagrama de secuencia - CU Calificar Película (1 de 2)

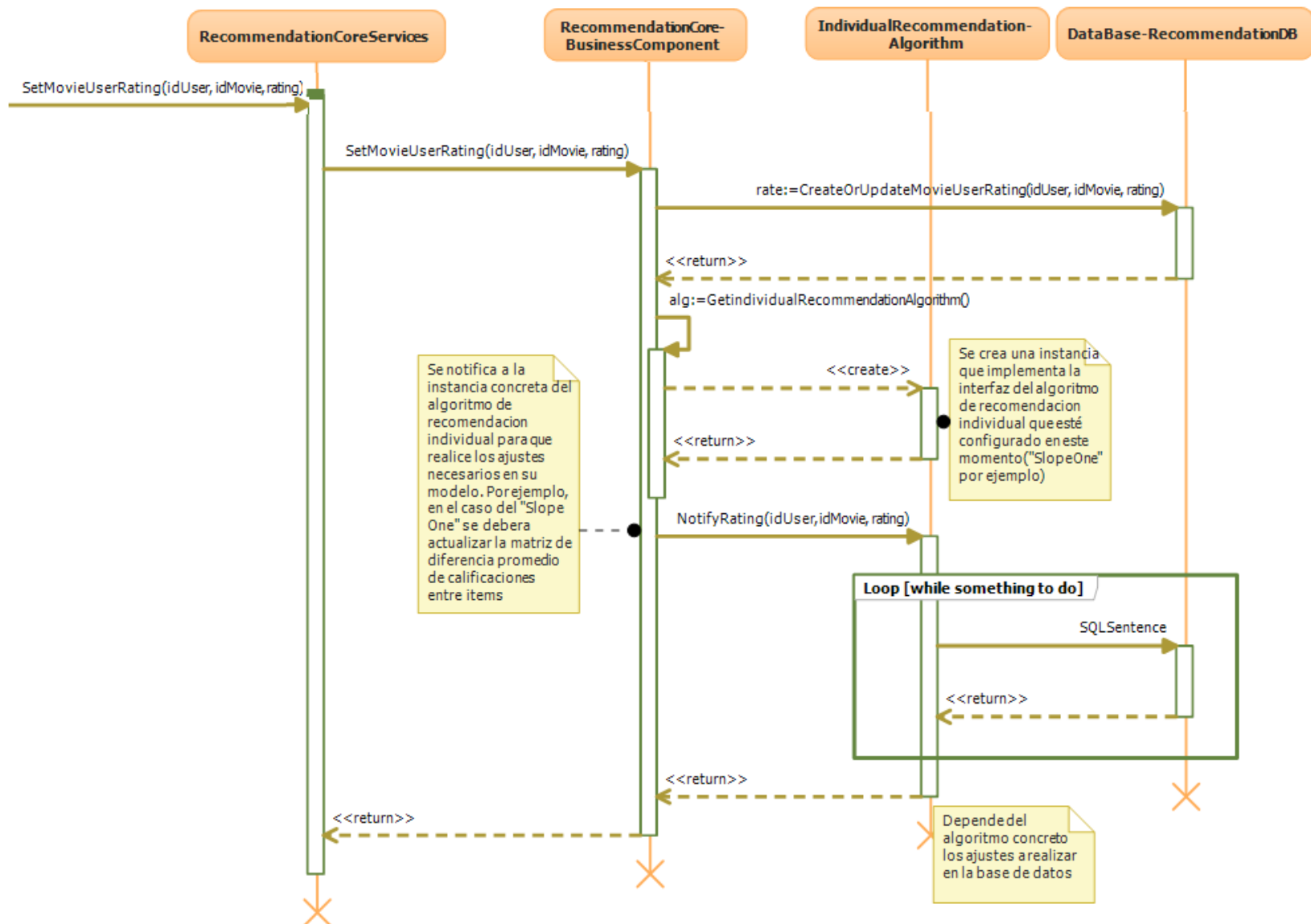


Ilustración 41: Diagrama de secuencia - CU Calificar Película (2 de 2)

F.6 CU6 - Recomendar Película Usuario

El usuario quiere que el sistema le recomiende una película y poder reproducirla. Para mejorar la recomendación, el usuario puede ingresar una serie de datos para filtrar el conjunto de películas sobre el cual quiere que se le recomiende.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Recomendación de Película para Usuario		
	2 – El sistema presenta los siguientes campos de búsqueda: Título, Reparto, Director, Escritor, País Origen, Idioma, Género. Además permite modificar la cantidad de películas recomendadas, por defecto en 10 películas.	
3 – El usuario ingresa los datos y solicita la recomendación.		
	4 – El sistema solicita al Servicio de Recomendación que ejecute el algoritmo de recomendación individual que esta utilizando para el usuario.	
	.	5 – El Servicio de Recomendación ejecuta el algoritmo individual y retorna una lista con la cantidad de ítems recomendables.
	6 – El sistema muestra la lista con la cantidad películas recomendadas de forma ordenada, desde la más recomendable hasta la menos recomendable para el usuario	
7 – El usuario elige una película		
	8 – El sistema muestra los datos de la película.	
9 – Incluir Caso de Uso Reproducción de película		
10 – Incluir Caso de Uso Calificar Película		

Flujos Alternativos:

3A – Cantidad recomendada

3A.1 El usuario ingresa los datos, cambia la cantidad de películas recomendadas y solicita la recomendación.

4A – No se encuentran películas para recomendar

4A.1 El sistema no encuentra ninguna película para recomendar.

4A.2 Vuelve al punto 1.

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones:

El sistema le recomendó una lista de películas (por defecto 10) al usuario de las cuales el usuario eligió una para reproducir y luego le otorgó su calificación.

Diagrama de secuencia: En las ilustraciones 42, 43, 44, 45 y 46 se presenta el diagrama de secuencia de las operaciones más relevantes del caso de uso.

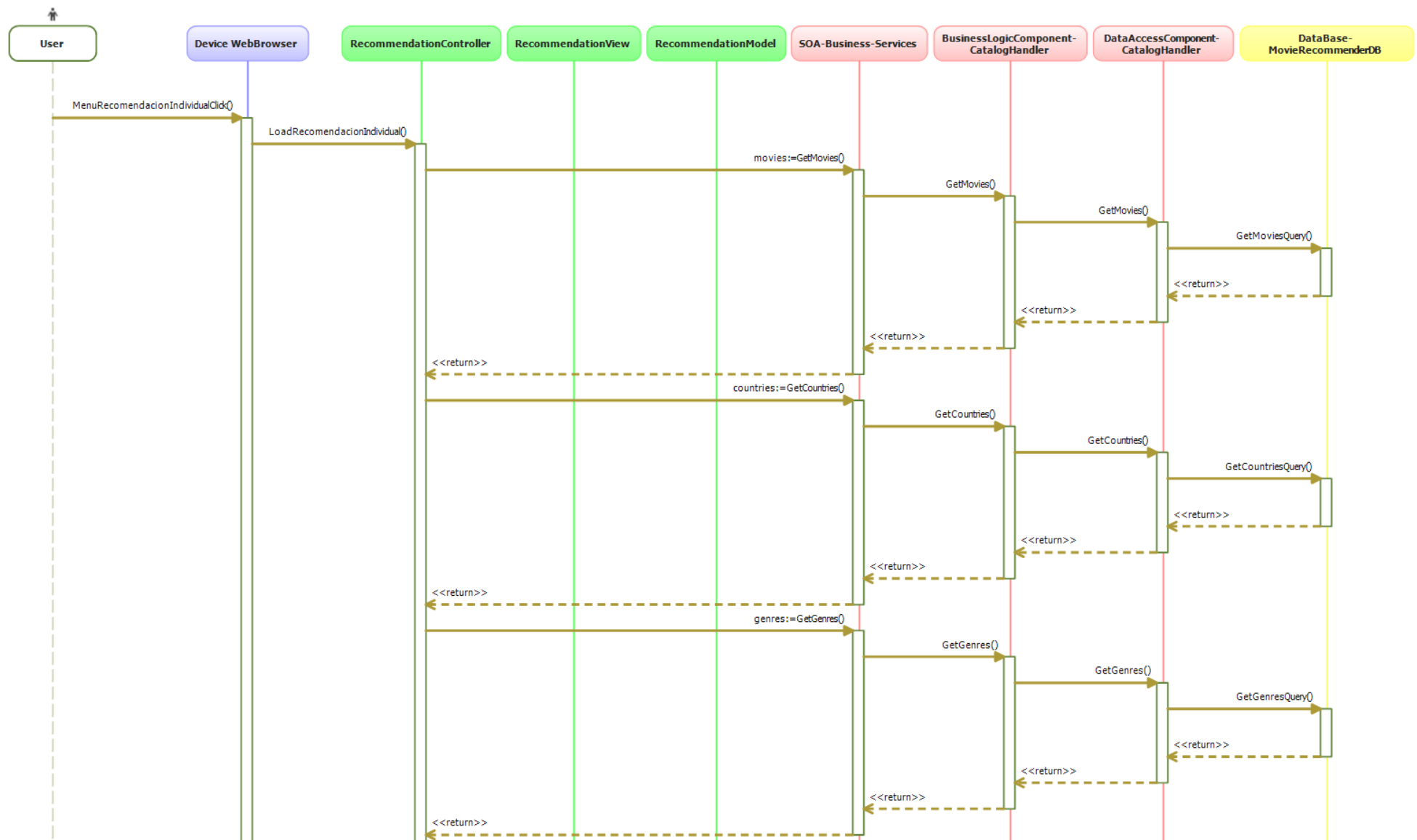


Ilustración 42: Diagrama de secuencia - CU Recomendar Película Usuario (1 de 5)

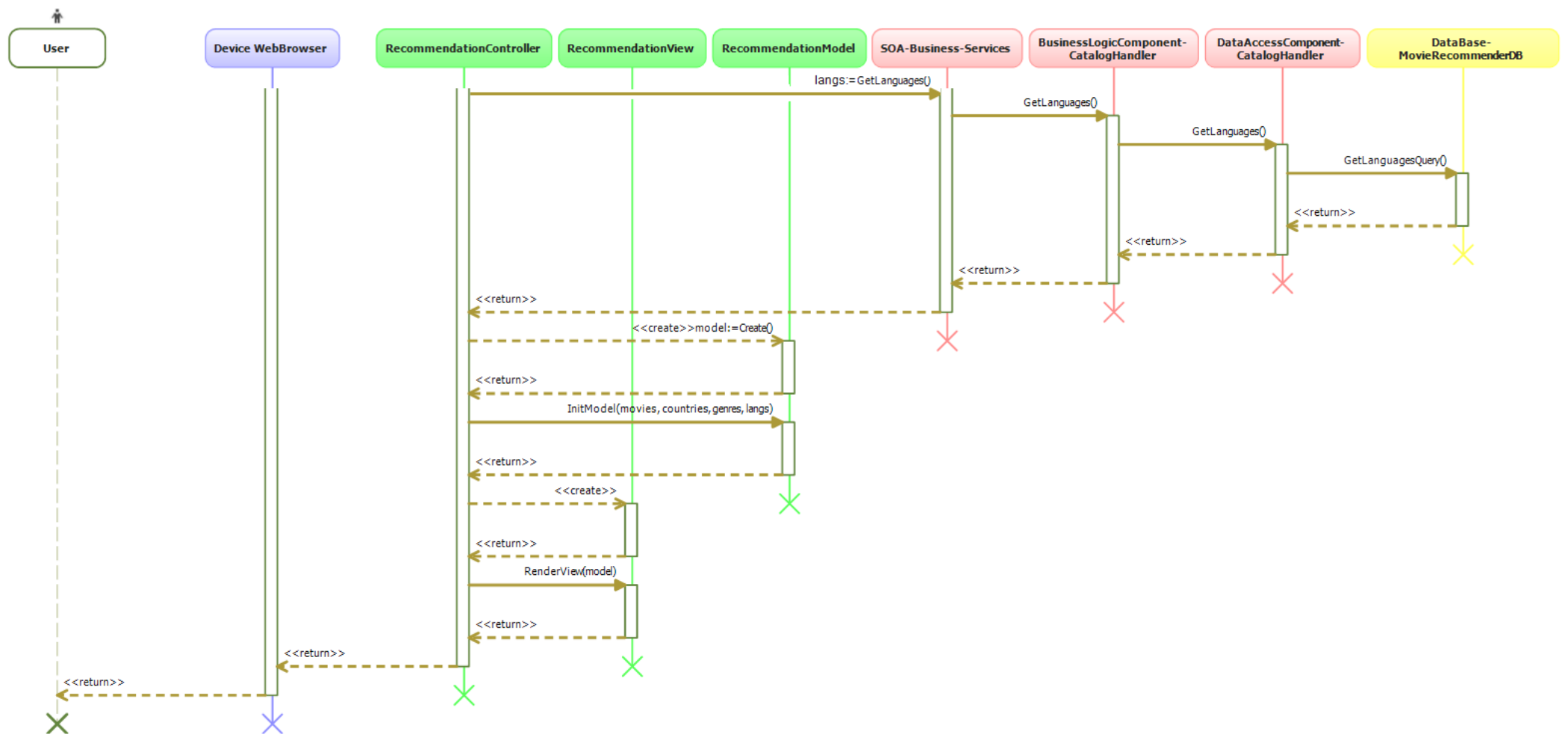


Ilustración 43: Diagrama de secuencia - CU Recomendar Película Usuario (2 de 5)

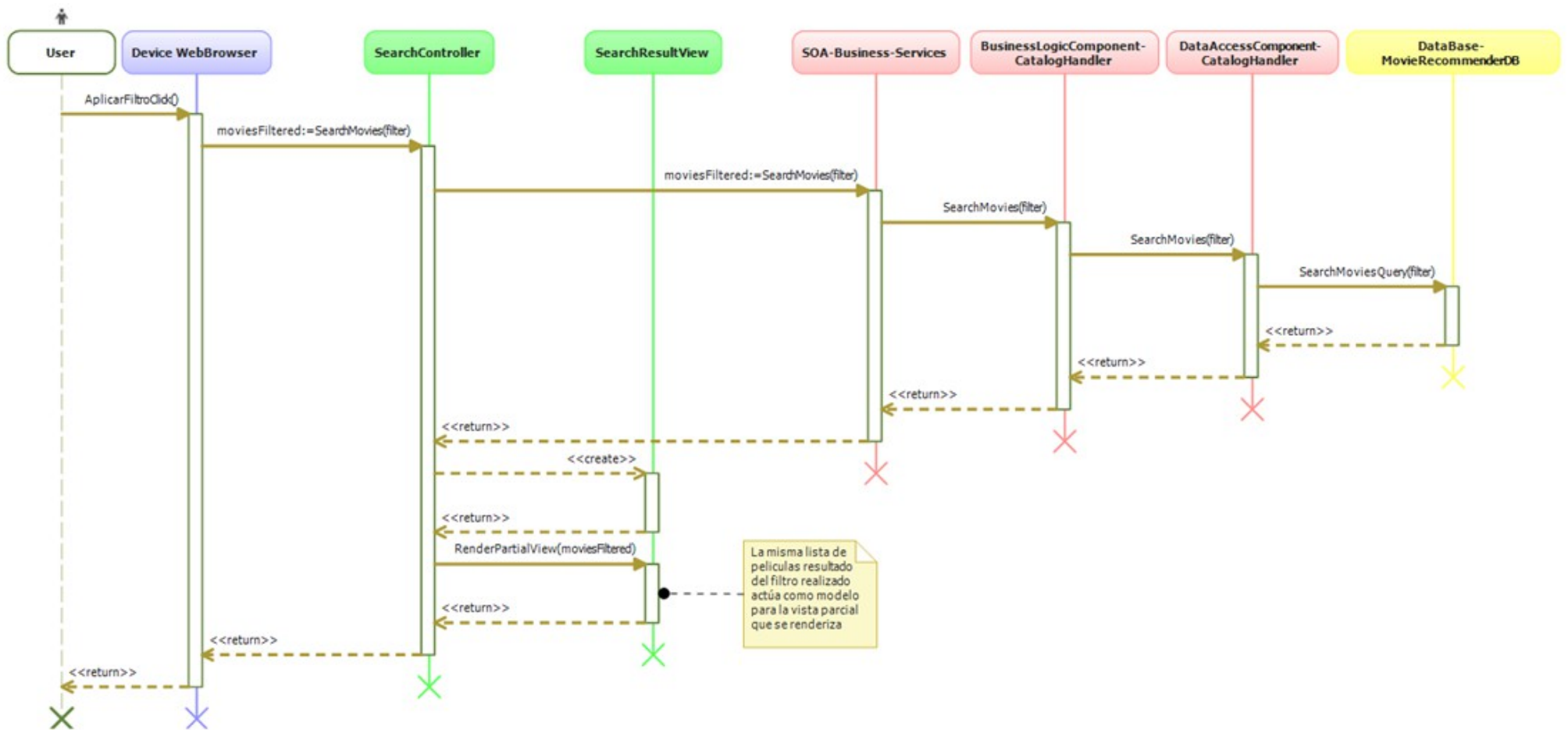


Ilustración 44: Diagrama de secuencia - CU Recomendar Película Usuario (3 de 5)

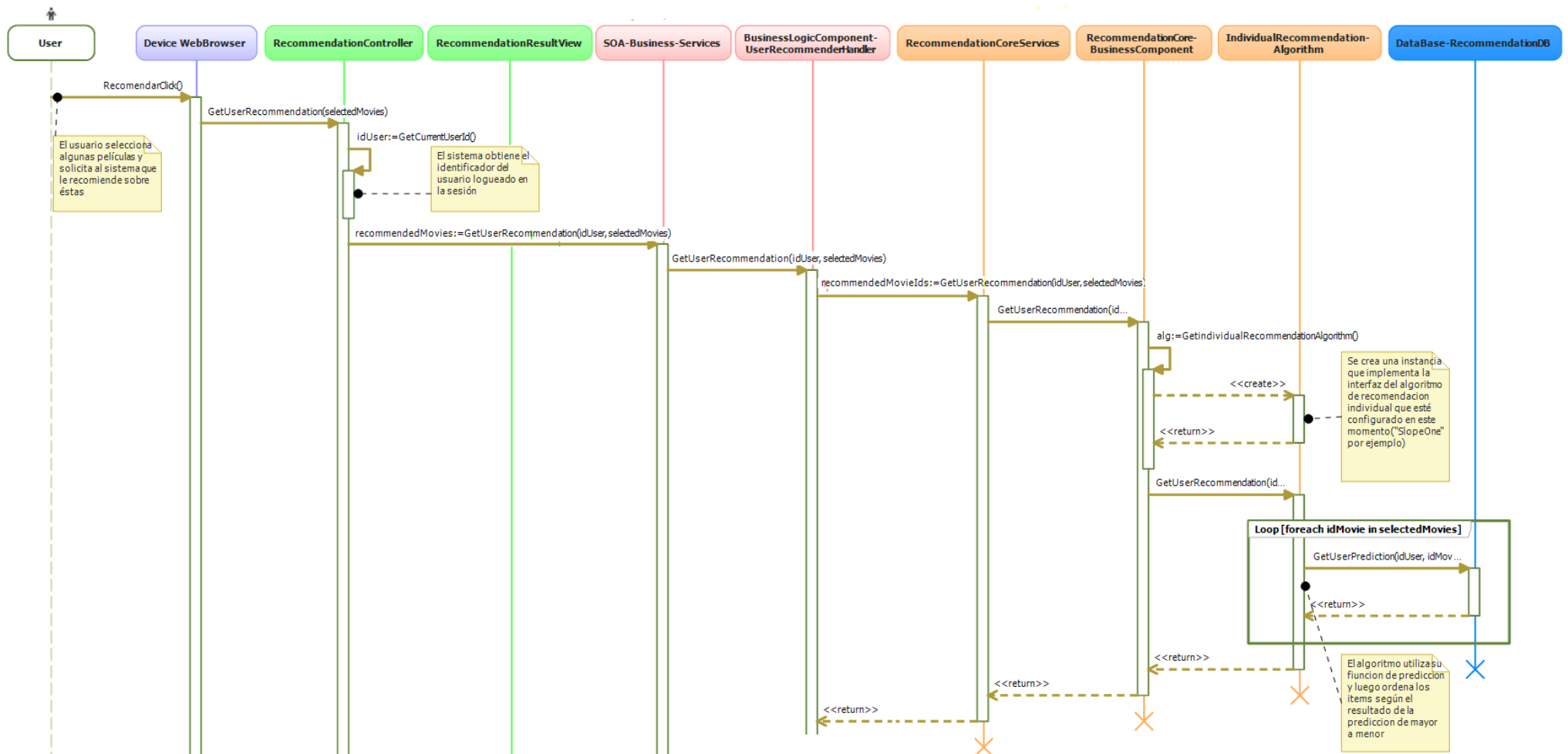


Ilustración 45: Diagrama de secuencia - CU Recomendar Película Usuario (4 de 5)

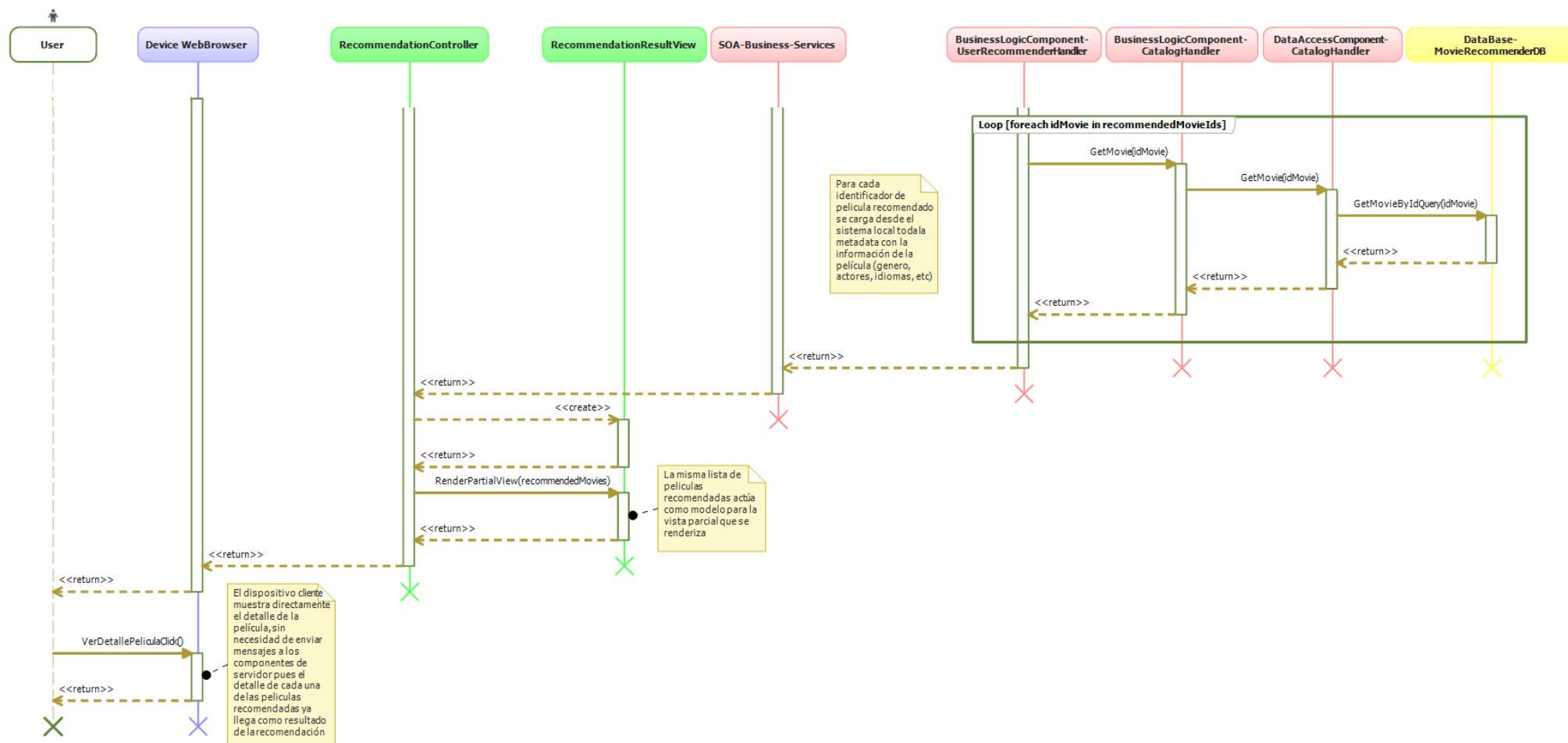


Ilustración 46: Diagrama de secuencia - CU Recomendar Película Usuario (5 de 5)

F.7 CU7 - Recomendar Película Grupo

Existe un grupo de usuarios que quieren que el sistema le recomiende una película y poder reproducirla. Para mejorar la recomendación, los usuarios pueden ingresar una serie de datos para filtrar el conjunto de películas sobre el cual desean que les recomiende.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.
- Todos los usuarios del grupo deben tener registrados al menos 10 calificaciones en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Recomendación de Película para Grupos.		
	2 – El sistema presenta los grupos que integra el usuario.	
3 – El usuario elige un grupo.		
	4 – El sistema presenta los siguientes campos: Título, Reparto, Director, Escritor, País Origen, Idioma, Género. Además permite modificar la cantidad de películas recomendadas, por defecto en 10 películas.	
5 – El usuario ingresa los datos y solicita la recomendación para el grupo elegido.		
	6 – El sistema solicita al Servicio de Recomendación que ejecute el algoritmo de recomendación grupal que esta utilizando para el grupo.	
	.	7 – El Servicio de Recomendación ejecuta el algoritmo grupal y retorna una lista con la cantidad de ítems recomendables.
	8 – El sistema muestra la lista con las cantidad de películas recomendadas de forma ordenada, desde la más recomendable hasta la menos recomendable para el grupo	
9 – El usuario elige una película que el grupo quiere reproducir.		
	10 – El sistema muestra los datos de la película.	
11 – Incluir Caso de Uso Reproducción de película.		
12 – Incluir Casos de Uso Calificar Película		

Flujos Alternativos:

5A – Cantidad recomendada

5A.1 El usuario ingresa los datos, cambia la cantidad de películas recomendadas y solicita la recomendación.

6A – No se encuentran películas para recomendar

6A.1 El sistema no encuentra ninguna película para recomendar.

6A.2 Vuelve al punto 1.

3A – Grupo Efímero.

3A.1 El usuario elige “Grupo efímero”

3A.2 El sistema agrega al usuario a un “Grupo efímero”

3A.3 El sistema solicita usuario y contraseña de un nuevo usuario para agregar al “Grupo efímero”.

3A.4 El usuario ingresa usuario y contraseña.

3A.5 El sistema valida usuario y contraseña

3A.6 Vuelve a paso 3A.3

Flujo alternativo de 3A:

3A.5.A – Usuario o contraseña incorrecta

3A.5.A.1. El sistema indica que el usuario o la contraseña no es válida

3A.5.A.2. Vuelve a paso 3A.4

Flujo alternativo de 3A:

3A.6.A – Terminar

3A.6.A.1. Vuelve a paso 4

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones:

El sistema le recomendó una lista de (por defecto 10) películas para un grupo de usuarios, de las cuales eligen una para reproducir.

F.8 CU8 - Crear Evento

Un usuario quiere invitar a un grupo de usuarios al cine, para esto crea un evento con una fecha de inicio y fecha de fin y los invita.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema
1 – El usuario elige la opción Crear Eventos.	
	2 – El sistema solicita que se ingresen los siguientes campos: Título del evento, Comentario, Fecha Inicio Fecha de Fin y la lista de usuarios invitados.
3 – El usuario ingresa los datos.	
	4 – El sistema crea el evento y las invitaciones para cada uno de los usuarios especificados.

Flujos Alternativos:

G1. Cancelar

G1.1 El usuario cancela la creación del evento.

Post-Condiciones:

El sistema creó un evento con las invitaciones para cada uno de los usuarios especificados.

F.9 CU9 - Aceptar Invitaciones

Un usuario quiere ver si fue invitado en algún evento para ir al cine y aceptar o rechazar dicha invitación.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Servicio Recomendación
1 – El usuario elige la opción Invitaciones.		
	2 – El sistema muestra la lista de invitaciones a las cuales fue invitado el usuario.	
3 – El usuario elige una invitación y la acepta.		
	4 – El sistema agrega al usuario como miembro del grupo del evento. El sistema solicita al servicio de recomendación que recomiende 10 películas para el grupo a partir de la cartelera de películas incluidas entre la fecha de inicio del evento y la fecha de fin.	
		5 – El Servicio de Recomendación ejecuta el algoritmo grupal y retorna una lista de 10 ítems recomendables.
	6 – El sistema actualiza la recomendación para el grupo	

Flujos Alternativos:

2A – Sin invitaciones.

2A.1 El sistema indica que no hay invitaciones pendientes para el usuario

3A.2 Finaliza el caso de uso.

3A – Rechazar invitación.

3A.1 El usuario opta por rechazar la invitación

3A.2 El sistema registra que el usuario rechazó la invitación.

3A.3 Finaliza el caso de uso.

Post-Condiciones:

El sistema agrega al usuario como miembro del evento y recalculó la recomendación grupal.

F.10 CU10 - Recomendación Evento

Los usuarios miembros del grupo obtienen una recomendación grupal del sistema a partir de una cartelera cinematográfica. Además pueden ver qué eligió cada miembro del grupo.

Pre-Condiciones:

- El usuario debe estar autenticado en el sistema.
- El usuario debe ser miembro del grupo.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema
1 – El usuario elige la opción Recomendación Evento.	
	2 – El sistema muestra la lista de películas de la cartelera cinematográfica y muestra la lista ordenada por predicción de películas recomendadas para el grupo, incluyendo la predicción de calificación para el grupo en cada película. Además indica qué película le gusta y no le gusta a cada miembro del evento.
3 – El usuario indica qué película elige para ver.	
	4 – Sistema registra que película ha elegido el usuario.

Flujos Alternativos:

3A – Detalle película.

- 3A.1 El usuario indica qué película no elige para ver
- 3A.2 El sistema muestra todos los datos disponibles de la película
- 3A.3 El usuario elige Volver
- 3A.4 Vuelve al paso 3

3B – No elegir.

- 3A.1 El usuario indica que no elige una película
- 3A.2 Vuelve al paso 3

Punto de extensión:

Punto 3.A.2: Calificar Película

G1. Cancelar

- G1.1 El usuario no elige ninguna película

Post-Condiciones:

El sistema registra qué película quiere ver el usuario.

F.11 CU11 - Sincronizar Proveedor

Se sincroniza el catálogo de películas con los proveedores de contenido que tienen disponibles.

Pre-Condiciones:

- El usuario Administrador debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Proveedor Contenido
1 – El Usuario Administrador ingresa a la opción de Sincronizar Proveedores de Contenido		
	2 – El sistema presenta la lista de proveedores configurados	
3 – El usuario elige un proveedor e indica sincronizar.		
	4 – El sistema consulta el catálogo del proveedor.	
		5 – El proveedor de contenido retorna el catálogo de películas que tiene disponibles.
	6 – El Sistema verifica un nuevo contenido del proveedor y solicita la información para agregarlo como fuente en el caso.	
		7 – El proveedor de contenido devuelve la información solicitada.
	8 – El Sistema guarda los datos de la película que brinda el proveedor de contenido y vuelve al paso 6.	
	9 – El Sistema detecta las bajas del proveedor de contenido y las borra de su lista de proveedores.	

Flujos Alternativos:

8A. Terminar

8A.1 El Sistema sincronizó todas las nuevas películas.

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones:

Se agregan como fuente de contenido las nuevas películas que ha agregado el proveedor y se eliminan las que ya no están disponibles del catálogo que tiene el sistema.

F.12 CU12 - Sincronizar Catálogo

Se sincroniza el catálogo de películas con los proveedores de información de catálogos.

Pre-Condiciones:

- El usuario Administrador debe estar autenticado en el sistema.

Flujo de eventos:

Acción de Actor	Respuesta del Sistema	Proveedor Contenido
1 – El Usuario Administrador ingresa a la opción de Sincronizar Proveedores de Catálogo		
	2 – El sistema presenta la lista de proveedores configurados	
3 – El usuario elige un proveedor e indica sincronizar.		
	4 – El sistema consulta el catálogo del proveedor.	
		5 – El proveedor de contenido retorna el catálogo de películas que tiene disponibles.
	6 – El Sistema verifica un nuevo contenido del proveedor y solicita la información para agregar el contenido.	
		7 – El proveedor de contenido devuelve la información solicitada.
	8 – El Sistema guarda la nueva película y vuelve al paso 6.	

Flujos Alternativos:

8A. Terminar

8A.1 El Sistema sincronizó todas las nuevas películas.

8A.2 Terminar.

G1. Cancelar

G1.1 El usuario cancela la recomendación.

Post-Condiciones:

Se agregan las nuevas películas que ofrece el proveedor de catálogo.

Anexo G Casos de prueba

G.1 Pruebas funcionales

G.1.1 Escenarios

CU1 - Registro de Usuario

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 4A – Alternativo G1
Escenario 3	Flujo normal – Alternativo 4B – Flujo normal
Escenario 3	Flujo normal – Alternativo 4C – Flujo normal

CU2 - Login

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 4A – Alternativo G1
Escenario 3	Flujo normal – Alternativo 4A – Flujo normal
Escenario 4	Flujo normal – Alternativo 2A
Escenario 5	Flujo normal – Alternativo 3A – Flujo normal

CU3 - Consulta datos de Películas

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Punto extensión 6 Reproducir película – Punto extensión 6: Calificar Película
Escenario 3	Flujo normal – Alternativo 4A – Flujo normal
Escenario 4	Flujo normal – Alternativo 4A – Alternativo G1

CU6 - Recomendar Película Usuario

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 4A – Flujo normal
Escenario 3	Flujo normal – Alternativo 4A – Alternativo G1
Escenario 4	Flujo normal – Alternativo 3A –

	Flujo normal
--	--------------

CU7 - Recomendar Película Grupo

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 6A – Flujo normal
Escenario 3	Flujo normal – Alternativo 6A – Alternativo G1
Escenario 4	Flujo normal – Alternativo 3A – Alternativo 3A – Alternativo 3A6A – Flujo normal
Escenario 5	Flujo normal – Alternativo 3A – Alternativo 3A5A – Flujo normal
Escenario 6	Flujo normal – Alternativo 5A – Flujo normal

CU8 - Crear Evento

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo G1

CU9 – Aceptar Invitaciones

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 2A
Escenario 3	Flujo normal – Alternativo 3A

CU10 - Recomendación Evento

Escenarios	Flujo
Escenario 1	Flujo normal
Escenario 2	Flujo normal – Alternativo 3A – Flujo normal
Escenario 3	Flujo normal – Alternativo 3A – Punto extensión 3.A.2 Alternativo 3A – Flujo normal
Escenario 4	Flujo normal – Alternativo 3B – Flujo normal

G.1.2 Casos de prueba

CU1 - Registro de Usuario

Id - Caso de prueba	Escenario / Condición	Usr	Contra-seña	Confir. Contra-seña	Nombre completo	Correo electrónico	Resultado esperado
CU1-CP1	Escenario 1 / Registro exitoso	V	V	V	N/A	N/A	Alta del usuario con los datos ingresado.
CU1-CP2	Escenario 2 / Ya existe el usuario	I	V	V	N/A	N/A	Mensaje de error “Usuario ya existe”. No se da de alta el usuario.
CU1-CP3	Escenario 3 / Confirmación de contraseña incorrecta	V	I	I	N/A	N/A	Mensaje de error “No coinciden la contraseña y la confirmación”. El usuario corrige y queda registrado.
CU1-CP4	Escenario 4 / Faltan datos	V	V	V	N/A	I	Mensaje de error “Debe ingresar todos los datos”. El usuario ingresa el correo electrónico y queda registrado.

CU2 - Login

Id - Caso de prueba	Escenario / Condición	Usuario	Contraseña	Recordar	Resultado esperado
CU2-CP1	Escenario 1 / V Login exitoso	V	V	V	El usuario se loguea.
CU2-CP2	Escenario 2 / I Nombre usuario incorrecto	V	V	V	Mensaje de error "Nombre de usuario y/o contraseña incorrecta". El usuario no se loguea.
CU2-CP3	Escenario 3 / V Contraseña incorrecta	I	V	V	Mensaje de error "Nombre de usuario y/o contraseña incorrecta". El usuario ingresa los datos correctamente y se loguea.
CU2-CP4	Escenario 4 / N/A Autenticación recordada	N/A	N/A	V	El usuario se loguea.
CU2-CP5	Escenario 5 / V Recordar autenticación	V	V	V	El usuario se loguea y se recuerda la autenticación para el navegador.

CU3 - Consulta datos de Películas

Id - Caso de prueba	Escenario / Condición	Título	Reparto	Director	Escritor	País Origen	Idioma	Género	Voto	Resultado esperado
CU3-CP1	Escenario 1 / # películas = 1	V	V	V	V	V	V	V	N/A	Se muestra una película que coincide.
CU3-CP2	Escenario 1 / # películas > 1	V	V	V	V	V	V	V	N/A	Se muestra una lista de películas.
CU3-CP3	Escenario 2 / V Reproducir proveedor IMDB y Vota 5	V	V	V	V	V	V	V	V	Reproduce la película desde IMDB y luego la vota.
CU3-CP4	Escenario 2 / V Reproducir proveedor IVA y Vota 0	V	V	V	V	V	V	V	V	Reproduce la película desde IVA y luego la vota.
CU3-CP5	Escenario 3 / No se encuentran películas, continúa buscando	I	I	I	I	I	I	I	N/A	Mensaje "No se encontraron películas". El usuario vuelve a ingresar los datos de búsqueda
CU3-CP6	Escenario 3 / No se encuentran películas, Cancela	I	I	I	I	I	I	I	N/A	Mensaje "No se encontraron películas". El usuario cancela la búsqueda

CU6 - Recomendar Película Usuario

Id - Caso de prueba	Escenario / Condición	Genero	Actores	Idioma	Director	Cantidad recomendada	Resultado esperado
CU6-CP1	Escenario 1 / Recomendación por género	V	N/A	N/A	N/A		Recomienda 10 películas dentro del género elegido
CU6-CP2	Escenario 2 / no se encuentran películas	N/A	N/A	N/A	N/A		Mensaje “No se encontraron películas”.
CU6-CP3	Escenario 3 / Cancelar	N/A	N/A	N/A	I		Mensaje “No se encontraron películas”. El usuario cancela
CU6-CP4	Escenario 4 / Cantidad recomendada	N/A	N/A	N/A	V	5	Recomienda 10 películas dentro del director elegido
CU6-CP5	Escenario 4 / Cantidad recomendada incorrecta	N/A	N/A	N/A	I	0	No recomienda películas y muestra mensaje “La cantidad de películas recomendadas debe ser mayor a 0”

CU7 - Recomendar Película Grupo

Id - Caso de prueba	Escenario / Condición	Genero	Actores	Idioma	Director	Cantidad recomendada	Resultado esperado
CU7-CP1	Escenario 1 / Recomendación por género	V	N/A	N/A	N/A		Recomienda 10 películas dentro del género elegido
CU7-CP2	Escenario 2 / no se encuentran películas	N/A	N/A	N/A	N/A		Mensaje “No se encontraron películas”.
CU7-CP3	Escenario 3 / Cancelar	N/A	N/A	N/A	I		Mensaje “No se encontraron películas”. El usuario cancela
CU7-CP4	Escenario 4 / Cantidad recomendada	N/A	N/A	N/A	V	5	Recomienda 10 películas dentro del director elegido

CU8 - Crear Evento

Id - Caso de prueba	Escenario / Condición	Título	Comentario	Fecha Inicio	Fecha Fin	Lista invitaciones	Resultado esperado
CU8-CP1	Escenario 1 / Crear un evento	V	V	V	V	V	Se crea el evento y se ingresan las invitaciones
CU8-CP2	Escenario 2 / Cancelar	I	I	I	I		No se crea el evento.
CU8-CP3	Escenario 3 / Sin invitaciones	V	V	V	V	I	Se crea el evento sin invitaciones

CU9 - Invitaciones Eventos

Id - Caso de prueba	Escenario / Condición	Lista invitaciones	Resultado esperado
CU9-CP1	Escenario 1 / Invitaciones	V	Ver invitaciones pendientes de aprobación
CU9-CP2	Escenario 2 / Sin Invitaciones	I	El sistema muestra un mensaje: "No hay invitaciones pendientes para el usuario"
CU9-CP3	Escenario 3 / Rechazar	V	La invitación queda rechazada y no se vuelve a mostrar.

CU10 - Recomendación Evento

Id - Caso de prueba	Escenario / Condición	Cartelera	Elegir	Voto	Resultado esperado
CU10CP1	Escenario 1 / Recomendación Eventos	V	V	V	Queda registrado que el usuario elige una película
CU10-CP2	Escenario 2 / Detalle película	V	V	V	El sistema muestra los datos de la película seleccionada
CU10-CP3	Escenario 3 / No elegir	V	I	N/A	Queda registrado que el usuario no elige esa película.

Ciclos funcionales

Eventos

- 1) Con usuario1 Crear evento, que invite a usuario2, usuario3 y usuario4
- 2) Con usuario2 y usuario3 Ver invitaciones y aceptar
- 3) Con usuario4 ver invitaciones y rechazar
- 4) Con usuario1 marcar 1 película como me gusta y 2 como no me gusta
- 5) Con usuario2 marcar 2 película como me gusta y 1 como no me gusta
- 6) Con usuario3 marcar 3 película como me gusta y 4 como no me gusta
- 7) Finalizar evento
- 8) Con usuario2 calificar
- 9) Con usuario3 ver eventos para calificar
- 10) Con usuario1 calificar
- 11) Con usuario2 ver si esta para calificar o como evento

Anexo H Guía de uso del sitio MovieRecommender

En esta guía se presentan los pasos a seguir, en distintas situaciones, al utilizar el sitio web MovieRecommender. No se pretende abarcar el total de funcionalidades del sitio sino aquellas que son de mayor relevancia.

H.1 Pasos a seguir

H.1.1 Cómo buscar una película para calificarla?

En este punto se explica los pasos a seguir para buscar una película, ver sus detalles y/o calificarla.

Primero que nada, ubicados en el Home del sitio, debemos seleccionar el icono de **Búsqueda Avanzada**, bastará con dar un click en los binoculares.

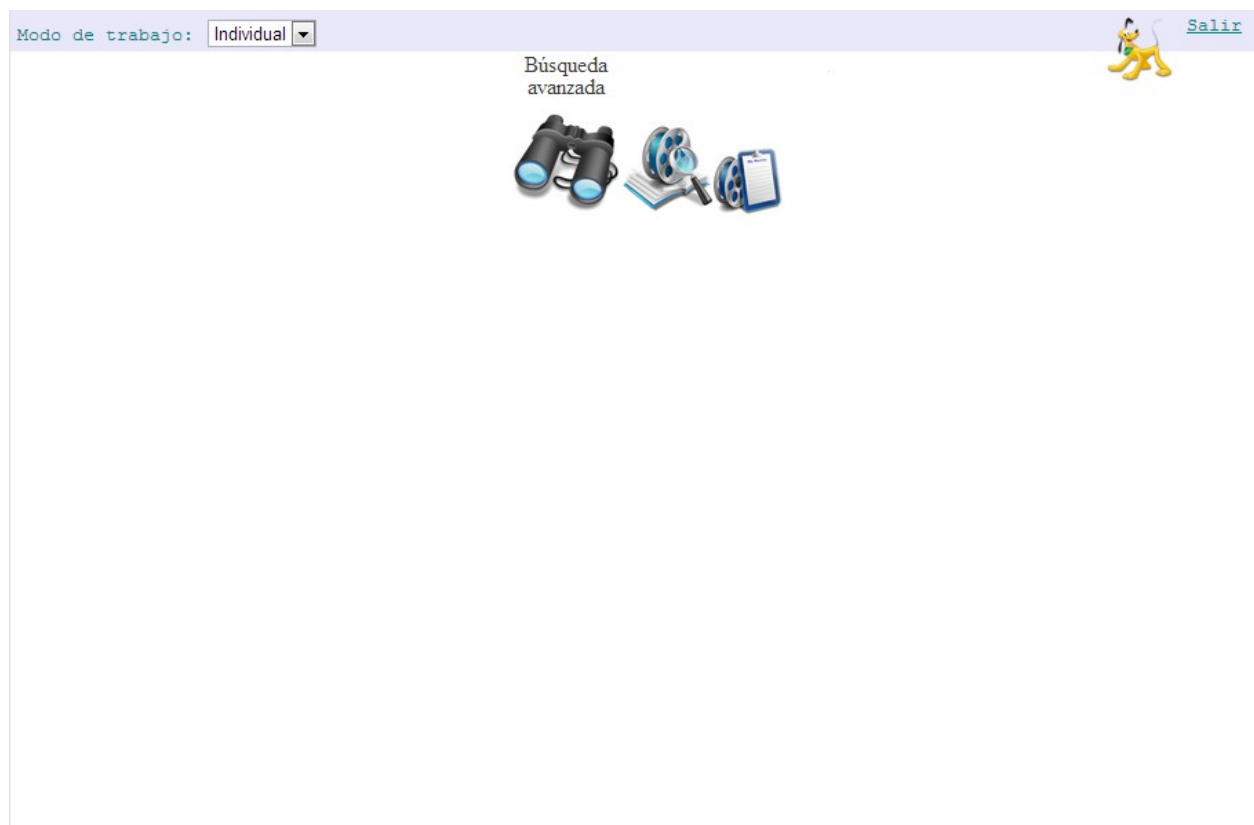


Ilustración 47: Home - entrar en Búsqueda Avanzada

Al entrar verán una página como se muestra en la ilustración 48.

En esta podrán utilizar los diversos filtros que se presentan para encontrar la película que deseen (tener presente que se cuenta con detalles de películas de hasta el año 2009).

A modo de ejemplo mostraremos los resultados y pasos a seguir si quisieramos buscar películas de *Predator* (*Depredador* su nombre en ingles).

Para esto comenzaremos buscando por título, ingresando "*Predator*" y damos click en el botón **Buscar**.



Ilustración 48: Página de Búsqueda Avanzada

Finalizada la búsqueda nos mostrará las películas que haya encontrado como se muestra en la ilustración 49. Como se puede apreciar, a los extremos de la página, en caso de haber muchas películas para los criterios de búsqueda, se cuenta con flechas que sirve para movernos entre las películas.

Cuando encontramos la película que nos interesa calificar basta con dar un click sobre la imagen de esta par poder ver los detalles y/o calificarla. En la ilustración 50 se muestra un ejemplo de haber seleccionado la película *Predator* (que protagoniza Schwarzenegger).

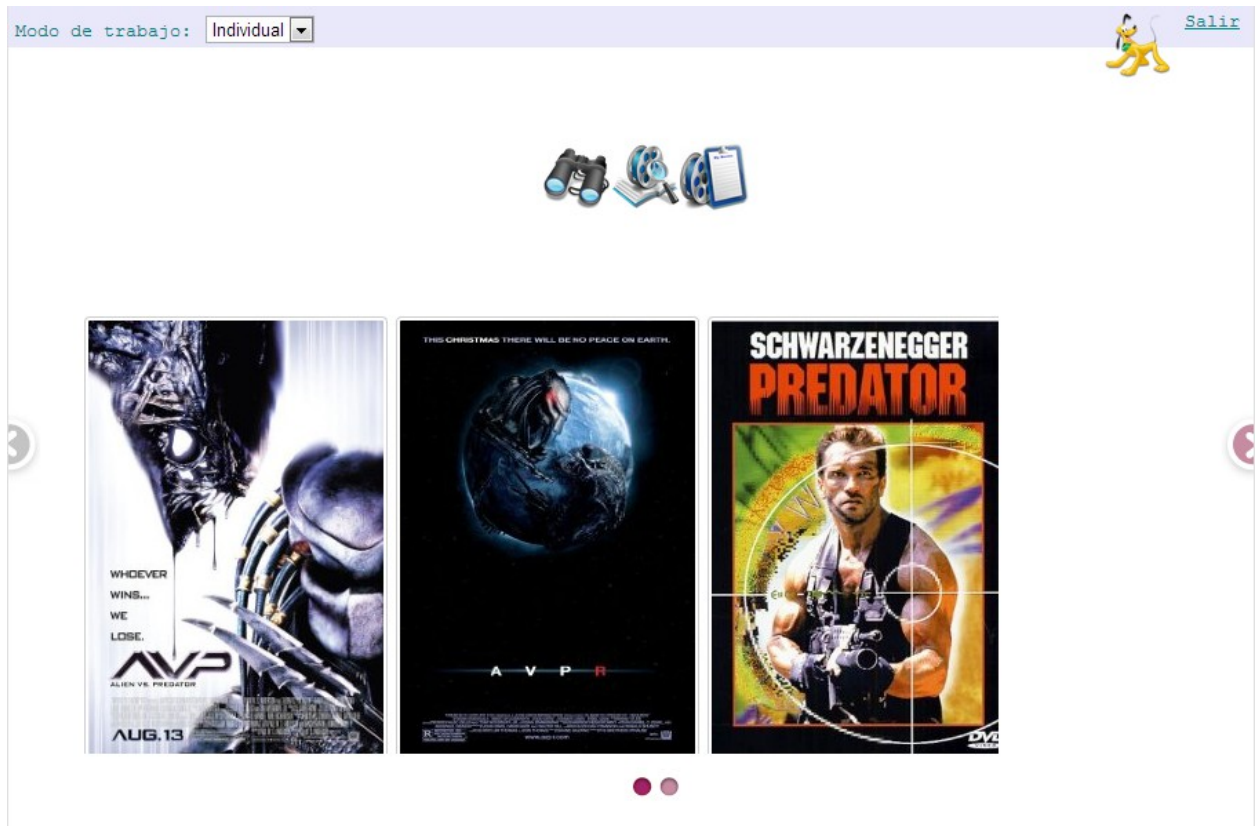


Ilustración 49: Resultado - página 1/2 - búsqueda de Predator



Ilustración 50: Detalles de película Predator

Arriba y debajo del título de la película "*Predator (1987)*" pueden verse 5 estrellas en blanco. Estas estrellas son las que utilizamos para calificar la película con valores de 0.5 (no me gustó), 1, 1.5, 2, y así sucesivamente hasta 5 puntos de máximo (me encantó). Al hacerlo, la próxima vez que entremos a ver los detalles de la película veremos algo similar a la ilustración 51 en donde se ha votado la película un valor de 3.5 (bueno).

Listo!



Ilustración 51: Calificación de película Predator

Hasta aquí hemos logrado buscar y calificar una película.

H.1.2 Como pedir una predicción para cierta película?

Supongamos que nos interesa saber si una determinada película nos gustará o si simplemente queremos corroborar con qué calificación la puntuaríamos en base a nuestro gusto, es decir en base a las películas que hemos previamente calificado.

Para esto en el Home debemos ingresar a la sección **Predicción Individual** como se muestra en la ilustración 52.

Una vez dentro veremos un campo donde introducir palabras del título de la película. Es decir, podremos buscarla por su nombre como se muestra en la ilustración 53 donde se selecciona la película "Predator 2".

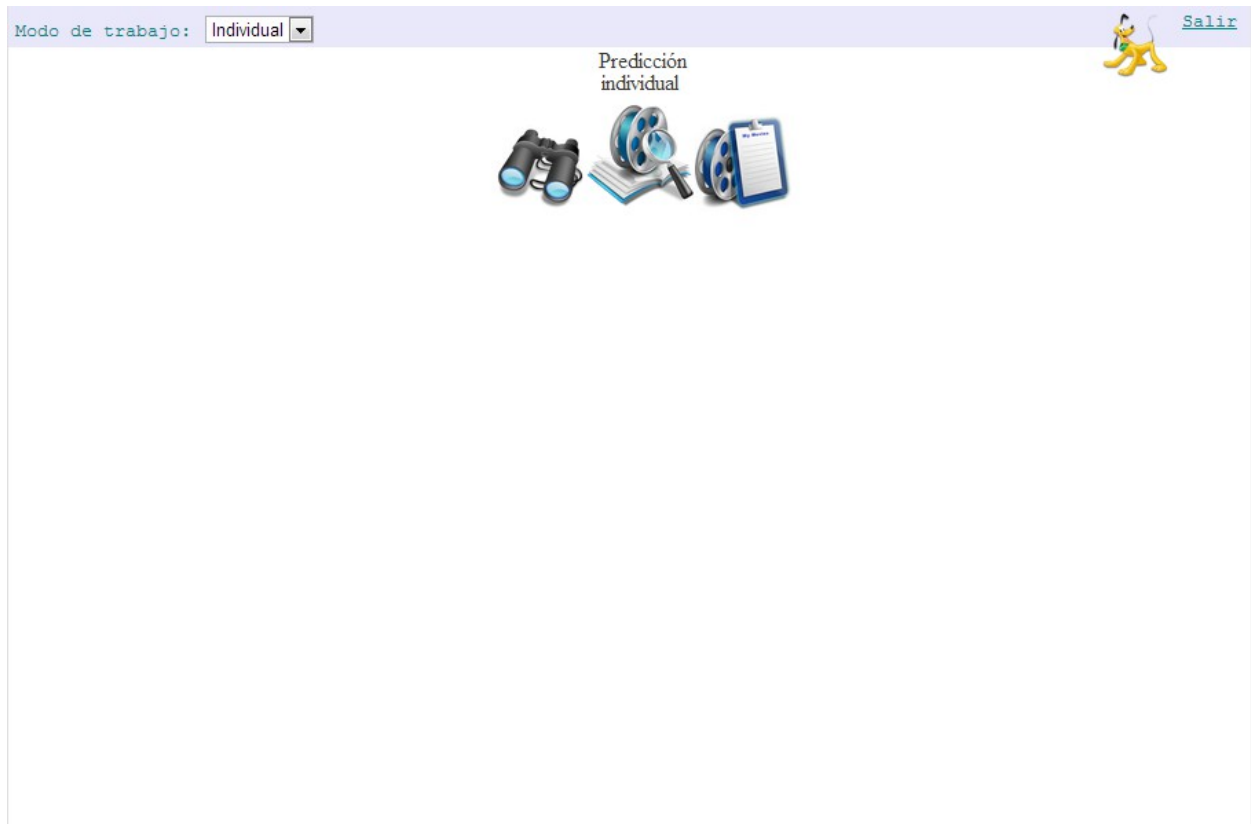


Ilustración 52: Home - entrar en Predicción Individual

Al hacerlo y luego de un breve instante, se mostrará en el redondel anaranjado el valor de predicción, es decir, la calificación que el sistema calcula que le otorgaríamos según nuestras calificaciones previas (ver ilustración 54).

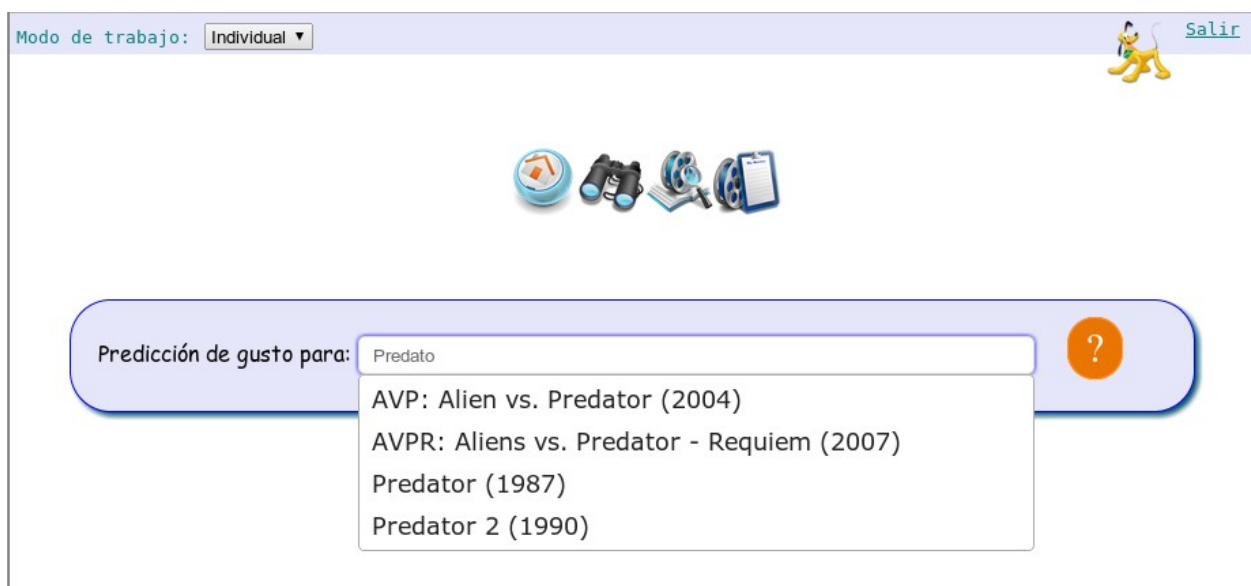


Ilustración 53: Predicción Individual - Selección película "Predator 2"



Illustration 54: Predicción Individual - Predicción película "Predator 2"

Pronto, así hemos podido consultar la calificación que predijo el sistema.

H.1.3 Cómo pedir una recomendación sobre películas que me interesan?

En este caso nos interesa indicarle varias películas al sistema y que este nos recomiende cuales nos gustaría ver.

Para esto vamos desde el Home a la sección **Recomendación Individual** como se muestra en la ilustración 55.

Una vez ahí veremos una lista con las películas donde para cada película podremos marcar el checkbox si nos interesa que la incluya al momento de recomendarnos. Por defecto vienen todas seleccionadas. Si deseamos solo seleccionar unas pocas, podemos des-seleccionarlas todas al desmarcar el checkbox *Seleccionar todas* y luego movernos en la lista de películas o utilizar la opción *Filtrar catálogo* para ir marcando las que son de nuestro interés.

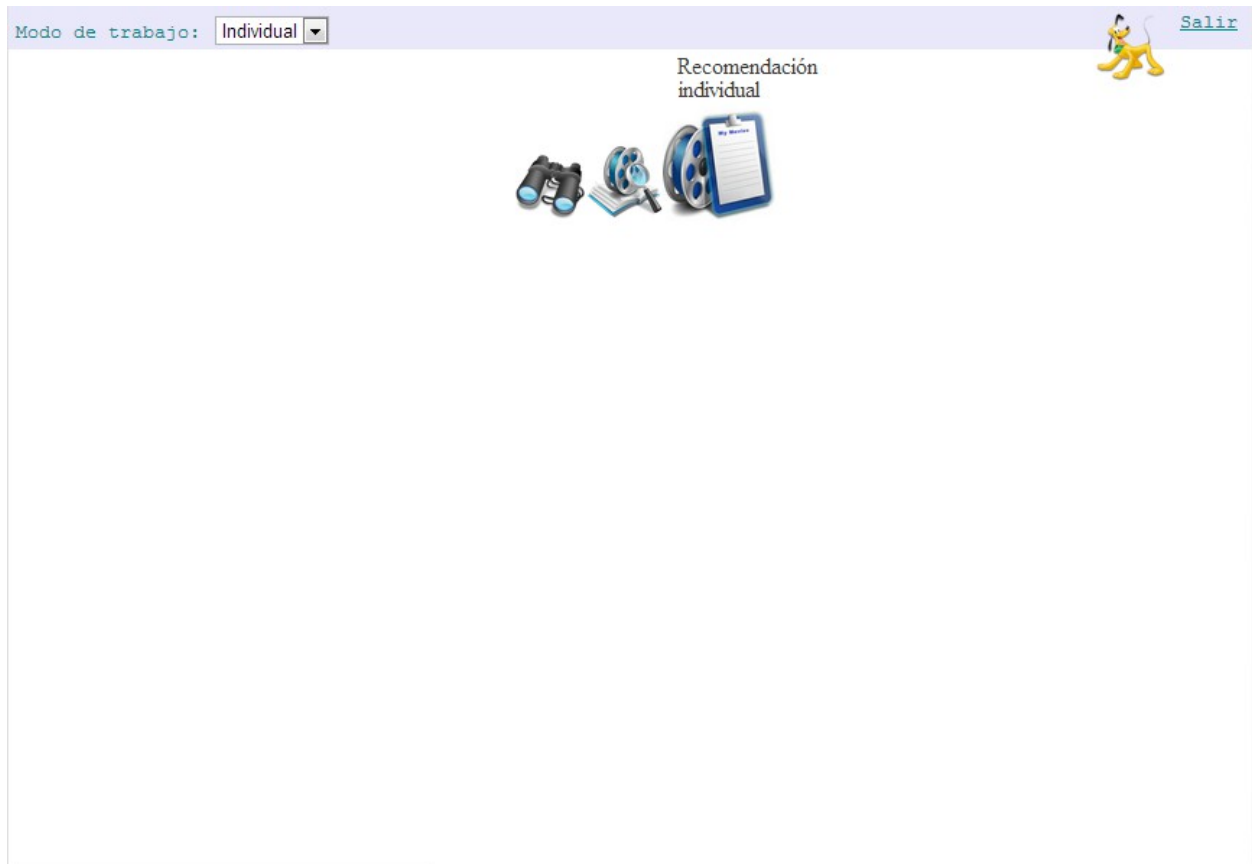


Ilustración 55: Home - entrar en Recomendación Individual

En caso de seleccionar el filtro avanzado, se mostrarán las mismas opciones de filtro que se utilizan en la sección “*Búsqueda Avanzada*” como se ve en la ilustración 56.

En el Filtro de catálogo podemos, por ejemplo, ingresar el título “Predator” y luego dar en el botón *Aplicar filtro* para que solos nos muestre las películas que cumplen con el filtro como se aprecia en la ilustración 57. Ahí podremos seleccionar las películas que deseemos y luego volver a aplicar el filtro para seleccionar otras películas.

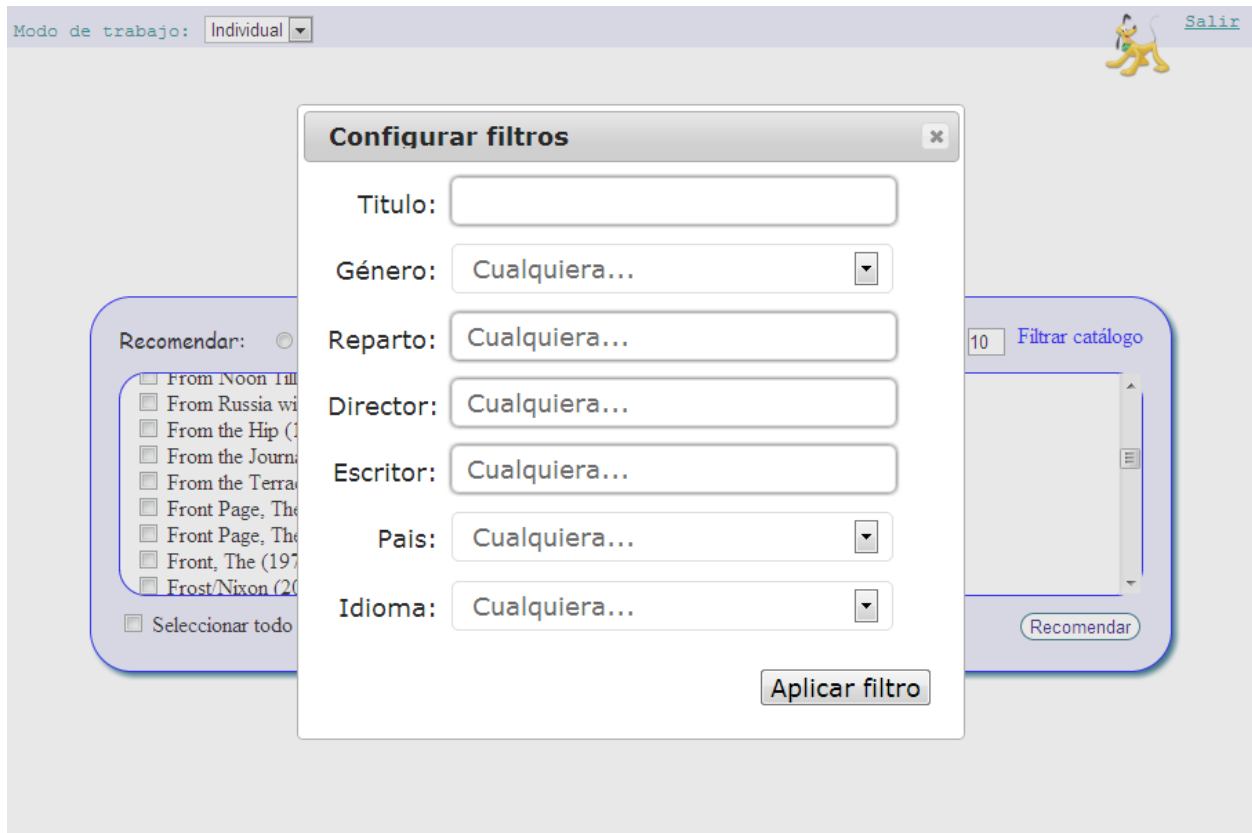


Ilustración 56: Recomendación Individual - Filtro catálogo

Cuando estemos listos y deseemos obtener la recomendación sobre estas películas debemos dar click sobre el botón *Recomendar*.

En la ilustración 11 se muestra el resultado pedir la recomendación para las películas que se seleccionan hasta este punto (ver ilustración 57). Se puede apreciar que en la esquina superior izquierda de cada película se adjunta una estrella, en esta figura el valor que le asignó el sistema según nuestro gusto. Este listado de películas se encuentra ordenado de la película con mejor recomendación primero y luego descende hasta la que obtuvo el peor valor.

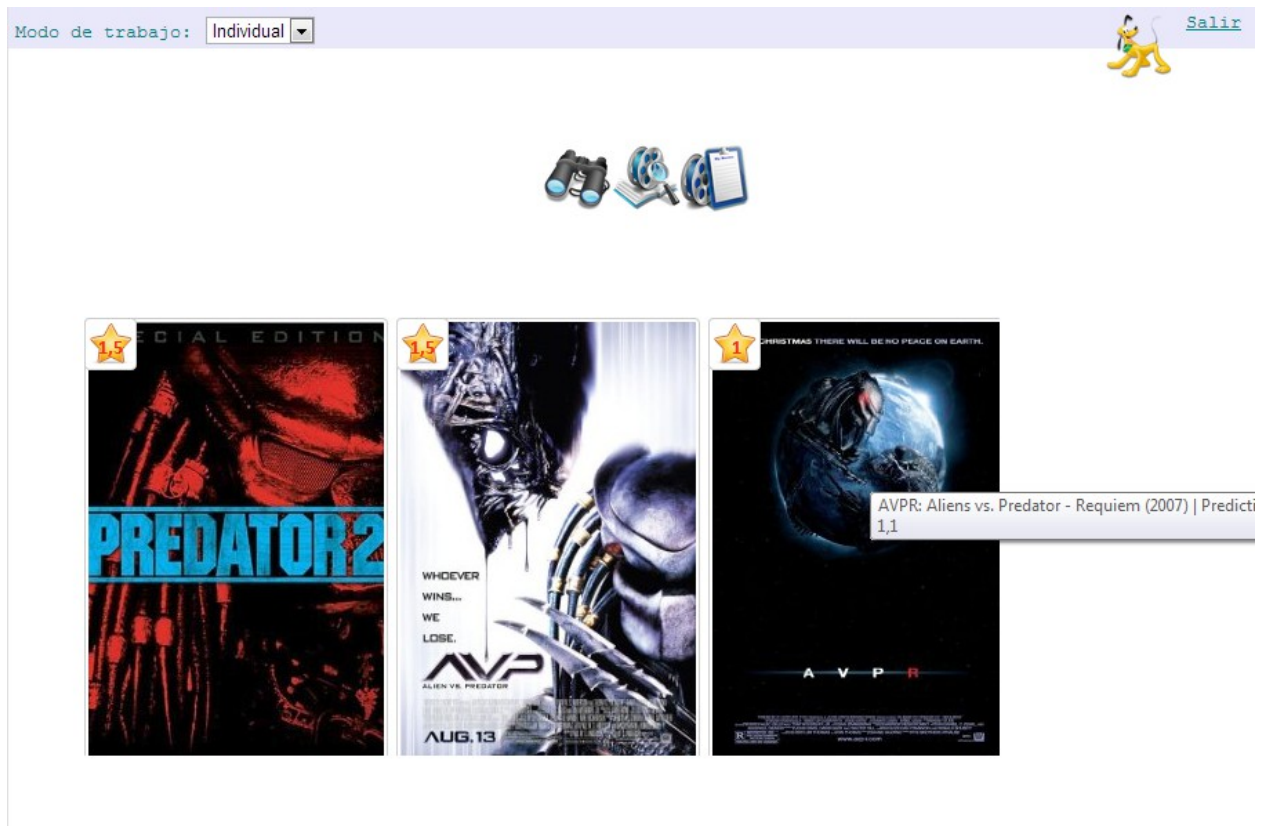


Ilustración 58: Recomendación Individual - Resultado de recomendación

Hasta este punto hemos obtenido la recomendación para ciertas películas de nuestro interés en base a las calificaciones previas y al gusto popular de los demás usuarios.